Interpolation and Quadrature Methods
for Ordinary Differential Equations

By G. J. Cooper*

Summary. A class of single-step methods is constructed for solving systems of differential equations. These methods are based on the use of interpolation and quadrature formulae, and are related to some implicit methods of Runge-Kutta type. Weight functions may be used to cope with difficult behaviour.

1. Introduction. This article is a generalization in various ways of the use of quadrature formulae to integrate differential equations. The principle is well known and various special methods are available [1]. The main problem is to obtain adequate approximations to the function values at the chosen abscissae. In this respect linear differential equations are particularly amenable to treatment [2]. In the general case there appear to be a number of ways to obtain these approximations and one such approach is considered here.

In some recent articles [3], [4], single-step methods of Runge-Kutta type were investigated, and it was pointed out that there is a connection with quadrature methods. These papers are, in some respects, an extension of investigations by Butcher [5], [6], who likewise obtained methods related to the use of quadrature formulae [6], [7]. It seems natural to attempt to reverse the analysis. In so doing, useful additional results are obtained and new methods become available.

Thus we consider a system of nonlinear differential equations of arbitrary orders,

$$(1.1) \qquad y_r^{(n_r)}(t) = f_r(t; \mathbf{y}(t)) \equiv f_r(t; \{y_p^{(n_p-m)}(t)\}) , \qquad r = 1(1)q ,$$

where

$$\mathbf{y}(t) = \{y_p^{(n_p-m)}(t)\} = (y_1^{(0)}(t), \cdots, y_1^{(n_1-1)}(t); \cdots; y_q^{(0)}(t), \cdots, y_q^{(n_q-1)}(t)) ,$$

is a point in the real Euclidean N-space, R_N,

$$N = \sum_{r=1}^{q} n_r .$$

Initial values, $\mathbf{y}(x)$, are given for some value, x, of the real variable t, and approximations, $\bar{\mathbf{y}}(x + h)$, to the values $\mathbf{y}(x + h)$, are required for some given step length h.

We are concerned with a set of s abscissae, μ_i, $i = 1(1)s$, and define $\mu_0 = 0$, $\mu_{s+1} = 1$. For a given step length h, these abscissae define a closed interval (a, b),

$$a = \min_{i=0(1)s+1} \{x_i\} , \qquad b = \max_{i=0(1)s+1} \{x_i\} ,$$

$$x_i = x + \mu_i h , \qquad i = 0(1)s + 1 .$$

* Present address: Department of Computer Science, University of Edinburgh.

It is assumed that for some nonnegative integer $\mathbf{p}$ the derivatives, $y_r^{(n_r+\mathbf{p})}(t)$, $r = 1(1)q$, are continuous for $t \in (a, b)$. Let $\mathbf{z}$ be a point in R_N,

$$\mathbf{z} = \{z_\rho^{[m]}\} ,$$

where the element $z_\rho^{[m]}$ corresponds with $y_\rho^{(n\rho-m)}(t)$ of $\mathbf{y}(t)$, and D some convex domain of R_N containing the solution $\mathbf{y}(t)$, $t \in (a, b)$. Let the functions be real-valued mappings of R_{N+1} onto R_1 satisfying the Lipschitz conditions,

$$|f_r(t; \mathbf{z}) - f_r(t; \mathbf{w})| \leqq L \max_{\rho,m} |z_\rho^{[m]} - w_\rho^{[m]}| ,$$

(1.2)
$$r, \rho = 1(1)q , m = 1(1)n_\rho ,$$

for $\mathbf{z}, \mathbf{w} \in D$, $t \in (a, b)$. For a suitable domain D these various conditions ensure that the initial-value problem has a unique solution such that the derivatives

$$y_r^{(\nu)}(t) , \qquad r = 1(1)q , \qquad \nu = 0(1)n_r + \mathbf{p} ,$$

are continuous in (a, b) [8, pp. 71–2].

2. Quadrature Methods. Define

$$T_r^{[\nu]}(\mu h) = \sum_{\tau=0}^{\nu-1} \frac{(\mu h)^\tau}{\tau!} y_r^{(n_r-\nu+\tau)}(x) , \qquad r = 1(1)q , \qquad \nu = 1(1)n_r ,$$

where μ lies in the closed interval I bounded by the greatest and least of μ_i, $i = 0(1)s + 1$. We introduce weight functions $w_r(t)$, and then integration by parts gives

$$y_r^{(n_r-\nu)}(x + h) = T_r^{[\nu]}(h) + \frac{h^\nu}{(\nu - 1)!} \int_0^1 F_r(x + \mu h)(1 - \mu)^{\nu-1} w_r(x + \mu h) d\mu ,$$

(2.1)
$$F_r(t) = w_r^{-1}(t) y_r^{(n_r)}(t) ,$$

for $r = 1(1)q$, $\nu = 1(1)n_r$. In a quadrature method the integrals are replaced by weighted sums,

(2.2)
$$\tilde{y}_r^{(n_r-\nu)}(x + h) = T_r^{[\nu]}(h) + \frac{h^\nu}{\nu!} \sum_{i=1}^s \alpha_{ri}^{[\nu]} F_r(x_i) ,$$

where the μ_i, $i = 1(1)s$, are selected abscissae, and the weights may be chosen so that the quadrature is exact if the functions, $F_r(x + \mu h)$, are polynomials in μ of degree less than some given integer. Here we restrict attention to quadratures of interpolatory type and a single set of distinct abscissae. Even so, there are a number of approaches, of which two are considered. We could, however, use other integration rules such as the trapezoidal rule or methods based on the integration of rational functions.

Let $l(\mu)$ be the polynomial of degree s,

$$l(\mu) = \prod_{i=1}^s (\mu - \mu_i) ,$$

and $l_i(\mu)$ the unique polynomials of degree $s - 1$, such that

$$\begin{aligned} l_i(\mu_j) = \delta_{ij} &= 1, \qquad i = j, \\ &= 0, \qquad i \neq j, \end{aligned} \qquad i, j = 1(1)s .$$

Now we can interpolate either set of functions,

(i) $F_r(x + \mu h)$, $r = 1(1)q$,

(ii) $(1 - \mu)^{\nu-1}F_r(x + \mu h)$, $r = 1(1)q$, $\nu = 1(1)n_r$.

In the first case we have [9, pp. 164–5]

$$\alpha_{ri}^{[\nu]} = \nu \int_0^1 l_i(\mu)(1 - \mu)^{\nu-1}w_r(x + \mu h)d\mu ,$$

(2.3) $\quad y_r^{(n_r-\nu)}(x + h) - \tilde{y}_r^{(n_r-\nu)}(x + h)$

$$= \frac{h^{s+\nu}}{s!(\nu - 1)!} \int_0^1 l(\mu)(1 - \mu)^{\nu-1}\left[\frac{d^s}{dt^s} F_r(t)\right]_{t=\xi'(\mu)} w_r(x + \mu h)d\mu ,$$

for $r = 1(1)q$, $\nu = 1(1)n_r$, and $\xi'(\mu) \in (a, b)$ and depends also on x and h. Similarly, in the second case

$$\alpha_{ri}^{[\nu]} = \nu(1 - \mu_i)^{\nu-1} \int_0^1 l_i(\mu)w_r(x + \mu h)d\mu ,$$

(2.4) $\quad y_r^{(n_r-\nu)}(x + h) - \tilde{y}_r^{(n_r-\nu)}(x + h)$

$$= \frac{h^{s+1}}{s!(\nu - 1)!} \int_0^1 l(\mu)\left[\frac{d^s}{dt^s} (x + h - t)^{\nu-1}F_r(t)\right]_{t=\xi''(\mu)} w_r(x + \mu h)d\mu .$$

The second relation (2.4) seems to be particularly suitable for computation, since for fixed r, i, the parameters are given for all ν in terms of $\alpha_{ri}^{[1]}$. On the other hand, the error term in the first case is of higher order in h if $\nu > 1$. In general, the parameters depend on x, h, and r, and it is conventional to choose a unit weight function, eliminating this dependence.

If a weight function, $w(t)$, nonnegative in the closed interval $(x, x + h)$ and independent of r, is chosen, the error terms can be improved. Select as abscissae, the zeros of the associated orthogonal polynomial of degree s defined on $(0, 1)$. Then for example, in the second case [10, p. 152]

$$y_r^{(n_r-\nu)}(x + h) - \tilde{y}_r^{(n_r-\nu)}(x + h) = \frac{h^{2s+1}}{(2s)!(\nu - 1)!}$$

$$\times \left[\frac{d^{2s}}{dt^{2s}} \{(x + h - t)^{\nu-1}F_r(t)\}\right]_{t=\xi} \int_0^1 l^2(\mu)w(x + \mu h)d\mu ,$$

$\xi \in (a, b)$. Here we also have

$$\alpha_{ri}^{[\nu]} = \nu(1 - \mu_i)^{\nu-1}\lambda_i , \qquad r = 1(1)q , \qquad \nu = 1(1)n_r , \qquad i = 1(1)s ,$$

where the λ_i are the corresponding Christoffel numbers [11, pp. 47–8]. For (i) an analysis shows that the error terms now remain of $O(h^{2s+1})$.

The various error terms given depend on the existence of the integrals and the continuity of the derivatives involved. It is possible to relax the continuity requirements [12, pp. 288–92].

Choose a unit weight function, and as abscissae the zeros of the Legendre polynomial of degree s, $P_s(2\mu - 1)$. Then the quadrature formulae can be made exact for given ν, for $y_r^{(n_r)}(x + \mu h)$ a polynomial in μ of degree less than $2s + 1 - \nu$. This

is the best we can do (in this sense) if all the terms, $y_r^{(n_r-\nu)}(x + h)$, are to be determined. On the other hand, if we only want to compute $y_r^{(0)}(x + h)$, this can be improved. We use a quadrature formula based on the orthogonal polynomials associated with the weight function $(1 - \mu)^{n_r-1}$ and the interval $(0, 1)$. The quadrature is then exact for polynomials of degree less than $2s$, though if the n_r differ we require different quadrature formulae to achieve this. These results were also obtained in [3], [4]. Similar remarks apply to alternative weight functions, $w(t) \neq 1$, but we are now concerned with the behaviour of $F_r(t)$ rather than $y_r^{(n_r)}(t)$. Thus we can choose weight functions which will specifically cope with known difficult behaviour of the functions $f_r(t; \mathbf{y}(t))$.

Before quadratures of the form (2.2) (or other integration rules) can be applied, approximations,

$$(2.5) \qquad \bar{y}_r^{(n_r)}(x_i) = y_r^{(n_r)}(x_i) - \epsilon_r^{[0]}(x_i) , \qquad r = 1(1)q , \qquad i = 1(1)s ,$$

must be obtained. These values can be determined from the differential equations (1.1) provided adequate estimates,

$$(2.6) \qquad \begin{aligned} \bar{y}_r^{(n_r-\nu)}(x_i) &= y_r^{(n_r-\nu)}(x_i) - \epsilon_r^{[\nu]}(x_i) , \\ r &= 1(1)q , \qquad \nu = 1(1)n_r , \qquad i = 1(1)s , \end{aligned}$$

are known. Such estimates are, however, not usually known initially and this approximation problem is fundamental in the application of quadrature methods.

The approximations may be obtained by integration, using previously integrated points and some suitable integration rule. Alternatively, an extrapolation procedure could be applied using these points. Such methods are not, strictly, of single-step type. On the other hand, an explicit Runge-Kutta method may be used to give these approximations [1] but for high-order methods this requires explicit methods of high order, few of which are available. We discuss instead an alternative approach, and attempt to define the approximations (2.5) implicitly. Thus consider $\bar{y}_r^{(n_r)}(x + \mu h)$ as functions of μ, uniquely determined (in some specified way) by the s values at μ_i, $i = 1(1)s$. Then the estimates (2.6) can be obtained by integration in a way similar to (2.1), and the differential equations (1.1) now define the values (2.5) implicitly. The determination of the functions $\bar{y}_r^{(n_r)}(x + \mu h)$, and the choice of quadrature formula, characterize the method. The obvious way to determine the functions is to use the Lagrange interpolation formula, but we can also use other techniques such as rational function approximation and Hermitian interpolation.

3. Lagrange Interpolation. Integration by parts gives

$$y_r^{(n_r-\nu)}(x_j) = T_r^{[\nu]}(\mu_j h) + \frac{h^\nu}{(\nu - 1)!} \int_0^{\mu_j} (\mu_j - \mu)^{\nu-1} F_r(x + \mu h) w_r(x + \mu h) d\mu ,$$

for $r = 1(1)q$, $\nu = 1(1)n_r$, $j = 1(1)s + 1$. We now interpolate either $F_r(x + \mu h)$ or $(\mu_j - \mu)^{\nu-1} F_r(x + \mu h)$ and integrate the interpolation polynomial. In the first case we obtain

$$(3.1) \qquad y_r^{(n_r-\nu)}(x_j) = T_r^{[\nu]}(\mu_j h) + \frac{(\mu_j h)^\nu}{\nu!} \sum_{i=1}^s \lambda_{rji}^{[\nu]} F_r(x_i) + R_r^{[\nu]}(\mu_j h) ,$$

for $r = 1(1)q$, $\nu = 1(1)n_r$, $j = 1(1)s$, where

$$\lambda_{rji}^{[\nu]} = \nu \int_0^1 l_i(\mu_j\mu)(1 - \mu)^{\nu-1}w_r(x + \mu_j\mu h)d\mu \,,$$

(3.2)
$$R_r^{[\nu]}(\mu_jh) = \frac{h^{s+\nu}}{s!(\nu - 1)!} \int_0^{\mu_j} l(\mu)(\mu_j - \mu)^{\nu-1}\left[\frac{d^s}{dt^s} F_r(t)\right]_{t=\zeta'(\mu)} w_r(x + \mu h)d\mu \,.$$

From (3.2) we obtain

(3.3)
$$\sum_{i=1}^s \mu_i^\tau \lambda_{rji}^{[\nu]} = \nu\mu_j^\tau \int_0^1 \mu^\tau(1 - \mu)^{\nu-1}w_r(x + \mu_j\mu h)d\mu \,,$$

for $j = 1(1)s$, $\tau = 0(1)s - 1$, and these equations provide an alternative definition for the $\lambda_{rji}^{[\nu]}$. For a unit weight function they correspond with equations obtained by Butcher [6] for a system of first-order differential equations and by the author [4] for more general systems. Thus there is a close connection with implicit Runge-Kutta methods.

In the second case (3.1) holds with

(3.4)
$$\lambda_{rji}^{[\nu]} = \nu\left(1 - \frac{\mu_i}{\mu_j}\right)^{\nu-1} \int_0^1 l_i(\mu_j\mu)w_r(x + \mu_j\mu h)d\mu \,,$$

provided $\mu_j \neq 0$ (and if $\mu_j = 0$ we may take $\lambda_{rji}^{[\nu]} = 0$ for all r, ν, i),

$$R_r^{[\nu]}(\mu_jh) = \frac{h^{s+1}}{s!(\nu - 1)!} \int_0^{\mu_j} l(\mu)\left[\frac{d^s}{dt^s} \{(x + h - t)^{\nu-1}F_r(t)\}\right]_{t=\zeta''(\mu)} w_r(x + \mu h)d\mu \,.$$

Again, the second method seems more suitable for computation in that

(3.5)
$$\lambda_{rji}^{[\nu+1]} = (1 + 1/\nu)(1 - (\mu_i/\mu_j))\lambda_{rji}^{[\nu]} \,, \qquad \nu = 1(1)n_r - 1 \,,$$

but the first method provides error terms of higher order in h for $\nu > 1$.

Now define the approximate values (2.5) via the differential Eqs. (1.1),

(3.6) $\quad \bar{y}_r^{(n_r)}(x_j) = f_r\left(x_j; \left\{T_\rho^{[m]}(\mu_jh) + \frac{(\mu_jh)^m}{m!} \sum_{i=1}^s \lambda_{\rho ji}^{[m]}w_\rho^{-1}(x_i)\bar{y}_\rho^{(n_\rho)}(x_i)\right\}\right),$

for $j = 1(1)s$, $r = 1(1)q$. Thus, in general, these values are defined implicitly and have to be determined iteratively. The differential equations (1.1), the Lipschitz conditions (1.2), and (2.5), (3.1) give

$$\left|\epsilon_r^{[0]}(x_j)\right| \leq L \max_{\rho,m}\left|\frac{(\mu_jh)^m}{m!} \sum_{i=1}^s \lambda_{\rho ji}^{[m]}w_\rho^{-1}(x_i)\epsilon_\rho^{[0]}(x_i) + R_\rho^{[m]}(\mu_jh)\right|,$$

$r, \rho = 1(1)q$, $m = 1(1)n_\rho$, $j = 1(1)s$. If we define

$$\epsilon = \max_{r,j} |\epsilon_r^{[0]}(x_j)| \,, \quad \lambda = \max_{r,\nu,j}\sum_{i=1}^s\left|h^{\nu-1}\frac{\mu_j^\nu}{\nu!}\lambda_{rji}^{[\nu]}w_r^{-1}(x_i)\right|,$$

for $r = 1(1)q$, $\nu = 1(1)n_r$, $j = 1(1)s$, then we have

$$\epsilon(1 - L\lambda|h|) \leq L \max_{r,\nu,j} |R_r^{[\nu]}(\mu_jh)| \,,$$

and hence at least $\epsilon = O(h^{s+1})$. Much the same type of analysis shows that (3.6) has a unique solution for sufficiently small h, which may be determined iteratively [6].

If we use a quadrature of interpolatory type, the approximations determined

from (3.6) may be substituted into (2.2) to give

(3.7)
$$\bar{y}_r^{(n_r-\nu)}(x+h) = T_r^{[\nu]}(h) + \frac{h^\nu}{\nu!} \sum_{i=1}^{s} \alpha_{ri}^{[\nu]} w_r^{-1}(x_i)\bar{y}_r^{(n_r)}(x_i) ,$$

$$y_r^{(n_r-\nu)}(x+h) - \bar{y}_r^{(n_r-\nu)}(x+h) = O(h^{s+1}) .$$

Indeed, if we use a method with parameters defined by (2.3) and (3.2), then if $L\lambda|h| < \frac{1}{2}$,

$$|y_r^{(n_r-\nu)}(x+h) - \bar{y}_r^{(n_r-\nu)}(x+h)|$$

$$\leqq \frac{|h|^{s+\nu}}{s!(\nu-1)!} F(1 + 2s\alpha L|h|e^{|h|}) \max_{j,\nu} \int_0^{\mu_j} |l(\mu)(\mu_j - \mu)^{\nu-1} w_r(x+\mu h)| d\mu ,$$

$r = 1(1)q$, $\nu = 1(1)n_r$, $j = 1(1)s + 1$, where

$$\alpha = \max_{r,i,\nu} |\alpha_{ri}^{[\nu]} w_r^{-1}(x_i)| , \qquad F = \max_{r,t} |F_r^{(s)}(t)| , \qquad t \in (a,b) .$$

The analysis is not sufficiently refined to show if we can do better than this by suitably choosing the abscissae. This is indeed the case (for a unit weight function) for a system of first-order equations [6] and special cases exist for higher-order equations [3]. For a unit weight function the parameters are independent of x and h, and we can alternatively regard Eqs. (3.6) as defining an implicit Runge-Kutta type process [4].

The methods are particularly suitable for systems of linear equations. For then Eqs. (3.6) give a set of linear equations for the approximate values.

A simple example which illustrates some of these remarks is the first-order non-linear differential equation

$$2y_1^{(1)}(t) = \sqrt{t} + \sqrt{(y_1(t))} , \quad \text{with} \quad y_1(0) = 0 .$$

The Lipschitz condition does not hold in the vicinity of the origin, but this is not required for the numerical solution. Further, there is a solution such that for small t, $y_1(t) = O(t^{3/2})$ [13]. This suggests the use of a weight function $w(t) = t^{1/2}$. Thus for the *first* step away from the origin ($x = 0$) we can use quadrature formulae based on the zeros of the Jacobi polynomial $P_s^{(0,1/2)}(2\mu - 1)$ [11, pp. 58–98]. With $s = 2$ the zeros are

$$126\mu_1 = 70 - 4\sqrt{70} , \qquad 126\mu_2 = 70 + 4\sqrt{70} ,$$

and the Christoffel numbers are

$$\alpha_{11}^{[1]} = \lambda_1 = \sqrt{h}\,\frac{50 - \sqrt{70}}{150} , \qquad \alpha_{12}^{[1]} = \lambda_2 = \sqrt{h}\,\frac{50 + \sqrt{70}}{150} .$$

From (3.2) or (3.4) we obtain

$$\lambda_{111}^{[1]} = \sqrt{(\mu_1 h)}\,\frac{16 + \sqrt{70}}{30} , \qquad \lambda_{122}^{[1]} = \sqrt{(\mu_2 h)}\,\frac{16 - \sqrt{70}}{30} ,$$

$$\lambda_{112}^{[1]} = \sqrt{(\mu_1 h)}\,\frac{4 - \sqrt{70}}{30} , \qquad \lambda_{121}^{[1]} = \sqrt{(\mu_2 h)}\,\frac{4 + \sqrt{70}}{30} .$$

Eqs. (3.6) give

$$2\overline{y}_1^{(1)}(x_j) = \sqrt{(\mu_j h)} + [\mu_j h \{\lambda_{1j1}^{[1]} w^{-1}(x_1)\overline{y}_1^{(1)}(x_1) + \lambda_{1j2}^{[1]} w^{-1}(x_2)\overline{y}_1^{(1)}(x_2)\}]^{1/2} ,$$

$$j = 1, 2 .$$

Define $F_j = w^{-1}(x_j)\overline{y}_1^{(1)}(x_j)$, $j = 1, 2$, and then

$$4F_1^2 - \frac{F_1}{30}[120 + (\mu_1 h)^{1/2}(16 + \sqrt{70})] + 1 = \sqrt{(\mu_1 h)}\frac{4 - \sqrt{70}}{30} F_2 ,$$

$$4F_2^2 - \frac{F_2}{30}[120 + (\mu_2 h)^{1/2}(16 - \sqrt{70})] + 1 = \sqrt{(\mu_2 h)}\frac{4 + \sqrt{70}}{30} F_1 .$$

With $h = 0.25$ we obtain, after a few iterations

$$F_1 = 0.67204\ 4902 , \qquad F_2 = 0.72888\ 3492 , \qquad \overline{y}_1(0.25) = 0.05876\ 8305 .$$

We can compare this result with some obtained using the Heun method [10, p. 193]. With 20 steps of length 0.0125, the Heun method gives an appreciably poorer result. With $h = 0.0625$, our two-stage quadrature method gives, in one step,

$$\overline{y}_1(0.0625) = 0.006653\ 7117 .$$

The first eight terms of the series expansion about zero give 0.00664 2000, and the quadrature result appears almost as good as that given by the Heun method using 50 steps, 0.006631 5805 [10, p. 193].

4. Further Remarks. The establishment of quadrature methods for arbitrary x, h, and general classes of weight functions involves considerable algebra. Even if the abscissae are fixed arbitrarily, integrals of the form (2.4), (3.2) have to be evaluated, and it is probably most convenient to do this on a computer. There seems to be little difficulty in devising a program to automate the process, provided integrals of the form

$$\int_0^1 \mu^r w(x + \mu h)d\mu ,$$

can be determined.

Convergence of the methods has not been discussed. However, a proof given previously [4] carries over with only trivial changes.

If the derivatives $y_r^{(n_r+1)}(t)$ are available in terms of $y_r^{(n_r)}(t)$ and known functions, then Hermitian interpolation can be used. Alternatively we may interpolate $y_r^{(n_r-1)}(t)$, using also $y_r^{(n_r)}(t)$. The same type of analysis holds and error terms $O(h^{2s+1})$ are obtained. Rational approximation methods could also be treated in a similar fashion.

Institute of Computer Science
London, England

1. L. STOLLER & D. MORRISON, "A method for the numerical integration of ordinary differential equations," *MTAC*, v. 12, 1958, pp. 269–272. MR **21** #974.

2. J. T. DAY, "A one-step method for the numerical integration of the differential equation $y'' = f(x)y + g(x)$," *Comput. J.*, v. 7, 1965, pp. 314–317. MR **30** #4385.

3. G. J. COOPER & E. GAL, "Single step methods for linear differential equations," *Numer. Math.* (To appear.)

4. G. J. COOPER, "A class of single-step methods for systems of nonlinear differential equations," *Math. Comp.*, v. 21, 1967, pp. 597–610.

5. J. C. BUTCHER, "Coefficients for the study of Runge-Kutta integration processes," *J. Austral. Math. Soc.*, v. 3, 1963, pp. 185–201. MR **27** #2109.

6. J. C. BUTCHER, "Implicit Runge-Kutta processes," *Math. Comp.*, v. 18, 1964, pp. 50–64. MR **28** #2641.

7. J. C. BUTCHER, "Integration processes based on Radau quadrature formulas," *Math. Comp.*, v. 18, 1964, pp. 233–244. MR **29** #2973.

8. E. L. INCE, *Ordinary Differential Equations*, Dover, New York, 1956.

9. L. M. MILNE-THOMSON, *The Calculus of Finite Differences*, Macmillan, New York, 1960.

10. J. TODD (EDITOR), *A Survey of Numerical Analysis*, McGraw-Hill, New York, 1962. MR **24** #B1271.

11. G. SZEGÖ, *Orthogonal Polynomials*, rev. ed., Amer. Math. Soc. Colloq. Publ., Vol. 23, Amer. Math. Soc., Providence, R. I., 1959. MR **21** #5029.

12. A. M. OSTROWSKI, *Solution of Equations and Systems of Equations*, Academic Press, New York, 1966.

13. W. RICHTER, "Estimation de l'erreur commise dans la méthode de M. W. E. Milne pour l'intégration d'un système de n équations différentielles du premier ordre," *Bull. Soc. Neuchâteloise Sci. Nat.*, v. 75, 1952, pp. 1–43. (Thèse, Univ. de Neuchâtel, 1952.) MR **16**, 865.