JOURNAL OF COMPUTATIONAL PHYSICS 9, 120–137 (1972)

A New Method for Solving Two-Point Boundary Value Problems Using Optimal Node Distribution

V. E. DENNY

University of California, Los Angeles, California

AND

R. B. LANDIS

San Fernando Valley State College, Northridge, California

Received January 19, 1971

A new method for solving two-point boundary value problems by finite difference methods has been developed. The basis for the method is the observation that local truncation errors associated with central difference analogues of the defining differential equation become arbitrarily small as the interior node points are arranged in an optimal sequence. The method is applied to several examples and comparisons are made with other approaches.

INTRODUCTION

It long has been recognized that improved accuracy is obtained for such numerical procedures as interpolation and differentiation, quadrature, and integration of differential equations, when estimates of associated truncation errors are included in the solution algorithm [1–3]. Alternatively, improved accuracies can be attained by reducing the magnitudes of local truncation errors via reductions in mesh spacings between adjacent node points. The purpose of the present contribution is to introduce a new method for solving two-point boundary value problems which operates on the total truncation error of the difference analogue for the differential equation in such manner as to effect an optimal node point distribution. In so doing, the set of truncation errors $\{T_i\}$ associated with successive nodal distributions $\{y_i^{(k)}\}$, where i denotes node index and (k) iteration index, is systematically reduced to zero. The method is shown to be exact for a homogeneous linear problem and gives highly accurate results for nonlinear problems.

The method is applied here to the second-order equation

$$u'' + P(u, y)\, u' + Q(u, y) = 0 \tag{1}$$

with Dirichlet boundary conditions

$$u(0) = 0 \qquad \text{and} \quad u(1) = 1. \tag{2}$$

It is required that u and its higher derivatives be continuous on the interval $0 \leqslant y \leqslant 1$. Results are obtained for several illustrative examples and comparisons are made with other approaches. In particular, an approach described by Fox [1] is extended so as to include, to any order, evaluations of the local truncation error T_i in the solution algorithm for the finite difference equations.

Analysis

Introducing 3-point central difference forms for u' and u'', Eq. (1) may be written

$$\left(\frac{2}{\delta_{12}}\right)\left(\frac{u_{i+1} - u_i}{\delta_2} - \frac{u_i - u_{i-1}}{\delta_1}\right) + P_i\left(\frac{\delta_1}{\delta_{12}}\frac{u_{i+1} - u_i}{\delta_2} + \frac{\delta_2}{\delta_{12}}\frac{u_i - u_{i-1}}{\delta_1}\right) + T_i + Q_i = 0 \qquad (i = 2, 3, \ldots, I), \tag{3}$$

where

$$\delta_2 \equiv y_{i+1} - y_i\,, \qquad \delta_1 \equiv y_i - y_{i-1}\,, \qquad \delta_{12} \equiv y_{i+1} - y_{i-1} \tag{4}$$

with $y_1 = 0$, $y_{I+1} = 1$, and

$$\begin{aligned} P_i &= P(u_i\,, y_i), \\ Q_i &= Q(u_i\,, y_i), \\ T_i &= -\frac{P_i\delta_1\delta_2}{\delta_{12}}\sum_{n=3}^{\infty}\frac{u_i^{(n)}}{n!}\left[\delta_2^{n-2} - (-1)^n\,\delta_1^{n-2}\right] \\ &\quad - \frac{2}{\delta_{12}}\sum_{n=3}^{\infty}\frac{u_i^{(n)}}{n!}\left[\delta_2^{n-1} + (-1)^n\,\delta_1^{n-1}\right]. \end{aligned} \tag{5}$$

Here, T_i represents the Taylor series terms which remain on extracting u_i' and u_i'' from the expressions

$$u_{i+1} = u_i + \delta_2 u_i' + \frac{\delta_2^{\,2}}{2!}u_i'' + \sum_{n=3}^{\infty}\frac{\delta_2^{\,n}}{n!}u_i^{(n)},$$

$$u_{i-1} = u_i - \delta_1 u_i' + \frac{\delta_1^{\,2}}{2!}u_i'' + \sum_{n=3}^{\infty}\frac{(-1)^n\,\delta_1^{\,n}}{n!}u_i^{(n)}.$$

Equation (3) may be rearranged to give

$$u_i = A_i u_{i+1} + B_i u_{i-1} + C_i , \qquad i = 2, 3,..., I, \tag{6}$$

where

$$A_i = (2/\delta_2 + P_i\delta_1/\delta_2)/D_i ,$$
$$B_i = (2/\delta_1 - P_i\delta_2/\delta_1)/D_i ,$$
$$C_i = \delta_{12}(T_i + Q_i)/D_i ,$$
$$D_i = (2/\delta_2 + P_i\delta_1/\delta_2 + 2/\delta_1 - P_i\delta_2/\delta_1).$$

From Eq. (5) it is seen that

$$T_i = T_i(y_{i-1} , y_i , y_{i+1})$$

and, hence, that

$$dT_i = (\partial T_i/\partial y_{i-1})\, dy_{i-1} + (\partial T_i/\partial y_i)\, dy_i + (\partial T_i/\partial y_{i+1})\, dy_{i+1} . \tag{7}$$

The partial derivatives $\partial T_i/\partial y_{i-1}$, $\partial T_i/\partial y_i$, $\partial T_i/\partial y_{i+1}$ are obtained in straightforward fashion from Eq. (5) upon noting that P_i and $u_i^{(n)}$ are functions of y_i only.

Equations (6) and (7) constitute sets of algebraic equations for which the coefficient matrices are tri-diagonal; thus, they yield to efficient solution by means of elimination techniques [4]. For computational purposes the series for T_i , Eq. (5), is truncated at the Nth term.

Now, given an initial distribution $\{y_i^{(1)}\}$ with associated $\{T_i^{(1)}\}$, the problem is one of obtaining successive distributions $\{y_i^{(k)}\}$ for which $\{T_i^{(k)}\} \to \{0\}$. These are obtained from Eq. (7) which becomes, in explicit difference form,

$$-T_i^{(k-1)} = \left(\frac{\partial T_i}{\partial y_{i-1}}\right)^{(k-1)} (y_{i-1}^{(k)} - y_{i-1}^{(k-1)}) + \left(\frac{\partial T_i}{\partial y_i}\right)^{(k-1)} (y_i^{(k)} - y_i^{(k-1)})$$
$$+ \left(\frac{\partial T_i}{\partial y_{i+1}}\right)^{(k-1)} (y_{i+1}^{(k)} - y_{i+1}^{(k-1)}). \tag{8}$$

The superscript (k) denotes iterate number, iteration being required because the set is nonlinear. In order to implement Eq. (8), intermediate values of $\{u_i\}$ and $\{u_i^{(n)}\}$ are needed. These were obtained as follows:

(i) Assuming $\{T_i^{(k)}\}$ to be zero, $\{u_i^{(k)}\}$ was obtained from Eq. (6), iterating on the nonlinear coefficients $\{A_i\}$, $\{B_i\}$, $\{C_i\}$ until a specified convergence was attained.

(ii) Given $\{u_i^{(k)}\}$, $\{u_i'^{,(k)}\}$ was obtained by means of numerical differentiation.

(iii) Given $\{u_i'^{,(k)}\}$, $u_i^{(n),(k)}$ was obtained by recursively differentiating Eq. (1) analytically:

$$u^{(n)} = -(Pu' + Q)^{(n-2)}, \qquad n = 2, 3,..., N + 1. \tag{9}$$

Initially, a uniform distribution $\{y_i\}$ was assumed ($\delta_1 = \delta_2$). Successive distributions $\{y_i^{(k)}\}$ were then obtained by

(iv) computing $T_i^{(k-1)}$ from Eq. (5) and $(\partial T_i/\partial y_{i-1})^{(k-1)}$, etc., from the partial derivatives of Eq. (5) with respect to y_{i-1}, etc., and

(v) setting $\{T_i^{(k)}\} = 0$ and applying Eq. (8).

Since $\{u_i^{(k)}\}$ changes as $\{y_i^{(k)}\}$ changes, it was found that computational efficiency is promoted if the prescribed precision of the "converged" values

$$\epsilon^{(k)} \geqslant | u_i^{(k),(l)} - u_i^{(k),(l-1)} |/u_i^{(k),(l)},$$

where (l) denotes iterate number in performing step (i), is increased with increasing (k). Typically, $\epsilon^{(k)} = 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$, etc., as $k = 1, 2, 3, 4,...$, proved to be near optimal.

The ultimate accuracy attainable by the method depends, to first order on errors in $\{u_i'\}$. For the illustrative examples presented below, u_i' was extracted from a five-point Lagrangian interpolating polynomial [5] at node points $i = 3$ through $I - 1$, with u_i' at $i = 1, 2$, and I being obtained from Taylor series extrapolation. For example, at $i = 2$

$$u_2' = u_3' + \sum_{n=1}^{N} \frac{(-1)^n \, \delta_3{}^n u_3^{(n+1)}}{n!}. \tag{10}$$

In addition, the accuracy of the method is curtailed when the (nonlinear) coefficient $P(u, y)$ is of such form as to introduce additional truncation errors. This is illustrated below for Case III for the integrodifferential equation

$$u'' + \left(a + b \int_0^y u \, dy\right) u' = 0.$$

(Although such truncation errors could in principle be incorporated in the method, this was not done because the coefficient matrix for the set of algebraic Eqs. (7) would not, in general, remain tri-diagonal.)

Results and Discussion

To illustrate the method, Eq. (1), subject to the boundary conditions (2), was solved for the following special cases:

I. $P = 4, \quad Q = 0,$

II. $P = \text{constant}; \quad Q = \gamma e^{-\beta/(1+\alpha u)}, \quad \beta = 10, \quad \alpha = 0.25,$

A. $P = 4, \quad \gamma = 0.5e^{10},$

B. $P = 0, \quad \gamma = 0.5e^{10},$

C. $P = -4, \quad \gamma = 0.5e^{10},$

D. $P = 4, \quad \gamma = e^{10},$

III. $P = \alpha(\beta + \gamma \int_0 u dy), \quad Q = 0,$

A. $\alpha = 1, \quad \beta = 0, \quad \gamma = 18,$

B. $\alpha = 1.125, \quad \beta = 5, \quad \gamma = 2.25,$

C. $\alpha = 4.03, \quad \beta = -0.75, \quad \gamma = 8.06.$

With $u(1) = 1$, Cases I and II represent, for example, energy transfer in a Couette flow [6] with mass injection where, for Case II, the injected fluid undergoes a zero th- order chemical reaction with the rate of reaction exhibiting an Arrhenius type dependence [7] on the dimensionless temperature u. With $u(1) \to 1$, Case III is representative of self-similar boundary layer flows [8].

In order to assess the relative accuracy and computational efficiency of the method, solutions to the above problems were also obtained for $\delta_1 = \delta_2 = \delta = 1/I$, i.e., for regular node spacing with (i) no corrections ($T_i = 0$ in Eq. (3)) and (ii) Taylor series corrections (T_i given by Eq. (5)). Whereas Fox [1] calculates $\{T_i\}$ by extracting the required $\{u_i^{(n)}\}$ from high order differences of the $\{u_i\}$, the procedure used here is to calculate $\{u_i'\}$ numerically and then obtain $\{u_i^{(n)}\}$, $n = 2,..., N + 1$, from Eq. (9). As well be seen, this has the advantage of enabling retention of more terms in Eq. (5) and, for the examples treated here, proves to be more accurate than the method of Fox, where the number of terms and the accuracy are limited by the number of internal points $I - 1$. Of course, the accuracy of the present methods is, as mentioned above, limited by numerical errors in calculating $\{u_i'\}$. Where possible, solutions also were obtained by means of the fourth-order Runge–Kutta method.

Numerical results for the illustrative examples are presented in Tables I–III-C, where results are shown for regular node spacing, with and without Taylor correction, and for optimal node spacing. Values of u_i were computed at nine interior node points except for Case II-D where nineteen nodes were used. A graph indicating the variation in the dependent variable is included for each case and the optimal node distribution is marked by vertical lines. The error listed for each method is defined as

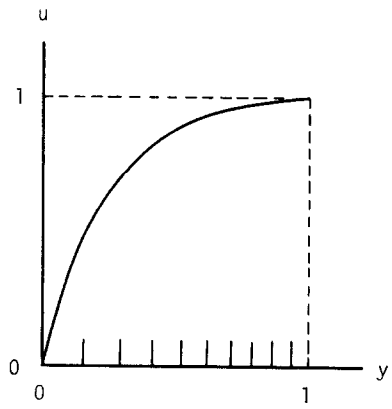
$$\% \text{ Error} = 100(u - u_e)/u_e ,$$

where u_e is the "exact" solution. For the nonlinear cases, u_e was obtained using (i) fourth-order Runge–Kutta and (ii) Eq. (3) with successive reductions in mesh

TABLE I

Results for $u'' + 4u' = 0$; $u(0) = 0$, $u(1) = 1$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	u	% Error	u	% Error	y	u	% Error
0.1	0.339216	1.008	0.335834	8.8×10^{-4}	0.158967	0.479303	1.7×10^{-14}
0.2	0.565360	0.787	0.560949	6.9	0.295937	0.706817	1.2×10^{-14}
0.3	0.716122	0.601	0.711847	5.3	0.416295	0.825972	3.4×10^{-15}
0.4	0.816631	0.447	0.812997	3.9	0.523653	0.893242	9.3×10^{-15}
0.5	0.883636	0.322	0.880800	2.8	0.620557	0.933542	1.6×10^{-14}
0.6	0.928307	0.222	0.926249	2.0	0.708869	0.958872	1.7×10^{-14}
0.7	0.958087	0.144	0.956714	1.3	0.789993	0.975439	1.4×10^{-14}
0.8	0.977940	0.082	0.977135	7.3×10^{-5}	0.865013	0.986643	9.8×10^{-15}
0.9	0.991176	0.036	0.990824	3.2	0.934787	0.994439	2.8×10^{-15}



Remarks

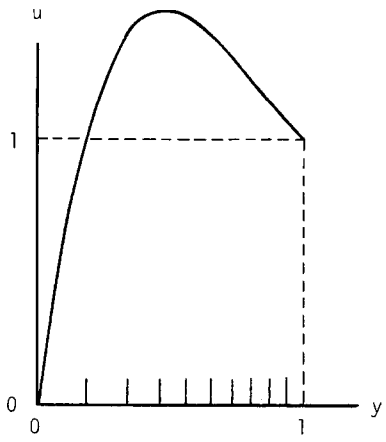
[a] N = 13, k = 5

[b] N = 20, k = 6

TABLE II-A

Results for $u'' + 4u' + 0.5\, e^{10}\, e^{-10/(1+0.25u)}$; $u(0) = 0$, $u(1) = 1$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	u	% Error	u	% Error	y	u	% Error
0.1	0.686258	3.34	0.664048	4.3×10^{-4}	0.181978	1.028592	2.5×10^{-4}
0.2	1.125742	3.18	1.091087	3.6	0.334546	1.395574	1.4
0.3	1.381268	2.94	1.341834	2.7	0.456912	1.480301	1.1
0.4	1.497352	2.61	1.459276	2.5	0.558775	1.457082	6.2×10^{-5}
0.5	1.511251	2.20	1.478763	2.0	0.647414	1.392628	5.7
0.6	1.455851	1.73	1.431060	1.8	0.727145	1.313521	5.3
0.7	1.358846	1.25	1.342041	1.1	0.800636	1.231114	5.0
0.8	1.241570	0.79	1.231849	8.9×10^{-5}	0.869700	1.150342	4.6
0.9	1.118872	0.37	1.114780	7.2	0.935730	1.073164	3.4



Remarks

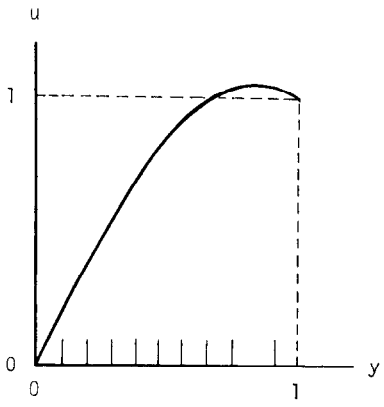
[a] N = 9, k = 6

[b] N = 8, k = 4

TABLE II-B

Results for $u'' + 0.5\, e^{10}\, e^{-10/(1+0.25u)} = 0$; $u(0) = 0$, $u(1) = 1$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	u	% Error	u	% Error	y	u	% Error
0.1	0.193173	0.073	0.193032	-1.1×10^{-4}	0.104292	0.201163	3.0×10^{-5}
0.2	0.378420	0.096	0.378055	-1.0	0.201116	0.380061	2.9
0.3	0.551801	0.120	0.551138	-1.0	0.292918	0.539373	2.8
0.4	0.708376	0.142	0.707372	-9.6×10^{-5}	0.381640	0.680200	2.7
0.5	0.842441	0.158	0.841115	-8.9	0.469070	0.802508	2.5
0.6	0.948029	0.163	0.946483	-7.6	0.557217	0.905245	2.2
0.7	1.019648	0.154	1.018080	-5.9	0.648992	0.986074	1.6
0.8	1.053147	0.126	1.051825	-4.8	0.750647	1.040085	$<1.0\times10^{-6}$
0.9	1.046458	0.075	1.045672	-1.9	0.906860	1.043783	1.2×10^{-5}



Remarks

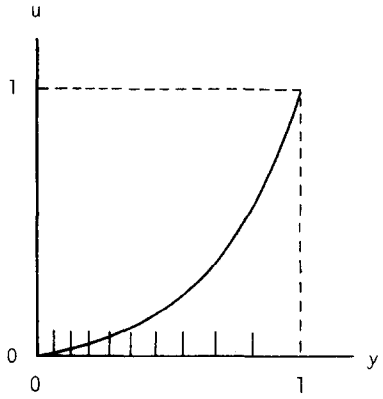
[a] N = 8, k = 5

[b] N = 8, k = 5

TABLE II-C

Results for $u'' - 4u' + 0.5\, e^{10}\, e^{-10/(1+0.25u)} = 0$; $u(0) = 0$, $u(1) = 1$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	u	% Error	u	% Error	y	u	% Error
0.1	0.023393	-1.60	0.023771	-5.8×10^{-4}	0.060836	0.013937	-2.1×10^{-5}
0.2	0.051857	-1.63	0.052717	-5.6	0.126135	0.030766	-1.7
0.3	0.087451	-1.64	0.088903	-5.1	0.196624	0.051636	-1.9
0.4	0.133100	-1.59	0.135249	-4.4	0.273232	0.078361	-2.6
0.5	0.192950	-1.49	0.195867	-3.6	0.357188	0.113932	-3.3
0.6	0.272823	-1.33	0.276491	-2.4	0.450177	0.163567	-3.9
0.7	0.380796	-1.10	0.385022	-1.4	0.554639	0.236993	-4.7
0.8	0.527849	-0.80	0.532102	-1.0	0.674426	0.354075	-5.4
0.9	0.728376	-0.43	0.731540	$+6.8\times10^{-5}$	0.816635	0.561229	-6.6



Remarks

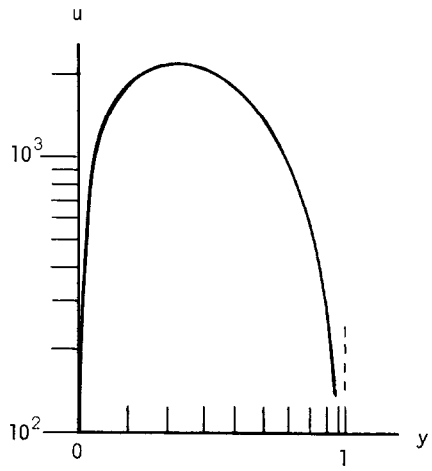
[a] N = 11, k = 6

[b] N = 8, k = 4

TABLE II-D

Results for $u'' + 4u' + e^{10}\, e^{-10/(1+0.25u)} = 0$; $u(0) = 0$, $u(1) = 1$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	$ux10^{-3}$	% Error	$ux10^{-3}$	% Error	y	$ux10^{-3}$	% Error
0.1	1.266966	0.487	1.260887	$5.1x10^{-3}$	0.184950	1.859920	$6.4x10^{-3}$
0.2	1.939262	0.414	1.931322	2.7	0.339118	2.230741	2.9
0.3	2.211036	0.367	2.202973	2.0	0.472413	2.093691	1.7
0.4	2.214210	0.337	2.206814	1.8	0.589758	1.769894	$9.0x10^{-4}$
0.5	2.037570	0.320	2.031128	1.3	0.694263	1.382967	4.3
0.6	1.740850	0.321	1.735394	1.9	0.787536	0.987342	-1.0
0.7	1.364359	0.337	1.359813	2.3	0.868861	0.617315	-8.3
0.8	0.935644	0.404	0.931907	3.1	0.933456	0.313456	$-2.4x10^{-3}$
0.9	0.474534	0.630	0.471590	5.9	0.976421	0.110112	$-3.3x10^{-3}$



Remarks

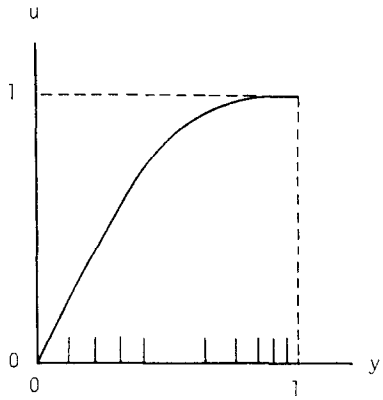
[a] N = 11, k = 6

[b] N = 11, k = 13

TABLE III-A

Results for $u'' + 18 \int_0^y u\,dy\; u' = 0;\ u(0) = 0,\ u(1) = 0.998980$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	u	% Error	u	% Error	y	u	% Error
0.1	0.199476	0.270	0.198931	-3.6×10^{-3}	0.127208	0.252672	1.8×10^{-3}
0.2	0.395409	0.414	0.393766	-3.4	0.228416	0.447168	1.5
0.3	0.577837	0.535	0.574745	-3.2	0.321324	0.610337	1.2
0.4	0.733417	0.607	0.728967	-2.9	0.413346	0.746920	8.9×10^{-4}
0.5	0.851182	0.606	0.846029	-2.6	0.642713	0.945503	1.4
0.6	0.928146	0.521	0.923316	-2.3	0.759850	0.981448	-4.1×10^{-5}
0.7	0.970559	0.372	0.966948	-1.6	0.842457	0.992381	-3.8
0.8	0.989851	0.208	0.987788	-1.0	0.905990	0.996438	-2.1
0.9	0.996936	0.078	0.996159	-2.6×10^{-4}	0.957263	0.998164	-8.2×10^{-6}



Remarks

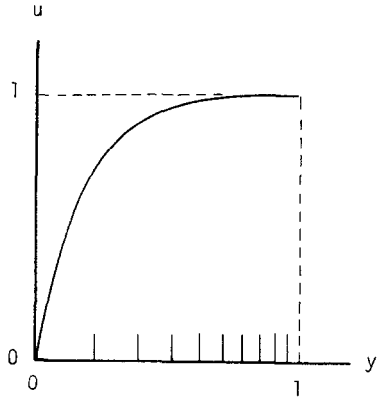
[a] N = 9, k = 6

[b] N = 12, k = 27

TABLE III-B

Results for $u'' + 1.125(5 + 2.25\int_0^y u\,dy)u' = 0$; $u(0) = 0$, $u(1) = 0.999020$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	u	% Error	u	% Error	y	u	% Error
0.1	0.454987	2.13	0.445545	7.3×10^{-3}	0.226362	0.743341	9.0×10^{-3}
0.2	0.708519	1.54	0.697787	5.4	0.391308	0.910717	4.2
0.3	0.847536	1.07	0.838587	4.0	0.518840	0.962204	2.2
0.4	0.922140	0.708	0.915683	2.8	0.621986	0.981684	1.2
0.5	0.961203	0.444	0.956976	1.9	0.708160	0.990209	6.8×10^{-4}
0.6	0.981122	0.261	0.978576	1.2	0.781908	0.994365	3.9
0.7	0.991004	0.142	0.989602	8.0×10^{-4}	0.846199	0.996566	2.2
0.8	0.995770	0.069	0.995092	4.0	0.903072	0.997812	1.1
0.9	0.998003	0.025	0.997758	1.8	0.953981	0.998555	4.5×10^{-5}



Remarks

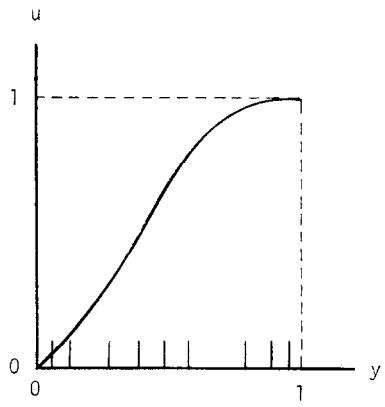
[a] N = 11, k = 6

[b] N = 19, k = 22

TABLE III-C

Results for $u'' + 4.03(-0.75 + 8.06\int_0^y udy)u' = 0$; $u(0) = 0$, $u(1) = 0.999030$

	Regular Node Spacing				Optimal Node Spacing[b]		
	No Correction		Taylor Correction[a]				
y	u	% Error	u	% Error	y	u	% Error
0.1	0.087385	-0.692	0.087977	-1.9×10^{-2}	0.060916	0.050457	5.2×10^{-3}
0.2	0.204249	-0.381	0.204990	-2.0	0.131065	0.121014	5.3
0.3	0.353278	-0.026	0.353299	-2.0	0.279729	0.320966	4.8
0.4	0.526830	0.334	0.524976	-1.9	0.385580	0.499600	4.3
0.5	0.702087	0.637	0.697525	-1.7	0.481560	0.667194	3.5
0.6	0.847053	0.795	0.840246	-1.5	0.575487	0.809631	2.6
0.7	0.939720	0.731	0.932783	-1.3	0.785999	0.974106	5.3×10^{-4}
0.8	0.982849	0.471	0.978155	-9.8×10^{-3}	0.884378	0.993234	5.3×10^{-5}
0.9	0.996449	0.179	0.994638	-3.4	0.950600	0.997663	2.4×10^{-6}



Remarks

[a] N = 11, k = 6

[b] N = 19, k = 35

spacing δ and tightening of error tolerances until both (i) and (ii) agreed to nine significant figures. To avoid roundoff errors, the computations were carried out in double precision. The number of terms, N, retained in the series for T_i (Eq. (5)) and the number of iterations needed to converge the solution, (k), are also presented.

Case I. For this case, a closed form solution exists; furthermore, it is seen from Eqs. (5), (7), and (8) that the solution algorithm for extracting the optimal distribution is homogeneous in u_i'. Such is not the case for the Taylor series correction method where, as may be seen from Eq. (5), errors in T_i are, to first order, proportional to errors in u_i'. This is verified in Table I where the results for the optimal node method are seen to be exact (within roundoff errors). Both methods converged rapidly, (k) being 6 in the former case and 5 in the latter.

The relative computational efficiencies of the methods are depicted in Fig. 1, where execution times are plotted as a function of percent error for (i) the standard

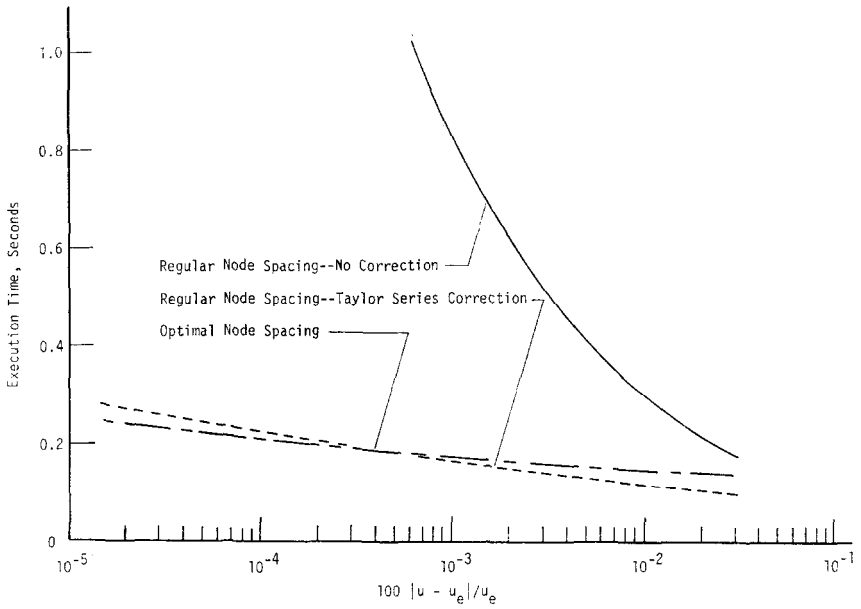


FIG. 1. Comparison of computational time for equal and optimal node distribution—Case I.

finite difference procedure without corrections, (ii) the Taylor series correction method, and (iii) the optimal node method. We note that the execution times for the Taylor and optimal node methods are comparable and increase very slightly with increasing accuracy, whereas for the standard finite difference method execution time increases very rapidly as the error is decreased below 0.1 %.

Case II. For this case, the effects of a nonlinear source term ($Q = \gamma e^{-\beta/(1+\alpha u)}$) are investigated for various "blowing" parameters ($P = 4, 0, -4$). As discussed above, the ultimate accuracy attainable by the method is now limited by inaccuracies in u_i'. However, as shown in Tables II-A–II-C, the results for the optimal method are significantly better than those for the Taylor series method. The reason for this is shown in Fig. 2, where errors in the optimal method are seen to be insensitive to small errors in the precise location of the optimal distribution. (Shown in Fig. 2 is Case II-A, similar trends being exhibited by the other cases.)

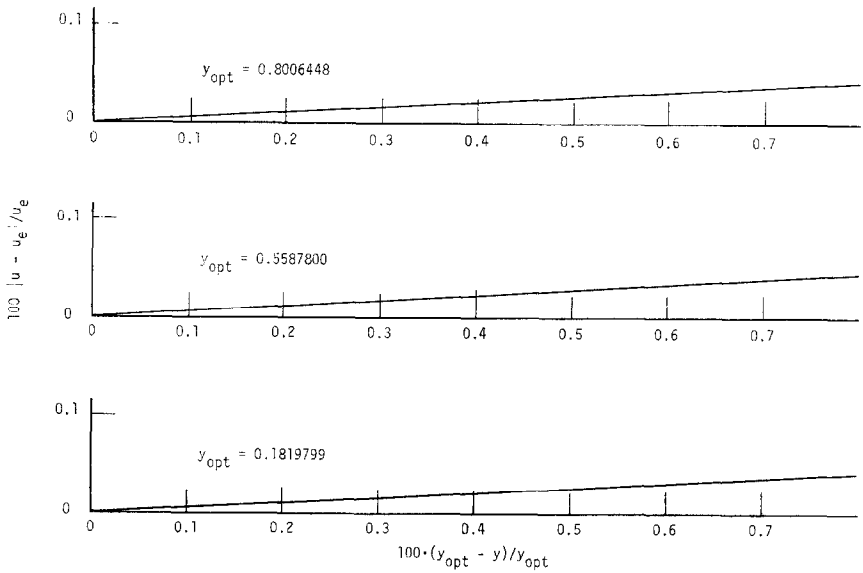


FIG. 2. Effect of deviations from optimal node point distribution on accuracy.

Also of interest are the effects of the parameters of the problem on the distribution of node points. As may be seen from the figures in Tables II-A and II-C, "suction" ($P = 4$) tends to move the distribution out whereas "blowing" ($P = -4$) tends to move points in. For $P = 0$ (Table II-B), however, the distribution is more or less regular with the exception of a large step about the maximum value of u. Additionally, we note that the presence of the term Pu' is not required to obtain an optimal distribution.

Not shown are results which were obtained using fourth-order Runge–Kutta. For the same accuracy, Runge–Kutta (as is well known) is more efficient than the methods cited above. However, it also is known that shooting techniques, such as Runge–Kutta, fail when gradients become large [9]. Such is the case for problem

II-D, where, for a two-fold increase in γ, the solution assumes the form shown in Table II-D. (The dramatic change in the nature of the solution with increasing γ is characteristic of so called "multiple" steady-state problems [10].) Despite reasonable efforts to implement Runge–Kutta for this case (even to the extent that the initial "guessed" value of $u'(0)$ was input correct to five significant figures), it failed to converge.

Computational efficiency of the various methods is compared for Case II-A in Fig. 3. The results of this nonlinear problem are similar to those obtained for Case I in that execution times for the optimal node and Taylor series correction methods are comparable, with the standard finite difference method requiring substantially longer times to achieve accuracies better than 0.5 %.

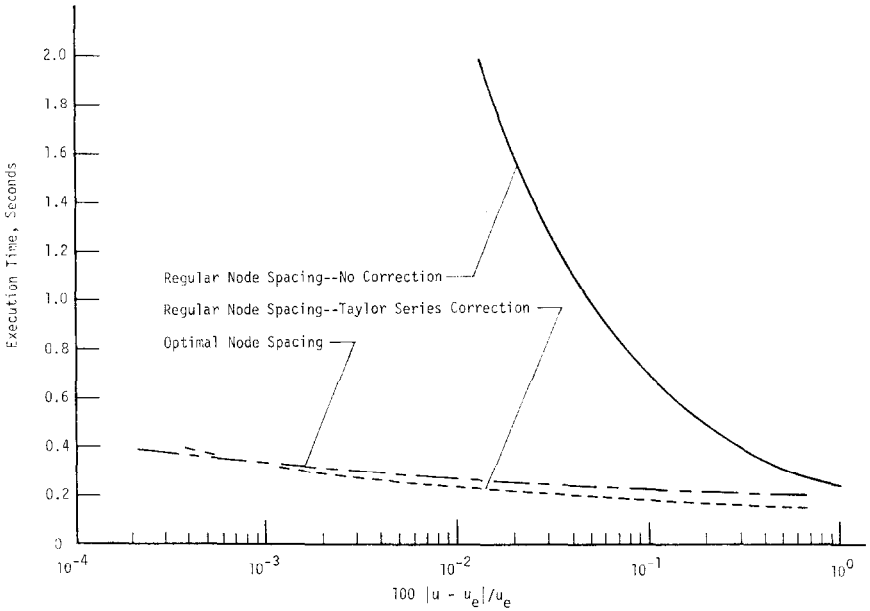


FIG. 3. Comparison of computational time for equal and optimal node distribution—Case II-A.

Case III. For this case, the effects of a nonlinear coefficient P on the solution method are investigated. An additional feature of the case is that the nonlinearity introduces further truncation error; this would not be the case, for example, if P were u itself. (As mentioned earlier, the solution procedure could be modified to include such errors; however, considering the quadrature involved in evaluating P it is evident that to do so would require that a full matrix be inverted to yield the

optimal node distribution.) Despite the added error, the results (Tables III-A–III-C) are still uniformly excellent.

Of additional interest are the node point distributions for the optimal method, particularly the distribution for the classical Blasius problem (Case III-A). As might be expected, the search for so distorted a node distribution was highly nonlinear, stability requiring that the values of $y_i^{(k)} - y_i^{(k+1)}$ be damped at each iterate. This is reflected in the larger values of (k) required to achieve convergence.

Closing Remarks

Of practical concern in solving two-point boundary value problems by means of finite difference methods is the extraction of accurate gradients at the boundaries. This concern is accentuated in the present method where not only is the node distribution coarse but, as may be seen for various cases in Tables I through III, node spacings near one or the other of the boundaries are relatively larger. Given $\{u_i\}$, one could extract u' at a boundary by means of numerical differentiation of the near wall data. For the results cited above, such a procedure would yield relatively inaccurate values. However, an important feature of the present method (which applies as well to the Taylor series approach) is that (more accurate) central difference expressions for u' at interior node points may be extrapolated to a boundary using Eq. (10).

The above observations are summarized in Table IV where values of $u'(0)$, as obtained by differentiating a parabolic fit to the values u_1, u_2, and u_3, are compared with those obtained using Eq. (10). If greater precision is required, one

TABLE IV

Comparisons of Methods for Extracting the Wall Gradient, $u'(0)$

	Taylor Correction				Optimal Node Spacing			
	Parabolic Fit		Taylor Extrapolation		Parabolic Fit		Taylor Extrapolation	
Case	$u'(0)$	% Error	$u'(0)$	% Error	$u'(0)$	% Error	$u'(0)$	% Error
II-A	7.8255	−3.5	8.09719	−0.129	7.4185	−8.5	8.10973	0.025
II-B	1.9704	0.6	1.95975	−0.002	1.9709	5.7	1.95960	−0.010
II-C	0.2118	−2.4	0.21706	−0.020	0.2153	−0.8	0.21709	−0.006
II-D	15561.	−4.5	16374.9	0.294	14229.	−12.7	16353.4	0.346
III-A	1.9688	−1.2	1.99066	−0.086	2.0222	1.5	1.98892	−0.158
III-B	5.4220	−7.0	5.80800	−0.338	4.5965	−21.1	5.78301	−0.770
III-C	0.7346	−2.6	0.75570	0.153	0.7458	−1.2	0.75320	−0.178

could of course re-solve Eq. (1) in the interval between $y = 0$ and the first interior node point, assigning the numerical value of u_2 as exterior boundary condition and renormalizing the independent variable. For example, when this was done for Case III-B, the error in $u'(0)$ was reduced from $-0.77\,\%$ to $-0.0015\,\%$.

Although the method has been applied here to ordinary differential equations, it could in principle be applied to "marching" problems. Consider, for example, the simple diffusion equation

$$\partial\psi/\partial t = \partial^2\psi/\partial y^2,$$

where time-dependent optimal distributions $\{y_i\}$ are sought. The higher-order derivatives $\partial^n\psi/\partial y^n$ required to implement the method would be obtained by recursively differentiating the above equation

$$\partial^n\psi/\partial y^n = \partial(\partial^{n-2}\psi/\partial y^{n-2})/\partial t,$$

where, now, time derivatives of the spatial derivatives would be required. Furthermore, the optimal distribution varies with time and it would be necessary to interpolate spatially values of the dependent variable as well as its derivatives with respect to y at upstream stations. However, for cases in which $\{y_{i,\mathrm{opt}}\}$ varies weakly with time, the method may show computational advantage.

REFERENCES

1. L. FOX, Numerical solution of two-point boundary value problems, Oxford Press, London, 1957.
2. V. J. KRYLOV, Approximate calculation of integrals, Macmillan, New York, 1962.
3. R. ZURMÜHL, Praktische mathematik für ingenieure und physiker, Springer, Berlin, 1963.
4. J. DOUGLAS, "The Effect of round-off error in the numerical solution of the heat equation," *Journal of the Association for Computing Machinery*, **6** (1959), 48–58.
5. F. B. HILDEBRAND, Introduction to numerical analysis, McGraw-Hill, New York, 1956.
6. M. W. RUBESIN, "An analytical estimation of the effect of transpiration cooling on the heat transfer and skin friction characteristics of a compressible turbulent boundary layer," National Advisory Committee for Aeronautics Report No. TN 3341, 1954.
7. J. F. CLARKE AND M. MCCHESNEY, The dynamics of real gases, Butterworths, London, 1964.
8. H. SCHLICHTING, Boundary layer theory, 6th ed., McGraw-Hill, New York, 1968.
9. D. L. HARRIS AND W. H. REID, "On the stability of viscous flow between rotating cylinders, Part 2. Numerical analysis," *J. Fluid Mech.* **20** (1964), 95–101.
10. G. R. GAVALAS, Nonlinear differential equations of chemically reacting systems, Springer-Verlag, New York, 1968.