JOURNAL OF COMPUTATIONAL PHYSICS **21**, 55–84 (1976)

An Implicit Finite-Element Algorithm for the Boundary Layer Equations*

Z. POPINSKI AND A. J. BAKER

Textron Bell Aerospace, P.O. Box 1, Buffalo, New York 14240

Received December 27, 1974; revised September 3, 1975

The two-dimensional boundary layer equations are presented in a finite-element form using the Galerkin criterion within the method of weighted residuals. A linear shape function and one-dimensional natural coordinates, applied locally, are used to discretize the coordinate direction transverse to the main flow. Streamwise derivatives are replaced by an implicit integration algorithm and yield tridiagonal or diagonal matrix algebraic equations that are solved by an efficient procedure. Solutions are obtained by marching in the streamwise direction. Numerical comparisons are made between the finite-element and Crank–Nicolson finite-difference techniques, as well as comparison to the Blasius solution for accuracy and convergence evaluation in incompressible flow. Extensive error correlations are established to evaluate the effect of *dx* and *dy* step sizes on solution numerics as well as the implications on round-off and truncation error as influenced by these step sizes. The effect of the aspect ratio of the element has been evaluated, and realistic values for accurate computational processes are established. As a result of these studies, it is concluded that the linear element finite-element technique, on the multiple basis of accuracy, convergence, and computation time, is competitive with the best second-order accurate finite difference method.

1. INTRODUCTION

Most boundary value problems in fluid mechanics, of interest in engineering applications, require numerical solution. The increased capabilities of computers during the last decade promoted development of sophisticated finite-difference solution algorithms [1] that have met the engineering needs for numerical solutions to the partial differential equations of fluid mechanics. Recently, increased attention has been directed to application of the finite-element solution technique of boundary value problems in fluid mechanics [2]. This interest was initially stimulated mainly by the greater flexibility in discretization of the solution domain

* This work was supported as part of a Bell Aerospace Company IR&D Program.

as afforded by the finite-element approach. The numerical result presented in this paper afford a critical evalutation of the finite-element algorithm in comparison to the accepted finite-difference procedure, as well as an analytical solution.

The theoretical foundation of the finite-element method presented in this analysis is a local adoption of the Galerkin criterion within the method of weighted residuals [3]. The unknown dependent variable Q, is expressed in the interpolation form

$$Q_m{}^*(x_i, \xi) = \sum_{i=1}^{n} \phi(x_i)\, Q_i(\xi), \tag{1}$$

where ϕ is a set of trial functions, and the Q_i are the (unknown) expansion coefficients with dependent variable values corresponding to node locations of the discretization. Substituting the approximation $Q_m{}^*$ into the partial-differential equation to be solved, and rendering the solution residuals orthogonal to the space of the trial functions through an inner product operation, yields a set of ordinary-differential equations for the unknown coefficients Q_i. In the process of formulation of the equation in the finite-element representation, operations on products of matrices are required. It is convenient to perform these operations in a local system using natural coordinates.

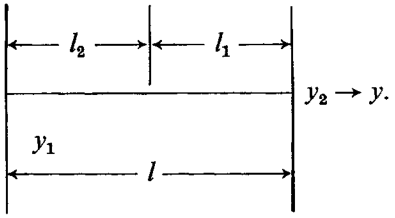
2. The Finite-Element Algorithm

2.1. *Finite-Element Algorithm Development*

Using the concept of a finite element, the total domain of physical phenomenon under study is considered as an assemblage of disjoint interior subdomains (finite elements), interconnected at gridpoints and spanning the boundary value limited solution domain. The choice of discretization depends on the geometry of the domain, and on the number of independent space coordinates necessary to describe the problem. For the two-dimensional boundary layer flow, only a one-dimensional element is required. Such an element is simply a straight line with nodal points at least at the ends. Computations are simplified by the introduction of natural coordinates, defined in local coordinates for a particular element. For a one-dimensional line element (parallel to the global y-axis), the relation between the natural coordinates ϕ_1 and ϕ_2, of any point p, and the local coordinate system is

$$\begin{bmatrix} 1 & 1 \\ y_1 & y_2 \end{bmatrix} \begin{Bmatrix} \phi_1 \\ \phi_2 \end{Bmatrix} = \begin{Bmatrix} 1 \\ y \end{Bmatrix}, \tag{2}$$

where $\phi_1 = l_1/l$, and $\phi_2 = l_2/l$.



Solution of (2) yields,

$$\phi_1 = 1 - (y/l), \quad \phi_2 = (y/l), \tag{3}$$

where l is the length of the element. Integration of polynomials over domains, which is required to obtain the finite-element formulation of the boundary layer partial-differential equations is straightforward using the formula [4]:

$$\int_l \phi_1{}^p \phi_2{}^q \, dl = l \frac{p! \, q!}{(p+q+1)!}. \tag{4}$$

The two-dimensional equations of incompressible boundary layer flow with zero pressure gradient are:

$$L_1(u) = u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} - \frac{1}{\rho} \frac{\partial}{\partial y} \left(\mu \frac{\partial u}{\partial y} \right) = 0, \tag{5}$$

$$L_2(v) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0. \tag{6}$$

The unknown functions u and v, throughout the solution domain, are approximated by finite-element interpolation polynomials within the mth subdomain R_m, as

$$u_m{}^* = \{\phi(y)\}^T \{u(x)\}_m = \{u(x)\}_m^T \{\phi(y)\}, \tag{7}$$

$$v_m{}^* = \{v(x)\}_m^T \{\phi(y)\}, \tag{8}$$

where $\{\phi(y)\}^T = \{\phi_1, \phi_2\}$.

To determine the unknown values $u(x)$, the concept of weighted residuals is introduced. We define a weighted average integral of u over the solution domain y as the integral

$$\langle w, u \rangle \equiv \int_\delta wu \, dy, \tag{9}$$

where w is the prescribed weighting function, usually $\{\phi\}$. The $u(x)$ values are obtained by equating to zero the weighted integrated residuals over the solution domain

$$\int_\delta \{\phi\} L_1(u^*) \, dy = 0,$$

or

$$\int_{\delta} \{\phi\} \left[u^* \frac{\partial u^*}{\partial x} + v^* \frac{\partial u^*}{\partial y} - \frac{1}{\rho} \frac{\partial}{\partial y} \left(\mu \frac{\partial u^*}{\partial y}\right)\right] dy = 0. \tag{10}$$

For the finite-element solution procedure, term by term for Eq. (10) is evaluated with each finite subdomain δ_m, i.e., finite element as follows:

$$\int_{\delta_m} \{\phi\}\, u_m{}^* u_{m,x}^*\, dy = \int_{\delta_m} \{\phi\}\{u(x)\}_m^T \{\phi\}\{\phi\}^T\, dy \{u(x)\}_{m,x} . \tag{11}$$

The kinematic viscosity ν is also approximated by the interpolation formula (1) within the mth subdomain as,

$$v_m{}^* = \{v\}_m^T \{\phi\}. \tag{12}$$

Evaluating the remaining terms of Eq. (10), one obtains for the finite-element solution algorithm form of (10) evaluated within the mth solution subdomain

$$\begin{aligned} \{u\}_m^T \frac{l_m}{12} \begin{bmatrix} \begin{Bmatrix} 3 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \\ \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 3 \end{Bmatrix} \end{bmatrix} \{u\}_{m,x} + \{v\}_m^T \frac{1}{6} \begin{bmatrix} \begin{Bmatrix} -2 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 2 \\ 1 \end{Bmatrix} \\ \begin{Bmatrix} -1 \\ -2 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 2 \end{Bmatrix} \end{bmatrix} \{u(x)\}_m \\ = - \frac{\{v\}_m^T}{2l_m} \begin{bmatrix} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -1 \end{Bmatrix} \\ \begin{Bmatrix} -1 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \end{bmatrix} \{u(x)\}_m . \end{aligned} \tag{13}$$

Equation (13) must be evaluated within each finite element l_m, and the totality assembled into the global equation system using Boolean algebra by the imposition of nodal compatibility. This process of constructing the algebraic equations for the global system from the algebraic equations for the individual elements, which is routine practice in structural analysis, requires that all elements adjacent to a particular node must have the same value of the dependent variables at that node. This results in a system of n equations containing u and $u_{,x}$ derivatives. After nondimensionalization, the following global form is obtained for the element equation (13):

$$\begin{aligned} &[(u_{n-1} + u_n)\, u_{n-1,x} + (u_{n-1} + 6u_n + u_{n+1})\, u_{n,x} + (u_n + u_{n+1})\, u_{n+1,x}] \\ &\quad = (6/\mathrm{Re})[(v_{n-1} + v_n)\, u_{n-1} - (v_{n-1} + 2v_n + v_{n+1})\, u_n + (v_n + v_{n+1})\, u_{n+1}] \\ &\quad\quad - 2[-(v_{n-1} + 2v_n)\, u_{n-1} + (v_{n-1} - v_{n+1})\, u_n + (2v_n + v_{n+1})\, u_{n+1}], \end{aligned} \tag{14}$$

where $\mathrm{Re} = lu_\infty/\nu_{\mathrm{ref}}$.

$$
\begin{bmatrix}
(3u_1+u_2)(u_1+u_2) \\
(u_1+u_2)(u_1+6u_2+u_3)(u_2+u_3) \\
\qquad (u_2+u_3)(u_2+6u_3+u_4)(u_3+u_4) \\
\vdots \\
(u_{n-3}+u_{n-2})(u_{n-3}+6u_{n-2}+u_{n-1})(u_{n-2}+u_{n-1}) \\
\qquad (u_{n-2}+u_{n-1})(u_{n-2}+6u_{n-1}+u_n)(u_{n-1}+u_n) \\
\qquad\qquad (u_{n-1}+u_n)(u_{n-1}+3u_n)
\end{bmatrix}
\begin{Bmatrix}
u_{1,x} \\ u_{2,x} \\ u_{3,x} \\ u_{4,x} \\ \vdots \\ u_{n-2,x} \\ u_{n-1,x} \\ u_{n,x}
\end{Bmatrix}
$$

$$
= \frac{6\theta_E{}^D}{\mathrm{Re}}
\begin{bmatrix}
(\nu_1+\nu_2)-(\nu_1+\nu_2) \\
(\nu_1+\nu_2)-(\nu_1+2\nu_2+\nu_3) \\
\qquad (\nu_2+\nu_3)-(\nu_2+2\nu_3+\nu_4)(\nu_2+\nu_4) \\
\vdots \\
(\nu_{n-2}+\nu_{n-2})-(\nu_{n-3}+2\nu_{n-2}+\nu_{n-1})(\nu_{n-2}+\nu_{n-1}) \\
\qquad (\nu_{n-2}+\nu_{n-1})-(\nu_{n-2}+2\nu_{n-1}+\nu_n)(\nu_{n-1}+\nu_n) \\
\qquad\qquad -(\nu_{n-1}+\nu_n)(\nu_{n-1}+\nu_n)
\end{bmatrix}
\begin{Bmatrix}
u_1{}^i \\ u_2{}^i \\ u_3{}^i \\ \vdots \\ u^i_{n-2} \\ u^i_{n-1} \\ u_n{}^i
\end{Bmatrix}
+ \frac{6\theta_E}{\mathrm{Re}}
\begin{bmatrix}
(\nu_1+\nu_2)-(\nu_1+\nu_2) \\
(\nu_1+\nu_2)-(\nu_1+2\nu_2+\nu_3)(\nu_2+\nu_3) \\
\qquad (\nu_2+\nu_3)-(\nu_2+2\nu_3+\nu_4)(-\nu_3+\nu_4) \\
\vdots \\
(\nu_{n-3}+\nu_{n-2})-(\nu_{n-3}+2\nu_{n-2}+\nu_n)(\nu_{n-1}+\nu_n) \\
\qquad (\nu_{n-2}+\nu_{n-1})-(\nu_{n-2}+2\nu_{n-1}+\nu_n)(\nu_{n-1}+\nu_n) \\
\qquad\qquad -(\nu_{n-1}+\nu_n)(\nu_{n-1}+\nu_n)
\end{bmatrix}
\begin{Bmatrix}
u^{i+1}_1 \\ u^{i+1}_2 \\ u_3{}^2 \\ \vdots \\ u^{i+1}_{n-3} \\ u^{i+1}_{n-2} \\ u^{i+1}_{n-1} \\ u^{i+1}_n
\end{Bmatrix}
$$

$$
-2\theta^D_{EV}
\begin{bmatrix}
-(2v_1+v_2)(2v_1+v_2) \\
-(v_1+2v_2)(v_1-v_3)(2v_2+v_3) \\
\qquad -(v_2+2v_3)(v_2-v_4)(2v_3+v_4) \\
\vdots \\
-(v_{n-3}+2v_{n-2})(v_{n-3}-v_{n-1})(2v_n+v_{n-1}) \\
\qquad -(v_{n-2}+2v_{n-1})(v_{n-2}-v_n)(2v_{n-1}+v_n) \\
\qquad\qquad -(v_{n-1}+2v_n)(v_{n-1}+2v_n)
\end{bmatrix}
\begin{Bmatrix}
u_1{}^i \\ u_2{}^i \\ u_3{}^i \\ \vdots \\ u^i_{n-2} \\ u^i_{n-1} \\ u_n{}^i
\end{Bmatrix}
-2\theta_{EV}
\begin{bmatrix}
-(2v_1+v_2)(2v_1+v_2) \\
-(v_1+2v_2)(v_1-v_3)(2v_2+v_3) \\
\qquad -(v_2+2v_3)(v_2-v_4)(2v_3+v_4) \\
\vdots \\
-(v_{n-3}+2v_{n-2})(v_{n-3}-v_{n-1})(2v_n+v_{n-1}) \\
\qquad -(v_{n-2}+2v_{n-1})(v_{n-2}-v_n)(2v_{n-1}+v_n) \\
\qquad\qquad -(v_{n-1}+2v_n)(v_{n-1}+2v_n)
\end{bmatrix}
\begin{Bmatrix}
u^{i+1}_1 \\ u^{i+1}_2 \\ u^{i+1}_3 \\ \vdots \\ u^{i+1}_{n-2} \\ u^{i+1}_{n-1} \\ u^{i+1}_n
\end{Bmatrix}
$$

FIG. 1. FE method-tridiagonal convective term $uu_{,x}$. $\theta^D = 1 - \theta$, θ_E = weighting factor for diffusion term, θ_{EV} = weighting factor for convective term.

Equation (14) is a global representation of the momentum equation for a node at a normal location n and it contains values and derivatives at the adjacent nodes $n + 1$ and $n - 1$. If we introduce on the right-hand side of Eq. (14) a weighted average for the streamwise velocity u_n at the normal station n:

$$u_n = u_n^{i+1}\theta_e + u_n{}^i(1 - \theta_e), \tag{15}$$

(where $u_n{}^i$ are the known values and $u^i{}_n^{+1}$ are the values to be computed) and expand Eq. (14) for n nodes in the normal direction, a matrix representation of Fig. 1 is obtained. Thus, the initial value term is expressed in terms of three derivatives centered at point n. Herein, such a system of equations with tridiagonal matrix of the initial value term is defined as a consistent system.

Introducing into Eq. (13) approximations that are consistent with accuracy of the scheme and are exact in the limit as $dx \to 0$, $dy \to 0$, a lumping of the convective term $uu_{,x}$ is possible.

$$\{u\}_m^T \frac{l_m}{6} \begin{bmatrix} \begin{Bmatrix} 2 \\ 1 \end{Bmatrix} & \\ 0 & \begin{Bmatrix} 1 \\ 2 \end{Bmatrix} \end{bmatrix} \{u\}_{m,x} \, . \tag{16}$$

This procedure eliminates the necessity of inverting the initial value term matrix in order to solve for $u_{,x}$ terms.

After a global assemply, the following form is obtained:

$$\begin{aligned} &2(u_{n-1} + 4u_n + u_{n+1})\, u_{n,x} \\ &\quad = (6/\mathrm{Re})[(\nu_{n-1} + \nu_n)\, u_{n-1} - (\nu_{n-1} + 2\nu_n + \nu_{n+1})\, u_n + (\nu_n + \nu_{n+1})\, u_{n+1}] \\ &\quad\quad - 2\{-(v_{n-1} + 2v_n)\, u_{n-1} + (v_{n-1} - v_{n+1})\, u_n + (2v_n + v_{n+1})\, u_{n+1}\}. \end{aligned} \tag{17}$$

The approximations to the convective term $uu_{,x}$ can be introduced in many ways. This leads to several expressions for the coefficient of the lumped convective term $uu_{.x}$, $a_n{}^i$, $i = 1, 2, 3, 4$:

$$\begin{aligned} &\text{I.} \quad a_n{}^1 = 2(u_{n-1} + 4u_n + u_{n+1}), \\ &\text{II.} \quad a_n{}^2 = (3u_{n-1} + 6u_n + 3u_{n+1}), \\ &\text{III.} \quad a_n{}^3 = u_{n-1} + 10u_n + u_{n+1}\,, \\ &\text{IV.} \quad a_n{}^4 = 12u_n\,. \end{aligned} \tag{18}$$

After introducing a weighted average for velocity u_n (Eq. (15))), an equation with lumped version of the convective term in matrix form shown in Fig. 2, is obtained. An interpretation of the various forms (18) is provided in the next Section.

$$\begin{bmatrix} a_1{}^i & & & & & & & \\ & a_2{}^i & & & & & & \\ & & a_3{}^i & & & & & \\ & & & \cdot & & & & \\ & & & & a_k{}^i & & & \\ & & & & & \cdot & & \\ & & & & & & a^i_{n-1} & \\ & & & & & & & a_n{}^i \end{bmatrix} \begin{Bmatrix} u_{1,x} \\ u_{2,x} \\ u_{3,x} \\ u_{4,x} \\ \vdots \\ u_{n-2,x} \\ u_{n-1,x} \\ u_{n,x} \end{Bmatrix}$$

$$= \frac{6\theta_E{}^D}{\mathrm{Re}} \begin{bmatrix} (\nu_1+\nu_2) & -(\nu_1+\nu_2) & & & \\ (\nu_1+\nu_2) & -(\nu_1+2\nu_2+\nu_3) & (\nu_2+\nu_3) & & \\ & (\nu_2+\nu_3) & -(\nu_2+2\nu_3+\nu_4) & (\nu_3+\nu_4) & \\ & & \vdots & & \\ (\nu_{n-3}+\nu_{n-2}) & -(\nu_{n-3}+2\nu_{n-2}+\nu_{n-1}) & (\nu_{n-2}+\nu_{n-1}) & & \\ & (\nu_{n-2}+\nu_{n-1}) & -(\nu_{n-2}+2\nu_{n-1}+\nu_n) & (\nu_{n-1}+\nu_n) & \\ & & -(\nu_{n-1}+\nu_n) & (\nu_{n-1}+\nu_n) & \end{bmatrix} \begin{Bmatrix} u_1{}^i \\ u_2{}^i \\ u_3{}^i \\ \vdots \\ u^i_{n-2} \\ u^i_{n-1} \\ u_n{}^i \end{Bmatrix} + \frac{6\theta_E}{\mathrm{Re}} \begin{bmatrix} (\nu_1+\nu_2) & -(\nu_1+\nu_2) & & & \\ (\nu_1+\nu_2) & -(\nu_1+2\nu_2+\nu_3) & (\nu_2+\nu_3) & & \\ & (\nu_2+\nu_3) & -(\nu_2+2\nu_3+\nu_4) & (-\nu_3+\nu_4) & \\ & & \vdots & \cdot & \\ (\nu_{n-3}+\nu_{n-2}) & -(\nu_{n-3}+2\nu_{n-2}+\nu_{n-1}) & (\nu_{n-2}+\nu_{n-1}) & & \\ & (\nu_{n-2}+\nu_{n-1}) & -((\nu_{n-2}+2\nu_{n-1}+\nu_n) & (\nu_{n-1}+\nu_n) & \\ & & -(\nu_{n-1}+\nu_n) & (\nu_{n-1}+\nu_n) & \end{bmatrix} \begin{Bmatrix} u^{i+1}_1 \\ u^{i+1}_2 \\ u^{i+1}_3 \\ \vdots \\ u^{i+1}_{n-3} \\ u^{i+1}_{n-2} \\ u^{i+1}_{n-1} \\ u^{i+1}_n \end{Bmatrix}$$

$$-2\theta^D_{EV} \begin{bmatrix} -(2v_1+v_2)(2v_1+v_2) & & \\ -(v_1+2v_2)(v_1-v_3)(2v_2+v_3) & & \\ & -(v_2+2v_3)(v_2-v_4)(2v_3+v_4) & \\ & \vdots & \\ -(v_{n-3}+2v_{n-2})(v_{n-3}+v_{n-1})(2v_n+v_{n-1}) & & \\ & -(v_{n-2}+2v_{n-1})(v_{n-2}-v_n)(2v_{n-1}+v_n) & \\ & & -(v_{n-1}+2v_n)(v_{n-1}+2v_n) \end{bmatrix} \begin{Bmatrix} u_1{}^i \\ u_2{}^i \\ u_3{}^i \\ \vdots \\ u^i_{n-2} \\ u^i_{n-1} \\ u_n{}^i \end{Bmatrix} -2\theta_{EV} \begin{bmatrix} -(2v_1+v_2)(2v_1+v_2) & & \\ -(v_1+2v_2)(v_1-v_3)(2v_2+v_3) & & \\ & -(v_2+2v_3)(v_2-v_4)(2v_3+v_4) & \\ & \vdots & \\ -(v_{n-3}+2v_{n-2})(v_{n-3}-v_{n-1})(2v_n+v_{n-1}) & & \\ & -(v_{n-2}+2v_{n-1})(v_{n-2}-v_n)(2v_{n-1}+v_n) & \\ & & -(v_{n-1}+2v_n)(v_{n-1}+2v_n) \end{bmatrix} \begin{Bmatrix} u^{i+1}_1 \\ u^{i+1}_2 \\ u^{i+1}_3 \\ \vdots \\ u^{i+1}_{n-2} \\ u^{i+1}_{n-1} \\ u^{i+1}_n \end{Bmatrix}$$

FIG. 2. FE method-lumped convective term $uu_{,x}$. $\theta^D = 1 - \theta$, θ_E = weighting factor for diffusion term, θ_{EV} = weighting factor for convective term.

$$
\begin{bmatrix}
u_1 & & & & & & \\
& u_2 & & & & & \\
& & u_3 & & & & \\
& & & \ddots & & & \\
& & & & u_{n-2} & & \\
& & & & & u_{n-1} & \\
& & & & & & u_n
\end{bmatrix}
\begin{Bmatrix} u_{1,x} \\ u_{2,x} \\ u_{3,x} \\ \vdots \\ u_{n-3,x} \\ u_{n-2,x} \\ u_{n-1,x} \\ u_{n,x} \end{Bmatrix}
= \frac{\theta}{4\,\mathrm{Re}\,\Delta y^2}
\begin{bmatrix}
-(\nu_3-4\nu_2-\nu_1) & -8\nu_2 & (\nu_3+4\nu_2-\nu_1) & \\
& -(\nu_4-4\nu_3-\nu_2) & -8\nu_3 & (\nu_4+4\nu_3-\nu_2) \\
& & \vdots & \\
-(\nu_{n-1}-4\nu_{n-2}-\nu_{n-3}) & -8\nu_{n-2} & (\nu_{n-1}+4\nu_{n-2}-\nu_{n-3}) & \\
& -(\nu_n-4\nu_{n-1}-\nu_{n-2}) & -8\nu_{n-1} & (\nu_n+4\nu_{n-1}-\nu_{n-2})
\end{bmatrix}
\begin{Bmatrix} u_1^{i+1} \\ u_2^{i+1} \\ u_3^{i+1} \\ \vdots \\ u_{n-2}^{i+1} \\ u_{n-1}^{i+1} \\ u_n^{i+1} \end{Bmatrix}
$$

$$
+ \frac{\theta_0}{4\,\mathrm{Re}\,\Delta y^2}
\begin{bmatrix}
-(\nu_3-4\nu_2-\nu_1) & -8\nu_2 & (\nu_3+4\nu_2-\nu_1) & \\
& -(\nu_4-4\nu_3-\nu_2) & -8\nu_3 & (\nu_4+4\nu_3-\nu_2) \\
& & \vdots & \\
-(\nu_{n-1}-4\nu_{n-2}-\nu_{n-3}) & -8\nu_{n-2} & (\nu_{n-1}+4\nu_{n-2}-\nu_{n-3}) & \\
& -(\nu_n-4\nu_{n-1}-\nu_{n-2}) & -8\nu_{n-1} & (\nu_n+4\nu_{n-1}-\nu_{n-2})
\end{bmatrix}
\begin{Bmatrix} u_1^{\,i} \\ u_2^{\,i} \\ u_3^{\,i} \\ \vdots \\ u_{n-1}^i \\ u_{n-1}^i \\ u_n^{\,i} \end{Bmatrix}
$$

$$
+ \frac{\theta}{2\Delta y}
\begin{bmatrix}
v_2 & 0 & -v_2 & & & & \\
& v_3 & 0 & -v_3 & & & \\
& & & \ddots & & & \\
& & & & v_{n-2} & 0 & -v_{n-2} \\
& & & & & v_{n-1} & 0 \quad -v_{n-1}
\end{bmatrix}
\begin{Bmatrix} u^{i+1} \\ u_2^{i+1} \\ u_3^{i+1} \\ u_4^{i+1} \\ \cdot \\ u_{n-3}^{i+1} \\ u_{n-2}^{i+1} \\ u_{n-1}^{i+1} \\ u_n^{i+1} \end{Bmatrix}
+ \frac{\theta_0}{2\Delta y}
\begin{bmatrix}
v_2 & 0 & -v_2 & & & & \\
& v_3 & 0 & -v_3 & & & \\
& & & \ddots & & & \\
& & & & v_{n-2} & 0 & -v_{n-2} \\
& & & & & v_{n-1} & 0 \quad -v_{n-1}
\end{bmatrix}
\begin{Bmatrix} u_1^{\,i} \\ u_2^{\,i} \\ u_3^{\,i} \\ \vdots \\ u_{n-3}^i \\ u_{n-2}^i \\ u_{n-1}^i \\ u_n^{\,i} \end{Bmatrix}
$$

FIG. 3. FD method, Crank–Nicolson and implicit. $\theta_D = 1-\theta$; $\theta = 1/2$, Crank–Nicolson; $\theta = 1$, implicit.

2.2. *Overview of the Finite-Difference Algorithm*

A second-order accurate finite-difference representation of the streamwise momentum equation takes the form:

$$\begin{aligned} u_n{}^i u^i_{n,x} = \frac{\nu_n}{\mathrm{Re}\,\Delta y^2} \Big\{ & \theta[u^{i+1}_{n+1} - 2u^{i+1}_n + u^{i+1}_{n-1}] \\ & + \frac{1}{4\,\mathrm{Re}\,\Delta y^2}(\nu_{n+1} - \nu_{n-1})[\theta(u^{i+1}_{n+1} - u^{i+1}_{n-1}) + (1-\theta)(u^i_{n+1} - u^i_{n-1})] \\ & + (1-\theta)[u^i_{n+1} - 2u_n{}^i + u^i_{n-1}]\Big\} \\ & - \frac{v_n}{2\Delta y}\{\theta[u^{i+1}_{n+1} - u^{i+1}_{n-1}] + (1-\theta)[u^i_{n+1} - u^i_{n-1}]\}. \end{aligned} \tag{19}$$

The finite-difference form of the momentum equation in a matrix form appears in Fig. 3. Distinction is made between the weighting factor θ_{ev} for the convective term and θ_e for the diffusion term. Viewing Figs. 1, 2, and 3, it can be seen that the largest differences occur in the handling of the nonlinear initial value term, and these differences are primarily responsible for the variation in accuracies of the various methods.

The differences in treating the initial value term $uu,_x$ in the assembled matrix equation can be traced back to the differences in the integration formula resulting from the integration over a subinterval:

$$\int_0^l uu_{,x}\,dy = (l/12)\{u\}^T\{A\}\{u\}_{,x}, \tag{20}$$

where $\{u\}^T\{A\} = a$, and

$$\begin{aligned} \{A\}^T_{\mathrm{I}} &= \{2 \quad 8 \quad 2\}, \\ \{A\}^T_{\mathrm{III}} &= \{1 \quad 10 \quad 1\}, \\ \{A\}^T_{\mathrm{IV}} &= \{0 \quad 12 \quad 0\}, \end{aligned} \tag{21}$$

are identified with schemes I, III, and IV, respectively Eq. (18). Thus, the fomula for Scheme I:

$$(l/12)\{u\}^T\{A\}_{\mathrm{I}} = (l/12)(2u_{n-1} + 8u_n + 2u_{n+1}), \tag{22}$$

is equivalent to linear approximation for u within the subinterval. This formula is exact for $u\alpha y$, which can be easily verified.

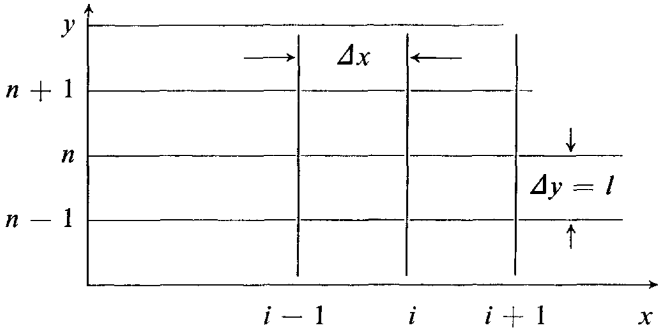
Scheme III is identified with quadratic or cubic approximations for u: $u\alpha y^2$ or $u\alpha y^3$.

Scheme IV is simply a result of application of the trapezoidal rule. The inference of these analogies has been confirmed by Strang, [5].

2.3. *The Implicit Integration Algorithm*

The finite-element and finite-difference equations presented in matrix form in Figs. 1, 2, and 3, have to be rearranged to a form suitable for numerical solution, in particular, to a tridiagonal form. Equations of this form are solved easily by an efficient algorithm. The finite element representation of equations with a consistent matrix, e.g., Eq. (14), can be rearranged by replacing the streamwise derivative $u_{,x}$ by a forward finite difference

$$u_{n,x} = \frac{u_n^{i+1} - u_n{}^i}{\Delta x} \tag{23}$$



Introducing on the right-hand side of Eq. (14), a weighted average for u_n (Eq. (15)), we obtain a tridiagonal equation for the unknown values u_n^{i+1}:

$$u_{n-1}^{i+1} a_n + u_n^{i+1} b_n + u_{n+1}^{i+1} c_n = d_n\,, \tag{24}$$

where

$$\begin{aligned}
A_n &= u_{n-1} + u_n\,,\\
B_n &= u_{n-1} + 6u_n + u_{n+1}\,,\\
C_n &= u_n + u_{n+1}\,,\\
a_n &= A_n - (6\Delta x\theta_e/\mathrm{Re})(\nu_{n-1} + \nu_n) - 2\Delta x\theta_{ev}(v_{n-1} + 2v_n),\\
b_n &= B_n + (6\Delta x\theta_e/\mathrm{Re})(\nu_{n-1} + 2\nu_n + \nu_{n+1}) + 2\Delta x\theta_{ev}(v_{n-1} - v_{n+1}),\\
c_n &= C_n - (6\Delta x\theta_e/\mathrm{Re})(\nu_n + \nu_{n+1}) + 2\Delta x\theta_{ev}(2v_n + v_{n+1}),\\
d_n &= A_n u_{n-1}^i + B_n u^i + C_n u_{n+1}^i + (6\Delta x\theta_e{}^d/\mathrm{Re})[(v_{n-1} + \nu_n)\, u_{n-1}^i\\
&\quad - (\nu_{n-1} + 2\nu_n + \nu_{n+1})\, u_n{}^i + (\nu_n + \nu_{n+1})\, u_{n+1}^i] - 2\Delta x\theta_{ev}^d\\
&\quad \times [-(v_{n-1} + 2v_n)\, u_{n-1}^i + (v_{n-1} - v_{n+1})\, u_n{}^i + (2v_n + v_{n+1})\, u_{n+1}^i],
\end{aligned} \tag{25}$$

and u_n in A_n, B_n, and C_n is expressed by a weighted average

$$u_n = u_n'^{i+1}\theta + (1-\theta)\, u_n{}^i, \tag{26}$$

with $u_n'^{i+1}$ taken from previous iteration.

The system of algebraic equations of tridiagonal form has a solution of the form:

$$u_n^{i+1} = u_{n+1}^{i+1} E_n + F_n\,. \tag{27}$$

Making substitutions similar to those for Eq. (14), shown in Eqs. (15) and (23), Eq. (17) reduces to a tridiagonal form (Eq. 24) with the following coefficients:

$$\begin{aligned}
a_n &= -(6\Delta x\theta_e/\mathrm{Re})(\nu_{n-1} + \nu_n) - 2\Delta x\theta_{ev}(v_{n-1} + 2v_n),\\
b_n &= a + (6\Delta x\theta_e/\mathrm{Re})(\nu_{n-1} + 2\nu_n + \nu_{n+1}) + 2\Delta x\theta_{ev}(v_{n-1} - v_{n+1}),\\
c_n &= -(6\Delta x\theta_e/\mathrm{Re})(\nu_n + \nu_{n+1}) + 2\Delta x\theta_{ev}(2v_n + v_{n+1}),\\
d_n &= au_{1,n} + (6\Delta x\theta_e{}^d/\mathrm{Re})\\
&\quad \times [(\nu_{n-1} + \nu_n)\, u_{n-1}^i - (\nu_{n-1} + 2\nu_n + \nu_{n+1})\, u_n{}^i + (\nu_n + \nu_{n+1})\, u_{n+1}^i]\\
&\quad - 2\Delta x\theta_e{}^d[-(v_{n-1} + 2v_n)\, u_{n-1}^i + (v_{n-1} - v_{n+1})\, u_n{}^i + (2v_n + v_{n+1})\, u_{n+1}^i],
\end{aligned} \tag{28}$$

where a (Eq. (18)) is expressed in terms of u_n (Eq. (26)). Similarly, the finite-difference equation (19) reduces after introducing Eqs. (15), and (23) to a tridiagonal form with coefficients:

$$\begin{aligned}
a_n &= -\left(\frac{\nu_n}{\mathrm{Re}\,\Delta y} + \frac{v_n}{2}\right)\frac{\Delta x\theta}{\Delta y} + \frac{\Delta x\theta}{4\,\mathrm{Re}\,\Delta y^2}\,(\nu_{n+1} - \nu_{n-1}),\\
b_n &= u_n + \frac{\nu_n\,\Delta x}{\mathrm{Re}\,\Delta y^2}\,2\theta,\\
c_n &= \left(-\frac{\nu_n}{\mathrm{Re}\,\Delta y} + \frac{v_n}{2}\right)\frac{\Delta x\theta}{\Delta y} - \frac{\Delta x\theta}{4\,\mathrm{Re}\,\Delta y^2}\,(\nu_{n+1} - \nu_{n-1}),\\
d_n &= u_n u_{1,n} + \frac{\nu_n\,\Delta x}{\mathrm{Re}\,\Delta y^2}\,[\theta_D(u_{n+1}^i - 2u_n{}^i + u_{n-1}^i)]\\
&\quad - \frac{v_n\,\Delta x}{2\Delta y}\,\theta_D[u_{n+1}^i - u_{n-1}^i]\\
&\quad + \frac{\Delta x}{4\,\mathrm{Re}\,\Delta y^2}\,(\nu_{n+1} - \nu_{n-1})\,\theta_D(u_{n+1}^i - u_{n-1}^i),
\end{aligned} \tag{29}$$

where

$$\theta_D = (1-\theta). \tag{30}$$

We have for

$$\theta = 1, \qquad \text{implicit method,} \tag{31}$$

and

$$\theta = \tfrac{1}{2}, \qquad \text{Crank–Nicolson method.}$$

The condensed form of the finite-element equation (Eq. (17)) with various expressions for the coefficient a also can be solved without using the tridiagonal form by introducing Eqs. (23) and (26) and solving for u_n^{i+1} in terms of values $u_n'^{i+1}$ from a previous iteration. Combined with iteration on u and v, the values of u^{i+1} are essentially identical to those obtained by using the tridiagonal algorithm.

2.4. *Continuity Equation*

Having computed values of u_n^{i+1} by Eq. (27), the normal velocity v is computed from the continuity equation:

$$\frac{\partial}{\partial x}(u) + \frac{\partial}{\partial y}(v) = 0. \tag{32}$$

The continuity equation is of the form suggested by Krause [6] and was derived using the identity

$$v_n^{i+(1/2)} = \tfrac{1}{2}(v_n{}^i + v_n^{i+1}) + o(\Delta x^2). \tag{33}$$

The continuity equation at the mid-station $(i + \frac{1}{2})$ used in an iterative way for the incompressible flow has the form:

$$v_n^{i+(1/2)} = v_{n-1}^{i+(1/2)} + (\Delta y/2\Delta x)(u_n{}^i - u_n^{i+1} + u_{n-1}^i - u_{n-1}^{i+1}). \tag{34}$$

The normal velocity distribution from the continuity equation is obtained this way for both finite-element and finite-difference solutions of the boundary layer equation, therefore, the effect of a different solution procedure for values of transverse velocity v has been minimized for these studies. The momentum equation and the continuity equation are solved sequentially in an iterative manner until u_n^{i+1} become stationary.

3. Numerical Results, Accuracy and Convergence

3.1. *Evaluation Bases*

The finite-element and finite difference algorithms presented in the preceding section were programmed into a single computer code to obtain the incompressible laminar boundary layer solution for flat plate flow in physical coordinates.

Accuracy and convergence was assessed by comparison with the Blasius solution [7].

The convergence criterion applied to u is $\sum_{i=1}^{n} | u_a - u_c | < \epsilon$, where the summation was applied over the boundary layer thickness and $\epsilon = 0.0001$–0.005 was satisfactory in all cases and resulted in stable results with two iterations, including the continuity equation. Previous values of u and v were used for starting computations at the next x station.

The accuracy was evaluated by computing the error in the streamwise velocity u and in the skin friction C_f. Three error norms are evaluated:

1. An absolute velocity error defined as a maximum deviation between the computed, u_c, and exact value of u anywhere within the boundary layer thickness δ, u_a:

$$E_{u\max} = | u_c - u_a |/u_\infty$$

An average of 10 values in the streamwise direction x, downstream of the relaxation zone, over the range of computations is used.

2. A mean square (L_2) error of u defined as

$$\bar{E}_u = \left[\int_0^\delta E^2(y)\, dy\right]^{1/2}$$

is computed for some selected cases for comparison with $E_{u\max}$.

3. A relative error of the skin friction coefficient

$$E_{c_f} = | c_{f_a} - c_{f_c} |/c_{f_a},$$

was computed and was used as a criterion for evaluation. The usual definition of the skin friction coefficient is used:

$$c_f = \frac{\mu(\partial u/\partial y)|_w}{(1/2)\,\rho_\infty u_\infty{}^2}.$$

The slope $(\partial u/\partial y) \mid w$ was evaluated by a four-point finite-difference expression with $o(\Delta y^3)$.

The boundary layer computations were performed for incompressible flow of air for the following flow conditions:

$$\begin{aligned} p_\infty &= 0.2 \times 10^4 \text{ lb/ft}^2, \\ T_\infty &= 530^\circ\text{R}, \\ U_\infty &= 300 \text{ ft/sec}, \\ M_\infty &= 0.266. \end{aligned}$$

The skin friction error was evaluated at $x = 0.75$ ft and at $\text{Re}_x = 0.135 \times 10^7$. All dimensions are in feet.

All computations were performed with single precision using IBM 360 computer. The computations are initiated using values of u and v obtained from the Blasius solution at the streamwise coordinate $x = 0.3$ ft. It was observed that computations over a certain number of steps have to be performed to reach a relaxation of the computed values. Due to a finite number of steps across the boundary layer, up to 10 steps were required, depending upon the method, for the computations to stabilize, and the largest number of steps was required by the finite-element consistent method. The velocity error across the boundary layer has a parabolic distribution with a peak occurring approximately in the middle of the boundary layer.

The error was evaluated using results obtained after the relaxation zone was reached. In general, the accuracy is better with decreasing dy steps. Computations of the integral parameters, boundary layer thickness δ, displacement δ^*, and the momentum thickness θ, were computed for various methods for different sets of values of dx and dy, and were compared with exact Blasius values. In general, the agreement is good, and the deviations from the Blasius values decrease as dy step is reduced. The largest deviations seem to be inherent with the finite-element

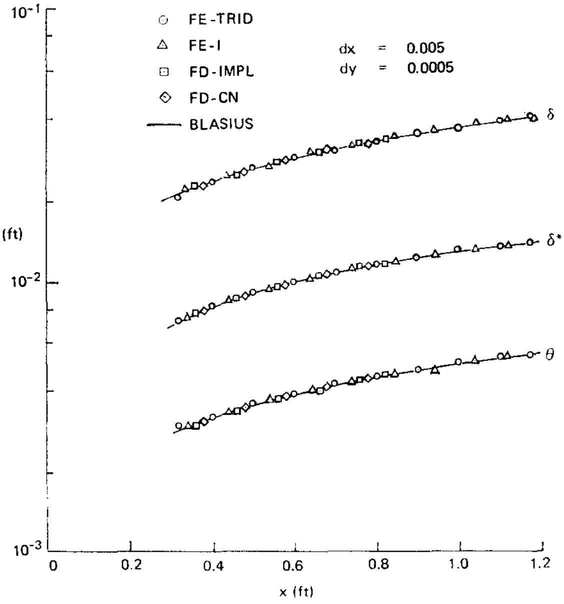


FIG. 4. Boundary layer, displacement thickness, and momentum thickness distribution for FE and FD methods.

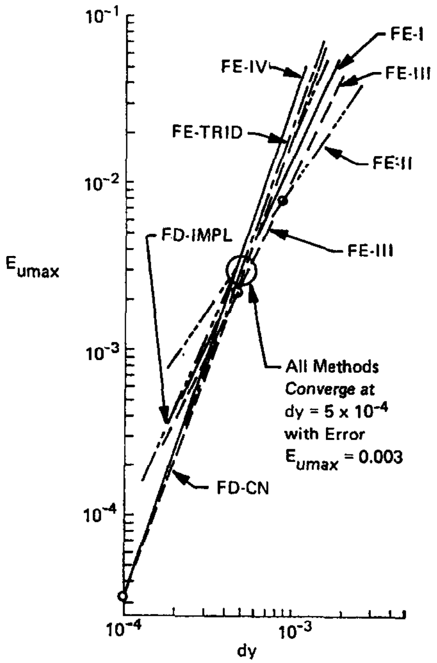


FIG. 5. Error map for FE and FD methods.

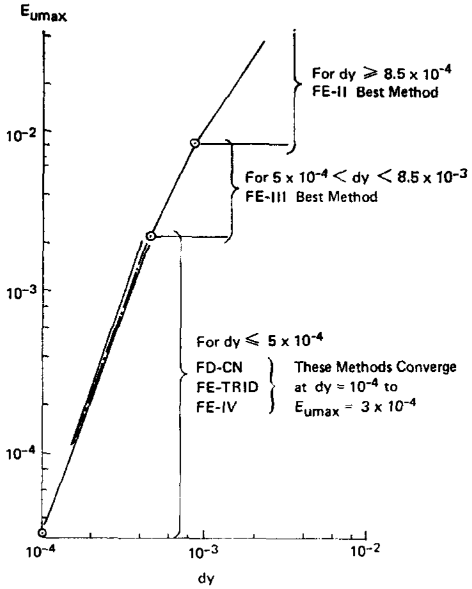


FIG. 6. Best methods as a function of *dy*.

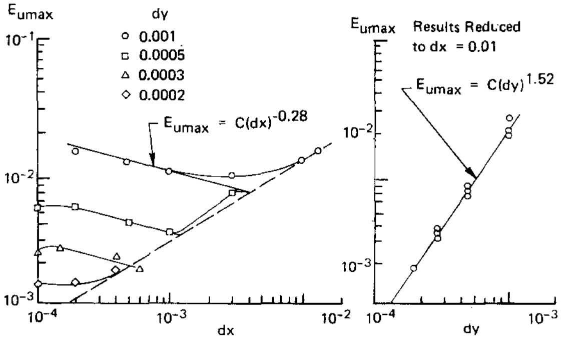
tridiagonal method. The agreement improves with the streamwise distance x, which confirms the existance of a relaxation zone. The error seems to be greater for δ and δ^*, than for θ (Fig. 4).

A series of computations for each method was performed for various step sizes in the streamwise and normal directions dx and dy. It was established that the accuracy of a given solution, and the convergence with discretization refinement depends on the magnitude of dx and dy, and their ratio in a distinct manner for each method. The trends of various methods are discussed in detail for each method of solution and a critical comparison of the merits of each method is made. A summary of the results for all finite-element and finite-difference methods and a comparison of correlations for computational errors is shown in Fig. 5. It reveals that all investigated methods converge at $dy = 0.0005$, to an error $E_{u\max} = 0.003$, and at this condition, no method offers an advantage over any other method. However, some methods show advantages of smaller error for $dy < 0.0005$, or $dy > 0.0005$. Thus, an optimum method will depend on the magnitude or range of dy values used. A delineation of ranges of dy with optimum performance of computational methods is shown in Fig. 6. It is noticeable that for $dy \leqslant 0.0005$, the Crank–Nicolson and the finite-element method Scheme IV with the finite-element tridiagonal convective term are of comparable accuracy, especially for decreasing dy.

3.2. *Finite-Element Results*

Computations were performed for a number of different values of dx and dy in order to establish the effect of step sizes. Some typical computational results for the finite-element method are presented below.

Figure 7a shows the maximum error in u, $E_{u\max}$, for the finite-element method Scheme II for $dy = 0.0002$ to 0.001 as a function of dx. The error $E_{u\max}$ decreases



a. Influence of dx on $E_{u\max}$

b. Influence of dy on $E_{u\max}$

with increasing *dx* step as expressed by the relation $E_{u\max} \sim C(dx)^{-0.28}$. The variation of *dx* step results in a decrease of $E_{u\max}$ with increasing magnitude of *dx*, a minimum value at certain values of *x* and an increase $E_{u\max}$ for increasing value of *dx*.

The minimum value of $E_{u\max}$ shifts to higher *dx* step for increasing *dy* values (Fig. 7a).

This behavior is due to varying contribution of the truncation and round-off error as *dx* is changed. At small *dx* values, the round-off error is predominant, whereas the truncation error is negligible. As *dx* goes up, the round-off error becomes less significant and the overall error decreases. A minimum of error occurs at *dx* values corresponding to negligible round-off error and small truncation error. As *dx* increases, the round-off error becomes less pronounced and the contribution of the truncation error increases. Thus, the interplay of the round-off and the truncation error results in two branches of the error curve having negative and positive slope with respect to *dx* coordinate. The branch having negative slope is dominated by the strong effect of the round-off error and the positive slope branch is dominated by the truncation error. Figure 7b shows the error as a function of *dy* (all results were reduced to $dx = 0.01$). The dependence on *dy* is expressed by a power relation:

$$E_{u\max} = C(dy)^{1.52}.$$

The magnitude of steps *dx* and *dy* and their ratio *dx*/*dy* have a strong influence on error and on the convergence of the computation scheme. Figure 8 indicates regions where error and convergence are satisfactory and regions with *dx*/*dy*

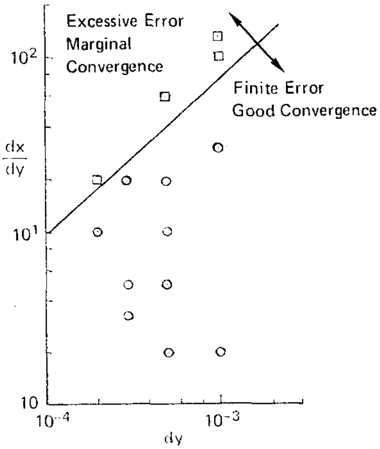
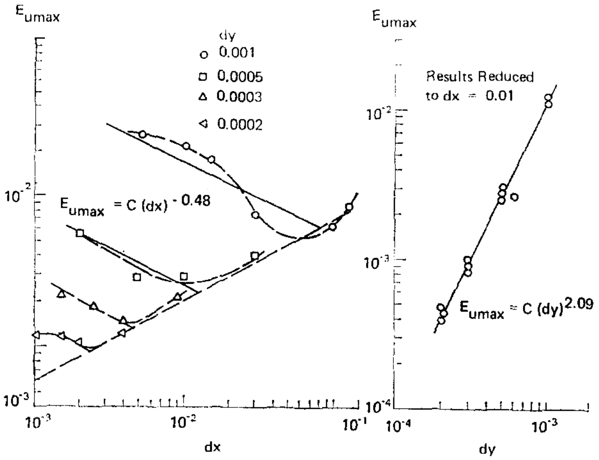
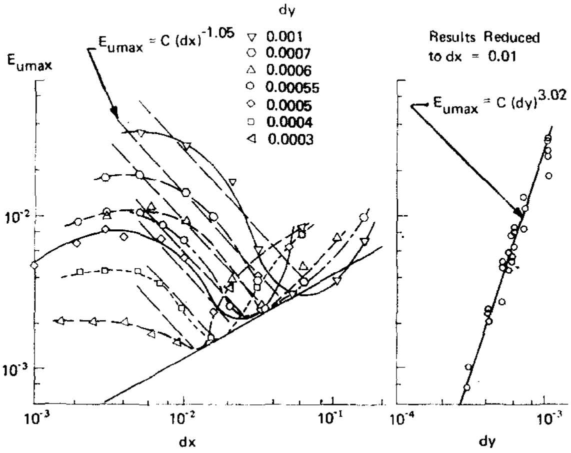


FIG. 8. Effect of *dx* and *dy* on error and covergence. FE method, Scheme II.



a. Influence of dx on E_{umax} b. Influence of dy on E_{umax}

FIG. 9. Maximum error $E_{u\max}$, FE method, Scheme III.



a. Influence of dx on E_{umax} b. Influence of dy on E_{umax}

FIG. 10. Maximum error $E_{u\max}$, FE method, Scheme IV.

ratio, where the error becomes excessive and convergence is marginal. This behavior sets a practical limit for increasing the step size in the streamwise direction.

Scheme III leads to results of error having a dx dependence of the form $E_{u\max} = C(dx)^{0.48}$ (Fig. 9a). The dependence of error on dy for Scheme III has the form (Fig. 9b).

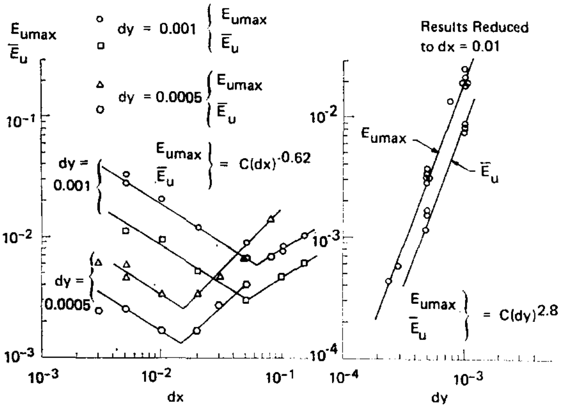
$$E_{u\max} = C(dy)^{2.09}.$$

Scheme IV exhibits a strong dx dependence of error and very distinct minimum of $E_{u\max}$ for each dy value from $dy = 0.0003$ to 0.001: $E_{u\max} = C(dy)^{-1.05}$ (Fig. 10a). The dependence on dy is very strong as expressed by power law (Fig. 10b):

$$E_{u\max} = C(dy)^{3.02}.$$

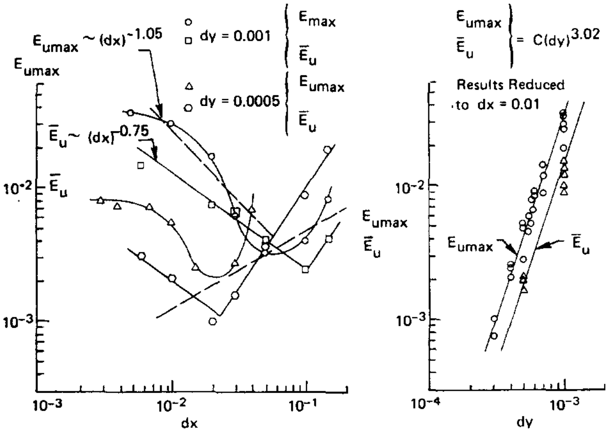
3.3. *Comparison to Finite Difference Results*

Computational results were also performed with the same computer code using both implicit and Crank–Nicolson, finite-difference method for comparison.



a. Influence of dx on E_{umax} and $\bar{E}_u$ b. Influence of dy on E_{umax} and $\bar{E}_u$

FIG. 11. Maximum error $E_{u\max}$ and $\bar{E}_u$. FE method, Crank–Nicolson.



a. Influence of dx on E_{umax} and $\bar{E}_u$ b. Influence of dy on E_{umax} and $\bar{E}_u$

FIG. 12. Maximum error $E_{u\max}$ and $\bar{E}_u$. FE method, Scheme IV.

The solutions for Crank–Nicolson method in Fig. 11a show a distinct minimum of $E_{u\max}$ at $dx = 0.015$ for $dy = 0.0005$ and at about $dx = 0.05$ for $dy = 0.001$. A mean square error $\bar{E}_u$ plotted for comparison in the same figure reveals similar dx dependence, although the magnitude of $\bar{E}_u$ is about $\frac{1}{2}$ of $E_{u\max}$. This is also noticeable in Fig. 11b, where the dy dependence is characterized by the same slope: $\bar{E}_{u\max}$, $\bar{E}_u \sim (dy)^{2.8}$.

The mean square error correlation for the finite-element method, Scheme IV superimposed for comparison with $E_{u\max}$ in Fig. 12a, has a form $\bar{E}_u = C(dx)^{-0.75}$ as compared with correlation for maximum error $E_{u\max} = C(dx)^{-1.05}$. The dy dependence displays a mean square error about twice lower than $E_{u\max}$, but follows the same slope $\bar{E}_u = C(dy)^{3.02}$ (Fig. 12b). A comparison of the finite-element solutions for Scheme IV with Crank–Nicolson solutions are shown in Fig. 13. At $dy = 0.0003$, the maximum error $E_{u\max}$ is about 20 % higher, and the mean square error E_u is about 30 % higher for finite-element method than the

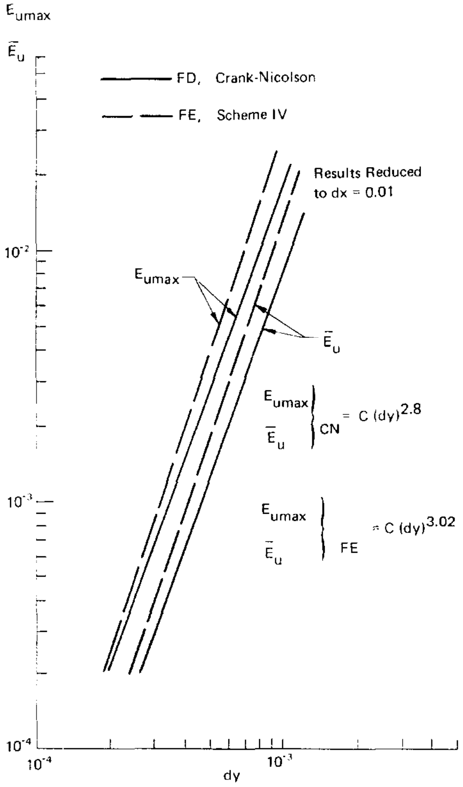
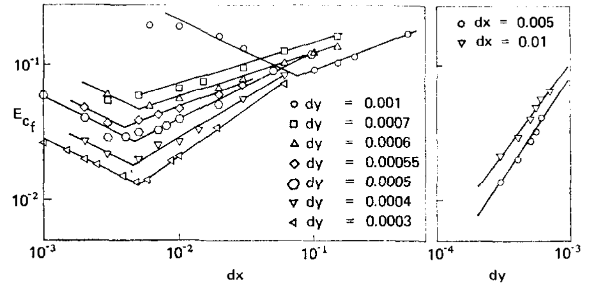


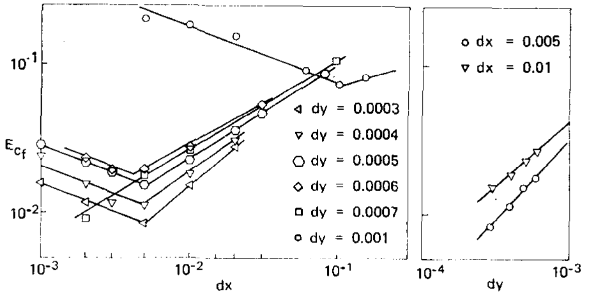
FIG. 13. Maximum error $E_{u\max}$ and $\bar{E}_u$ for FE and FD methods.



a. Influence of dx on E_{c_f}

b. Influence of dy on E_{c_f}

FIG. 14. Error of skin friction E_{c_f}. FE method, Scheme IV.



a. Influence of dx on E_{c_f}

b. Influence of dy on E_{c_f}

FIG. 15. Error of skin friction E_{c_f}. FD method, Crank–Nicolson.

error for the Crank–Nicolson method. This difference is diminishing as dy step decreases and is eventually reversed due to higher power relation for the finite-element method.

Thus so far, an error of u was evaluated and discussed. In order to assess the direct effect of an error of u on a quantity of considerable practical importance, an error of the skin friction c_f, E_{c_f}, was computed. The x-dependence of E_{c_f} for the finite-element method, Scheme IV, is marked by a minimum that occurs at $dx = 0.005$ for all dy values (Fig. 14a). A strong dx-dependence is noticeable for $dx \leqslant 0.005$ while a weaker dependence on dy occurs for $dx > 0.005$. At about $dx = 0.1$, the error seems to reach the same value of about 10% regardless of dy values. The effect of dy depends on the dx step as is indicated in Fig. 14b. Higher error occurs for $dx = 0.01$ than for $dx = 0.005$ and the correlation for $dx = 0.01$ has a smaller slope due to convergence of all lines for $dx > 0.005$. Therefore, for meaningful comparison of computational results, the magnitude of steps dx and dy must be stated. Similar results for E_{c_f} were obtained for Crank–Nicolson method in Fig. 15. A weaker effect of dy for $dx > 0.005$ is noticeable and practically no dy effect at $dx = 0.1$. The same qualitative effect of dy on E_{c_f} is shown in

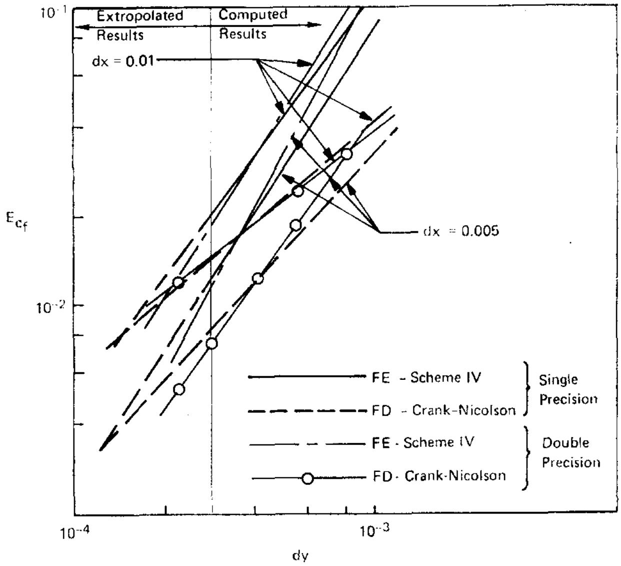


FIG. 16. Error of skin friction E_{c_f}. FE and FD methods (four-point differentiation formula, single and double precision).

Fig. 15b. A comparison of results for the finite element IV and Crank–Nicolson methods of Fig. 16 indicates higher error for $dy < 0.00015$ for the finite-difference method and lower error for finite-difference method for $dy > 0.00015$ (all data were extrapolated for $dy < 0.0003$).

To substantiate the effect of the steps dx and dy on the error E_u and E_{c_f}, and the postulate that there exists a dx value for which the error is minimized and results as an interplay of the round-off and truncation error, a systematic study of the error was made. This study was based on E_{c_f} as computed by various differentiation formulas in single and double precision. The test function selected for this purpose was $f = \sin(y)$, and three differentiation formulas were used: three-point, and four-point finite-difference formulas and cubic-spline differentiation formula. The following finite-difference formulas were used:

1. Three-point derivative at point 1:

$$f_{x,1} = (1/2\,dx)(-3f_1 + 4f_2 - f_3) + (1/3)(dx)^2 f_{xxx,1}$$

2. Four-Point derivative at point 1:

$$f_{x,1} = (1/6\,dx)(2f_4 - 9f_3 + 18f_2 - 11f_1) + (1/4)(dx)^3 f_{xxxx,1}$$

The derivatives $(df/dw)\,|_w$ computed by various methods in single precision are shown in Fig. 17. The three-point formula is indeed of second order provided $dy \geqslant 0.006$. For smaller step sizes, the error is established at about 10^{-6} and one

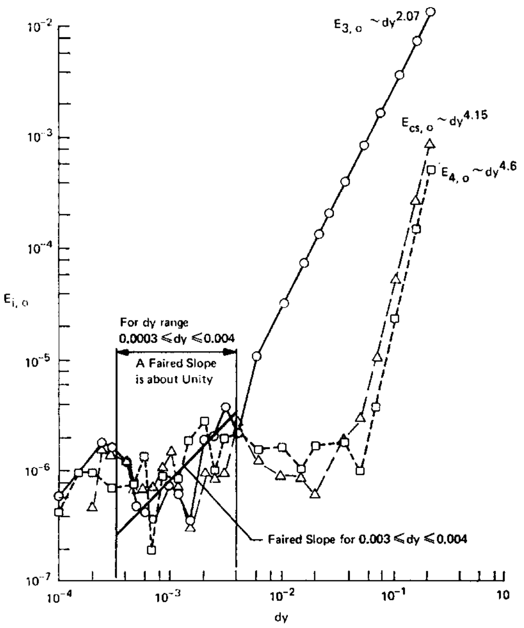


FIG. 17. Derivative error study, single precision. Influence of dy and precision on E_{c_f} for various differentiation formulas. $f = \sin y$; $df/dy = f_e' = \cos y$ = exact derivative; f_{ci} = derivative computed by method i; i = 3–3-point finite difference = 4–4-point finite difference = cs-cubic spline. $E_{i,0} = E_{i,y \to 0} \,|f_{ci}' - f_e'|/f_e'$ = relative error of derivative.

can fair a slope of about unity for the range $0.0003 < dy < 0.004$. The four-point formula and cubic-spline methods are of the order 4 or more based on computations for $dy \geqslant 0.05$ and the error fluctuates about a value 10^{-6} for $dy < 0.05$. The effect of the double precision is brought out in Fig. 18. The three-point formula and the cubic-spline formulas are of the second and fourth order, respectively, for steps down to $dy = 10^{-4}$. However, the four-point formula is of fourth order for $dy \geqslant 0.04$ and results in an error about 2.3×10^{-7} for $dy < 0.04$. By using double precision, the expected accuracy of three-point and cubic-spline formulas is extended to very small dy steps as a result of the effects on the truncation and round-off errors. After having qualified the accuracy of three differentiation formulas using a test function $f = \sin(y)$, it is of interest to obtain similar comparisons of skin friction error for the finite-difference and finite-element methods.

The results for the Crank–Nicolson method are shown in Fig. 19, and a number of comments can be made. For $dy = 0.006$ and double precision, the three-point formula shows $Ec_{f\min}$ at $dx = 0.03$. For the four-point equation E_{c_f} decreases with dx and the error is reduced from 0.2 to 0.15 at $dx = 10^{-3}$. The cubic-spline formula reduces the error with decreasing dx down to $dx = 10^{-3}$ with the same

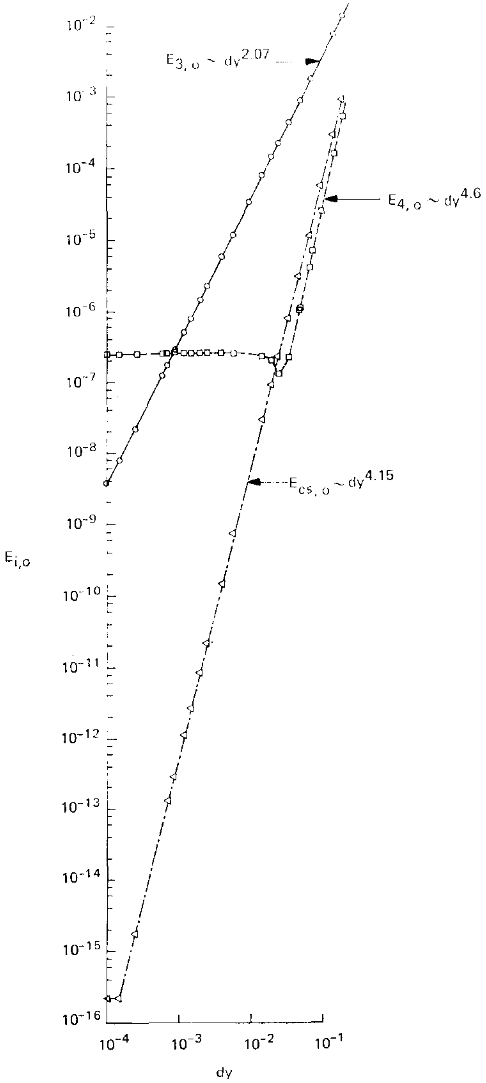


FIG. 18. Derivative error study, double precision. Influence of dy and precision on E_{f_c} for various differentiation formulas. $f = \sin y$; $dt/dy = f_e' = \cos y$ = exact value; f_{c_i} = derivative computed by method i; i = 3-3-point finite difference = 4-4-point finite difference = *cs*-cubic spline. $E_{i,0} = E_{i,v=0} = |f_{ci}' - f_e'|/f_e'$ = relative error of derivative.

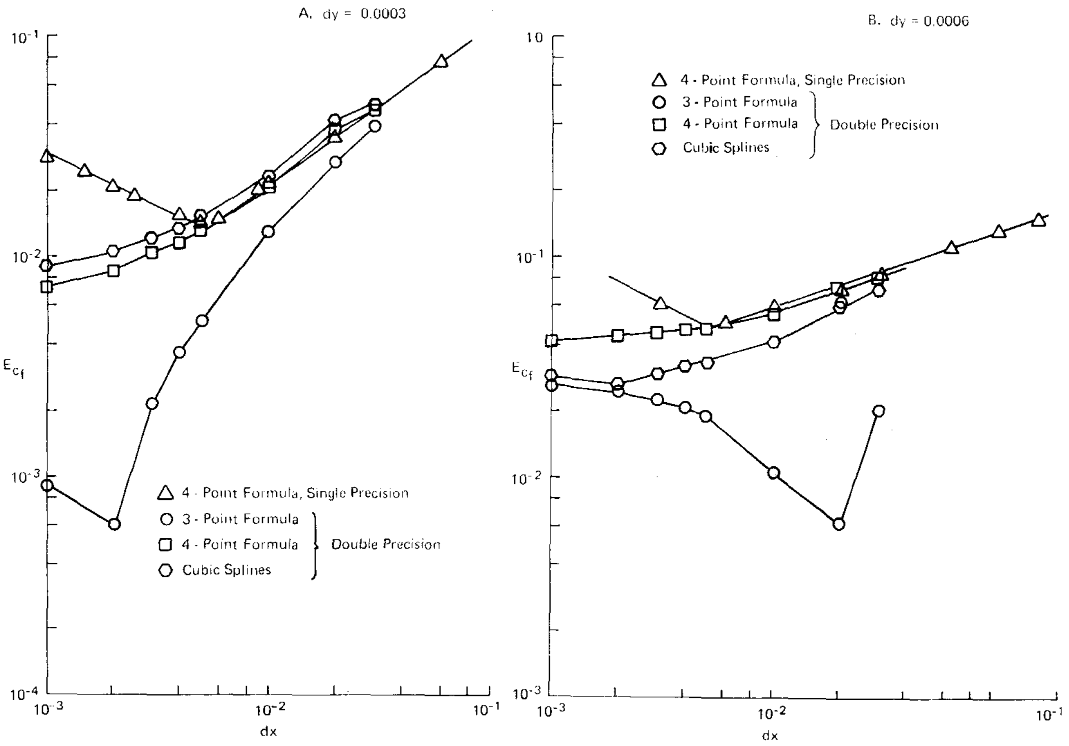


FIG. 19. Error of skin friction E_{c_f}. Finite-difference method, Crank–Nicolson. Influence of dx and precision on E_{c_f} for various differentiation formulas.

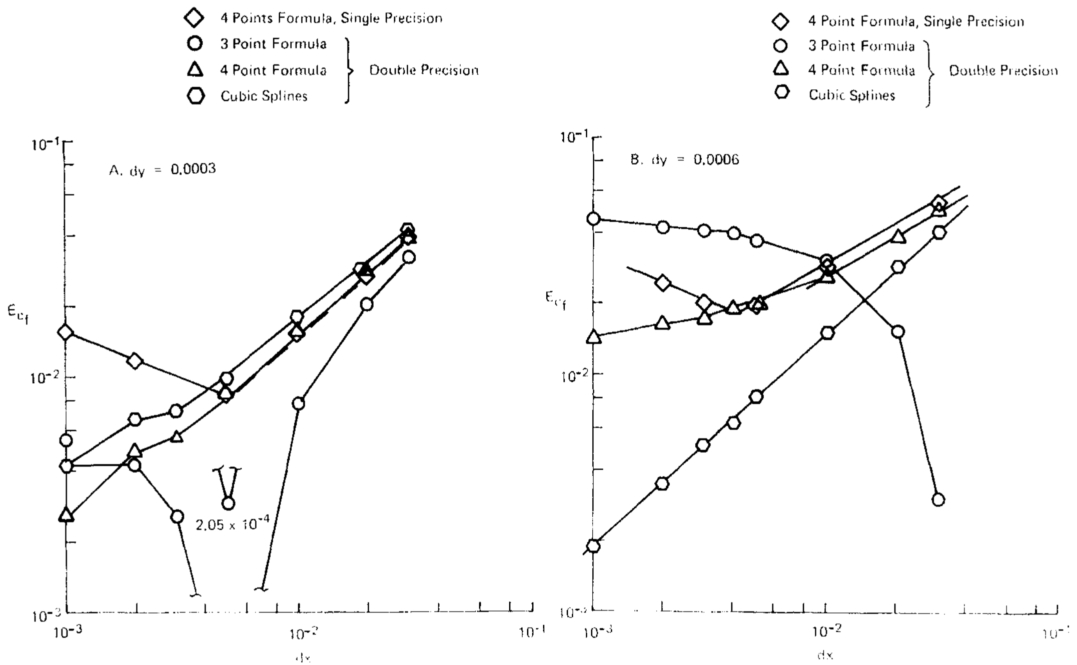


FIG. 20. Error of skin friction E_{c_f}. Finite-element method, Scheme IV. Influence of dx and precision on E_{c_f} for various differentiation formulas.

slope of the error curve. For $dy = 0.0003$, the three-point formula, E_{c_f} reaches a minimum at $dx = 0.005$ and the error goes up for $0.005 \leqslant dx \leqslant 0.005$. The error is substantially lower than for the other formulas. The four-point expression extends the results for error to $dx = 10^{-3}$ with the same error curve slope, with the error decreasing with decreasing dx step. Similarly, the cubic-spline method shows a decreasing error for dx down to a value 10^{-3} with the slope retaining the same value.

Similar results were obtained by finite-element method (Scheme IV, Fig. 20). For $dy = 0.0006$ and double precision, the three-point formula displays a minimum of E_{c_f} at $dx = 0.02$ and increases for $dx < 0.02$, and the magnitude of the error becomes larger than that of the four-point and cubic-spline methods. Using the four-point equation, a reduction of E_{c_f} with decreasing dx all the way down to $dx = 10^{-3}$ is obtained. Also, application of the cubic-spline method reduces the error E_{c_f} with decreasing dx to value 10^{-3}.

The effect of the double precision on the error E_{c_f} as a function of dx and dy for a four-point differentiation formula is shown for both finite-difference and finite-elements methods in Fig. 16. The order of error E_{c_f} is increased for the finite-element method for $dx = 0.01$ and for $dx = 0.005$ ($E_{c_f} \sim dy^{1.91}$) and for the finite-difference method, the order is increased for $dx = 0.005$($E_{c_f} \sim dy^{1.38}$) and it is decreased for $dx = 0.01$.

Both finite-elements and finite-difference schemes required essentially comparable computation time (Fig. 21). This is not surprising, since although two different

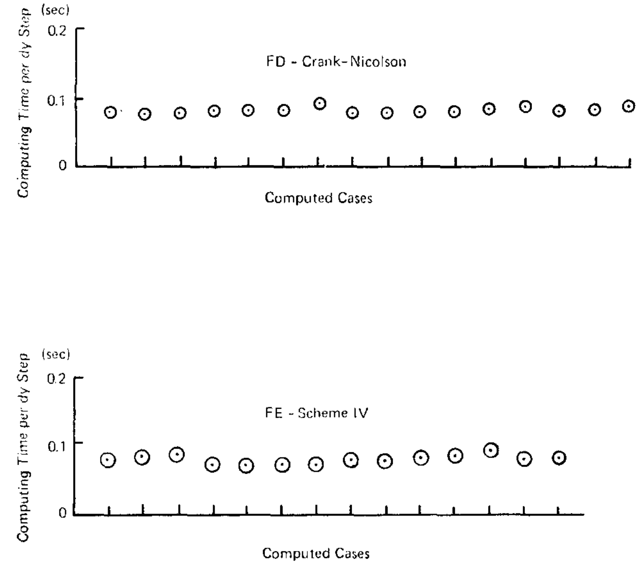


FIG. 21. Computing time for FD method, Crank-Nicolson and FE method, Scheme IV.

methods were used to derive an approximation for the momentum equation, both resulted in difference equations that require essentially the same computation times.

4. Conclusions

An implicit finite-element algorithm for solution of the boundary layer equations is developed. Within the finite-element approach, a variant with a tridiagonal convective term $uu_{,x}$ and several variants with condensed convective term matrix were developed. This condensation formalized by an order of magnitude argument has resulted in several matrix forms that are equivalent to several integration formulas for the integral $\int u\phi_j^h(y)\,dy$ over a subinterval with linear, quadratic, or cubic local approximation for u. A delineation and identification of these formulas was presented.

Introducing a finite-difference approximation for the derivative $u_{,x}$ yields a simple tridiagonal equation form. This implicit integration formulation eliminates more complex, explicit integration and allows recovery of u_n^{i+1} by simple solution of a tridiagonal equation, or by simple substitution, if values from the previous iterate, $u_n'^{i+1}$ are used in the coefficients a_n, b_n, and c_n. Using these algorithms, extensive incompressible boundary layer computations were performed and compared with the Blasius solution and with results obtained by second-order accurate finite-difference methods. All algorithms used in these studies displayed essentially identical stability characteristics. From the studies of accuracy based on the velocity and skin friction error of these schemes, it was concluded that the finite-element methods yield results that are comparable in accuracy and convergence with the results obtained by finite-difference methods. A delineation of the effect of the dx and dy step sizes on velocity and skin friction error was obtained and optimum conditions for numerical computations were established. The computations can be optimized by using the finite-element variants in various ranges of dy and dx step values. The aspect ratios of dx/dy up to about 100 were used, although it is desirable to use lower values for finite error (i.e., 20–50). All methods produce comparable results at $dy = 0.0005$ in the physical normal direction, and there are ranges of optimum conditions for various finite-element variants.

In general, the error is reduced by decreasing the step size in the normal direction dy; and as a result of the interplay of the round-off and truncation error, a minimum error may occur at a certain optimum value of dx, which may, in the case of E_{c_f}, depend on the dy value, method of computing the derivative $(\partial u/\partial y)|_w$ and on the precision.

Although both the finite-difference and finite-element methods are second order or better as far as E_u is concerned, E_{c_f} depends on the method of computing the derivative $(\partial u/\partial y)|_w$ and on the precision of computations, especially in physical coordinates where the step sizes are much smaller than the usual step sizes in the transformed coordinates. The error results in stabilization of the accuracy when the step sizes are reduced below certain values.

The single and double precision affect the error E_{c_f} as postulated due to the effect of the interplay between the round-off and truncation error as established with the test function $f = \sin(y)$. For Blasius solution, the same trend is established for both Crank–Nicolson and finite-element (Scheme IV) methods, although the trend is not as pronounced and is obscured by the fact that the computed curve $u(y)$ [illegible]

In the range of dy of this study, $0.0003 \leqslant dy \leqslant 0.001$ the slope of (df/dy) is of the order of one for the second- and fourth-order methods. These methods are second and fourth order, respectively, for larger dy steps, which are usual in transformed coordinates when $\Delta\eta = 0.02\text{-}0.2$ [8]. Lower slope would be expected for Blasius solution having more irregular $u(y)$ curve.

As a result of these studies, it is concluded that the linear finite-element technique, on the multiple bases of accuracy, convergence, and computation time, is competitive with the best second-order accurate finite-difference methods, and the greater flexibility of discretization of the solution domain makes this technique suitable for many problems with irregular geometry.

5. Postlogue

In the context of this study, dealing with a comparison of the finite-element and finite difference methods in fluid mechanics, a timely conjecture about an interesting dialogue in the field of the elasticity concerning the merits of finite-elements versus finite-difference Methods [9] is in order. Many advantages and disadvantages of both methods are advanced.

Some authors claim that there seems to be a convergence of both methods (or finite-difference analysis can be a subset of finite-element analysis [10]) and state that if one replaces the explicit integration, in finite elements by the implicit integration, more common in finite differences, in a sense that one replaces the first derivative by a finite-difference expression in the right way, a finite-difference formulation is obtained. This has been confirmed in this study by introducing an implicit integration scheme, usual in finite differences, instead of an explicit one, and is considered as a contribution of this study.

It is our belief that a dialogue similar to the one in the structural area will be forthcoming shortly in computational fluid mechanics, and it will bring into better perspective the advantages and disadvantages of each of these methods.

6. APPENDIX: NOMENCLATURE

$[\ ]$ = rectangular matrix.
$\{\ \}$ = column vector.
$\{\ \}^T$ = transposed column vector.
$(\ )_{,x}$ = $(\partial/\partial x)(\ )$ derivative with respect to x.
c_f = coefficient of friction.
i, n = indices.
l = length of an element.
u, v = velocity.
$Q_m{}^*$ = approximation for dependent variable Q.
Re = Reynolds number.
x, ξ = coordinate in predominant flow direction.
y = normal coordinate.
ρ = density.
μ = dynamic viscosity.
ν = kinematic viscosity.
ϕ = trial function.
θ = weighting factor.

REFERENCES

1. R. J. ROACHE, "Computational Fluid Dynamics," Hermosa, Albuquerque, New Mexico.
2. A. J. BAKER, Nonlinear initial boundary value solutions by the finite elements method, *in* "Proc., Int. Conf. Comput. Methods Nonlinear Mechanics," University of Texas at Austin, 1974.
3. B. A. FINLAYSON AND L. E. SCRIVEN, *Appl. Mech.* **19** (1966), 735–748.
4. O. C. ZIENKIEWICZ, "The Finite Element Method in Engineering Science," McGraw–Hill, New York, 1971.
5. G. STRANG AND G. J. FIX, "An Analysis of the Finite Element Method," Prentice–Hall, Englewood Cliffs, N.J., 1973.
6. E. KRAUSE, *AIAA J.* **5** (1967), 1231–1237.
7. H. SCHLICHTING, "Boundary Layer Theory," McGraw–Hill, New York, 1960.
8. J. W. CHRISTIAN *et al.* Similar solutions of the attached and separated compressible laminar boundary layer with heat transfer and pressure gradient, ARL70-0023, February 1970.
9. R. F. HARTUNG, Ed., Finite elements versus finite differences, panel discussion, *in* "Proc. Conf. Computer Oriented Analysis of Shell Structures, Palo Alto, California, 1971, AFFDL-TR-71-79, June 1971.
10. R. J. MELOSH, The optimum approach to analysis of elastic continua, *in* Proc., Conf. Computer Oriented Analysis of Shell Structures," (R. F. Hartung, Ed.), Palo Alto, California, 1971.