JOURNAL OF COMPUTATIONAL PHYSICS 32, 289–324 (1979)

REVIEW ARTICLE

Utility of a Finite Element Solution Algorithm for Initial-Value Problems

A. J. BAKER* AND M. O. SOLIMAN†

University of Tennessee at Knoxville, Department of Engineering Science and Mechanics,
Knoxville, Tennessee 37916

Received January 11, 1978; revised November 28, 1978

The Galerkin criterion within a finite element Weighted Residuals formulation is employed to establish an implicit solution algorithm for an initial-value partial differential equation. Numerical solutions of a transient parabolic and a hyperbolic equation, obtained using linear, quadratic and two cubic finite element basis functions, are employed to quantize accuracy and confirm and refine theoretical convergence rate estimates. The linear basis algorithm for the hyperbolic equation displays excellent accuracy on a coarse computational grid and a high-order convergence rate with discretization refinement. Good accuracy and a strong convergence rate in surface flux are determined for a nonhomogeneous Neumann boundary constraint applied to a parabolic equation. The results amply demonstrate the impact of the non-diagonal finite element initial-value matrix structure on solution accuracy and/or convergence rate.

INTRODUCTION

The role that finite element theory may play in computational fluid mechanics requires a critical evaluation. The engineers working in aircraft structural analysis in the late 1950's are generally credited with conception of a finite element as a discretization and a solution procedure (Ref. [1]). With the concurrent emergence of practical digital computer systems, the use of the method mushroomed for static analysis of heretofor intractible structural systems. As a historical repetition of the legitimization of Heaviside's operator calculus by the Laplace transform, the variational calculus and strain energy minimization soon emerged as the mathematical support for this engineering analysis. The theory developed very rapidly, computer program practice boomed, and finite element solution methodology became synonymous with linearized analysis of impressively complicated structural systems.

To the outsider, perhaps the dominant visible feature of this developed finite element solution methodology was its geometrical flexibility, as exemplified by a mesh of triangles subdividing a domain bounded by an arbitrary curve. Naturally, great interest was generated in assessing the viability of the method for other fields in mechanics. Extension to steady-state heat conduction and potential flow was obvious (cf. Ref. [2]), since the energy functional equivalence of the Laplacian is well known. A linear transient heat conduction analysis was formulated within the variational

Currently on leave as Visiting Scientist, AFFDL, WPAFB, Ohio.

0021-9991/79/090289–36$02.00/0

framework using the convolution (Ref. [3]). The desire to extend the methodology to a more general non-linear problem description, fluid mechanics for example, initiated the search for variational principles (Ref. [4]). This was of minimal duration, however, upon recognition of the Method of Weighted Residuals (Ref. [5]) and rediscovery of the Galerkin criterion (Ref. [6]) as a theoretical basis, alternative to the variational calculus. This facilitated direct formulation of a finite element algorithm for arbitrary non-linearity, including for example a complete Navier–Stokes algorithm and prediction of a recirculating flow (cf. Ref. [7]). Thereafter, the early 1970's experienced a great explosion of development of finite element algorithms applied to a multitude of non-linear problem classes in fluid dynamics, including for example turbulent reacting ducted flows (Ref. [8]), transonic aerodynamics (Refs. [9, 10]), and environmental hydrodynamics (Ref. [11]).

As opposed to the obvious geometric flexibility, which has become of much less significance with the emergence of regularizing coordinate transformations (cf. Ref. [12]), the finite element algorithm provides a thoroughly structured procedure for transformation of a given partial differential equation into an equivalent larger order equation system written on the selected discrete variable distribution. The tools available for use in this procedure include calculus and vector field theory, and the Neumann boundary condition statement if present is automatically included. Should the parent partial differential equation exhibit initial value character, an analytical separation of variables yields the discretized form as a system of ordinary differential equations, eligible for integration using any appropriate procedure. The significant contrast with finite differences therefore, may be in the elimination of the requirement to derive a consistent differenced equation system (cf. Ref. [13]).

That this is indeed a benefit requires affirmation, and this paper addresses the issue with specific emphasis on quantization of efficiency, accuracy, and convergence with discretization refinement as a function of appropriate influencing parameters, in particular higher degree interpolation polynomials. A linear problem class is addressed, and accuracy and convergence are measured in the intrinsic energy or L^2 norm. The theoretical foundation for elliptic equations is well developed (cf. Refs. [14, 15]); a similar basis exists for first-order and second-order hyperbolics (Refs. [16, 17], as well as a parabolic initial-value description (Ref. [18]). The primary requirement is to quantize performance for initial-value problem descriptions, since finite element theory introduces a feature distinctly different from finite differences by yielding an ordinary differential equation system coupled in the derivatives (superscript prime) of the form

$$[C]\{q\}' = \{f\} \tag{1}$$

Using finite difference theory, the C matrix in equation (1) would be defined as the identity matrix. Finite element theory predicts that $[C]$ be non-diagonal with bandwidth a function of discretization and the degree of the finite element interpolation polynomial. The finite element C matrix can be "condensed" to a diagonal form to reproduce the finite difference equivalent; this has been common practice to render more straightforward the application of an explicit integration algorithm.

The intent of this study is to firmly quantize solution accuracy level, and to confirm or refine theoretically predicted convergence rates with discretization refinement, primarily as a function of the initial-value matrix structure and interpolation polynomial degree. Since the long-term goal is applications in fluid mechanics, the study problem is the advection-diffusion equation. The finite element solution algorithm is derived using a hypermatrix formalism that significantly reduces computer storage requirements and facilitates inclusion of parameter variations in a convenient manner. A brief review is given of the theoretical basis for estimation of convergence properties of the algorithm. A tightly controlled sequence of computational experiments was conducted for a parabolic and a hyperbolic partial differential equation, and numerical solutions generated using linear, quadratic and two cubic interpolation polynomials within the finite element algorithm. An implicit finite difference procedure is employed to solve the resultant ordinary differential equation system. Solution accuracy is quantized and convergence rates with discretization refinement assessed as a function of Dirichlet, Neumann, and radiation boundary conditions as well as initial-value matrix structure. In all cases, Richardson extrapolation is employed to determine integration step-sizes for which truncation error is an inconsequential contribution to the measured solution error. Finite element results are compared appropriately to finite difference solutions. The summary concludes that finite element theory does display desirable features when employed to derive numerical solution algorithms for linear initial-value problem descriptions.

Finite Element Solution Algorithm

Consider the scalar field $q(x_i, t)$, the transient evolution of which is required determined on n-dimensional space R^n spanned by the x_i coordinate system, $1 \leqslant i \leqslant n$. The domain of the solution is $\Omega \equiv R^n \times t \in x_i \times [t_0, t)$ with closure $\partial\Omega \equiv R^{n-1} \times t$. On Ω, q is the solution to the advection-diffusion equation

$$L(q) = \frac{\partial q}{\partial t} + \frac{\partial}{\partial x_j}(qu_j) - \frac{\partial}{\partial x_j}\left[k\frac{\partial q}{\partial x_j}\right] + f = 0 \tag{2}$$

where u_j is the n-dimensional velocity vector, k is the effective diffusion coefficient, and f is a source/sink term. The boundary condition constraints applicable on $\partial\Omega$ are contained within

$$l(q) = a_1 q + k\frac{\partial q}{\partial x_j}\hat{n}_j + a_3 = 0 \tag{3}$$

where $\hat{n}$ is the local outward-pointing unit normal vector, and the a_i are specified coefficients that may depend upon time. An initial condition on $\Omega_0 \equiv R^n \times t_0$ is required as

$$q(x_j, t_0) \equiv q_0(x_j) \tag{4}$$

The fundamental concept of a finite element algorithm is the assumption of known functional dependence for $q(x_j, t)$ on disjoint, contiguous subdomains of Ω. These "computational control volumes" are termed finite elements (of Ω) with domain $\Omega_e \equiv R_e^n \times t$. The common procedure is to assume q adequately interpolated on Ω_e by a truncated power series of the form

$$q_e(x_j, t) \equiv \{\phi_k(x_j)\}^T\{Q(t)\}_e \tag{5}$$

In equation (5), the elements of the row matrix $\{\phi_k(x_j)\}^T$ are linearly independent, kth degree polynomial functions of x_i, so constructed that they form a cardinal basis (cf. Ref. [14, p. 48]). The $\{\phi_k\}$ possess compact support, which greatly simplifies the integral operations to follow, and their construction on elementary finite element shapes is quite straightforward (cf. Ref. [19, Chaps. 7–8]). As observed in Equation (5), separation of variables has been assumed valid, and the matrix elements of $\{Q(t)\}_e$ are the (unknown) series expansion coefficients. These correspond to the temporal evolution of q_e at the node points of the finite element discretization of R^n.

The Galerkin–Weighted Residuals formulation provides the basis for developing an algorithm, for determination of the expansion coefficients, that degenerates in the special case of a linear elliptic differential equation to extremization of an energy functional. Quite directly, the approximation function equation (5) is introduced into the differential statements (2)–(3). Since equation (5) is not the solution, the error that is induced in this operation is rendred orthogonal to each member of the approximation function set $\{\phi_k(x_i)\}$ in the classical manner. Since the $\{\phi_k(x_i)\}$ enjoy compact support, the integrals are non-vanishing only on Ω_e, which greatly simplifies the algebra. Identifying the mapping function S, more familiarly termed the assembly operator, which carries the integral operation on Ω_e onto Ω, the orthogonality requirement is directly expressed as

$$\underset{e}{S}\left[\int_{\Omega_e} \{\phi_k\}\, L(q_e)\, d\tau + \lambda \int_{\partial\Omega_e \cap \partial\Omega \neq 0} \{\phi_k\}\, l(q_e)\, d\sigma\right] = \{0\} \tag{6}$$

The order of equation (6) is identical to the total number of node points at which the transient solution, i.e. $\sum_e \{Q(t)\}_e$, is to be determined, and λ is a convenient scalar multiplier. Independent of the dimension n of R^n (the space spanned by the x_i coordinate system), for equations (2)–(3), equation (6) yields an ordinary differential equation system for solution of $\sum_e \{Q(t)\}_e \equiv \{Q\}$, of the form

$$\underset{e}{S}\,[[C]_e\{Q\}_e' + [U]_e\{Q\}_e + [K]_e\{Q\}_e + \{y\}_e] = \{0\} \tag{7}$$

The superscript prime denotes (ordinary) differentiation with respect to time. For equations (2)–(3), the specific forms for the various matrices defined in equation (7) are

$$[C]_e \equiv \int_{R_e{}^n} \{\phi_k\}\{\phi_k\}^T \, d\tau \tag{8}$$

$$[U]_e \equiv \int_{R_e{}^n} \{\phi_k\} \frac{\partial}{\partial x_i} [\{U_i\}_e^T \{\phi_k\}\{\phi_k\}^T] \, d\tau \tag{9}$$

$$[K]_e \equiv \int_{R_e{}^n} \frac{\partial}{\partial x_j} \{\phi_k\} \, k_e \frac{\partial}{\partial x_j} \{\phi_k\}^T \, d\tau + \int_{\partial R_e{}^n \cap \partial R \neq 0} \{\phi_k\} \, a_1 \{\phi_k\}^T \, d\sigma \tag{10}$$

$$\{y\}_e \equiv \int_{R_e{}^n} \{\phi_k\} \, f_e \, d\tau + \int_{\partial R_e{}^n \cap \partial R \neq 0} \{\phi_k\} \, a_3 \, d\sigma \tag{11}$$

and λ has been set to cancel the boundary contribution, generated from use of a Green–Gauss theorem on the first term in equation (2), with the corresponding term in equation (3), as they appear in equation (6). The domains of integration in equations (8)–(11) are only $R_e{}^n$, and by definition $\partial R_e{}^n \equiv R_e^{n-1}$; hence, all integrals can be evaluated once and for all. The n-dimensional velocity field u_i, as well as k and f, are assumed interpolated on Ω_e using equation (5). Index summation is implied on all repeated tensor subscripts, and the order and specific forms taken are strictly dependent upon n and k, the finite element polynomial degree, equation (5). The integrals can be performed analytically for $n = 1$ for all k, $1 \leqslant k \leqslant 3$; they can be performed numerically to any required accuracy for all n and k. Appendix A presents a summary of establishment of appropriate cardinal bases and evaluation of the element matrices (8)–(11) for select k and n.

Integration Algorithm

Equation (7) is solvable using any (finite difference) numerical integration algorithm. Specifically, it may be explicit, implicit, single-step, multi-step and/or predictor-corrector. The salient features may be illustrated directly by employing the familiar single-step implicit Euler algorithm

$$\{Q\}_{j+1} = \{Q\}_j + h[\theta\{Q\}'_{j+1} + (1 - \theta)\{Q\}'_j] \tag{12}$$

In equation (12), j is the time-step index, h is the integration step-size and θ is a parameter $0 \leqslant \theta \leqslant 1$ controlling implicitness. For $\theta = 0$, equation (12) is explicit Euler, $\theta = 1$ is fully implicit or backwards Euler, and $\theta = 1/2$ yields the familiar trapezoidal rule. For θ non-zero, a modified Newton iteration algorithm can be established, that is equally applicable to non-linear problem specifications, in the form

$$[J(\{F\}_{j-l})]\{\delta Q\}_{j+1}^{p+1} = -\{F(\{Q\}_{j+1}^{p})\} \tag{13}$$

The dependent variable in equation (13) is the iteration vector, defined as

$$\{Q\}_{j+1}^{p+1} \equiv \{Q\}_{j+1}^{p} + \{\delta Q\}_{j+1}^{p+1} \tag{14}$$

where p is the iteration index, and $l \geqslant 0$ is an integer that retards evaluation of the Jacobian as an economy measure, should it indeed be a function of the dependent variable. The right side of equation (13) is equation (7) evaluated for the pth iterate after substitution of equation (12) to replace the ordinary derivative matrix. By direct substitution,

$$\{F\}_{j+1}^{p} = \underset{e}{\overset{S}{}}[[C]_e (\{Q\}_{j+1}^{p} - \{Q\}_j) + h(\theta\{g_e\}_{j+1}^{p} + (1 - \theta)\{g_e\}_j)] \tag{15}$$

where

$$\{g_e\}_l^p = ([U]_e + [K]_e)\{Q\}_l^p + \{y\}_e \tag{16}$$

Note that equations (15)–(16) are defined only as inner products on elements, with the assembly operator yielding the equivalent global expression. The vanishing of $\{F\}$, to within a specified definition of computed zero, yields equation (13) homogeneous, hence convergence of the iteration process for any evaluation of the Jacobian. The initial estimate $\{Q\}_{j+1}^{1}$ for any iteration can be determined using $\theta = 0$.

Equations (13)–(15) could be directly rearranged to yield a non-iterative algorithm for a linear problem, but formation of the required Jacobian is unaffected. By definition, the Jacobian is the partial derivative of equation (15) with respect to $\{Q\}$. Hence,

$$[J] = \underset{e}{\overset{S}{}}\left[[C]_e + h\theta([U]_e + [K]_e) + \left(\frac{\partial[C]_e}{\partial\{Q\}_e} + h\theta\,\frac{\partial([U]_e + [K]_e)}{\partial\{Q\}_e}\right)\{Q\}_{j-l}\right] \tag{17}$$

where the final term accounts in general for a contribution stemming from implied non-linearity, for example in k_e for the present case. Equation (17) illustrates the analytical expression obtainable using the presented formalism, as well as the role of retarded evaluation at time step $j - l$ if required, see equation (13). All operations are again limited to matrix inner products on an elemental basis, hence implicitly independent of the dimension n of R^n. The rank of $[J]$ is obviously equal to the order of $\{\delta Q\}$, and specific (Dirichlet) boundary constraints which reduce the order below the total number of nodes in the discretization can be applied at any convenient point in the solution process, including within the evaluation of $\{F\}$.

Error and Convergence Measurement

The express purpose of this analysis is to measure solution error for the contribution of finite element theory applied to an initial-value problem, and to establish and/or confirm expected convergence rates with discretization refinement and other appropriate parameters. Since equation (2) is a conservation law statement, appropriate accuracy measures include the integral over Ω of q and $\sqrt{q^2}$. Hence, using equation (5), define

$$p_1(t) \equiv \int_{R^n} q \, d\tau = \sum_e \int_{R_e{}^n} \{\phi_k\}^T \, d\tau \, \{Q(t)\}_e \tag{18}$$

$$p_2(t) \equiv \frac{1}{2} \left[\int_{R^n} q^2 \, d\tau \right] = \frac{1}{2} \sum_e \left[\{Q(t)\}_e^T \int_{R_e{}^n} \{\phi_k\}\{\phi_k\}^T \, d\tau \, \{Q(t)\}_e \right] \tag{19}$$

using specific recognition of the independence of $\{Q(t)\}_e$ from the integrations.

Convergence properties of a finite element algorithm are traditionally measured in an "energy" norm (cf. Ref. [15, Chap. 2]). The solution energy corresponds to the sum of quadratic terms in the functional whose extremization yields the corresponding differential equation statement for a linear boundary value problem, i.e.,

$$E(p, q) \equiv \frac{1}{2} \int_{R^n} (\alpha \, \nabla q \cdot \nabla p + \beta \, qp) \, d\tau \tag{20}$$

where α and β are constants of the motion. Generalizing the concept to include energy at the boundary, the finite element energy functional appropriate for equations (2)–(3), using equation (5), is

$$\begin{aligned} E(q_e , q_e) &= \frac{1}{2} \sum_e \left[\int_{R_e{}^n} k \frac{\partial q_e}{\partial x_i} \frac{\partial q_e}{\partial x_i} \, d\tau + \int_{\partial R_e{}^n} a_1 q_e{}^2 \, d\sigma \right] \\ &= \frac{1}{2} \sum_e |\{Q(t)\}_e^T \, [K]_e \, \{Q(t)\}_e| \end{aligned} \tag{21}$$

The energy in equation (21) is a function of time, and $[K]_e$ is defined in equation (10). The square root of equation (21) is the H^1 Sobolev norm.

Convergence properties of a finite element algorithm are expressed in terms of the energy norm, equation (20), or equivalently, the H^1 norm. Briefly (cf. Ref. [15, Chap. 2 and 5]), for a linear boundary value problem, the finite element-generated approximation to system energy is extremum in comparison to all other numerical evaluations including finite differences. Furthermore, since the error is orthogonal to the solution surface, as enforced using equation (6), the energy in the error is equal to the error in the energy. Finally, the error in the energy $E(q - q_e , q - q_e)$ is bounded from above by a constant, independent of discretization, by the measure (length, area, etc.) of the largest element in the discretization raised to an exponent, and a norm related to the maximum (required) derivative of the solution. Hence, generalizing to a transient description, convergence with discretization refinement should occur in the error energy norm as

$$E(q - q_e , q - q_e) \leqslant C\Delta_e^{2(k+1-m)} \, \| q \|_\alpha^2 \tag{22}$$

In equation (22), k is the lowest complete degree of the finite element interpolation polynomial $\{\phi_k(x_i)\}$, equation (5), $2m$ is the highest order derivative in equation (2),

Δ_e is the measure of the largest finite element on R^n, $\alpha = k + 1$ is related to the required smoothness of the exact solution, and

$$\| q \|_\alpha^2 \equiv \sum_{i=1}^{\alpha} \int_{R^n} \left(\frac{d^i q}{dx^i} \right)^2 d\tau \tag{23}$$

For C a constant independent of Δ_e, and equation (23) set by the problem statement, numerically measured convergence in energy, as obtained by measuring equation (21), can confirm adherence to the exponent $2(k + 1 - m)$ in equation (22). For the second order partial differential equation (2), $m = 1$, and the exponent should approximate 2, 4 and 6 for linear, quadratic, and standard cubic functions, $k = 1, 2, 3$ respectively in equation (5).

For the parabolic form of equation (2), particular interest lies in evaluating the accuracy with which the dependent variable is determined on $\partial\Omega_e$ where at a gradient boundary condition is applied. Two max norms of specific use are, therefore,

$$e_1 \equiv \max | q_e - q | \tag{24}$$

$$e_2 \equiv \max \left| \frac{dq_e}{dx_n} - \frac{dq}{dx_n} \right| \tag{25}$$

where x_n corresponds to the coordinate normal to the closure segment $\partial\Omega_e$.

The focus of this study excludes consideration of integration algorithms alternative to equation (12), and $\theta = 1/2$ was used exclusively. All computations were performed in double precision. To assure that truncation error was not clouding measured accuracy, Richardson extrapolation (cf. Ref. [20]) was employed to generate a higher-order accurate solution at appropriate time points. The cited error norms were evaluated using both finite element solutions, and the integration step-size was adjusted to cause truncation error to be at least two digits removed from convergence-measuring significance in the error norm.

Results

Numerical determination of solution accuracy and convergence rates for the finite element algorithm, equation (6), are required for dominantly hyperbolic and parabolic forms of equations (2)–(3). On two-dimensional space, the elementary finite element cardinal basis functions, equation (5), are linear polynomials on a triangular, and bi-linear functions on a quadrilateral element geometry. Using the natural coordinate system shown in Figure 1, the latter is

$$\{\phi_{1+}(x_1, x_2)\} = \frac{1}{4} \begin{Bmatrix} (1 - \eta_1)(1 - \eta_2) \\ (1 + \eta_1)(1 - \eta_2) \\ (1 + \eta_1)(1 + \eta_2) \\ (1 - \eta_1)(1 + \eta_2) \end{Bmatrix} \tag{26}$$

Alteration of the node numbering permutes the locations of the elements of $\{\phi_{1+}\}$, the η_i coordinate system is normalized, and the element shape illustrated also results from an isoparametric transformation of a general curved-sided quadrilateral element (cf., Ref. [19], Chap. 8).

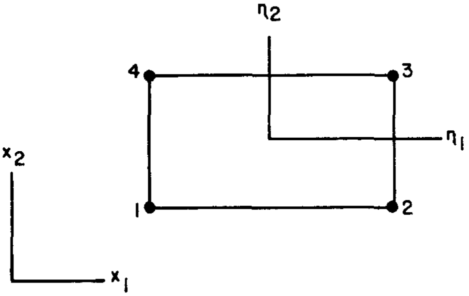

FIG. 1. Natural coordinate description for quadrilateral finite element.

Inserting equation (26) into equation (8), performing the integrals and defining a "standard" matrix named [B200], yields the initial-value element matrix

$$[C]_e \equiv \Delta_e[B200] = \Delta_e \frac{1}{36}\begin{bmatrix} 4 & 2 & 1 & 2 \\ & 4 & 2 & 1 \\ & & 4 & 2 \\ (\mathrm{sym}) & & & 4 \end{bmatrix} \tag{27}$$

In equation (27), Δ_e is the measure (area) of the element, and [B200] is an integer array in a DATA statement. Hence, the words of storage required is equal to the number of elements Ω_e on R^2 plus eleven. In the definition of [B200], the B signifies a matrix on two-dimensional space, 2 states that two $\{\phi_k\}$ are defined in the matrix, and the two 0's are Boolean counters indicating that neither of the $\{\phi_k\}$ are differentiated. Correspondingly, A and C prefixes denote matrices on one- and three-dimensional spaces, and non-zero counters indicate appropriate differentiation.

The impressed velocity field is assumed divergence free. Hence, in equation (9), after using the chain rule and noting that $\{U_i\}_e^T\{\phi_k\}$ is a scalar, the two-dimensional convection matrix becomes

$$[U]_e = \{U_i\}_e^T \int_{R_e^2} \{\phi_{1+}\}\{\phi_{1+}\} \frac{\partial}{\partial x_i} \{\phi_{1+}\}^T \, dx_1 \, dx_2 \tag{28}$$

As discussed in Appendix A, the integral in equation (28) defines a square hypermatrix of global order four with elements that are themselves column matrices of order four. Employing the matrix naming convention, the bilinear function equivalent for convection is

$$[U]_e = \Delta_e\{U\}_e^T\,[B3001] + \Delta_e\{V\}_e^T\,[B3002] \tag{29}$$

The defined matrices are given in Appendix B, and $\{U\}_e$ and $\{V\}_e$ are nodal arrays of the x_j scalar components of u_j on Ω_e. Storage required for the convection matrices is 128 words, which could be reduced to 32 by an efficient array permutation algorithm.

By similar operations, and assuming the diffusion coefficient k_e variable, equation (10) is

$$[K]_e = \Delta_e\{K\}_e^T\,[B3011] + \Delta_e a_1[A200] \tag{30}$$

The $[B3011]$ is symmetric and requires 40 words of storage, and $[A200]$ results for the boundary condition constraint since $\partial R^2 = R^1$, i.e., a one-dimensional element. These matrices are also listed in Appendix A and Appendix B, and the measure Δ_e for $[A200]$ corresponds to the length of the one-dimensional closure segment. The final required matrix is equation (11),

$$\{y\}_e = \Delta_e[B200]\{F\}_e + \Delta_e a_3\{A10\} \tag{31}$$

where f_e is assumed variable with nodal values $\{F\}_e$, $[B200]$ was identified in equation (27), and $\{A10\}$ is the corresponding one-dimensional boundary condition matrix, see Appendix A. No additional storage is required. The developed forms are appropriate for uniform discretization of Ω. Appendix B also lists the corresponding definitions for non-square elements. Only the storage requirements for Δ_e and $[B3011]$ are doubled, and the diffusion coefficient can exhibit directional character with no additional penalty. The hypermatrix formalism is extendible to use with curved-sided, isoparametric finite elements as well; however, this lies outside the scope of the present development.

Primary focus is on accuracy and convergence of the finite element algorithm for the limiting cases of equations (2)–(3). The hyperbolic limiting case is achieved by setting $k \equiv 0$; assuming only vanishing gradient boundary conditions are employed, and that f vanishes identically, equation (7) becomes, for a uniform grid,

$$\underset{e}{S}\,[[B200]\{Q\}_e' + (\{U\}_e^T\,[B3001] + \{V\}_e^T\,[B3002])\{Q\}_e] = \{0\} \tag{32}$$

Note the analytical similarity between equations (32) and (2). With a little experience, equations of the form (32) can be established directly from the differential equation statement. Similarly, for the matrix iterative solution, the Jacobian, equation (17) is

$$[J] = \underset{e}{S}\,[[B200] + h\theta(\{U\}_e^T\,[B3001] + \{V\}_e^T\,[B3002])] \tag{33}$$

The test case is advection of a concentration packet across the diagonal of a square domain by a constant velocity field. The initial distribution was developed using

$$q(\Delta_e\,, 0) = 1 + \cos\frac{\pi}{4}\left(3 + \frac{n\Delta_e}{2M}\right) \qquad n = 1, 2,\ldots, M \tag{34}$$

where M is an integer which spreads the distribution over $2n\,\Delta_e$, where Δ_e is the measure of the uniform grid. The distribution is then rotated about the maximum to produce the two-dimensional initial condition $q(x_i, 0)$, equation (4). The least M for which the distribution can approximate an inflection point is 5. Figure 2a displays the resultant initial distribution on a 32×32 finite element grid, and the numerically computed distributions after 50, 100 and 150 time steps for a Courant number equal to 0.1, where

$$C \equiv \frac{U\,\Delta t}{\Delta_e} \tag{35}$$

The peak value is retained to within 1% indicating the algorithm is essentially free of numerical diffusion. The trailing wakes, which are indicative of dispersion error, have maximum amplitude $\pm 2\%$ of the peak value and nominal preserved wavelenth $2\,\Delta_e$. For comparison, Fig. 2b shows the results obtained using the finite difference, Crank–Nicolson equivalent for equation (32). As mentioned in the Introduction, and as will become immediately apparent, this is achieved by rendering $[C]_e$ the diagonal matrix $\lceil C \rfloor_e$. From equation (27),

$$\lceil C \rfloor_e \equiv \frac{\Delta_e}{4}\begin{bmatrix} 1 & 0 & 0 & 0 \\ & 1 & 0 & 0 \\ & & 1 & 0 \\ (\text{sym}) & & & 1 \end{bmatrix} \tag{36}$$

as obtained either by row-wise summation or normalization of the diagonal entries in $[C]_e$. This operation literally destroys the fidelity of the previous solution for this discretization, concerning both numerical diffusion and dispersion. Of course, discretization refinement or decreasing the Courant number will improve the accuracy of all solutions in a predictable manner, as will be documented. Finally, for comparison, Fig. 2c presents the results obtained using the finite element form, equation (32), for a factor of two increase in Courant Number to 0.2. These results, produced in half the computational time, are superior to those produced using equation (36).

These solutions display exact symmetry about the plane containing u_i and bisecting the peak of the distribution. Hence, determination of solution accuracy and convergence with respect to discretization refinement, Courant number, degree k of the finite element approximation equation (5), and initial-value matrix $[C]_e$ structure is facilitated by solution of the one-dimensional equivalent. In the present context, and in comparison to equation (32), the algorithmic statement is

$$\underset{e}{S}\,[[A200]\{Q\}_e' + \{U\}_e^T\,[A3001]\{Q\}_e] = \{0\} \tag{37}$$

The defined A-prefix matrices are given in Appendix A. For the case of a linear finite element function, $k = 1$ in equation (5), writing equation (37) at the element

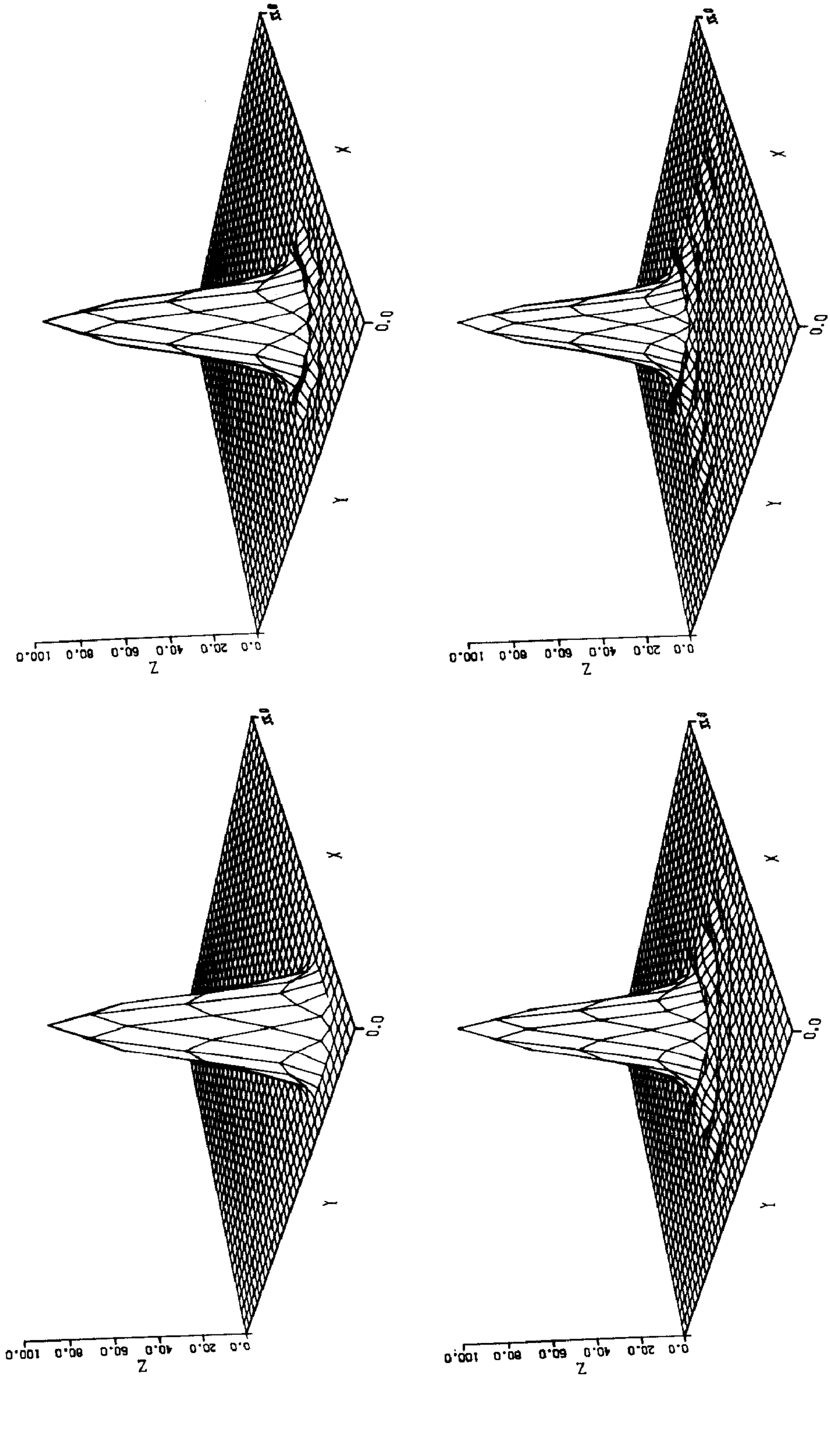

FIG. 2a. Advection of concentration cone. $C = 0.1$, $n\Delta t = 0, 50, 100, 150$.

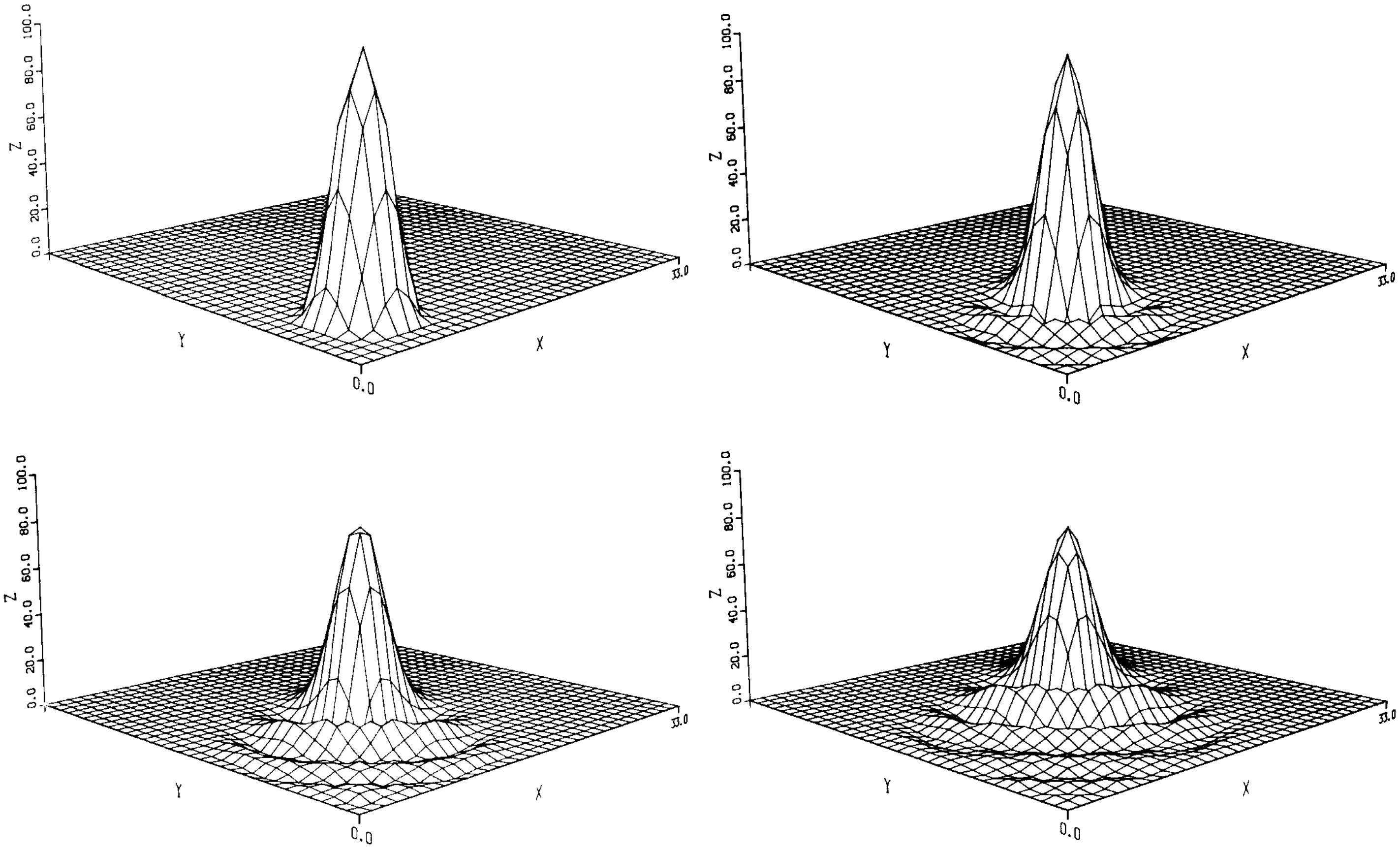

FIG. 2b. Advection of concentration cone. $C = 0.1$, diagonal transient matrix, $n\Delta t = 0$, 50,100, 150.

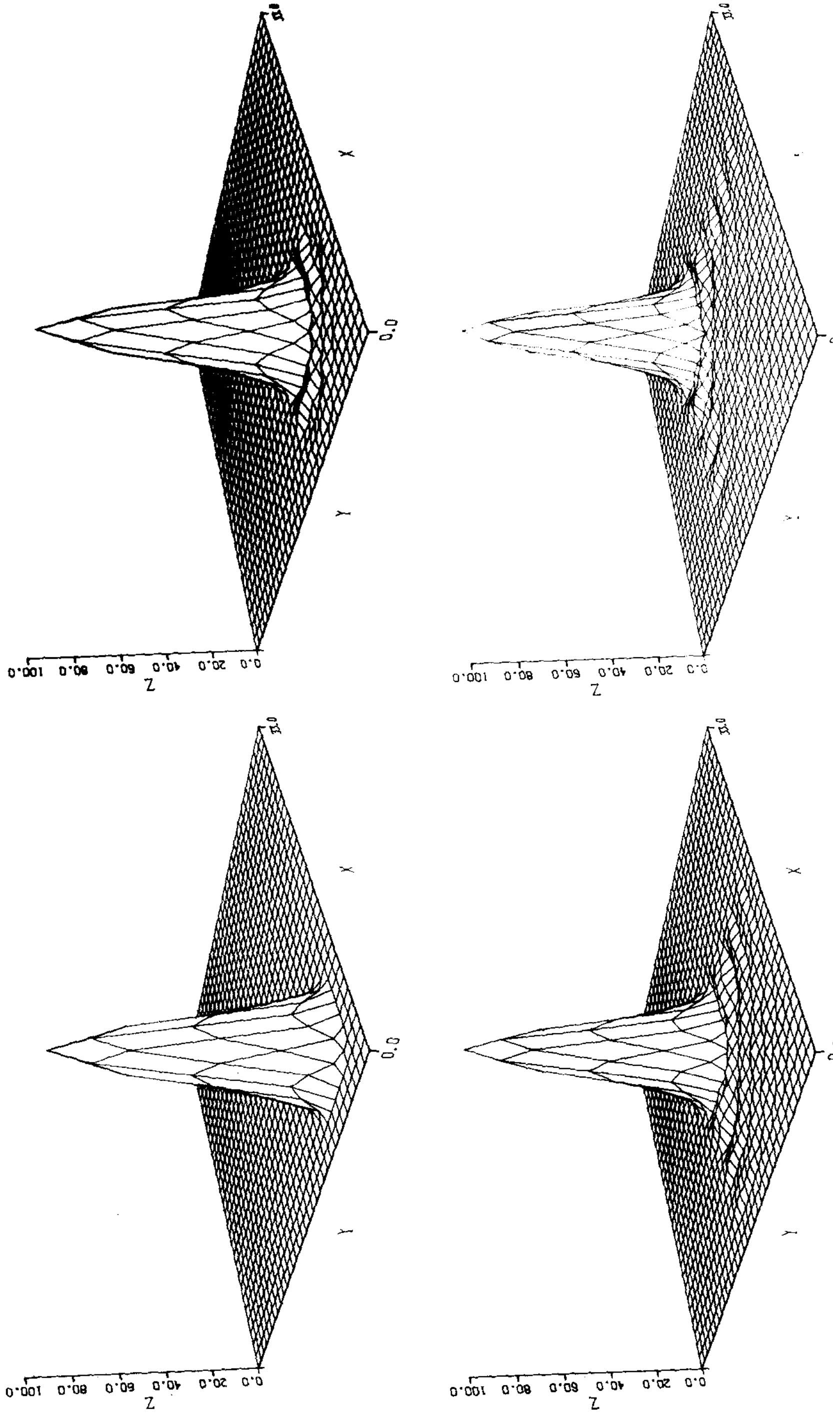

FIG. 2c. Advection of concentration cone. $C = 0.2$, $n\Delta t = 0, 50, 100, 150$.

level, for the element spanning the space between the $J -$ 1st and the Jth nodes,

$$\frac{\Delta_e}{6}\begin{bmatrix}2 & 1\\1 & 2\end{bmatrix}\begin{Bmatrix}Q'_{j-1}\\Q'_j\end{Bmatrix} + \begin{Bmatrix}U_{j-1}\\U_j\end{Bmatrix}^T \frac{1}{6\Delta_e}\begin{bmatrix}\begin{Bmatrix}-2\\-1\end{Bmatrix} & \begin{Bmatrix}2\\1\end{Bmatrix}\\ \begin{Bmatrix}-1\\-2\end{Bmatrix} & \begin{Bmatrix}1\\2\end{Bmatrix}\end{bmatrix}\begin{Bmatrix}Q_{j-1}\\Q_j\end{Bmatrix} = \{0\} \tag{38a}$$

Similarly, for the adjacent element connecting the Jth node to the $J +$ 1st node,

$$\frac{\Delta_e}{6}\begin{bmatrix}2 & 1\\1 & 2\end{bmatrix}\begin{Bmatrix}Q'_{j}\\Q'_{j+1}\end{Bmatrix} + \begin{Bmatrix}U_{j}\\U_{j+1}\end{Bmatrix}^T \frac{1}{6\Delta_e}\begin{bmatrix}\begin{Bmatrix}-2\\-1\end{Bmatrix} & \begin{Bmatrix}2\\1\end{Bmatrix}\\ \begin{Bmatrix}-1\\-2\end{Bmatrix} & \begin{Bmatrix}1\\2\end{Bmatrix}\end{bmatrix}\begin{Bmatrix}Q_{j}\\Q_{j+1}\end{Bmatrix} = \{0\} \tag{38b}$$

Assuming a uniform discretization, the assembled matrix equation (37), written at the jth node, is

$$\frac{\Delta_e}{6}\begin{bmatrix}\ddots & & & \\ & 2, & 1 & \\ & 1, & 4, & 1 \\ & & 1, & 2 \\ & & & \ddots\end{bmatrix}\begin{Bmatrix}\vdots\\Q'_{j-1}\\Q'_j\\Q'_{j+1}\\\vdots\end{Bmatrix}$$

$$+\frac{1}{6\Delta_e}\begin{bmatrix}\ddots & & & \\ & -2U_{j-1} - U_j, & 2U_{j-1} + U_j & \\ & -U_{j-1} - 2U_j, & 2U_j + U_{j+1}U_{j-1} - U_{j+1}, & \\ & & -U_j - 2U_{j+1}, & U_j + 2U_{j+1} \\ & & & \ddots\end{bmatrix}\begin{Bmatrix}\vdots\\Q_{j-1}\\Q_j\\Q_{j+1}\\\vdots\end{Bmatrix} = \{0\} \tag{39}$$

Equation (39) is then directly re-expressed in the (finite difference) recursion form, assuming a uniform velocity,

$$\frac{1}{6}\frac{d}{dt}\left[Q_{j-1} + 4Q_j + Q_{j+1}\right] + \frac{u}{2\Delta x}\left[Q_{j+1} - Q_{j-1}\right] = 0 \tag{40}$$

Performing a Taylor series expansion of equation (40) about $x = x_j$ yields

$$L(Q) = \frac{\partial Q}{\partial t} + U\frac{\partial Q}{\partial x} + \left(\frac{1}{120} - \frac{1}{72}\right) U\,\Delta x^4 \frac{\partial^5 Q}{\partial x^5} + \cdots = 0 \tag{41}$$

Hence, equation (37) is a spatially fourth-order accurate representation of equation (2), achieved using elementary linear basis functions and the equivalent of central differencing for the advection term. This accounts for the absence of numerical diffusion indicated for the two-dimensional solutions, Fig. 2a. Inserting the implicit integration algorithm, equation (12), into equation (40) and denoting $Q_j{}^n \equiv Q(x_j, n\,\Delta t)$ yields

$$
\begin{aligned}
L(Q_j{}^n) = \frac{1}{6}\,[(Q_{j-1}^{n+1} - Q_{j-1}^{n}) + 4(Q_j^{n+1} - Q_j{}^n) + (Q_{j+1}^{n+1} - Q_{j+1}^{n})] \\
+ \frac{C}{2}\,[\theta(Q_{j+1}^{n+1} - Q_{j-1}^{n+1}) + (1-\theta)(Q_{j+1}^{n} - Q_{j-1}^{n})] = 0 \qquad (42)
\end{aligned}
$$

where C is the Courant number. To examine stability, assume a Fourier expansion for $Q_j{}^n$ as

$$Q_j{}^n \equiv g^n e^{i(\lambda j \Delta x)} \qquad (43)$$

where $i = \sqrt{-1}$. Substituting equation (43) into equation (42), setting $\theta = 1/2$ and simplifying yields for the amplification factor g,

$$g = \frac{1 + \frac{1}{2}\cos(\lambda\,\Delta x) - i(3C/4)\sin(\lambda\,\Delta x)}{1 + \frac{1}{2}\cos(\lambda\,\Delta x) + i(3C/4)\sin(\lambda\,\Delta x)} \qquad (44)$$

Since the numerator and denominator are complex conjugates, the magnitude of g is unity for all λ, Δt and Δx. Hence, the algorithm is neutrally stable with neither amplification nor damping of waves, as evidenced in the illustrated solutions. In contrast, the diagonalizing operation that yielded equation (28), when applied to the recursive relation form for one-dimensional space, yields for equation (42)

$$Q^{n+1} - Q_j{}^n = -\frac{C}{2}\,[\theta(Q_{j+1}^{n+1} - Q_{j-1}^{n+1}) + (1-\theta)(Q_{j+1}^{n} - Q_{j-1}^{n})] \qquad (45)$$

Equation (45) is immediately recognized as the Crank–Nicolson, second-order accurate finite difference algorithm for equation (2). Hence, the differences in numerical diffusion evidenced in the results in Figure 2 can be explained, at least in part, in terms of order of accuracy.

A definitive assessment of the accuracy and convergence can be achieved in terms of the error norms established in the previous section. For reference, Figure 3a–j presents computed profiles of the wave, following advection over a solution domain of span $4M$, for $M = 10$ and for various Courant and Reynolds numbers, finite element (FE) and finite difference (FD) initial-value matrix forms, and for $k = 1$ and 2 in equation (5). The dashed curves correspond to the analytical solution.

Of particular interest is the comparison between Figures 3c and 3f. In both cases there is approximately 4% loss in the peak value. For Figure 3c, this loss is due to the numerical diffusion associated with the second-order accurate Crank–Nicolson

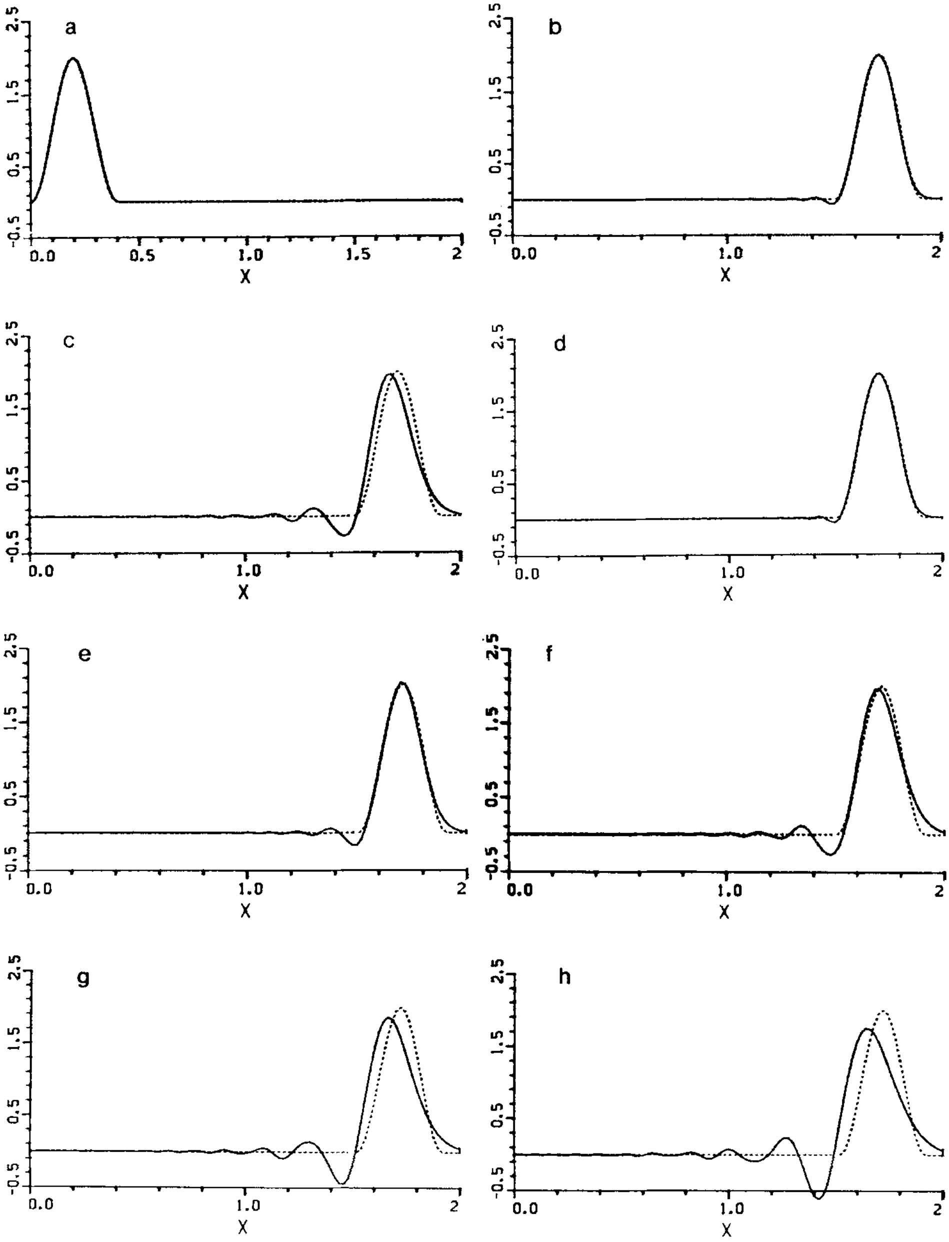

FIG. 3. Advection of a concentration packet. (a) Initial condition, dashed line is analytical solution. (b) Finite element, $k = 1$, $C = 0.4$, $\mathrm{Re} = \infty$. (c) Crank-Nicolson, $C = 0.4$, $\mathrm{Re} = \infty$. (d) Finite element, $k = 2$, $C = 0.4$, $\mathrm{Re} = \infty$. (e) Finite element, $k = 1$, $C = 0.8$, $\mathrm{Re} = \infty$. (f) Finite element, $k = 1$, $C = 1.2$, $\mathrm{Re} = \infty$. (g) Finite element, $k = 1$, $C = 1.6$, $\mathrm{Re} = \infty$. (h) Crank-Nicolson, $C = 1.6$, $\mathrm{Re} = \infty$. (i) Finite element, $k = 1$, $C = 0.4$, $\mathrm{Re} = 10^2$. (j) Finite element, $k = 1$, $C = 0.4$, $\mathrm{Re} = 10^4$.

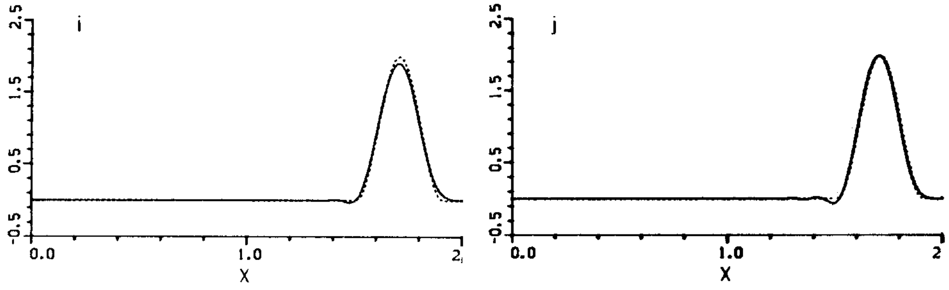

FIG. 3 — *Continued.*

equation (45). When the diffusion coefficient is included, for Figure 3i, the governing equation assumes the form

$$q_t + uq_x - \frac{1}{\mathrm{Re}} q_{xx} = 0 \tag{46}$$

The finite element algorithmic statement for this equation, in comparison to equation (37), is

$$\underset{e}{S}\left[[A200]\{Q\}_e' + \{U\}_e^T\,[A3001]\{Q\}_e + \frac{1}{\mathrm{Re}}\,[A211]\{Q\}_e\right] = \{0\} \tag{47}$$

Both advection and diffusion are present in equation (47), and the 4 % loss in the peak value in Figure 3*i* is associated with a Reynolds number of 100. Note in this casę that the fourth-order accurate finite element integration scheme is essentially free of numerical diffusion as evidenced by the vanishing of the trailing wakes of Figure 3c.

Based upon results of this type, Figures 4-6 present accuracy and convergence rates as determined numerically in the energy and p norms, equations (18)–(21). In Figure 4, the E and p_2 norms appear rather insensitive measures of the differences illustrated in Figures 3b and c. The illustrated quadratic convergence rate with discretization refinement, for both FE and FD forms, is in agreement with the convergence exponent in equation (22) taking the form $2(k + 1 - m) \Rightarrow k + 1 = 2$, as derived by Oden and Reddy (Ref. [18]) for a linear hyperbolic equation. The p_1 norm does respond definitively to the solution differences; the FD convergence rate is 4, while the FE rate is 5, and for $M = 40$ the FE results are a factor of 5 more accurate. It is important to note that this increase in accuracy accrues at no additional cost for an implicit algorithm. Note also that accuracy for the coarse grid solution, $M = 10$, is better than that predicted by strict adherence to the computed convergence rates.

To quantize dispersion error, evaluation of the p_1 (and p_2) norm was split left and right of the computed peak, and convergence with discretization refinement and Courant number determined. For linear and quadratic finite element functions, $k = 1, 2$ equation (5), convergence with discretization refinement in the factored p_1

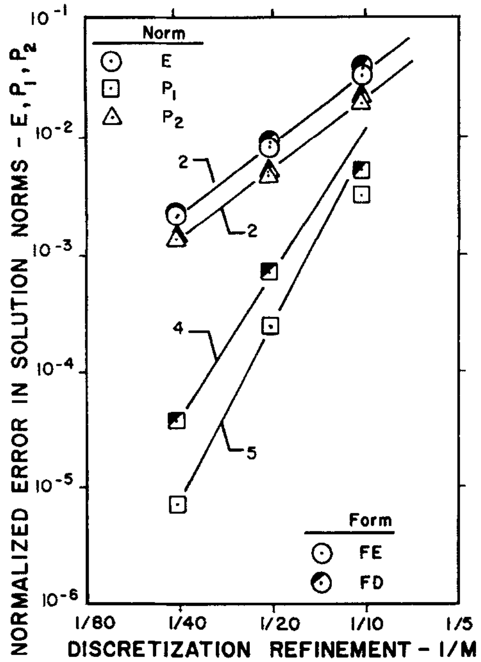

Fig. 4. Accuracy and convergence with discretization refinement in energy, sum and sum-squared norms, hyperbolic equation, linear elements.

norm is uniformly quadratic, see Figure 5. For linear elements, the absolute error in the FD form is 5 times that of the FE form. Most interestingly, for this problem, the quadratic element solutions are insensitive to initial-value matrix structure, with an accuracy comparable to that produced by twice as many linear elements. Of course, since quadratic elements possess an interior node, the rank of $[J]$ (hence work required to generate a solution) for $M = 20$ quadratic and $M = 40$ linear elements is identical. On this basis, the quadratic element solution accuracy is at best a marginal improvement over that of the linear element. Essential quadratic convergence with Courant number, i.e., the trapezoidal integration algorithm, was confirmed in both factored p norms, see Figure 6. On all comparison bases, therefore, the performance of the implicit finite element algorithmic solution form, for equation (2) dominantly hyperbolic, appears superior to the equivalent-complexity finite difference form.

Corresponding evaluations are required for equation (2) parabolic, as obtained by setting $u_i \equiv 0$. As in the hyperbolic case, examination of the one-dimensional statement can best facilitate the study. The critical focus is on predicting algorithm performance at a boundary whereat equation (3) is not identically satisfied; hence the majority of results are generated for equation (2) homogeneous. To preclude exact interpolation of a solution for any specific k, and to demonstrate the facility of mixed interpolation, the problem statement is cast on R^2 spanned by a polar coordinate system. The finite element statement, equations (6)–(7), remains unaltered. However the specific matrix forms, equations (8)–(11), will uniquely reflect this point. For example, the initial value matrix is

$$[C]_e \equiv \iint_{R_e^2} \{\phi_k\}\{\phi_k\}^T r\, dr\, d\theta \tag{48}$$

Since r is interpolated exactly on Ω_e by $k = 1$ in equation (5), and denoting $\{R\}_e$ as the element matrix of node coordinates, equation (48) becomes

$$\begin{aligned}[C]_e &\equiv 2\pi\{R\}_e^T \int_{R_e^1} \{\phi_1\}\{\phi_k\}\{\phi_k\}^T\, dr \\ &= 2\pi\, \Delta_e\{R\}_e^T\, [A3000]\end{aligned} \tag{49}$$

The order and specific entries in the hypermatrix [A3000] depend uniquely upon k as discussed in Appendix A for $1 \leqslant k \leqslant 3$. The several [$A$3000] are stored in BLOCK DATA; the additional storage requirement is only three words per element. Proceeding through similar operations for equations (10)–(11), assuming for simplicity that f vanishes and k and Δ_e are uniform, and cancelling 2π throughout, the statement of

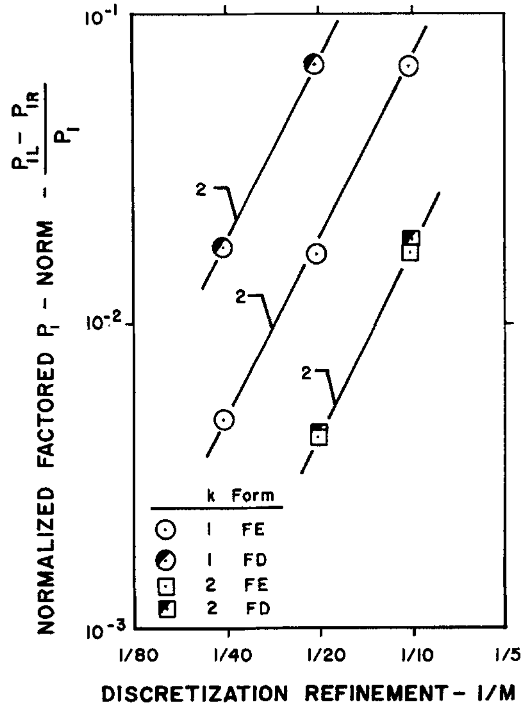

FIG. 5. Accuracy and convergence with discretization refinement in factored sum norm, hyperbolic equation, linear and quadratic finite elements.

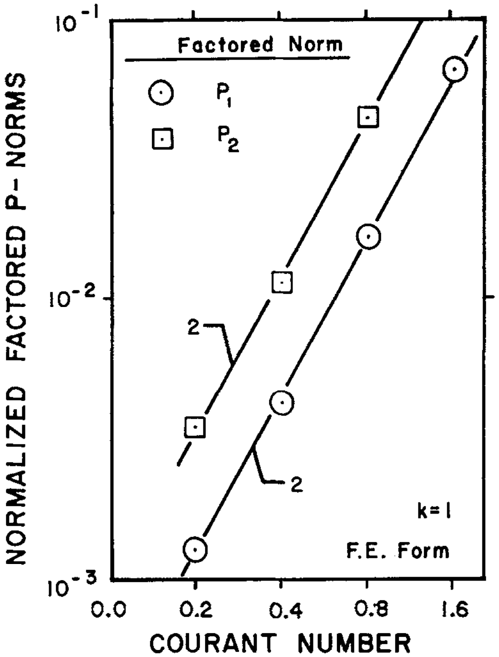

FIG. 6. Accuracy and convergence with Courant number in factored sum and sum-squared norms, hyperbolic equation, linear elements.

the finite element algorithm comparable to equations (32) and/or (37) for solution of the parabolic form of equations (2)–(3) is

$$\underset{e}{S}\,[\{R\}_e^T\,[A3000]\{Q\}'_e + \{R\}_e^T\,[A3011]\{Q\}_e + R(a_1[A0]\{Q\}_e + a_3\{A0\})] \equiv \{0\} \quad (50)$$

The last two terms in equation (50), which represent the finite element equivalent of the gradient boundary condition constraint equation (3), warrant comment. Of primary impact, since these constraints are imbedded directly within the statement for solution of the interior field, the influence of boundary constraints is not retarded by a time step but enforced instantaneously. For one-dimensional elements, the integrals on ∂R_e^1 in equations (10)–(11) become evaluations at a point; hence, since the $\{\phi_k\}$ are cardinal bases, they produce the Kronecker delta, and $\int d\sigma = 2\pi R$, where R is the coordinate radius at which the gradient is applied. Therefore, also, $[A0]$ and $\{A0\}$ define the matrix equivalent of the Kronecker delta, and possess only one non-vanishing element (of unity) independent of k, equation (5). These observations infer that the finite element equivalent of a gradient boundary constraint must be consistent for use with all order-accurate interpolations on Ω_e, as will be confirmed. The matrices are listed in Appendix A.

The test problem for accuracy and convergence evaluations is an axisymmetric hollow cylinder loaded convectively on the interior surface. The energy norm, equation (21), for the flux dependent upon the (computed) surface value of q, is dominated by the boundary contribution, a_1 in equation (3), see also equations (10) and (41). Alternatively, for the fixed flux case, $a_1 \equiv 0$ in equations (3) and (21). For either case, convergence in energy is theoretically predicted by equations (22). Table 1 presents the anticipated convergence exponents for $m = 1$ in the three pertinent norms for the range of k evaluated. Note that $k = 3'$ corresponds to a cardinal basis generated using Hermite cubic interpolation, wherein derivative degrees of freedom are employed (cf. Ref. [74]). Hence, since the derivatives are interpolated

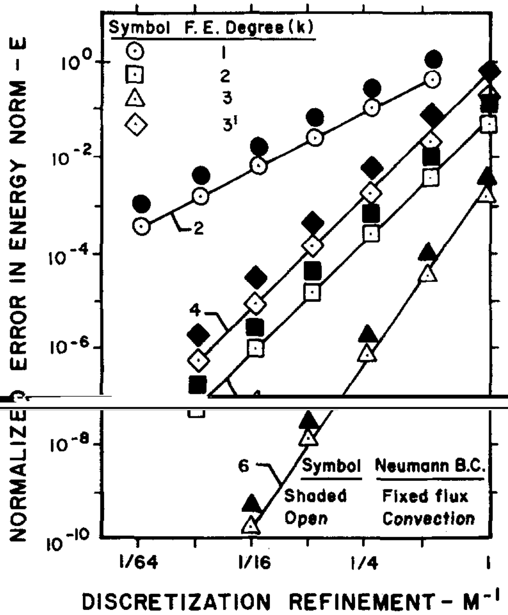

FIG. 7. Accuracy and convergence with discretization refinement in energy norm, parabolic equation, linear quadratic and two cubic finite elements.

one-degree lower than the function, the convergence properties should correspond more closely to those of quadratic interpolation except in e_2. The remaining cardinal bases are generated using Lagrange interpolation, see Appendix A.

The summary assessment of numerically measured accuracy and convergence rates with discretization refinement are presented in Figures 7–11. To confirm practice for the transient equation solutions, Figure 7 presents the numerical results generated for the asymptotic steady-state solution of equation (50), for both fixed and convective Neumann boundary conditions. Essentially exact adherence to the theoretically predicted convergence rates is verified. The absolute error for the fixed case is just slightly larger than for convection, and a dramatic improvement in accuracy is evidenced with an increase in degree of Lagrange interpolation. Accuracy of the Hermite cubic solution is just marginally poorer than the quadratic. Figure 8 presents corresponding results generated at the initiation of the transient solution of equation (50), with curves of even integer slope faired through the refined grid solutions. All but the Hermite cubic demonstrate anticipated convergence rates, and all but the linear element solutions are more accurate on a coarse grid than predicted by strict adherence to these rates. As might be expected, the absolute level of inaccuracy is uniformly greater than for the steady-state solutions. It should also be noted that convergence in energy was not monotonic for any $k > 1$ for this case, as theoretically predicted

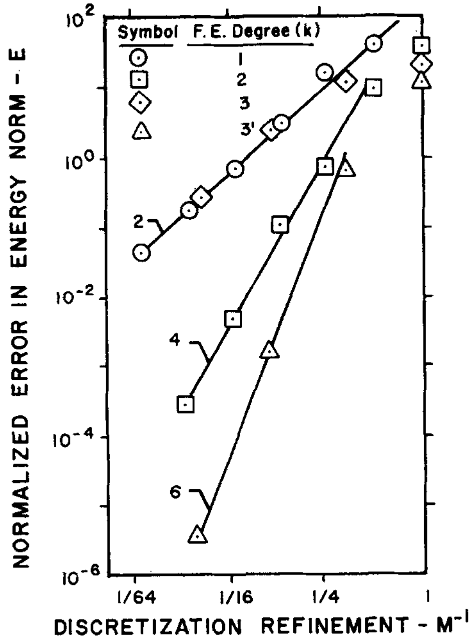

FIG. 8. Accuracy and convergence with discretization refinement in energy norm, transient parabolic equation, linear, quadratic and two cubic finite elements.

and measured for the asymptotic steady-state solution. The relatively poorer overall performance of the Hermite cubic is probably the direct consequence of the extremal surface derivative (hence, degree of freedom in the solution matrix) existent at the start of a gradient boundary condition-driven problem.

The impact of diagonalizing the initial-value matrix structure of the finite element algorithm, equation (50), was evaluated. As shown in Figure 9, the favorable convergence rates measured for the quadratic and cubic elements in Figure 8 have been obliterated. The Hermite cubic and linear element convergence rates are essentially unaffected by diagonlization of $[C]_e$, but the absolute solution error is marginally larger in both cases. A point in defense of a diagonalized initial-value matrix structure, as results for example in a classical finite difference algorithm, can be made when employing linear elements and an explicit integration algorithm. In this case, the maximum allowable integration step-size for which absolute stability can be maintained (cf., Ref. [20]) is limited by the largest eigenvalue of the ordinary differential equation (50). Shown in Table 2 is the maximum eigenvalue spread computed for equation (50) for several discretizations and interpolating degrees as a function of finite element and finite difference initial-value matrix structure. The diagonalization procedure decreases the spread by a factor of roughly 2.5. Hence, the maximum explicit integration step-size for the latter is correspondingly larger.

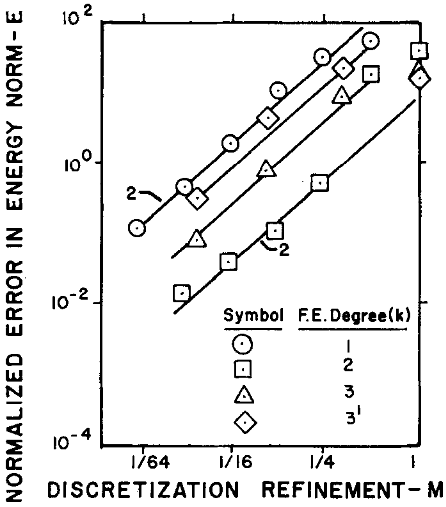

FIG. 9. Accuracy and convergence with discretization refinement in energy norm, transient parabolic equation, finite difference initial-value matrix structure.

TABLE 1

Theoretical Convergence Rates in Three Error Norms for Finite Element Solution of a Linear Parabolic Equation at Steady-State

	Norm convergence rate		
Finite element degree — k	Energy E	Function max e_1	Gradient max e_2
1	2	2	1
2	4	3	2
3	6	4	3
3′	4	3	3

The final evaluation is determination of finite element solution accuracy and convergence in the max norms e_1 and e_2, equations (24)–(25), at the location on $\partial\Omega$ whereat a non-homogeneous Neumann constraint is applied. The evaluations shown

TABLE 2

Eigenvalue Spread of Ordinary Differential Equation (50) as Function of Initial-Value Matrix Structure

k	M	Finite element	Finite difference
1	2	2.90	1.13
	4	14.7	5.17
	8	62.7	21.2
	16	255.	85.2
2	2	17.1	8.08
	4	76.7	32.2
	8	317.	129.
	16	1277.	512.
3	2	50.4	26.9
	4	220.	105.
	8	900.	422.
	16	3622.	1687.
3′	2	11.4	3.80
	4	52.9	16.8
	8	221.	68.8
	16	892.	277.

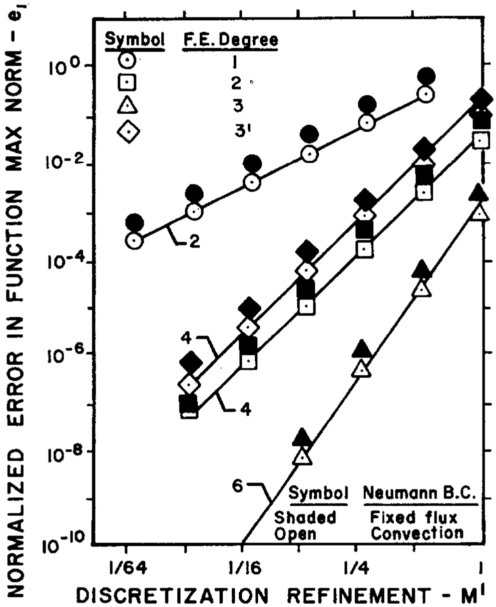

FIG. 10. Accuracy and convergence with discretization refinement in max norm e_1 at boundary, parabolic equation, linear, quadratic and two cubic finite elements.

in Figures 10-11 were established at the asymptotic steady-state solution of equation (50). Figure 10 illustrates convergence rates and absolute solution accuracies in e_1 that are essentially identical to those measured in E, Figure 7. The quadratic and cubic convergence rates are higher than anticipated. Similarly, measured convergence in the surface gradient max norm e_2 is higher than expected, see Figure 11 and Table 1. except for the Lagrange cubic which agrees exactly. Two items of particular note

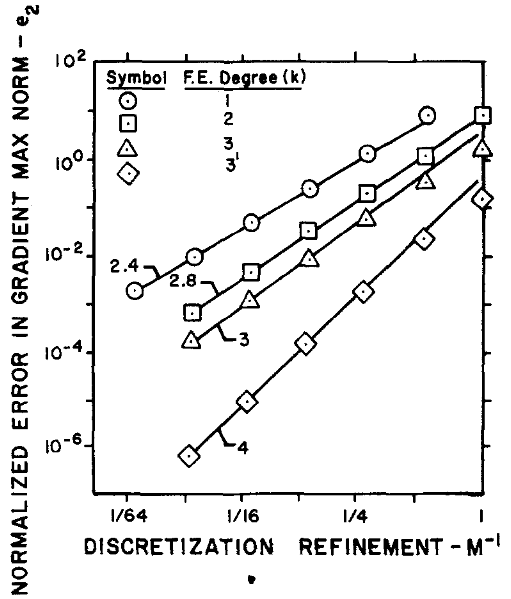

FIG. 11. Accuracy and convergence with discretization refinement in max norm e_2 at boundary, parabolic equation, linear, quadratic and two cubic finite elements.

are that the linear element gradient convergence at the boundary is superior to that of the entire solution on Ω (2.4 vs. 2.0), and that the Hermite cubic, which contains dq/dx_n as a dependent variable, produces a definitely more accurate solution and a high convergence rate for surface flux.

SUMMARY AND CONCLUSIONS

An implicit finite element numerical algorithm for solution of initial-valued parabolic and hyperbolic partial differential equations has been evaluated. Numerical tests were performed to quantize accuracy and assess convergence as a function of discretization refinement, boundary conditions, degree of the finite element interpolating polynomial, and initial-value matrix structure for select parabolic and hyperbolic equations. In all cases tested, the non-diagonal initial-value matrix produced by finite element theory yielded an algorithm of superior performance for no additional computational effort within an implicit integration algorithm. The use of a diagonal structure was assessed to destroy the favorable convergence obtained with higher degree interpolation for the parabolic equation. The overall accuracy and convergence of the Hermite cubics was disappointing. A non-homogeneous gradient boundary condition is not time-retarded within the finite element algorithm, and was observed to produce high order convergence on the boundary of application. These numerous computational features indicate finite element theory viable for development of numerical solution algorithms for initial-boundary value problems.

APPENDIX A: Cardinal Bases and Hypermatrix Formulation

Efficient utilization of the uniformity of the finite element algorithmic procedure is dependent upon formulation of cardinal basis functions and a unified notation. The concepts, which are extendible to multi-dimensional space, can be effectively illustrated one-dimensionally. Figure A.1 shows a one-dimensional element with vertex nodes 1 and 2. Linear interpolation of the distribution of a variable $q(x)$ over R_e^1 is written directly as

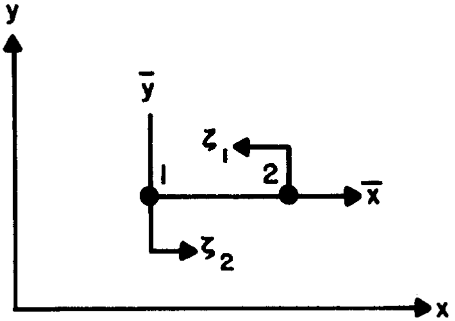

Fig. A.1. One-dimensional finite element coordinate systems.

$$q_e(x) = a + b\frac{x}{\Delta_e} \qquad \text{(A.1)}$$

The coefficients in (A.1) are easily reexpressed in terms of the nodal values Q_i of q_e as,

$$q_e(x) = Q_1\left[1 - \frac{\bar{x}}{\Delta_e}\right] + Q_2\left[\frac{\bar{x}}{\Delta_e}\right] \qquad \text{(A.2)}$$

where the origin of $\bar{x}$ is node 1. Rewriting equation (A.2) in the matrix form of equation (5) yields the linear one-dimensional function set $\{\phi_1(\bar{x})\}$ definition

$$\{\phi_1(\bar{x})\} \equiv \begin{Bmatrix} 1 - \dfrac{\bar{x}}{\varDelta_e} \\ \dfrac{\bar{x}}{\varDelta_e} \end{Bmatrix} \tag{A.3}$$

Equation (A.3) defines the linearly dependent, normalized natural coordinate system of use for one-dimensional space, i.e., $\{\zeta\} = \{\phi_1(\bar{x})\}$. The coordinates of the ζ_i system are also shown in Figure A.1, and all cardinal bases on $R_e^{\,1}$ can be written as polynomials on the ζ_i. Equation (A.1) for general order interpolation becomes

$$q_e(x) = a + b\frac{x}{\varDelta_e} + c\left(\frac{x}{\varDelta_e}\right)^2 + d\left(\frac{x}{\varDelta_e}\right)^3 + \cdots \tag{A.4}$$

Equation (A.4) rewritten in terms of the cardinal basis $\{\phi_k\}$ is

$$q_e(x) \equiv \{\phi_k(\zeta)\}^T\{Q\}_e \tag{A.5}$$

for both Lagrange and Hermite constraints on equation (A.4). Figures A.2 illustrate locations and preferred numbering (to minimize matrix bandwidth) of additional nodes (x) required to evaluate the general expansion coefficients in equation (A.4). The resultant cardinal bases, as specified by equation (A.5), are directly determined as

$$k = 1: \qquad \{\phi_1\} = \{\zeta\} = \begin{Bmatrix} \zeta_1 \\ \zeta_2 \end{Bmatrix} \tag{A.6}$$

$$k = 2: \qquad \{\phi_2\} = \begin{Bmatrix} \zeta_1(2\zeta_1 - 1) \\ 4\zeta_1\zeta_2 \\ \zeta_2(2\zeta_2 - 1) \end{Bmatrix} \tag{A.7}$$

$$k = 3: \qquad \{\phi_3\} = \frac{9}{2}\begin{Bmatrix} \zeta_1(\zeta_2{}^2 - \zeta_2 + \tfrac{2}{9}) \\ \zeta_1\zeta_2(2 - 3\zeta_2) \\ \zeta_1\zeta_2(3\zeta_2 - 1) \\ \zeta_2(\zeta_2{}^2 - \zeta_2 + \tfrac{2}{9}) \end{Bmatrix} \tag{A.8}$$

$$k = 3': \qquad \{\phi_3'\} = \begin{Bmatrix} 1 - (\zeta_2)^2\,(1 + 2\zeta_1) \\ [\zeta_1\zeta_2 - \zeta_1(\zeta_2)^2]\,\varDelta_e \\ (\zeta_2)^2\,(1 + 2\zeta_1) \\ [-\zeta_1(\zeta_2)^2]\,\varDelta_e \end{Bmatrix} \tag{A.9}$$

Note that only for the Hermite cubic, $k = 3'$ (which involves derivative degrees of freedom as well as the dependent variable at vertex nodes), does the basis function depend upon the element measure $\varDelta_e$. Also note that all elements in equations (A.6)–

(A.9) could be written in terms of ζ_1 or ζ_2 only, since they are simply related. The selected forms display parent symmetry.

The significant utility of the cardinal basis is that the integrals required to establish the finite element algorithm, equations (7)–(11), are all readily evaluated since arbitrary polynomials in ζ_i integrate in closed form as

$$\int_{R_e^1} \zeta_1^{p_1}\zeta_2^{p_2}\, d\bar{x} = \Delta_e \frac{p_1!\, p_2!}{(1 + p_1 + p_2)!} \tag{A.10}$$

Therefore, although evaluation of the integrals can be tedious, with the advent of the hypermatrix formalism they need be formed only once and thereafter reside as a DATA statement.

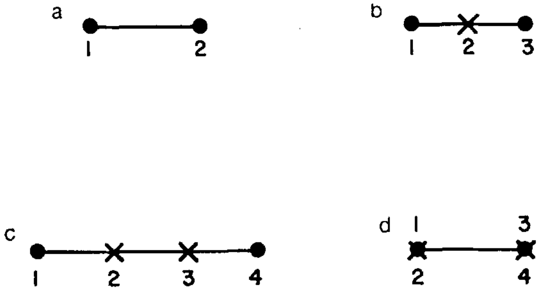

FIG. A.2. Node locations for one-dimensional finite elements spanned by various degree interpolation polynomials $\{\phi_k\}$. (a) Linear, $k = 1$. (b) Quadratic, $k = 2$. (c) Cubic Lagrange, $k = 3$. (d) Cubic Hermite, $k = 3'$.

As an example, the elemental initial value matrix $[C]_e$, equation (8), formed using quadratic interpolation on a one-dimensional space spanned by a Cartesian coordinate becomes

$$[C]_e = \int_{R_e^1} \{\phi_2\}\{\phi_2\}^T\, dx$$

$$= \int_{R_e^1} \begin{Bmatrix} \zeta_1(2\zeta_1 - 1) \\ 4\zeta_1\zeta_2 \\ \zeta_2(2\zeta_2 - 1) \end{Bmatrix} \{\zeta_1(2\zeta_1 - 1), 4\zeta_1\zeta_2\, , \zeta_2(2\zeta_2 - 1)\}\, d\bar{x} \tag{A.11}$$

Expanding and with repeated use of equation (A.10), the result is

$$[C]_e \equiv \Delta_e[A2002] = \frac{\Delta_e}{60} \begin{bmatrix} 8 & 4 & -2 \\ 4 & 32 & 4 \\ -2 & 4 & 8 \end{bmatrix} \tag{A.12}$$

Equation (A.12) illustrates definition of the standard matrix nomenclature. In particular, for the matrix [A2002], the A signifies a matrix on one-dimensional space the first 2 indicates two cardinal bases form the matrix, both 0's indicate neither basis is differentiated, and the final 2 indicates the bases are quadratic, $k = 2$ in equation (A.5). Since [A2002] is element independent, the integers and divisor reside in a DATA statement.

To illustrate mixed interpolation, reevaluate $[C]_e$ on one-dimensional space spanned by a polar coordinate. Then, in equation (8),

$$d\tau = r\,dr = \{R\}_e^T \{\phi_1\}\, d\bar{x} \tag{A.13}$$

which is exact. Hence, with scalar rearrangement

$$\begin{aligned}[C]_e &= \int_{R_e^1} \{R\}_e^T \{\phi_1\}\{\phi_2\}\{\phi_2\}^T \, d\bar{x} \\ &= \{R\}_e^T \int_{R_e^1} \begin{Bmatrix} \zeta_1 \\ \zeta_2 \end{Bmatrix} \begin{Bmatrix} \zeta_1(2\zeta_1 - 1) \\ 4\zeta_1\zeta_2 \\ \zeta_2(2\zeta_2 - 1) \end{Bmatrix} \{\zeta_1(2\zeta_1 - 1), 4\zeta_1\zeta_2\,, \zeta_2(2\zeta_2 - 1)\}\, d\bar{x}\end{aligned} \tag{A.14}$$

From equation (A.11), the outer product on $\{\phi_2\}$ forms a square matrix; hence, in equation (A.14), premultiplication by $\{\phi_1\}$ forms a square matrix with elements which are themselves column matrices. Invoking the definitions presented, and employing equation (A.10), the initial value matrix becomes

$$[C]_e \equiv \Delta_e \{R\}_e^T \,[A30002] = \frac{\Delta_e}{60} \{R\}_e^T \begin{bmatrix} \begin{Bmatrix} 7 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 4 \\ 0 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -1 \end{Bmatrix} \\ & \begin{Bmatrix} 16 \\ 16 \end{Bmatrix} & \begin{Bmatrix} 0 \\ 4 \end{Bmatrix} \\ (\text{sym}) & & \begin{Bmatrix} 1 \\ 7 \end{Bmatrix} \end{bmatrix} \tag{A.15}$$

The hypermatrix $[A30002]$ is element-independent, and the order of each hyperelement depends upon the contraction matrix. Note that if the elements of $\{R\}_e$ are a uniform constant, premultiplication can be completed which yields the useful identity

$$\{1, 1\}[A30002] = [A2002] \tag{A.16}$$

Therefore, lower-order hypermatrices can always be directly obtained. Note also that the order of matrix multiplication is important. Post multiplication of $[A30002]$ by $\{Q\}_e$ cannot be completed before premultiplication by $\{R\}_e^T$.

The hypermatrix concept generalizes readily for all k. For example, consider the element convection matrix $[U]_e$, equation (9). Assuming the velocity field divergence free yields, with scalar rearangement and for Cartesian one-dimensional space

$$[U]_e = \int_{R_e^1} \{U\}_e^T \{\phi_k\}\{\phi_k\} \frac{d}{dx} \{\phi_k\}^T \, d\bar{x} \tag{A.17}$$

The derivative of $\{\phi_k\}$ with x is readily evaluated in terms of ζ_i using the chain rule, see equations (A.6)–(A.9). Assuming for illustration that $k = 2$, equation (A.17) becomes

$$[U]_e \equiv \{U\}_e^T\,[A30012]$$

$$= \frac{1}{\Delta_e}\{U\}_e^T \int_{R_e^1} \begin{Bmatrix} \zeta_1(2\zeta_1 - 1) \\ 4\zeta_1\zeta_2 \\ \zeta_2(2\zeta_1 - 1) \end{Bmatrix} \begin{Bmatrix} \zeta_1(2\zeta_1 - 1) \\ 4\zeta_1\zeta_2 \\ \zeta_2(2\zeta_2 - 1) \end{Bmatrix} \{1 - 4\zeta_1, 4(\zeta_1 - \zeta_2), 4\zeta_2 - 1\}\, d\bar{x} \quad \text{(A.18)}$$

Note that differentiation of $\{\phi_2\}$ has extracted the common multiplier Δ_e^{-1}. Resultant integrations using equation (A.10) will produce Δ_e in the numerator; hence, $[U]_e$ for $1 \leqslant k \leqslant 3$ is element-independent. Proceeding through the algebra, the fully quadratic equivalent for convection becomes

$$[A30012] = \frac{1}{30}\begin{bmatrix} \begin{Bmatrix} -10 \\ -6 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 11 \\ 8 \\ 0 \end{Bmatrix} & \begin{Bmatrix} -2 \\ -2 \\ -1 \end{Bmatrix} \\ \begin{Bmatrix} -6 \\ -16 \\ 2 \end{Bmatrix} & \begin{Bmatrix} 8 \\ 0 \\ -8 \end{Bmatrix} & \begin{Bmatrix} -2 \\ 16 \\ 6 \end{Bmatrix} \\ \begin{Bmatrix} 1 \\ 2 \\ 2 \end{Bmatrix} & \begin{Bmatrix} 0 \\ -8 \\ -12 \end{Bmatrix} & \begin{Bmatrix} -1 \\ 6 \\ 10 \end{Bmatrix} \end{bmatrix} \quad \text{(A.19)}$$

As with the mixed interpolation hypermatrix, the premultiplication of equation (A.19) by $\{U\}_e^T$ is required prior to any conventional matrix operations. The elements of equation (A.19) are stored in a DATA statement.

Equation (10) defines an element matrix involving products of derivatives. Assuming the diffusion coefficient $k_e(x)$ interpolated linearly over R_e^1 spanned by a Cartesian coordinate and neglecting the second term, yields,

$$[K]_e \equiv \{k\}_e^T \int \{\phi_1\} \frac{d}{dx}\{\phi_k\} \frac{d}{dx}\{\phi_k\}^T\, d\bar{x}$$

$$= \frac{1}{\Delta_e}\{k\}_e^T\,[A3011K] \quad \text{(A.20)}$$

For $1 \leqslant k \leqslant 3$, the resultant [A3011$K$] are:

$$k = 1: \qquad [A30111] = \begin{bmatrix} \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -1 \end{Bmatrix} \\ \begin{Bmatrix} -1 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \end{bmatrix} \quad \text{(A.21)}$$

$$k = 2: \qquad [A30112] = \frac{1}{6}\begin{bmatrix} \begin{Bmatrix} 11 \\ 3 \end{Bmatrix} & \begin{Bmatrix} -12 \\ -4 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} \\ \begin{Bmatrix} -12 \\ -4 \end{Bmatrix} & \begin{Bmatrix} 16 \\ 16 \end{Bmatrix} & \begin{Bmatrix} -4 \\ -12 \end{Bmatrix} \\ \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} & \begin{Bmatrix} -4 \\ -12 \end{Bmatrix} & \begin{Bmatrix} 3 \\ 11 \end{Bmatrix} \end{bmatrix} \tag{A.22}$$

$$k = 3: \qquad [A30113] = \frac{1}{80}\begin{bmatrix} \begin{Bmatrix} 262 \\ 34 \end{Bmatrix} & \begin{Bmatrix} -327 \\ -51 \end{Bmatrix} & \begin{Bmatrix} 78 \\ 30 \end{Bmatrix} & \begin{Bmatrix} -13 \\ -13 \end{Bmatrix} \\ \begin{Bmatrix} -327 \\ -51 \end{Bmatrix} & \begin{Bmatrix} 594 \\ 270 \end{Bmatrix} & \begin{Bmatrix} -297 \\ -297 \end{Bmatrix} & \begin{Bmatrix} 30 \\ 78 \end{Bmatrix} \\ \begin{Bmatrix} 78 \\ 30 \end{Bmatrix} & \begin{Bmatrix} -297 \\ -297 \end{Bmatrix} & \begin{Bmatrix} 270 \\ 594 \end{Bmatrix} & \begin{Bmatrix} -51 \\ -327 \end{Bmatrix} \\ \begin{Bmatrix} -13 \\ -13 \end{Bmatrix} & \begin{Bmatrix} 30 \\ 78 \end{Bmatrix} & \begin{Bmatrix} -51 \\ -327 \end{Bmatrix} & \begin{Bmatrix} 34 \\ 262 \end{Bmatrix} \end{bmatrix} \tag{A.23}$$

The second terms in equations (10)–(11) correspond to the Galerkin–Weighted Residuals equivalent of a non-homogeneous gradient constraint on the solution domain closure, see equation (3). Since the cardinal bases degenerate to the Kronecker delta at node coordinates of R_e, the integrals become point evaluations. Hence, from equation (41), and assuming the condition applied at node 1, the boundary gradient constraint matrices for Lagrangian interpolation for all k are imbedded within those for $k = 3$, which are trivially

$$[A0] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ & 0 & 0 & 0 \\ & & 0 & 0 \\ & & & 0 \end{bmatrix} \tag{A.24}$$

$$\{A0\} = \begin{Bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{A.25}$$

The order of equations (A.24)–(A.25) are appropriately reduced to yield the equivalent for $k < 3$.

Extension of the concepts to multi-dimensional space is straightforward, although the algebra can become very detailed. The multidimensional Taylor series

$$q_e(x_i) = a + bx_1 + cx_2 + dx_3 + ex_1x_2 + \cdots \tag{A.26}$$

forms the basis of construction of a cardinal basis. Three and four expansion coefficients are required evaluated, in terms of nodal values in equation (A.26), respectively for two and three-dimensional spaces. Hence, the "natural" element shapes are correspondingly the triangle and tetrahedron. Quadratic and cubic functionals can be directly imposed by uniform extension of the developed one-dimensional concepts (cf., Ref. [19]). Both domain shapes are conveniently spanned by an $(n+1)$-dimensional, normalized linearly-dependent natural coordinate system, and the cardinal basis $\{\phi_k(x_i)\}$ involves polynomials on ζ_i. All integrals are evaluable analytically using the n-dimensional extension of equation (A.10),

$$\int_{R_e^n} \prod_{i=1}^{n+1} \zeta_i^{p_i}\, d\tau = \Delta_e \frac{\pi(p_i)!}{(n + \sum_{i=1}^{n+1} p_i)!} \tag{A.27}$$

where Δ_e is the multi-dimensional measure of R_e^n.

The familiar parallel-sided solution domains are obtained by retaining the bilinear terms in the Taylor series, equation (A.24). This yields four and eight non-vanishing terms respectively for two- and three-dimensional space. Present purposes are served by restricting consideration to the two-dimensional rectangular-shaped domain spanned by the linearly independent η_i natural coordinate system, see Figure 1. The corresponding bilinear cardinal basis, equation (20), is determined by inspection, which yields the interpolation for q_e as

$$q_e(x_1, x_2) = \{\phi_{1+}(\eta_i)\}^T \{Q\}_e \tag{A.28}$$

The resultant two-dimensional bilinear form for the convection matrix $[U]_e$ is

$$[U]_e \equiv \{U_i\}_e^T \int_{R_e^2} \{\phi_{1+}\}\{\phi_{1+}\} \frac{\partial}{\partial x_i} \{\phi_{1+}\}^T \, dx_1\, dx_2 \tag{A.29}$$

Equation (A.29) defines two, 4 × 4 square hypermatrices with elements of order four, since the subscript i is a summation index over the range of $n = 2$. Replacing $\{\phi_{1+}\}$ with a cardinal basis constituted of higher-degree polynomials would increase the matrix orders respectively, as illustrated for the one-dimensional examples. Appendix B lists the various standard matrices required for solution of the sample problem.

APPENDIX B: Two-Dimensional Finite-Element Hypermatrices for Advection-Diffusion

Differential Equation:

$$L(q) = \frac{\partial q}{\partial t} + u_i \frac{\partial q}{\partial x_i} - \mu \frac{\partial^2 q}{\partial x_i^2} = 0 \tag{B.1}$$

Finite Element Algorithm:

$$\underset{e}{S}[\Delta_e[B200]\{Q\}'_e + \Delta_e(\{U\}_e^T\,[B3001] + \{V\}_e^T\,[B3002] + \{XMU\}_e^T\,[B3033])\{Q\}_e] = \{0\} \tag{B.2}$$

Finite Element Hypermatrices for Bilinear Interpolation on R^2:

$$[B200] = \frac{1}{36}\begin{bmatrix} 4 & 2 & 1 & 2 \\ & 4 & 2 & 1 \\ & & 4 & 2 \\ \text{(sym)} & & & 4 \end{bmatrix} \tag{B.3}$$

$$[B3001] = \frac{1}{144a}\begin{bmatrix}
\begin{Bmatrix} -6 \\ -3 \\ -1 \\ -2 \end{Bmatrix} & \begin{Bmatrix} 6 \\ 3 \\ 1 \\ 2 \end{Bmatrix} & \begin{Bmatrix} 2 \\ 1 \\ 1 \\ 2 \end{Bmatrix} & \begin{Bmatrix} -2 \\ -1 \\ -1 \\ -2 \end{Bmatrix} \\
\begin{Bmatrix} -3 \\ -6 \\ -2 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 3 \\ 6 \\ 2 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -2 \\ -2 \\ -1 \end{Bmatrix} \\
\begin{Bmatrix} -1 \\ -2 \\ -2 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 2 \\ 2 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 2 \\ 6 \\ 3 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -2 \\ -6 \\ -3 \end{Bmatrix} \\
\begin{Bmatrix} -2 \\ -1 \\ -1 \\ -2 \end{Bmatrix} & \begin{Bmatrix} 2 \\ 1 \\ 1 \\ 2 \end{Bmatrix} & \begin{Bmatrix} 2 \\ 1 \\ 3 \\ 6 \end{Bmatrix} & \begin{Bmatrix} -2 \\ -1 \\ -3 \\ -6 \end{Bmatrix}
\end{bmatrix} \tag{B.4}$$

$$[B3002] = \frac{1}{144b}\begin{bmatrix}
\begin{Bmatrix} -6 \\ -2 \\ -1 \\ -3 \end{Bmatrix} & \begin{Bmatrix} -2 \\ -2 \\ -1 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 2 \\ 2 \\ 1 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 6 \\ 2 \\ 1 \\ 3 \end{Bmatrix} \\
\begin{Bmatrix} -2 \\ -2 \\ -1 \\ -1 \end{Bmatrix} & \begin{Bmatrix} -2 \\ -6 \\ -3 \\ -1 \end{Bmatrix} & \begin{Bmatrix} 2 \\ 6 \\ 3 \\ 1 \end{Bmatrix} & \begin{Bmatrix} 2 \\ 2 \\ 1 \\ 1 \end{Bmatrix} \\
\begin{Bmatrix} -1 \\ -1 \\ -2 \\ -2 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -3 \\ -6 \\ -2 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 3 \\ 6 \\ 2 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 1 \\ 2 \\ 2 \end{Bmatrix} \\
\begin{Bmatrix} -3 \\ -1 \\ -2 \\ -6 \end{Bmatrix} & \begin{Bmatrix} -1 \\ -1 \\ -2 \\ -2 \end{Bmatrix} & \begin{Bmatrix} 1 \\ 1 \\ 2 \\ 2 \end{Bmatrix} & \begin{Bmatrix} 3 \\ 1 \\ 2 \\ 6 \end{Bmatrix}
\end{bmatrix} \tag{B.5}$$

$$[B3033] = [B3011] + [B3022] \tag{B.6}$$

$$[B3011] = \frac{1}{96a^2}\begin{bmatrix} \begin{Bmatrix}3\\3\\1\\1\end{Bmatrix} & \begin{Bmatrix}-3\\-3\\-1\\-1\end{Bmatrix} & \begin{Bmatrix}-1\\-1\\-1\\-1\end{Bmatrix} & \begin{Bmatrix}1\\1\\1\\1\end{Bmatrix} \\ & \begin{Bmatrix}3\\3\\1\\1\end{Bmatrix} & \begin{Bmatrix}1\\1\\1\\1\end{Bmatrix} & \begin{Bmatrix}-1\\-1\\-1\\-1\end{Bmatrix} \\ & & \begin{Bmatrix}1\\1\\3\\3\end{Bmatrix} & \begin{Bmatrix}-1\\-1\\-3\\-3\end{Bmatrix} \\ \text{(sym)} & & & \begin{Bmatrix}1\\1\\3\\3\end{Bmatrix} \end{bmatrix} \qquad \text{(B.7)}$$

$$[B3022] = \frac{1}{96b^2}\begin{bmatrix} \begin{Bmatrix}3\\1\\1\\3\end{Bmatrix} & \begin{Bmatrix}1\\1\\1\\1\end{Bmatrix} & \begin{Bmatrix}-1\\-1\\-1\\-1\end{Bmatrix} & \begin{Bmatrix}-3\\-1\\-1\\-3\end{Bmatrix} \\ & \begin{Bmatrix}1\\3\\3\\1\end{Bmatrix} & \begin{Bmatrix}-1\\-3\\-3\\-1\end{Bmatrix} & \begin{Bmatrix}-1\\-1\\-1\\-1\end{Bmatrix} \\ & & \begin{Bmatrix}1\\3\\3\\1\end{Bmatrix} & \begin{Bmatrix}1\\1\\1\\1\end{Bmatrix} \\ \text{(sym)} & & & \begin{Bmatrix}3\\1\\1\\3\end{Bmatrix} \end{bmatrix} \qquad \text{(B.8)}$$

Element Measure: Finite element spans $2a \times 2b$, see Figure 1; therefore

$$\Delta_e = 4ab \qquad \text{(B.9)}$$

ACKNOWLEDGMENTS

This research was primarily supported by a NASA Grant, NSG-1261, from NASA Langley Research Center.

The authors wish to express their gratitude to Dr. Jerrold Rosenbaum and Dr. Darrell Pepper for many useful discussions, and to Mr. Paul Manhardt, for programming assistance.

References

1. M. Turner, R. Clough, H. Martin, and L. Topp, *J. Aeronaut. Sci.* **23**, No. 9 (1956), 805–823.
2. O. C. Zienkiewicz and Y. K. Cheung, *Engineer* (1965), 507–510.
3. E. L. Wilson and R. E. Nickell, *Nucl. Engr. Design* **4** (1966), 276–286.
4. B. A. Finlayson and L. E. Scriven, *Int. J. Heat Mass Transfer* **10**, No. 6 (1967), 799–821.
5. B. A. Finlayson and L. E. Scriven, *Appl. Mech. Rev.* **19**, No. 9 (1968), 735–748.
6. B. G. Galerkin, *Engr. Bull.* **19**, No. 9 (1915), 897–908.
7. A. J. Baker, *Int. J. Num. Method Eng.* **6** (1973), 89–101.
8. S. W. Zelazny, A. J. Baker, and W. L. Rushmore, "Modeling of Three-Dimensional Mixing and Reacting Ducted Flows," NASA CR-2661, 1976.
9. M. M. Hafez, E. M. Murman, and L. C. Wellford, "Application of Finite Element Approach to Transonic Flow Problems," pp. 1377–1386, NASA CP-2001, 1976.
10. W. J. Phares and K. R. Kneile, "Solution to the Eulerian Equations by the Finite Element Method with an Application to Transonic Flow," AEDC-TR-76-86, 1976.
11. W. G. Gray, G. F. Pinder, and C. A. Brebbia (Eds.), "Finite Elements in Water Resources," Pentech Press, London, 1977.
12. F. C. Thames, J. F. Thompson, C. W. Mastin, and R. L. Walker, *J. Comput. Phys.* **24**, No. 3 (1977), 245–273.
13. Z. Popinski and A. J. Baker, *J. Comput. Phys.* **21**, No. 1 (1976), 55–84.
14. P. M. Prenter, "Splines and Variational Methods," Wiley, New York, 1975.
15. G. Strang and G. J. Fix, "An Analysis of the Finite Element Method," Prentice–Hall, Englewood Cliffs, N. J., 1973.
16. T. Dupont, *SINUM* **10**, No. 5 (1973), 880–889.
17. T. Dupont, *SINUM* **10**, No. 5 (1973), 890–899.
18. J. T. Oden and J. N. Reddy, "An Introduction to the Mathematical Theory of Finite Elements," Wiley, New York, 1976.
19. O. C. Zienkiewicz, "The Finite Element Method in Engineering Science," McGraw-Hill, London, 1971.
20. P. Henrici, "Discrete Variable Methods in Ordinary Differential Equations," Wiley, New York, 1962.