NASA CONTRACTOR REPORT

NASA CR-2764

THE RELATION OF FINITE ELEMENT AND FINITE DIFFERENCE METHODS

Marcel Vinokur

Prepared by
THE UNIVERSITY OF SANTA CLARA
Santa Clara, Calif. 95053
for Ames Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • DECEMBER 1976

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-2764		

4. Title and Subtitle	5. Report Date
"The Relation of Finite Element and Finite Difference Methods"	December 1976
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Marcel Vinokur	
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
The University of Santa Clara, Univ. Santa, Clara, CA. 95053	NSG-2086

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics & Space Administration Washington, D.C. 20546	Final Technical Report 8/1/75 - 7/31/76
	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

Finite element and finite difference methods are examined in order to bring out their relationship. It is shown that both methods use two types of discrete representations of continuous functions. They differ in that finite difference methods emphasize the discretization of independent variable, while finite element methods emphasize the discretization of dependent variable (referred to as functional approximations). An important point is that finite element methods use global piecewise functional approximations, while finite difference methods normally use local functional approximations. A general conclusion is that finite element methods are best designed to handle complex boundaries, while finite difference methods are superior for complex equations. It is also shown that finite volume difference methods possess many of the advantages attributed to finite element methods.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Numerical Integration Finite Element Method Finite Difference Theory	UNCLASSIFIED-UNLIMITED STAR Category 64

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
UNCLASSIFIED	UNCLASSIFIED	52	$3.75

*For sale by the National Technical Information Service, Springfield, Virginia 22161

THE RELATION OF FINITE ELEMENT AND FINITE DIFFERENCE METHODS

By
Marcel Vinokur

SUMMARY

Finite element and finite difference methods are examined in order to bring out their relationship. It is shown that both methods use two types of discrete representations of continuous functions. They differ in that finite difference methods emphasize the discretization of independent variables, while finite element methods emphasize the discretization of dependent variables (referred to as functional approximations). An important point is that finite element methods use global piecewise functional approximations, while finite difference methods normally use local functional approximations. A general conclusion is that finite element methods are best designed to handle complex boundaries, while finite difference methods are superior for complex equations. It is also shown that finite volume difference methods possess many of the advantages attributed to finite element methods.

INTRODUCTION

The theoretical prediction of a three-dimensional flow past an arbitrary body requires a numerical solution. The traditional approach, which has been highly developed, is to use a finite difference method. Recently, the finite element method has been proposed as an alternative procedure. In order to evaluate the relative advantages and disadvantages of the two methods, it is essential to understand their common basis as well as their fundamental differences. The present work attempts to do this by showing that both methods use two types of discrete representatations of continuous functions. The differences in the two methods stem from the relative emphasis given to these representations.

Since both finite difference and finite element descriptions employ different notations, each with a myriad of indices, we will use a rather cavalier notation with a minimum of indices. The notation, as well as some mathematical concepts that may be somewhat unfamiliar, are discussed in the next section. This is followed by a description of the two types of discrete representations of continuous functions. This framework is then used to examine and relate the finite difference and finite element methods as applied to continuous field problems.

MATHEMATICAL PRELIMINARIES AND NOTATION

A lower case letter will denote a function of real variables, e.g.,

$$u = f(x) \ . \tag{1}$$

The letter f denotes the functional rule, while x stands for a set of independent variables (which may be general, curvilinear coordinates) spanning a domain V with a boundary S. The dependent variable u can represent a vector set of unknowns, in which case (1) is a set of equations. If the dependence on only some of the independent variables will be discretized, we will write

(1) as

$$u = f(x,t) \ , \tag{2}$$

where the dependence on the variables x will be discretized, while the dependence on the remaining variables t will remain continuous. A subscript will denote a partial derivative, or a component of a gradient. Thus, the differential of (1) is written as

$$du = u_x dx = f_x dx \ , \tag{3}$$

where a summation or dot product is implied. On the other hand, a divergence will be denoted by the symbol $\partial/\partial x$. Integrations over a domain or a boundary will be indicated by the letters V or S under the integral sign. If n is the normal at the boundary, the divergence theorem can be written as

$$\int_V \frac{\partial f}{\partial x} dx = \int_S fndx \ . \tag{4}$$

(Note that the sumbol dx has three different meanings in (3) and (4)).

A capital letter will denote an operator acting on a set of functions, e.g.,

$$v = F(u) \ . \ . \tag{5}$$

Here F is the operator rule, u(x) stands for a set of functions, and v(x) the resulting function(s) after performing the operation(s). A local operator involves only algebraic or differential operations, while a non-local operator involves shifting or integral operations. Let $\delta u(x)$ denote the variation of the function u(x), which is a small change in u(x), keeping x fixed. Thus,

$$\delta x \equiv 0 \ , \tag{6}$$

and from their definitions, variation and differentiation are commutative, i.e.,

$$\delta u_x = (\delta u)_x \ . \tag{7}$$

The "differential" of (5) is then defined as

$$\delta v = \delta F(u, u) = \lim_{||\delta u|| \to 0} F(u+\delta u) - F(u) , \tag{8}$$

where $||\delta u||$ is some norm measuring the magnitude of δu. Here $\delta F(u,\delta u)$ is called the Fréchet differential of F at the point u in the direction δu. If F is a local operator, one can define a "derivative". Consider the operator

$$F(u) = f(x,u,u_x,u_{xx}) . \tag{9}$$

Using (6), (7), and (8), one can write

$$\delta F(u,\delta u) = f_u \delta u + f_{u_x} (\delta u)_x + f_{u_{xx}} (\delta u)_{xx} . \tag{10}$$

Using the rule for product differentiation, (10) can be transformed into

$$\delta F(u,\delta u) = \{f_u - \frac{\partial}{\partial x} [f_{u_x} - \frac{\partial}{\partial x} (f_{u_{xx}})]\}\delta u + \frac{\partial}{\partial x} \{[f_{u_x} - \frac{\partial}{\partial x} (f_{u_{xx}})]\delta u + f_{u_{xx}} \delta u_x\}. \tag{11}$$

By analogy with (3), the operator

$$F_u \equiv f_u - \frac{\partial}{\partial x} [f_{u_x} - \frac{\partial}{\partial x} (f_{u_{xx}})] \tag{12}$$

can be called the Fréchet derivative of (9) at the point u. In general, the the Fréchet differential of any local operator F(u) can be expressed as

$$\delta F(u,\delta u) = F_u \delta u + \frac{\partial}{\partial x} (F_o \delta u + F_1 \delta u_x + F_2 \delta u_{xx} + \ldots) . \tag{13}$$

In order to determine if a given operator is a Fréchet derivative, one must define an adjoint operator. If δu_1 and δu_2 are two arbitrary variations of u, one can obtain from (9) the expression

$$\begin{aligned} \delta u_2 \delta F(u,\delta u_1) = {} & \{f_u \delta u_2 - \frac{\partial}{\partial x} [\{f_{u_x} - \frac{\partial}{\partial x} (f_{u_{xx}})\}\delta u_2 - f_{u_{xx}} \delta u_{2x}]\}\delta u_1 \\ & + \frac{\partial}{\partial x} \{[f_{u_x} - \frac{\partial}{\partial x} (f_{u_{xx}})]\delta u_2 \delta u_1 + f_{u_{xx}} \delta u_2 \delta u_{1x} - f_{u_{xx}} \delta u_{2x} \delta u_1\} . \end{aligned} \tag{14}$$

The coefficient of δu_1 in (14) has the form of a Fréchet differential of some other operator in the direction δu_2. We can thus define an operator $\tilde{F}(u)$ (within an arbitrary additive function of x) which is adjoint to F(u), such that

$$\delta\tilde{F}(u,\delta u) = f_u\delta u - \frac{\partial}{\partial x}\{f_{u_x} - \frac{\partial}{\partial x}(f_{u_{xx}})]\delta u - f_{u_{xx}}\delta u_x\}. \tag{15}$$

In general, for any local operator F(u) we can write

$$\delta u_2\delta F(u,\delta u_1) = \delta u_1\delta F(u,\delta u_2) + \frac{\partial}{\partial x}(F_{00}\delta u_2\delta u_1 + F_{01}\delta u_2\delta u_{1x} + F_{10}\delta u_1\delta u_{2x}+\ldots)\,. \tag{16}$$

An operator is self-adjoint if $F(u) = \tilde{F}(u)$. Given the operators G(u), $G_0(u)$, $G_1(u)$, etc., the conditions under which

$$G\delta u + \frac{\partial}{\partial x}(G_0\delta u + G_1\delta u_x + \ldots)$$

is equal to a Fréchet differential $\delta F(u,\delta u)$ can be easily determined from the relation

$$\delta F(u,\delta u_1 + \delta u_2) = \delta F(u,\delta u_1) + \delta F(u + \delta u_1,\delta u_2) = \delta F(u,\delta u_2) + \delta F(u_1 + \delta u_2,\delta u_1). \tag{17}$$

The condition is found to be

$$\begin{aligned} &\delta u_2\delta G(u,\delta u_1) + \frac{\partial}{\partial x}[\delta G_0(u,\delta u_1)\delta u_2 + \delta G_1(u,\delta u_1)\delta u_{2x} + \ldots] \\ &= \delta u_1\delta G(u,\delta u_2) + \frac{\partial}{\partial x}[\delta G_0(u,\delta u_2)\delta u_1 + \delta G_1(u,\delta u_2)\delta u_{1x} + \ldots]. \end{aligned} \tag{18}$$

It follows that G(u) must be a self-adjoint operator. If (18) is satisfied, one can easily show that F(u) is given (within an arbitrary additive function of x) by

$$F(u) = u\int_0^1 G(\lambda u)d\lambda + \frac{\partial}{\partial x}\left[\int_0^1 G_0(\lambda u)d\lambda + u_x\int_0^1 G_1(\lambda u)d\lambda + \ldots\right]. \tag{19}$$

An operator acting on a set of functions which results in a real number is called a functional. (The norm $||\delta u||$ in (8) is an example.) The discussion of Fréchet differentials and adjoint operators reveals the presence of annoying divergence terms. Since these can, in a sense, be removed using the divergence theorem (4), this suggests that a useful functional is the integral of an operator over the domain of x, i.e.,

$$I(u) = \int_V F(u)dx\,. \tag{20}$$

Using (13) and (4), it follows that

$$\delta I(u,\delta u) = \int_V \delta F(u,\delta u)dx = \int_V F_u \delta u dx + \int_S (nF_0\delta u + nF_1\delta u_x + nF_2\delta u_{xx} + \ldots)dx \ . \tag{21}$$

Expressing the gradient at the boundary in terms of normal derivatives, this can be written as

$$\delta I(u,\delta u) = \int_V F_u \delta u dx + \int_S (\overline{F}_0\delta u + \overline{F}_1\delta u_n + \overline{F}_2\delta u_{nn} + \ldots)dx \ , \tag{22}$$

where the subscript n signifies a normal derivative, and $\overline{F}_0$, $\overline{F}_1$, $\overline{F}_2$, etc., are again newly defined operators. Equation (22) is the basis for a variational principle, which is the starting point for one form of the finite element method.

DISCRETE REPRESENTATIONS OF CONTINUOUS FUNCTIONS

Given an arbitrary function of the form (2), the most direct way to discretize the dependence on the variables x is to discretize x itself. The simplest procedure is to choose a set of N arbitrary points x_i (i = 1 to N), and to specify an approximation to u at those points. We thus define N functions of t,

$$u_i^*(t) \rightleftharpoons f(x_i,t) \ , \tag{23}$$

where the superscript * signifies an approximate representation. We will refer to this as a Lagrange representation. In finite element terminology the points x_i are called nodes, and the functions $u_i^*(t)$ are sometimes called nodal parameters. A more sophisticated procedure, requiring a smaller number of points, is to specify also approximations to derivatives of u (which in the most general case need not be consecutive). An example would be to specify the set of first partial derivatives,

$$u_{xi}^*(t) \rightleftharpoons f_x(x_i,t) \ . \tag{24}$$

Such a representation will be called Hermite. Note that each point (node) would now have associated with it more than one parameter. If u represents a set of dependent variables, it is possible to discretize each by a different

set of discrete points. This is often done in practice.

An alternative procedure is to divide the domain of x into N discrete volume elements V^i (i = 1 to N), and to specify an approximation to some functional of u defined over V^i. A typical choice would be the integrated average

$$\overline{u}^{i*}(t) \approx \frac{1}{V^i} \int_{V^i} f(x,t)dx \ . \tag{25}$$

By analogy with a Hermite representation for point discretization, we can define higher order representations for volume discretization by specifying approximations to integrated higher moments of u. Volume discretization is useful in the finite difference solution of equations written in divergence (conservation) form. It is also necessary to define piecewise functional approximations (see below). In finite element terminology, the volume elements V^i are called finite elements.

Both types of discrete representations involve two degrees of freedom. One is the arbitrariness in the location of the points x_i (or volume elements V^i). Any knowledge about the behavior of the function to be approximated can be used to make a judicious choice. The other freedom is the choice of the number and nature of parameters to specify at each point (or volume element). Here the nature of the equations and the numerical scheme can be a determining factor.

Discretization of Dependent Variables

The point discretization discussed above cannot represent integrals, or derivatives of higher order than the order of the representation. Also, a given representation gives no direct information at points other than the discretization points. Therefore, in order to obtain a numerical solution, one must also utilize (even if implicitly) an analytic representation of the arbitrary function. Any analytic function can be represented as an infinite

series in a complete set of chosen functions (providing the series converges). An obvious discretization is to choose N terms, and let the coefficients be the discretization parameters. We will generalize this notion, and approximate u by

$$u^*(x,t) \approx g[x;\ c_i(t)] \ , \tag{26}$$

where g is any arbitrary, chosen function of x and the N parameters c_i (i = 1 to N). The parameters c_i are themselves functions of the undiscretized variables t. If u stands for a set of dependent variables, each one can be represented by a different function g, and the parameters $c_i(t)$ would be sets of parameters.

A general representation which is nonlinear in the c_i cannot be easily integrated, and differentiation can rapidly lead to very complex expressions. For this reason, it is normally used only in curve fitting, and to approximate purely algebraic terms. An exception is the rational function approximation

$$u^*(x,t) \approx \frac{c_{00}(t) + c_{01}(t)x + c_{02}(t)x^2 + \ldots}{c_{10}(t) + c_{11}(t)x + c_{12}(t)x^2 + \ldots} \ , \tag{27}$$

whose derivative maintains a simple form. Since (27) has some advantages over a polynomial, it has found uses in solving equations involving only local operators. In general, though, one chooses a linear representation in the c_i, of the form

$$u^*(x,t) \approx \sum_{i=1}^{N} c_i(t)\phi_i(x) \ , \tag{28}$$

where the $\phi_i(x)$ are an arbitarily chosen set of linearly independent functions, sometimes referred to as basis functions. Since the basis functions should be easily integrated and differentiated, they are often taken to be powers of x, so that (28) becomes a polynomial in x. Other popular choices are trigonometric and exponential functions. Representations (26) and (28) will be referred to as functional approximations, or approximation by trial functions.

An important specialization of the linear representation (28) is to combine it with the point discretization (23) by requiring that u^* equals the nodal parameters u_j^* at a set of N nodes x_j, i.e.,

$$u_j^*(t) = \sum_{i=1}^{N} c_i(t)\phi_i(x_j) \ . \tag{29}$$

Since the $\phi_i(x)$ are linearly independent, one can always choose a set of x_j for which the matrix $\phi_i(x_j)$ is non-singular, and thus solve for the $c_i(t)$ in terms of the nodal parameters $u_j^*(t)$. The $c_i(t)$ are then said to be determined by interpolatory constraints, and the approximation (28) is then called a Lagrange interpolate to $f(x,t)$ at the nodes x_j. It can be represented directly in terms of the $u_j^*(t)$ by introducing new basis functions $\tilde{\phi}_i(x)$, called canonical basis functions, with the defining property

$$\tilde{\phi}_i(x_j) = \delta_{ij} \ , \tag{30}$$

where δ_{ij} is the Kronecker delta. They can be easily obtained from the original basis functions $\phi_i(x)$ by seeking the representation

$$\tilde{\phi}_j(x) = \sum_{i=1}^{N} c_{ji}\phi_i(x) \ . \tag{31}$$

It follows from (30) and (31) that

$$\sum_{i=1}^{N} c_{ji}\phi_i(x_k) = \delta_{jk} \ . \tag{32}$$

Since $\phi_i(x_k)$ is non-singular, the coefficients c_{ji} are uniquely determined by (32). The Lagrange interpolate to $f(x,t)$ at the nodes x_i can thus be expressed succinctly as

$$u^*(x,t) = \sum_{i=1}^{N} u_i^*(t)\tilde{\phi}_i(x) \ . \tag{33}$$

Canonical basis functions can also be defined for Hermite interpolation. For a first order representation, defined by (23) and (24), one can introduce the functions $\tilde{\phi}_{i0}(x)$ and $\tilde{\phi}_{i1}(x)$, satisfying

$$\tilde{\phi}_{i0}(x_j) = \delta_{ij} \ , \ \tilde{\phi}_{i0x}(x_j) = 0 \tag{34}$$

and

$$\tilde{\phi}_{i1}(x_j) = 0 \quad , \ \tilde{\phi}_{i1x}(x_j) = \delta_{ij} \ . \tag{35}$$

These can be obtained in a manner analogous to that described above for Lagrange canonical basis functions. The Hermite interpolate to $f(x,t)$ at the nodes x_i can then be written as

$$u^*(x,t) \rightleftharpoons \sum_i \left[u_i^*(t)\tilde{\phi}_{i0}(x) + u_{xi}^*(t)\tilde{\phi}_{i1}(x)\right] \ , \tag{36}$$

where the summation is over the total number of nodes. More general Hermite interpolates can be similarly formed.

Piecewise Functional Approximation

A single representation of the form (26) or (28) will be poor approximation for functions that undergo rapid variation in the x domain. It is also difficult to construct such representations for domains with complex boundaries when the x domain is multidimensional. It is then advantageous to combine such representations with a volume discretization, and define a separate representation, in each of M volume elements V^j, of the form

$$u^{*j}(x,t) \rightleftharpoons g^j[x; c_i^j(t)] \quad (x\epsilon V^j, \ i=i \text{ to } N^j) \tag{37}$$

in the general case, or

$$u^{*j}(x,t) \rightleftharpoons \sum_{i=1}^{N^j} c_i^j(t)\phi_i^j(x) \quad (x\epsilon V^j) \tag{38}$$

in the linear case, where N^j is the number of parameters in element V^j. Such a representation is called a piecewise functional approximation, or approximation by piecewise trial functions. If the approximations $u^{*j}(x,t)$ are independently chosen in each volume element, the resulting global representation would be discontinuous.

A representation with some degree of continuity requires matching conditions at interelement boundaries, which effectively limits one to the linear case (38). A practical method is to determine the $c_i^j(t)$ by interpolatory constraints. We thus superimpose on the volume discretization an independent point discretization defining a set of N nodes x_i and associated nodal parameters. Matching is simply obtained by locating some of the nodes on interelement boundaries, where they are shared by more than one element. The N^j nodes belonging to element V^j therefore satisfy the inequality

$$\sum_{j=1}^{M} N^j > N . \tag{39}$$

One can again choose the set of nodes x_k so that $\phi_i^j(x_k)$ is nonsingular ($x_k \varepsilon V^j$) in each element V^j. This condition will be sufficient to obtain continuity for an arbitary set of $\phi_i^j(x)$ if x is a one-dimensional variable, but continuity for multidimensional domains imposes restrictions on the set $\phi_i^j(x)$. To show this clearly, we first discuss the one-dimensional case, but in a manner that can be immediately generalized to several dimensions.

One-dimensional Representation. Let x be one-dimensional, and consider a piecewise representation (38) that is everywhere continuous, but whose derivatives can be discontinuous at interelement boundaries. It is therefore sufficient to choose Lagrange interpolation, placing one node at each interelement boundary, and additional nodes in the interior of each V^j for which $N^j > 2$. One can then again introduce new basis functions $\tilde{\phi}_i(x)$, called Lagrange cardinal basis functions, satisfying

$$\tilde{\phi}_i(x_k) = \delta_{ik} \tag{40}$$

for all i and k. In those elements V^j which do not contain x_i, $\tilde{\phi}_i(x)$ must interpolate to zero at all the element nodes. Since $\phi_i^j(x_k)$ is assumed non-

singular, it follows that $\tilde{\phi}_i(x) = 0$ in those elements. Thus $\tilde{\phi}_i(x)$ is non-zero only over those elements containing node x_i, i.e., two adjoining elements for a boundary node and a single element for an interior node. One-dimensional Lagrange cardinal basis functions for elements containing one interior node are sketched in the top row of figure 1. Since the interelement boundaries consist of one point at which an interpolating node is located, the functions $\tilde{\phi}_i(x)$ are continuous. Consequently, the global representation

$$u^*(x,t) \approx \sum_{i=1}^{N} u_i^*(t)\, \tilde{\phi}_i(x) \tag{41}$$

is also continuous. The localized nature of the $\tilde{\phi}_i(x)$ has important computational advantages. For example, integrals of products of $u^*(x,t)$ over the domain define matrix elements k_{ij} given by

$$k_{ij} = \int_V \tilde{\phi}_i(x)\tilde{\phi}_j(x)dx \ . \tag{42}$$

It is seen that $k_{ij} = 0$ unless nodes i and j are contained in the same element, so that k_{ij} is a sparse matrix. Similar results hold for integrals of products of derivatives of $u^*(x,t)$.

In finite element applications it is convenient to define for each element V^j a set of element cardinal basis functions $\tilde{\phi}_i^j(x)$ satisfying

$$\tilde{\phi}_i^j(x_k) = \delta_{ik} \quad (x_i, x_k \epsilon V^j) \ . \tag{43}$$

If one extends the $\tilde{\phi}_i^j(x)$ by defining them to equal zero if x_i or x lie outside of V^j, i.e.,

$$\tilde{\phi}_i^j(x) \equiv 0 \quad \text{if} \quad x_i \ \text{ or } \ x \epsilon V^j \ , \tag{44}$$

one can then represent $u^*(x,t)$ for each V^j as

$$u^{*j}(x,t) \approx \sum_{i=1}^{N} u_i^*(t)\tilde{\phi}_i^j(x) = \sum_{i \epsilon V^j} u_i^*(t)\tilde{\phi}_i^j(x) \ , \tag{45}$$

where the second form results from (44). It also follows from (44) that

$$u^{*j}(x,t) = 0 \quad \text{if} \quad x \notin V^j \ . \tag{46}$$

Using (43) through (46), one shows immediately that global and element representations are related by

$$\tilde{\phi}_i(x) = \sum_{j=1}^{M} \tilde{\phi}_i^j(x) \tag{47}$$

and

$$u^*(x,t) = \sum_{j=1}^{M} u^{*j}(x,t) . \tag{48}$$

In order for (47) and (48) to be valid at interelement boundaries, the volume elements V^j must be considered disjoint, and to butt together at the boundaries. Element basis functions $\tilde{\phi}_i^j(x)$ are sketched in the bottom row of figure 1.

Another useful computational device is to define for each element V^j a set of local coordinates x^j, each related to the global coordinates x through transformations $x = x(x^j)$ and $x^j = x^j(x)$. (A special case is $x^j = x$). The nodes contained in each element V^j can then be designated as x_i^j, where i is a local node number ($i = 1$ to N^j), completely independent of its global node number. Thus there exist mapping relations which map local node numbers into global node numbers, and vice versa. The local nodal parameters attached to a local node x_i^j are designated as $u_i^{*j}(t)$. The element cardinal basis function corresponding to local node x_i^j would then be written as $\tilde{\phi}_i^j(x^j)$, and the representation of $u^*(x,t)$ in V^j becomes

$$u^{*j}(x^j,t) = \sum_{i=1}^{N^j} u_i^{*j}(t)\tilde{\phi}_i^j(x^j) \ . \tag{49}$$

The use of local coordinates can result in functions $\tilde{\phi}_i^j(x^j)$ that are easy to manipulate analytically. A major advantage results if all the volume elements are geometrically similar in x space (which is trivially so in one dimension), since then they can all be described by the identical local coordinates. If the same set of basis functions $\phi_i^j(x^j)$ is chosen for each

element, and the nodes x_i^j are defined at geometrically similar locations, the element cardinal basis functions $\tilde{\phi}_i^j(x^j)$ will also be the same for all elements. It is thus possible to create a single subroutine, valid for all elements, in order to perform calculations for a single element. Of course, in summing the results to obtain a global solution, the coordinate transformations and node number mappings must be invoked.

If continuity of derivatives is required for the piecewise representation, one must use Hermite interpolation. It is only necessary to define derivatives at boundary nodes, and not at interior nodes. In fact, in most applications of piecewise Hermite interpolation, nodes are only defined at interelement boundaries. The extension of this subsection to piecewise Hermite interpolation follows the general manner indicated by (34) through (36) for the case of a single, global Hermite interpolation.

Tensor Products

A piecewise representation can be easily obtained in several dimensions if the global boundaries of the domain lie along the coordinate surfaces. One can then choose volume elements and nodes to lie along coordinate surfaces, and construct cardinal basis functions which are products of one-dimensional cardinal basis functions known as tensor products. We indicate the process for two dimensions, departing from our notational convention, by using x and y to represent the two (not necessarily Cartesian) coordinates.

Let V^k, x_i, and $\tilde{\phi}_i(x)$ be one-dimensional volume elements, global nodes, and Lagrange cardinal basis functions along the x coordinate. Similarly, define V^ℓ, y_j, and $\tilde{\phi}_j(y)$ to be one-dimensional volume elements, global nodes, and Lagrange cardinal basis functions along the y coordinate. These define two-dimensional volume elements designated as $V^{k\ell}$, and the double index node number ij for the node located at x_i and y_j. The function

$$\tilde{\phi}_{ij}(x,y) \equiv \tilde{\phi}_i(x)\tilde{\phi}_j(y) \tag{50}$$

has the property

$$\tilde{\phi}_{ij}(x_m,y_n) = \delta_{im}\delta_{jn} , \tag{51}$$

and is therefore a two-dimensional Lagrange cardinal basis function. Consequently,

$$u^*(x,y,t) \approx \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} u^*_{ij}(t)\tilde{\phi}_{ij}(x,y) \tag{52}$$

where

$$u^*_{ij}(t) \approx f(x_i,y_j,t) \quad (i=1 \text{ to } N_x,\ j=1 \text{ to } N_y) . \tag{53}$$

Representation (52) is everywhere continuous, and can be extended to higher dimensions and to the case of Hermite interpolation.

General Multidimensional Representation

If the global boundaries of a multidimensional domain are too complex to allow a tensor product piecewise representation, one must use volume elements of a more general shape. All the results of the subsection on the one-dimensional representation can be immediately generalized, with the exception of the continuity conditions. If x_i is an interior node located in element V^j, we require that $\tilde{\phi}_i(x)$ (or $\tilde{\phi}^j_i(x)$) equals zero on the boundaries of V^j. But this is only guaranteed at the boundary nodes of V^j. Thus the combination of basis functions $\phi^j_i(x)$ in (38) and nodes x_i cannot be arbitrarily chosen, but must be such as to yield $\tilde{\phi}_i(x) = 0$ on the boundary for interior nodes of V^j. If x_i lies on one or more boundaries of V^j, then we require that $\tilde{\phi}^j_i(x) = \tilde{\phi}^k_i(x)$ on each boundary for all other volume elements V^k sharing that boundary. In addition we still require that $\tilde{\phi}^j_i(x) = 0$ on the boundaries of V^j that do not contain x_i. Piecewise Hermite interpolation puts even more stringent requirements on the $\phi^j_i(x)$.

Up to this time, the shape of the volume elements V^j and the nature of the basis functions $\phi^j_i(x)$ have been considered arbitrary. The above-mentioned

continuity requirements effectively limit one to triangles (or tetrahedrons) in x space (which could be curvilinear in physical space), and polynomials in x for the $\phi_i^j(x)$. It also puts restrictions on the location of the nodes x_i. The simplest case is Lagrange interpolation with linear basis functions $\phi_i^j(x)$, for which one only requires nodes at the vertices of the triangles (or tetrahedrons). The finite element literature is replete with various combinations of nodes x_i and corresponding cardinal basis functions (usually called shape functions) $\tilde{\phi}_i^j(x)$, for both triangles and tetrahedrons, and for Lagrange and Hermite interpolation.

The polynominal nature of the $\phi_i^j(x)$ also allows one to estimate the errors in $u^*(x)$, when $f(x)$ is assumed exact at the nodes (so that (23) is an exact equality). (We suppress the dependence on t for the moment.) Such estimates are derived in reference 1, where it is shown that the error bound for Lagrange interpolation over a triangle is inversely proportional to the sine of the smallest angle. This would rule out extremely acute triangles. Actually, the author has shown (ref. 2) that the sine of the largest angle enters into the error bound, ruling out only extremely obtuse triangles. For the simple linear case, the author obtained least upper bounds for the errors. Let

$$M \equiv |f_{xx}|_{max} \tag{54}$$

be the maximum absolute value of the second directional derivative of f in any direction, at any point in the triangle. If θ and h are the maximum angle and side of the triangle, then the results are

$$|u^*-f| \leq \frac{Mh^2}{6} \tag{55}$$

and

$$|(u^*-f)_x| \leq \frac{Mh}{2\sin\theta} , \tag{56}$$

where $|(u^*-f)_x|$ is the magnitude of the gradient of u^*-f.

Since arbitrary, curved global boundaries cannot be easily fit by a global curvilinear coordinate, one is faced with the need to use curvilinear elements in $\underline{x}$ space. This can be done if one can find transformations $x(\xi)$ which transform the curvilinear elements in x space into straight sided "parent" elements in ξ space. The representation of u over the curvilinear element is only approximate, being accurate only at the nodes. It is therefore sufficient to treat the transformations $x(\xi)$ in the same manner. This is the basis for the isoparametric transformations developed by Irons (ref. 3). Let $x^j = x$ be the local coordinates for the curvilinear element, and x_i^j be a set of local nodes chosen on the boundary of (and possibly within) the element. (There must be at least 3 nodes per side and at least 4 nodes per face to define curved boundaries.) In the transformed ξ plane, let ξ^j, ξ_i^j, and $\tilde{\phi}_i^j(\xi^j)$ be local coordinates, local nodes, and element cardinal basis functions for the corresponding "parent" element. Then the isoparametric transformation has the approximate representation,

$$x^{*j}(\xi^j) \approx \sum_{i=1}^{N^j} x_i^j \, \tilde{\phi}_i^j(\xi^j) \ . \tag{57}$$

(In some cases it is practical to use a lower (higher) number of nodes and order of basis function to represent the geometric transformation than are used to represent $u^{*j}(x)$ over the element. Such transformations are then called sub (super) parametric.) Using (57) and its derivatives, integrals over element V^j in x space can be transformed into integrals over the "parent" element in ξ space.

Splines

The piecewise representations discussed so far involved only interpolatory constraints to determine the $c_i^j(t)$ in (38). If additional smoothness constaints are imposed by matching higher derivatives (than those prescribed by interpola-

tion) at boundary nodes, the representations are called splines. The additional continuity requirements on the $\phi_i^j(x)$ are so great for general multidimensional elements as to make such spline representations totally impractical. We are thus restricted to one-dimensional splines (and their tensor product generalization to higher dimensions). In practice, splines are further limited to volume elements with Lagrange interpolatory nodes only at the two ends of each element. Thus for a division of the one-dimensional x space into M volume elements, the total number of nodes N = M + 1. Smoothness constraints are applied at the M - 1 nodes that lie in the interior of the global domain, and are

$$\sum_{j=1}^{M} N^j - 2M$$

in number, where we recall that N^j is the number of parameters in element V^j. In the usual case where N^j is the same for all V^j, one can specify exactly $N^j - 2$ smoothness constraints at each of the (M - 1) interior nodes, leaving exactly $N^j - 2$ conditions to be specified at the two ends of the global domain. If there are no additional end conditions on the function f(x) to be represented, the $N^j - 2$ conditions must be arbitrarily specified and apportioned at the two ends. Such splines are therefore not unique. It is also clear that an even number for N^j will prevent a bias towards one end. While splines can be constructed for arbitrary $\phi_i^j(x)$, in most applications they are limited to polynomials.

One can again construct cardinal basis functions $\tilde{\phi}_i(x)$, and employ representation (41). Since the smoothness constraints couple the elements together, the cardinal basis functions are not at all localized, but extend over the global domain. They are thus inconvenient for computational purposes. There are two approaches that are used. In one, the original basis $\phi_i^j(x)$ is used, and the derivatives u_{xi}^*, u_{xxi}^*, etc., are introduced as additional unknowns. The

interpolation and matching conditions enable one to solve for these derivatives in terms of the u_i^*, by inverting banded matrices. The other procedure, valid for equal intervals, is to introduce a new basis $\phi_i^B(x)$, known as B splines, which possess the smoothness property, but do not have the cardinal property $\phi_i^B(x_j) = \delta_{ij}$. The B splines are non-zero only over N^j elements, and thus have a localized nature. The coefficients of B spline expansions can again be obtained in terms of the u_i^* by inverting banded matrices. A popular choice for polynomial splines is piecewise cubic ($N^j = 4$), which leads to easily invertible tridiagonal matrices. An elementary discussion of splines is found in reference 1.

Our discussion of functional approximations was aimed at their application in numerical solutions of operator equations. Their obvious role is to obtain expressions for derivatives and integrals in terms of nodal parameters, and to evaluate functions at points other than nodes. There are several other applications, which should be briefly mentioned.

One application is to use piecewise functional approximations to obtain approximate analytic solutions of certain differential equations. In this method, known variable coefficients are replaced by simpler piecewise representations in terms of known nodal parameters, so that the resulting equations possess an analytic solution in each element. The unknown solution coefficients are obtained by matching the solutions and their derivatives at interelement boundaries and applying boundary conditions at the global boundaries. The solution of the differential equations is thus reduced to that of an algebraic system for the coefficients. Further details are found in the works of Gordon (ref. 4) and Canosa and de Oliveira (ref. 5).

Another important area of application is the representation of complex surfaces in physical space. The independent variables x are the two parameters defining parametric curves on the surface, while u stands for the position

vector. If the surface is very complex, one needs a piecewise representation, dividing the surface into patches. One class of such representation uses tensor products, interpolating through data given at the corners of the patches. Examples are programs developed at McDonnell Douglas (ref. 6), using piecewise cubic Hermite polynomials, and the work of Riesenfeld (ref. 7) employing B splines. In another class of representation, the curves defining the boundary of the patch are analytically prescribed, and one seeks what are referred to as blended interpolations for points inside the patch. Examples are the work of Coons (ref. 8) using Hermite polynomials, and Gordon (ref. 9) using splines. All of these approximate surface representations can play an important role in generating finite difference and finite element grids and formulating surface boundary conditions, for the solution of flows past complex boundaries.

In closing we list the various degrees of freedom in a functional approximation. One is the choice of single versus piecewise representation, and the nature, size, and location of volume elements in the latter instance. Another is the functional form, which involves a choice of basis functions in the linear case. The number, location, and nature of interpolating nodes is another degree of freedom. Finally, for piecewise representations, there is the choice of using additional smoothness constraints to define splines. While continuity and convergence criteria make some of these choices interdependent, it still allows for large degree of flexibility in constructing functional approximations.

FORMULATION OF THE EXACT EQUATIONS

There are two mathematically equivalent ways to formulate the equations describing continuous fields. In the direct approach, the equations and boundary conditions of the problem are given. For certain classes of equations, an indirect variational formulation is possible, which incorporates some of the boundary conditions. A finite difference numberical solution is

usually based on the direct formulation, while the variational formulation is the starting point for one form of the finite element method. These two formulations are briefly discussed below, where x will stand for the complete set of independent variables.

Direct Formulation

The normal way to formulate a field problem is to specify that u(x) is a solution of an operator equation (s)

$$G(u) = 0 \ . \tag{58}$$

A complete specification requires subsidiary equations valid on subspaces of x called boundaries. If S^i represents the ith boundary subspace, the subsidiary equations

$$B^i(u) = 0 \ , \quad x \epsilon S^i \tag{59}$$

are called the boundary conditions on S^i. The boundary S^i can be prescribed or free, i.e., implicitly defined in terms of another operator equation. In many problems S^i is defined in the limit as x approaches infinity. If $S = \sum_i S^i$ defines a closed subspace, then (58) and (59) define a boundary value problem. If x is one-dimensional, one can also have an initial value problem, where all the boundary conditions are specified at only one boundary. For multidimensional x, a mixed type of initial boundary value problem is possible, which is an initial value problem with respect to one (time-like) independent variable, and a boundary value problem in the subspace defined by the other independent variables.

Two other points should be made with respect to a problem formulation. In certain problems, internal boundaries (such as shocks or slip surfaces) can occur, where the solution is discontinuous. The conditions at these surfaces are not boundary conditions in the sense used here, since they are actually limiting forms of the field equations (58). The other point refers to certain

classes of boundary value problems, in which a well behaved solution only exists for specific values of certain parameters, called eigenvalues. In eigenvalue problems, the determination of the eigenvalues can be an important, if not the principal objective.

Variational Formulation

An indirect variational formulation exists for boundary value problems in which the operator G(u) (58) is self adjoint. Then G(u) is a Fréchet derivative of another operator F(u) , i.e.,

$$G = F_u \ . \tag{60}$$

The operator F(u) defines the integral functional I(u) given by (20), whose Fréchet differential is given by (22). A variational statement of the original problem states that

$$\delta I(u,\delta u) = 0 \tag{61}$$

for all variations δu. This immediately implies (58), the original operator equation, which is then referred to as the Euler equation. But it also requires the boundary conditions (59) on each S^i to be such that the boundary integral in (22) is equal to zero. If only the first term exists, the requirement is that either δu is zero, i.e., u is prescribed on the boundary, or $\overline{F}_o$ is zero. Thus (59) would be limited to

$$B^i(u) = u - \phi^i(x) \tag{62a}$$

or

$$B^i(u) = \overline{F}_o(u) \ , \tag{62b}$$

where $\phi^i(x)$ is a prescribed function of x on the boundary S^i. If the second term also exists in the boundary integral in (22), we additionally require that either u_n is prescribed, or $\overline{F}_1$ is zero, etc. The conditions u, u_n, etc., prescribed are called the <u>principal</u> boundary conditions, while the alternate conditions $\overline{F}_o = 0$, $\overline{F}_1 = 0$, etc., are called the <u>natural</u> boundary conditions.

Thus, corresponding to each term in the boundary integral in (22), a boundary condition (59) must exist on each S^i, which is either the principal condition, or the natural condition determined implicitly by (58). The variational statement (61) is thus subject to the constraints of the principal conditions, but automatically incorporates the natural conditions.

There are problems for which $G(\tilde{u})$ is self adjoint, which involve boundary conditions (59) that are neither principal nor natural, as defined above. It is usually possible to extend the variational principle to include those cases. Let

$$H^i(u) = h^i(x,u,u_n,u_{nn}, \ldots) , \quad x\varepsilon S^i \tag{63}$$

be a local operator defined on the boundary subspace S^i. The Fréchet differential can be written as

$$H^i(u,\delta u) = H^i_o\delta u + H^i_1\delta u_n + H^i_2\delta u_{nn} + \ldots , \quad x\varepsilon S^i \tag{64}$$

Then the extended integral functional

$$I(u) = \int_V F(u)dx + \sum_i \int_{S^i} H^i(u)dx \tag{65}$$

has the Fréchet differential

$$\delta I(u,\ u) = \int_V F_u\delta u dx + \sum_i \int_{S^i} [(\overline{F}_o + H^i_o)\delta u + (\overline{F}_1+H^i_1)\delta u_n + (\overline{F}_2+H^i_2)\delta u_{nn} + \ldots]dx . \tag{66}$$

The extended variational principle (61) now possesses extended natural boundary conditions $\overline{F}_o + H^i_o = 0$, $\overline{F}_1 + H^i_1 = 0$, etc. For most cases, one can find operators $H^i(u)$ such that boundary conditions (59) that are not principal conditions can be made to be extended natural conditions as defined by the extended functional (65). One can also show that several different choices for $H^i(u)$ are possible in some situations.

When (58) or (59) involve several equations it is possible to handle some of them using Lagrange multipliers. Specifically, if (58) or (59) are

replaced by

$$G(u) = 0 \quad \text{and} \quad G_o(u) = 0 \tag{67}$$

and

$$B^i(u) = 0 \quad \text{and} \quad B^i_o(u) = 0 \qquad x\varepsilon S^i \ , \tag{68}$$

where $G(u)$ is the Fréchet derivative of $F(u)$, the variational principle can then be stated in terms of the functional

$$I(u) = \int_V [F(u) + \lambda G_o(u)]dx + \sum_i \int_{S^i} [H^i(u) + \mu^i B^i_o(u)]dx \ , \tag{69}$$

where the functions $\lambda(x)$ and $\mu^i(x)$ are parameters to be varied independently. $G(u) = 0$ and $G_o(u) = 0$ are the Euler equations corresponding to the variations of δu and $\delta\lambda$, respectively. Similarly, some of the equations $B^i(u) = 0$, and $B^i_o(u) = 0$, are the natural conditions corresponding to the variation of δu, δu_n, ..., and $\delta\lambda$ on the boundary S^i. Sometimes the roles of $G(u)$ and $G_o(u)$ can be reversed, leading to alternate variational principles for the same problem.

The variational formulations discussed so far have been restricted to prescribed boundaries S^i. We indicate the modification due to a free boundary by considering the case where the boundary S^i and boundary condition $B^i(u)$ are determined by the solution of

$$g^i[u(x),x] = 0 \ , \tag{70}$$

which implicitly defines $S^i(u)$ and $V(u)$. The functional $I(u)$ is now written as

$$I(u) = \int_{V(u)} F(u)dx \ , \tag{71}$$

and its Fréchet differential is

$$\delta I(u,\delta u) = \int_{\overline{V}} \delta F(u,\delta u)dx + \sum_i \int_{\overline{S}^i} F(u)\delta n^i dx \ , \tag{72}$$

where δn^i is the amount $\bar{S}^i$ moves normal to itself due to the variation $\delta u(x)$, and $\bar{V}$ and $\bar{S}^i$ are the domain and boundary before u is varied. If $\delta n(\bar{S}^i)$ represents the variation δu at the fixed boundary $\bar{S}^i$, one can show from (70) that

$$\delta n^i = \frac{- g_u^i \delta u(\bar{S}^i)}{g_n^i + g_u^i u_n} \quad . \tag{73}$$

Combining (72) and (73), we find that the free boundary modifies the natural boundary condition to read

$$\bar{F}_o - \frac{F g_u^i}{g_n^i + g_u^i u_n} = 0 \quad . \tag{74}$$

The variational formulation has two advantages over the direct formulation. The operator F(u) is a lower order operator than G(u), permitting a functional approximation with a lower degree of continuity. Also, since the variational formulation has the natural boundary conditions built into it, it therefore has fewer boundary conditions to satisfy than the direct formulation. It has the disadvantage of being indirect, and only existing for a certain class of problems.

We are now ready to examine how the two types of discretizations discussed in the previous section are used to obtain approximate solutions to continuous field problems, starting from either of the two formulations discussed above.

FINITE DIFFERENCE METHODS

Any approximate method of solving a continuous field problem whose starting point is the discretization of some of the independent variables will be termed a finite difference method. The most common procedure employs point discretization at N nodes x_i, with their associated unknown nodal parameters which can be functions of the variables t. This lends itself naturally to the direct formulation, by evaluating (58) approximately at N evaluation points $\bar{x}_j$, which do not necessarily coincide with the x_i. (Recall that if u stands for several

dependent variables, each may be discretized by a different set of nodes.) The operator G(u) involves differential operators in x which must be approximated by finite difference operators in terms of the unknown nodal parameters. This has two consequences. In order to obtain simple difference approximations, it is highly desirable to choose the nodes to lie along coordinate surfaces if x is multidimensional. The other point refers to the nature of the functional approximation which is implied by the difference approximation. In the previous section, functional approximations were defined over global regions. Yet in initial value and initial boundary value problems, the solution along the time-like coordinate is only known up to the point presently reached in the calculation. Even in boundary value problems, a difference approximation based on a global functional approximation would be overly complex, and result in the need to invert very dense matrices. For these reasons, traditional finite difference approximations are based on functional approximations that interpolate data at nodes x_i limited to a neighborhood of the evaluation point $\bar{x}_j$. We discuss such local difference approximations first, and subsequently examine some recent difference approximations based on global functional approximations.

Methods Based on Local Functional Approximations

The first observation one should make is that local functional approximations used at neighboring evaluation points are in general incompatible. This can be simply seen by considering second order Lagrange polynomial interpolations in one dimensions for $\bar{x}_i = x_i$. Using symmetrically placed points (leading to central difference formulas), the local functional approximation at $\bar{x}_i$ is a parabola through the points u^*_{i-1}, u^*_i, and u^*_{i+1}, while that at $\bar{x}_{i+1}$ is a parabola through u^*_i, u^*_{i+1}, and u^*_{i+2}. These two approximations describe two different curves in their region of overlap between x_i and x_{i+1}. Once the approximate solution for the u^*_j is obtained, it is not clear which of the curves to use in order to interpolate for the values of u^* between nodes

(presumably a weighted average would give the best results). Actually, the question is rather academic, since the difference in the values given by the two approximations should be no greater than the errors in the approximations u_j^*.

By contrast, the piecewise functional approximations of the previous section, although localized in nature, are disjoint functions that butt together with no regions of overlap. They therefore give unambiguous values for any quantity (except derivatives at interelement boundaries of an order higher than that demanded by the smoothness of the approximation). Yet it is this very ambiguity in the local functional approximation which gives a local finite difference approximation its flexibility and power. If G(u) is quasi-linear, the local value of u determines the nature of the operator, which in tern determines the optimum type of difference approximation. Thus the nature of the local approximation can be determined at each evaluation point by the local solution. This is the basis for upwind differencing and the type differencing of transonic flows. Even at the same evaluation point, different terms in the operator G(u) can be approximated separately. The nature of the approximation can be made to change during an iterative solution, or a marching solution with respect to another independent variable. ADI methods and splitting or factorization techniques are applications of this degree of flexibility.

The local functional approximation also has to be modified for evaluation points $\bar{x}_j$, near or at global boundaries, in order to satisfy boundary conditions. This can be done most readily if the global boundary is a coordinate surface. For a more general boundary which does not conform to the coordinate system, the approximation can become quite involved, if one wants to maintain the same level of accuracy. For this reason, a nonconforming boundary should be avoided if possible.

Lagrange Representation

The simplest types of finite difference formulas for derivative operators are based on Lagrange polynomial interpolation. This is the basis for the standard forward, backward and central difference formulas for partial derivatives of any order, and of any order of accuracy. Lagrange interpolation can also be used to express u_i^* in terms of neighboring $u_j^*(j \neq i)$, assuming that u_i^* is unknown. Such a device is used in some numerical algorithms.

In solutions involving time-like coordinates, final values of derivatives are already known at points previously computed. In boundary value problems, one needs to compute the same derivative at all nodal points. This suggests the use of Hermite interpolation to provide more accurate difference formulas without increasing storage requirements.

Hermite Representation

An example of a Hermite finite difference formula in one dimension is derived from the specification of u_{i-1}^*, $u_{xx(i-1)}^*$, u_i^*, u_{i+1}^* and $u_{xx(i+1)}^*$, which define a unique quartic polynomial. (This is an example of a Hermite representation with nonconsecutive derivatives.) Evaluating the second derivative of $u^*(x)$ at $\bar{x}_i = x_i$ (for equal spatial intervals h), one obtains

$$h^2(u_{xx(i-1)}^* + 10\, u_{xxi}^* + u_{xx(i+1)}^*) = 12(u_{i-1}^* + 2u_i^* + u_{i+1}^*) \ , \tag{75}$$

which is the standard Hermite centered finite difference formula (ref. 10). The solution for u_{xxi}^* is obtained by tridiagonal inversion. Other Hermite difference formulas involving any partial derivatives can be similarly obtained.

An important application of Hermite interpolation is the construction of difference formulas for initial value problems of the form

$$G(u) = u_x - f(x,u) = 0 \ , \tag{76}$$

where x is one-dimensional. If the solution is known up to the point x_i, the values of u_j^* and u_{xj}^* for all $j \leq i$ are available to construct a variety of

local Hermite interpolates from which one can explicitly predict u^*_{i+1}. A more accurate but implicit difference formula is obtained by including $u^*_{x(i+1)}$ in the representation. Such a formula is normally used as a corrector in an iterative solution, where a predictor formula and (76) were first used to calculate a first approximation for $u^*_{x(i+1)}$.

Another approach to the numerical solution of initial value problems employs higher derivatives u_{xx}, u_{xxx}, etc., which can be obtained in terms of lower derivatives by differentiating (76). One can then construct the Taylor series

$$u^*(x) = u^*_i + u^*_{xi}(x - x_i) + \frac{1}{2} u^*_{xxi}(x - x_i)^2 + \ldots \qquad (77)$$

If the series is truncated after a finite number of terms, the result can be looked at as a local Hermite interpolate through the single point x_i. Thus any step in a finite difference algorithm for the solution of an initial value problem can be obtained from a local Hermite interpolation (although the local functional approximation corresponding to a given algorithm is not necessarily unique).

Different representations can be used in obtaining difference formulas for initial boundary value problems. Let t be the time-like variable, and assume that by differentiating (58) one can express u_t, u_{tt}, etc., as functions of u_x, u_{xx}, etc., where x represents the remaining independent variables. If one first discretizes x space, and defines Lagrange parameters $u^*_i(t)$ at the nodes x_i, one can then use a local functional approximation and Lagrange interpolation to evaluate u_x, u_{xx}, etc., and obtain expressions for du^*_i/dt, $d^2u^*_i/dt^2$, etc. The latter can then be used to define a Hermite discretization of the t coordinate, and the solution can be advanced in t, using (77) (with x replaced by t), which represents Hermite interpolation through a single point in t space.

In summary, any standard finite difference algorithm for solving a set of partial differential operator equations can be derived by applying sequences of

local functional approximations, and interpolating parameters of Lagrange or Hermite representations. The number of points and the order of interpolation determine the accuracy of the approximation (i.e., truncation errors). This still leaves freedom in the choice of points and parameters, and nature of the functional approximations. These can all be optimized to provide the best stability properties for the numerical solution.

Methods Based on Global Functional Approximations

We turn now to finite difference methods based on global functional approximations, limiting ourselves to Lagrange discretization at nodes x, and the case $\bar{x}_i = x_i$. Thus the nodal parameters are $u_i^*(t)$, where t represents the remaining undiscretized variables. Partial derivatives are special examples of linear operators obeying the property

$$L(au + bv) = aL(u) + bL(v) \ , \tag{78}$$

where u and v are two arbitrary functions, and a and b are constants. Thus a local operator G(u) can be written generally as

$$G(u) = g[x,t,u,\ L_t u,\ L_x u,\ L_x(L_t u)] \ , \tag{79}$$

where the subscripts indicate the variables on which the linear operator L operates, and g is an arbitrary function of the six arguments. For any set of linearly independent basis functions $\phi_i(x)$ the linear representation (28) can be expressed in terms of canonical basis functions $\tilde{\phi}_i(x)$ and the nodal parameters $u_i^*(t)$ as

$$u^*(x,t) \approx \sum_{i=1}^{N} u_i^*(t)\ \tilde{\phi}_i(x). \tag{33}$$

The two basis functions are related by defining matrix elements a_{ij} as

$$a_{ij} = \phi_i(x_j) \ , \tag{80}$$

and the inverse matrix with elements b_{ij} satisfying

$$b_{ij}a_{jk} = \delta_{ik} \ . \tag{81}$$

Then

$$\tilde{\phi}_i(x) = b_{ij}\phi_j(x) \ . \tag{82}$$

If we define

$$\chi_i(x) = L_x[\phi_i(x)] \ , \tag{83}$$

it follows from (33), (82), and (83) that

$$L_x[u^*(x,t)] \approx \sum_{i=1}^{N} u_i^*(t)\tilde{\chi}_i(x) \ , \tag{84}$$

where

$$\tilde{\chi}_i(x) = b_{ij}\chi_j(x) = L_x[\tilde{\phi}_i(x)] \ . \tag{85}$$

For linear operators operating on t we obtain

$$L_t[u^*(x,t)] = \sum_{i=1}^{N} L_t[u_i^*(t)]\tilde{\phi}_i(x) \ . \tag{86}$$

Evaluating (79) at the evaluation points $\bar{x}_j = x_j$, we obtain the following sets of equations for the parameters $u_i^*(t)$:

$$g[x_j,t,u_j^*(t),\ L_t[u_j^*(t)],\ \sum_{i=1}^{N} u_i^*(t)\tilde{\chi}_i(x_j),\ \sum_{i=1}^{N} L_t[u_i^*(t)]\tilde{\chi}_i(x_j)] = 0 \ . \tag{87}$$

For an arbitrary set of $\phi_i(x)$, the matrices a_{ij} are dense, and their inversion is inefficient. The practical use of (87) requires restrictions on the functions $\phi_i(x)$. Three such choices will be described, each leading to a practical finite difference method in x space. An arbitrary, independent method can be used in each case to perform the numerical solution in the t space.

Finite Fourier Series

If x is one-dimensional, with periodic boundary conditions, a convenient choice is

$$\phi_k(x) = e^{2\pi ikx/L} \ , \tag{88}$$

where L is the length of the region, and $i = \sqrt{-1}$ The representation is the

finite Fourier series

$$u^*(x,t) = \sum_{k=-K}^{K} c_k(t) e^{2\pi ikx/L} , \tag{89}$$

where $N = 2K+1$. If x_j are chosen to be equally spaced, the transformation between $u_j^*(t)$ and the $c_k(t)$ (corresponding to matrix multiplication by a_{ij} and b_{ij}) is accomplished efficiently by using fast Fourier transforms (ref. 11). For linear differential operators, the functions $\chi_i(x)$ as defined by (83) are just proportional to the $\phi_i(x)$, leading to further simplifications. Finite difference methods using (89) are referred to as pseudospectral (ref. 12) or "accurate space derivative" (ref. 13) methods. They can be extended to higher dimensions using tensor products.

A Fourier series can be reformulated as an expansion in Chebycheff polynomials, based on the identity

$$\cos nx = \sum_{k=0}^{n} a_k \cos^k x .$$

The nodes x_i are no longer equally spaced in the new x domain, but are located at the roots of the Nth order Chebycheff polynomial. Thus the basis for the accuracy of such a difference scheme is the same one that underlies Gaussian quadrature.

Differential Quadrature

The ideas behind the polynomial formulation of a Fourier method can be generalized to any set of orthogonal polynomials, with the nodes x_i again chosen at the roots of the Nth order polynomial. The matrix elements $\tilde{\chi}_i(x_j)$ are easily calculated, using the properties of the orthogonal polynomials. The method, known as differential quadrature, is described in reference 14.

Spline Differencing

A third approach using global functional approximations is to use piecewise approximations, with nodes and evaluation points located on interelement

boundaries. In one dimension, a spline approximation is necessary, with smoothness constraints determined by the highest derivative present in the operator L_x. As indicated previously, cardinal basis functions are not practical, and the original basis functions are employed, with the derivatives at the interelement boundaries as additional unknowns. These derivatives are related to the nodal parameters through banded matrix relations, rather than explicitly as in (84). For a cubic polynomial spline, the relation for first derivatives is identical to the one given by Hermite differencing. The second derivative relation differs from the Hermite formula, with (75) replaced by

$$h^2(2u^*_{xx(i-1)} + 8u^*_{xxi} + 2u^*_{xx(i+1)}) = 12(u^*_{i-1} + 2u^*_i + u^*_{i+1}) \ . \tag{90}$$

Sequences of one-dimensional differencing using ADI methods and splitting, are used in multidimensional problems. Further details on the use of spline differencing in the numerical solutions of partial differential equations are found in reference 15.

Finite Volume Differencing

Finite difference equations based on volume discretizations are often employed when the operator equation (58) can be written in divergence (or conservative) form

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}[F(u)] = 0 \ . \tag{91}$$

Here F(u) is a locator operator on u. Let x be discretized into N volume elements V^i, each of which is enclosed by a set of boundaries S^{ij}. If (91) is integrated over element V^i, and the divergence theorem (4) is applied, the result can be written as

$$\frac{\partial \bar{u}^{*i}}{\partial t} + \frac{1}{V^i} \sum_j S^{ij} \bar{F}^{ij} = 0 \ , \tag{92}$$

where $\bar{u}^{*i}$ is defined by (25), and $\bar{F}^{ij}$ is defined as

$$\overline{F}^{ij} \approx \frac{1}{S^{ij}} \int_{S^{ij}} F(u^*)\, n\, dx \ . \tag{93}$$

The unknown parameters are the $\overline{u}^{*i}$, and a functional approximation is required to express the average normal fluxes $\overline{F}^{ij}$ in terms of these parameters. This can be most easily demonstrated for one-dimensional differencing. Letting the subscript k refer to a global numbering of interelement boundaries (which are nodes in one dimension), a local functional approximation for element V^i can be written as

$$u^{*i}(x) \approx \sum_{k(i)} u_k^* \tilde{\phi}_k^i(x) \ . \tag{94}$$

The notation k(i) indicates a particular choice of nodes k in the neighborhood of element V^i, and $\tilde{\phi}_k^i(x)$ is a local canonical basis function. The latter is <u>not</u> to be confused with the element cardinal basis functions defined by (43) and (44). The nodes k(i) need not be contained in V^i, and $\tilde{\phi}_k^i(x) \neq 0$ for those nodes. Integrating (94) over element V^i, one obtains

$$\overline{u}^{*i} = \sum_{k(i)} u_k^* \overline{\tilde{\phi}}_k^i \ , \tag{95}$$

where

$$\overline{\tilde{\phi}}_k^i \approx \frac{1}{V^i} \int_{V^i} \tilde{\phi}_k^i(x)\, dx \ . \tag{96}$$

The u_k^* can then be expressed in terms of the $\overline{u}^{*i}$ by inverting a sparse matrix in a manner similar to that which exists for Hermite differencing.

The determination of the $\overline{F}^{ij}$, when F(u) involves differential operators, again creates ambiguities resulting from the incompatibility of local functional approximations at neighboring elements V^i. Once the u_k^* are obtained by inverting (95), the local representation $u^{*i}(x)$ can be obtained from (94). One can then determine $F(u^{*i})$, and use (93) to calculate the terms $\overline{F}^{ij}$ for the two boundaries along the x direction. If this procedure is

followed, the value of $\overline{F}^{ij}$ for a given boundary separating two elements will be independently calculated for each of the elements. Yet global conservation, obtained by summing (92) over all the elements, requires that the two $\overline{F}^{ij}$ be equal in magnitude and opposite in sign. One must therefore choose a single, unambiguous $\overline{F}^{ij}$ for each boundary, using some averaging or biasing. If t is a time-like variable, the bias can be alternated with each marching step. Note that at global boundaries exact prescribed values of $\overline{F}^{ij}$ can be imposed.

If x is multidimensional, the above one-dimensional differencing can be used sequentially along several coordinates, using splitting techniques. A particular advantage of finite volume differencing is that the original equation (58) has been integrated, so that the operator F(u) involves lower order differential operators than G(u). Therefore a cruder local approximation (94) can be employed. The possibility of alternating the bias when t is time-like, allows even still cruder approximation for each marching step (ref. 16). The conservative, or integral nature of the numerical solution also guarantees that jump conditions across discontinuities are automatically satisfied, even if the discontinuities are smeared out by the calculation.

Methods Based on a Variational Formulation

We conclude this section by describing briefly a finite difference approach based on the variational formulation (61) and (65). The starting point is the same as for the direct formulation. One first chooses a set of nodes x_j and evaluation points $\overline{x}_j$ inside the domain V and on the boundaries S^i, the type of nodal representation, and the nature of the functional approximation. These are then used to evaluate the operators $F(u^*)$ and $H^i(u^*)$ at the evaluation points $\overline{x}_j$, as functions of the nodal parameters. The next step is to approximate the integral functional I(u) in terms of the discretized F(u) and $H^i(u)$. This is done by appropriate quadrature formulas of the form

$$I(u^*) \approx \sum_j w_j \, [F(u^*)]_j + \sum_i \sum_k w_k^i \, [H^i(u^*)]_k . \tag{97}$$

Here w_j and w_k^i are weight coefficients defined implicitly by some functional approximations of the respective integrands. (These functional approximations can in general be independent of those used in obtaining $[F(u^*)]_j$ and $[H^i(u^*)]_j$ in terms of the nodal parameters.) The summations in j and k are over the evaluation points contained in the domain V and boundary S^i, respectively. In many cases one simply chooses $w_j = w_k^i = 1$.

With $I(u^*)$ expressed as a function of the nodal parameters u_j^* through (97) (assuming Lagrange representation for the moment), the variational principle (61) simply becomes

$$\frac{\partial I}{\partial u_j^*} = 0 , \qquad j = 1 \text{ to } N , \tag{98}$$

providing N equations for the N unknown parameters. This method is sometimes called the Euler method, and is described further in reference 17. Actually, the method bears a striking resemblence to methods based on functional approximations, being somewhat hybrid in nature, with one foot in each camp. It is therefore a good point to leave finite difference methods, and turn our attention to functional approximation methods.

FUNCTIONAL APPROXIMATION METHODS

Any approximate method of solving a continuous field problem whose starting point is the discretization of the dependent variables will be termed a functional approximation method. We will describe such methods in terms of the general functional representation (26), applying the approximation first to a variational formulation, and subsequently to the direct formulation. The results will then be specialized to linear and piecewise representations, the latter giving what we normally called finite element methods.

Variational Formulation

The functional approximation (26) lends itself naturally to the variational formulation (61) and (65). The method will be first described for the case when the dependence on all the variables will be approximated so that the functions c_i in (26) become constant, and there is no variable t. This case is usually called the Ritz or Rayleigh-Ritz method. The function $g(x;c_j)$ must first be chosen so as to satisfy the principal boundary conditions. Substitution of (26) into (65) yields

$$I(c_j) \approx \int_V F[g(x;c_j)]dx + \sum_i \int_{S^i} H^i[g(x;c_j)]dx \ . \tag{99}$$

This restricts $g(x;c_j)$ further to functions with sufficient continuity for the integrals to exist. The variational principle (61), applied to all variations δc_j, gives the set of equations

$$\frac{\partial I}{\partial c_j} \approx 0 \qquad j = 1 \text{ to } N \tag{100}$$

for the N parameters c_j.

The method can be extended to functional approximations (26), where c_j are now functions of undiscretized variables t. It is then referred to as the Kantorovich method. The integral functional (65) must now be written as

$$I(u) = \int_T \int_{V(t)} F(u)dxdt + \sum_i \int_{S_t^i} \int_{S^i(t)} H^i(u)dxdt \ , \tag{101}$$

where T and S_t^i refer to the subdomain of variables t, and their boundaries. Substitution of (26) into (101) yields

$$I[c_j(t)] \approx \int_T \left[\int_{V(t)} F\{g[x;c_j(t)]\}dx \right]dt + \sum_i \int_{S_t^i} \left[\int_{S^i(t)} H^i\{g[x;c_j(t)]\}dx \right]dt . \tag{102}$$

Equation (102) is now considered an integral functional over t space involving the unknown functions $c_j(t)$. The variational principle (61) then states that the Fréchet differential

$$\delta I(c_j, \delta c_j) = 0 \ , \tag{103}$$

which results in the set of equations and boundary conditions necessary to determine the set of unknown functions $c_j(t)$.

As indicated before, the operator F(u) involves lower order differential operators than G(u), permitting a functional approximation with lower order of smoothness. The approximation also need not satisfy the natural boundary conditions, since they are automatically satisfied in the variational process (to the same degree that the equation G(u) = 0 is satisfied). For these reasons a Ritz or Kantorovich method is much to be preferred. Unfortunately, it is limited to boundary value problems in which G(u) is self-adjoint. There have therefore been many attempts to create so-called "variational" principles designed to solve problems for which a true variational principle does not exist. These new principles may be classed as adjoint variational, quasi-variational, or restricted variational. Finlayson and Scriven (ref. 18) have shown that they are all either based on a direct formulation in disguise, or offer no real advantage over a method based on a direct formulation. There is therefore no further need to consider any of these formulations.

A new method which makes use of a variational formulation in an iterative procedure is the pseudo-functional method of Norrie and deVries (ref. 19). It is designed for problems which come close to admitting a variational principle. More precisely, assume that (58) is given by

$$G(u) = F_u(u) + G_o(u) = 0 \ , \tag{104}$$

and the boundary conditions (59) that are not principal conditions can be written as

$$B^i(u) = \overline{F}_j(u) + B^i_j(u) = 0 \ , \quad x\varepsilon S^i, \ j = 0,1,\text{etc.}, \tag{105}$$

where the operators $\overline{F}_j(u)$ are related to $F(u)$ as in (20) and (22). If the terms $G_o(u)$ and $B^i_j(u)$ are sufficiently small, then (104) and (105) can be solved by iteration. Let u^{m-1} represent the solution after m-1 iterations. Then u^m is defined as the solution of

$$F_u(u^{*m}) + G_o(u^{*(m-1)}) \eqsim 0 \ , \tag{106}$$

subject to the natural boundary conditions

$$\overline{F}_j(u^{*m}) + B^i_j(u^{*(m-1)}) \eqsim 0, \quad x\varepsilon S^i, \ j = 0,1,\text{etc.}, \tag{107}$$

Equations (106) and (107) are thus seen to follow from the application of the variational principal (61) to the functional

$$I(u^{*m}) \eqsim \int_V [F(u^{*m}) + G(u^{*(m-1)})u^{*m}]dx + \sum_i \int_{S^i} H^i(u^{*(m-1)})u^{*m}dx \ . \tag{108}$$

An iterative Ritz procedure can be applied to (108), until a converged solution for the c_j is obtained.

The Method of Weighted Residuals

If a variational formulation does not exist, even approximately, then a functional approximation method must be based on a direct formulation. To accomplish this, (58) and (59) must be converted into functionals. To see how this can be done, let us rewrite the Ritz procedure applied to a variational formulation, in terms of the equivalent direct formulation. If we substitute (26) and (66), and apply the variational principle (61) to all variations δc_j, we obtain

$$\int_V \frac{\partial g}{\partial c_j} G(u^*)dx + \sum_i \int_{S^i} [\frac{\partial g}{\partial c_j} B^i_o(u^*) + \frac{\partial g_n}{\partial c_j} B^i_1(u^*) + \frac{\partial g_{nn}}{\partial c_j} B^i_2(u^*) + \ldots]dx \eqsim 0 \ , \tag{109}$$

where we used (60) and let $B_o^i = \overline{F}_o + H_o^i = 0$, $B_1^i = \overline{F}_1 + H_1^i = 0$, etc., represent the extended natural boundary conditions in (59). (The principal boundary conditions give $\partial g/\partial c_j = 0$, $\partial g_n/\partial c_j = 0$, etc., on S^i, and are assumed satisfied by the choice of $g(x;c_j)$. Thus contributions to the boundary integral terms in (109) will only come from the natural boundary conditions.) By using integration by parts and the divergence theorem (4), one can show the equivalence of the sets of equations (109) and (100). But (109) could have been obtained from the direct formulation by integrating (58) and the natural boundary conditions (59) over their respective domains, after first multiplying by appropriate weighting functions. Particular linear combinations of these integrals then yield (109). Note that the weighting function for $B_o^i(u)$ is the same as that for $G(u)$, but those for $B_1^i(u)$, $B_2^i(u)$, etc., (if they are present) are different.

The above considerations suggest that the direct formulation (58) and (59) be recast in the equivalent weak form

$$\int_V \psi(x)\, G(u)dx = 0 \,, \tag{110}$$

and

$$\int_{S^i} \psi(x)\, B^i(u)dx = 0, \tag{111}$$

where (110) and (111) are assumed valid for all arbitrary functions $\psi(x)$. A functional approximation method can be obtained by choosing a finite set of linearly independent weighting functions $\psi_j(x)$ to approximate $\psi(x)$, and substituting (26) and each $\psi_j(x)$ in turn into (110). If $G(u^*)$ is termed the residual, the resulting set of equations is thus obtained by equating to zero the integrals of weighted residuals over the domain. The method is therefore often referred to as the method of weighted residuals. If all the boundary conditions are not satisfied by the choice of (26), additional boundary residual equations

(111) must be calculated. These normally use the same weighting functions as in (110), although (109) shows that different weighting functions may be appropriate for some $B^i(u)$.

By analogy with the variational case, integration by parts can be used to obtain integrals involving lower order differential operators. It is also possible to combine equation residuals and boundary residuals in the same equation, as was done in (109). To indicate these procedures, consider a term in G(u) that can be written as a divergence $\partial F/\partial x$. Then the integral for that term can be written as

$$\int_V \psi_j \frac{\partial F}{\partial x}\, dx = \int_S \psi_j n F dx - \int_V \psi_{jx} F dx \ . \tag{112}$$

If one of the terms in $B^i(u)$ is $nF(u)$, it is then clear how (110) and (111) can be combined to eliminate that term. Note that (112) imposes smoothness conditions on $\psi_j(x)$. We will henceforth examine the method of weighted residuals based on (110) with the understanding that these can be transformed by integration by parts and combined with (111) to eliminate certain boundary residual terms. When this is not possible, boundary integrals (111) would be treated in the same manner as (110).

Let us generalize (110) by introducing the undiscretized variables t, and considering integrations over the domain and boundary of the discretized variables x. Thus, given (58), (26) and a set of weighting functions $\psi_j(x,t)$, the method of weighted residuals gives the equations

$$\int_V \psi_j(x,t)\ G[g(x;c_i(t)]dx \approx 0 \qquad j = 1 \text{ to } N \tag{113}$$

for the unknown functions $c_j(t)$. There are many possible choices for $\psi_j(x,t)$, each one leading to a different method. They are fully discussed in the book by Finlayson (ref. 20). The various classifications are briefly summarized below.

Method of Moments

If $\psi_j(x,t)$ form an arbitrary, linearly independent set of functions, we have the general method of moments. Normally, it is restricted to functions of x only, and are typically members of a complete set of functions. A popular choice is polynomials in x.

Galerkin Method

In the Galerkin method, the weighting function is chosen to give the same equations as those provided by a variational formula. It follows from (109) that we must have

$$\psi_j(x,t) = \frac{\partial g}{\partial c_j}\,[(x;c_i(t)]\ , \tag{114}$$

where g is considered a function of x and c_i in performing the partial derivative, i.e., t is considered a fixed parameter. This is probably the most popular method, particularly in finite element applications.

Least Squares Method

In this method we set

$$\psi_j(x,t) = \frac{\partial G}{\partial c_j}\,[g(x;c_i(t)]\ , \tag{115}$$

where again G(g) is considered a function of x and c_i. The name of the method becomes obvious on substituting (115) into (113) and interchanging integration and differentiation, to obtain

$$\frac{\partial}{\partial c_j}\int G^2[g(x;c_i(t))]dx \eqsim 0\ . \tag{116}$$

While (116) minimizes the integrated square of the residual, a more logical procedure would be to determine the maximum value of G^2 in the domain for a given choice of c_i, and to minimize this maximum among all choices of c_i. While this has been used historically, it is difficult to apply in practice, and has been superseded by (116). One disadvantage of the least squares method

is that the order of the differential operators cannot be lowered through integration by parts.

Collocation Method

If we admit discontinuous functions for ψ_j, several new methods are available. Let $\bar{x}_j$ (j = 1 to N) be a set of n arbitrary points in the x domain, called collocation points. Then if

$$\psi_j(x) = \delta(x - \bar{x}_j) \ , \tag{117}$$

where δ represents the Dirac delta function, substitution of (108) in (103) yields

$$G[g(\bar{x}_j; c_i(t)] \approx 0 \ , \tag{118}$$

i.e., the residual is set equal to zero at the collocation points; hence, the name collocation method. Note that integration by parts is not possible.

Subdomain Method

If one divides the domain V into arbitrary subdomains $\bar{V}^j$, one can define a less violent alternative to the Dirac delta function; namely, the characteristic function

$$\begin{aligned} \psi_j(x) &= 1 \quad \text{if} \quad x \in \bar{V}^j \\ &= 0 \quad \text{if} \quad x \notin \bar{V}^j \ . \end{aligned} \tag{119}$$

Equation (113) now becomes

$$\int_{\bar{V}^j} G[g(x; c_i(t)]dx \approx 0 \ . \tag{120}$$

Thus the integrated residual is set equal to zero in each subdomain $\bar{V}^j$; hence, the name subdomain method. Note that terms in G(u) that can be written as a divergence can be converted via the divergence theorem (4) to integrals over the boundaries of $\bar{V}^j$ involving lower order operators. This method is sometimes called the method of integral relations.

Least Squares Collocation Method

The methods described above can often be combined. An example is the least squares-collocation method. We start with the least squares method (106), and approximate the integral by appropriate quadrature formulas over M arbitrary collocation points $\bar{x}_i$, where $M \geq N$. The resulting equations for the parameters c_k are

$$\frac{\partial}{\partial c_j} \sum_{i=1}^{M} w_i G^2[g(\bar{x}_i;c_k(t))] = 2 \sum_{i=1}^{M} w_i G[g(\bar{x}_i;c_k(t))] \frac{\partial}{\partial c_j} G[g(\bar{x}_i;c_k(t))] \approx 0 . \tag{121}$$

In practice, one often chooses $w_i = 1$. As M approaches infinity, the method approaches the least squares method. On the other hand, if M = N and $\partial G[g(\bar{x}_i;c_k(t)]/\partial c_j$ is non-singular, then (111) is reduced to the collocation method (108) (assuming that all w_i are non-zero).

The equivalence of the N-point quadrature approximation to the least squares method and the collocation method can be generalized to any residual method involving continuous weighting functions. Omitting the dependence on t, we can write the N-point quadrature approximation to (113) as

$$\int_V \psi_j(x) G[g(x;c_k)]dx \approx \sum_{i=1}^{N} w_i \psi_j(\bar{x}_i) G[g(\bar{x}_i;c_k)] \approx 0 . \tag{122}$$

This reduces to the collocation method if $\psi_j(\bar{x}_i)$ is non-singular and w_i are non-zero. Thus, if the integrals in a residual method are too complex to evaluate analytically, and no integration by parts is employed, an N-point quadrature approximation is identical to a collocation method. By a judicious choice of collocation points $\bar{x}_i$, this method gives results whose accuracy is consistent with the original functional approximation. The choice can be made rational if the functional representation is linear, which is the case we consider next.

Linear Functional Approximation

All the methods described so far have been considered for a general functional approximation (26). In practice, one normally uses the linear representation (28). This simplifies some of the methods. For example, the weighting function for the Galerkin method becomes

$$\psi_j(x) = \phi_j(x) \ , \tag{123}$$

i.e., the weighting functions are the basis functions themselves. If $\phi_j(x)$ are given by (78), i.e., if (28) is a finite Fourier series, then the Galerkin method is called a spectral method (ref. 21).

A linear representation allows the introduction of nodal parameters as unknowns by choosing an arbitrary point discretization x_i. It is then possible to establish a correspondence with finite difference methods. The most obvious one is through the collocation method. If the collocation points are identified as the evaluation points, it is evident that the collocation method is identical to a nodal finite difference method employing a global functional approximation. The method of differential quadrature has its analogue in the collocation method, where it is referred to as orthogonal collocation.

Most conventional finite difference methods employ local functional approximations. Since those functional approximation methods based on discontinuous weighting functions (i.e., collocation or subdomain) yield equations evaluated at disjoint points or subdomains, one can generalize them by permitting local functional approximations. One can then say that all conventional finite difference methods are collocation methods using local functional approximations. Similarly, one can consider finite volume difference methods as subdomain methods (with the divergence theorem applied) using local functional approximations. Finally, the Euler difference method may be

thought of as a variational-collocation method using local functional approximations.

Weighted residual methods using continuous weighting functions, and which do not employ quadratures, can only be formulated in terms of a global functional approximation. Even for discontinuous weighting functions, a global functional approximation may be preferred. For complex domains, the integrals resulting from such a global approximation could not be calculated analytically. Even for one-dimensional or tensor product approximations, global functional approximations would lead to dense matrices. Both of these difficulties can be avoided by using piecewise functional approximations, which will now be discussed.

Finite Element Methods

Any functional approximation method using a piecewise functional representation is termed a finite element method. Thus, the domain of x is divided into M volume elements V^k, called finite elements, and a set of N global nodes x_j, and their associated nodal parameters $u_j^*(t)$. (We assume a Lagrange representation for now.) For each element V^k we have a set of local coordinates x^k, N^k local nodes x_ℓ^k, the associated local nodal parameters $u_\ell^{*k}(t)$, and element cardinal basis functions $\tilde{\Phi}_\ell^k(x^k)$. The latter are called element shape functions. The representation of $u^*(x,t)$ in element V^k is

$$u^{*k}(x^k,t) \approx \sum_{\ell=1}^{N^k} u_\ell^{*k}(t)\tilde{\Phi}_\ell^k(x^k) \quad . \tag{49}$$

The use of (49) in the two types of functional approximation methods will be briefly outlined.

Variational Finite Element Method

We will describe the Ritz method for simplicity. The variational formulation (99) and (100) can be easily reformulated in terms of the finite elements. Let I^k be the contribution to the integral functional (99) from element V^k,

given by

$$I^k(u_\ell^{*k}) \approx \int_{V^k} F[\sum_{\ell=1}^{N^k} u_\ell^{*k} \tilde{\phi}_\ell^k(x^k)] \frac{\partial x}{\partial x^k} dx^k + \sum_i \int_{S^{ki}} H^i[\sum_{\ell=1}^{N^k} u_\ell^{*k} \tilde{\phi}_\ell^k(x^k)] \frac{\partial x}{\partial x^k} dx^k . \tag{124}$$

Here $\partial x/\partial x^k$ represents the transformation Jacobians for elements V^k and the element boundaries S^{ki}. The boundary integrals exist only for elements lying on the global boundary, with contributions coming from those boundaries S^i that border element V^k. From (126) one can then determine $\partial I^k/\partial u_\ell^{*k}$. By means of the mapping between local and global node numbers, this can be rewritten as $\partial I^k/\partial u_j^*$, in terms of global nodal parameters. The variational principal (100) is obtained by summing over all the elements, i.e.,

$$\sum_{k=1}^{M} \frac{\partial I^k}{\partial u_j^*} \approx 0 , \qquad j = 1 \text{ to } N . \tag{125}$$

Note that contributions to (125) come only from elements containing global node x_j, and the resulting equation involves only the nodal parameters contained in those elements. This insures sparse matrices in the solution of the algebraic system (125).

Residual Finite Element Methods

The method of moments does not provide a useful finite element method, since the weighting function is not localized. The most popular method is the Galerkin method. Omitting the dependence on t for the moment, if x_ℓ^k is the local node in element V^k corresponding to global node x_j, it follows from (123) and (113) that the contribution to (113) from element V^k is

$$I_\ell^k(u_m^{*k}) \approx \int_{V^k} \tilde{\phi}_\ell^k(x^k)\, G\, [\sum_{m=1}^{N^k} u_m^{*k} \tilde{\phi}_m^k(x^k)] \frac{\partial x}{\partial x^k} dx^k . \tag{126}$$

If (126) is renumbered with a global node numbering, and written as I_j^k in terms of global nodal parameters, then (113) becomes

$$\sum_{k=1}^{M} I_j^k \simeq 0, \qquad j = 1 \text{ to } N , \tag{127}$$

where contributions to (127) again come only from elements containing node x_j. If part of G(u) had been integrated as in (112) to create boundary integrals, then additional boundary terms would be needed in (126) for nodes x_j on the global boundary.

The least squares finite element method is formulated in a manner similar to the variational method, based on (116). The collocation finite element method follows directly by substitution into (118). If some of the collocation points $\bar{x}_j$ lie on element boundaries, then Hermite or spline representations are required to provide sufficient smoothness to calculate the operator G. Lower order representations are sufficient if all the collocation points are in the interior of elements.

An important advantage of finite element methods is that prescribed boundary conditions on global boundaries are simply satisfied by setting the appropriate nodal parameters equal to their boundary values. Equations (124) or (126) would not be calculated for those nodes. Derivative boundary conditions can be satisfied by using Hermite shape functions. The number of unknown nodal parameters can be further reduced when some of the elements contain interior nodes. If x_j is an interior node located inside element V^k, then (125) (or (127)) is the only equation involving u_j^*. The set of equations for all the interior nodes in V^k can be solved for the interior nodal parameters in terms of the nodal parameters on the element boundary. By this process, called condensation, the final set of equations contains only nodal parameters associated with nodes on interelement boundaries.

CONCLUSION

Finite difference methods have been discussed from a rather unorthodox viewpoint in order to bring out their relationship to functional approximation

methods. Let us now examine this relationship by first comparing the nodal finite difference method with the finite element method. Both methods rely on a discretization of the x domain into nodes and the introduction of associated nodal parameters to represent the unknown function u(x). It is in the manner in which one obtains equations to solve for these parameters that the two methods diverge.

The finite element method requires two additional discretizations. One is the discretization inherent in the global functional approximation which permits an unambiguous evaluation of u, or any operator on u, at an arbitrary point x. The other discretization, which is peculiar to the finite element method, is the additional discretization of the x domain into volume elements that define a piecewise functional approximation. These three discretizations are not independent, but are interrelated to provide desired smoothness to the approximation with the minimum of complexity. It is the achievement of these two contradictory goals that is the hallmark of the art of the finite element method. Finally, a variational or weighted residual method must be chosen to define appropriate integral functionals. The latter choice also involves some ingenuity, since integration by parts for continuous weighting functions, or proper choice of collocation points, can lessen the smoothness requirements. The choice of method is thus also coupled to the three discretizations.

The conventional nodal finite difference method is essentially a collocation method, with nodes and collocation points aligned along coordinate lines if x is multidimensional. The finite difference approximations to the governing equations can be interpreted as resulting from local functional approximations. The approximation can therefore vary from point to point, and even for individual terms in equations. The art in the finite difference method is to use this great flexibility to obtain efficient and stable solutions.

The power of the finite element method lies in its ability to handle complex boundaries through the freedom in choosing the volume discretizations, and the ease in satisfying boundary conditions. An additional advantage exists for variational and certain weighted residual methods, where we can deal with operators of lower differential order and admit approximations of lower smoothness. Since a single global functional approximation is required, the method appears to be less flexible in dealing with the complex physical phenomena associated with highly nonlinear equations, such as those of fluid dynamics. Some progress has recently been reported in simulating type differencing (ref. 22) and upwind differencing (ref. 25) within the finite element method.

The nodal finite difference method has the flexibility to cope with the phenomena associated with the complexities of the equations. On the other hand, boundary conditions can be satisfied accurately in practice only if the boundaries are coordinate surfaces. Here the recent work of Thompson (ref. 24) is generating coordinate systems for arbitrary surfaces gives promise to free the finite difference method from its major disadvantage. The use of piecewise approximations (a finite element concept!) to represent arbitary surfaces can also play an important role.

For initial boundary value problems, finite volume differencing can be thought of as the subdomain method with local functional approximations. Since it also results in lower order differential operators, it can be said to possess the other advantage attributed to finite element methods. Actually, its ability to treat discontinuities easily gives it somewhat of an advantage.

In conclusion, finite element methods are best designed to handle complex boundaries, while finite difference methods appear to be superior for complex equations. Time and further research will tell if one of the methods will be able to overcome its shortcomings and emerge as clearly superior in solving boundary and initial boundary value problems.

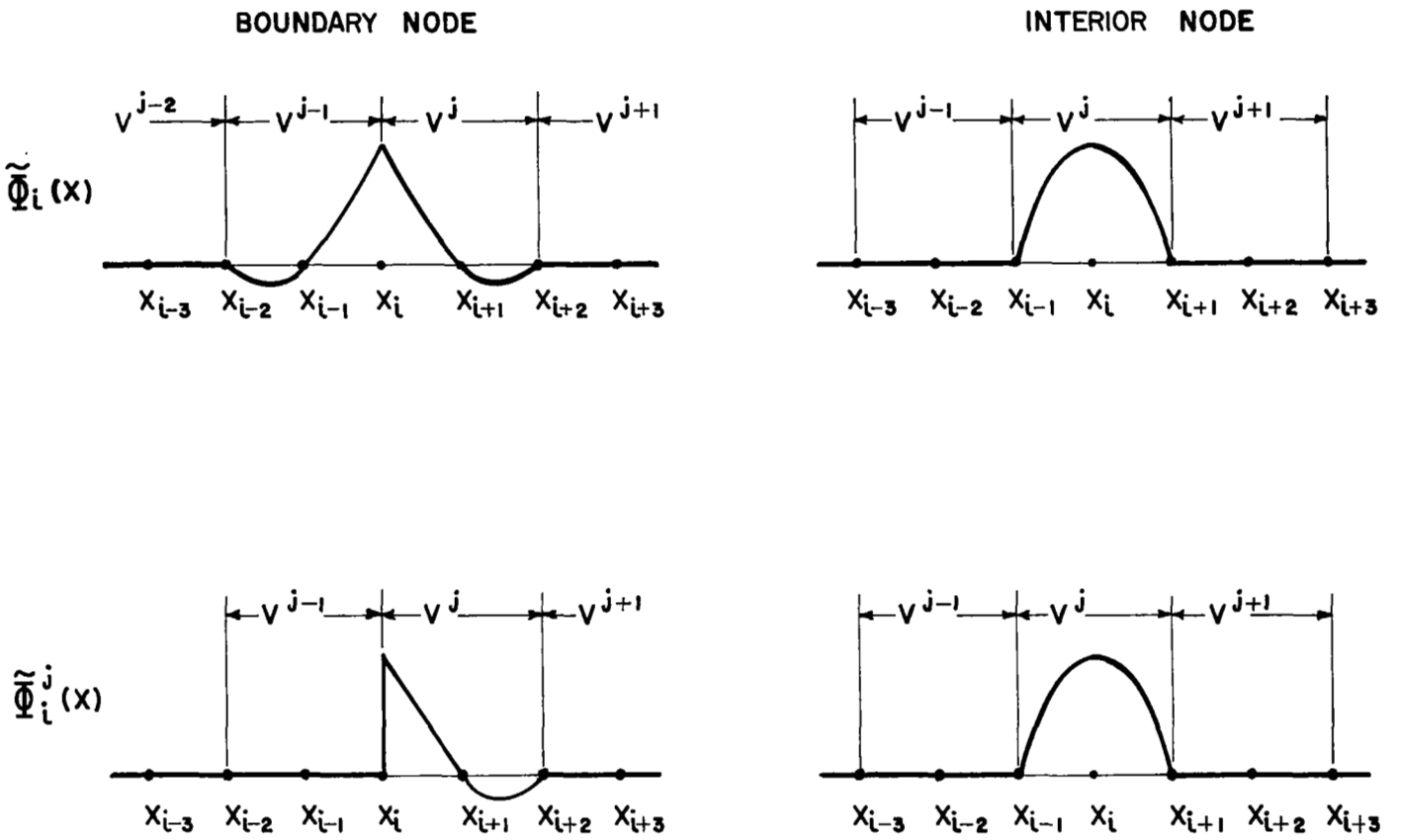

FIGURE 1: ONE-DIMENSIONAL LAGRANGE CARDINAL BASIS FUNCTIONS WITH ONE INTERIOR NODE

REFERENCES

1. Prenter, P. M.: Splines and Variational Methods. John Wiley & Sons, Inc., 1975.

2. Vinokur, M.: On Local Error Bounds for Triangular Finite Element Interpolation. (To be published).

3. Ergatoudis, J.; Irons, B. M.; and Zienkiewicz, O. C.: Curved Isoparametric Quadrilateral Elements for Finite Element Analysis. Int. J. Solids Struct., Vol. 4, No. 1, 1968, pp. 31-42.

4. Gordon, R. G.: New Method for Constructing Wavefunctions for Bound States and Scattering. J. Chem. Phys., Vol. 51, No. 1, July 1969, pp. 14-25.

5. Canosa, J.; and de Oliveira, R. G.: A New Method for the Solution of the Schrodinger Equation. J. Comp. Phys., Vol. 5, No. 2, Apr. 1970, pp. 188-207.

6. Timmer, H. G.: Development of a General Parametric Cubic Geometry Representation Computer Program. Final Report. McDonnell Douglas Report MDC G5969, May 1975.

7. Riesenfeld, R.: Applications of B-Spline Approximation to Geometric Problems of Computer-Aided Design. University of Utah UTEC-CSc-73-126, March 1973.

8. Coons, S. A.: Surfaces for Computer-Aided Design of Space Forms. MIT Project MAC TR-41, June 1967.

9. Gordon, W. J.: Spline-Blended Surface Interpolation Through Curve Networks. J. Math. Mech., Vol. 18, No. 10, Apr. 1969, pp. 931-952.

10. Collatz, L.: The Numerical Treatment of Differential Equations. Third ed., Springer Verlag, 1960.

11. Brigham, E. O.: The Fast Fourier Transform. Prentice-Hall, 1974.

12. Orszag, S. A.: Numerical Simulation of Incompressible Flows within Simple Boundaries. 1. Galerkin (Spectral) Representations. Stud. in Appl. Math.,Vol. 50, No. 4, Dec. 1971, pp. 293-327.

13. Gazdag, J.: Numerical Convective Schemes Based on Accurate Computation of Space Derivatives. J. Comp. Phys., Vol. 13, No. 1, Sep. 1973, pp. 100-113.

14. Bellman, R.; Kashef, B. G.; and Casti, J.: Differential Quadrature: A Technique for the Rapid Solution of Nonlinear Partial Differential Equations. J. Comp. Phys., Vol. 10, No. 1, Aug. 1972, pp. 40-52.

15. Rubin, S. G.; and Graves, R. A., Jr.: Viscous Flow Solutions with a Cubic Spline Approximation. Comp. Fluids, Vol. 3, No. 1, Mar. 1975, pp. 1-37.

16. MacCormack, R. W.: The Effect of Viscosity in Hypervelocity Impact Cratering. AIAA Paper No. 69-354, Apr. 1969.

17. Forray, M. J.: Variational Calculus in Science and Engineering. McGraw-Hill, 1968.

18. Finlayson, B. A.; and Scriven, L. E.; On the Search for Variational Principles. Int. J. Heat Mass Transfer, Vol. 10, No. 6, June 1967, pp. 799-820.

19. Norrie, D. H.; and de Vries G.: Application of the Pseudo-Functional Finite Element Method to Non-Linear Problems. Finite Elements in Fluids, Vol. 2, R. H. Gallagher, J. T. Oden, C. Taylor, and O. C. Zienkiewcz (Ed.), John Wiley & Sons, Inc., 1975, pp. 55-65.

20. Finlayson, B. A.: The Method of Weighted Residuals and Variational Principles. Academic Press, Inc., 1972.

21. Orszag, S.A.: Numerical Simulation of Incompressible Flows within Simple Boundaries: Accuracy. J. Fluid Mech., Vol. 49, pt. 1, Sep. 1971, pp. 75-112.

22. Chan, S. T. K.; Brashears, M. R.; and Young, V. Y. C.: Finite Element Analysis of Transonic Flow by the Method of Weighted Residuals. AIAA Paper No. 75-79, Jan. 1975.

23. Piva, R.; and Di Carlo, A.: Appropriate Interpolation Schemes for F. E. Models of Viscous Incompressible Flows. GAMM Conference on Numerical Methods in Fluid Mechanics, Proz-Wahn, W. Germany, Oct. 1975.

24. Thompson, J. F.; Thames, F. C.; and Mastin, C. W.: Automatic Numerical Generation of Body Fitted Curvilinear Coordinate System for Field Containing any Number of Arbitrary Two-Dimensional Bodies. J. Comp. Phys., Vol. 15, No. 3, July 1974, pp. 299-319.