FINITE ELEMENT METHODS

C.Temperton
European Centre for Medium-Range Weather Forecasts
Shinfield Park, Reading, UK

1. INTRODUCTION

In designing a numerical weather prediction model, one of the most fundamental aspects is the choice of discretization technique in each of the spatial dimensions. In the vertical, by far the most popular choice is the finite difference method; while in the horizontal, both finite-difference and (especially for global models) spectral methods are widely employed. A third possibility, the finite element method, has been used rather less widely, though with considerable success.

The aim of this paper is to present the essentials of the finite element method, and to review some of the applications to numerical weather prediction models.

No attempt is made here to go deeply into the theoretical aspects of the finite element method, nor into the richness of the technique when applied to a wider class of problems; for these, numerous textbooks are available. A good place to start is the early volume by *Strang and Fix* (1973).

2. WHAT IS THE FINITE ELEMENT METHOD?

The essence of the finite element method can be seen by considering various ways of representing a function $f(x)$ on an interval $a \le x \le b$. In the *finite-difference* method the function is defined only on a set of gridpoints; i.e., $f(x_j)$ is defined for a set of $x_j \in [a, b]$, but there is no explicit information about how the function behaves between the gridpoints. In the *spectral* method, on the other hand, the function is defined in terms of a finite set of basis functions:

$$f(x) = \sum_{k=0}^{N} a_k e_k(x) \tag{2.1}$$

where the basis functions $e_k(x)$ are *global* (e.g., Fourier series, or spherical harmonics for two-dimensional functions on the surface of a sphere), and the $\{a_k\}$ are the spectral coefficients. Equation (2.1) defines $f(x)$ *everywhere* on the interval, and the representation is independent of any set of gridpoints.

In the *finite-element* method, the function is again represented in terms of a finite set of basis functions:

$$f(x) = \sum_{k=0}^{N} a_k e_k(x) \tag{2.2}$$

but this time the basis functions $e_k(x)$ are *local*, i.e., they are non-zero only on a small sub-interval. As in the spectral method, the $\{a_k\}$ are the coefficients of the basis functions, and $f(x)$ is defined everywhere; but as in the finite-difference method, there is an underlying mesh of gridpoints (nodes) involved in the representation.

To clarify this idea we consider the simple choice of *linear* finite elements. The interval $[a, b]$ is divided into subintervals by specifying a set of mesh points, say $\{x_0, x_1, ..., x_N\}$. The basis function $e_k(x)$ is defined to be 1 at x_k, decreasing linearly to zero at x_{k-1} and x_{k+1}, and to be zero outside the interval $[x_{k-1}, x_{k+1}]$. Thus the defining equations are:

$$e_k(x) = \frac{x-x_{k-1}}{x_k-x_{k-1}}, \quad x \in [\, x_{k-1}\, ,\, x_k\,] \tag{2.3}$$

$$= \frac{x_{k+1}-x}{x_{k+1}-x_k}, \quad x \in [\, x_k\, ,\, x_{k+1}\,]$$

$$= 0 \text{ otherwise}$$

The situation is illustrated in Fig. 1; note that the mesh may be non-uniform.

Suppose now that $f(x)$ is given at the set of gridpoints $\{x_k\}$, as in Fig. 2; how do we determine the coefficients $\{a_k\}$ of the basis functions, in order to use the representation given by Eq. (2.2)? The answer is almost obvious: since $e_k(x)$ is the only basis function which is non-zero at x_k, we must have

$$a_k = f(x_k),\ 0 \le k \le N.$$

Between the gridpoints, say between x_k and x_{k+1}, just *two* of the basis functions (e_k and e_{k+1}) are non-zero; since both are linear, $f(x)$ as defined by Eq. (2.2) is linear on the subinterval $[x_k, x_{k+1}]$. Thus the behaviour of $f(x)$ between gridpoints is determined simply by linear interpolation.

At this point, it may seem that we have gained very little over the simple gridpoint representation of $f(x)$. The benefits of the representation in terms of linear basis functions will become clear in the next section, when we consider elementary operations with functions.

Finally in this section, we note that finite element basis functions can be even simpler (piecewise *constant* on subintervals) or more complicated (piecewise quadratic, piecewise cubic, and so on).

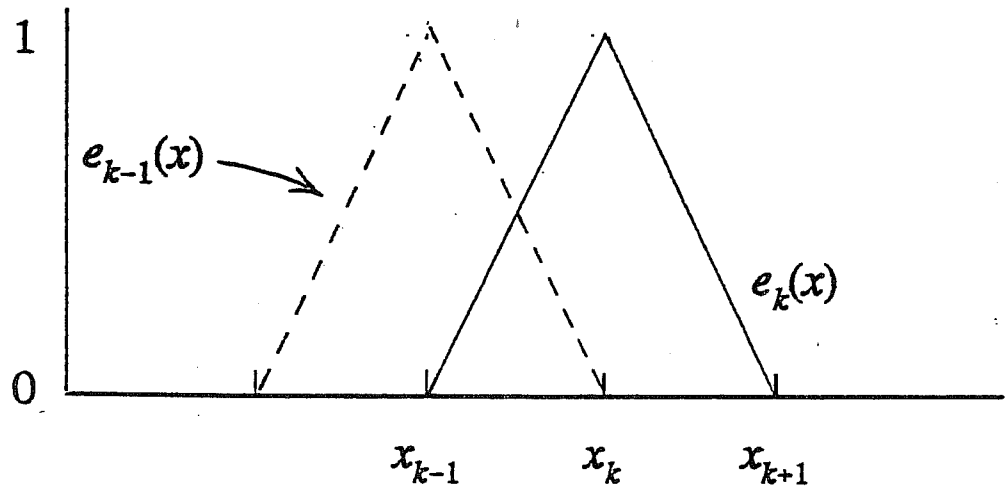

Figure 1. Linear finite element basis functions.

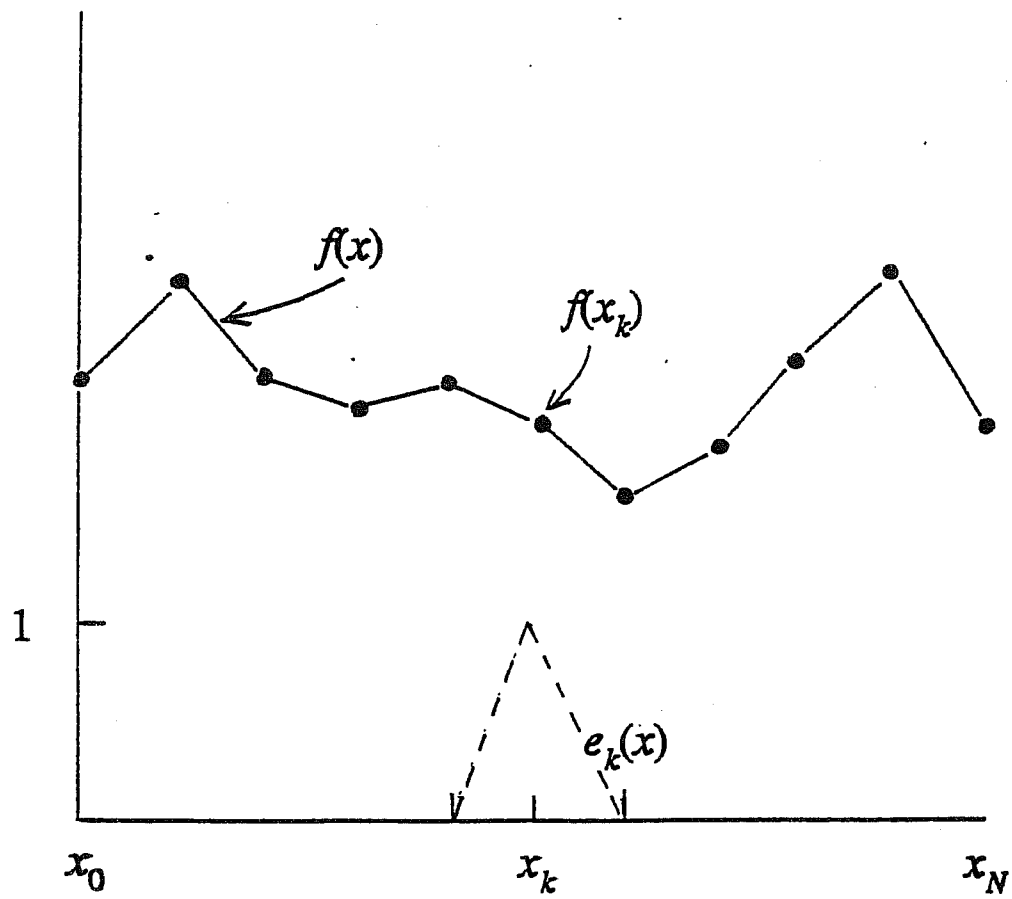

Figure 2. Representation of $f(x)$.

3. SIMPLE OPERATIONS WITH LINEAR FINITE ELEMENTS

In this section we show how to perform three simple operations with linear finite elements in one dimension: differentiation, multiplication, and taking second derivatives.

(a) Differentiation

Suppose we are given $u_i = u(x_i)$ at a set of nodes x_i $(0 \le i \le N)$, and we wish to differentiate u to find $v = u_x$. We start by expanding u and v in terms of the basis functions $e_i(x)$, as shown in Section 2:

$$u(x) = \sum_{i=0}^{N} u_i e_i(x), \quad v(x) = \sum_{i=0}^{N} v_i e_i(x)$$

where the coefficients v_i $(0 \le i \le N)$ are the unknowns of our problem. The series for u can be differentiated term by term, so that $v = u_x$ becomes

$$\sum_{i=0}^{N} v_i e_i(x) = \sum_{i=0}^{N} u_i e_i'(x) \tag{3.1}$$

In Eq. (3.1) we use the notation $e_i'(x)$ to denote the x-derivative of the basis function, to avoid multiple subscripts. From the definition of the (piecewise linear) basis functions, $e_i'(x)$ is piecewise *constant.*

The next step is to apply the *Galerkin* technique, namely to orthogonalize the error in (3.1) to the basis: i.e., set

$$\langle\, v - u_x \,,\, e_k \,\rangle = 0 \quad \text{for all } k \ (0 \le k \le N)$$

where the inner product $\langle\ ,\ \rangle$ is defined by

$$\langle\, f(x) \,,\, g(x) \,\rangle = \int_{x_0}^{x_N} f(x) g(x) dx.$$

It is easily seen that this is equivalent simply to multiplying both sides of (3.1) by $e_k(x)$ and integrating from x_0 to x_N:

$$\int_{x_0}^{x_N} \sum_{i=0}^{N} v_i e_i(x) e_k(x) dx = \int_{x_0}^{x_N} \sum_{i=0}^{N} u_i e_i'(x) e_k(x) dx. \tag{3.2}$$

Since everything is well-behaved we can exchange the order of the integration and the summation.

Moreover, since v_i and u_i are coefficients of basis functions, they are *not* functions of x, and so we can take them outside the integrals. Hence (3.2) becomes:

$$\sum_{i=0}^{N} v_i \int_{x_0}^{x_N} e_i(x)e_k(x)dx = \sum_{i=0}^{N} u_i \int_{x_0}^{x_N} e_i'(x)e_k(x)dx. \qquad (3.3)$$

The integrands remaining in (3.3) depend only on the mesh, not on the functions u and v. It is easily seen that for example $e_i(x)e_k(x)$ is zero everywhere unless $i = k$ or $i = k\pm1$; in fact it is a simple exercise to show that:

$$\int_{x_0}^{x_N} e_i(x)e_k(x)dx = \frac{h_{k-1}}{6} \text{ for } i = k-1$$

$$= \frac{h_{k-1}+h_k}{3} \text{ for } i = k$$

$$= \frac{h_k}{6} \text{ for } i = k+1$$

$$= 0 \text{ otherwise,}$$

where

$$h_k = x_{k+1}-x_k.$$

Similarly,

$$\int_{x_0}^{x_N} e_i'(x)e_k(x)dx = -\frac{1}{2} \text{ for } i = k-1$$

$$= \frac{1}{2} \text{ for } i = k+1$$

$$= 0 \text{ otherwise.}$$

The outcome is most easily expressed in matrix/vector notation: let

$$\underset{\sim}{u} = (u_0, u_1, \ldots, u_N)^T$$

$$\underset{\sim}{v} = (v_0, v_1, \ldots, v_N)^T,$$

then using the above results Eq. (3.3) becomes

$$\underset{\sim}{P}\underset{\sim}{v} = \underset{\sim}{P}_x\underset{\sim}{u} \qquad (3.4)$$

where $\underline{\underline{P}}$ is tridiagonal; $\underline{\underline{P}}_x$ is also tridiagonal, but with zeros on the diagonal itself. The matrix $\underline{\underline{P}}$ is diagonally dominant, and the usual algorithm for solving tridiagonal systems can safely be used.

It is very instructive to consider (3.4) in the case of a *uniform* mesh; at interior points, we have just

$$\frac{1}{6}(v_{k-1}+4v_k+v_{k+1}) = \frac{1}{2h}(u_{k+1}-u_{k-1}) \tag{3.5}$$

where h is the gridlength.

If we perform a Taylor series analysis of (3.5), we find that on a uniform mesh we have 4th-order accuracy. General finite-element theory tells us that we only have a right to expect 2nd-order accuracy in the derivative if the basis functions are linear; here we have an example of "superconvergence" in which the (second-order) errors happen to cancel by symmetry.

Notice also that the manipulations leading to (3.4) automatically yield the appropriate equations for determining $v = u_x$ at the boundary points x_0 and x_N, in contrast to the finite-difference case (especially for fourth-order schemes) where in effect we have to "invent" additional values beyond the boundary.

The right-hand side of (3.5) is of course just the usual second-order finite-difference approximation to $v = u_x$, which is known to be too "smooth" (Fourier analysis shows that the derivative is underestimated, the degree of underestimation depending on the number of gridpoints per wavelength). The left-hand side of (3.5) - i.e., the matrix $\underline{\underline{P}}$ - is itself a three-point smoothing filter. To solve (3.4) we multiply the right-hand side by the *inverse* of $\underline{\underline{P}}$, which is thus a "de-smoothing" or "sharpening" operation. This simple argument provides some insight into the superiority of linear finite elements over second-order finite differences.

In finite element parlance, $\underline{\underline{P}}$ is often called the "mass matrix".

(b) Multiplication

Suppose now we are given $u_i = u(x_i)$ and $v_i = v(x_i)$ at a set of nodes x_i $(0 \le i \le N)$, and we wish to find the product $w = uv$. Again we expand u and v in terms of the basis functions $e_i(x)$, and similarly let

$$w(x) = \sum_{i=0}^{N} w_i e_i(x).$$

Applying the Galerkin technique as before,

$$\int_{x_0}^{x_N} \sum_{i=0}^{N} w_i e_i(x) e_k(x) dx = \int_{x_0}^{x_N} \left(\sum_{i=0}^{N} u_i e_i(x) \right) \left(\sum_{j=0}^{N} v_j e_j(x) \right) e_k(x) dx \qquad (3.6)$$

and we obtain a matrix problem

$$\underline{\underline{P}}\,\underline{w} = \underline{r}$$

where $\underline{\underline{P}}$ is the same tridiagonal matrix as in the previous sub-section. The right-hand side can also be expanded in a similar way; it is easily seen for example that the integral is zero unless both i and j are equal to $k-1$, k or $k+1$. In practice it is more efficient (*much* more efficient in 2 or 3 dimensions) to evaluate the integral by numerical quadrature. The right-hand side of (3.6) is a piecewise cubic, and can be evaluated exactly by using an appropriate integration formula. In fact we have a choice between Gaussian quadrature and using Simpson's formula; *Staniforth and Beaudoin* (1986) show that the second alternative is twice as efficient as the first (and that in a three-dimensional problem, Simpson quadrature is an order of magnitude more efficient than a straightforward term-by-term evaluation).

Notice that in deriving (3.6) we orthogonalized the error to the basis, so that the result $w(x)$ is an *alias-free* representation of the product uv, just as in the case of the spectral method. The product is fourth-order accurate on a uniform mesh (*Cullen and Morton*, 1980).

(c) Second derivatives

Finally, suppose we are given $u_i = u(x_i)$ at the nodes x_i $(0 \le i \le N)$, and we wish to obtain the second derivative $v = u_{xx}$. If we let

$$v(x) = \sum_{i=0}^{N} v_i e_i(x)$$

and proceed exactly as in Section 2(a), we obtain the following analogue of Eq. (3.2):

$$\int_{x_0}^{x_N} \sum_{i=0}^{N} v_i e_i(x) e_k(x) dx = \int_{x_0}^{x_N} \sum_{i=0}^{N} u_i e_i''(x) e_k(x) dx. \qquad (3.7)$$

Clearly we are in trouble here, since $e_i''(x)$ is zero everywhere. The trick is to rewrite the right-hand side of (3.7) as

$$\int_{x_0}^{x_N} u_{xx} e_k(x) dx$$

and then integrate by parts. We obtain

$$\int_{x_0}^{x_N} \sum_{i=0}^{N} v_i e_i(x) e_k(x) dx = [\, u_x e_k(x) \,]_{x_0}^{x_N} - \int_{x_0}^{x_N} u_x e_k'(x) dx \tag{3.8}$$

Now we can use the expansion of u in terms of the basis functions to replace u_x in the integration:

$$\int_{x_0}^{x_N} \sum_{i=0}^{N} v_i e_i(x) e_k(x) dx = [\, u_x e_k(x) \,]_{x_0}^{x_N} - \int_{x_0}^{x_N} \sum_{i=0}^{N} u_i e_i'(x) e_k'(x) dx,$$

and thus

$$\sum_{i=0}^{N} v_i \int_{x_0}^{x_N} e_i(x) e_k(x) dx = [\, u_x e_k(x) \,]_{x_0}^{x_N} - \sum_{i=0}^{N} u_i \int_{x_0}^{x_N} e_i'(x) e_k'(x) dx. \tag{3.9}$$

The left-hand side of (3.9), in matrix/vector notation, is just the familiar $\underline{\underline{P}}\underline{v}$ again. The first term on the right-hand side is zero except at the boundary points. The second term on the right-hand side contains the easily-evaluated integrals

$$\int_{x_0}^{x_N} e_i'(x) e_k'(x) dx = -\frac{1}{h_{k-1}} \text{ for } i = k-1$$

$$= \left(\frac{1}{h_{k-1}} + \frac{1}{h_k}\right) \text{ for } i = k$$

$$= -\frac{1}{h_k} \text{ for } i = k+1$$

where again $h_k = x_{k+1} - x_k$. Thus (3.9) has the form

$$\underline{\underline{P}}\underline{v} = \underline{\underline{P}}_{xx}\underline{u}$$

where $\underline{\underline{P}}$ and $\underline{\underline{P}}_{xx}$ are both tridiagonal.

On a uniform grid at interior points, (3.9) becomes

$$\frac{1}{6}(v_{k-1}+4v_k+v_{k+1}) = \frac{1}{h^2}(u_{k-1}-2u_k+u_{k+1}) \tag{3.10}$$

and again the right-hand side is just the usual second-order finite difference approximation. Unlike the case of taking a first derivative, however, inverting the mass matrix does *not* provide fourth-order accuracy; the accuracy remains second-order.

There is a way round this problem, provided we are willing to bend the rules of the Galerkin technique. If we replace the left-hand side of (3.10) by

$$\frac{1}{12}(v_{k-1}+10v_k+v_{k+1}),$$

in effect using a different "mass matrix", then we recover fourth-order accuracy on a uniform grid, with no extra computational work.

4. EFFICIENCY, ACCURACY AND CONSERVATION

(a) Efficiency

In the one-dimensional examples of the previous section, there was only one way to divide the domain into subintervals. As soon as we move to higher-dimensional problems, there is a choice. For example, in two dimensions we might choose to subdivide the domain into triangles (especially if the boundary of the domain is irregular) or into rectangles. In either case, linear finite-element basis functions can be defined on the subdivisions. *Staniforth* (1987) has powerfully argued the case for using a rectangular mesh if the geometry of the problem allows it.

The fundamental reason is the cost of inverting the mass matrix $\underline{P}$. On a *rectangular* mesh, the linear finite-element basis functions are *separable*:

$$e_{kl}(x,y) = e_k(x)e_l(y)$$

where e_{kl} is the basis function centered at the mesh point (k,l). As a result, the two-dimensional mass matrix can be inverted simply by solving a set of tridiagonal systems in the x-direction, followed by a set of tridiagonal systems in the y-direction (or vice versa). Another way of looking at this is that the two-dimensional mass matrix is just a tensor product $(\underline{P}^x \otimes \underline{P}^y)$, where $\underline{P}^x$ and $\underline{P}^y$ are the mass matrices associated with the one-dimensional sets of basis functions $\{e_k(x)\}$ and $\{e_l(y)\}$. This tensor product algorithm for inverting the two-dimensional mass matrix was first demonstrated by *Staniforth and Mitchell* (1977).

On a *triangular* mesh the separability of the basis functions is lost, and inverting the mass matrix is much more difficult. If the mesh is completely regular then a reasonably efficient FFT-based direct method could be used, but generally it is necessary to resort to iterative methods or approximate inverses (*Cullen*, 1974b). In engineering applications of the finite element method, the mass matrix is sometimes simply approximated by the identity matrix to circumvent this problem ("mass lumping"), but as we have seen in Section 3 this can seriously compromise the accuracy of the solution.

Staniforth (1987) also points out several other efficiency "tricks" which exist for linear finite elements on rectangles, but which do not carry over to triangular meshes. In three dimensions, the arguments for trilinear basis functions on "bricks" rather than on tetrahedra are even stronger.

(b) Accuracy

So far we have considered only methods based on *linear* finite elements. Is it worth using higher-order elements? In several respects, the answer seems to be no. Quadratic elements can be *less* accurate than linear elements (no superconvergence properties), so the extra expense is not likely to be justified. Cubic elements do have superconvergence properties and can provide a high order of accuracy, but they are much more expensive to compute with than linear elements; also, the additional degrees of freedom can result in computational modes (noise) being excited. *Staniforth* (1987) puts it succinctly: "the law of diminishing returns seems to apply".

(c) Conservation

Finite-difference schemes for nonlinear problems are often designed to conserve quadratic invariants of the original partial differential equations. In general, such conservation properties are not maintained in simple linear finite element schemes. They can be recovered for example by spatial staggering of the elements, or by choosing different orders of element for different variables (*Lee et al.*, 1980; *Cliffe*, 1981; *Girard*, 1983; *Steppeler*, 1987b). The extent to which the extra computation is worthwhile seems to be rather debatable, and in any case is certain to be problem-dependent.

5. APPLICATIONS

The first application of finite element methods to a meteorological problem seems to have been that of *Wang et al.* (1972), who integrated the one-dimensional shallow-water equations using Hermite cubic basis functions. They concluded that the finite element method requires less computational work than a fourth-order finite difference scheme, for a given level of accuracy.

Cullen (1974b), following some initial experiments in Cartesian geometry (*Cullen* 1973, 1974a), applied the finite element method to the shallow-water equations on a sphere. He used a grid based on an icosahedron

whose faces were subdivided into triangles, over which linear finite elements were defined. Tests with Rossby-Haurwitz waves suggested that the finite element method was competitive with finite-difference methods.

Cullen's finite element scheme in the horizontal was later coupled with the vertical finite difference scheme of the UK Meteorological Office's 5-level general circulation model. Results were reported by *Carson and Cullen* (1977) and *Cullen and Hall* (1979). In general these results continued to suggest that finite elements were competitive with finite differences, though perhaps not worth the additional effort.

Meanwhile, *Staniforth and Mitchell* (1977) applied linear finite elements on rectangles to the problem of the shallow-water equations on a polar stereographic projection. An important aspect of this work, besides recognizing the computational efficiency of the rectangular mesh, was the use of the vorticity/divergence form of the equations rather than the u–v form, and they concluded that their scheme gave significantly better results than a fourth-order finite difference scheme. The following year, *Staniforth and Mitchell* (1978) generalized their scheme to a non-uniform grid, with uniform high resolution only over the "area of interest" (see Fig. 3), and demonstrated that this gave an efficient configuration for short-range numerical weather prediction.

The first use of finite elements for *vertical* discretization was due to *Staniforth and Daley* (1977), who developed a fully Galerkin three-dimensional hemispheric model by coupling linear finite elements in the vertical with a spectral discretization in the horizontal, with encouraging results. Later, *Staniforth and Daley* (1979) produced the first model which used linear finite elements in all three dimensions, coupling the vertical discretization of *Staniforth and Daley* (1977) with the horizontal discretization of *Staniforth and Mitchell* (1977, 1978) on a polar stereographic projection with non-uniform resolution. They concluded that for short-range forecasts their model was competitive with a hemispheric spectral model.

Béland and Beaudoin (1985) extended the finite-element/spectral model of *Staniforth and Daley* (1977) to a global domain, and modified the vertical discretization to eliminate some noise problems. *Staniforth and Temperton* (1986) adapted the shallow-water model of *Staniforth and Mitchell* (1978) to use a semi-Lagrangian treatment of advection, demonstrating that this technique could be coupled with a finite element discretization.

Finite-element discretizations in the vertical were also tested in the ECMWF global spectral model (*Burridge et al.*, 1986; *Steppeler*, 1987a). Although the results were in many ways encouraging, noise problems in the vertical prevented operational implementation.

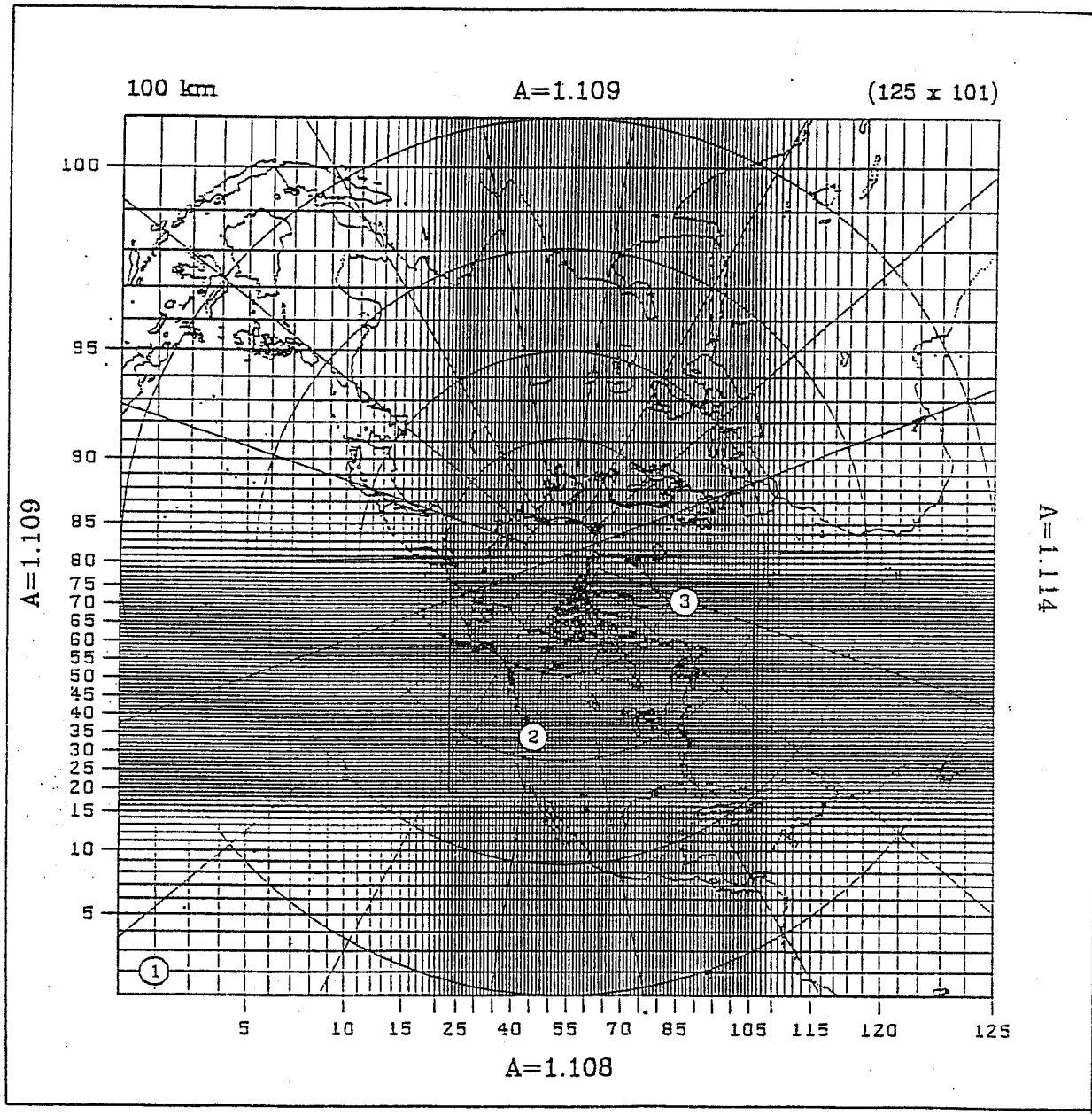

Figure 3. Variable-resolution mesh for a finite-element regional model.

Two models incorporating finite element discretization schemes are currently operational in Canada. *Tanguay et al.* (1989) adapted the model of *Staniforth and Daley* (1979), based on three-dimensional linear finite elements on a polar stereographic projection, to use a semi-Lagrangian treatment of advection; this model is used for short-range forecasts. Data assimilation and medium-range forecasts are provided by a global spectral model with finite elements in the vertical and a semi-Lagrangian integration scheme, as described by *Ritchie* (1991).

A recent Canadian development is the concept of "pseudo-staggering", in which the error is orthogonalized to a set of test functions which are constant over grid *cells*. In the shallow-water equations, this permits a return to the u–v formulation (rather than vorticity/divergence), without the usual problems of energy propagating in the wrong direction for small horizontal scales. Semi-Lagrangian advection schemes and variable resolution are easily incorporated. The technique is demonstrated for Cartesian geometry (stereographic projection) by *Côté, Gravel and Staniforth* (1990), and for spherical geometry (global model) by *Côté and Staniforth* (1990).

6. SUMMARY

In this brief survey, we have seen that finite-element methods are a viable alternative to finite-difference schemes, and that linear finite elements (on rectangles or "bricks" for 2- or 3-dimensional problems) have many advantages.

In the future, finite-element discretizations in the horizontal may seriously challenge the spectral technique for very high-resolution global models, when (and if) the computational work in the spectral method becomes overwhelmed by the contribution of the north-south Legendre transforms.

REFERENCES

Béland, M., and C. Beaudoin, 1985: A global spectral model with a finite element formulation for the vertical discretization: adiabatic formulation. Mon.Wea.Rev., 113, 1910-1919.

Burridge, D.M., J. Steppeler and R. Strüfing, 1986: Finite element schemes for the vertical discretization of the ECMWF forecast model using linear elements. ECMWF Technical Report No. 54, 51pp.

Carson, D.J., and M.J.P. Cullen, 1977: Intercomparison of short-range numerical forecasts using FD and FE models from the U.K. Meteorological Office. Beitr.Phys.Atmos., 50, 1-15.

Cliffe, K.A., 1981: On conservative finite element formulations of the inviscid Boussinesq equations. Int.J.Num.Meth.Fluids, 1, 117-127.

Côté, J., S. Gravel and A. Staniforth, 1990: Improving variable-resolution finite-element semi-Lagrangian integration schemes by pseudostaggering. Mon.Wea.Rev., 118, 2718-2731.

Côté, J., and A. Staniforth, 1990: An accurate and efficient finite-element global model of the shallow-water equations. Mon.Wea.Rev., 118, 2707-2717.

Cullen, M.J.P., 1973: A simple finite-element method for meteorological problems. J.Inst.Math.Applic., 11, 15-31.

Cullen, M.J.P., 1974a: A finite-element method for a non-linear initial value problem. J.Inst.Math.Applic., 13, 233-247.

Cullen, M.J.P., 1974b: Integrations of the primitive equations on a sphere using the finite element method. Q.J.R. Meteorol. Soc., 100, 555-562.

Cullen, M.J.P., and C.D. Hall, 1979: Forecasting and general circulation results from finite-element models. Q.J.R. Meteorol. Soc., 105, 571-592.

Cullen, M.J.P., and K.W. Morton, 1980: Analysis of evolutionary error in finite element and other methods. J.Comp.Phys., 34, 245-267.

Girard, C., 1983: Conservative vertical discretizations of the primitive equations. Part I: Theoretical aspects. Notes Scientifiques et Techniques, No. 11, Division de Recherche en Prévision Numérique, Dorval, Québec H9P 1J3, Canada, 88pp.

Lee, R.L., P.M. Gresho, S.T. Chan and R.L. Sani, 1980: A comparison of several conservative forms for finite-element formulations of the incompressible Navier-Stokes or Boussinesq equations. Proc.3rd Int.Conf. on Finite Elements in Flow Problems, Banff, Canada, 216-227.

Ritchie, H., 1991: Application of the semi-Lagrangian method to a multilevel spectral primitive-equations model. Q.J.R. Meteorol. Soc., 117, 91-106.

Staniforth, A.N., and R.W. Daley, 1977: A finite-element formulation for the vertical discretization of sigma-coordinate primitive-equation models. Mon.Wea.Rev., 105, 1108-1118.

Staniforth, A.N., and H.L. Mitchell, 1977: A semi-implicit finite-element barotropic model. Mon.Wea.Rev., 105, 154-169.

Staniforth, A.N., and H.L. Mitchell, 1978: A variable-resolution finite-element technique for regional forecasting with the primitive equations. Mon.Wea.Rev., 106, 439-447.

Staniforth, A.N., and R.W. Daley, 1979: A baroclinic finite-element model for regional forecasting with the primitive equations. Mon.Wea.Rev., 107, 107-121.

Staniforth, A., and C. Beaudoin, 1986: On the efficient evaluation of certain integrals in the Galerkin FE method. Int.J.Num.Meth.Fluids, 6, 317-324.

Staniforth, A., and C. Temperton, 1986: Semi-implicit semi-Lagrangian integration schemes for a barotropic finite-element regional model. Mon.Wea.Rev., 114, 2078-2090.

Staniforth, A., 1987: Review: Formulating efficient finite-element codes for flows in regular domains. Int.J.Num.Meth.Fluids, 7, 1-16.

Steppeler, J., 1987a: Quadratic Galerkin finite element schemes for the vertical discretization of numerical forecast models. Mon.Wea.Rev., 115, 1575-1588.

Steppeler, J., 1987b: Energy conserving Galerkin finite element schemes for the primitive equations of numerical weather prediction. J.Comp.Phys., 69, 258-264.

Strang, G., and G. Fix, 1973: An analysis of the finite element method. Prentice-Hall, Englewood Cliffs, N.J.

Tanguay, M., A. Simard, and A. Staniforth, 1989: A three-dimensional semi-Lagrangian scheme for the Canadian regional finite-element forecast model. Mon.Wea.Rev., 117, 1861-1871.

Wang, H.H., P. Halpern, J. Douglas, and T. Dupont, 1972: Numerical solutions of the one-dimensional primitive equations using Galerkin approximations with localized basis functions. Mon.Wea.Rev., 100, 738-746.