Chapter 13

Numerical Modeling and Prediction

Dynamical meteorology provides the theoretical basis and methodology for modern weather forecasting. Stated simply, the objective of dynamical forecasting is to predict the future state of the atmospheric circulation from knowledge of its present state by use of numerical approximations to the dynamical equations. Fulfilling this objective requires observations of the initial state of the field variables, a closed set of prediction equations relating the field variables, and a method of integrating the equations in time to obtain the future distribution of the field variables.

Numerical prediction is a highly specialized field, which is continually evolving. Operational forecast centers utilize complex prediction models that require the largest available supercomputers to obtain a solution. It is difficult to provide more than a superficial introduction to such models in an introductory text. Fortunately, however, many aspects of numerical prediction can be illustrated using a simple model, such as the barotropic vorticity equation. In fact, this equation was the basis of the earliest operational numerical prediction models.

13.1 HISTORICAL BACKGROUND

The British scientist L. F. Richardson made the first attempt to predict the weather numerically. His book, *Weather Prediction by Numerical Process*, published in 1922, is the classic treatise in this field. In his work Richardson showed how the differential equations governing atmospheric motions could be written approximately as a set of algebraic difference equations for values of the tendencies of various field variables at a finite number of points in space. Given the observed values of the field variables at these grid points, the tendencies could be calculated numerically by solving the algebraic difference equations.

By extrapolating the computed tendencies ahead a small increment in time, an estimate of the fields at a short time in the future could be obtained. The new values of the field variables could then be used to recompute the tendencies, which could in turn be used to extrapolate further ahead in time, and so on. Even

An Introduction to Dynamic Meteorology. DOI: 10.1016/B978-0-12-384866-6.00013-1

for short-range forecasting over a small area of Earth, this procedure requires an enormous number of arithmetic operations. Richardson did not foresee the development of high-speed digital computers. He estimated that a workforce of 64,000 people would be required just to keep up with the weather on a global basis!

Despite the tedious labor involved, Richardson worked out one example forecast for surface pressure tendencies at two grid points. Unfortunately, the results were very poor. Predicted pressure changes were an order of magnitude larger than those observed. At the time this failure was thought to be due primarily to the poor initial data available, especially the absence of upper-air soundings. However, it is now known that there were other, even more serious problems with Richardson's scheme.

After Richardson's failure to obtain a reasonable forecast, numerical prediction was not again attempted for many years. Finally, after World War II interest in numerical prediction revived partly because of the vast expansion of the meteorological observation network, which provided much improved initial data. Even more important was the development of digital computers, which made the enormous volume of arithmetic operations required in a numerical forecast feasible. At the same time it was realized that Richardson's scheme was not the simplest possible scheme for numerical prediction. His equations not only governed the slow-moving meteorologically important motions, but also included high-speed sound and gravity waves as solutions. Such waves are in nature usually very weak in amplitude. However, for reasons that will be explained later, if Richardson had carried his numerical calculation beyond the initial time step, these oscillations would have amplified spuriously, thereby introducing so much "noise" in the solution that the meteorologically relevant disturbances would have been obscured.

The American meteorologist J. G. Charney showed in 1948 how the dynamical equations could be simplified by systematic introduction of the geostrophic and hydrostatic approximations so that sound and gravity oscillations were filtered out. The equations that resulted from Charney's filtering approximations were essentially those of the quasi-geostrophic model. Thus, Charney's approach utilized the conservative properties of potential vorticity. A special case of this model, the *equivalent barotropic model*, was used in 1950 to make the first successful numerical forecast.

This model provided forecasts of the geopotential near 500 hPa. Thus, it did not forecast "weather" in the usual sense. It could, however, be used by forecasters as an aid in predicting the local weather associated with large-scale circulations. Later multilevel versions of the quasi-geostrophic model provided explicit predictions of surface pressure and temperature distributions, but the accuracy of such predictions was limited because of the approximations inherent in the quasi-geostrophic model.

With the development of vastly more powerful computers and more sophisticated modeling techniques, numerical forecasting has now returned to models

that are quite similar to Richardson's formulation and are far more accurate than quasi-geostrophic models. Nevertheless, it is still worth considering the simplest filtered model, the barotropic vorticity equation, to illustrate some of the technical aspects of numerical prediction in a simple context.

13.2 NUMERICAL APPROXIMATION OF THE EQUATIONS OF MOTION

The equations of motion are an example of a general class of systems known as *initial value problems*. A system of differential equations is referred to as an initial value problem when the solution depends not only on boundary conditions but also on the values of the unknown fields or their derivatives at some initial time. Clearly, weather forecasting is a primary example of a *nonlinear* initial value problem. Due to its nonlinearity, even the simplest forecast equation, the barotropic vorticity equation, is rather complicated to analyze. Fortunately, general aspects of the numerical solution of initial value problems can be illustrated using linearized prototype equations that are much simpler than the barotropic vorticity equation.

13.2.1 Finite Differences

The equations of motion involve terms that are quadratic in the dependent variables (the advection terms). Such equations generally cannot be solved analytically. Rather, they must be approximated by some suitable discretization and solved numerically. The simplest form of discretization is the *finite difference* method.

To introduce the concept of finite differencing, we consider a field variable $\psi(x)$, which is a solution to some differential equation in the interval $0 \le x \le L$. If this interval is divided into J subintervals of length δx, then $\psi(x)$ can be approximated by a set of $J+1$ values as $\Psi_j = \psi(j\delta x)$, which are the values of the field at the $J+1$ grid points given by $x = j\delta x, j = 0, 1, 2, \ldots, J$, where $\delta x = L/J$. Provided that δx is sufficiently small compared to the scale on which ψ varies, the $J+1$ grid point values should provide good approximations to $\psi(x)$ and its derivatives.

The limits of accuracy of a finite difference representation of a continuous variable can be assessed by noting that the field can also be approximated by a finite Fourier series expansion:

$$\psi(x) = \frac{a_0}{2} + \sum_{m=1}^{J/2} \left[a_m \cos \frac{2\pi m x}{L} + b_m \sin \frac{2\pi m x}{L} \right] \tag{13.1}$$

The available $J+1$ values of Ψ_j are just sufficient to determine the $J+1$ coefficients in (13.1). That is, it is possible to determine a_0 plus a_m and b_m for wave numbers $m = 1, 2, 3, \ldots, J/2$. The shortest wavelength component in (13.1)

has wavelength $L/m = 2L/J = 2\delta x$. Thus, the shortest wave that can be resolved by finite differencing has a wavelength twice that of the grid increment. Accurate representation of derivatives is only possible, however, for wavelengths much greater than $2\delta x$.

We now consider how the grid point variable Ψ_j can be used to construct a finite difference approximation to a differential equation. That is, we wish to represent derivatives such as $d\psi/dx$ and $d^2\psi/dx^2$ in terms of the finite difference fields. We first consider the Taylor series expansions about the point x_0:

$$\psi(x_0 + \delta x) = \psi(x_0) + \psi'(x_0)\delta x + \psi''(x_0)\frac{(\delta x)^2}{2} + \psi'''(x_0)\frac{(\delta x)^3}{6} + \mathrm{O}\left[(\delta x)^4\right] \tag{13.2}$$

$$\psi(x_0 - \delta x) = \psi(x_0) - \psi'(x_0)\delta x + \psi''(x_0)\frac{(\delta x)^2}{2} - \psi'''(x_0)\frac{(\delta x)^3}{6} + \mathrm{O}\left[(\delta x)^4\right] \tag{13.3}$$

where the primes indicate differentiation with respect to x, and $\mathrm{O}[(\delta x)^4]$ means that terms with order of magnitude $(\delta x)^4$ or less are neglected.

Subtracting (13.3) from (13.2) and solving for $\psi'(x)$ give a finite difference expression for the first derivative of the form

$$\psi'(x_0) = [\psi(x_0 + \delta x) - \psi(x_0 - \delta x)]/(2\delta x) + \mathrm{O}\left[(\delta x)^2\right] \tag{13.4}$$

while adding the same two expressions and solving for $\psi''(x)$ give a finite difference expression for the second derivative of the form

$$\psi''(x_0) = [\psi(x_0 + \delta x) - 2\psi(x_0) + \psi(x_0 - \delta x)]/(\delta x)^2 + \mathrm{O}\left[(\delta x)^2\right] \tag{13.5}$$

Because the difference approximations in (13.4) and (13.5) involve points at equal distances on either side of x_0, they are called *centered differences*. These approximations neglect terms of order $(\delta x)^2$. We thus say that the truncation error is order $(\delta x)^2$. Higher accuracy can be obtained by decreasing the grid interval, but at the cost of increasing the density of grid points. Alternatively, it is possible to obtain higher-order accuracy without decreasing the grid spacing by writing formulas analogous to (13.2) and (13.3) for the interval $2\delta x$ and using these together with (13.2) and (13.3) to eliminate error terms less than order $(\delta x)^4$. This approach, however, has the disadvantage of producing more complicated expressions and can be difficult to implement near boundary points.

13.2.2 Centered Differences: Explicit Time Differencing

As a prototype model, consider the linear one-dimensional advection equation

$$\partial q/\partial t + c\partial q/\partial x = 0 \tag{13.6}$$

with c a specified speed and $q(x, 0)$ a known initial condition. This equation can be approximated to second-order accuracy in x and t by the centered difference equation

$$[q(x, t + \delta t) - q(x, t - \delta t)]/(2\delta t) = -c\,[q\,(x + \delta x, t) - q\,(x - \delta x, t)]/(2\delta x) \tag{13.7}$$

The original differential equation (13.6) is thus replaced by a set of algebraic equations (13.7), which can be solved to determine solutions for a finite set of points that define a grid mesh in x and t (see Figure 13.1). For notational convenience it is useful to identify points on the grid mesh by indices m, s. These are defined by letting $x = m\delta x$, where $m = 0, 1, 2, 3, \ldots, M$, and $t = s\delta t$; $s = 0, 1, 2, 3, \ldots, S$. Writing $\hat{q}_{m,s} \equiv q(m\delta x, s\delta t)$, the difference equation (13.7) can then be expressed as

$$\hat{q}_{m,s+1} - \hat{q}_{m,s-1} = -\sigma\,(\hat{q}_{m+1,s} - \hat{q}_{m-1,s}) \tag{13.8}$$

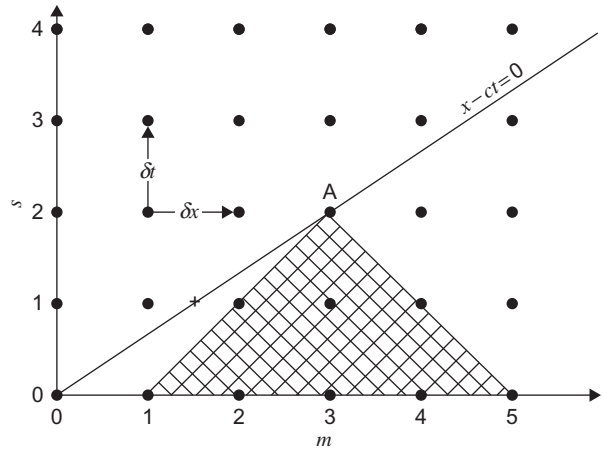

FIGURE 13.1 Grid in $x - t$ space showing the domain of dependence of the explicit finite-difference solution at $m = 3$ and $s = 2$ for the one-dimensional linear advection equation. *Solid circles* show grid points. The *sloping line* is a characteristic curve along which $q(x, t) = q(0, 0)$, and the "+" shows an interpolated point for the semi-Lagrangian differencing scheme. In this example the leapfrog scheme is unstable because the finite difference solution at point A does not depend on $q(0, 0)$.

where $\sigma \equiv c\delta t/\delta x$ is the *Courant number*. This form of time differencing is referred to as the *leapfrog* method, as the tendency at time step s is given by the difference in values computed for time steps $s+1$ and $s-1$ (i.e., by leaping across point s).

Leapfrog differencing cannot be used for the initial time $t = 0$ ($s = 0$), as $\hat{q}_{m,-1}$ is not known. For the first time step, an alternative method is required such as the forward difference approximation

$$\hat{q}_{m,1} - \hat{q}_{m,0} \equiv -(\sigma/2)(\hat{q}_{m+1,0} - \hat{q}_{m-1,0}) \tag{13.9}$$

The centered difference scheme for the advection equation (13.8) is an example of an *explicit* time differencing scheme. In an explicit difference scheme, the value of the predicted field at a given grid point for time step $s + 1$ depends only on the known values of the field at previous time steps. (In the case of the leapfrog method the fields at time step s and $s - 1$ are used.) The difference equation can then be solved simply by marching through the grid and obtaining the solution for each point in turn. The explicit leapfrog scheme is thus simple to solve. However, as shown in the next section, it has disadvantages in that it introduces a spurious "computational" mode and has stringent requirements on the maximum value permitted for the Courant number. There are a number of alternative explicit schemes that do not introduce a computational mode (e.g., see Problem 13.3), but these still require that the Courant number be sufficiently small.

13.2.3 Computational Stability

Experience shows that solutions to finite difference approximations such as (13.8) will not always resemble solutions to the original differential equations even when the finite difference increments in space and time are very small. It turns out that the character of the solutions depends critically on the *computational stability* of the difference equations. If the difference equations are not stable, numerical solutions will exhibit exponential growth even when, as in the case of the linear advection equation, the original differential equation system has solutions whose amplitudes remain constant in time.

In the example (13.8), stability considerations place stringent limitations on the value of the parameter σ, as we show later. If the initial condition is specified as

$$q\,(x, 0) = \mathrm{Re}\,[\exp(ikx)] = \cos\,(kx)$$

The analytic solution of (13.6) that satisfies this initial condition is

$$q(x, t) = \mathrm{Re}\,\{\exp\,[ik\,(x - ct)]\} = \cos\,(kx - ct) \tag{13.10}$$

We now compare (13.10) with the solution of the finite difference system (13.8) and (13.9). In finite difference form the initial condition is

$$\hat{q}_{m,0} = \exp(ikm\delta x) = \exp(ipm) \tag{13.11}$$

where $p \equiv k\delta x$. Noting that the analytic solution (13.10) is separable in x and t, we consider solutions of (13.8) and (13.9) of the form

$$\hat{q}_{m,s} = B^s \exp(ipm) \tag{13.12}$$

where B is a complex constant. Substituting into (13.8) and dividing through by the common factor B^{s-1}, we obtain a quadratic equation in B:

$$B^2 + (2i \sin\theta_P)B - 1 = 0 \tag{13.13}$$

where $\sin\theta_p \equiv \sigma \sin p$. Equation (13.13) has two roots, which may be expressed in the form

$$B_1 = \exp(-i\theta_p), \quad B_2 = -\exp(+i\theta_p)$$

The general solution of the finite difference equation is thus

$$\hat{q}_{m,s} = \left[CB_1^s + DB_2^s\right]\exp(ipm) = Ce^{i(pm-\theta_p s)} + D(-1)^s e^{i(pm+\theta_p s)} \tag{13.14}$$

where C and D are constants to be determined by the initial conditions (13.11) and the first time step (13.9). The former gives $C + D = 1$, whereas the latter gives

$$Ce^{-i\theta_p} - De^{+i\theta_p} = 1 - i\sin\theta_p \tag{13.15}$$

Thus,

$$C = \frac{(1+\cos\theta_p)}{2\cos\theta_p}, \quad D = -\frac{(1-\cos\theta_p)}{2\cos\theta_p} \tag{13.16}$$

From inspection of (13.14) it is clear that the solution will remain finite for $s \to \infty$ provided that θ_p is real. If θ_p is imaginary, then one term in (13.14) will grow exponentially and the solution will be unbounded for $s \to \infty$. This sort of behavior is referred to as *computational instability*. Now because

$$\theta_p = \sin^{-1}(\sigma \sin p) \tag{13.17}$$

θ_p will be real only for $|\sigma \sin p| \le 1$, which can only be valid for all waves (i.e., for all p) if $\sigma \le 1$. Thus, computational stability of the difference equation (13.8) requires that

$$\sigma = c\delta t/\delta x \le 1 \tag{13.18}$$

which is referred to as the Courant–Friedrichs–Levy (CFL) stability criterion.

The CFL criterion for this example states that for a given space increment δx, the time step δt must be chosen so that the dependent field will be advected a distance less than one grid length per time step. The restriction on σ given by (13.18) can be understood physically by considering the character of the solution in the x, t plane as shown in Figure 13.1. In the situation shown in this figure, $\sigma = 1.5$. Examination of the centered difference system (13.8) shows that the numerical solution at point A in Figure 13.1 depends only on grid points within the cross-hatch region. However, because A lies on the *characteristic* line $x - ct = 0$, the true solution at point A depends only on the initial condition at $x = 0$ (i.e., a parcel that is initially at the origin will be advected to the point $3\delta x$ in time $2\delta t$). The point $x = 0$ is outside the *domain of influence* of the numerical solution. Hence, the numerical solution cannot possibly faithfully reproduce the solution to the original differential equation, since, as shown in Figure 13.1, the value at point A in the numerical solution has no dependence on the conditions at $x = 0$. Only when the CFL criterion is satisfied will the domain of influence of the numerical solution include the characteristic lines of the analytic solution.

Although the CFL condition (13.18) guarantees stability of the centered difference approximation to the one-dimensional advection equation, in general the CFL criterion is only a necessary and not sufficient condition for computational stability. Other ways of finite differencing the one-dimensional advection equation may lead to more stringent limits on σ than given in (13.18).

The existence of computational instability is one of the prime motivations for using filtered equations. In the quasi-geostrophic system, no gravity or sound waves occur. Thus, the speed c in (13.18) is just the maximum wind speed. Typically, $c < 100\,\mathrm{m\,s^{-1}}$ so that for a grid interval of 200 km, a time increment of over 30 min is permissible. However, in the nonhydrostatic equations commonly used in cloud-resolving models, the solution would have characteristics corresponding to acoustic modes, and to ensure that the domain of influence included such characteristics, c would need to be set equal to the speed of sound, which is the fastest wave described in that set of equations. In that case, $c \approx 300\,\mathrm{ms^{-1}}$, and for a 1-km vertical grid interval, a time step of only a few seconds would be permitted.

13.2.4 Implicit Time Differencing

The spurious computational mode introduced by the leapfrog time differencing scheme does not occur in a number of other explicit schemes, such as the Euler backward scheme discussed in Problem 13.3. Such schemes do, however, retain the time step limitation imposed by the CFL condition. This restriction and the computational mode are eliminated by utilization of an alternative finite differencing scheme called the *trapezoidal implicit scheme*. For the linear advection

equation (13.6), this scheme can be written in the form

$$\frac{(\hat{q}_{m,s+1} - \hat{q}_{m,s})}{\delta t} = -\frac{c}{2}\left[\frac{(\hat{q}_{m+1,s+1} - \hat{q}_{m-1,s+1})}{2\delta x} + \frac{(\hat{q}_{m+1,s} - \hat{q}_{m-1,s})}{2\delta x}\right] \tag{13.19}$$

Substituting the trial solution (13.12) into (13.19) yields

$$B^{s+1} = \left[\frac{1 - i\,(\sigma/2)\sin p}{1 + i\,(\sigma/2)\sin p}\right] B^s \tag{13.20}$$

where as before $\sigma = c\delta t/\delta x$ and $p = k\delta x$. Defining

$$\tan\theta_p \equiv (\sigma/2)\sin p \tag{13.21}$$

and eliminating the common term B^s in (13.20), it can be shown that

$$B = \left(\frac{1 - i\tan\theta_p}{1 + i\tan\theta_p}\right) = \exp(-2i\theta_p) \tag{13.22}$$

so that the solution may be expressed simply as

$$\hat{q}_{m,s} = A\exp\left[ik(m\delta x - 2\theta_p s/k)\right] \tag{13.23}$$

Equation (13.19) involves only two time levels. Hence, unlike (13.14), the solution yields only a single mode, which has phase speed $c' = 2\theta_p/(k\delta t)$. According to (13.21), θ_p remains real for all values of δt. (This should be contrasted to the situation for the explicit scheme given by (13.17).) Thus, the implicit scheme is absolutely stable. The truncation errors, however, can become large if θ_p is not kept sufficiently small (see Problem 13.9). A disadvantage of the implicit scheme is that the integration cannot proceed by marching through the grid, as in the explicit case. In (13.19) there are terms involving the $s + 1$ time level on both sides of the equal sign, and these involve the values at a total of three grid points. Thus, the system (13.19) must be solved simultaneously for all points on the grid. If the grid is large, this may involve inverting a very large matrix, and so is computationally intensive. Furthermore, it is usually not feasible to utilize the implicit approach for nonlinear terms, such as the advection terms in the momentum equations. Semi-implicit schemes in which the linear terms are treated implicitly and the nonlinear terms explicitly are discussed briefly in Section 13.5.

13.2.5 The Semi-Lagrangian Integration Method

The differencing schemes discussed previously are Eulerian schemes in which the time integration is carried out by computing the tendencies of the predicted fields at a set of grid points fixed in space. Although it would be possible in theory to carry out predictions in a Lagrangian framework by following a set of marked fluid parcels, in practice this is not a viable alternative, since shear and stretching deformations tend to concentrate marked parcels in a few regions. Thus, it is difficult to maintain uniform resolution over the forecast region. It is possible to take advantage of the conservative properties of Lagrangian schemes, while maintaining uniform resolution, by employing a semi-Lagrangian technique. This approach permits relatively long time steps while retaining numerical stability and high accuracy.

The semi-Lagrangian method can be illustrated in a very simple fashion with the one-dimensional advection equation (13.6). According to this equation, the field q is conserved following the zonal flow at speed c. Thus, for any grid point, $x_m = m\delta x$, and time $t_s = s\delta t$:

$$q(x_m, t_s + \delta t) = q(\tilde{x}_m^s, t_s) \tag{13.24}$$

Here, $\tilde{x}_m^s$ is the location at time t_s for the air parcel that is located at point x_m at time $t_s + \delta t$. This position in general does not lie on a grid point (see the "plus" marked on Figure 13.1), so evaluation of the right-hand side of (13.24) requires interpolating from the grid point values at time t. For $c > 0$ the position $\tilde{x}_m^s$ lies between the grid points x_{m-p} and x_{m-p-1}, where p is the integer part of the expression $c\delta t/\delta x$ (a measure of the number of grid points traversed in a time step). If linear interpolation is used,

$$q(\tilde{x}_m^s, t_s) = \alpha q(x_{m-p-1}, t_s) + (1 - \alpha) q(x_{m-p}, t_s)$$

where $\alpha = (x_{m-p} - \tilde{x}_m^s)/\delta x$. Thus, in Figure 13.1, $p = 1$, and to predict q at point A, data are interpolated between the points $m = 1$ and $m = 2$ to the point shown by the "+".

In an actual prediction model the velocity field is predicted rather than known, as in this simple example. Thus, for a two-dimensional field

$$q\,(x, y, t + \delta t) = q\,(x - u\delta t, y - v\delta t, t) \tag{13.25}$$

where the velocity components at time t can be used to estimate the fields at $t + \delta t$; once these are obtained they can be used to provide more accurate approximations to the velocities on the right in (13.25). The right side in (13.25) is again estimated by interpolation, which now must be carried out in two dimensions.

As shown in Figure 13.1, the semi-Lagrangian scheme guarantees that the domain of influence in the numerical solution corresponds to that of the physical problem. Thus, the scheme is computational stability for time steps much

longer than possible with an explicit Eulerian scheme. The semi-Lagrangian scheme also preserves the values of conservative properties quite accurately and is particularly useful for accurately advecting trace constituents such as water vapor.

13.2.6 Truncation Error

To be useful it is necessary not only that a numerical solution be stable but that it also provide an accurate approximation to the true solution. The difference between the numerical solution to a finite difference equation and the solution to the corresponding differential equation is called the *discretization error*. If this error approaches zero with δt and δx, the solution is called *convergent*. The difference between a differential equation and the finite difference analog to it is referred to as a *truncation error* because it arises from truncating the Taylor series approximation to the derivatives. If this error approaches zero as δt and δx go to zero, the scheme is called *consistent.* According to the Lax equivalence theorem,[1] if the finite difference formulation satisfies the consistency condition, then stability is the necessary and sufficient condition for convergence. Thus, if a finite difference approximation is consistent and stable, one can be certain that the discretization error will decrease as the difference intervals are decreased, even if it is not possible to determine the error exactly.

Because numerical solutions are as a rule sought only when analytic solutions are unavailable, it is usually not possible to determine the accuracy of a solution directly. For the linear advection equation with constant advection speed considered in Section 13.2.3, it is possible, however, to compare the solutions of the finite difference equation (13.8) and the original differential equation (13.6). We can then use this example to investigate the accuracy of the difference method introduced earlier.

From the preceding discussion we can already conclude that the magnitude of the truncation error in the present case will be of order δx^2 and δt^2. It is possible to obtain more precise information on accuracy from examination of the solution (13.14). Note that for $\theta_p \to 0$, $C \to 1$ and $D \to 0$. The part of the solution proportional to C is the *physical mode*. The part proportional to D is called the *computational mode* because it has no counterpart in the analytic solution to the original differential equation. It arises because centered time differencing has turned a differential equation that is first order in time into a second-order finite difference equation. The accuracy of the finite difference solution depends not only on the smallness of D and the closeness of C to unity but on the match between the phase speed of the physical mode and the phase speed in the analytic solution. The phase of the physical mode is given in (13.14) by

$$pm - \theta_p s = (p/\delta x)\left(m\delta x - \theta_p s\delta x/p\right) = k\left(x - c't\right)$$

[1] See Richtmyer and Morton (1967).

TABLE 13.1 Centered Finite Difference Accuracy

$L/\delta x$	p	θ_p	c'/c	$\lvert D\rvert/\lvert C\rvert$
2	π	π	–	∞
4	$\pi/2$	0.848	0.720	0.204
8	$\pi/4$	0.559	0.949	0.082
16	$\pi/8$	0.291	0.988	0.021
32	$\pi/16$	0.147	0.997	0.005

Note: Results apply to the advection equation as a function of resolution for $\sigma = 0.75$.

where $c' = \theta_p \delta x/(p\delta t)$ is the phase speed of the physical mode. Its ratio to the true phase speed is

$$c'/c = \theta_p \delta x/(pc\delta t) = \sin^{-1}(\sigma \sin p)/(\sigma p)$$

so that $c'/c \to 1$ as $\sigma p \to 0$. The dependence of c'/c and $|D|/|C|$ on wavelength is shown in Table 13.1 for the particular case where $\sigma = 0.75$.

It is clear from Table 13.1 that phase speed and amplitude errors both increase as the wavelength decreases. Short waves move slower than long waves in the finite difference solution, even though in the original equation all waves move at speed c. This dependence of phase speed on wavelength in the difference solution is called *numerical dispersion*. It is a serious problem for numerical modeling of any advected field that has sharp gradients (and thus large-amplitude short-wave components).

Short waves also suffer from having significant amplitude in the computational mode. This mode, which has no counterpart in the solution to the original differential equation, propagates opposite to the direction of the physical mode and changes sign from one time step to the next. This behavior makes it easy to recognize when the computational mode has significant amplitude.

13.3 THE BAROTROPIC VORTICITY EQUATION IN FINITE DIFFERENCES

The simplest example of a dynamical forecast model is the barotropic vorticity equation (11.14), which for a Cartesian β plane can be written in the form

$$\frac{\partial \zeta}{\partial t} = -F(x, y, t) \tag{13.26}$$

where

$$F(x, y, t) = \mathbf{V}_\psi \cdot \nabla (\zeta + f) = \frac{\partial}{\partial x}\left(u_\psi \zeta\right) + \frac{\partial}{\partial y}\left(v_\psi \zeta\right) + \beta v_\psi \tag{13.27}$$

and $u_\psi = -\partial\psi/\partial y$, $v_\psi = \partial\psi/\partial x$, and $\zeta = \nabla^2\psi$. We have here used the fact that the horizontal velocity is nondivergent ($\partial u_\psi/\partial x + \partial v_\psi/\partial y = 0$) to write the advection term in flux form. The advection of absolute vorticity $F(x, y, t)$ may be calculated provided that we know the field of $\psi(x, y, t)$. Equation (13.26) can then be integrated forward in time to yield a prediction for ζ. It is then necessary to solve the Poisson equation $\zeta = \nabla^2\psi$ to predict the streamfunction.

A straightforward solution method is the leapfrog scheme discussed in Section 13.2.2. This requires writing (13.27) in finite difference form. Suppose that the horizontal x,y space is divided into a grid of $(M + 1) \times (N + 1)$ points separated by distance increments δx and δy. Then we can write the coordinate position of a given grid point as $x_m = m\delta x$, $y_n = n\delta y$, where $m = 0, 1, 2, \ldots, M$ and $n = 0, 1, 2, \ldots, N$. Thus, any point on the grid is uniquely identified by the indices (m, n). A portion of such a grid space is shown in Figure 13.2.

Centered difference formulas of the type (13.4) can then be used to approximate derivatives in the expression $F(x, y, t)$. For example, if we assume that $\delta x = \delta y \equiv d$,

$$\begin{aligned} u_\psi &\approx u_{m,n} = -(\psi_{m,n+1} - \psi_{m,n-1})/2d \\ v_\psi &\approx v_{m,n} = +(\psi_{m+1,n} - \psi_{m-1,n})/2d \end{aligned} \tag{13.28}$$

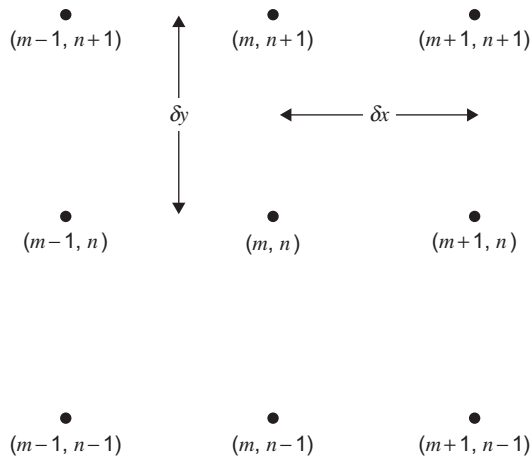

FIGURE 13.2 Portion of a two-dimensional (x, y) grid mesh for solution of the barotropic vorticity equation.

Similarly, with the aid of (13.5) we find that the horizontal Laplacian can be approximated as

$$\nabla^2\psi \approx (\psi_{m+1,n} + \psi_{m-1,n} + \psi_{m,n+1} + \psi_{m,n-1} - 4\psi_{m,n})/d^2 = \zeta_{m,n} \tag{13.29}$$

The finite difference form of the Laplacian is proportional to the difference between the value of the function at the central point and the average value at the four surrounding grid points. If there are $(M-1)\times(N-1)$ interior grid points, then (13.29) yields a set of $(M-1)\times(N-1)$ simultaneous equations, which together with suitable boundary conditions determine $\psi_{m,n}$ for a given array $\zeta_{m,n}$. This set can be solved by standard methods of matrix inversion.

Before expressing the advection term $F(x, y, t)$ in finite difference form, it is worth noting that if ψ is taken to be constant on the northern and southern boundaries of the β-plane channel, it is easily shown by integration over the area of the channel that the average value of F is zero. This implies that the mean vorticity is conserved for the channel. It is also possible, with a little more algebra, to show that the mean kinetic energy and the mean square vorticity (called the *enstrophy*) are conserved.

For accuracy of long-term integrations, it is desirable that any finite difference approximation to F satisfy the same conservation constraints as the original differential form; otherwise, the finite difference solution would not be conservative. The mean vorticity, for example, might then drift systematically in time purely because of the nature of the finite difference solution. Finite difference schemes that simultaneously conserve vorticity, kinetic energy, and enstrophy have been designed. They are, however, rather complex. For our purposes it is sufficient to note that by writing the advection in the flux form (13.27) and using centered space differences we can conserve both mean vorticity and mean kinetic energy:

$$\begin{aligned} F_{m,n} = \frac{1}{2d}\big[&\left(u_{m+1,n}\zeta_{m+1,n} - u_{m-1,n}\zeta_{m-1,n}\right) \\ &+ \left(v_{m,n+1}\zeta_{m,n+1} - v_{m,n-1}\zeta_{m,n-1}\right)\big] + \beta v_{m,n} \end{aligned} \tag{13.30}$$

It is verified readily that if ψ satisfies periodic boundary conditions, there is a cancellation of terms when (13.30) is summed over the domain. Thus,

$$\sum_{m=1}^{M}\sum_{n=1}^{N} F_{m,n} = 0 \tag{13.31}$$

Therefore, mean vorticity is conserved (except for errors introduced by time differencing) when (13.30) is used as the finite difference form of the advection term. This form also conserves mean kinetic energy (see Problem 13.2).

Enstrophy is not conserved in this difference formulation, and it is conventional to add a small diffusion term in order to control any numerically generated increase in enstrophy.

The procedure for preparing a numerical forecast with the barotropic vorticity equation can now be summarized as follows:

1. The observed geopotential field at the initial time is used to compute the initial streamfunction $\psi_{m,n}(t = 0)$ at all grid points.
2. $F_{m,n}$ is evaluated at all grid points.
3. $\zeta_{m,n}(t + \delta t)$ is determined using centered differencing except at the first time step when a forward difference must be used.
4. The simultaneous set (13.29) is solved for $\psi_{m,n}(t + \delta t)$.
5. The predicted array of $\psi_{m,n}$ is used as data, and steps 2 through 4 are repeated until the desired forecast period is reached. For example, a 24-h forecast with 30-min time increments would require 48 cycles.

13.4 THE SPECTRAL METHOD

In the finite difference method, the dependent variables are specified on a set of grid points in space and time, and derivatives are approximated using finite differences. An alternative approach, referred to as the spectral method, involves representing the spatial variations of the dependent variables in terms of finite series of orthogonal functions called *basis functions*. For the Cartesian geometry of a midlatitude β-plane channel, the appropriate set of basis functions is a double Fourier series in x and y. For spherical Earth, however, the appropriate basis functions are the spherical harmonics.

A finite difference approximation is *local* in the sense that the finite difference variable $\Psi_{m,n}$ represents the value of $\psi(x, y)$ at a particular point in space, and the finite difference equations determine the evolution of the $\Psi_{m,n}$ for all grid points. The spectral approach, however, is based on *global* functions—that is, the individual components of the appropriate series of basis functions. In the case of Cartesian geometry, for example, these components determine the amplitudes and phases of the sinusoidal waves that when summed determine the spatial distribution of the dependent variable. The solution proceeds by determining the evolution of a finite number of Fourier coefficients. Because the distribution in wave number space of the Fourier coefficients for a given function is referred to as its *spectrum*, it is appropriate to call this approach the spectral method.

At low resolution the spectral method is generally more accurate than the grid point method, partly because, for linear advection, the numerical dispersion discussed in Section 13.2.4 can be severe in a grid point model but does not occur in a properly formulated spectral model. For the range of resolutions used commonly in forecast models the two approaches are comparable in accuracy, and each has its advocates.

13.4.1 The Barotropic Vorticity Equation in Spherical Coordinates

The spectral method is particularly advantageous for solution of the vorticity equation. When the proper set of basis functions is chosen, it is trivial to solve the Poisson equation for the streamfunction. This property of the spectral method not only saves computer time but eliminates the truncation error associated with finite differencing the Laplacian operator.

In practice the spectral method is applied most frequently to global models. This requires use of spherical harmonics, which are more complicated than Fourier series. To keep the discussion as simple as possible, it is again useful to consider the barotropic vorticity equation as a prototype forecast model in order to illustrate the spectral method on the sphere.

The barotropic vorticity equation in spherical coordinates may be expressed as

$$\frac{D}{Dt}\left(\zeta + 2\Omega \sin\phi\right) = 0 \tag{13.32}$$

where, as usual, $\zeta = \nabla^2\psi$, with ψ a streamfunction, and

$$\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + \frac{u}{a\cos\phi}\frac{\partial}{\partial\lambda} + \frac{v}{a}\frac{\partial}{\partial\phi} \tag{13.33}$$

It turns out to be convenient to use $\mu \equiv \sin\phi$ as the latitudinal coordinate, in which case the continuity equation can be written

$$\frac{1}{a}\frac{\partial}{\partial\lambda}\left(\frac{u}{\cos\phi}\right) + \frac{1}{a}\frac{\partial}{\partial\mu}\left(v\cos\phi\right) = 0 \tag{13.34}$$

so that the streamfunction is related to the zonal and meridional velocities according to

$$\frac{u}{\cos\phi} = -\frac{1}{a}\frac{\partial\psi}{\partial\mu}; \quad v\cos\phi = \frac{1}{a}\frac{\partial\psi}{\partial\lambda} \tag{13.35}$$

The vorticity equation can then be expressed as

$$\frac{\partial\nabla^2\psi}{\partial t} = \frac{1}{a^2}\left[\frac{\partial\psi}{\partial\mu}\frac{\partial\nabla^2\psi}{\partial\lambda} - \frac{\partial\psi}{\partial\lambda}\frac{\partial\nabla^2\psi}{\partial\mu}\right] - \frac{2\Omega}{a^2}\frac{\partial\psi}{\partial\lambda} \tag{13.36}$$

where

$$\nabla^2\psi = \frac{1}{a^2}\left\{\frac{\partial}{\partial\mu}\left[\left(1-\mu^2\right)\frac{\partial\psi}{\partial\mu}\right] + \frac{1}{1-\mu^2}\frac{\partial^2\psi}{\partial\lambda^2}\right\} \tag{13.37}$$

The appropriate orthogonal basis functions are the *spherical harmonics*, which are defined as

$$Y_\gamma\left(\mu,\lambda\right) \equiv P_\gamma\left(\mu\right)e^{im\lambda} \tag{13.38}$$

where $\gamma \equiv (n, m)$ is a vector containing the integer indices for the spherical harmonics. These are given by $m = 0, \pm 1, \pm 2, \pm 3, \ldots, n = 1, 2, 3, \ldots$, where it is required that $|m| \leq n$. Here, P_γ designates an associated Legendre function of the first kind of degree n. From (13.38) it is clear that m designates the zonal wave number. It can be shown[2] that $n - |m|$ designates the number of nodes of P_γ in the interval $-1 < \mu < 1$ (i.e., between the poles), and thus measures the meridional scale of the spherical harmonic. The structures of a few spherical harmonics are shown in Figure 13.3.

An important property of the spherical harmonics is that they satisfy the relationship

$$\nabla^2 Y_\gamma = -\frac{n\,(n+1)}{a^2} Y_\gamma \tag{13.39}$$

so that the Laplacian of a spherical harmonic is proportional to the function itself, which implies that the vorticity associated with a particular spherical harmonic component is simply proportional to the streamfunction for the same component.

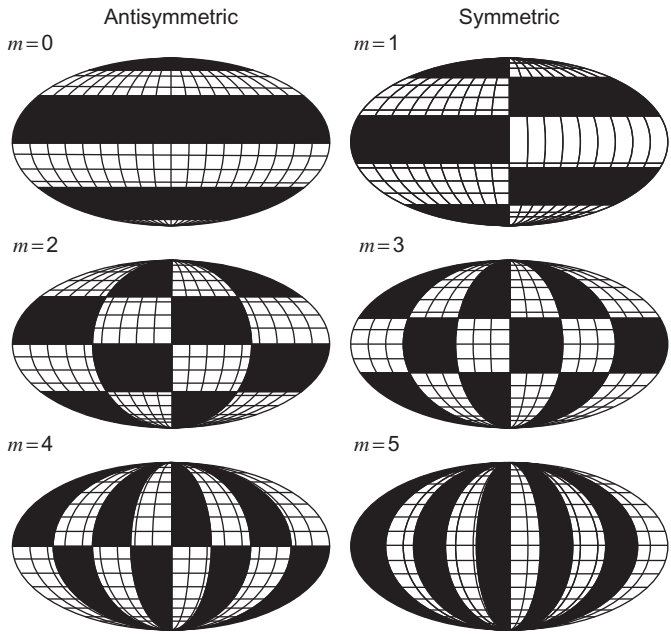

FIGURE 13.3 Patterns of positive and negative regions for the spherical harmonic functions with $n = 5$ and $m = 0, 1, 2, 3, 4, 5$. (*After Washington and Parkinson, 1986; adapted from Baer, 1972. Copyright © American Meteorological Society. Reprinted with permission.*)

[2]See Washington and Parkinson (1986) for a discussion of the properties of the Legendre function.

In the spectral method on the sphere, the streamfunction is expanded in a finite series of spherical harmonics by letting

$$\psi(\lambda, \mu, t) = \sum_{\gamma} \psi_\gamma(t) Y_\gamma(\mu, \lambda) \tag{13.40}$$

where ψ_γ is the complex amplitude for the Y_γ spherical harmonic and the summation is over both n and m. The individual spherical harmonic coefficients ψ_γ are related to the streamfunction $\psi(\lambda, \mu)$ through the inverse transform

$$\psi_\gamma(t) = \frac{1}{4\pi} \int_S Y_\gamma^* \psi(\lambda, \mu, t) dS \tag{13.41}$$

where $dS = d\mu d\lambda$ and Y_γ^* designates the complex conjugate of Y_γ.

13.4.2 Rossby–Haurwitz Waves

Before considering numerical solution of the barotropic vorticity equation, it is worth noting that an exact analytic solution of the nonlinear equation can be obtained in the special case where the streamfunction is equal to a single spherical harmonic mode. Thus, we let

$$\psi(\lambda, \mu, t) = \psi_\gamma(t)\, e^{im\lambda} P_\gamma(\mu) \tag{13.42}$$

Substituting from (13.42) into (13.36) and applying (13.39), we find that the nonlinear advection term is identically zero so that the amplitude coefficient satisfies the ordinary linear equation

$$-n\,(n+1)\, d\psi_\gamma/dt = -2\Omega i m \psi_\gamma \tag{13.43}$$

which has the solution $\psi_\gamma(t) = \psi_\gamma(0) \exp(i\nu_\gamma t)$, where

$$\nu_\gamma = 2\Omega m/\,[n\,(n+1)] \tag{13.44}$$

is the dispersion relationship for *Rossby–Haurwitz waves*, which is the name given to planetary waves on a sphere. (This should be compared to (5.110) with $\bar{u} = 0$, which is the equivalent expression for a midlatitude β-plane.) Because the horizontal scale of a spherical harmonic mode is proportional to n^{-1}, (13.44) shows that a single mode propagates westward on a sphere at a speed that is approximately proportional to the square of the horizontal scale. This solution also suggests why for some problems the spectral method is superior to the finite difference method at coarse resolution. A model containing even a single Fourier component can represent a realistic meteorological field (the Rossby wave), while many grid points are required for an equivalent representation in finite differences.

13.4.3 The Spectral Transform Method

When many spherical harmonic modes are present, the solution of (13.36) by a purely spectral method requires evaluation of the nonlinear interactions among various modes due to the advection term. It turns out that the number of interaction terms increases as the square of the number of modes retained in the series (13.40) so that this approach becomes computationally inefficient for models with the sort of spatial resolution required for prediction of synoptic-scale weather disturbances. The spectral transform method overcomes this problem by transforming between spherical harmonic wave number space and a latitude–longitude grid at every time step and carrying out the multiplications in the advection term in grid space. Thus, that it is never necessary to compute products of spectral functions.

To illustrate this method it is useful to rewrite the barotropic vorticity equation in the form

$$\frac{\partial \nabla^2 \psi}{\partial t} = -\frac{1}{a^2}\left[2\Omega \frac{\partial \psi}{\partial \lambda} + A\left(\lambda, \mu\right)\right] \tag{13.45}$$

where

$$A(\lambda, \mu) \equiv \left[-\frac{\partial \psi}{\partial \mu}\frac{\partial \nabla^2 \psi}{\partial \lambda} + \frac{\partial \psi}{\partial \lambda}\frac{\partial \nabla^2 \psi}{\partial \mu}\right] \tag{13.46}$$

Substituting from (13.40) into (13.45) then yields for the spherical harmonic coefficients

$$d\psi_\gamma / dt = i\nu_\gamma \psi_\gamma + A_\gamma [n\,(n+1)]^{-1} \tag{13.47}$$

where A_γ is the γ component of the transform of $A(\lambda, \mu)$:

$$A_\gamma = \frac{1}{4\pi}\int_0^{2\pi}\int_{-1}^{+1} A\left(\lambda, \mu\right) Y_\gamma^* d\lambda d\mu \tag{13.48}$$

The transform method utilizes the fact that if the sum $\gamma = (n, m)$ is taken over a finite number of modes, the integral in the transform (13.48) can be evaluated exactly by numerical quadrature—that is, by summing appropriately weighted values of $A(\lambda, \mu)$ evaluated at the grid points (λ_j, μ_k) of a latitude–longitude grid mesh. Computation of the distribution of $A(\lambda, \mu)$ for all grid points can be carried out without need to introduce finite differences for derivatives by noting that we can express the advection term in the form

$$A\left(\lambda_j, \mu_k\right) = \left(1 - \mu^2\right)^{-1}\left[F_1 F_2 + F_3 F_4\right] \tag{13.49}$$

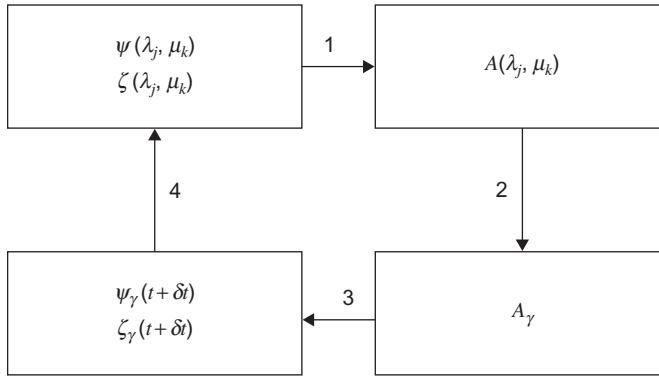

FIGURE 13.4 Steps in the prediction cycle for the spectral transform method of solution of the barotropic vorticity equation.

where

$$F_1 = -(1-\mu^2)\partial\psi/\partial\mu, \quad F_2 = \partial\nabla^2\psi/\partial\lambda$$
$$F_3 = (1-\mu^2)\partial\nabla^2\psi/\partial\mu, \quad F_4 = \partial\psi/\partial\lambda$$

The quantities $F_1 - F_4$ can be computed exactly for each grid point using the spectral coefficients ψ_γ and the known differential properties of the spherical harmonics. For example,

$$F_4 = \partial\psi/\partial\lambda = \sum_\gamma im\psi_\gamma Y_\gamma(\lambda_j, \mu_k)$$

Once these terms have been computed for all grid points, $A(\lambda, \mu)$ can be computed at the grid points by forming the products F_1F_2 and F_3F_4; no finite difference approximations to the derivatives are required in this procedure. The transform (13.48) is then evaluated by numerical quadrature to compute the spherical harmonic components A_γ. Finally, (13.47) can be stepped ahead by a time increment δt to obtain new estimates of the spherical harmonic components of the streamfunction. The whole process is then repeated until the desired forecast period is reached. The steps in forecasting with the barotropic vorticity equation using the spectral transform method are summarized in schematic form in Figure 13.4.

13.5 PRIMITIVE EQUATION MODELS

Modern numerical forecast models are based on a formulation of the dynamical equations, referred to as the *primitive equations*, which is essentially the formulation proposed by Richardson. The primitive equations differ from the

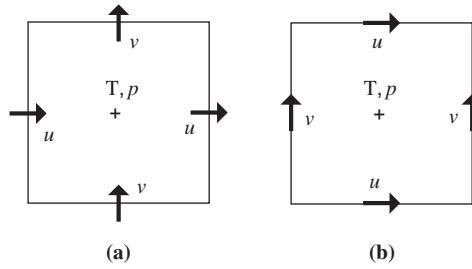

FIGURE 13.5 Horizontal grid staggering schemes for the Arakawa-C (a) and Arakawa-D (b) grids. Thermodynamic (T, p) and other scalar fields are located at the center of the grid cell; horizontal velocity components u and v are located along the grid edge.

complete momentum equations (2.19), (2.20), and (2.21) in that the vertical momentum equation is replaced by the hydrostatic approximation, and the small terms in the horizontal momentum equations given in columns C and D in Table 2.1 are neglected. In most models some version of the σ-coordinate system introduced in Section 10.3.1 is used, and the vertical dependence is represented by dividing the atmosphere into a number of levels and utilizing finite difference expressions for vertical derivatives. Both finite differencing and the spectral method have been used for horizontal discretization in operational primitive equation forecast models.

Grid point models utilize the finite difference method described in Section 13.2.1. Most grid point models utilize a *grid-staggering* approach to improve the accuracy of certain fields. Two examples of where the momentum field is staggered relative to the thermodynamic variables are shown in Figure 13.5. The Arakawa-C grid (a) is staggered so that the horizontal divergence ($\partial u/\partial x + \partial v/\partial y$) is located in the center of the grid cell, whereas the Arakawa-D grid (b) is staggered so that the vertical vorticity $\partial v/\partial x - \partial u/\partial y$ is located in the center of the grid cell.

13.5.1 Spectral Models

Most operational forecast centers utilize spectral models as their primary global forecast models, whereas grid point models are generally employed for fine-scale limited area models. Operational spectral models employ a primitive equation version of the spectral transform method described in Section 13.4.3. In this method the values of all meteorological fields are available in both spectral and grid point domains at each time step. Vorticity and divergence are employed as predictive variables rather than u and v. Physical computations involving such processes as radiative heating and cooling, condensation and precipitation, and convective overturning are calculated in physical space on the grid mesh, whereas differential dynamical quantities, such as pressure gradients and velocity gradients, are evaluated exactly in spectral space. This combination

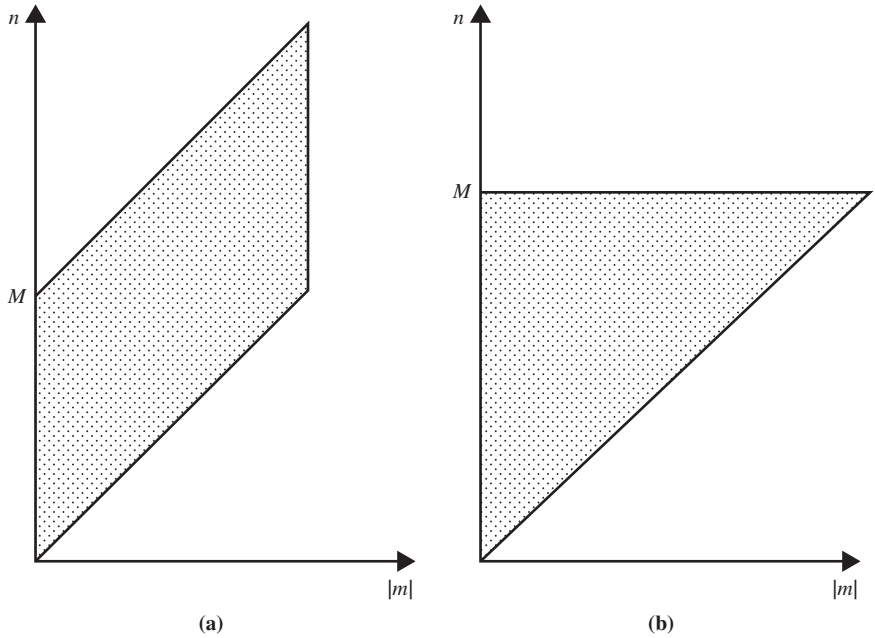

FIGURE 13.6 Regions of wave number space (n, m) in which spectral components are retained for the rhomboidal truncation (a) and triangular truncation (b). (*After Simmons and Bengtsson, 1984. Used with permission.*)

preserves the simplicity of the grid point representation for physical processes that are "local" in nature, while retaining the superior accuracy of the spectral method for dynamical calculations.

Two spectral truncations are typically employed in global spectral modes. The first, *triangular truncation*, refers to the fact that on a plot of m versus n (see 13.38) the retained modes occupy a triangular area. The second spectral truncation is *rhomboidal truncation*, which has $N = |m| + M$. Both of these truncations are shown schematically in Figure 13.6. In triangular truncation, the horizontal resolution in the zonal and meridional directions is nearly equal. In rhomboidal truncation, however, the latitudinal resolution is the same for every zonal wave number. Rhomboidal truncation has some advantages for low-resolution models, but at high resolution the triangular truncation appears to be superior.

13.5.2 Physical Parameterizations

The physical processes included in modern operational forecast models are generally the same as those included in general circulation models and were shown schematically in Figure 10.21. Inclusion of such processes as boundary layer

fluxes, vertical mixing by dry and moist convection, formation of clouds and precipitation, and the interaction of cloud and radiation fields requires that the relevant subgrid-scale processes be represented in terms of model-predicted fields. The approximation of unresolved processes in terms of resolved variables is referred to as *parameterization*; it is probably the most difficult and controversial area of weather and climate modeling.

Perhaps the most important physical process that must be parameterized is convection. Vertical heat transfer by convection is essential in maintaining the observed tropospheric lapse rate and moisture distribution. The simplest way to mimic this effect of unresolved convective motions is through *convective adjustment*. In a simple version of this method, the relative humidity and lapse rate in each grid column are examined at the end of each time step. If the lapse rate is super-adiabatic, the temperature profile is adjusted to dry static neutrality in a manner that conserves energy; if the column is conditionally unstable, and the humidity exceeds a specified value, the column is adjusted to moist static neutrality. More sophisticated schemes utilize the fact that moist convection is dependent on low-level moisture convergence, and they include a moisture budget as part of the parameterization.

Various methods are used to relate clouds to the resolved humidity, temperature, and wind fields. None is completely satisfactory. In some models the model-predicted cloud fields are used only in the portion of the model concerned with precipitation, and cloud–radiation interactions are parameterized using specified cloud climatologies.

13.6 DATA ASSIMILATION

Since forecasting is an initial value problem, observations are needed to establish the initial condition, which is called an analysis. Observations are distributed inhomogeneously in space and time, in contrast to numerical models, which have regularly spaced grids and forecasts that begin at specific times. Data assimilation is a technique that maps the inhomogeneous observations onto a regular grid, and also accounts for the errors in observations. As the name suggests, observations are assimilated into the numerical model, and the model is used to transmit the information from observations to a future time when observations are again available.

For this reason, data assimilation involves blending observations with estimates of the observations from short-range forecasts. Since the short-range forecasts start from relatively accurate initial conditions, the error in the model-estimated observations is so small that, for operational forecast systems, only about 15% of the analysis comes from millions of observations; the other 85% derives from the model, which has accumulated the information from all observations before that time. An essential aspect of data assimilation is that nothing is known exactly and the problem must be treated probabilistically.

13.6.1 Data Assimilation for a Single Variable

To define the data assimilation problem, and different techniques used to solve it, we begin with a single scalar before moving on to multiple variables. Given a scalar variable x, assume that there are two estimates of its true value: an observation, y, and a prior estimate, or background, x_b. The background may come from different sources, but mainly it comes from a short-range forecast. We wish to estimate the true value of x given the background and observation. The mathematical transcription of this statement involves a conditional probability

$$p(x_a) = p(x|y) \tag{13.50}$$

In words, the probability density of the analysis, x_a, is equal to the conditional probability density of the true state, x, given observation y (see Appendix G for background on conditional probability). The background information comes in through Bayes' Rule, which allows the conditional probability density to be "inverted" so that

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \tag{13.51}$$

where, in this case, $p(y)$ is just a scaling constant that may be neglected.[3] Since $p(y|x)$ is a function of the second argument, x, it is a *likelihood function* for the observation, given x. A more complete description is given in Appendix G, but the basic idea can be illustrated by flipping a coin to determine the likelihood that the coin is fair (e.g., determine the likelihood that the coin is fair from observing three consecutive tosses that all yield heads). This problem is the reverse of *knowing* that the coin is fair and determining the probability of an outcome conditioned on that knowledge (e.g., the probability of three heads in a row for a fair coin).

At this point the probability density functions are general, but for problems with many variables, a simplification is needed. The main simplification is that the variables are distributed normally (or Gaussian), which for a scalar reduces the probability density description to the mean and variance,

$$\begin{aligned} p(y|x) &= c_1 e^{-\frac{1}{2}\left(\frac{y-x}{\sigma_y}\right)^2} \\ p(x) &= c_2 e^{-\frac{1}{2}\left(\frac{x-x_b}{\sigma_b}\right)^2} \end{aligned} \tag{13.52}$$

where c_1 and c_2 are constants. The error variances of the observation and background are σ_y^2 and σ_b^2, respectively. Using (13.52) and (13.51) in (13.50) gives the analysis probability density

$$p(x_a) = Ce^{-\frac{1}{2}\left(\frac{x-x_b}{\sigma_b}\right)^2} e^{-\frac{1}{2}\left(\frac{y-x}{\sigma_y}\right)^2} \tag{13.53}$$

[3] The integral over $p(x_a)$, like all probability density functions, must be unity.

This result involves a product of Gaussians, which may be simplified to a new Gaussian distribution. To do so it proves useful to consider the following metric, J, of the probability density:

$$J(x) \equiv -\log\left[p(x)\right] = \frac{1}{2}\frac{(x-x_b)^2}{\sigma_b^2} + \frac{1}{2}\frac{(y-x)^2}{\sigma_y^2} - \log(C) \tag{13.54}$$

Metric J, the "cost function," is a quadratic measure of the misfit between the true value of the scalar and the two estimates we have, with smaller values of J indicating a better fit. Finding the best fit involves solving for the minimum value of J by taking $\partial J/\partial x$ and setting the result to zero. Solving for x at this point yields the analysis value, x_a,

$$x_a = \left(\frac{\sigma_y^2}{\sigma_y^2+\sigma_b^2}\right)x_b + \left(\frac{\sigma_b^2}{\sigma_y^2+\sigma_b^2}\right)y \tag{13.55}$$

This result reveals that the best estimate of the true state is given by a linear combination[4] of the prior estimate and the observation, with weights determined by the errors in these quantities. In the limit of vanishing error in the observation ($\sigma_y^2 \to 0$) the weight on the prior estimate approaches zero and the weight on the observation approaches one. The error variance for x_a is given by

$$\sigma_a^2 = \frac{\sigma_b^2}{1+\left(\frac{\sigma_b}{\sigma_y}\right)^2} = \frac{\sigma_y^2}{1+\left(\frac{\sigma_y}{\sigma_b}\right)^2} < \sigma_b^2, \sigma_y^2 \tag{13.56}$$

which shows that the error in the analysis is always less than the error in *both* the prior estimate and the observation. Thus the analysis is Gaussian distributed with mean x_a and variance σ_a^2.

The generalization from a scalar to vectors in the case of many analysis variables and many observations is facilitated by rewriting (13.55) as

$$x_a = x_b + K(y - x_b) \tag{13.57}$$

and (13.56) as

$$\sigma_a^2 = (1-K)\,\sigma_b^2 \tag{13.58}$$

where

$$K = \frac{\sigma_b^2}{\sigma_b^2+\sigma_y^2} \tag{13.59}$$

From (13.57) we see that the analysis may be viewed as a linear combination of the prior estimate and the *innovation*, or new information, contained in the

[4]The result is identical to the solution obtained using the method of linear least squares.

observation. If the observation gives the same value as our prior estimate, nothing has been learned, and there is no need to adjust the prior estimate. When there is new information, its weight relative to the background is controlled by the *gain* weighting factor, K, which is given by the ratio of the variance in the prior to the variance in the innovation.[5] K has an upper limit of one for $\sigma_y^2 \to 0$ and a lower limit of zero for $\sigma_b^2 \to 0$. Notice in (13.58) that, in the case of a perfect observation, the error in the prior is reduced to zero in the analysis.

A graphical illustration of these results is shown in Figure 13.7. Notice that, by assumption, the distribution in the observation (*thin solid line*) has less error than the prior (dashed line) and, as a result, the analysis (*thick solid line*) is weighted closer to the observation than the prior. Similarly, the distribution for the analysis has higher probability density than either the prior or the observation, because the analysis error variance is smaller as well. The thick gray curve shows the cost function, J, which reaches a minimum at the mean value of x_a.

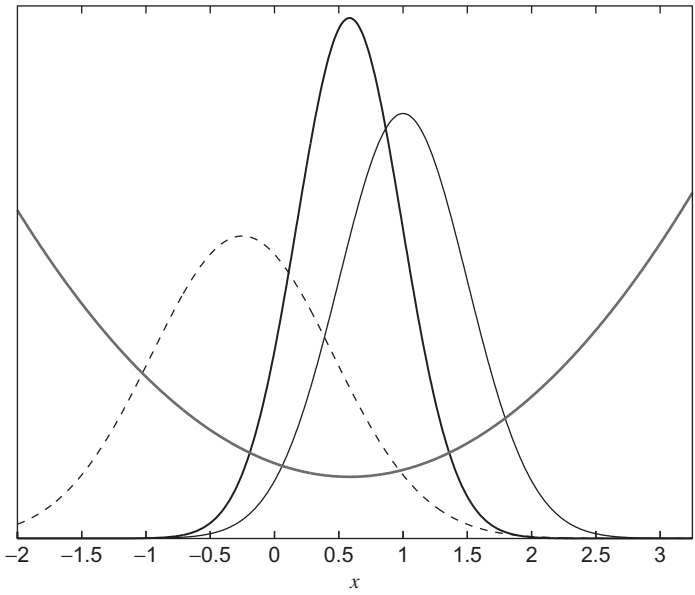

FIGURE 13.7 Data assimilation for scalar variable x assuming Gaussian error statistics. Prior estimate x_b, given by the *dashed line*, has mean −0.25 and variance 0.5. Observation y, given by the *thin solid line*, has mean 1.0 and variance 0.25. The analysis, x_a, given by the *thick solid line*, has mean 0.58 and variance 0.17. The parabolic *gray curve* denotes the cost function, J, which takes a minimum at the mean value of x_a.

[5]The variance of the innovation is given by the sum of variances $\sigma_b^2 + \sigma_y^2$ provided that errors in the prior and observation are uncorrelated.

13.6.2 Data Assimilation for Many Variables

The generalization to many variables involves a straightforward extension of the scalar case provided that matrix notation is used. Because the form of the equations is so similar to the scalar case, readers unfamiliar with matrix algebra should still be able to follow the discussion, if not the main points of the math as well.

The system is described by a *state vector*, $\mathbf{x}$, which is a column vector containing all of the grid point (or spectral) variables. For many variables, the cost function (13.54) becomes

$$J(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}_b)^{\mathrm{T}} \mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}_b) + \frac{1}{2}(\mathbf{y} - \mathbf{H}\,\mathbf{x})^{\mathrm{T}} \mathbf{R}^{-1}(\mathbf{y} - \mathbf{H}\mathbf{x}) \qquad (13.60)$$

As before, J is a scalar measuring the misfit between the true state and both the prior estimate and the observations. The weight on each term is no longer determined by the variances, σ_b^2 and σ_y^2, but rather by the covariance matrices, $\mathbf{B}$ and $\mathbf{R}$, for the prior and the observations, respectively. The diagonal elements of $\mathbf{B}$ contain the error variance of each variable in $\mathbf{x}$, whereas the off-diagonal elements contain the covariance between the variables, which measure the linear relationships between the variables (e.g., the relationship between pressure and wind may obey gradient wind balance).

Similarly, the diagonal elements of $\mathbf{R}$ contain the error variance of each observation; the off-diagonal elements, the covariance between the observations. Superscript T denotes a transpose, turning column vectors into row vectors, so that the (inner) products in both terms yield scalars. Matrix $\mathbf{H}$ is the "observation operator," which maps the state variables to the observations. Simple examples of $\mathbf{H}$ involve linear interpolation of grid point values of, for example, temperature, to the observation location where a measurement was made. A more complicated example involves radiative transfer calculations through the entire model atmosphere to yield an estimate of a satellite radiance.

The minimum value of J again determines the analysis, although this time careful application of matrix calculus and algebra are required to yield the multivariate analog of (13.57):

$$\mathbf{x}_a = \mathbf{x}_b + \mathbf{K}(\mathbf{y} - \mathbf{H}\mathbf{x}_b) \qquad (13.61)$$

The interpretation of (13.61) is identical to that of (13.57), except that now the weight is given by the *Kalman gain* matrix

$$\mathbf{K} = \mathbf{B}\mathbf{H}^{\mathrm{T}}\left[\mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}} + \mathbf{R}\right]^{-1} \qquad (13.62)$$

Although seemingly more complicated than (13.59), $\mathbf{K}$ may be interpreted similarly as the ratio of a prior covariance to an innovation covariance. The quantity $\mathbf{H}\mathbf{B}\mathbf{H}^{\mathrm{T}}$ measures the error covariance in the prior estimate of the observations, a direct analog of $\mathbf{R}$.

The multivariate version of (13.58) involves the analysis error covariance, **A**, given by

$$\mathbf{A} = (\mathbf{I} - \mathbf{KH})\mathbf{B} \tag{13.63}$$

where **I** is the identity matrix, and we see that the prior covariance is "reduced" by $\mathbf{I} - \mathbf{KH}$.

Analysis equations (13.61) and (13.63) represent one part of the Kalman filter, which is a recursive algorithm for estimating the state of a system sequentially in time. The second part of the Kalman filter includes a forecast from the analysis to a future time when observations are again available. In addition to a forecast of the prior estimate, (13.61) and (13.63) require a forecast of the error covariance **B**. Since the size of **B** is square in the number of variables, which can be on the order of 100 million for operational forecast systems, this calculation is prohibitively expensive. There are two leading approximations to deal with this problem: One gives variational assimilation, and the other gives the Ensemble Kalman filter (EnKF).

Rather than solving (13.61), which involves operations on enormous matrices including matrix inversion, in variational data assimilation (13.60) is solved by a direct search. Starting from the background, a descent strategy is pursued toward smaller J until changes become small within a tolerance. These strategies normally involve both J and its gradient, $\partial J/\partial \mathbf{x}$, which must be updated during the search. A common approximation in the variational approach is to assume a fixed covariance matrix for the prior, **B**, which eliminates the need to propagate a large matrix.

When the variational approach is performed at a single time, it is called three-dimensional variational assimilation (3DVAR). Augmenting the cost function (13.60) to include the misfit to observations over a window of time yields four-dimensional data assimilation (4DVAR). A discussion of the details is beyond the scope of this book, but 4DVAR requires substantially more resources than 3DVAR because the gradient descent involves an iteration over a period of time and specialized models (so-called "adjoints" of the forecast model), or ensembles, to accomplish this task. An example of a 4DVAR procedure, as used at the European Centre for Medium Range Weather Forecasts (ECMWF), is shown schematically in Figure 13.8. The ECMWF scheme utilizes a 12-h assimilation window extending from 9 h prior to 3 h after the analysis times, which are 00Z and 12Z. An assimilation cycle is begun by using a forecast from the initial state at the previous analysis time as the prior estimate. All available observations within the assimilation window are used to update the analysis.

The EnKF approach deals directly with (13.61) and renders the calculation manageable either by processing the observations one at a time or by breaking the domain into smaller subdomains where the matrix operations are tractable. Furthermore, rather than propagate the large matrix **B**, the EnKF employs a sampling strategy, where the nonlinear forecast model is used to evolve each of

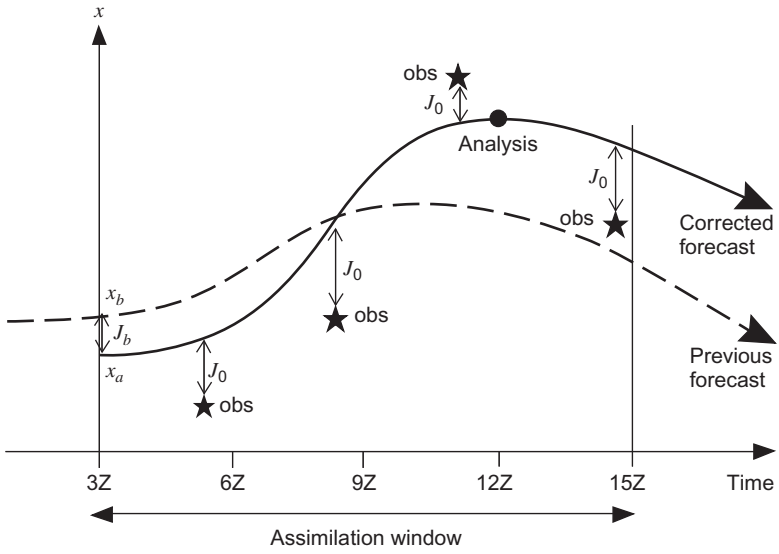

FIGURE 13.8 ECMWF data assimilation process. The *vertical axis* indicates the value of an atmospheric field variable. The *curve* labeled x_b is the first guess or "background" field, and x_a is the analysis field. J_b indicates the misfit between the background and analysis fields. *Stars* indicate the observations, and J_0 designates the "cost function," a measure of the misfit between the observations and the analysis, which along with J_b is minimized to produce the best estimate of the state of the atmosphere at the analysis time (*Courtesy ECMWF.*)

a chosen number of ensemble members. This sample provides the basis to estimate **B**, taking care to deal with sampling error. There are two main strategies to updating the forecast ensemble with observations to produce an ensemble analysis. The first strategy, called "perturbed observations" adds random error consistent with the error covariance matrix, **R**, to the observations, so that every ensemble member assimilates slightly different observations. The second strategy, called "square root filters," uses the observations to update the ensemble mean forecast to an ensemble mean analysis, and then uses a modified version of (13.61) to update the deviations of each ensemble member from the mean. There are several, statistically equivalent, techniques that accomplish the square root update. In addition to the benefits of using state-dependent estimates of the prior covariance, **B**, the EnKF also has the attractive property that an ensemble analysis is immediately available for ensemble forecasting.

13.7 PREDICTABILITY AND ENSEMBLE FORECASTING

Atmospheric prediction is a fundamentally probabilistic problem. Errors in the observations and prior are nonzero and, as a result, errors in the analysis are as well. The problem of atmospheric prediction is to determine how these errors,

as defined by a probability density function, evolve toward a future time. This probability density, p, obeys the *Liouville equation*,

$$\frac{Dp}{Dt} = -p\nabla \cdot \mathbf{F}(\mathbf{x}) \tag{13.64}$$

which expresses the conservation of total probability in direct analogy with mass conservation. The material derivative in this equation applies following the motion in *state-space* coordinates. The state-space velocity vector describes the change of the state vector, $\mathbf{x}$, in time through a set of coordinates defined by a particular basis (e.g., spectral or grid point variables):

$$\frac{D\mathbf{x}}{Dt} = \mathbf{F}(\mathbf{x}) \tag{13.65}$$

The velocity, $\mathbf{F}$, is a vector-valued function that defines the dynamics of the system—for example, conservation of momentum, mass, and energy.

Having thus far made no approximations, it is a remarkable fact that, since the Liouville equation is a linear first-order PDE in p, there is an exact solution to the atmospheric predictability problem:

$$p[\mathbf{x}(t), t] = p[\mathbf{x}(t_0), t_0] e^{-\int_{t_0}^{t} \nabla \cdot \mathbf{F}(\mathbf{x})\, dt'} \tag{13.66}$$

Given an initial probability density function, $p\,[\mathbf{x}(t_0), t_0]$, the probability density at a future time, $p\,[\mathbf{x}(t), t]$, is determined by the divergence of state-space trajectories. When state-space trajectories diverge, $\nabla \cdot \mathbf{F} > 0$, probability density decreases exponentially fast. A **chaotic system** is characterized by a sensitivity to initial conditions, which is realized by diverging state-space trajectories (solutions become different exponentially fast). As a result, for chaotic systems such as the atmosphere, probability density approaches zero over the attractor, while the probability over state space remains unity.

The evolution of a probability density function with time is clearly illustrated for the scalar differential equation (Ehrendorfer, 2006)

$$\frac{dx}{dt} = x - x^3 \tag{13.67}$$

An initially Gaussian distribution with a slightly positive mean shows that probability density quickly decreases and spreads in x as the solution diverges toward stable fixed points located at $x = -1$ and $x = 1$ (Figure 13.9a). State-space velocity (13.67) is positive (negative) for x positive (negative), so state-space trajectory divergence is maximized at $x = 0$, and becomes negative (convergence) for $|x| > 1/\sqrt{3}$. The solution ultimately converges at the fixed points with increasing probability density.

Unfortunately, numerical solution of (13.66) for any realistic problem is completely intractable. Consider discretizing the probability density by 100 bins

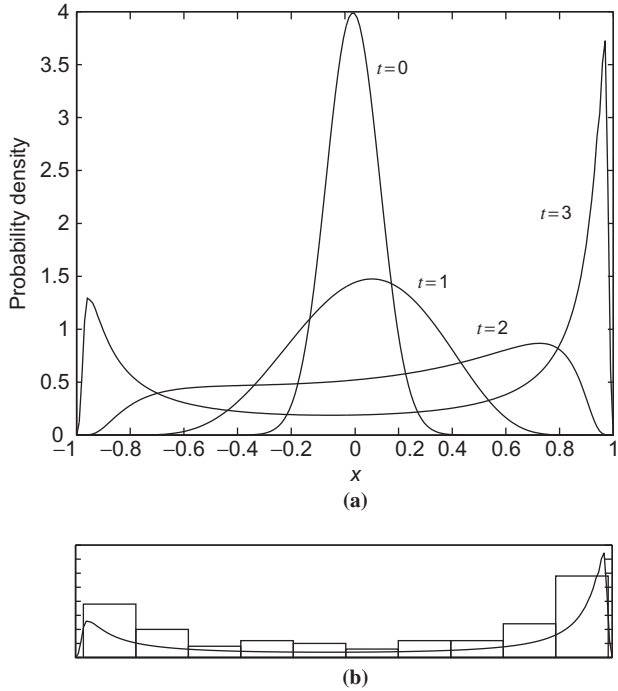

FIGURE 13.9 Time evolution of an initially Gaussian probability density function subject to the dynamics in (13.67) from numerically solving the Liouville equation (a). Populating a 100-member ensemble by randomly sampling from the initial probability density function, and solving (13.67) for the state at $t = 3$—summarized by a histogram in (b)—provides an estimate of the solution of the Liouville equation (*solid line*).

for each state-space variable, as was done for Figure 13.9. For a three-variable system, we have a formidable calculation involving 1 million degrees of freedom, which becomes an intractable 10 billion for a five-variable system; obviously, solutions for a modern NWP system having 100 million variables is out of the question. This motivates a sampling strategy using an ensemble approach, whereby only a random selection of states from the initial distribution are integrated in time. An example is shown in the lower panel of Figure 13.9, where a 100-member ensemble is randomly drawn from the initial distribution, and then integrated in time. The ensemble-based histogram at time $t = 3$ provides a coarse, but useful, approximation to the Liouville solution (*solid line*).

Forecast accuracy is ultimately limited by sensitivity to the initial state, which is an essential characteristic of chaotic systems. A simple illustration of sensitivity to initial errors is given by the three-variable model of Lorenz (1963) (see Exercise M13.9). The sensitivity of forecasts in this system is

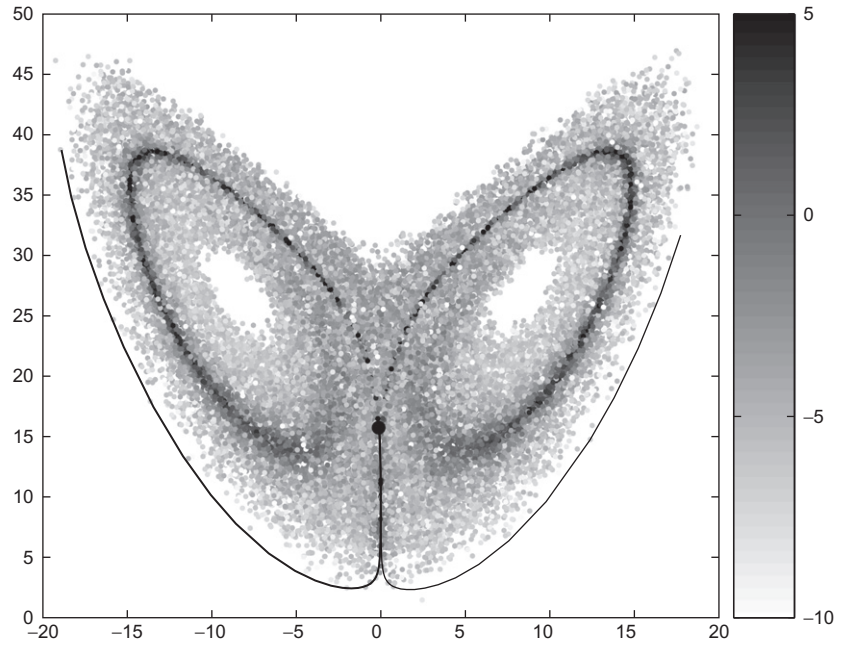

FIGURE 13.10 Sensitivity of forecasts to initial condition errors for the Lorenz (1963) attractor (forecasts are one time unit long). The natural logarithm of the errors is depicted in *gray-scale*, with *darker gray* indicating points that yield forecasts with larger errors. Solutions for the most sensitive location are given by *lines*.

illustrated in Figure 13.10, which shows the squared error in forecasts of fixed length (one nondimensional time unit) over the attractor. Errors are plotted on a logarithmic scale, so a wide range of errors is evident, with regions of large errors (*darker areas*) wrapping around the two fixed points toward the midpoint. Forecasts starting from a sensitive location are shown by the *solid curves*, which reveal that initially small differences grow suddenly as the two dynamical trajectories split in opposite directions. Such behavior is characteristic of a wide variety of systems governed by deterministic equations, including atmospheric flows.

Estimates of how this sort of error growth limits the inherent predictability of the atmosphere can be made using primitive equation forecast models. In these predictability experiments a "control" run is made using initial data corresponding to the observed flow at a given time. The initial data are then perturbed by introducing small random errors, as in Figure 13.10, and the model is run again. The growth of inherent error can then be estimated by comparing the second "forecast" run with the control. Results from a number of such studies indicate that the doubling time for the root mean square geopotential height error is

about 2 to 3 days for small errors and somewhat greater for large errors. Thus, the theoretical limit for predictability on the synoptic scale is probably about 2 weeks.

Actual predictive skill with present deterministic models is, however, less than the theoretical limit imposed by inherent error growth. A number of factors are probably responsible for the failure of current models to reach the theoretical forecast skill limits. These include observational and analysis errors in initial data, inadequate model resolution, and unsatisfactory representation of physical processes involving radiation, clouds, precipitation, and boundary layer fluxes. An indication of the skill of the present generation of global forecast models is given in Figure 13.11, in which forecast skill is plotted in terms of the *anomaly correlation* of the 500-hPa geopotential height field, defined as the correlation between observed and predicted deviations from climatology. Subjective evaluations suggest that useful forecasts are obtained when the anomaly correlation is greater than 0.6. Thus, over the past 20 years the range for which ECMWF forecasts show useful skill has increased from 5 days to more than 7 days. There can, of course, be quite wide variations in skill from one situation to another reflecting variations in the degree of predictability from one atmospheric flow regime to another.

Ensembles provide the main technique for estimating the evolution of error in initial value problems. A few basic statistical results are available to evaluate the performance of ensemble forecast systems. For an ensemble of size M, the

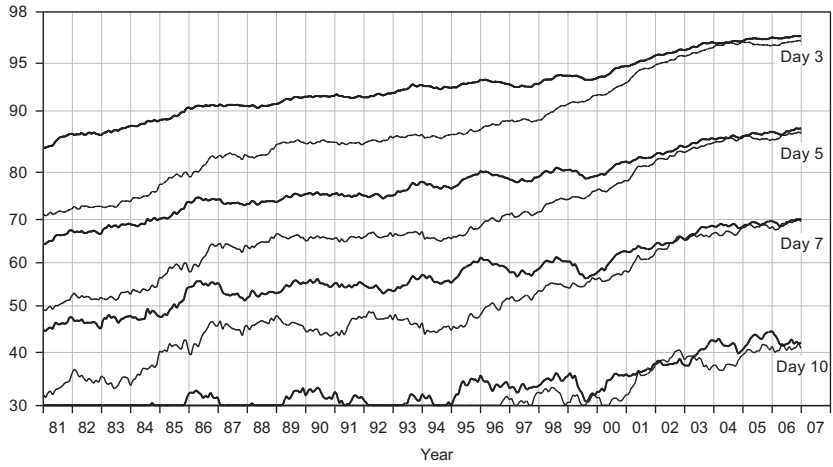

FIGURE 13.11 Anomaly correlation of 500-hPa height for 3-, 5-, 7-, and 10-day forecasts for the ECMWF operational model as a function of year. *Thick and thin lines* correspond to Northern and Southern Hemispheres, respectively. Note that the difference in skill between the two hemispheres has nearly disappeared in the last decade, a result of the successful assimilation of satellite data. (*Courtesy ECMWF.*)

ensemble spread, S, about the ensemble mean is defined as

$$S = \frac{1}{M-1}\sum_{j=1}^{M}\left(u_j - \overline{u}\right)^2 \tag{13.68}$$

where $\overline{u}$ is the ensemble mean and u_j is an ensemble member. This describes the variance about the ensemble mean. Averaging S over an infinite number of probability density functions, and over all states on the attractor (the "climate") gives

$$\langle S \rangle \equiv D \tag{13.69}$$

The error in ensemble mean is defined by

$$e_M = (\overline{u} - u_t)^2 \tag{13.70}$$

where u_t is the true (unknown) state. Again, averaging as above, it can be shown that (Murphy, 1989)

$$\langle e_M \rangle = \frac{M+1}{M} D \tag{13.71}$$

As a result, we see immediately that, for a deterministic forecast ($M = 1$), $\langle e_1 \rangle = 2D$.

Two main results derive from (13.69) and (13.71). The first is that the ratio of the error in the ensemble mean to a single deterministic forecast is

$$\frac{\langle e_M \rangle}{\langle e_1 \rangle} = \frac{M+1}{2M} \tag{13.72}$$

This clearly shows the benefit of averaging over errors with an ensemble: In the limit of large ensembles, the error in the ensemble mean is, on average, half of that for a single deterministic forecast (i.e., a randomly drawn ensemble member). The second main result involves the ratio of the error in the ensemble mean to the ensemble spread:

$$\frac{\langle e_M \rangle}{\langle S \rangle} = \frac{M+1}{M} \tag{13.73}$$

This shows that, except for small ensembles, the error in the ensemble mean should equal the spread in the ensemble. A common problem with ensemble forecast systems is that they have too little spread, so that (13.73) is larger than it would be for a well-calibrated system.

SUGGESTED REFERENCES

Durran, *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*, is an excellent graduate-level textbook covering the numerical methods used in the atmospheric sciences.

Kalnay, *Atmospheric Modeling, Data Assimilation and Predictability*, is an excellent graduate-level text on all aspects of modern numerical weather forecasting.

PROBLEMS

13.1. Show that for the barotropic vorticity equation on the Cartesian β plane (13.26) enstrophy and kinetic energy are conserved when averaged over the whole domain—that is, that the following integral constraints are satisfied:

$$\frac{d}{dt}\iint \frac{\zeta^2}{2}dxdy = 0, \qquad \frac{d}{dt}\iint \frac{\nabla\psi \cdot \nabla\psi}{2}dxdy = 0$$

Hint: To prove energy conservation, multiply (13.26) through by $-\psi$ and use the chain rule of differentiation.

13.2. Verify expression (13.31). Use periodic boundary conditions in both x and y.

13.3. The Euler backward method of finite differencing the advection equation is a two-step method consisting of a forward prediction step followed by a backward corrector step. In the notation of Section 13.2.2 the complete cycle is thus defined by

$$\hat{q}^*_m - \hat{q}_{m,s} = -\frac{\sigma}{2}\left(\hat{q}_{m+1,s} - \hat{q}_{m-1,s}\right)$$
$$\hat{q}_{m,s+1} - \hat{q}_{m,s} = -\frac{\sigma}{2}\left(\hat{q}^*_{m+1} - \hat{q}^*_{m-1}\right)$$

where $\hat{q}^*_m$ is the first guess for time step $s+1$. Use the method of Section 13.2.3 to determine the necessary condition for stability of this method.

13.4. Carry out truncation error analyses analogous to that of Table 13.1 for the centered difference approximation to the advection equation but for the cases $\sigma = 0.95$ and $\sigma = 0.25$.

13.5. Suppose that the streamfunction ψ is given by a single sinusoidal wave $\psi(x) = \mathrm{A}\sin(kx)$. Find an expression for the error of the finite difference approximation

$$\frac{\partial^2\psi}{\partial x^2} \approx \frac{\psi_{m+1} - 2\psi_m + \psi_{m-1}}{\delta x^2}$$

for $k\delta x = \pi/8, \pi/4, \pi/2,$ and π. Here $x = m\delta x$ with $m = 0, 1, 2, \ldots$.

13.6. Using the method given in Section 13.2.3, evaluate the computational stability of the following two finite difference approximations to the one-dimensional advection equation:

(a) $\hat{\zeta}_{m,s+1} - \hat{\zeta}_{m,s} = -\sigma\left(\hat{\zeta}_{m,s} - \hat{\zeta}_{m-1,s}\right)$

(b) $\hat{\zeta}_{m,s+1} - \hat{\zeta}_{m,s} = -\sigma\left(\hat{\zeta}_{m+1,s} - \hat{\zeta}_{m,s}\right)$

where $\sigma = c\delta t/\delta x > 0$. (The schemes labeled (a) and (b) are referred to as *upstream* and *downstream* differencing, respectively.) Show that scheme (a) damps the advected field and compute the fractional damping rate per time step for $\sigma = 0.25$ and $k\delta x = \pi/8$ for a field with the initial form $\zeta = \exp(ikx)$.

13.7. Using a staggered horizontal grid analogous to that shown in the left panel of Figure 13.5 (but for an equatorial β-plane geometry), express the linearized shallow water equations (11.27), (11.28), and (11.29) in finite difference form.

13.8. Verify the equality

$$\left(\frac{1 - i\tan\theta_p}{1 + i\tan\theta_p}\right) = \exp\left(-2i\theta_p\right)$$

given in (13.22).

13.9. Compute the ratio of the numerical phase speed to the true phase speed, c'/c, for the implicit differencing scheme of (13.19) for $p = \pi$, $\pi/2$, $\pi/4$, $\pi/8$, and $\pi/16$. Let $\sigma = 0.75$ and $\sigma = 1.25$. Compare your results to those of Table 13.1.

13.10. Using the technique of Section 13.2.1, show that the following four-point difference formula for the first derivative is of fourth-order accuracy:

$$\begin{aligned}\psi'(x_0) \approx \frac{4}{3}\left(\frac{\psi(x_0 + \delta x) - \psi(x_0 - \delta x)}{2\delta x}\right) \\ -\frac{1}{3}\left(\frac{\psi(x_0 + 2\delta x) - \psi(x_0 - 2\delta x)}{4\delta x}\right)\end{aligned} \tag{13.74}$$

13.11. The Dufort–Frankel method for approximating the one-dimensional diffusion equation

$$\frac{\partial q}{\partial t} = K\frac{\partial^2 q}{\partial x^2}$$

can be expressed in the notation of Section 13.2.2 as

$$\hat{q}_{m,s+1} = \hat{q}_{m,s-1} + r\left[\hat{q}_{m+1,s} - \left(\hat{q}_{m,s+1} + \hat{q}_{m,s-1}\right) + \hat{q}_{m-1,s}\right]$$

where $r \equiv 2K\delta t/(\delta x)^2$. Show that this scheme is an explicit differencing scheme and that it is computationally stable for all values of δt.

13.12. Starting from (13.54), derive (13.55).

MATLAB Exercises

M13.1. The MATLAB script **finite_diff1.m** can be used to compare the centered difference first derivative formula (13.4) for the function $\psi(x) = \sin(\pi x/4)$ to the analytic expression $d\psi/dx = (\pi/4)\cos(\pi x/4)$ for various numbers of grid intervals in the domain $-4 \le x \le 4$. Graph the maximum error as a function of number of grid intervals, *ngrid*, in the range of 4 to 64. Carry out a similar analysis for the second derivative. (Note that first-order differencing is used next to the boundaries.)

M13.2. Modify the script of M13.1 to evaluate the error in the first derivative of the function $\tanh(x)$ using the same domain in x.

M13.3. The MATLAB script **advect_1.m** demonstrates the leapfrog differencing scheme for the one-dimensional advection equation as given in (13.8). By running the script for various grid intervals, find the dependence of phase error (degrees per wave period) on the number of grid intervals per wavelength for the range 4 to 64. Compare your results with Table 13.1.

M13.4. Modify the script **advect_1.m** by substituting the upstream differencing scheme described in Problem 13.6 and the Euler backward scheme described in Problem 13.3. Compare the accuracy in phase and amplitude of these two schemes with the leapfrog scheme of M13.3.

M13.5. The MATLAB script **advect_2.m** is similar to the script of M13.3 except that the initial tracer distribution is a localized positive definite pulse of width 0.25. Run the script for 100, 200, and 400 grid intervals. Explain why the pulse changes shape as it is advected. Using the results of Problem 13.10, modify the script to provide a fourth-order accurate approximation to the advection term and compare the accuracy of the fourth-order version to that of the second-order accurate system for the case with 400 grid intervals.

M13.6. The MATLAB script **advect_3.m** is a variant of the second-order accurate script of M13.5 in which the implicit differencing of (13.19) is utilized. Run this script with $\sigma = 1$ and 1.25 and 400 grid intervals and then try running **advect_2.m** with these values of σ. Give a qualitative explanation of these results referring to Problem 13.9.

M13.7. The MATLAB script **barotropic_model.m** can be used to solve a finite difference approximation to the barotropic vorticity equation, using the flux form of the nonlinear terms given in (13.30) and leapfrog time differencing. In this example the initial flow consists of a localized vortex and a constant zonal-mean flow. Run the model for a 10-day simulation using different values of the time step to determine the maximum time step permitted to maintain numerical stability. How does this compare with the CFL criterion (13.18)?

M13.8. The MATLAB script **Lorenz_model.m** gives an accurate numerical solution to the famous three-component Lorenz equations, which are often used to demonstrate sensitive dependence on initial conditions in chaotic systems (e.g., the atmosphere). The equations for the Lorenz model can be written as

$$dX/dt = -\sigma X + \sigma Y \tag{13.75}$$

$$dY/dt = -XZ + rX - Y \tag{13.76}$$

$$dZ/dt = XY - bZ \tag{13.77}$$

Here (X, Y, Z) may be regarded as a vector defining the "climate" and σ, r, and b are constants. Run the script, letting the initial value of $X = 10$ and verify that in the $X - Z$ plane the resulting trajectory of solution points has the well-known "butterfly wings" shape. Modify the code to save and plot the time histories of the variables and compare solutions in which the initial condition on X is increased and decreased by 0.1%. How long (in nondimensional time units) do the three solutions remain within 10% of each other?

M13.9. Modify the code in **Lorenz_model.m** to include a constant forcing $F = 10$ on the right side of the equation governing dX/dt. Describe how the character of the solution changes in this case. (See Palmer, 1993.)

M13.10. Use the results of Problem 13.7 to write a staggered grid version of the MATLAB script **forced_equatorial_mode2.m** (see Problem M11.6) for the shallow water model on the equatorial β plane. Set the staggered grid such that u and Φ are defined at the equator and v is defined at points $\delta y/2$ north and south of the equator. Compare the results of the staggered grid model with those of **forced_equatorial_mode2.m** when the latter has a grid spacing half that of the former.

M13.11. The MATLAB script **nonlinear_advect_diffuse.m** provides a numerical approximation to the one-dimensional nonlinear advection–diffusion equation

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} + K\frac{\partial^2 u}{\partial x^2}$$

with initial condition $u(x, 0) = \sin(2\pi x/Lx)$. The script uses leapfrog differencing for the advective term and forward differencing for the diffusion term. In the absence of diffusion, the flow would quickly evolve to a shock, but diffusion prevents this from occurring. Run the script and by varying δt determine the maximum time step for a stable solution. Modify the code by expressing the diffusion term in the Dufort–Frankel differencing scheme given in Problem 13.11. Determine the maximum time step allowed for stability in this case. How does the accuracy of the solution change as the time step is increased?