

# Diffusive Processes

## ABSTRACT

All geophysical motions are diffusive because of turbulence. Here, we consider a relatively crude way of representing turbulent diffusion, by means of an eddy diffusivity. Although the theory is straightforward, numerical handling of diffusion terms requires care, and the main objective of this chapter is to treat the related numerical issues, leading to the fundamental concept of numerical stability.

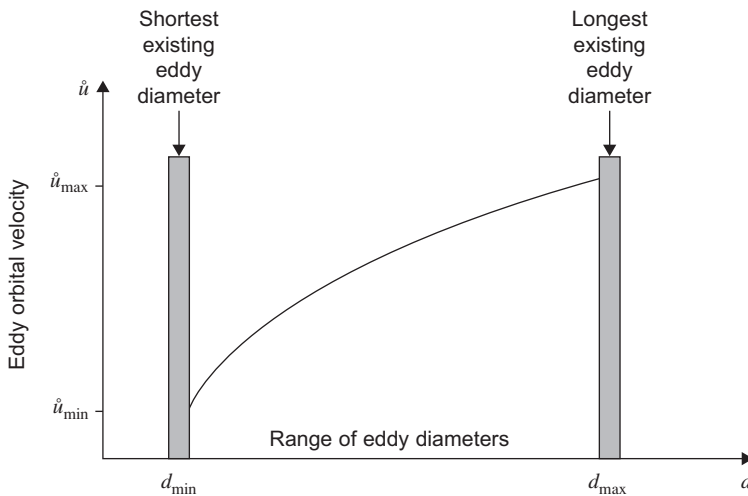
## 5.1 ISOTROPIC, HOMOGENEOUS TURBULENCE

It was mentioned in Sections 3.4 and 3.5 that fluid properties such as heat, salt, and humidity diffuse, that is, they are exchanged between neighboring particles. In laminar flow, this is accomplished by random (so-called Brownian) motion of the colliding molecules, but in large-scale geophysical systems, turbulent eddies accomplish a similar effect far more efficiently. The situation is analogous to mixing milk in coffee or tea: Left alone, the milk diffuses very slowly through the beverage, but the action of a stirrer generates turbulent eddies that mix the two liquids far more effectively and create a homogeneous mixture in a short time. The difference is that eddying in geophysical fluids is generally not induced by a stirring mechanism but is self-generated by hydrodynamic instabilities.

In Section 4.1, we introduced turbulent fluctuations without saying anything specific about them; we now begin to elucidate some of their properties. At a very basic level, turbulent motion can be interpreted as a population of many eddies (vortices), of different sizes and strengths, embedded within one another and forever changing, giving a random appearance to the flow (Fig. 5.1). Two variables then play a fundamental role:  $d$ , the characteristic diameter of the eddies and  $\hat{u}$ , their characteristic orbital velocity. Since the turbulent flow consists of many eddies, of varying sizes and speeds,  $\hat{u}$  and  $d$  do not each assume a single value but vary within a certain range. In stationary, homogeneous, and isotropic turbulence, that is, a turbulent flow that statistically appears unchanging in time, uniform in space, and without preferential direction, all eddies of a given size (same  $d$ ) behave more or less in the same way and can be



**FIGURE 5.1** Drawing of a turbulent flow by Leonardo da Vinci circa 1507–1509, who recognized that turbulence involves a multitude of eddies at various scales.

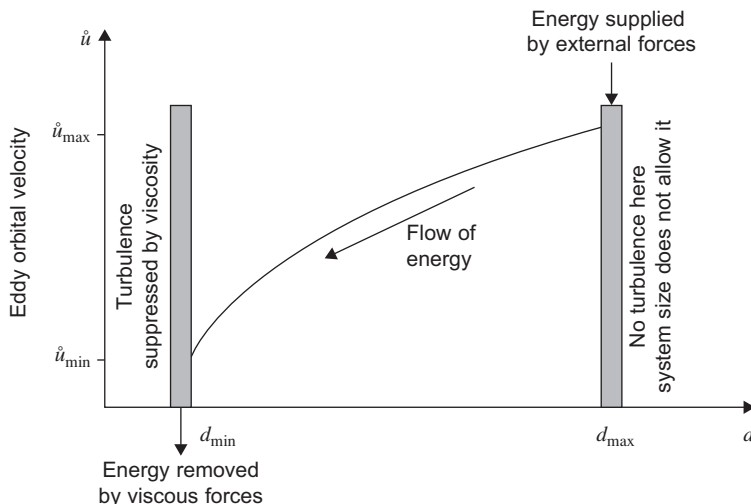


**FIGURE 5.2** Eddy orbital velocity versus eddy length scale in homogeneous and isotropic turbulence. The largest eddies have the largest orbital velocity.

assumed to share the same characteristic velocity  $\bar{u}$ . In other words, we make the assumption that  $\bar{u}$  is a function of  $d$  (Fig. 5.2).

### 5.1.1 Length and Velocity Scales

In the view of Kolmogorov (1941), turbulent motions span a wide range of scales, from a macroscale at which the energy is supplied to a microscale at which energy is dissipated by viscosity. The interaction among the eddies of



**FIGURE 5.3** The turbulent energy cascade. According to this theory, the energy fed by external forces excites the largest possible eddies and is gradually passed to over smaller eddies, all the way to a minimum scale where this energy is ultimately dissipated by viscosity.

various scales passes energy gradually from the larger eddies to the smaller ones. This process is known as the *turbulent energy cascade* (Fig. 5.3).

If the state of turbulence is statistically steady (statistically unchanging turbulence intensity), then the rate of energy transfer from one scale to the next must be the same for all scales, so that no group of eddies sharing the same scale sees its total energy level increase or decrease over time. It follows that the rate at which energy is supplied at the largest possible scale ( $d_{\max}$ ) is equal to that dissipated at the shortest scale ( $d_{\min}$ ). Let us denote by  $\epsilon$  this rate of energy supply/dissipation, per unit mass of fluid:

$$\begin{aligned}\epsilon &= \text{energy supplied to fluid per unit mass and time} \\ &= \text{energy cascading from scale to scale, per unit mass and time} \\ &= \text{energy dissipated by viscosity, per unit mass and time.}\end{aligned}$$

The dimensions of  $\epsilon$  are

$$[\epsilon] = \frac{ML^2T^{-2}}{MT} = L^2T^{-3}. \quad (5.1)$$

With Kolmogorov, we further assume that the characteristics of the turbulent eddies of scale  $d$  depend solely on  $d$  and on the energy cascade rate  $\epsilon$ . This is to say that the eddies know how large they are, at what rate energy is supplied to them, and at what rate they must supply it to the next smaller eddies in the

cascade. Mathematically,  $\dot{u}$  depends only on  $d$  and  $\epsilon$ . Since  $[\dot{u}] = LT^{-1}$ ,  $[d] = L$ , and  $[\epsilon] = L^2T^{-3}$ , the only dimensionally acceptable possibility is

$$\dot{u}(d) = A(\epsilon d)^{1/3}, \quad (5.2)$$

in which  $A$  is a dimensionless constant.

Thus, the larger  $\epsilon$ , the larger  $\dot{u}$ . This makes sense for a greater energy supply to the system that generates stronger eddies. Equation (5.2) further tells us that the smaller  $d$ , the weaker  $\dot{u}$ , and the implication is that the smallest eddies have the lowest speeds, whereas the largest eddies have the highest speeds and thus contribute most of the kinetic energy.

Typically, the largest possible eddies in the turbulent flow are those that extend across the entire system, from boundary to opposite boundary, and therefore

$$d_{\max} = L, \quad (5.3)$$

where  $L$  is the geometrical dimension of the system (such as the width of the domain or the cubic root of its volume). In geophysical flows, there is a noticeable scale disparity between the short vertical extent (depth, height) and the long horizontal extent (distance, length) of the system. We must therefore clearly distinguish eddies that rotate in the vertical plane (about a horizontal axis) from those that rotate horizontally (about a vertical axis). The nearly two-dimensional character of the latter gives rise to a special form of turbulence, called geostrophic turbulence, which will be discussed in Section 18.3. In this chapter, we restrict our attention to three-dimensional isotropic turbulence.

The shortest eddy scale is set by viscosity and can be defined as the length scale at which molecular viscosity becomes dominant. Molecular viscosity, denoted by  $\nu$ , has dimensions<sup>1</sup>:

$$[\nu] = L^2T^{-1}.$$

If we assume that  $d_{\min}$  depends only on  $\epsilon$ , the rate at which energy is supplied to that scale, and on  $\nu$  because these eddies feel viscosity, then the only dimensionally acceptable relation is

$$d_{\min} \sim \nu^{3/4} \epsilon^{-1/4}. \quad (5.4)$$

The quantity  $\nu^{3/4} \epsilon^{-1/4}$ , called the *Kolmogorov scale*, is typically on the order of a few millimeters or shorter. We leave it to the reader to verify that at this length scale, the corresponding Reynolds number is on the order of unity.

---

<sup>1</sup> Values for ambient air and water are  $\nu_{\text{air}} = 1.51 \times 10^{-5} \text{ m}^2/\text{s}$  and  $\nu_{\text{water}} = 1.01 \times 10^{-6} \text{ m}^2/\text{s}$ .

The span of length scales in a turbulent flow is related to its Reynolds number. Indeed, in terms of the largest velocity scale, which is the orbital velocity of the largest eddies,  $U = \dot{u}(d_{\max}) = A(\epsilon L)^{1/3}$ , the energy supply/dissipation rate is

$$\epsilon = \frac{U^3}{A^3 L} \sim \frac{U^3}{L}, \quad (5.5)$$

and the length scale ratio can be expressed as

$$\begin{aligned} \frac{L}{d_{\min}} &\sim \frac{L}{\nu^{3/4} \epsilon^{-1/4}} \\ &\sim \frac{LU^{3/4}}{\nu^{3/4} L^{1/4}} \\ &\sim Re^{3/4}, \end{aligned} \quad (5.6)$$

where  $Re = UL/\nu$  is the Reynolds number of the flow. As we could have expected, a flow with a higher Reynolds number contains a broader range of eddies.

### 5.1.2 Energy Spectrum

In turbulence theory, it is customary to consider the so-called *power spectrum*, which is the distribution of kinetic energy per mass across the various length scales. For this, we need to define a wavenumber. Because velocity reverses across the diameter of an eddy, the eddy diameter should properly be considered as half of the wavelength:

$$k = \frac{2\pi}{\text{wavelength}} = \frac{\pi}{d}. \quad (5.7)$$

The lowest and highest wavenumber values are  $k_{\min} = \pi/L$  and  $k_{\max} \sim \epsilon^{1/4} \nu^{-3/4}$ .

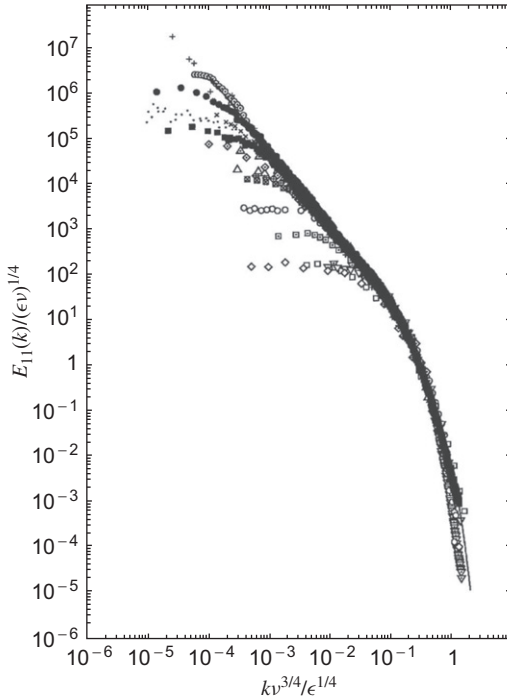
The kinetic energy  $E$  per mass of fluid has dimensions  $ML^2T^{-2}/M = L^2T^{-2}$ . The portion  $dE$  contained in the eddies with wavenumbers ranging from  $k$  to  $k + dk$  is defined as

$$dE = E_k(k) dk.$$

It follows that the dimension of  $E_k$  is  $L^3T^{-2}$ , and dimensional analysis prescribes

$$E_k(k) = B \epsilon^{2/3} k^{-5/3}, \quad (5.8)$$

where  $B$  is a second dimensionless constant. It can be related to  $A$  of Eq. (5.2) because the integration of  $E_k(k)$  from  $k_{\min} = \pi/L$  to  $k_{\max} \sim \infty$  is the total energy



**FIGURE 5.4** Longitudinal power spectrum of turbulence calculated from numerous observations taken outdoors and in the laboratory. (From Saddoughi and Veeravalli, 1994).

per mass in the system, which in good approximation is that contained in the largest eddies, namely  $U^2/2$ . Thus,

$$\int_{k_{\min}}^{\infty} E_k(k) dk = \frac{U^2}{2}, \quad (5.9)$$

from which follows

$$\frac{3}{2\pi^{2/3}} B = \frac{1}{2} A^2. \quad (5.10)$$

The value of  $B$  has been determined experimentally and found to be about 1.5 (Pope, 2000, page 231). From this, we estimate  $A$  to be 1.45.

The  $-5/3$  power law of the energy spectrum has been observed to hold well in the *inertial range*, that is, for those intermediate eddy diameters that are remote from both largest and shortest scales. Figure 5.4 shows the superposition of a large number of longitudinal power spectra.<sup>2</sup> The straight line where most data overlap in the range  $10^{-4} < kv^{3/4}/\epsilon^{1/4} < 10^{-1}$  corresponds to the  $-5/3$

<sup>2</sup>The longitudinal power spectrum is the spectrum of the kinetic energy associated with the velocity component in the direction of the wavenumber.

decay law predicted by the Kolmogorov turbulent cascade theory. The higher the Reynolds number of the flow, the broader the span of wavenumbers over which the  $-5/3$  law holds. Several crosses visible at the top of the plot, which extend from a set of crosses buried in the accumulation of data below, correspond to data in a tidal channel (Grant, Stewart & Moilliet, 1962) for which the Reynolds number was the highest.

There is, however, some controversy over the  $-5/3$  power law for  $E_k$ . Some investigators (Long, 1997, 2003; Saffman, 1968) have proposed alternative theories that predict a  $-2$  power law.

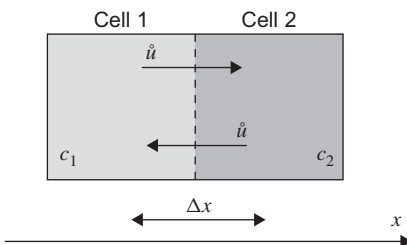
## 5.2 TURBULENT DIFFUSION

Our concern here is not to pursue the study of turbulence but to arrive at a heuristic way to represent the dispersive effect of turbulence on those scales too short to be resolved in a numerical model.

*Turbulent diffusion* or *dispersion* is the process by which a substance is moved from one place to another under the action of random turbulent fluctuations in the flow. Given the complex nature of these fluctuations, it is impossible to describe the dispersion process in an exact manner but some general remarks can be made that lead to a useful parameterization.

Consider the two adjacent cells of Fig. 5.5 exchanging fluid between each other. The fluid in the left cell contains a concentration (mass per volume)  $c_1$  of some substance, whereas the fluid in the right cell contains a different concentration  $c_2$ . Think of  $c_1$  being less than  $c_2$ , although this does not necessarily have to be the case. Further assume, in order to focus exclusively on diffusion, that there is no net flow from one cell to the other but that the only exchange velocity is due to a single eddy moving fluid at velocity  $\hat{u}$  on one flank and at velocity  $-\hat{u}$  on its opposite flank. The amount of substance carried per unit area perpendicular to the  $x$ -axis and per time, called the *flux*, is equal to the product of the concentration and the velocity,  $c_1\hat{u}$  from left to right and  $c_2\hat{u}$  in the opposite direction. The net flux  $q$  in the  $x$ -direction is the flux from 1 to 2 minus the flux from 2 to 1:

$$\begin{aligned} q &= c_1 \hat{u} - c_2 \hat{u} \\ &= -\hat{u} \Delta c, \end{aligned}$$



**FIGURE 5.5** Exchange between two adjacent cells illustrating turbulent diffusion. Because of the difference between concentrations, the exchange between cells is uneven. The cell with the least concentration loses less than it receives.

where  $\Delta c = c_2 - c_1$  is the concentration difference. Multiplying and dividing by the distance  $\Delta x$  between cell centers, we may write:

$$q = -(\dot{u}\Delta x) \frac{\Delta c}{\Delta x}.$$

When considering the variation of  $c$  over larger scales, those for which the eddy-size  $\Delta x$  appears to be small, we may approximate the previous equation to

$$q = -D \frac{dc}{dx}, \quad (5.11)$$

where  $D$  is equal to the product  $\dot{u}\Delta x$  and is called the turbulent diffusion coefficient or *diffusivity*. Its dimension is  $[D] = L^2 T^{-1}$ .

The diffusive flux is proportional to the gradient of the concentration of the substance. In retrospect, this makes sense; if there were no difference in concentrations between cells, the flux from one into the other would be exactly compensated by the flux in the opposite direction. It is the concentration difference (the gradient) that matters.

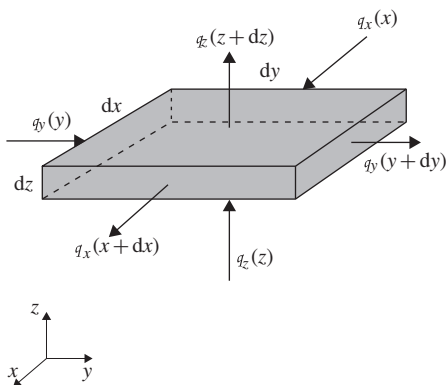
Diffusion is “down-gradient,” that is, the transport is from high to low concentrations, just as heat conduction moves heat from the warmer side to the colder side. (In the preceding example with  $c_1 < c_2$ ,  $q$  is negative, and the net flux is from cell 2 to cell 1.) This implies that the concentration increases on the low side and decreases on the high side, and the two concentrations gradually become closer to each other. Once they are equal ( $dc/dx = 0$ ), diffusion stops, although turbulent fluctuations never do. Diffusion acts to homogenize the substance across the system.

The pace at which diffusion proceeds depends critically on the value of the diffusion coefficient  $D$ . This coefficient is inherently the product of two quantities, a velocity ( $\dot{u}$ ) and a length scale ( $\Delta x$ ), representing respectively the magnitude of fluctuating motions and their range. Since the numerical model resolves scales down to the grid scale  $\Delta x$ , the turbulent diffusion that remains to represent is that due to the all shorter scales, starting with  $d = \Delta x$ . As seen in the previous section, for shorter scales,  $d$  correspond to slower eddy velocities  $\dot{u}$  and thus lower diffusivities. It follows that diffusion is chiefly accomplished by eddies at the largest unresolved scale,  $\Delta x$ , because these generate the greatest value of  $\dot{u}\Delta x$ :

$$\begin{aligned} D &= \dot{u}(\Delta x) \Delta x \\ &= A \epsilon^{1/3} \Delta x^{4/3}. \end{aligned} \quad (5.12)$$

The manner by which the dissipation rate  $\epsilon$  is related to local flow characteristics, such as a velocity gradient, opens the way to a multitude of possible parameterizations.





**FIGURE 5.6** An infinitesimal piece of fluid for the local budget of a substance of concentration  $c$  in the fluid.

The preceding considerations in one dimension were generic in the sense that the direction  $x$  could stand for any of the three directions of space,  $x$ ,  $y$ , or  $z$ . Because of the typical disparity in mesh size between the horizontal and vertical directions in GFD models ( $\Delta x \approx \Delta y \gg \Delta z$ ), care must be taken to use two distinct diffusivities, which we denote  $\mathcal{A}$  for the horizontal directions and  $\kappa$  for the vertical direction.<sup>3</sup> While  $\kappa$  must be constructed from the length scale  $\Delta z$ ,  $\mathcal{A}$  must be formed from a length scale that is hybrid between  $\Delta x$  and  $\Delta y$ . The Smagorinsky formulation presented in (4.10) is a good example.

The components of the three-dimensional flux vector are

$$q_x = -\mathcal{A} \frac{\partial c}{\partial x} \quad (5.13a)$$

$$q_y = -\mathcal{A} \frac{\partial c}{\partial y} \quad (5.13b)$$

$$q_z = -\kappa \frac{\partial c}{\partial z}. \quad (5.13c)$$

And, we are in a position to write a budget for the concentration  $c(x, y, z, t)$  of the substance in the flow, by taking an elementary volume of fluid of size  $dx$ ,  $dy$ , and  $dz$ , as illustrated in Fig. 5.6. The net import in the  $x$ -direction is the difference in  $x$ -fluxes times the area  $dy \, dz$  they cross, that is,  $[q_x(x, y, z) - q_x(x + dx, y, z)] \, dy \, dz$ , and similarly in the  $y$ - and  $z$ -directions. The net import from all directions is then

$$\begin{aligned} \text{Net import in } dx \, dy \, dz = & [q_x(x, y, z) - q_x(x + dx, y, z)] \, dy \, dz \\ & + [q_y(x, y, z) - q_y(x, y + dy, z)] \, dx \, dz \\ & + [q_z(x, y, z) - q_z(x, y, z + dz)] \, dx \, dy, \end{aligned}$$

<sup>3</sup>GFD models generally use the same horizontal diffusivity for all variables, including momentum and density—see (4.21)—but distinguish between various diffusivities in the vertical.

on a per-time basis. This net import contributes to increasing the amount  $c \, dx \, dy \, dz$  inside the volume:

$$\frac{d}{dt}(c \, dx \, dy \, dz) = \text{Net import.}$$

In the limit of an infinitesimal volume (vanishing  $dx$ ,  $dy$  and  $dz$ ), we have

$$\frac{\partial c}{\partial t} = -\frac{\partial q_x}{\partial x} - \frac{\partial q_y}{\partial y} - \frac{\partial q_z}{\partial z}, \quad (5.14)$$

and, after replacement of the flux components by their expressions (5.13),

$$\frac{\partial c}{\partial t} = \frac{\partial}{\partial x} \left( \mathcal{A} \frac{\partial c}{\partial x} \right) + \frac{\partial}{\partial y} \left( \mathcal{A} \frac{\partial c}{\partial y} \right) + \frac{\partial}{\partial z} \left( \kappa \frac{\partial c}{\partial z} \right), \quad (5.15)$$

where  $\mathcal{A}$  and  $\kappa$  are, respectively, the horizontal and vertical eddy diffusivities. Note the similarity with the dissipation terms in the momentum and energy equations (4.21) of the previous chapter.

For a comprehensive exposition of diffusion and some of its applications, the reader is referred to Ito (1992) and Okubo and Levin (2002).

### 5.3 ONE-DIMENSIONAL NUMERICAL SCHEME

We now illustrate discretization methods for the diffusion equation and begin with a prototypical one-dimensional system, representing a horizontally homogeneous piece of ocean or atmosphere, containing a certain substance, such as a pollutant or tracer, which is not exchanged across either bottom or top boundaries. To simplify the analysis further, we begin by taking the vertical diffusivity  $\kappa$  as constant until further notice. We then have to solve the following equation:

$$\frac{\partial c}{\partial t} = \kappa \frac{\partial^2 c}{\partial z^2}, \quad (5.16)$$

with no-flux boundary conditions at both bottom and top:

$$q_z = -\kappa \frac{\partial c}{\partial z} = 0 \quad \text{at} \quad z = 0 \text{ and } z = h, \quad (5.17)$$

where  $h$  is the thickness of the domain.

To complete the problem, we also prescribe an initial condition. Suppose for now that this initial condition is a constant  $C_0$  plus a cosine function of amplitude  $C_1$  ( $C_1 \leq C_0$ ):

$$c(z, t=0) = C_0 + C_1 \cos\left(j\pi \frac{z}{h}\right), \quad (5.18)$$

with  $j$  being an integer. Then, it is easily verified that

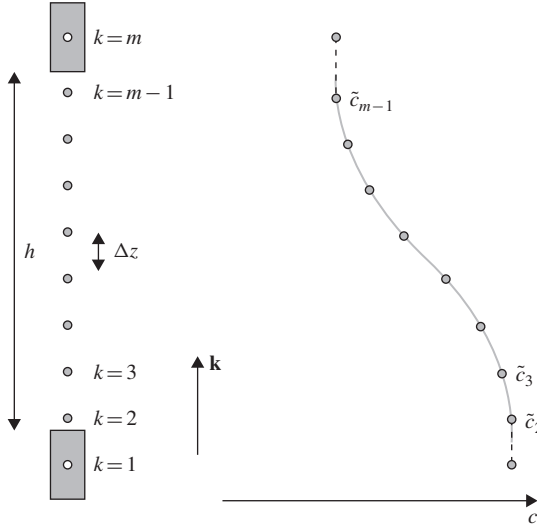
$$c = C_0 + C_1 \cos\left(j\pi \frac{z}{h}\right) \exp\left(-j^2 \pi^2 \frac{\kappa t}{h^2}\right) \quad (5.19)$$

satisfies the partial differential equation (5.16), both boundary conditions (5.17), and initial condition (5.18). It is thus the exact solution of the problem. As we can expect from the dissipative nature of diffusion, this solution represents a temporal attenuation of the nonuniform portion of  $c$ , which is more rapid under stronger diffusion (greater  $\kappa$ ) and shorter scales (higher  $j$ ).

Let us now design a numerical method to solve the problem and check its solution against the preceding, exact solution. First, we discretize the spatial derivative by applying a standard finite-difference technique. With a Neumann boundary condition applied at each end, we locate the end grid points not at, but surrounding the boundaries (see Section 4.7) and place the grid nodes at the following locations:

$$z_k = \left(k - \frac{3}{2}\right) \Delta z \quad \text{for } k = 1, 2, \dots, m, \quad (5.20)$$

with  $\Delta z = h/(m-2)$  so that we use  $m$  grid points, among which the first and last are ghost points lying a distance  $\Delta z/2$  beyond the boundaries (Fig. 5.7).



**FIGURE 5.7** Gridding of a vertical domain with  $m$  nodes, of which the first and last lie beyond the bottom and top boundaries, respectively. Such points are called *ghost points*. With  $m$  nodes and  $m-1$  intervals between nodes among which two are only half long, it follows that  $(m-2)$  segments cover the domain, and the grid spacing is thus  $\Delta z = h/(m-2)$ . Neumann conditions (zero derivatives) at both boundaries are implemented by assigning the values  $\tilde{c}_1 = \tilde{c}_2$  and  $\tilde{c}_m = \tilde{c}_{m-1}$  to the end points, which implies zero derivatives in the middle of the first and last intervals. The calculations using the discretized form of the equation then proceed from  $k=2$  to  $k=m-1$ .

Discretizing the second spatial derivative with a three-point centered scheme and before performing time discretization, we have

$$\frac{d\tilde{c}_k}{dt} = \frac{\kappa}{\Delta z^2} (\tilde{c}_{k+1} - 2\tilde{c}_k + \tilde{c}_{k-1}) \quad \text{for } k=2, \dots, m-1. \quad (5.21)$$

We thus have  $m-2$  ordinary, coupled, differential equations for the  $m-2$  unknown time dependent functions  $\tilde{c}_k$ . We can determine the numerical error introduced in this *semidiscrete* set of equations by trying a solution similar to the exact solution:

$$\tilde{c}_k = C_0 + C_1 \cos\left(j\pi \frac{z_k}{h}\right) a(t). \quad (5.22)$$

Trigonometric formulas provide the following equation for the temporal evolution of the amplitude  $a(t)$ :

$$\frac{da}{dt} = -4a \frac{\kappa}{\Delta z^2} \sin^2 \phi \quad \text{with } \phi = j\pi \frac{\Delta z}{2h}, \quad (5.23)$$

of which the solution is

$$a(t) = \exp\left(-4 \sin^2 \phi \frac{\kappa t}{\Delta z^2}\right). \quad (5.24)$$

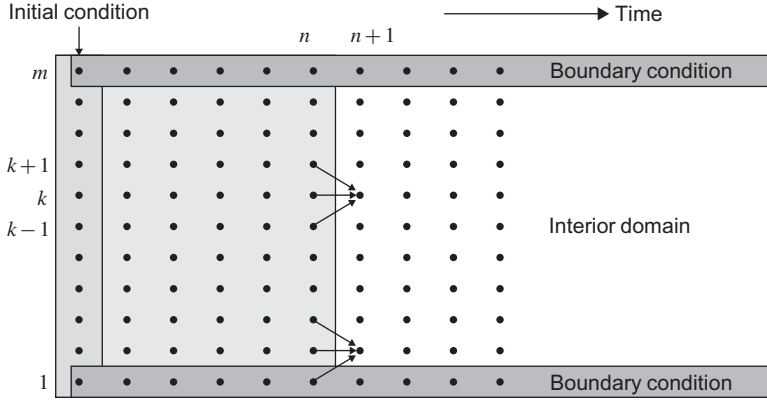
With this spatial discretization, we thus obtain an exponential decrease of amplitude  $a$ , like in the exact [equation \(5.19\)](#) but with a different damping rate. The ratio  $\tau$  of the numerical damping rate  $4\kappa \sin^2 \phi / \Delta z^2$  to the true damping rate  $j^2 \pi^2 \kappa / h^2$  is  $\tau = \phi^{-2} \sin^2 \phi$ . For small  $\Delta z$  compared with the length scale  $h/j$  of the  $c$  distribution,  $\phi$  is small, and the correct damping is nearly obtained with the semidiscrete numerical scheme. Nothing anomalous is therefore expected from the approach thus far as long as the discretization of the domain is sufficiently dense to capture adequately the spatial variations in  $c$ . Also, the boundary conditions cause no problem because the mathematical requirement of one boundary condition on each side of the domain matches exactly what we need to calculate the discrete values  $\tilde{c}_k$  for  $k=2, \dots, m-1$ . An initial condition is also needed at each node to start the time integration. This is all consistent with the mathematical problem.

We now proceed with the time discretization. First, let us try the simplest of all methods, the explicit Euler scheme:

$$\frac{\tilde{c}_k^{n+1} - \tilde{c}_k^n}{\Delta t} = \frac{\kappa}{\Delta z^2} (\tilde{c}_{k+1}^n - 2\tilde{c}_k^n + \tilde{c}_{k-1}^n) \quad \text{for } k=2, \dots, m-1 \quad (5.25)$$

in which  $n \geq 1$  stands for the time level. For convenience, we define a dimensionless number that will play a central role in the discretization and solution:

$$D = \frac{\kappa \Delta t}{\Delta z^2}. \quad (5.26)$$



**FIGURE 5.8** Initialized for each grid point, algorithm (5.27) advances the value at node  $k$  to the next time step (from  $n$  to  $n+1$ ) using the previous values on stencil spanning points  $k-1$ ,  $k$ , and  $k+1$ . A boundary condition is thus needed on each side of the domain, as the original mathematical problem requires. The calculations for the discretized governing equations proceed from  $k=2$  to  $k=m-1$ .

This definition allows us to write the discretized equation more conveniently as

$$\tilde{c}_k^{n+1} = \tilde{c}_k^n + D(\tilde{c}_{k+1}^n - 2\tilde{c}_k^n + \tilde{c}_{k-1}^n) \quad \text{for } k=2, \dots, m-1. \quad (5.27)$$

The scheme updates the discrete  $\tilde{c}_k$  values from their initial values and with the aforementioned boundary conditions (Fig. 5.8). Obviously, the algorithm is easily programmed (e.g., `firstdiffusion.m`) and can be tested rapidly.

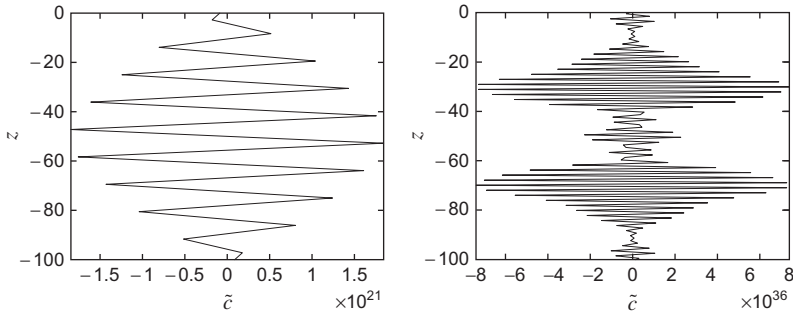
For simplicity, we start with a gentle profile ( $j=1$ , half a wavelength across the domain) and, equipped with our insight in scale analysis, we use a sufficiently small grid spacing  $\Delta z \ll h$  to resolve the cosine function well. To be sure, we take 20 grid points. For the time scale  $T$  of the physical process, we use the scale provided by the original equation:

$$\frac{\partial c}{\partial t} = \kappa \frac{\partial^2 c}{\partial z^2}$$

$$\frac{\Delta c}{T} \approx \kappa \frac{\Delta c}{h^2}$$

to find  $T = h^2/\kappa$ . Dividing this time scale in 20 steps, we take  $\Delta t = T/20 = h^2/(20\kappa)$  and begin to march algorithm (5.27) forward.

Surprisingly, it is not working. After only 20 time steps, the  $\tilde{c}_k$  values do not show attenuation but have instead increased by a factor  $10^{20}$ ! Furthermore, increasing the spatial resolution to 100 points and reducing the time step proportionally does not help but worsens the situation (Fig. 5.9). Yet, there has been no programming error in `firstdiffusion.m`. The problem is more serious: We have stumbled on a crucial aspect of numerical integration, by falling prey to *numerical instability*. The symptoms of numerical instability are explosive



**FIGURE 5.9** Profile of  $\tilde{c}$  after 20 time steps of the Euler scheme (5.27). Left panel: 20 grid points and  $\Delta t = T/20$ . Right panel: 100 grid points and  $\Delta t = T/100$ . Note the vast difference in values between the two solutions ( $10^{21}$  and  $10^{36}$ , respectively), the second solution being much more explosive than the first. Conclusion: Increasing resolution worsens the problem.

behavior and worsening of the problem with increased spatial resolution. At best, the scheme is used outside of a certain domain of validity or, at worst, it is hopeless and in need of replacement by a better, stable scheme. What makes a scheme stable and another unstable is the objective of numerical stability analysis.

## 5.4 NUMERICAL STABILITY ANALYSIS

The most widely used method to investigate the stability of a given numerical scheme is due to John von Neumann.<sup>4</sup> The basic idea of the method is to consider the temporal evolution of simple numerical solutions. As continuous signals and distributions can be expressed as Fourier series of sines and cosines, discrete functions can, too, be decomposed in elementary functions. If one or several of these elementary functions increase without bound over time (“explode”), the reconstructed solution, too, will increase without bound, and the scheme is unstable. Put the other way: A scheme is stable if none among all elementary functions grows without bound over time.

As for Fourier series and simple wave propagation, the elementary functions are periodic. In analogy with the continuous function

$$c(z, t) = A e^{i(k_z z - \omega t)}, \quad (5.28)$$

we use the discrete function  $\tilde{c}_k^n$  formed by replacing  $z$  by  $k\Delta z$  and  $t$  by  $n\Delta t$ :

$$\tilde{c}_k^n = A e^{i(k_z k\Delta z - \omega n\Delta t)}, \quad (5.29)$$

<sup>4</sup>See biography at the end of this chapter.

where  $k_z$  is a vertical wavenumber and  $\omega$  a frequency. To consider periodic behavior in space and possibly explosive behavior in time,  $k_z$  is restricted to be real positive, whereas  $\omega = \omega_r + i\omega_i$  may be complex. Growth without bound occurs if  $\omega_i > 0$ . (If  $\omega_i < 0$ , the function decreases exponentially and raises no concern). The origins of  $z$  and  $t$  do not matter, for they can be adjusted by changing the complex amplitude  $A$ .

The range of  $k_z$  values is restricted. The lowest value is  $k_z = 0$ , corresponding to the constant component in Eq. (5.22). At the other extreme, the shortest wave is the “2 $\Delta x$  mode” or “saw-tooth” (+1, −1, +1, −1, etc.) with  $k_z = \pi/\Delta z$ . It is most often with this last value that trouble occurs, as seen in the rapidly oscillating values generated by the ill-fated Euler scheme (Fig. 5.9) and, earlier, aliasing (Section 1.12).

The elementary function, or trial solution, can be recast in the following form to distinguish the temporal growth (or decay) from the propagating part:

$$\tilde{c}_k^n = A e^{+\omega_i \Delta t n} e^{i(k_z \Delta z k - \omega_r \Delta t n)}. \quad (5.30)$$

An alternative way of expressing the elementary function is by introducing a complex number  $\varrho$  called the *amplification factor* such that

$$\tilde{c}_k^n = A \varrho^n e^{i(k_z \Delta z)k} \quad (5.31a)$$

$$\varrho = |\varrho| e^{i \arg(\varrho)} \quad (5.31b)$$

$$\omega_i = \frac{1}{\Delta t} \ln |\varrho|, \quad \omega_r = -\frac{1}{\Delta t} \arg(\varrho). \quad (5.31c)$$

The choice of expression among Eqs. (5.29), (5.30), and (5.31a) is a matter of ease and convenience.

Stability requires a nongrowing numerical solution, with  $\omega_i \leq 0$  or equivalently  $|\varrho| \leq 1$ . Allowing for *physical* exponential growth—such as the growth of a physically unstable wave—we should entertain the possibility that  $c(t)$  may grow as  $\exp(\omega_i t)$ , in which case  $c(t + \Delta t) = c(t) \exp(\omega_i \Delta t) = c(t)[1 + \mathcal{O}(\Delta t)]$  and  $\varrho = 1 + \mathcal{O}(\Delta t)$ . In other words, instead of  $|\varrho| \leq 1$ , we should adopt the slightly less demanding criterion

$$|\varrho| \leq 1 + \mathcal{O}(\Delta t). \quad (5.32)$$

Since there is no exponential growth associated with diffusion, the criterion  $|\varrho| \leq 1$  applies here.

We can now try Eq. (5.31a) as a solution of the discretized diffusion equation (5.27). After division by the factor  $A \varrho^n \exp[i(k_z \Delta z)k]$  common to all terms, the discretized equation reduces to

$$\varrho = 1 + D \left[ e^{+i k_z \Delta z} - 2 + e^{-i k_z \Delta z} \right], \quad (5.33)$$

which is satisfied when the amplification factor equals

$$\begin{aligned}\varrho &= 1 - 2D[1 - \cos(k_z \Delta z)] \\ &= 1 - 4D \sin^2\left(\frac{k_z \Delta z}{2}\right).\end{aligned}\quad (5.34)$$

Since in this case  $\varrho$  happens to be real, the stability criterion stipulates  $-1 \leq \varrho \leq 1$ , that is,  $4D \sin^2(k_z \Delta z/2) \leq 2$ , for all possible  $k_z$  values. The most dangerous value of  $k_z$  is the one that makes  $\sin^2(k_z \Delta z/2) = 1$ , which is  $k_z = \pi/\Delta z$ , the wavenumber of the saw-tooth mode. For this mode,  $\varrho$  violates  $-1 \leq \varrho$  unless

$$D = \frac{\kappa \Delta t}{\Delta z^2} \leq \frac{1}{2}. \quad (5.35)$$

In other words, the Euler scheme is stable only if the time step is shorter than  $\Delta z^2/(2\kappa)$ . We are in the presence of *conditional stability*, and Eq. (5.35) is called the *stability condition* of the scheme.

Generally, criterion (5.35) or a similar one in another case is neither necessary nor sufficient since it neglects any effect due to boundary conditions, which can either stabilize an unstable mode or destabilize a stable one. In most situations, however, the criterion obtained by this method turns out to be a necessary condition since it is unlikely that in the middle of the domain, boundaries could stabilize an unstable solution, especially the shorter waves that are most prone to instability. On the other hand, boundaries can occasionally destabilize a stable mode in their vicinity. For the preceding scheme applied to the diffusion equation, this is not the case, and criterion (5.35) is both necessary and sufficient.

In addition to stability information, the amplification-factor method also enables a comparison between a numerical property and its physical counterpart. In the case of the diffusion equation, it is the damping rate, but, should the initial equation have described wave propagation, it would have been the dispersion relation. The general solution (5.19) of the exact equation (5.16) leads to the relation

$$\omega_i = -\kappa k_z^2, \quad (5.36)$$

which we can compare with the numerical damping rate

$$\begin{aligned}\tilde{\omega}_i &= \frac{1}{\Delta t} \ln |\varrho| \\ &= \frac{1}{\Delta t} \ln \left| 1 - 4D \sin^2\left(\frac{k_z \Delta z}{2}\right) \right|.\end{aligned}\quad (5.37)$$



The ratio  $\tau$  of the numerical damping to the actual damping rate is then given by

$$\tau = \frac{\tilde{\omega}_i}{\omega_i} = -\frac{\ln |1 - 4D \sin^2(k_z \Delta z/2)|}{Dk_z^2 \Delta z^2}, \quad (5.38)$$

which for small  $k_z \Delta z$ , that is, numerically well-resolved modes, behaves as

$$\tau = 1 + \left(2D - \frac{1}{3}\right) \left(\frac{k_z \Delta z}{2}\right)^2 + \mathcal{O}(k_z^4 \Delta z^4). \quad (5.39)$$

For  $D < 1/6$ , the numerical scheme dampens less fast than the physical process ( $\tau < 1$ ), whereas for larger values  $1/6 < D < 1/2$  (i.e., relatively large but still stable time steps), overdamping occurs ( $\tau > 1$ ). In practice, when  $D > 1/4$  (leading to  $\varrho < 0$  for the higher  $k_z$  values), this overdamping can be unrealistically large and unphysical. The shortest wave resolved by the spatial grid with  $k_z \Delta z = \pi$  exhibits not only a saw-tooth pattern in space (as it should) but also a flip-flop behavior in time. This is because, for real negative  $\varrho$ , the sequence  $\varrho^1, \varrho^2, \varrho^3, \dots$  alternates in sign. For  $-1 < \varrho < 0$ , the solution vanishes not by monotonically decreasing toward zero but instead by oscillating around zero. Though the scheme is stable, the numerical solution behaves unlike the exact solution, and this should be avoided. It is therefore prudent to keep  $D \leq 1/4$  to guarantee a realistic solution.

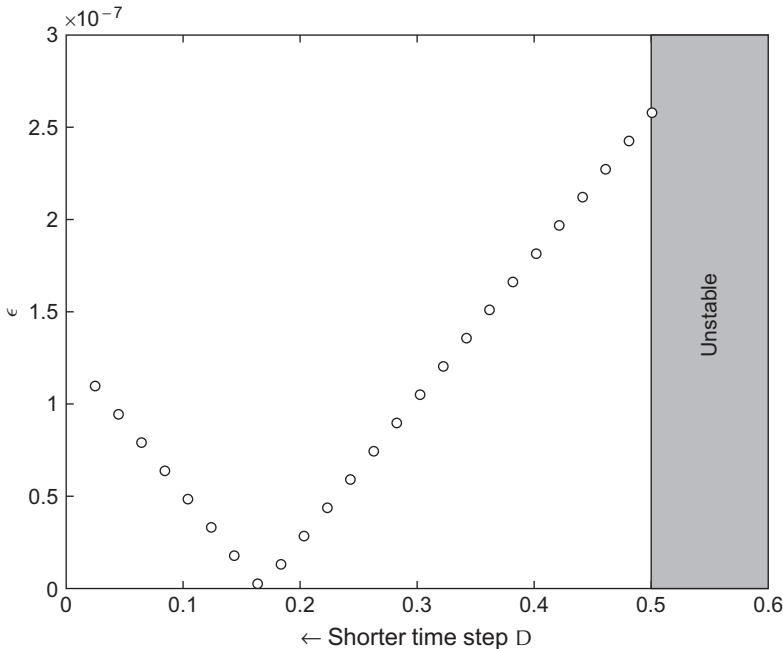
Let us now give a physical interpretation of the stability condition  $2\Delta t \leq \Delta z^2/\kappa$ . First, we observe that the instability appears most strongly for the component with the largest wavenumber according to Eq. (5.34). Since the length scale of this signal is  $\Delta z$ , the associated diffusion time scale is  $\Delta z^2/\kappa$ , and the stability criterion expresses the requirement that  $\Delta t$  be set shorter than a fraction of this time scale. It is equivalent to ensuring that the time step provides an adequate representation of the *shortest* component resolved by the spatial grid. Even when this shortest component is absent from the mathematical solution (in our initial problem only a single length scale,  $h$ , was present), it does occur in the numerical solution because of computer round-off errors, and stability is thus conditioned by the *possible* presence of the shortest resolved component. The stability condition ensures that all possible solution components are treated with an adequate time step.

As the preceding simple example shows, the amplification-factor method is easily applied and provides a stability condition, as well as other properties of the numerical solution. In practice, however, nonconstant coefficients (such as a spatially variable diffusivity  $\kappa$ ) or nonuniform spacing of grid points may render its application difficult. Since nonuniform grids may be interpreted as a coordinate transformation, stretching and compressing grid node positions (see also Section 20.6.1), a nonuniform grid is equivalent to introducing nonconstant coefficients into the equation. The procedure is to “freeze” the coefficients at some value before applying the amplification-factor method and then repeat

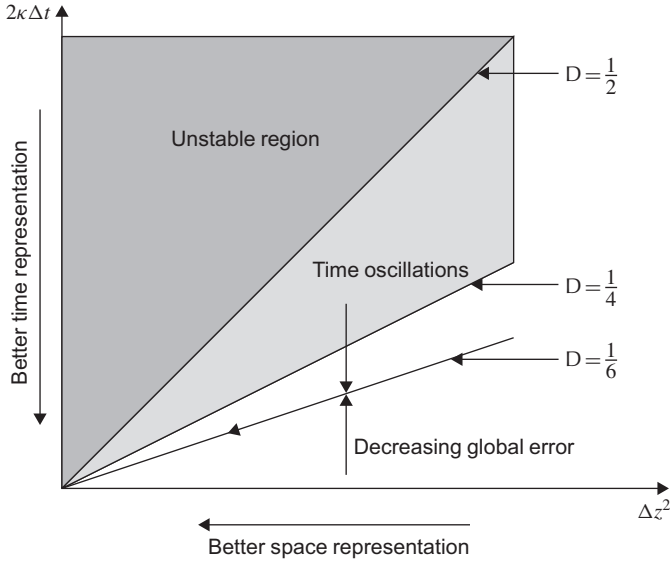
the analysis with different frozen values within the allotted ranges. Generally, this provides quite accurate estimates of permissible time steps. For nonlinear problems, the approach is to perform a preliminary linearization of the equation, but the quality of the stability condition is not always reliable. Finally, it is important to remember that the amplification-factor method does not deal with boundary conditions. To treat accurately cases with variable coefficients and nonuniform grids and to take boundary conditions into account, the so-called *matrix method* is available (e.g., Kreiss, 1962; Richtmyer & Morton, 1967).

We now have some tools to guarantee stability. Since our diffusion scheme is also consistent, we anticipate convergence by virtue of the Lax–Richtmyer Theorem (see Section 2.7). Let us then verify numerically whether the scheme leads to a linear decrease of the error with decreasing time step. Leaning on the exact solution (5.19) for comparison, we observe (Fig. 5.10) that the numerical solution does indeed exhibit a decrease of the error with decreasing time step, but only up to a point (for  $D$  decreasing from the stability limit of 0.5 to  $1/6$ ). The error increases again for smaller  $\Delta t$ . What happens?

The fact is that two sources of errors (space and time discretization) are simultaneously present and what we are measuring is the *sum* of these errors,



**FIGURE 5.10** Root mean square of error  $c - \tilde{c}$  scaled by the initial variation  $\Delta c$  at time  $T = h^2/\kappa$  for a fixed space grid ( $m=50$ ) and decreasing time step (going from right to left). Above  $D=1/2$ , the scheme is unstable and the error is extremely large (not plotted). For shorter time steps, the scheme is stable and the error first decreases linearly with  $D$ . Below  $D=1/6=0.167$ , the error increases again.



**FIGURE 5.11** Different paths to convergence in the  $(\Delta z^2, \Delta t)$  plane for the explicit scheme. For excessive values of  $\Delta t$ ,  $D \geq 1/2$ , the scheme is unstable. Convergence can only be obtained by remaining within the stability region. When  $\Delta t$  alone is reduced (progressing vertically downward in the graph), the error decreases and then increases again. If  $\Delta z$  alone is decreased (progressing horizontally to the left in the graph), the error similarly decreases first and then increases, until the scheme becomes unstable. Reducing both  $\Delta t$  and  $\Delta z$  simultaneously at fixed  $D$  within the stability sector leads to monotonic convergence. The convergence rate is highest along the line  $D = 1/6$  because the scheme then happens to be fourth-order accurate.

not the temporal error in isolation. This can be shown by looking at the modified equation obtained by using a Taylor-series expansion of discrete values  $\tilde{c}_{k+1}^n$  etc. around  $\tilde{c}_k^n$  in the difference equation (5.21). Some algebra leads to

$$\frac{\partial \tilde{c}}{\partial t} = \kappa \frac{\partial^2 \tilde{c}}{\partial z^2} + \frac{\kappa \Delta z^2 (1 - 6D)}{12} \frac{\partial^4 \tilde{c}}{\partial z^4} + \mathcal{O}(\Delta t^2, \Delta z^4, \Delta t \Delta z^2), \quad (5.40)$$

which shows that the scheme is of first order in time (through  $D$ ) and second order in space. The rebounding error exhibited in Fig. 5.10 when  $\Delta t$  is gradually reduced (changing  $D$  alone) is readily explained in view of Eq. (5.40).

To check on convergence, we should consider the case when both parameters  $\Delta t$  and  $\Delta z$  are reduced simultaneously (Fig. 5.11). This is most naturally performed by keeping fixed the stability parameter  $D$ , which is a combination of both according to Eq. (5.26). The leading error (second term on the right) decreases as  $\Delta z^2$ , except when  $D = 1/6$  in which case the scheme is then of fourth order. It can be shown<sup>5</sup> that in that case the error is on the order of  $\Delta z^4$ .

<sup>5</sup>To show this, consider that for  $D = 1/6$ ,  $\Delta t = \Delta z^2 / 6\kappa$  and all contributions to the error term become proportional to  $\Delta z^4$ .

This is consistent with Eq. (5.39), where the least error on the damping rate is obtained with  $2D = 1/3$ , that is,  $D = 1/6$ , and with Fig. 5.10, where the error for fixed  $\Delta z$  is smallest when the time step corresponds to  $D = 1/6$ .

## 5.5 OTHER ONE-DIMENSIONAL SCHEMES

A disadvantage of the simple scheme (5.25) is its fast increase in cost when a higher spatial resolution is sought. For stability reasons  $\Delta t$  decreases as  $\Delta z^2$ , forcing us to calculate values at not only more grid points but also more frequently. For integration over a fixed length of time, the number of calculations grows as  $m^3$ . In other words, 1000 times more calculations must be performed if the grid size is divided by 10. Because this penalizing increase is rooted in the stability condition, it is imperative to explore other schemes that may have more attractive stability conditions. One such avenue is to consider implicit schemes. With a *fully implicit scheme*, the new values are used in the discretized derivative, and the algorithm is

$$\tilde{c}_k^{n+1} = \tilde{c}_k^n + D \left( \tilde{c}_{k+1}^{n+1} - 2\tilde{c}_k^{n+1} + \tilde{c}_{k-1}^{n+1} \right) \quad k = 2, \dots, m-1. \quad (5.41)$$

The application of the stability analysis provides an amplification factor  $\varrho$  given implicitly by

$$\varrho = 1 - \varrho 2D [1 - \cos(k_z \Delta z)],$$

of which the solution is

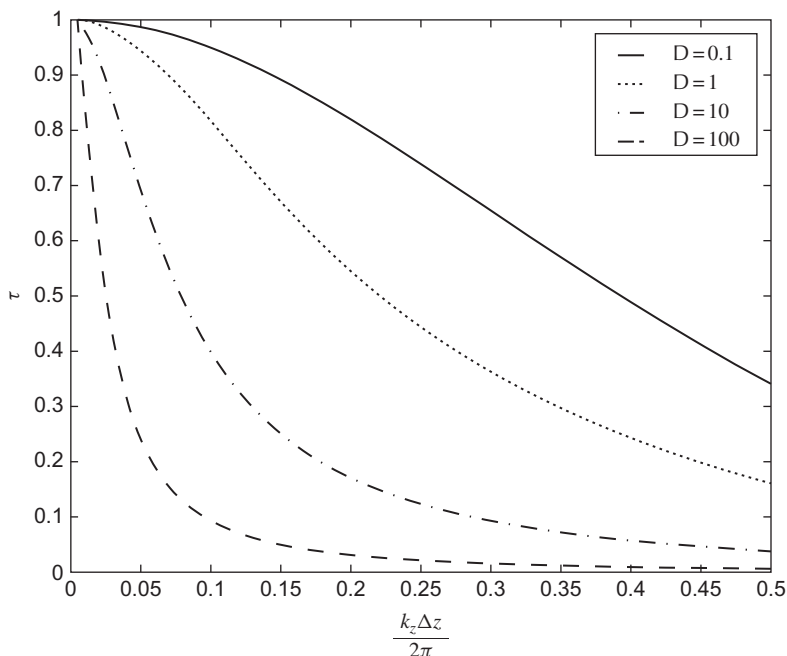
$$\varrho = \frac{1}{1 + 4D \sin^2(k_z \Delta z/2)} \leq 1. \quad (5.42)$$

Because this amplification factor is always real and less than unity, there is no stability condition to be met, and the scheme is stable for any time step. This is called *unconditional stability*. The implicit scheme therefore allows us in principle to use a time step as large as we wish. We immediately sense, of course, that a large time step cannot be acceptable. Should the time step be too large, the calculated values would not “explode” but would provide a very inaccurate approximation to the true solution. This is confirmed by comparing the damping of the numerical scheme against its true value:

$$\tau = \frac{\tilde{\omega}_i}{\omega_i} = \frac{\ln |1 + 4D \sin^2(k_z \Delta z/2)|}{4D (k_z \Delta z/2)^2}. \quad (5.43)$$

For small  $D$ , the scheme behaves reasonably well, but for larger  $D$ , even for scales ten times larger than the grid spacing, the error on the damping rate is similar to the damping rate itself (Fig. 5.12).

Setting aside the accuracy restriction, we still have another obstacle to overcome. To calculate the left-hand side of Eq. (5.41) at grid node  $k$ , we have to know the values of the still unknown  $\tilde{c}_{k+1}^{n+1}$  and  $\tilde{c}_{k-1}^{n+1}$ , which in turn depend on the



**FIGURE 5.12** Ratio  $\tau = \tilde{\omega}_i / \omega_i$  of the numerical damping rate of the implicit scheme to the exact value, as function of  $k_z \Delta z / 2$  for different values of  $D$ . For increasing time steps (increasing value of  $D$ ), the numerical damping deteriorates rapidly even for relatively well-resolved solution components, and it is prudent to use a short time step, if not for stability, at least for accuracy.

unknown values at their adjacent nodes. This creates a circular dependency. It is, however, a linear dependency, and all we need to do is to formulate the problem as a set of simultaneous linear equations, that is, to frame the problem as a matrix to be inverted, once at each time step. Standard numerical techniques are available for such problem, most of them based on the so-called *Gaussian elimination* or *lower-upper decomposition* (e.g., Riley, Hobson & Bence, 1977). These methods are the most efficient ones for inverting arbitrary matrices of dimension  $N$ , and their computational cost increases as  $N^3$ . For the one-dimensional case with  $N \sim m$ , the matrix inversion requires  $m^3$  operations to be performed.<sup>6</sup> Even if we executed only a single time step, the cost would be the same as for the execution of the explicit scheme during the full simulation. We may wonder: Is there some law of conservation of difficulty? Apparently there is, but we can exploit the particular form of the system to reduce the cost.

<sup>6</sup>If we anticipate generalization to three dimensions with  $N \sim 10^6 - 10^7$  unknowns, a matrix inversion would demand a number of operations proportional to  $N^3$  (at each time step!) and cannot be seriously considered as a viable approach.

Since the unknown value at one node depends on only the unknown values at the adjacent nodes and not those further away, the matrix of the system is not full but contains many zeros. All elements are zero except those on the diagonal and those immediately above (corresponding to one neighbor) and immediately below (corresponding to the neighbor on the other side). Such tridiagonal matrix, or *banded matrix*, is quite common, and techniques have been developed for their efficient inversion. The cost of inversion can be reduced to only 5m operations.<sup>7</sup> This is comparable to the number of operations for one step of the explicit scheme. And, since the implicit scheme can be run with a longer time step, it can be more efficient than the explicit scheme. A trade-off exists, however, between efficiency and accuracy.

An alternative time stepping is the *leapfrog method*, which “leaps” over the intermediate values, that is, the solution is marched from step  $n - 1$  to step  $n + 1$  by using the values at intermediate step  $n$  for the terms on the right-hand side of the equation. Applied to the diffusion equation, the leapfrog scheme generates the following algorithm:

$$\tilde{c}_k^{n+1} = \tilde{c}_k^{n-1} + 2D (\tilde{c}_{k+1}^n - 2\tilde{c}_k^n + \tilde{c}_{k-1}^n). \quad (5.44)$$

where  $D = \kappa \Delta t / \Delta z^2$  once again.

Because by the time values at time level  $n + 1$  are sought all values up to time level  $n$  are already known, this algorithm is explicit and does not require any matrix inversion. We can analyze its stability by considering, as before, a single Fourier mode of the type (5.29). The usual substitution into the discrete equation, this time (5.44), application of trigonometric formulas, and division by the Fourier mode itself then lead to the following equation for the amplification factor  $\varrho$  of the leapfrog scheme:

$$\varrho = \frac{1}{\varrho} - 8D \sin^2 \left( \frac{k_z \Delta z}{2} \right). \quad (5.45)$$

This equation is quadratic and has therefore two solutions for  $\varrho$ , corresponding to two temporal modes. Only a single mode was expected because the original equation had only a first-order time derivative in time, but obviously, the scheme has introduced a second, *spurious mode*. With  $b = 4D \sin^2(k_z \Delta z / 2)$ , the two solutions are

$$\varrho = -b \pm \sqrt{b^2 + 1}. \quad (5.46)$$

The physical mode is  $\varrho = -b + \sqrt{b^2 + 1}$  because for well-resolved components ( $k_z \Delta z \ll 1$  and thus  $b \ll 1$ ) it is approximately  $\varrho \simeq 1 - b \simeq 1 - Dk_z^2 \Delta z^2$ , as it should be [see Eq. (5.34)]. Its value is always less than one, and the physical

<sup>7</sup>See Appendix C for the formulation of the algorithm.

mode is numerically stable. The other solution,  $\varrho = -b - \sqrt{b^2 + 1}$ , corresponds to the spurious mode and, unfortunately, has a magnitude always larger than one, jeopardizing the overall stability of the scheme. This is an example of *unconditional instability*. Note, however, that although unstable in the diffusion case, the leapfrog scheme will be found to be stable when applied to other equations.

The spurious mode causes numerical instability and must therefore be suppressed. One basic method is *filtering* (see Section 10.6). Because numerical instability is manifested by flip-flop in time (due to the negative  $\varrho$  value), averaging over two consecutive time steps or taking some kind of running average, called filtering, eliminates the flip-flop mode. Filtering, unfortunately, also alters the physical mode, and, as a rule, it is always prudent not to have a large flip-flop mode in the first place. Its elimination should be done *a priori*, not *a posteriori*. In the case of models using leapfrog for the sake of other terms in the equation, such as advection terms which it handles in a stable manner, the diffusion term is generally discretized at time level  $n - 1$  rather than  $n$ , rendering the scheme as far as the diffusion part is concerned equivalent to the explicit Euler scheme with time step of  $2\Delta t$ .

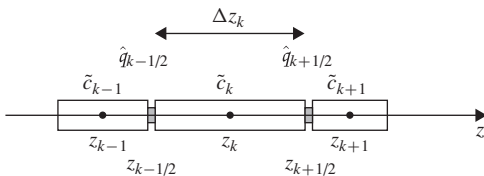
Finally, we can illustrate the *finite-volume technique* in the more general case of nonuniform diffusion and variable grid spacing. In analogy with Eq. (3.35), we integrate the diffusion equation over an interval between two consecutive cell boundaries and over one time step to obtain the grid-cell averages  $\bar{c}$  (Fig. 5.13)

$$\frac{\bar{c}_k^{n+1} - \bar{c}_k^n}{\Delta t_n} + \frac{\hat{q}_{k+1/2} - \hat{q}_{k-1/2}}{\Delta z_k} = 0, \quad (5.47)$$

assuming that the time-averaged flux at the interface between cells

$$\hat{q} = \frac{1}{\Delta t_n} \int_{t^n}^{t^{n+1}} -\kappa \frac{\partial c}{\partial z} dt \quad (5.48)$$

is somehow known. Up to this point, the equations are exact. The variable  $c$  appearing in the expression of the flux is the actual function, including all its subgrid-scale variations, whereas Eq. (5.47) deals only with space-time averages. Discretization enters the formulation as we relate the time-averaged flux



**FIGURE 5.13** Arrangement of cells and interfaces for the finite-volume technique. Concentration values are defined at cell centers, whereas flux values are defined between cells. Cell lengths do not have to be uniform.

to the space-averaged function  $\bar{c}$  to close the problem. We can, for example, estimate the flux using a factor  $\alpha$  of implicitness and a gradient approximation:

$$\hat{q}_{k-1/2} \simeq -(1-\alpha) \kappa_{k-1/2} \frac{\tilde{c}_k^n - \tilde{c}_{k-1}^n}{z_k - z_{k-1}} - \alpha \kappa_{k-1/2} \frac{\tilde{c}_k^{n+1} - \tilde{c}_{k-1}^{n+1}}{z_k - z_{k-1}}, \quad (5.49)$$

where  $\tilde{c}$  is now interpreted as the numerical estimate of the spatial averages. The numerical scheme reads

$$\begin{aligned} \tilde{c}_k^{n+1} = & \tilde{c}_k^n + (1-\alpha) \frac{\kappa_{k+1/2} \Delta t_n}{\Delta z_k} \frac{\tilde{c}_{k+1}^n - \tilde{c}_k^n}{z_{k+1} - z_k} - (1-\alpha) \frac{\kappa_{k-1/2} \Delta t_n}{\Delta z_k} \frac{\tilde{c}_k^n - \tilde{c}_{k-1}^n}{z_k - z_{k-1}} \\ & + \alpha \frac{\kappa_{k+1/2} \Delta t_n}{\Delta z_k} \frac{\tilde{c}_{k+1}^{n+1} - \tilde{c}_k^{n+1}}{z_{k+1} - z_k} - \alpha \frac{\kappa_{k-1/2} \Delta t_n}{\Delta z_k} \frac{\tilde{c}_k^{n+1} - \tilde{c}_{k-1}^{n+1}}{z_k - z_{k-1}}. \end{aligned} \quad (5.50)$$

With uniform grid spacing,  $\kappa$  constant, and  $\alpha=0$ , we recover Eq. (5.25). Since the present finite-volume scheme is by construction conservative (see Section 3.9), we have incidentally proven that (5.25) is conservative in the case of a uniform grid and constant diffusivity, a property that can be verified numerically with `firstdiffusion.m` even in the unstable case.

In practice, it is expedient to program the calculations with the flux values defined and stored alongside the concentration values. The computations then entail two stages in every step: first the computation of the flux values from the concentration values at the same time level and then the updation of the concentration values from these most recent flux values. In this manner, it is clear how to take into account variable parameters such as the local value of the diffusivity  $\kappa$  (at cell edges rather than at cell centers), local cell length, and momentary time step. The approach is also naturally suited for the implementation of flux boundary conditions.

## 5.6 MULTI-DIMENSIONAL NUMERICAL SCHEMES

Explicit schemes are readily generalized to two and three dimensions<sup>8</sup> with indices  $i, j$ , and  $k$  being grid positions in the respective directions  $x, y$ , and  $z$ :

$$\begin{aligned} \tilde{c}^{n+1} = & \tilde{c}^n + \frac{A \Delta t}{\Delta x^2} (\tilde{c}_{i+1}^n - 2\tilde{c}^n + \tilde{c}_{i-1}^n) \\ & + \frac{A \Delta t}{\Delta y^2} (\tilde{c}_{j+1}^n - 2\tilde{c}^n + \tilde{c}_{j-1}^n) \\ & + \frac{\kappa \Delta t}{\Delta z^2} (\tilde{c}_{k+1}^n - 2\tilde{c}^n + \tilde{c}_{k-1}^n). \end{aligned} \quad (5.51)$$

<sup>8</sup>In order not to overload the notation, indices are written here only if they differ from the local grid point index. Therefore,  $\tilde{c}(t^n, x_i, y_j, z_k)$  is written  $\tilde{c}^n$ , whereas  $\tilde{c}_{j+1}^n$  stands for  $\tilde{c}(t^n, x_i, y_{j+1}, z_k)$ .



The stability condition is readily obtained by using the amplification-factor analysis. Substituting the Fourier mode

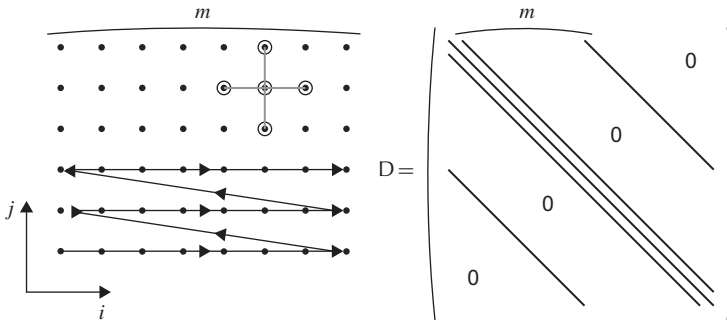
$$\tilde{c}^n = A \rho^n e^{i(i k_x \Delta x)} e^{i(j k_y \Delta y)} e^{i(k k_z \Delta z)} \quad (5.52)$$

in the discrete equation, we obtain the following generalization of Eq. (5.35):

$$\frac{\mathcal{A} \Delta t}{\Delta x^2} + \frac{\mathcal{A} \Delta t}{\Delta y^2} + \frac{\kappa \Delta t}{\Delta z^2} \leq \frac{1}{2}. \quad (5.53)$$

The implicit formulation of the scheme is not much more complicated and is, again, unconditionally stable. The associated matrix, however, is no longer tridiagonal but has a slightly more complicated structure (Fig. 5.14). Unfortunately, there exists no direct solver for which the cost remains proportional to the size of the problem. Several strategies can be developed to keep the method “implicit” with affordable costs.

In any case, a direct solver is in some way an overkill. It inverts the matrix exactly up to rounding errors, and such precision is not necessary in view of the much larger errors associated with the discretization (see Section 4.8). We can therefore afford to invert the matrix only approximately, and this can be accomplished by the use of iterative methods, which deliver solutions to any degree of approximation depending on the number of iterations performed. A small number of iterations usually yields an acceptable solution because the starting guess values may be taken as the values computed at the preceding time step. Two popular *iterative solvers* of linear systems are the Gauss–Seidel method and the Jacobi method, but there exist many other iterative solvers, more or less optimized for different kinds of problems and computers (e.g., Dongarra, Duffy, Sorensen & van der Vorst, 1998). In general, most software libraries offer



**FIGURE 5.14** If the numerical state vector is constructed row by row in two dimensions,  $\tilde{c}_{i,j}$  is the element  $(j-1)m+i$  of  $\mathbf{x}$ . Since the diffusion operator at point  $i, j$  involves  $\tilde{c}_{i,j}$ ,  $\tilde{c}_{i+1,j}$ ,  $\tilde{c}_{i-1,j}$ ,  $\tilde{c}_{i,j-1}$ , and  $\tilde{c}_{i,j+1}$ , the matrix to be inverted has zero elements everywhere, except on the diagonal (the point itself), the superdiagonal (point  $i+1, j$ ), the subdiagonal (point  $i-1, j$ ), and two lines situated  $\pm m$  away from the diagonal (point  $i, j \pm 1$ ).

a vast catalogue of methods, and we will only mention a few general approaches, giving more detail on specific methods later when we need to solve a Poisson equation for a pressure or streamfunction (Section 7.6).

Any linear system of simultaneous equations can be cast as

$$\mathbf{Ax} = \mathbf{b} \quad (5.54)$$

where the matrix  $\mathbf{A}$  gathers all the coefficients, the vector  $\mathbf{x}$  all the unknowns, and the vector  $\mathbf{b}$  the boundary values and external forcing terms, if any. The objective of an iterative method is to solve this system by generating a sequence  $\mathbf{x}^{(p)}$  that starts from a guess vector  $\mathbf{x}^0$  and gradually converges toward the solution. The algorithm is a repeated application of

$$\mathbf{Bx}^{(p+1)} = \mathbf{Cx}^{(p)} + \mathbf{b} \quad (5.55)$$

where  $\mathbf{B}$  must be easy to invert, otherwise there is no gain, and is typically a diagonal or triangular matrix (nonzero elements only on the diagonal or on the diagonal and one side of it). At convergence,  $\mathbf{x}^{(p+1)} = \mathbf{x}^{(p)}$  and we must therefore have  $\mathbf{B} - \mathbf{C} = \mathbf{A}$  to have solved Eq. (5.54). The closer  $\mathbf{B}$  is to  $\mathbf{A}$ , the faster the convergence since at the limit of  $\mathbf{B} = \mathbf{A}$ , a single iteration would yield the exact answer. Using  $\mathbf{C} = \mathbf{B} - \mathbf{A}$ , we can rewrite the iterative step as

$$\mathbf{x}^{(p+1)} = \mathbf{x}^{(p)} + \mathbf{B}^{-1} (\mathbf{b} - \mathbf{Ax}^{(p)}) \quad (5.56)$$

which is reminiscent of a time stepping method. Here,  $\mathbf{B}^{-1}$  denotes the inverse of  $\mathbf{B}$ . The Jacobi method uses a diagonal matrix  $\mathbf{B}$ , whereas the Gauss–Seidel method uses a triangular matrix  $\mathbf{B}$ . More advanced methods exist that converge faster than either of these. Those will be outlined in Section 7.6.

In GFD applications, diffusion is rarely dominant (except for vertical diffusion in strongly turbulent regime), and stability restrictions associated with diffusion are rarely penalizing. Therefore, it is advantageous to make the scheme implicit only in the direction of the strongest diffusion (or largest variability of diffusion), usually the vertical, and to treat the horizontal components explicitly:

$$\begin{aligned} \tilde{c}^{n+1} = & \tilde{c}^n + \frac{A\Delta t}{\Delta x^2} (\tilde{c}_{i+1}^n - 2\tilde{c}^n + \tilde{c}_{i-1}^n) \\ & + \frac{A\Delta t}{\Delta y^2} (\tilde{c}_{j+1}^n - 2\tilde{c}^n + \tilde{c}_{j-1}^n) \\ & + \frac{\kappa\Delta t}{\Delta z^2} (\tilde{c}_{k+1}^{n+1} - 2\tilde{c}^{n+1} + \tilde{c}_{k-1}^{n+1}). \end{aligned} \quad (5.57)$$

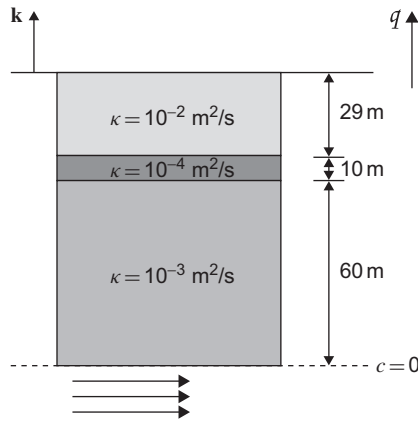
Then, instead of inverting a matrix with multiple bands of nonzero elements, we only need to invert a tridiagonal matrix at each point of the horizontal grid. *Alternating direction implicit* (ADI) methods use the same approach but

change the direction of the implicit sweep through the matrix at every time step. This helps when stability of the horizontal diffusion discretization is a concern.

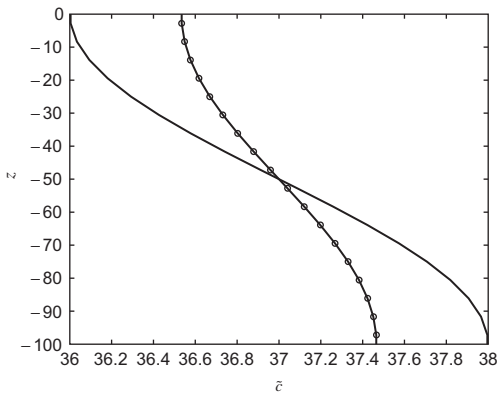
The biggest challenge associated with diffusion in GFD models is, however, not their numerical stability but rather their physical basis because diffusion is often introduced as a parameterization of unresolved processes. Occasionally, the unphysical behavior of the discretization may create a problem (e.g., Beckers, Burchard, Deleersnijder & Mathieu, 2000).

## ANALYTICAL PROBLEMS

- 5.1. What would the energy spectrum  $E_k(k)$  be in a turbulent flow where all length scales were contributing equally to dissipation? Is this spectrum realistic?
- 5.2. Knowing that the average atmospheric pressure on Earth's surface is  $1.013 \times 10^5 \text{ N/m}^2$  and that Earth's average radius is 6371 km, deduce the mass of the atmosphere. Then, using this and the fact that the earth receives  $1.75 \times 10^{17} \text{ W}$  from the sun globally, and assuming that half of the energy received from the sun is being dissipated in the atmosphere, estimate the rate of dissipation  $\epsilon$  in the atmosphere. Assuming finally that turbulence in the atmosphere obeys the Kolmogorov theory, estimate the smallest eddy scale in the air, its ratio to the largest scale (the earth's radius), and the large-scale wind velocity. Is this velocity scale realistic?
- 5.3. In a 15-m coastal zone, the water density is  $1032 \text{ kg/m}^3$  and the horizontal velocity scale is  $0.80 \text{ m/s}$ . What are the Reynolds number and the diameter of the shortest eddies? Approximately how many watts are dissipated per square meter of the ocean?
- 5.4. If you have to simulate the coastal ocean of the previous problem with a numerical model that includes 20 grid points over the vertical, what would be a reasonable value for the vertical eddy diffusivity?
- 5.5. Estimate the time it takes to reduce by a factor 2 a salinity variation in an ocean of depth  $H = 1000 \text{ m}$  in the presence of salt diffusion, with a diffusion coefficient  $\kappa$ . Compare two solutions, one using the molecular diffusion ( $\kappa = 10^{-9} \text{ m}^2/\text{s}$ ) and the other a turbulent diffusion typical of the deep ocean ( $\kappa = 10^{-4} \text{ m}^2/\text{s}$ ).
- 5.6. A deposition at the sea surface of a tracer (normalized and without units) can be modeled by a constant flux  $q = -10^{-4} \text{ m/s}$ . At depth  $z = -99 \text{ m}$ , a strong current is present and flushes the vertically diffused tracer so that  $c = 0$  is maintained at that level. Assuming the diffusion coefficient has the profile of Fig. 5.15, calculate the steady solution for the tracer distribution.



**FIGURE 5.15** Values of a nonuniform eddy diffusion for [Analytical Problem 5.6](#). A flux condition is imposed at the surface while  $c=0$  at the base of the domain.



**FIGURE 5.16** With a time step such that  $D=0.1$ , the initial condition (single line) of the 1D diffusion problem has been damped after 500 time steps and the numerical solution of the explicit scheme (open circles) is almost indistinguishable from the exact solution (shown as a line crossing the circles), even with only 30 grid points across the domain.

**5.7.** Verify the assertion made below [Eq. \(5.4\)](#) that the Reynolds number corresponding to the Kolmogorov scale is on the order of unity.

## NUMERICAL EXERCISES

- 5.1.** Cure the unstable version `firstdiffusion.m` by adapting the time step and verify that below the limit [\(5.35\)](#) the scheme is indeed stable and provides accurate solutions ([Fig. 5.16](#)).
- 5.2.** For a 1D Euler scheme with implicit factor  $\alpha$ , constant grid size and constant diffusion coefficient, prove that the stability condition is  $(1 - 2\alpha)D \leq 1/2$ .

- 5.3.** Implement periodic boundary conditions in the 1D diffusion problem (i.e.,  $c_{\text{top}} = c_{\text{bottom}}$  and  $q_{\text{top}} = q_{\text{bottom}}$ ). Then, search the internet for a tridiagonal matrix inversion algorithm adapted to periodic boundary conditions and implement it.
- 5.4.** Implement the 1D finite-volume method with an implicit factor  $\alpha$  and variable diffusion coefficient. Set the problem with the same initial and boundary conditions as in the beginning of [Section 5.3](#). Verify your solution against the exact solution (5.19).
- 5.5.** Apply the code developed in [Section 5.6](#) to the [Analytical Problem 5.6](#). Start with an arbitrary initial condition and march in time until the solution becomes stationary. Estimate a priori the permitted time step and the minimum total number of time steps, depending on the implicit factor. Take  $\Delta z = 2$ , track convergence during the calculations and compare your final solution with the exact solution. Also try to implement the naive discretization

$$\begin{aligned} \left. \frac{\partial}{\partial z} \left( \kappa \frac{\partial c}{\partial z} \right) \right|_{z_k} &= \kappa \left. \frac{\partial^2 c}{\partial z^2} \right|_{z_k} + \left. \frac{\partial \kappa}{\partial z} \right|_{z_k} \left. \frac{\partial c}{\partial z} \right|_{z_k} \\ &\sim \frac{\kappa_k (\tilde{c}_{k+1} - 2\tilde{c}_k + \tilde{c}_{k-1})}{\Delta z^2} \\ &\quad + \frac{(\kappa_{k+1} - \kappa_{k-1})}{2\Delta z} \frac{(\tilde{c}_{k+1} - \tilde{c}_{k-1})}{2\Delta z}. \end{aligned} \quad (5.58)$$

- 5.6.** The Dufort–Frankel scheme approximates the diffusion equation by

$$\tilde{c}_k^{n+1} = \tilde{c}_k^{n-1} + 2D \left[ \tilde{c}_{k+1}^n - (\tilde{c}_k^{n+1} + \tilde{c}_k^{n-1}) + \tilde{c}_{k-1}^n \right]. \quad (5.59)$$

Verify the consistency of this scheme. What relation must be imposed between  $\Delta t$  and  $\Delta z$  when each approaches zero to ensure consistency? Then, analyze numerical stability using the amplification-factor method.

**Andrey Nikolaevich Kolmogorov**  
**1903–1987**

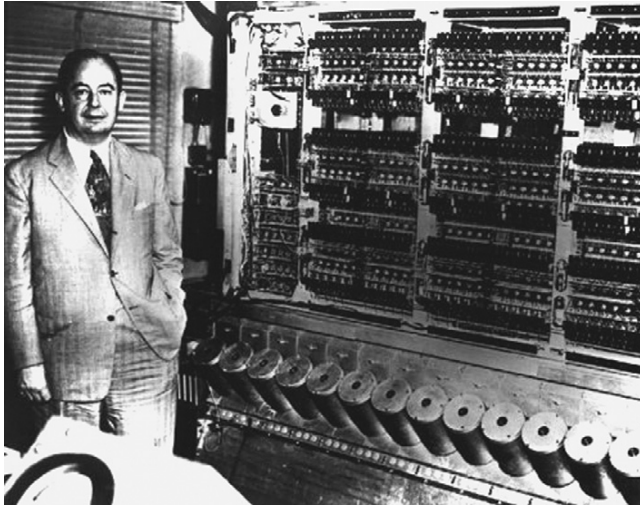


Andrey Kolmogorov was attracted to mathematics from an early age and, at the time of his studies at Moscow State University, sought the company of the most outstanding mathematicians. While still an undergraduate student, he began research and published several papers of international importance, chiefly on set theory. He had already 18 publications by the time he completed his doctorate in 1929. Kolmogorov's contributions to mathematics spanned a variety of topics, and he is perhaps best known for his work on probability theory and stochastic processes.

Research in stochastic processes led to a study of turbulent flow from a jet engine and, from there, to two famous papers on isotropic turbulence in 1941. It has been remarked that these two papers rank among the most important ones since Osborne Reynolds in the long and unfinished history of turbulence theory.

Kolmogorov found much inspiration for his work during nature walks in the outskirts of Moscow accompanied by colleagues and students. The brainstorming that had occurred during the walk often concluded in serious work around the dinner table upon return home. (*Photo from American Mathematical Society*)

### John Louis von Neumann 1903–1957



John von Neumann was a child prodigy. At age six, he could mentally divide eight-digit numbers and memorize the entire page of a telephone book in a matter of minutes, to the amazement of his parents' guests at home. Shortly after obtaining his doctorate in 1928, he left his native Hungary to take an appointment at Princeton University (USA). When the Institute for Advanced Studies was founded there in 1933, he was named one of the original Professors of Mathematics.

Besides seminal contributions to ergodic theory, group theory, and quantum mechanics, his work included the application of electronic computers to applied mathematics. Together with Jule Charney (see biography at end of Chapter 16) in the 1940s, he selected weather forecasting as the first challenge for the emerging electronic computers, which he helped assemble. Unlike Lewis Richardson before them, von Neumann and Charney started with a single equation, the barotropic vorticity equation. The results exceeded expectations, and scientific computing was launched.

A famous quote attributed to von Neumann is: "If people do not believe that mathematics is simple, it is only because they do not realize how complicated life is." (*Photo from Virginia Polytechnic Institute and State University*)