10

Grid functions and finite difference operators in 2D

This chapter is concerned with the extension of the difference operators introduced in Chapter 5 to two spatial dimensions. The 2D case is one of great interest in musical acoustics, given that many key components of musical instruments may be well described as such—for various percussion instruments such as drums, cymbals, and gongs, a 2D structure serves as the main resonating element, whereas in keyboard instruments and some stringed instruments it behaves as an auxiliary radiating element which imparts its own characteristic to the resulting sound. Perceptually speaking, the sound output from a 2D simulation is far richer than that of a 1D simulation. Part of this is due to the number of degrees of freedom, or modes, which, in the linear case, is considerably larger and of less regular a distribution than in 1D—sounds generated by 2D objects are generally inharmonic by nature. Beyond this, there are mechanisms at work, in particular in the nonlinear case, which lead to perceptual phenomena which have no real analogue in 1D; cymbal crashes are an excellent example of such behavior.

At the time of writing, there has been, so far, relatively little work on 2D problems in sound synthesis (with some exceptions: [139, 46, 140, 385, 384]), partly because, until recently, real-time synthesis from such systems on small computers was not possible. Another reason has been that percussion instruments have seen much less fundamental investigation from the point of view of musical acoustics than other instruments, though there is a growing body of work by Rossing and his collaborators (see [299, 136] for an overview), concerned mainly with experimental determination of modal frequencies, as well as considerable related work on time domain characterizations and nonlinear phenomena [283, 300]. On the other hand, such problems have a long research history in mainstream simulation, and, as a result, there is a wide expanse of literature and results which may be adapted to sound synthesis applications. Difference schemes are again a good choice for synthesis, and much of the material presented in Chapter 5 may be generalized in a natural way. The presentation here will be as brief as possible, except when it comes to certain features which are particular to 2D.

Partial differential operators in 2D, in both Cartesian and radial coordinates, are presented in Section 10.1, accompanied by frequency domain and energy analysis concepts and tools. Difference operators are then introduced in Cartesian coordinates in Section 10.2 and in radial coordinates in Section 10.3.

Numerical Sound Synthesis: Finite Difference Schemes and Simulation in Musical Acoustics Stefan Bilbao
 ISBN: 978-0-470-51046-9

10.1 Partial differential operators and PDEs in two space variables

The single largest headache in 2D, both at the algorithm design stage and in programming a working synthesis routine, is problem geometry. Whereas 1D problems are defined over a domain which may always be scaled to the unit interval, in 2D no such simplification is possible. As such, the choice of coordinates becomes important. Here, to keep the emphasis on basic principles, only two such choices, namely Cartesian and radial coordinates (certainly the most useful in musical acoustics), will be discussed. Despite this, it is worth keeping in mind that numerical simulation methods are by no means limited to such coordinate choices, though as the choice of coordinate system (generally governed by geometry) becomes more complex, finite difference methods lose a good deal of their appeal, and finite element methods (see page 386) become an attractive option.

10.1.1 Cartesian and radial coordinates

Certainly the simplest coordinate system, and one which is ideal for working with problems defined over square or rectangular regions, is the Cartesian coordinate system, where a position is defined by the pair (x, y). For problems defined over circles, radial coordinates (r, θ) defined in terms of Cartesian coordinates by

$$r = \sqrt{x^2 + y^2} \qquad \theta = \tan^{-1}(y/x) \tag{10.1}$$

may be more appropriate. See Figure 10.1 for an illustration of such coordinate systems.

In finite difference applications, Cartesian coordinates are undeniably much simpler to deal with, due to the symmetry between the x and y coordinate directions. Radial coordinates are trickier in some respects, especially due to the existence of a coordinate center, and also because differential operators exhibit a dependence on radius r.

10.1.2 Spatial domains

As in 1D, a 2D problem is defined over a given domain $\mathcal{D}$, a subset of the plane $\mathbb{R}^2$—see Figure 10.1 for an illustration of some of the regions to be discussed here. For analysis purposes, it is often convenient to work over the entire plane, or with $\mathcal{D} = \mathbb{R}^2$. In Cartesian coordinates,

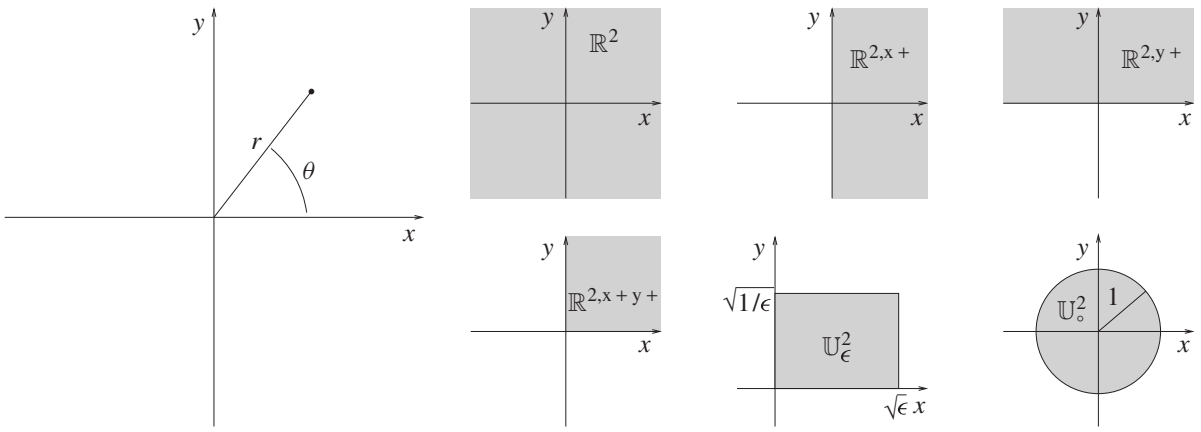


Figure 10.1 Coordinates (x, y), and (r, θ), and various regions of interest in Cartesian and radial coordinates.

sometimes, for the analysis of boundary conditions, it is useful to examine a semi-infinite domain, or half plane, of the form $\mathbb{R}^{2,x+} = \{(x, y) \in \mathbb{R}^2, x \geq 0\}$, or $\mathbb{R}^{2,y+} = \{(x, y) \in \mathbb{R}^2, y \geq 0\}$, and in order to deal with corner conditions, the quarter plane $\mathbb{R}^{2,x+y+} = \{(x, y) \in \mathbb{R}^2, x \geq 0, y \geq 0\}$.

In practice, however, at least in Cartesian coordinates, it is the $L_x \times L_y$ rectangular region, of the form $\{(x, y) \in \mathbb{R}^2, 0 \leq x \leq L_x, 0 \leq y \leq L_y\}$, which is of most interest. Through scaling of spatial variables, i.e., setting coordinates

$$x' = x/\sqrt{L_x L_y} \qquad y' = y/\sqrt{L_x L_y}$$

this region may always be reduced to the unit area rectangle of dimensions $\sqrt{\epsilon} \times \sqrt{1/\epsilon}$, where $\epsilon = L_x/L_y$ is the aspect ratio for the rectangle. This region will henceforth be called $\mathbb{U}^2_\epsilon$, and scaled coordinates will always be assumed (with the primed notation dropped). When $\epsilon = 1$ (i.e., the region is a square), the symbol $\mathbb{U}^2$ will be used.

In radial coordinates, the main region of interest is the circle of radius R, i.e., $\{(r, \theta) \in \mathbb{R}^2, 0 \leq r \leq R\}$. Again, through the introduction of a scaled coordinate this region may be reduced to the circle of radius 1, $\mathbb{U}^2_\circ$.

10.1.3 Partial differential operators

In Cartesian coordinates, the differential operators which appear, beyond partial time derivatives, which have already been discussed in Chapter 5, are of the form $\partial/\partial x$, $\partial/\partial y$, $\partial^2/\partial x^2$, $\partial^2/\partial y^2$, $\partial^2/\partial x \partial y$, etc. When applied to a function $u(x, y, t)$, the following notation will be used:

$$\frac{\partial u}{\partial x} = u_x \qquad \frac{\partial u}{\partial y} = u_y \qquad \frac{\partial^2 u}{\partial x^2} = u_{xx} \qquad \frac{\partial^2 u}{\partial y^2} = u_{yy} \qquad \frac{\partial^2 u}{\partial x \partial y} = u_{xy} \qquad \text{etc.}$$

Technical considerations having to do with interchanging the order of derivatives will be neglected here, so it may be assumed, for example, that $u_{xy} = u_{yx}$.

In radial coordinates, similar operators and accompanying notation are used, i.e.,

$$\frac{\partial u}{\partial r} = u_r \qquad \frac{\partial u}{\partial \theta} = u_\theta \qquad \frac{\partial^2 u}{\partial r^2} = u_{rr} \qquad \frac{\partial^2 u}{\partial \theta^2} = u_{\theta\theta} \qquad \frac{\partial^2 u}{\partial r \partial \theta} = u_{r\theta} \qquad \text{etc.}$$

For isotropic systems in musical acoustics, the most commonly occurring differential operator is not any of the above operators in isolation, but rather the 2D Laplacian Δ, defined in terms of its action on a function u as

$$\Delta u = u_{xx} + u_{yy} \qquad \Delta u = \frac{1}{r}(r u_r)_r + \frac{1}{r^2} u_{\theta\theta} \tag{10.2}$$

in Cartesian and radial coordinates, respectively. Also important, especially in problems in plate dynamics, is the fourth-order operator known as the bi-Laplacian, or biharmonic operator $\Delta\Delta$, a double application of the Laplacian operator. In Cartesian coordinates, for example, when applied to a function u, it behaves as

$$\Delta\Delta u = u_{xxxx} + 2u_{xxyy} + u_{yyyy} \tag{10.3}$$

10.1.4 Differential operators in the spatial frequency domain

Just as in 1D, it is possible to view differential operators in terms of their behavior in the spatial frequency domain—in general, this is only simple in Cartesian coordinates, where differential

operators remain shift invariant. (Notice that in radial coordinates, operators such as the Laplacian show an explicit dependence on the coordinate r.)

In 2D, the frequency domain ansatz (5.4) is generalized to

$$u(x, y, t) = e^{st+j\beta_x x+j\beta_y y}$$

When $s = j\omega$, this corresponds to a wave traveling in direction $\boldsymbol{\beta} = (\beta_x, \beta_y)$ in the Cartesian plane, of wavelength $2\pi/|\boldsymbol{\beta}|$, where $|\boldsymbol{\beta}| = (\beta_x^2 + \beta_y^2)^{1/2}$ is the wavenumber magnitude, and where β_x and β_y are the individual components. For such a test function, the various differential operators above act as

$$\frac{\partial u}{\partial x} = j\beta_x u \qquad \frac{\partial u}{\partial y} = j\beta_y u \qquad \frac{\partial^2 u}{\partial x^2} = -\beta_x^2 u \qquad \frac{\partial^2 u}{\partial y^2} = -\beta_y^2 u \qquad \frac{\partial^2 u}{\partial x \partial y} = -\beta_y \beta_x u$$

and

$$\Delta u = -(\beta_x^2 + \beta_y^2)u = -|\boldsymbol{\beta}|^2 u \qquad \Delta\Delta u = |\boldsymbol{\beta}|^4 u$$

Notice in particular that the operators Δ and $\Delta\Delta$ lead to multiplicative factors which depend only on the wavenumber magnitude $|\boldsymbol{\beta}|$, and not on the individual components β_x and β_y—this is a reflection of the isotropic character of such operations, which occur naturally in problems which do not exhibit any directional dependence. The same is not true, however, of the discrete operators which approximate them; see Section 10.2.2.

10.1.5 Inner products

The definition of the L_2 inner product in 2D is a natural extension of that in 1D. For two functions f and g, dependent on two spatial coordinates, and possibly time as well, one may write

$$\langle f, g\rangle_{\mathcal{D}} = \iint_{\mathcal{D}} fg\,dx\,dy \qquad\qquad \langle f, g\rangle_{\mathcal{D}} = \iint_{\mathcal{D}} fgr\,dr\,d\theta \tag{10.4}$$

in Cartesian and radial coordinates, respectively; the same notation will be used for both coordinate systems, though one should note the presence of the factor r implicit in the definition in the radial case. The norm of a function f may be defined, as in the 1D case, as $\|f\|_{\mathcal{D}} = \langle f, f\rangle_{\mathcal{D}}$.

Integration by parts follows; in Cartesian coordinates, for example, over $\mathcal{D} = \mathbb{R}^2$,

$$\langle f, g_x\rangle_{\mathbb{R}^2} = -\langle f_x, g\rangle_{\mathbb{R}^2} \qquad\qquad \langle f, g_y\rangle_{\mathbb{R}^2} = -\langle f_y, g\rangle_{\mathbb{R}^2} \tag{10.5}$$

The following identity also holds for inner products involving the Laplacian:

$$\langle f, \Delta g\rangle_{\mathbb{R}^2} = \langle \Delta f, g\rangle_{\mathbb{R}^2} \tag{10.6}$$

This holds in Cartesian coordinates, and in radial coordinates, provided that f and g and their radial derivatives are bounded near the origin.

The Cauchy–Schwartz inequality (5.7a) and triangle inequality (5.7b) hold as in 1D, over any domain $\mathcal{D}$.

Edges

As in 1D, when the domain $\mathcal{D}$ possesses a boundary, or, in this case, an edge, extra terms appear in the above identities. Consider first the half plane $\mathcal{D} = \mathbb{R}^{2,x+}$. Now, the first of the integration

by parts identities (10.5) becomes

$$\langle f, g_x \rangle_{\mathbb{R}^{2,x+}} = -\langle f_x, g \rangle_{\mathbb{R}^{2,x+}} - \{f, g\}_{(0,\mathbb{R})}$$

where $\{f, g\}_{(0,\mathbb{R})}$ indicates a 1D inner product over the domain boundary at $x = 0$, i.e.,

$$\{f, g\}_{(0,\mathbb{R})} = \int_{y=-\infty}^{\infty} f(0, y) g(0, y) dy \tag{10.7}$$

This special notation for the 1D inner product used to indicate boundary terms arising in 2D problems is distinct from that employed in previous chapters—see Section 5.1.3.

Additional terms appear when higher derivatives are involved. For the case of the Laplacian, over the same domain, (10.6) becomes

$$\langle f, \Delta g \rangle_{\mathbb{R}^{2,x+}} = \langle \Delta f, g \rangle_{\mathbb{R}^{2,x+}} - \{f, g_x\}_{(0,\mathbb{R})} + \{f_x, g\}_{(0,\mathbb{R})}$$

The case of the quarter plane, $\mathcal{D} = \mathbb{R}^{2,x+y+}$, is of particular interest in problems defined over rectangular regions. Now, in the case of the Laplacian, boundary terms appear along both edges:

$$\langle f, \Delta g \rangle_{\mathbb{R}^{2,x+y+}} = \langle \Delta f, g \rangle_{\mathbb{R}^{2,x+y+}} - \{f, g_x\}_{(0,\mathbb{R}^+)} + \{f_x, g\}_{(0,\mathbb{R}^+)} - \{f, g_y\}_{(\mathbb{R}^+,0)} + \{f_y, g\}_{(\mathbb{R}^+,0)}$$

Circular domains

The circular domain $\mathcal{D} = \mathbb{U}_\circ^2$ is of great practical utility in sound synthesis applications for certain percussion instruments, such as cymbals and gongs. There is only a single edge, at $r = 1$, though, in difference approximations, an artificial "edge" appears at $r = 0$, which must be treated carefully—see the discussion of the discrete Laplacian beginning on page 300. In order to examine boundary conditions, integration by parts is again a necessary tool. Here is an identity of great utility:

$$\langle f, \Delta g \rangle_{\mathbb{U}_\circ^2} = -\langle f_r, g_r \rangle_{\mathbb{U}_\circ^2} - \langle (1/r) f_\theta, (1/r) g_\theta \rangle_{\mathbb{U}_\circ^2} + \{f, g_r\}_{(1,[0,2\pi))} \tag{10.8a}$$

$$= \langle \Delta f, g \rangle_{\mathbb{U}_\circ^2} + \{f, g_r\}_{(1,[0,2\pi))} - \{f_r, g\}_{(1,[0,2\pi))} \tag{10.8b}$$

Of slight concern are the factors of $(1/r)$ which appear in the inner products above; it must further be assumed that f and g are bounded and single valued at $r = 0$.

10.2 Grid functions and difference operators: Cartesian coordinates

The extension of the definitions in Section 5.2.2 to two spatial coordinates is, in the Cartesian case, immediate. A grid function $u_{l,m}^n$, for $(l, m) \in \mathcal{D}$, and $n \geq 0$, represents an approximation to a continuous function $u(x, y, t)$, at coordinates $x = lh_x$, $y = mh_y$, $t = nk$. Here, $\mathcal{D}$ is a subset of the set of pairs of integers, $\mathbb{Z}^2$, and h_x and h_y are the grid spacings in the x and y directions. The semi-infinite domains, or half planes corresponding to $\mathbb{R}^{2,x+}$ and $\mathbb{R}^{2,y+}$, are $\mathbb{Z}^{2,x+} = \{(l, m) \in \mathbb{Z}^2, l \geq 0\}$ and $\mathbb{Z}^{2,y+} = \{(l, m) \in \mathbb{Z}^2, m \geq 0\}$. For the quarter plane, one can define $\mathbb{Z}^{2,x+y+} = \{(l, m) \in \mathbb{Z}^2, l, m \geq 0\}$. Most important, in real-world simulation, is the rectangular region $\mathbb{U}_{N_x,N_y}^2 = \{(l, m) \in \mathbb{Z}^2, 0 \leq l \leq N_x, 0 \leq m \leq N_y\}$.

Temporal operators behave exactly as those defined in 1D, in Section 5.2.1, and it is not worth repeating these definitions here. Spatial shift operators in the x and y directions may be defined as

$$e_{x+} u_{l,m}^n = u_{l+1,m}^n \qquad e_{x-} u_{l,m}^n = u_{l-1,m}^n \qquad e_{y+} u_{l,m}^n = u_{l,m+1}^n \qquad e_{y-} u_{l,m}^n = u_{l,m-1}^n$$

and forward, backward, and centered difference operators as

$$\delta_{x+} \triangleq \frac{1}{h_x}(e_{x+} - 1) \approxeq \frac{\partial}{\partial x} \qquad \delta_{x-} \triangleq \frac{1}{h_x}(1 - e_{x-}) \approxeq \frac{\partial}{\partial x} \qquad \delta_{x\cdot} \triangleq \frac{1}{2h_x}(e_{x+} - e_{x-}) \approxeq \frac{\partial}{\partial x}$$

$$\delta_{y+} \triangleq \frac{1}{h_y}\left(e_{y+} - 1\right) \approxeq \frac{\partial}{\partial y} \qquad \delta_{y-} \triangleq \frac{1}{h_y}\left(1 - e_{y-}\right) \approxeq \frac{\partial}{\partial y} \qquad \delta_{y\cdot} \triangleq \frac{1}{2h_y}\left(e_{y+} - e_{y-}\right) \approxeq \frac{\partial}{\partial y}$$

Centered second derivative approximations follow immediately as

$$\delta_{xx} = \delta_{x+}\delta_{x-} \approxeq \frac{\partial^2}{\partial x^2} \qquad\qquad \delta_{yy} = \delta_{y+}\delta_{y-} \approxeq \frac{\partial^2}{\partial y^2}$$

and various mixed derivative approximations, such as $\delta_{x+}\delta_{y-}$ approximating $\partial^2/\partial x \partial y$, may be arrived at through composition. It is also possible to define averaging operators in the x and y directions, such as μ_{x+}, μ_{y-}, etc., generalizing those presented in Section 5.2.2. See Figure 10.2 and Problem 10.1.

Equal grid spacings

Though, in general, one can choose unequal grid spacings h_x and h_y in the two Cartesian coordinates x and y, for simplicity of analysis and programming it is often easier to set them equal to a single constant, i.e., $h_x = h_y = h$. This is natural for problems which are isotropic (i.e., for which wave propagation is independent of direction). This simplification is employed in much of the remainder of this book. There are cases, though, for which such a choice can lead to errors which can become perceptually important when a very coarse grid is used (i.e., typically in simulating musical systems of high pitch). For some examples relating to the 2D wave equation, see Problem 11.7 and Programming Exercise 11.1 in the following chapter. When the system itself exhibits significant anisotropy, however (such as in the case of certain plates used as soundboards—see Section 12.5), this simplification must be revisited.

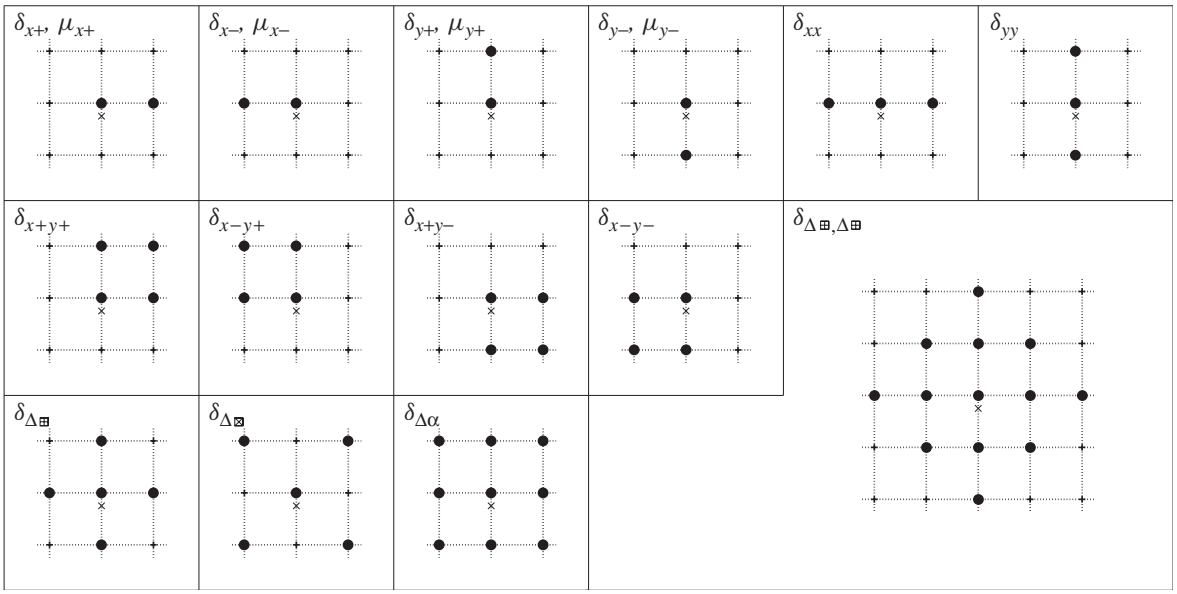


Figure 10.2 Stencils or footprints for various 2D spatial difference operators, in Cartesian coordinates, as indicated. In each case, the point at which the operator acts is indicated by a $\times$ symbol.

The discrete Laplacian and biharmonic operators

Quite important in musical sound synthesis applications is the approximation to the Laplacian operator, as given in (10.2)—there are clearly many ways of doing this. The simplest, by far, is to make use of what is known as the five-point Laplacian. Here are two possible forms of this operator, one making use of points adjacent to the center point, and another employing points diagonally adjacent:

$$\delta_{\Delta\boxplus} = \delta_{xx} + \delta_{yy} = \Delta + O(h^2) \qquad \delta_{\Delta\boxtimes} = \delta_{xx} + \delta_{yy} + \frac{h^2}{2}\delta_{xx}\delta_{yy} = \Delta + O(h^2) \tag{10.9}$$

These two operators may be combined in a standard way (see Section 2.2.2) to yield a so-called nine-point Laplacian, depending on a free parameter α:

$$\delta_{\Delta\alpha} \triangleq \alpha\delta_{\Delta\boxplus} + (1-\alpha)\delta_{\Delta\boxtimes} = \Delta + O(h^2) \tag{10.10}$$

Though it involves more grid points, it may be used in order to render an approximation more isotropic—see Section 11.3 for an application of this in the case of the 2D wave equation.

Also important, in the case of the vibrating stiff plate, is the discrete biharmonic operator, or bi-Laplacian, which consists of the composition of the Laplacian with itself, or $\Delta\Delta$. A simple approximation may be given as

$$\delta_{\Delta\boxplus,\Delta\boxplus} \triangleq \delta_{\Delta\boxplus}\delta_{\Delta\boxplus} = \Delta\Delta + O(h^2)$$

One could go further here and develop a family of approximations using a parameterized combination of the operators $\delta_{\Delta\boxplus}$ and $\delta_{\Delta\boxtimes}$—see [48] for more on this topic, and the text of Szilard [348], which covers difference approximations to the biharmonic operator in great detail.

The stencils or footprints of the above spatial difference operators are illustrated in Figure 10.2. The precise coefficients to be applied at the various points are not indicated, but easily determined—see Problem 10.1.

10.2.1 Interpolation in 2D and spreading operators

Interpolation, necessary for reading out a waveform from a discrete grid, and spreading, necessary when one is interested in exciting such a grid, or coupling it to another object, are direct extensions of their 1D counterparts, as described in Section 5.2.4.

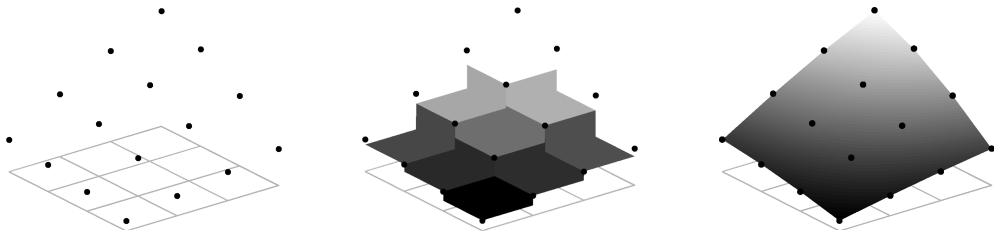
For a grid function $u_{l,m}$, defined for integer l and m, with grid spacings h_x and h_y, a zeroth-order (westward/southward) interpolant $I_0(x_\mathrm{o}, y_\mathrm{o})$ operating at position $(x_\mathrm{o}, y_\mathrm{o})$ is defined by

$$I_0(x_\mathrm{o}, y_\mathrm{o})u = u_{l_\mathrm{o},m_\mathrm{o}} \qquad \text{where} \qquad l_\mathrm{o} = \mathrm{floor}(x_\mathrm{o}/h_x), \quad m_\mathrm{o} = \mathrm{floor}(y_\mathrm{o}/h_y) \tag{10.11}$$

Such an interpolant corresponds to a crude "staircase" approximation, as illustrated in Figure 10.3(b). It is particularly useful in cases in which the interpolation point is static, and when there is good grid resolution.

Another choice which is appealing, due to its simplicity, is the bilinear interpolant, $I_1(x_\mathrm{o}, y_\mathrm{o})$. If the grid indices l_o and m_o are as given in (10.11), and furthermore the remainders by $\alpha_{x,\mathrm{o}} = x_\mathrm{o}/h_x - l_\mathrm{o}$ and $\alpha_{y,\mathrm{o}} = y_\mathrm{o}/h_y - m_\mathrm{o}$, it is defined as

$$\begin{aligned} I_1(x_\mathrm{o}, y_\mathrm{o})u = {} & (1-\alpha_{x,\mathrm{o}})(1-\alpha_{y,\mathrm{o}})u_{l_\mathrm{o},m_\mathrm{o}} + (1-\alpha_{x,\mathrm{o}})\alpha_{y,\mathrm{o}}u_{l_\mathrm{o},m_\mathrm{o}+1} \\ & + \alpha_{x,\mathrm{o}}(1-\alpha_{y,\mathrm{o}})u_{l_\mathrm{o}+1,m_\mathrm{o}} + \alpha_{x,\mathrm{o}}\alpha_{y,\mathrm{o}}u_{l_\mathrm{o}+1,m_\mathrm{o}+1} \end{aligned}$$



(a) (b) (c)

Figure 10.3 Interpolation in 2D: (a) values of a grid function; (b) the simple truncating interpolant I_0; and (c) a bilinear interpolant I_1. The interpolants are viewed as "reconstructing" a 2D function underlying the grid function shown in (a), from which interpolated values may then be drawn.

This interpolant makes use of the four grid points neighboring the interpolation point, using a bilinear function of these values—see Figure 10.3(c). It is more accurate than I_0, and the simplest interpolant one could realistically use in situations where the readout point is moving. See Programming Exercise 10.1.

Spreading grid functions $J_0(x_\mathrm{i}, y_\mathrm{i})$ and $J_1(x_\mathrm{i}, y_\mathrm{i})$, operating at position $(x_\mathrm{i}, y_\mathrm{i})$, may be similarly defined as the duals to these interpolants, i.e.,

$$J_{l,m,0}(x_\mathrm{i}, y_\mathrm{i}) = \frac{1}{h_x h_y} \begin{cases} 1, & l = l_\mathrm{i}, m = m_\mathrm{i} \\ 0, & \text{else} \end{cases}$$

$$J_{l,m,1}(x_\mathrm{i}, y_\mathrm{i}) = \frac{1}{h_x h_y} \begin{cases} (1-\alpha_{x,\mathrm{i}})(1-\alpha_{y,\mathrm{i}},) & l = l_\mathrm{i}, m = m_\mathrm{i} \\ (1-\alpha_{x,\mathrm{i}})\alpha_{y,\mathrm{i}}, & l = l_\mathrm{i}, m = m_\mathrm{i}+1 \\ \alpha_{x,\mathrm{i}}(1-\alpha_{y,\mathrm{i}}), & l = l_\mathrm{i}+1, m = m_\mathrm{i} \\ \alpha_{x,\mathrm{i}}\alpha_{y,\mathrm{i}}, & l = l_\mathrm{i}+1, m = m_\mathrm{i}+1 \\ 0, & \text{else} \end{cases}$$

where $l_\mathrm{i} = \mathrm{floor}(x_\mathrm{i}/h_x)$, $m_\mathrm{i} = \mathrm{floor}(y_\mathrm{i}/h_y)$, $\alpha_{x,\mathrm{i}} = x_\mathrm{i}/h_x - l_\mathrm{i}$, and $\alpha_{y,\mathrm{i}} = y_\mathrm{i}/h_y - m_\mathrm{i}$. Either approximates a 2D Dirac delta function $\delta(x - x_\mathrm{i}, y - y_\mathrm{i})$.

One could go further here, and develop higher-order interpolants—as those with a background in image processing may know, this is a much more complex matter in 2D than in the 1D case [306], and the matter is not pursued further here. See also Programming Exercise 11.2 for some exploration of the perceptual effects of the choice of a bilinear interpolant in audio applications.

10.2.2 Frequency domain analysis

Frequency domain analysis of difference operators in the Cartesian case is a straightforward generalization from 1D. Skipping over the definition of Fourier and z transforms, the ansatz is now

$$u_{l,m}^n = z^n e^{jh(l\beta_x + m\beta_y)} \tag{10.12}$$

where, again, β_x and β_y are components of a vector wavenumber $\boldsymbol{\beta} = (\beta_x, \beta_y)$ of magnitude $|\boldsymbol{\beta}| = (\beta_x^2 + \beta_y^2)^{1/2}$. The frequency domain behavior of temporal operators is unchanged from the 1D case.

Defining the variables p_x and p_y by

$$p_x = \sin^2(\beta_x h/2) \qquad p_y = \sin^2(\beta_y h/2) \tag{10.13}$$

it is true for a single component of the form (10.12) that

$$\delta_{xx}u = -\frac{4}{h^2}p_x u \qquad \delta_{yy}u = -\frac{4}{h^2}p_y u$$

$$\delta_{\Delta\boxplus}u = -\frac{4}{h^2}\left(p_x + p_y\right)u \quad \delta_{\Delta\boxtimes}u = -\frac{4}{h^2}\left(p_x + p_y - 2p_x p_y\right)u \quad \delta_{\Delta\boxplus,\Delta\boxplus}u = \frac{16}{h^4}\left(p_x + p_y\right)^2 u \tag{10.14}$$

Notice in particular that the frequency domain multiplication factors for the approximations to the Laplacian are bilinear functions of p_x and p_y defined over the unit square—one may use properties of such functions in order to simplify stability analysis. See Problem 10.2.

Anisotropic behavior

One new facet of finite difference schemes in 2D is numerical anisotropy—waves travel at different speeds in different directions, even when the underlying problem is isotropic. This is wholly due to the directional asymmetry imposed on a problem by introducing a grid and is a phenomenon which shows itself most prominently at high frequencies (or short wavelengths)—in the long-wavelength limit, the numerical behavior of operators which approximate isotropic differential operators becomes approximately isotropic.

As a simple example of this, consider the operator $\delta_{\Delta\boxplus}$, as defined in (10.9). It is perhaps easiest to examine the anisotropy in the frequency domain representation. Expanding from (10.14) in powers of the wavenumber components β_x and β_y gives

$$\delta_{\Delta\boxplus} \Longrightarrow -\frac{4}{h^2}\left(p_x + p_y\right) = -|\boldsymbol{\beta}|^2 + \frac{h^2}{12}\left(\beta_x^4 + \beta_y^4\right) + O(h^4)$$

Thus, as expected, the operator $\delta_{\Delta\boxplus}$ approximates the Laplacian Δ to second-order accuracy in the grid spacing h, but the higher-order terms cannot be grouped in terms of the wavenumber magnitude alone. Such numerical anisotropy, and ways of reducing it, will be discussed with regard to the 2D wave equation in Section 11.3. See also Problem 10.3 and Programming Exercise 10.2.

Amplification polynomials

Just as in the lumped and 1D cases, for LSI problems in 2D, in the analysis of difference schemes, one often arrives at amplification polynomials of the following form:

$$P(z) = \sum_{l=0}^{N} a_l(\beta_x, \beta_y) z^l = 0 \tag{10.15}$$

As before, a stability condition is arrived at by finding conditions such that the roots are bounded by one in magnitude, now for all values of β_x and β_y supported on the grid.

10.2.3 A discrete inner product

The discrete inner product and norm over a Cartesian grid in 2D are a direct extension of those given in Section 5.2.9, in the 1D case. For an arbitrary domain $\mathcal{D}$, the simplest definition, for two grid functions $f_{l,m}$, $g_{l,m}$ defined over a grid of uniform spacing h in the x and y directions, is

$$\langle f, g\rangle_{\mathcal{D}} = \sum_{(l,m)\in\mathcal{D}} h^2 f_{l,m} g_{l,m} \qquad \|f\|_{\mathcal{D}} = \langle f, f\rangle_{\mathcal{D}}$$

The Cauchy-Schwartz and triangle inequalities again hold, as in 1D.

Summation by parts and inequalities

Summation by parts extends naturally to 2D. Consider, in the first instance, grid functions f and g defined over the domain $\mathcal{D} = \mathbb{Z}^2$. It can be directly shown that

$$\langle f, \delta_{x-} g\rangle_{\mathbb{Z}^2} = -\langle \delta_{x+} f, g\rangle_{\mathbb{Z}^2} \qquad \langle f, \delta_{y-} g\rangle_{\mathbb{Z}^2} = -\langle \delta_{y+} f, g\rangle_{\mathbb{Z}^2} \tag{10.16}$$

From these identities, one may go further and show, for the five-point Laplacian, that

$$\langle f, \delta_{\Delta\boxplus} g\rangle_{\mathbb{Z}^2} = -\langle \delta_{x-} f, \delta_{x-} g\rangle_{\mathbb{Z}^2} - \langle \delta_{y-} f, \delta_{y-} g\rangle_{\mathbb{Z}^2} = \langle \delta_{\Delta\boxplus} f, g\rangle_{\mathbb{Z}^2} \tag{10.17}$$

Inequalities relating norms of grid functions under difference operators to norms of the grid functions themselves also follow:

$$\|\delta_{x+} u\|_{\mathbb{Z}^2} \leq \frac{2}{h}\|u\|_{\mathbb{Z}^2} \quad \|\delta_{y+} u\|_{\mathbb{Z}^2} \leq \frac{2}{h}\|u\|_{\mathbb{Z}^2} \quad \|\delta_{xx} u\|_{\mathbb{Z}^2} \leq \frac{4}{h^2}\|u\|_{\mathbb{Z}^2} \quad \|\delta_{yy} u\|_{\mathbb{Z}^2} \leq \frac{4}{h^2}\|u\|_{\mathbb{Z}^2} \tag{10.18}$$

from which it may be deduced that

$$\|\delta_{\Delta\boxplus} u\|_{\mathbb{Z}^2} = \|\left(\delta_{xx} + \delta_{yy}\right) u\|_{\mathbb{Z}^2} \leq \|\delta_{xx} u\|_{\mathbb{Z}^2} + \|\delta_{yy} u\|_{\mathbb{Z}^2} \leq \frac{8}{h^2}\|u\|_{\mathbb{Z}^2}$$

Boundary terms

When the domain has an edge, boundary terms appear in the summation by parts identities above. Consider, for example, the half plane $\mathcal{D} = \mathbb{Z}^{2,x+}$. Now, instead of the first of (10.16), one has

$$\langle f, \delta_{x-} g\rangle_{\mathbb{Z}^{2,x+}} = -\langle \delta_{x+} f, g\rangle_{\mathbb{Z}^{2,x+}} - \{f, e_{x-} g\}_{(0,\mathbb{Z})} \tag{10.19}$$

Here, the $\{\cdot,\cdot\}$ notation indicates a 1D inner product over the boundary of the region $\mathbb{Z}^{2,x+}$, i.e.,

$$\{f, e_{x-} g\}_{(0,\mathbb{Z})} = \sum_{m=-\infty}^{\infty} h f_{0,m} g_{-1,m}$$

Notice the appearance here of values of the grid function at virtual locations with $l = -1$; such values may be set once boundary conditions have been specified.

Similarly, over the domain $\mathcal{D} = \mathbb{Z}^{2,y+}$, summation by parts becomes

$$\langle f, \delta_{y-} g\rangle_{\mathbb{Z}^{2,y+}} = -\langle \delta_{y+} f, g\rangle_{\mathbb{Z}^{2,y+}} - \{f, e_{y-} g\}_{(\mathbb{Z},0)}$$

The above identities allow the determination of numerical boundary conditions when energy methods are employed. For examples, see the case of the 2D wave equation, in Section 11.2.2, and linear plate vibration, on page 338.

10.2.4 Matrix interpretation of difference operators

As in 1D, it is sometimes useful to represent difference operators in matrix form, especially in the case of implicit schemes which require linear system solution techniques. As a first step, the grid function to be operated on should be "flattened" to a vector. For a grid function $u_{l,m}$, for instance, defined over $\mathcal{D} = \mathbb{Z}^2$, one can stack the columns to create a vector $\mathbf{u}$, as

$$\mathbf{u} = [\ldots, \mathbf{u}_{l-1}^T, \mathbf{u}_l^T, \mathbf{u}_{l+1}^T, \ldots]^T$$

where $\mathbf{u}_l = [\ldots, u_{l,m-1}, u_{l,m}, u_{l,m+1}, \ldots]^T$. (In other words, consecutive vertical strips of the 2D grid function u are lined up end to end in a single-column vector.) Matrix forms of difference operators, then, have a particularly sparse form. Consider, for example, the operators δ_{xx} and δ_{yy}, corresponding to second derivatives in the x and y directions, respectively. In matrix form, and assuming equal grid spacings $h_x = h_y = h$, these look like

$$\mathbf{D}_{xx}^{(2)} = \frac{1}{h^2}\begin{bmatrix} \ddots & \ddots & & & \mathbf{0} \\ \ddots & -2\mathbf{I} & \mathbf{I} & & \\ & \mathbf{I} & -2\mathbf{I} & \mathbf{I} & \\ & & \mathbf{I} & -2\mathbf{I} & \mathbf{I} \\ & \mathbf{0} & & \mathbf{I} & -2\mathbf{I} & \ddots \\ & & & & \ddots & \ddots \end{bmatrix} \qquad \mathbf{D}_{yy}^{(2)} = \begin{bmatrix} \ddots & & & & \\ & \mathbf{D}_{yy}^{(1)} & & & \\ & & \mathbf{D}_{yy}^{(1)} & & \\ & & & \mathbf{D}_{yy}^{(1)} & \\ & & & & \ddots \end{bmatrix}$$

where the superscript (2) indicates that the matrix operators are not to be confused with their 1D counterparts. Here, $\mathbf{I}$ is again the identity matrix, and $\mathbf{D}_{yy}^{(1)}$ is a 1D difference matrix, identical in form to that of $\mathbf{D}_{xx}$ given in (5.17).

Matrix forms of other difference operators follow immediately. The difference operators $\delta_{\Delta\boxplus}$ and $\delta_{\Delta\boxplus,\Delta\boxplus}$, discrete approximations to the Laplacian and biharmonic operators, have the following forms, at interior points in the domain:

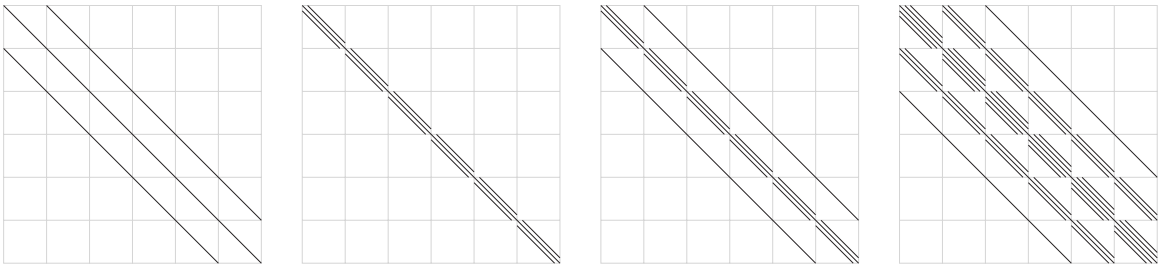
$$\mathbf{D}_{\Delta\boxplus} = \mathbf{D}_{xx}^{(2)} + \mathbf{D}_{yy}^{(2)} \qquad \mathbf{D}_{\Delta\boxplus,\Delta\boxplus} = \mathbf{D}_{\Delta\boxplus}\mathbf{D}_{\Delta\boxplus}$$

It is probably easiest to understand the action of these operators by examining sparsity plots, indicating where non-zero entries occur, as shown in Figure 10.4.

Boundary conditions

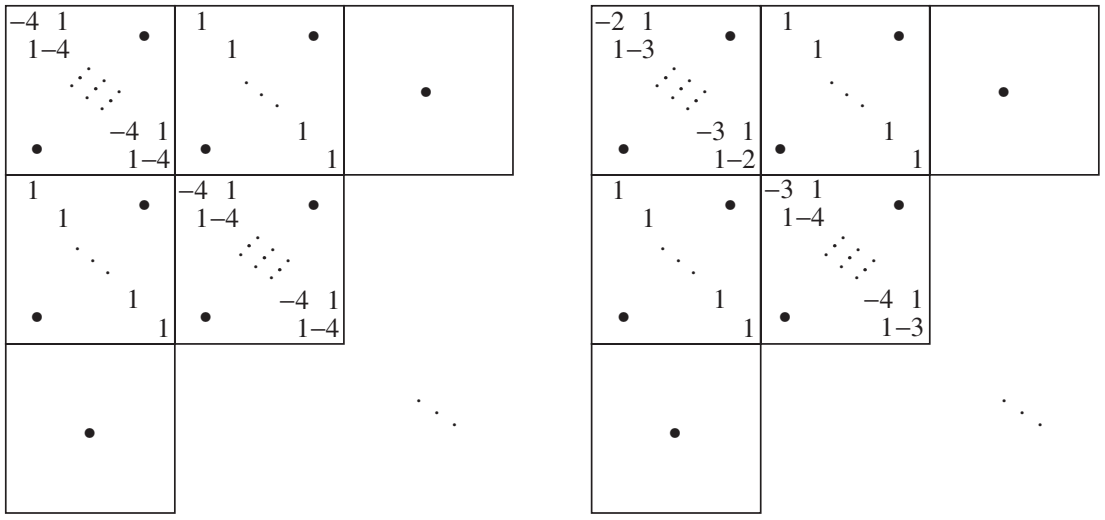
As in 1D, when boundary conditions are taken into account, the matrices are of finite size, though values in the extreme blocks, and also in extreme rows and columns of other blocks, must be modified. From an implementation point of view, it is useful to examine the form of such modifications.

As an example, consider the operator $\delta_{\Delta\boxplus}$, operating on a grid function $u_{l,m}$ defined over the finite rectangular domain $\mathcal{D} = \mathbb{U}^2_{N_x,N_y}$, or, in other words, the set of values of $u_{l,m}$ defined for $0 \le l \le N_x$, $0 \le m \le N_y$. A particularly simple case is that of Dirichlet, or fixed termination, in which case the values of $u_{l,m}$ are set permanently to zero at the domain boundary. The total grid consists of $N_x - 1$ vertical strips of length $N_y - 1$, leaving out the values on the boundary. Thus,



$\mathbf{D}_{xx}^{(2)}$ $\mathbf{D}_{yy}^{(2)}$ $\mathbf{D}_{\Delta\boxplus}$ $\mathbf{D}_{\Delta\boxplus,\Delta\boxplus}$

Figure 10.4 Sparsity patterns for various 2D difference operators written in matrix form, in Cartesian coordinates.



(a) (b)

Figure 10.5 Upper left-hand corners of the matrix representation $\mathbf{D}_{\Delta\boxplus}$ of the discrete Laplacian operator $\delta_{\Delta\boxplus}$ under (a) Dirichlet conditions and (b) Neumann conditions. Values are to be scaled by $1/h^2$, and • indicates zero entries.

if the state vector $\mathbf{u}$ is defined as the concatenation of these vertical strips, the matrix operator $\mathbf{D}_{\Delta\boxplus}$ is as shown in Figure 10.5(a); it consists of $(N_y - 1) \times (N_y - 1)$ blocks. See also Problem 10.4.

As another example, consider the same operator acting on a grid function over the same domain, but under Neumann, or zero normal derivative conditions on all sides:

$$\delta_{x-} u_{0,m} = 0, \qquad m = 0, \ldots, N_y \tag{10.20a}$$

$$\delta_{x+} u_{N_x,m} = 0, \qquad m = 0, \ldots, N_y \tag{10.20b}$$

$$\delta_{y-} u_{l,0} = 0, \qquad l = 0, \ldots, N_x \tag{10.20c}$$

$$\delta_{y+} u_{l,N_y} = 0, \qquad l = 0, \ldots, N_x \tag{10.20d}$$

Now, when operating at a boundary point, the operator makes use of virtual grid points outside the domain. Consider the condition $\delta_{x-} u_{0,m} = 0$, for $0 < m < N_y$ (i.e., excluding the corner points at $l = 0, m = 0$ and $l = 0, m = N_y$). This implies that, in terms of virtual grid points, one may write $u_{-1,m} = u_{0,m}$. Thus in order to evaluate the discrete Laplacian $\delta_{\Delta\boxplus}$ at such a point in terms of values over the grid interior, one has

$$\begin{aligned} \delta_{\Delta\boxplus} u_{0,m} &= \frac{1}{h^2} \left(u_{0,m+1} + u_{0,m-1} + u_{-1,m} + u_{1,m} - 4u_{0,m} \right) \\ &= \frac{1}{h^2} \left(u_{0,m+1} + u_{0,m-1} + u_{1,m} - 3u_{0,m} \right) \end{aligned} \tag{10.21}$$

A similar evaluation may be written for points with $0 < l < N_x$, and $m = 0$. At a domain corner, such as $l = 0, m = 0$, when Neumann conditions are employed along both edges, one has $\delta_{x-} u_{0,0} = \delta_{y-} u_{0,0} = 0$, and thus, for the Laplacian,

$$\delta_{\Delta\boxplus} u_{0,0} = \frac{1}{h^2} \left(u_{0,1} + u_{1,0} - 2u_{0,0} \right) \tag{10.22}$$

The matrix operator $\mathbf{D}_{\Delta\boxplus}$ operates over the full grid, which when written as a vector consists of $N_x + 1$ concatenated vertical strips of length $N_y + 1$ (i.e., the points on the boundary now form

part of the state). The matrix operator appears in Figure 10.5(b), where it is to be noted that the top left-hand block and the top row entries of the other central blocks have slightly modified values. For different approximations to the Neumann condition, the modifications are distinct. See Problem 10.5 and Programming Exercise 10.3. Both the Dirichlet and Neumann conditions presented here arise naturally in the analysis of finite difference schemes for the 2D wave equation—see Section 11.2.2.

As one can imagine, for operators of wider stencil, such as, for example, $\mathbf{D}_{\triangle\boxplus,\triangle\boxplus}$, more boundary conditions are required, and thus more values in the resulting matrix form must be modified. See Problem 10.6 and Programming Exercise 10.5.

10.3 Grid functions and difference operators: radial coordinates

Problems defined over a circular geometry play a large role in musical acoustics, in particular when it comes to the modeling of percussion instruments such as drums and gongs. Though finite element models are often used under such conditions, a finite difference approach is still viable, and can lead to very efficient and easily programmed sound synthesis algorithm designs. A grid function $u^n_{l,m}$, for $(l, m) \in \mathcal{D}$, and $n \geq 0$, represents an approximation to a continuous function $u(r, \theta, t)$, at coordinates $r = lh_r$, $\theta = mh_\theta$, and $t = nk$. In general, the grid spacings h_r and h_θ will not be the same. See Figure 10.6(a). Here, the main domain of interest will be the unit area circle $\mathcal{D} = \mathbb{U}^2_{\circ,N_r,N_\theta}$, the set of points l, m with $0 \leq l \leq N_r$, and $0 \leq m \leq N_\theta - 1$, where $h_r = 1/N_r$ and $h_\theta = 2\pi/N_\theta$. Also useful is the domain $\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}$ which is the same as $\mathbb{U}^2_{\circ,N_r,N_\theta}$ without the central grid location at $l = 0$. At the central grid point, at $l = 0$, for any m, any grid function u is assumed to be single valued, and its value here will be called $u_{0,0}$. A special grid function of interest is $r_l = lh_r$.

Spatial shift operators in the r and θ directions may be defined as

$$e_{r+}u^n_{l,m} = u^n_{l+1,m} \qquad e_{r-}u^n_{l,m} = u^n_{l-1,m} \qquad e_{\theta+}u^n_{l,m} = u^n_{l,m+1} \qquad e_{\theta-}u^n_{l,m} = u^n_{l,m-1}$$

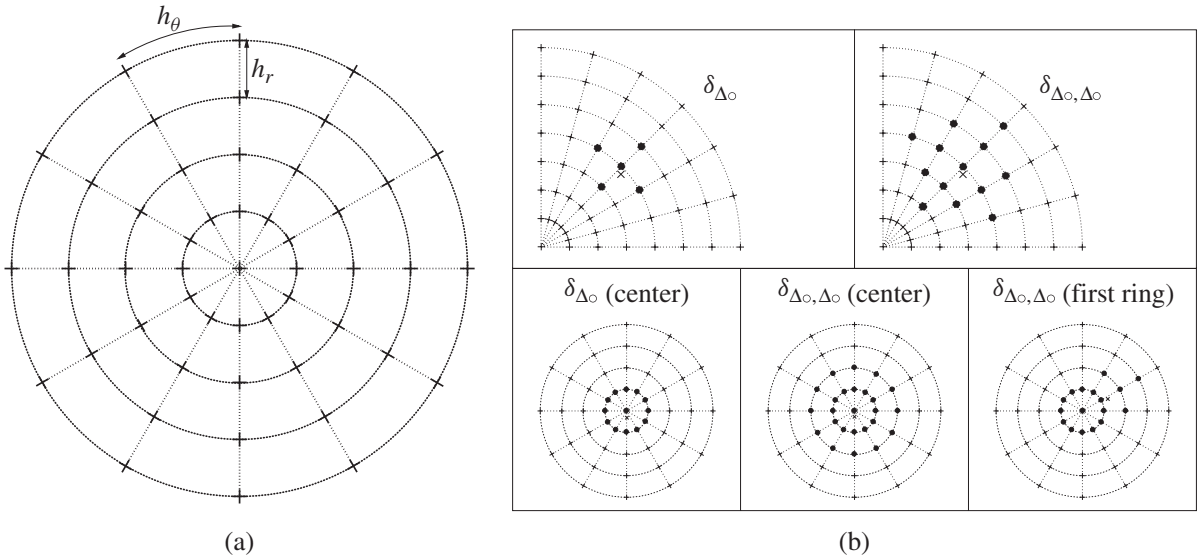


Figure 10.6 (a) Grid in radial coordinates, with radial spacing h_r and angular spacing h_θ. (b) Stencils of the discrete Laplacian $\delta_{\Delta\circ}$ and biharmonic operator $\delta_{\Delta\circ,\Delta\circ}$, with specialized forms near the grid center. In each case, the point at which the operator acts is indicated by a $\times$ symbol.

and forward, backward, and centered difference operators as

$$\delta_{r+} \triangleq \frac{1}{h_r}(e_{r+} - 1) \approxeq \frac{\partial}{\partial r} \qquad \delta_{r-} \triangleq \frac{1}{h_r}(1 - e_{r-}) \approxeq \frac{\partial}{\partial r} \qquad \delta_{r\cdot} \triangleq \frac{1}{2h_r}(e_{r+} - e_{r-}) \approxeq \frac{\partial}{\partial r}$$

$$\delta_{\theta+} \triangleq \frac{1}{h_\theta}(e_{\theta+} - 1) \approxeq \frac{\partial}{\partial \theta} \qquad \delta_{\theta-} \triangleq \frac{1}{h_\theta}(1 - e_{\theta-}) \approxeq \frac{\partial}{\partial \theta} \qquad \delta_{\theta\cdot} \triangleq \frac{1}{2h_\theta}(e_{\theta+} - e_{\theta-}) \approxeq \frac{\partial}{\partial \theta}$$

The operators e_{r-}, δ_{r-}, and $\delta_{r\cdot}$ above are only well defined at grid locations with $l \geq 1$. The operators in θ act on grid points with index m modulo N_θ, e.g., $\delta_{\theta-} u_{l,0} = (u_{l,0} - u_{l,N_\theta - 1})/h_\theta$.

In analogy with the second form in (10.4), one may define a discrete inner product, over the domain $\mathcal{D} = \overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}$, as

$$\langle f, g \rangle^2_{\overline{\mathbb{U}}_{\circ,N_r,N_\theta}} = \sum_{l=1}^{N_r} \sum_{m=0}^{N_\theta - 1} h_\theta h_r r_l f_{l,m} g_{l,m}$$

The discrete Laplacian and biharmonic operators in radial coordinates

The main operator of interest is, as in Cartesian coordinates, the Laplacian, defined in radial coordinates in (10.2). Here is a second-order accurate approximation, as applied to a grid function u:

$$\delta_{\Delta\circ} u = \frac{1}{r}\delta_{r+}((\mu_{r-} r)\delta_{r-} u) + \frac{1}{r^2}\delta_{\theta\theta} u \tag{10.23}$$

where $\delta_{\theta\theta} = \delta_{\theta+}\delta_{\theta-}$. When expanded, this reads as

$$\delta_{\Delta\circ} u_{l,m} = \frac{1}{l h_r^2}\left((l + 1/2)u_{l+1,m} - 2l u_{l,m} + (l - 1/2)u_{l-1,m}\right) + \frac{1}{l^2 h_r^2 h_\theta^2}\left(u_{l,m+1} - 2u_{l,m} + u_{l,m-1}\right)$$

The above difference operation holds at grid points $u_{l,m}$ with $l > 0$. At the center point, a special form is necessary [121]:

$$\delta_{\Delta\circ} u_{0,0} = \frac{4}{N_\theta h_r^2} \sum_{m=0}^{N_\theta - 1} \left(u_{1,m} - u_{0,0}\right) \tag{10.24}$$

It is not difficult to show that this operator indeed approximates the Laplacian to second-order accuracy at the domain center—see Problem 10.7. The matrix form of this operator may be constructed without much difficulty—see Programming Exercise 10.6.

The biharmonic operator in radial coordinates $\delta_{\Delta\circ,\Delta\circ}$ may be defined, at interior points in a domain, as

$$\delta_{\Delta\circ,\Delta\circ} u_{l,m} = \delta_{\Delta\circ}\delta_{\Delta\circ} u_{l,m} \qquad 2 \leq l \leq N_r - 2 \tag{10.25}$$

At grid points near the boundary, the form of the operator must be specialized to include boundary conditions. As for the case of the Laplacian, however, in the neighborhood of the center, at grid locations l, m, with $l = 0, 1$, special forms are necessary:

$$\delta_{\Delta\circ,\Delta\circ} u_{0,0} = \frac{16}{3N_\theta h_r^4} \sum_{m=0}^{N_\theta - 1} \left(u_{2,m} - 4u_{1,m} + 3u_{0,0}\right) \tag{10.26a}$$

$$\delta_{\Delta\circ,\Delta\circ} u_{1,m} = \frac{4}{N_\theta h_r^2} \sum_{m=0}^{N_\theta - 1} \delta_{\Delta\circ} u_{1,m} - \delta_{\Delta\circ} u_{0,0} \tag{10.26b}$$

Slightly different settings have been proposed—see [12]. See also Programming Exercise 10.7.

Summation by parts

For circular domains, identities analogous to integration by parts are available, but one must pay special attention to the definition of the Laplacian at the center point. As a practically important example, consider the inner product of a grid function f and the discrete Laplacian applied to a grid function g, over the domain $\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}$, the unit circle without its center grid point. In this case, the definition (10.23) holds at all points in the domain, and one has

$$\langle f, \delta_{\Delta\circ} g\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} = -\left\langle \delta_{r-} f, \frac{\mu_{r-} r}{r}\delta_{r-} g\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} - \left\langle \delta_{\theta-} f, \frac{1}{r^2}\delta_{\theta-} g\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} + \mathfrak{b}_{\text{outer}} - \mathfrak{b}_{\text{inner}}$$

where

$$\mathfrak{b}_{\text{outer}} = \{f, \mu_{r+} r\delta_{r+} g\}_{(N_r,[0,N_\theta-1])} \triangleq \sum_{m=0}^{N_\theta-1} h_\theta f_{N_r,m}\mu_{r+} r_{N_r}\delta_{r+} g_{N_r,m}$$

$$\mathfrak{b}_{\text{inner}} = \{e_{r-} f, \mu_{r-} r\delta_{r-} g\}_{(1,[0,N_\theta-1])} \triangleq \sum_{m=0}^{N_\theta-1} h_\theta f_{0,0}\mu_{r-} r_1\delta_{r-} g_{1,m}$$

The term $\mathfrak{b}_{\text{outer}}$ clearly corresponds to the boundary term at the domain edge in (10.8a). But the term $\mathfrak{b}_{\text{inner}}$ results purely from the choice of the domain of summation. But it may be rewritten as follows, using single-valuedness of f and g at the grid location $l = 0, m$, and the definition of the discrete Laplacian at the central grid point, from (10.24):

$$\mathfrak{b}_{\text{inner}} = \frac{f_{0,0} h_\theta}{2}\sum_{m=0}^{N_\theta-1}\left(g_{1,m} - g_{0,0}\right) = \frac{\pi h_r^2}{4} f_{0,0}\delta_{\Delta\circ} g_{0,0}$$

Thus, finally, one has

$$\langle f, \delta_{\Delta\circ} g\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} + \frac{\pi h_r^2}{4} f_{0,0}\delta_{\Delta\circ} g_{0,0} = -\left\langle \delta_{r-} f, \frac{\mu_{r-} r}{r}\delta_{r-} g\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} - \left\langle \delta_{\theta-} f, \frac{1}{r^2}\delta_{\theta-} g\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} + \mathfrak{b}_{\text{outer}} \tag{10.27}$$

Bounds

As before, bounds are available relating the norms of grid functions under difference operators to norms of the grid functions themselves. Here are two of interest in the analysis of schemes involving the discrete Laplacian:

$$\|\delta_{\theta-} f\|^2_{\overline{\mathbb{U}}_{\circ,N_r,N_\theta}} \le \frac{2}{h_\theta}\|f\|_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} \qquad \|\sqrt{\mu_{r-} r/r}\,\delta_{r-} f\|_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} \le \frac{4}{h_r^2}\|f\|_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} + 2\pi f_{0,0}^2 \tag{10.28}$$

10.4 Problems

Problem 10.1 *The stencils of various 2D difference operators are shown in Figure 10.2. Not shown, however, are the coefficients to be applied at each point included in the stencil. Determine these for all the operators shown in the figure, assuming equal grid spacings $h_x = h_y = h$. (For the*

operator $\delta_{\Delta\boxplus}$, for example, the coefficients from left to right, and top to bottom, are: $1/h^2$, $1/h^2$, $-4/h^2$, $1/h^2$, and $1/h^2$.)

There are certain operators which appear in the figure which are not defined in the main body of the text, namely averaging and mixed derivative operators:

$$\mu_{x+} = \frac{1}{2}(e_{x+} + 1) \qquad \mu_{x-} = \frac{1}{2}(1 + e_{x-}) \qquad \mu_{y+} = \frac{1}{2}\left(e_{y+} + 1\right) \qquad \mu_{y-} = \frac{1}{2}\left(1 + e_{y-}\right)$$

$$\delta_{x+y+} = \delta_{x+}\delta_{y+} \qquad \delta_{x+y-} = \delta_{x+}\delta_{y-} \qquad \delta_{x-y+} = \delta_{x-}\delta_{y+} \qquad \delta_{x-y-} = \delta_{x-}\delta_{y-}$$

Problem 10.2 *Given a general bilinear function $f(p, q)$ in the variables p and q*

$$f(p, q) = a + bp + cq + dpq$$

defined over the square region $0 \leq p, q \leq 1$, for constants a, b, c, and d, show that f takes on its maximum and minimum values at the corners of the domain.

Problem 10.3 *Recalling the analysis of the anisotropy of the operator $\delta_{\Delta\boxplus}$ in Section 10.2.1, consider the nine-point Laplacian operator as given in (10.10), which depends on the free parameter α, and show that its expansion, in terms of wavenumber components β_x and β_y, is*

$$\delta_{\Delta\alpha} \triangleq \alpha\delta_{\Delta\boxplus} + (1-\alpha)\delta_{\Delta\boxtimes} \Longrightarrow -\left(\beta_x^2 + \beta_y^2\right) + \frac{h^2}{12}\left(\beta_x^4 + 6(1-\alpha)\beta_x^2\beta_y^2 + \beta_y^4\right) + O(h^4)$$

For an arbitrary choice of α, can the $O(h^2)$ term be written in terms of the wavenumber magnitude $|\boldsymbol{\beta}| = (\beta_x^2 + \beta_y^2)^{1/2}$ alone? If not, is there a value (or values) of α such that it can be? If you can find such a value of α, then the anisotropic behavior of the parameterized operator will exhibit itself only at fourth order (though the operator remains a second-order accurate approximation to the Laplacian).

Problem 10.4 *Find the matrix form $\mathbf{D}_{\Delta\alpha}$ of the operator $\delta_{\Delta\alpha}$, assuming that it acts on a grid function defined over $\mathbb{U}^2_{N_x,N_y}$, under Dirichlet conditions.*

Problem 10.5 *Consider the operator $\delta_{\Delta\boxplus}$, operating over the rectangular domain $\mathbb{U}^2_{N_x,N_y}$, which is of size $(N_x+1)\times(N_y+1)$ points. A centered zero derivative, or Neumann condition, may be written as*

$$\begin{aligned}
\delta_{x\cdot}u_{0,m} &= 0, \qquad m = 0, \ldots, N_y \\
\delta_{x\cdot}u_{N_x,m} &= 0, \qquad m = 0, \ldots, N_y \\
\delta_{y\cdot}u_{l,0} &= 0, \qquad l = 0, \ldots, N_x \\
\delta_{y\cdot}u_{l,N_y} &= 0, \qquad l = 0, \ldots, N_x
\end{aligned}$$

Supposing that the state vector is defined as the concatenation of $N_x + 1$ vertical strips of length $N_y + 1$, write the matrix form of the operator under the conditions above. Assume that $h_x = h_y = h$. Again, as in the case of the non-centered condition discussed beginning on page 296, your matrix will consist of $(N_y+1)\times(N_y+1)$ blocks, each of which is Toeplitz except in the extreme rows and columns. See also Programming Exercise 10.3.

Problem 10.6 *Consider the discrete biharmonic operator $\delta_{\Delta\boxplus,\Delta\boxplus}$, operating over the rectangular domain $\mathbb{U}^2_{N_x,N_y}$, which is of size $(N_x+1)\times(N_y+1)$ points. Consider the following two sets of boundary conditions:*

Clamped:

$$\begin{aligned} \delta_{x-} u_{0,m} &= 0, & u_{0,m} &= 0, & m &= 0, \ldots, N_y \\ \delta_{x+} u_{N_x,m} &= 0, & u_{N_x,m} &= 0, & m &= 0, \ldots, N_y \\ \delta_{y-} u_{l,0} &= 0, & u_{l,0} &= 0, & l &= 0, \ldots, N_x \\ \delta_{y+} u_{l,N_y} &= 0, & u_{l,N_y} &= 0, & l &= 0, \ldots, N_x \end{aligned}$$

Simply supported:

$$\begin{aligned} \delta_{xx} u_{0,m} &= 0, & u_{0,m} &= 0, & m &= 0, \ldots, N_y \\ \delta_{xx} u_{N_x,m} &= 0, & u_{N_x,m} &= 0, & m &= 0, \ldots, N_y \\ \delta_{yy} u_{l,0} &= 0, & u_{l,0} &= 0, & l &= 0, \ldots, N_x \\ \delta_{yy} u_{l,N_y} &= 0, & u_{l,N_y} &= 0, & l &= 0, \ldots, N_x \end{aligned}$$

First, find the explicit form of the stencil of the operator, including all coefficients, when applied at any point directly adjacent to the boundary—take special care when near corners! Then, write the matrix form $\mathbf{D}_{\Delta\boxplus,\Delta\boxplus}$ *of the operator, assuming that it acts on a vector consisting of concatenated vertical strips of the grid function* $u_{l,m}$*. You may leave out values on the boundary itself (which are constrained to be zero in either case), so your matrix will be square and of size* $(N_x - 1)(N_y - 1) \times (N_x - 1)(N_y - 1)$*.*

Problem 10.7 *Consider the discrete Laplacian operator, in radial coordinates, acting on a grid function* $u_{l,m}$ *at the central grid point* $l = 0$*,* $m = 0$*. Given the definition of this operator, from (10.24), prove that it is indeed a second-order accurate approximation to the Laplacian. In order to do this, consider the values of the grid function* $u_{1,m}$ *which appear in the definition to be values of a continuous function at the location* $x = h_r \cos(mh_\theta)$*,* $y = h_r \sin(mh_\theta)$*, and perform Taylor expansions about the point* $x = 0$*,* $y = 0$*.*

10.5 Programming exercises

Programming Exercise 10.1 *Create a Matlab function which, for a given rectangular grid function, calculates an interpolated value at coordinates* x_o*,* y_o*, using either truncation or bilinear interpolation, as described in Section 10.2.1. Assume equal grid spacings in the x and y directions, and thus that the aspect ratio may be determined from the dimensions of the input grid function. Make sure that your code takes account of whether the input grid function includes values on its boundary—this must also be specified as an input parameter.*

Programming Exercise 10.2 *Consider the parameterized nine-point approximation* $\delta_{\Delta\alpha}$ *to the Laplacian, as defined in (10.10). When applied to a test function of the form* $u_{l,m} = e^{jh(\beta_x l + \beta_y m)}$*, it behaves as*

$$\delta_{\Delta\alpha} u = \frac{-4}{h^2} \left(\sin^2(\beta_x h/2) + \sin^2\left(\beta_y h/2\right) - 2(1-\alpha)\sin^2(\beta_x h/2)\sin^2(\beta_y h/2) \right) u = F_\alpha(\beta_x, \beta_y) u$$

Create a Matlab script which plots the function $-F_\alpha(\beta_x, \beta_y)/|\boldsymbol{\beta}|^2$ *as a function of* $\beta_x h$ *and* $\beta_y h$*, for* $-\pi \leq \beta_x h, \beta_y h \leq \pi$ *as a surface, for various values of the free parameter* α*. (This function is a comparison of the dispersive behavior of the difference operator with the continuous operator* Δ *as a function of wavenumber.) Verify that the difference operator is approximately isotropic when* $\alpha \approxeq 2/3$*.*

Programming Exercise 10.3 *Create a Matlab function which generates difference matrices* $\mathbf{D}_{\Delta\boxplus}$ *corresponding to the discrete Laplacian operator* $\delta_{\Delta\boxplus}$*, operating over the rectangular domain* $\mathbb{U}^2_{N_x,N_y}$*. Suppose also that the aspect ratio is* $\epsilon = N_x/N_y$*, and that the grid spacing is* $h = \sqrt{\epsilon}/N_x$*. Your code thus depends only on* N_x *and* N_y*, and should generate matrices corresponding to fixed*

or Dirichlet conditions, non-centered Neumann conditions, and centered Neumann conditions—the first two cases are discussed in Section 10.2.4 and the third in Problem 10.5. Note that your output matrix will be square, and of size $(N_x - 1)(N_y - 1) \times (N_x - 1)(N_y - 1)$ *in the first case, and* $(N_x + 1)(N_y + 1) \times (N_x + 1)(N_y + 1)$ *in the latter two. Be sure that the matrix is generated in sparse form—make use of the function* `sparse` *for this purpose.*

Programming Exercise 10.4 *Create a Matlab function which generates a difference matrix* $\mathbf{D}_{\Delta\alpha}$ *corresponding to the parameterized nine-point discrete Laplacian operator* $\delta_{\Delta\alpha}$*, operating over the rectangular domain* $\mathbb{U}^2_{N_x,N_y}$*, under Dirichlet conditions. Your code will necessarily depend on the free parameter* α*.*

Programming Exercise 10.5 *Create a Matlab function which generates difference matrices* $\mathbf{D}_{\Delta\boxplus,\Delta\boxplus}$ *corresponding to the discrete biharmonic operator* $\delta_{\Delta\boxplus,\Delta\boxplus}$*, operating over the rectangular domain* $\mathbb{U}^2_{N_x,N_y}$*. As above, suppose also that the aspect ratio is* $\epsilon = N_x/N_y$*, and that the grid spacing is* $h = \sqrt{\epsilon}/N_x$*. Your code should generate matrices corresponding to clamped and simply supported conditions, as described in Problem 10.6. Your output matrix will be square and of size* $(N_x - 1)(N_y - 1) \times (N_x - 1)(N_y - 1)$*.*

Programming Exercise 10.6 *Consider the discrete Laplacian operator* $\delta_{\Delta\circ}$ *in radial coordinates, operating over the domain* $\mathbb{U}^2_{\circ,N_r,N_\theta}$*, which is defined by (10.23) at grid points* $l > 0$ *and by (10.24) at* $l = 0$*. Create a Matlab function which generates, for a given* N_r *and* N_θ*, the matrix form* $\mathbf{D}_\circ$ *of the operator, under fixed conditions at the outer edge of the domain, at* $l = N_r$*.*

In preparation for this, suppose that the values of the grid function $u_{l,m}$ *on which* $\mathbf{D}_{\Delta\circ}$ *operates are written as a vector* $\mathbf{u}$ *consisting of the central value* $u_{0,0}$*, followed by concatenated concentric rings of the grid function, or as*

$$\mathbf{u} = [u_{0,0}, u_{1,0}, \ldots, u_{1,N_\theta-1}, \ldots, u_{N_r-1,0}, \ldots, u_{N_r-1,N_\theta-1}]^T$$

As such, the matrix $\mathbf{D}_{\Delta\circ}$ *will be of size* $(N_\theta(N_r - 1) + 1) \times (N_\theta(N_r - 1) + 1)$*.*

Programming Exercise 10.7 *Consider the discrete biharmonic operator* $\delta_{\Delta\circ,\Delta\circ}$ *in radial coordinates, operating over the domain* $\mathbb{U}^2_{\circ,N_r,N_\theta}$*, which is defined by (10.25) at grid points* $l > 1$ *and by (10.26) at* $l = 0$ *and* $l = 1$*. Create a Matlab function which generates, for a given* N_r *and* N_θ*, the matrix form* $\mathbf{D}_{\Delta\circ,\Delta\circ}$ *of the operator.*

Assume clamped boundary conditions at the domain edge, i.e., $u_{N_r,m} = \delta_{r+}u_{N_r,m} = 0$*, so that as in the case of the Laplacian in the previous exercise, the matrix operates only over values of the grid function for* $l < N_r$*.*

11

The 2D wave equation

A good starting point in the investigation of 2D synthesis is, as in 1D, the wave equation, introduced in Section 11.1, which serves as a useful test problem for the vibration of membranes as well as room acoustics, and also as another good point of comparison for the various physical modeling synthesis techniques, including finite difference schemes (Section 11.2 and Section 11.3), digital waveguide meshes (Section 11.4), lumped networks (Section 11.5), and modal methods (Section 11.6). Finally, finite difference schemes are developed in radial coordinates in Section 11.7. The previous chapter serves as a reference for the techniques to be discussed here and in the following two chapters.

11.1 Definition and properties

The wave equation in one spatial dimension (6.1) may be directly generalized to 2D as

$$u_{tt} = c^2 \Delta u \tag{11.1}$$

Here, u is the dependent variable of interest, c is a wave speed, and Δ is the 2D Laplacian operator as discussed in Chapter 10. The problem is assumed defined over some 2D domain $\mathcal{D}$. When the spatial coordinates are scaled using a characteristic length L, the 2D wave equation is of the form

$$u_{tt} = \gamma^2 \Delta u \tag{11.2}$$

where, again, $\gamma = c/L$ is a parameter with units of frequency. In some cases, a good choice of the scaling parameter L for problems defined over a finite region $\mathcal{D}$ is $L = \sqrt{|\mathcal{D}|}$, so that (11.2) is then defined over a region of unit area. For other domains, such as the circle of radius R, it may be more convenient to choose $L = R$, so that the problem is then defined over a unit circle.

The 2D wave equation, as given above, is a simple approximation to the behavior of a vibrating membrane; indeed, a lumped model of membrane is often employed in order to derive the 2D wave equation itself—see Section 11.5. It also can serve as a preliminary step toward the treatment of room acoustics problems (where, in general, a 3D wave equation would be employed); direct solution of the 2D and 3D wave equations for room modeling has been employed for some time [56], and artificial reverberation applications are currently an active research topic [249, 250, 28, 208]. Other applications include articulatory vocal tract modeling [246, 247], and detailed studies of wind

Numerical Sound Synthesis: Finite Difference Schemes and Simulation in Musical Acoustics Stefan Bilbao
 ISBN: 978-0-470-51046-9

instrument bores [258, 254, 255]. It will be assumed, for the moment, that the 2D wave equation is defined over $\mathbb{R}^2$, so that a discussion of boundary conditions may be postponed.

The wave equation, like all PDEs of second order in time, must be initialized with two functions (see Section 6.1.3); in Cartesian coordinates, for example, one would set, normally, $u(x, y, 0) = u_0(x, y)$ and $(\partial u/\partial t)(x, y, 0) = v_0(x, y)$. In the case of a membrane, as for the string, the first condition corresponds to a pluck and the second to a strike. A useful all-purpose initializing distribution is a 2D raised cosine, of the form

$$c_{\text{rc}}(x, y) = \begin{cases} \frac{c_0}{2}\left(1 + \cos\left(\pi\sqrt{(x - x_0)^2 + (y - y_0)^2}/r_{\text{hw}}\right)\right), & \sqrt{(x - x_0)^2 + (y - y_0)^2} \leq r_{\text{hw}} \\ 0, & \sqrt{(x - x_0)^2 + (y - y_0)^2} > r_{\text{hw}} \end{cases} \tag{11.3}$$

which has amplitude c_0, half-width r_{hw}, and is centered at coordinates (x_0, y_0). Such a distribution can be used in order to model both plucks and strikes. Of course, in full models of percussion instruments, a model of the excitation mechanism, such as a mallet, should be included as well—see Section 12.3 where this topic is taken up in the more musically interesting setting of plate vibration.

A pair of simulation results, under plucked and struck conditions, is illustrated in Figure 11.1. The main feature which distinguishes the behavior of solutions to the 2D wave equation from the 1D case is the reduction in amplitude of the wave as it evolves, due to spreading effects. Related to this is the presence of a "wake" behind the wavefront, which is absent in the 1D case, as well as the fact that, even when simple boundary conditions are employed, the solution is not periodic—reflections from domain boundaries occur with increasing frequency, as illustrated in Figure 11.2(a). One might guess, from these observations alone, that the efficiency of the digital waveguide formulation for the 1D case, built around waves which travel without distortion, will disappear in this case—this is in fact true, as will be discussed in Section 11.4.

11.1.1 Phase and group velocity

Assuming, in Cartesian coordinates, a plane-wave solution of the form

$$u(x, y, t) = e^{st + j(\beta_x x + \beta_y y)}$$

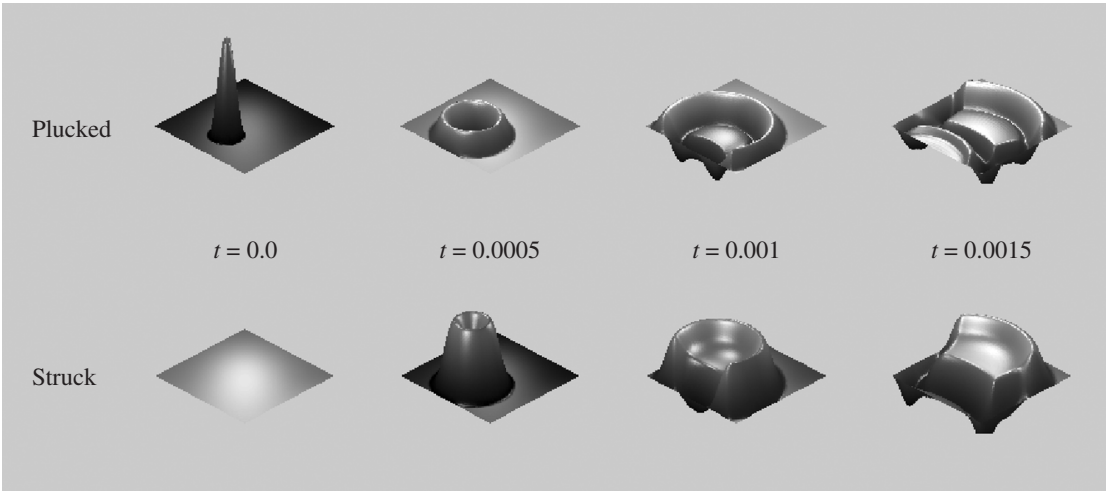


Figure 11.1 Time evolution of solutions to the 2D wave equation, defined over a unit square, with fixed boundary conditions. In this case, $\gamma = 400$, and snapshots of the displacement are shown at times as indicated, under plucked conditions (top row) and struck conditions (bottom row). In the first case, the initial displacement condition is a raised cosine of half-width 0.15, centered at coordinates $x = 0.3$, $y = 0.5$, and, in the second, the initial velocity profile is a raised cosine of half-width 0.15, centered at the same location.

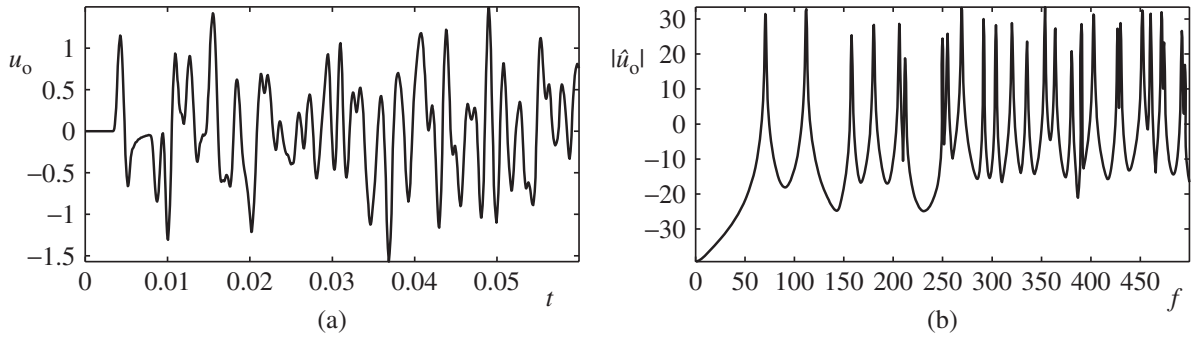


Figure 11.2 Solution of the 2D wave equation, with $\gamma = 100$, defined over a unit square, in response to a plucked excitation, in the form of a raised cosine, centered at $x = 0.7$, $y = 0.5$, of amplitude 10 and half-width 0.2. (a) Time response, at location $x = 0.3$, $y = 0.7$. (b) Its frequency spectrum.

leads to the characteristic equation and dispersion relation

$$s^2 = -\gamma^2 |\boldsymbol{\beta}|^2 \qquad \longrightarrow \qquad \omega = \pm\gamma |\boldsymbol{\beta}|$$

and, thus, to expressions for the phase and group velocity,

$$v_\phi = \gamma \qquad\qquad v_g = \gamma$$

Thus, just as in 1D, all wavelike components travel at the same speed. Note that as the Laplacian operator is isotropic, wave speed is independent not only of frequency or wavenumber, but also of direction.

In 1D, for LSI systems, it is often possible to rewrite the phase and group velocities as functions of frequency alone. While this continues to be possible for isotropic systems in 2D, it is not possible when the system exhibits anisotropy. Furthermore, most numerical methods exhibit spurious anisotropy, and thus the only way to analyze such behavior is through expressions of wavenumber.

11.1.2 Energy and boundary conditions

The 2D wave equation, like its 1D counterpart, is lossless. This can be seen, as usual, by writing the energy for the system, which is conserved. Assuming (11.2) to be defined over the infinite plane $\mathcal{D} = \mathbb{R}^2$, taking an inner product with u_t in Cartesian coordinates followed by integration by parts leads to

$$\langle u_t, u_{tt} \rangle_{\mathcal{D}} = \gamma^2 \langle u_t, \Delta u \rangle_{\mathcal{D}} = \gamma^2 \langle u_t, u_{xx} \rangle_{\mathcal{D}} + \gamma^2 \langle u_t, u_{yy} \rangle_{\mathcal{D}} = -\gamma^2 \langle u_{xt}, u_x \rangle_{\mathcal{D}} - \gamma^2 \langle u_{yt}, u_y \rangle_{\mathcal{D}}$$

or

$$\frac{d\mathfrak{H}}{dt} = 0 \qquad \text{with} \qquad \mathfrak{H} = \mathfrak{T} + \mathfrak{V}$$

and

$$\mathfrak{T} = \frac{1}{2} \|u_t\|_{\mathcal{D}}^2 \qquad \mathfrak{V} = \frac{\gamma^2}{2} \left(\|u_x\|_{\mathcal{D}}^2 + \|u_y\|_{\mathcal{D}}^2 \right) = \frac{\gamma^2}{2} \| |\nabla u| \|_{\mathcal{D}}^2 \tag{11.4}$$

where ∇ signifies the gradient operation. Such a result is obviously independent of the chosen coordinate system. The expression is, again, non-negative, and leads immediately to bounds on the growth of the norm of the solution, by exactly the same methods as discussed in Section 6.1.8.

Edges

In order to examine boundary conditions at, for instance, a straight edge, consider (11.1) defined over the semi-infinite region $\mathcal{D} = \mathbb{R}^{2,x+}$, as discussed in Section 10.1.2. Through the same manipulations above, one arrives at the energy balance

$$\frac{d\mathfrak{H}}{dt} = \mathfrak{B} \qquad \text{with} \qquad \mathfrak{H} = \mathfrak{T} + \mathfrak{V} \tag{11.5}$$

where $\mathfrak{T}$ and $\mathfrak{V}$ are as defined in (11.4) above. The boundary term $\mathfrak{B}$, depending only on values of the solution at $x = 0$, is

$$\mathfrak{B} = -\gamma^2 \{u_x, u_t\}_{(0,\mathbb{R})} \triangleq -\gamma^2 \int_{-\infty}^{\infty} u_x(0, y', t) u_t(0, y', t) dy' \tag{11.6}$$

Thus, the natural extensions of the lossless Dirichlet and Neumann conditions (6.18) and (6.19) for the 1D wave equation are

$$u(0, y, t) = 0 \qquad \text{or} \qquad u_x(0, y, t) = 0 \qquad\qquad \text{at} \quad x = 0$$

in which case $\mathfrak{B}$ vanishes. In the case of the membrane, where u is a displacement, these correspond to fixed and free conditions, respectively. For room acoustics problems, in which case u is a pressure, the Neumann condition corresponds to a rigid wall termination. As in the case of the 1D wave equation, it should be obvious that many other terminations, including those of lossless energy-storing type and lossy conditions, possibly nonlinear, and combinations of these can be arrived at through the above analysis. One particularly interesting family of conditions, useful in modeling terminations in this setting of room acoustics [208], appears in Problem 11.1.

Corners

An extra concern, especially when working in Cartesian coordinates in 2D, is the domain corner. Though it is not particularly problematic in the case of the wave equation, for more complex systems, such as the stiff plate, one must be very careful with such points, especially in the discrete setting. To this end, consider the wave equation defined over the quarter plane $\mathcal{D} = \mathbb{R}^{2,x+y+}$. The energy balance is again as in (11.5), but the boundary term is now

$$\begin{aligned} \mathfrak{B} &= -\gamma^2 \int_0^{\infty} u_x(0, y', t) u_t(0, y', t) dy' - \gamma^2 \int_0^{\infty} u_y(x', 0, t) u_t(x', 0, t) dx' \\ &= -\gamma^2 \{u_x, u_t\}_{(0,\mathbb{R}^+)} - \gamma^2 \{u_y, u_t\}_{(\mathbb{R}^+,0)} \end{aligned}$$

The system is lossless under a combination of the conditions

$$u_x(0, y, t) = 0 \quad \text{or} \quad u_t(0, y, t) = 0 \qquad \text{at} \qquad x = 0 \tag{11.7a}$$

$$u_y(x, 0, t) = 0 \quad \text{or} \quad u_t(x, 0, t) = 0 \qquad \text{at} \qquad y = 0 \tag{11.7b}$$

It is thus possible to choose distinct conditions over each edge of the domain; notice, however, that if free or Neumann conditions are chosen on both edges, then at the domain corner $x = y = 0$, both such conditions must be enforced.

Though only a quarter plane has been analyzed here, it should be clear that such results extend to the case of a rectangular domain—see Problem 11.2.

11.1.3 Modes

The modal decomposition of solutions to the 2D wave equation under certain simple geometries and boundary conditions is heavily covered in the musical acoustics literature—see, e.g., standard texts such as [136, 244]. It is useful, however, to at least briefly review such a decomposition in the case of the rectangular domain.

As in the case of the 1D wave equation (see Section 6.1.11) and the ideal bar equation (see Section 7.1.3), one may assume an oscillatory solution to the 2D wave equation of the form $u(x, y, t) = e^{j\omega t}U(x, y)$, leading to

$$-\omega^2 U = \gamma^2 \Delta U$$

The wave equation defined over a unit area rectangle $\mathbb{U}_\epsilon^2$ of aspect ratio ϵ and under fixed boundary conditions is separable, and the Fourier series solution to the above equation is

$$U_{p,q}(x, y) = \sin(p\pi x/\sqrt{\epsilon}) \sin(q\pi\sqrt{\epsilon}y) \qquad \omega_{p,q} = \pi\gamma\sqrt{p^2/\epsilon + \epsilon q^2} \tag{11.8}$$

The modal functions are illustrated in Figure 11.3, in the case of a square domain. (It is worth pointing out that, under some choices of the aspect ratio ϵ, it is possible for modal frequencies corresponding to distinct modes to coincide or become degenerate.)

Given the above expression for the modal frequencies $\omega_{p,q} = 2\pi f_{p,q}$, it is not difficult to show that the number of degrees of freedom, for frequencies less than or equal to $f_s/2$, will be[1]

$$N_m(f_s/2) = \frac{\pi f_s^2}{2\gamma^2} \tag{11.9}$$

See Problem 11.3. Twice this number is, as before, the number of degrees of freedom of the system when it is simulated at a sample rate f_s. This indicates that the density of modes increases

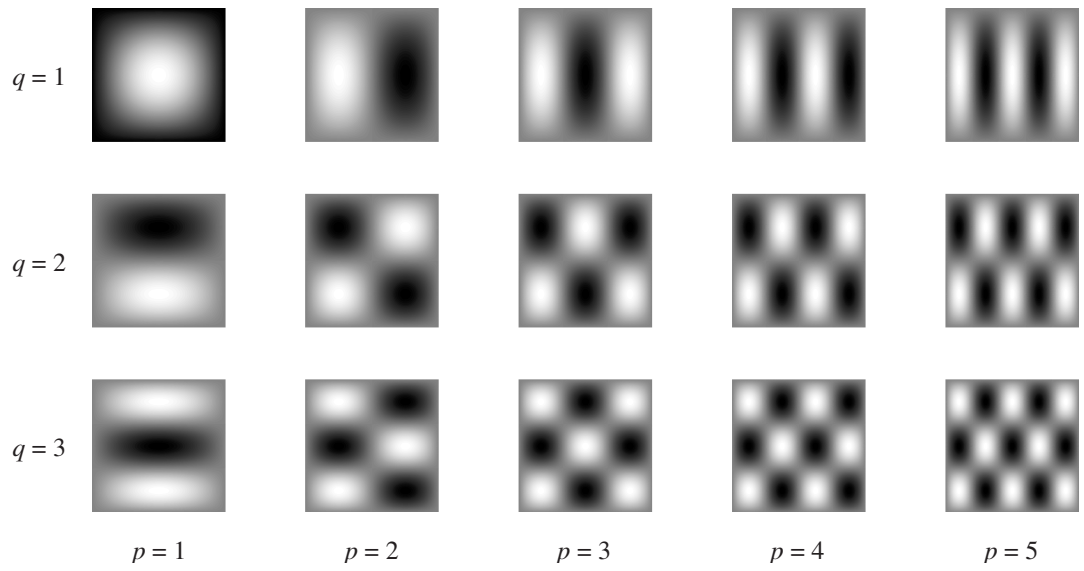


Figure 11.3 Modal shapes $U_{p,q}(x, y)$ for a square membrane, under fixed boundary conditions. In all cases, dark and light areas correspond to modal maxima and minima.

[1] This expression is approximate, and does not take into account modal degeneracy. It is correct in the limit of high f_s. See [243] for exact expressions for the numbers of modes in 3D.

strongly with frequency, which is easily visible in a spectral plot of output taken from the solution to the 2D wave equation, as shown in Figure 11.2(b). This is to be compared to the situation for the 1D wave equation, where modal density is roughly constant—see Section 6.1.11. Serious (and not unexpected) ramifications in terms of computational complexity result, as will be discussed shortly. For more on synthesis for the 2D wave equation using modal methods, see Section 11.6.

This modal analysis can be directly extended to the case of the wave equation in 3D, defined over a suitable region such as a cube; such analysis yields useful information about modal densities in the context of room acoustics. See Problem 11.4. Another system which may be analyzed in this way is the membrane coupled to a cavity, which serves as a simple model of a drum-like instrument—see Problem 11.5.

11.2 A simple finite difference scheme

Consider the wave equation in Cartesian coordinates. The simplest possible finite difference scheme employs a second difference in time, and a five-point Laplacian approximation $\delta_{\triangle\boxplus}$, as defined in Section 10.2:

$$\delta_{tt} u = \gamma^2 \delta_{\triangle\boxplus} u \tag{11.10}$$

where $u = u^n_{l,m}$ is a 2D grid function representing an approximation to the continuous solution $u(x, y, t)$ at $x = lh_x$, $y = mh_y$, $t = nk$, for integer l, m, and n.

The case of equal grid spacings $h_x = h_y = h$ is the most straightforward in terms of analysis and implementation; expanding out the operator notation above leads to

$$u^{n+1}_{l,m} = \lambda^2 \left(u^n_{l+1,m} + u^n_{l-1,m} + u^n_{l,m+1} + u^n_{l,m-1}\right) + 2\left(1 - 2\lambda^2\right) u^n_{l,m} - u^{n-1}_{l,m} \tag{11.11}$$

where again, as in 1D, the Courant number λ is defined as

$$\lambda = \frac{k\gamma}{h}$$

Notice that under the special choice of $\lambda = 1/\sqrt{2}$ (this case is of particular relevance to the so-called digital waveguide mesh, to be discussed in Section 11.4), the recursion above may be simplified to

$$u^{n+1}_{l,m} = \frac{1}{2}\left(u^n_{l+1,m} + u^n_{l-1,m} + u^n_{l,m+1} + u^n_{l,m-1}\right) - u^{n-1}_{l,m} \tag{11.12}$$

which requires only a single multiplication by the factor $1/2$, and one fewer addition than the general form (11.11). Again, as in the case of the special difference scheme for the 1D wave equation (6.54), the value at the central grid point at l, n is no longer employed. Related to this is the interesting observation that the scheme may be decomposed into two independent schemes operating over the black and white squares on a "checkerboard" grid. See Problem 11.6.

Scheme (11.10) continues to hold if the grid spacings are not the same, though the stability condition (discussed below) must be altered, and computational complexity increases. See Problem 11.7. There is a slight advantage if such a scheme is to be used for non-square regions, but perhaps not enough to warrant its use in synthesis applications, except for high values of γ, where the grid is necessarily coarse, and truncation becomes an issue. See Programming Exercise 11.1.

Output may be drawn from the scheme using 2D interpolation operators—see Section 10.2.1. For example, one may take output from a position x_o, y_o as $u_o = I(x_o, y_o)u$. For a static output

location, a simple truncated interpolant is probably sufficient, but for moving output locations, a bilinear interpolant is a better choice—see Programming Exercise 11.2.

Beyond being a synthesis algorithm for a simple percussion instrument, such a scheme can be modified to behave as an artificial reverberation algorithm through the insertion of an input signal—see Programming Exercise 11.4.

It is direct to develop schemes for the 3D wave equation in a similar manner (see Problem 11.8), as well as for the case of the membrane coupled to a cavity (see Problem 11.9 and Programming Exercise 11.5).

11.2.1 von Neumann analysis

Use of the ansatz $u_{l,m}^n = z^n e^{jh(l\beta_x + m\beta_y)}$ leads to the characteristic equation

$$z - 2 + 4\lambda^2 \left(p_x + p_y\right) + z^{-1} = 0$$

in the two variables $p_x = \sin^2(\beta_x h/2)$ and $p_y = \sin^2\left(\beta_y h/2\right)$, as defined in Section 10.2.2. This equation is again of the form (2.13), and the solutions in z will be of unit modulus when

$$0 \leq \lambda^2 \left(p_x + p_y\right) \leq 1$$

The left-hand inequality is clearly satisfied. Given that the variables p_x and p_y take on values between 0 and 1, the right-hand inequality is satisfied for

$$\lambda \leq \frac{1}{\sqrt{2}} \tag{11.13}$$

which is the stability condition for scheme (11.10). Notice that just as in the case of scheme (6.34) for the 1D wave equation, at the stability limit (i.e., when $\lambda = 1/\sqrt{2}$), a simplified scheme results, namely (11.12).

11.2.2 Energy analysis and numerical boundary conditions

In order to arrive at stable numerical boundary conditions for scheme (11.10), it is again of great use to derive an expression for the numerical energy. Considering first the case of the scheme defined over the infinite region $\mathcal{D} = \mathbb{Z}^2$, taking an inner product with $\delta_{t\cdot}u$ and using summation by parts (10.17) leads to

$$\begin{aligned}\langle \delta_{t\cdot}u, \delta_{tt}u\rangle_{\mathbb{Z}^2} = \gamma^2 \langle \delta_{t\cdot}u, \delta_{\Delta\boxplus}u\rangle_{\mathbb{Z}^2} &= \gamma^2 \left(\langle \delta_{t\cdot}u, \delta_{xx}u\rangle_{\mathbb{Z}^2} + \langle \delta_{t\cdot}u, \delta_{yy}u\rangle_{\mathbb{Z}^2}\right) \\ &= -\gamma^2 \left(\langle \delta_{t\cdot}\delta_{x+}u, \delta_{x+}u\rangle_{\mathbb{Z}^2} + \langle \delta_{t\cdot}\delta_{y+}u, \delta_{y+}u\rangle_{\mathbb{Z}^2}\right)\end{aligned}$$

and, finally, to energy conservation

$$\delta_{t+}\mathfrak{h} = 0 \qquad \text{with} \qquad \mathfrak{h} = \mathfrak{t} + \mathfrak{v}$$

and

$$\mathfrak{t} = \frac{1}{2}\|\delta_{t-}u\|_{\mathbb{Z}^2} \qquad \mathfrak{v} = \frac{\gamma^2}{2}\left(\langle \delta_{x+}u, e_{t-}\delta_{x+}u\rangle_{\mathbb{Z}^2} + \langle \delta_{y+}u, e_{t-}\delta_{y+}u\rangle_{\mathbb{Z}^2}\right) \tag{11.14}$$

The basic steps in the stability analysis are the same as in the 1D case: Beginning from the expression above for the numerical potential energy $\mathfrak{v}$, and assuming, for simplicity, $h_x = h_y = h$, one may bound it, using inequalities of the form (10.18), as

$$\mathfrak{v} \geq -\frac{\gamma^2 k^2}{8}\left(\|\delta_{x+}\delta_{t-}u\|_{\mathbb{Z}^2} + \|\delta_{y+}\delta_{t-}u\|_{\mathbb{Z}^2}\right) \geq -\frac{\gamma^2 k^2}{2h^2}\left(\|\delta_{t-}u\|_{\mathbb{Z}^2} + \|\delta_{t-}u\|_{\mathbb{Z}^2}\right)$$

and thus

$$\mathfrak{h} \geq \frac{1}{2}\left(1 - \frac{2\gamma^2 k^2}{h^2}\right) \|\delta_{t-} u\|^2_{\mathbb{Z}^2}$$

so the energy is non-negative under condition (11.13), obtained through von Neumann analysis.

To arrive at stable numerical boundary conditions, one may proceed as in the continuous case. Jumping directly to the case of a discrete quarter plane $\mathcal{D} = \mathbb{Z}^{2,x+y+}$, the energy balance above becomes

$$\delta_{t+}\mathfrak{h} = \mathfrak{b}$$

where

$$\begin{aligned}\mathfrak{b} &= -\gamma^2\left(\sum_{m=0}^{\infty} h\delta_{t\cdot} u_{0,m}\delta_{x-} u_{0,m} + \sum_{l=0}^{\infty} h\delta_{t\cdot} u_{l,0}\delta_{y-} u_{l,0}\right) \\ &= -\gamma^2\left(\{\delta_{t\cdot} u, \delta_{x-} u\}_{(0,\mathbb{Z}^+)} + \{\delta_{t\cdot} u, \delta_{y-} u\}_{(\mathbb{Z}^+,0)}\right)\end{aligned}$$

For more on these boundary terms, see the discussion on page 296. The lossless numerical conditions which then follow are

$$u_{0,m\geq 0} = 0 \quad \text{or} \quad \delta_{x-} u_{0,m\geq 0} = 0 \qquad \text{and} \qquad u_{l\geq 0,0} = 0 \quad \text{or} \quad \delta_{y-} u_{l\geq 0,0} = 0 \tag{11.15}$$

The first in each pair corresponds, naturally, to a Dirichlet condition, and the second to a Neumann condition, as in the continuous case as per (11.7). Given losslessness of the boundary condition (i.e., $\mathfrak{b} = 0$), stability analysis follows as for the case of the scheme defined over $\mathbb{Z}^2$. Implementation details of these conditions, as well as the matrix form of the discrete Laplacian, have been discussed in Section 10.2.4. Such conditions are first-order accurate, so there will be some distorting effect on modal frequencies; through the use of a different inner product, it is possible to arrive at second-order (or centered) boundary conditions which are provably stable—see Problem 11.10 and Programming Exercise 11.6 for some investigation of this.

11.3 Other finite difference schemes

In the case of the 1D wave equation, certain parameterized finite difference schemes have been examined in Section 6.3, but given the good behavior of the simplest explicit scheme, these variations are of virtually no use—the power of such parameterized methods has been seen to some extent in the case of bar and stiff string simulation, in Section 7.1.5 and Section 7.2.3, and is even more useful in the present case of multidimensional systems such as the 2D wave equation (among others). The chief interest is in reducing numerical dispersion. The 2D wave equation, like its 1D counterpart, is a standard numerical test problem, and, as such, finite difference schemes have seen extensive investigation—see, e.g., [366, 49, 40].

11.3.1 A parameterized explicit scheme

Instead of a five-point Laplacian approximation, as employed in scheme (11.10), one might try a nine-point approximation, of the type discussed in Section 10.2. This leads to

$$\delta_{tt} u = \gamma^2 \delta_{\Delta\alpha} u = \gamma^2\left(\alpha\delta_{\Delta\boxplus} u + (1-\alpha)\delta_{\Delta\boxtimes}\right) u \tag{11.16}$$

which reduces to scheme (11.10) when $\alpha = 1$. The computational stencil is, obviously, more dense in this case than for the simple scheme (11.10)—see Figure 11.4(b). Simplified forms are possible involving less computation, as in the case of scheme (11.10), as has been observed in relation to the so-called interpolated waveguide mesh [312, 41]. See Problem 11.11.

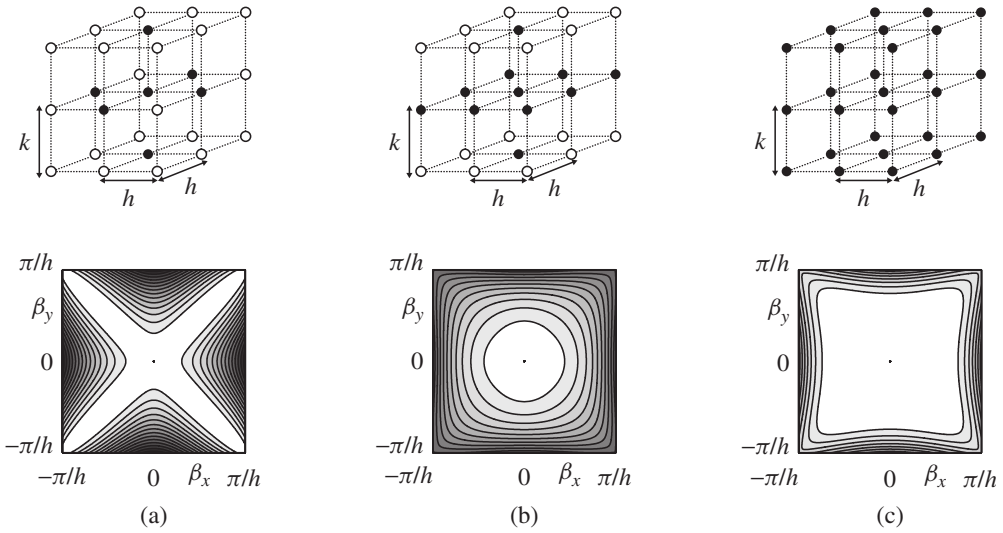


Figure 11.4 Computational stencil (top) and relative numerical phase velocity v_ϕ/γ (bottom) for: (a) the explicit scheme (11.10), with $\lambda = 1/\sqrt{2}$; (b) the parameterized scheme (11.16) with the special choice of $\alpha = 2/\pi$, and λ chosen to satisfy bound (11.17) with equality; and (c) the implicit scheme (11.18), with $\alpha = 2/\pi$, $\theta = 1.2$, and λ chosen to satisfy bound (11.19) with equality. Relative phase velocity deviations of 2% are plotted as contours in the wavenumber plane $-\pi/h \leq \beta_x, \beta_y \leq \pi/h$.

Using von Neumann analysis, it is straightforward to show that the stability conditions for this scheme are

$$\alpha \geq 0 \qquad \lambda \leq \min(1, 1/\sqrt{2\alpha}) \tag{11.17}$$

See Problem 11.12. The dispersion characteristics of this scheme depend quite heavily on the choice of α, and, as one might expect, for intermediate values of this parameter over the allowable range $\alpha \in [0, 1]$, the dispersion is far closer to isotropic—a desirable characteristic, given that the wave equation itself exhibits such behavior. Through trial and error, optimization strategies, or Taylor series analysis (see Problem 10.3) one may find particularly good behavior when α is between 0.6 and 0.8, as illustrated in the plot of relative phase velocity in Figure 11.4(b). The behavior of this scheme is clearly more isotropic than that of (11.10), but numerical dispersion still persists. See also the comparison among modal frequencies generated by the various schemes discussed here in Table 11.1.

Table 11.1 Comparison among modal frequencies of the 2D wave equation defined over a square, under fixed conditions, with $\gamma = 1000$ and modal frequencies (as well as their cent deviations from the exact frequencies) of the simple explicit scheme (11.10), the parameterized explicit scheme (11.16) with $\alpha = 2/\pi$, and the implicit scheme (11.18) with $\alpha = 2/\pi$ and $\theta = 1.2$, with a sample rate $f_s = 16\,000$ Hz, and where λ is chosen so as to satisfy the stability condition in each case as close to equality as possible.

Mode number	Exact freq.	Explicit		Nine-pt. explicit		Implicit	
		Freq.	Cent dev.	Freq.	Cent dev.	Freq.	Cent dev.
(1,1)	707.1	707.0	−0.3	706.3	−2.0	707.1	−0.1
(1,2)	1118.0	1114.0	−6.2	1114.9	−4.8	1118.0	0.0
(2,2)	1414.2	1413.1	−1.3	1407.6	−8.1	1413.4	−1.0
(1,4)	2061.6	2009.2	−44.5	2042.7	−15.9	2058.8	−2.2
(3,3)	2121.3	2117.5	−3.1	2098.7	−18.5	2115.7	−4.6

Such a choice may be justified, again, by looking at computational complexity (see Section 7.1.5 for similar analysis in the case of the ideal bar). For a surface of unit area, and when the stability bound above is satisfied with equality, the number of degrees of freedom will be

$$N_{\text{fd}} = \frac{2}{h^2} = \frac{2f_s^2 \min(1, 1/2\alpha)}{\gamma^2}$$

and equating this with the number of degrees of freedom in a modal representation, one arrives at a choice of $\alpha = 2/\pi = 0.637$.

11.3.2 A compact implicit scheme

A family of compact implicit schemes for the 2D wave equation is given by

$$\delta_{tt} u = \gamma^2 \left(1 + \frac{k^2(1-\theta)}{2}\delta_{tt}\right) \delta_{\Delta\alpha} u \tag{11.18}$$

This scheme depends on the free parameters θ, as well as α through the use of the nine-point discrete Laplacian operator $\delta_{\Delta\alpha}$.

von Neumann analysis, though now somewhat more involved, again allows fairly simple stability conditions on the free parameters and λ:

$$\alpha \geq 0 \qquad \theta \quad \text{unconstrained} \qquad \begin{cases} \lambda \leq \sqrt{\dfrac{\min(1, 1/2\alpha)}{\sqrt{(2\theta - 1)}}}, & \theta > \dfrac{1}{2} \\ \lambda \text{ unconstrained}, & \theta \leq \dfrac{1}{2} \end{cases} \tag{11.19}$$

See Problem 11.13. Under judicious choices of α and θ, an excellent match to the ideal phase velocity may be obtained over nearly the entire range of wavenumbers—see Figure 11.4(c). This good behavior can also be seen in the numerical modal frequencies of this scheme, which are very close to exact values—see Table 11.1. Such is the interest in compact implicit schemes: very good overall behavior, at the additional cost of sparse linear system solutions, without unpleasant side effects such as complex boundary termination. An even more general compact family of schemes is possible—see Problem 11.14.

In this case, as in 1D, a vector–matrix representation is necessary. Supposing that the problem is defined over the finite rectangular domain $\mathcal{D} = \mathbb{U}^2_{N_x, N_y}$, and, referring to the discussion in Section 10.2.4, scheme (11.18) may be written as

$$\mathbf{A}\mathbf{u}^{n+1} + \mathbf{B}\mathbf{u}^n + \mathbf{A}\mathbf{u}^{n-1} = 0 \tag{11.20}$$

where, here, $\mathbf{u}^n$ is a vector consisting of the consecutive vertical strips of the 2D grid function $u^n_{l,m}$ laid end to end (each such strip will consist of $N_y - 1$ values in the case of Dirichlet conditions, and $N_y + 1$ values for Neumann conditions). The matrices $\mathbf{A}$ and $\mathbf{B}$ may be written as

$$\mathbf{A} = \mathbf{I} - \frac{\gamma^2 k^2 (1-\theta)}{2}\mathbf{D}_{\Delta\alpha} \qquad \mathbf{B} = -2\mathbf{I} - \gamma^2 k^2 \theta \mathbf{D}_{\Delta\alpha}$$

where $\mathbf{D}_{\Delta\alpha}$ is a matrix operator corresponding to $\delta_{\Delta\alpha}$. See Programming Exercise 11.7.

11.3.3 Further varieties

Beyond the two families of schemes presented here, there are of course many other varieties—further properties of parameterized schemes for the 2D wave equation are detailed

elsewhere [40, 208], and other schemes are discussed in standard texts [342]. In musical acoustics, finite difference schemes for the wave equation in 2D and 3D have also been employed in detailed studies of wind instrument bores, sometimes involving more complex coordinate systems—see, e.g., [254, 255, 258].

11.4 Digital waveguide meshes

The extension of digital waveguides to multiple dimensions for sound synthesis applications was first undertaken by van Duyne and Smith in the mid 1990s [384–386]. It has continued to see a fair amount of activity in sound synthesis and artificial reverberation applications—see the comments and references in Section 1.2.3. Most interesting was the realization by van Duyne and others of the association with finite difference schemes [384, 41]. Waveguide meshes may be viewed as acoustic analogues to similar scattering structures which appear in electromagnetic simulation, such as the transmission line matrix method (TLM) [182, 83, 174].

A regular Cartesian mesh is shown in Figure 11.5(a). Here, each box labeled **S** represents a four-port parallel scattering junction. A scattering junction at location $x = lh$, $y = mh$ is connected to its four neighbors on the grid by four bidirectional delay lines, or waveguides, each of a single sample delay (of k seconds, where $f_s = 1/k$ is the sample rate). The signals or wave variables impinging on a given scattering junction at grid location l, m at time step n from a waveguide from the north, south, east, and west are written as $u_{l,m}^{n,(+),N}$, $u_{l,m}^{n,(+),S}$, $u_{l,m}^{n,(+),E}$, and $u_{l,m}^{n,(+),W}$, respectively, and those exiting as $u_{l,m}^{n,(-),N}$, $u_{l,m}^{n,(-),S}$, $u_{l,m}^{n,(-),E}$, and $u_{l,m}^{n,(-),W}$. The scattering operation at a given junction may be written as

$$u_{l,m}^{n} = \frac{1}{2}\left(u_{l,m}^{n,(+),N} + u_{l,m}^{n,(+),S} + u_{l,m}^{n,(+),E} + u_{l,m}^{n,(+),W}\right) \tag{11.21}$$

$$u_{l,m}^{n,(-),\bullet} = -u_{l,m}^{n,(+),\bullet} + u_{l,m}^{n} \tag{11.22}$$

Here, $u_{l,m}^{n}$ is the junction variable (often referred to as a junction pressure, and written as p in room acoustics applications). Shifting in digital waveguides themselves can be written as

$$u_{l,m}^{n,(+),N} = u_{l,m+1}^{n-1,(-),S} \qquad u_{l,m}^{n,(+),S} = u_{l,m-1}^{n-1,(-),N} \qquad u_{l,m}^{n,(+),E} = u_{l+1,m}^{n-1,(-),W} \qquad u_{l,m}^{n,(+),W} = u_{l-1,m}^{n-1,(-),E} \tag{11.23}$$

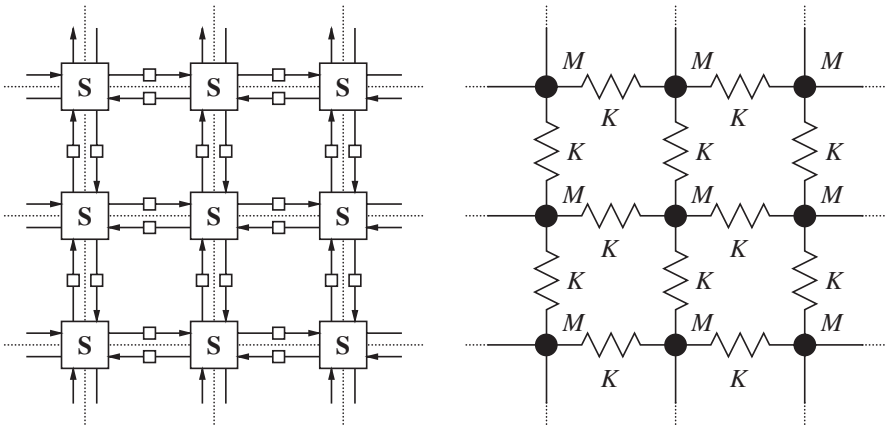


Figure 11.5 (a) A digital waveguide network corresponding to scheme (11.10) for the wave equation, under the special choice of $\lambda = 1/\sqrt{2}$. (b) A lumped network corresponding to the same scheme, for any value of λ.

The scattering and shifting operations are the basis of all wave-based numerical methods, including, in addition to digital waveguides, wave digital filter methods, as well as the transmission line matrix method mentioned above. Numerical stability for an algorithm such as the above is obvious: the shifting operations clearly cannot increase any norm of the state, in terms of wave variables, and the scattering operation corresponds to an orthogonal (i.e., l_2 norm-preserving) matrix multiplication.

On the other hand, by applying the scattering and shifting rules above, one may arrive at a recursion in the junction variables $u^n_{l,m}$, which is none other than scheme (11.10), with $\lambda = \gamma k/h = 1/\sqrt{2}$—see Problem 11.15. Notice, however, that the finite difference scheme requires two units of memory per grid point, whereas in the wave implementation, five are necessary (i.e., in order to hold the waves impinging on a junction at a given time step, as well as the junction variable itself). Furthermore, the mesh requires eight arithmetic operations (four in order to form the junction variable and four in order to perform scattering), whereas the finite difference scheme requires five. Thus the efficiency advantage of the waveguide in 1D, based around the use of delay lines, does not carry over to the multidimensional case. Indeed, the finite difference scheme computes an identical solution, at roughly half the cost. Still, the structural robustness of the waveguide mesh is a very desirable property—termination of a mesh structure using passive filtering blocks (thus ensuring stability) can be a nice alternative to the more involved energy analysis tools described here, though one must beware the temptation to extend such results to more complex systems. This mesh structure may be extended, through the addition of an extra "self-loop" at each junction, in order to correspond directly to scheme (11.10), for any allowable value of λ [41]. Other more commonly seen extensions include different tilings of 2D and 3D space, involving, for example, hexagonal, triangular, or tetrahedral grids [386, 28], which also correspond to well-known finite difference schemes [41]. Such non-Cartesian grids have also, of course, been examined in the mainstream simulation literature [420, 363].

11.5 Lumped mass–spring networks

A regular lumped network of masses and springs also, not surprisingly, may be viewed in terms of the same underlying finite difference scheme; the manipulations are very similar to those used in the case of the 1D wave equation, as discussed in Section 6.1.1. It may also, of course, be used as a starting point for a derivation of the 2D wave equation (see, e.g., [115]); 2D lumped structures simulating membranes appear as components in, for example, the TAO synth environment [267]. Suppose that, as shown in Figure 11.5(b), the network consists of a regularly spaced array of masses M, located at positions $x = lh$, $y = mh$, and constrained to move vertically (i.e., in and out of the plane array), with displacement indicated by $u_{l,m}(t)$. Each is connected to neighboring masses by springs of stiffness K, and the force exerted on the mass l, m by its neighbors to the north and east are indicated by $f_{l,m+1/2}$ and $f_{l+1/2,m}$, respectively.

The equation of motion for a single mass will be

$$M\frac{d^2u_{l,m}}{dt^2} = f_{l,m+1/2} - f_{l,m-1/2} + f_{l+1/2,m} - f_{l-1/2,m}$$

with

$$f_{l,m+1/2} \approxeq K(u_{l,m+1} - u_{l,m}) \qquad f_{l+1/2,m} \approxeq K(u_{l+1,m} - u_{l,m})$$

Just as in 1D, these two definitions coalesce as

$$M\frac{d^2u_{l,m}}{dt^2} = K\left(u_{l,m+1} + u_{l,m-1} + u_{l+1,m} + u_{l-1,m} - 4u_{l,m}\right)$$

Discretizing the above system of ODEs with a second time difference δ_{tt} leads immediately to scheme (11.10) for the 2D wave equation, with $\gamma = h\sqrt{K/M}$.

11.6 Modal synthesis

One of the interesting (and attractive) things about modal synthesis is that, once one has determined the modal frequencies, and the weighting coefficients, the form of the algorithm is completely insensitive to problem dimensionality—all one is doing is summing sine waves or, more generally, damped sine waves. Modal synthesis for 2D objects such as plates has been developed by various researchers, and membrane- and plate-like objects are components within the Modalys/MOSAIC synthesis environment developed at IRCAM [242].

Consider the case of the 2D wave equation, defined over the unit area rectangle of aspect ratio ϵ, $\mathcal{D} = \mathbb{U}^2_\epsilon$. Again, just as in 1D (see Section 6.1.11), once one has derived a set of modal functions $U_m(x, y)$ ordered according to some index m, where the mth modal frequency is ω_m, the modal decomposition is of the form

$$u(x, y, t) = \sum_m U_m(x, y)\Phi_m(t) \qquad \text{with} \qquad \frac{d^2\Phi_m(t)}{dt^2} = -\omega_m^2\Phi_m(t)$$

It is useful to order the modes such that $\omega_m \geq \omega_{m'}$ when $m \geq m'$. (In general, the 2D modal functions cannot be consistently ordered according to two indices except in regular geometries such as the rectangle and circle, and under particular boundary conditions; for the rectangle, Dirichlet or Neumann conditions at all edges are among these, and one may instead use functions $U_{p,q}(x, y)$ indexed according to p and q as described in Section 11.1.3.)

This form is now suitable for discretization along the lines presented in Chapter 3. For each SHO in the set above, one could use a simple scheme such as (3.12), but there is no reason not to use the exact scheme (3.39) at sample rate $f_s = 1/k$, for some time step k:

$$u^n(x_o, y_o) = \sum_{m=1}^{N_m/2} U_m(x_o, y_o)\Phi_m^n \qquad \Phi_m^{n+1} = 2\cos(\omega_m k)\Phi_m^n - \Phi_m^{n-1} \quad \text{for} \quad m = 1, \ldots, N_m/2 \tag{11.24}$$

Here x_o, y_o are the readout coordinates, and N_m, the number of degrees of freedom, should be chosen such that $\omega_{N_m/2} \leq \pi f_s$, to avoid aliasing. For an initial value problem, the variables Φ_m must be initialized using Fourier expansion coefficients of the initializing distribution—see Programming Exercise 11.8 for more details.

It is instructive to compare the output of this modal method to that of a finite difference scheme, such as (11.10)—see Figure 11.6, in which time responses for the two methods are plotted together,

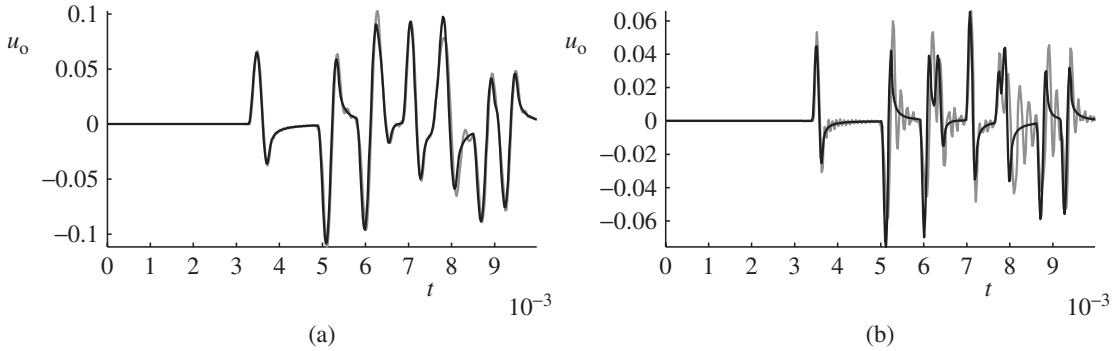


Figure 11.6 Comparison between modal synthesis output (in black), using scheme (11.24), and that of finite difference scheme (11.10) (in grey), for the 2D wave equation, with $\gamma = 200$ and aspect ratio $\epsilon = 1$. The system is subjected to a "plucked" initial condition in the form of a raised 2D cosine, of amplitude 1, centered at $x = 0.3$, $y = 0.3$, of half-width 0.1 in (a), and a narrower distribution of half-width 0.05 in (b). In both cases, output is produced at 44.1 kHz, and read from the position $x_o = 0.8$, $y_o = 0.8$.

in the cases of plucked initial conditions of (a) wide and (b) narrow spatial extent. For the wide excitation, which excites mainly the lower modes, the two responses are nearly identical; but in the case of the narrow excitation, numerical dispersion effects of the finite difference scheme, leading to "ringing" of sharp transients (due to reflections) in the signal, are clearly in evidence. For the modal method, no such dispersion is evident, and transients remain well localized. On the other hand, as this dispersion is only an issue at high frequencies, it is questionable whether it is in fact audible—to this end, the reader may wish to experiment with the two methods.

11.7 Finite difference schemes in radial coordinates

The 2D wave equation defined over a circular region is a good starting point for drum models. Here, due to the geometry, if one is interested in pursuing a finite difference synthesis approach, radial coordinates are convenient. The wave equation in radial coordinates, using the form of the Laplacian given on the right in (10.2), is as follows:

$$u_{tt} = \gamma^2 \left(\frac{1}{r}(r u_r)_r + \frac{1}{r^2} u_{\theta\theta} \right)$$

The equation is assumed defined over the unit circle $\mathcal{D} = \mathbb{U}_\circ^2$, and scaled such that, if the physical wave speed is c, and the radius is R, $\gamma = c/R$. Energy analysis leads to the usual expression for energy conservation, i.e.,

$$\frac{d\mathfrak{H}}{dt} = \mathfrak{B}$$

with

$$\mathfrak{T} = \frac{1}{2}\|u_t\|_{\mathbb{U}_\circ^2}^2 \qquad \mathfrak{V} = \frac{\gamma^2}{2}\left(\|u_r\|_{\mathbb{U}_\circ^2}^2 + \|u_\theta / r\|_{\mathbb{U}_\circ^2}^2 \right) \qquad \mathfrak{B} = \gamma^2 \{u_t, u_r\}_{(1,[0,2\pi))}$$

(See Section 10.1.5 for information on integration by parts in radial coordinates.) Clearly, the conditions $u = 0$ or $u_r = 0$ at the outer rim of the circle lead to losslessness.

11.7.1 An explicit finite difference scheme

An obvious choice of explicit difference scheme (and one the reader is advised against using—read on) is the following:

$$\delta_{tt} u = \gamma^2 \delta_{\Delta\circ} u \tag{11.25}$$

where the discrete Laplacian $\delta_{\Delta\circ}$ is as defined in (10.23), with the exception of at the center point, where the definition (10.24) is used. The update form of this scheme, when $l > 0$, is

$$u_{l,m}^{n+1} = 2u_{l,m}^n - u_{l,m}^{n-1} + \frac{\gamma^2 k^2 \mu_{r+} r_l}{r_l h_r^2} u_{l+1,m}^n - \frac{2\gamma^2 k^2}{h_r^2} u_{l,m}^n + \frac{\gamma^2 k^2 \mu_{r-} r_l}{r_l h_r^2} u_{l-1,m}^n$$
$$+ \frac{\gamma^2 k^2}{r_l^2 h_\theta^2} \left(u_{l,m+1}^n - 2u_{l,m}^n + u_{l,m-1}^n \right)$$

Here, the subscripts l and m refer to grid points, of spacing h_r and h_θ, in the radial and angular coordinate, respectively; the index m is taken modulo N_θ, where $h_\theta = 2\pi/N_\theta$. At the central grid point $l = 0$, the update is

$$u_{0,0}^{n+1} = 2u_{0,0}^n - u_{0,0}^{n-1} + \frac{4\gamma^2 k^2}{N_\theta h_r^2} \sum_{m=0}^{N_\theta - 1} \left(u_{1,m}^n - u_{0,0}^n \right)$$

Energy analysis and numerical boundary conditions

von Neumann analysis is not available in a direct form in order to determine stability conditions in radial coordinates, but energy methods remain viable. One must, however, take special care with regard to the center point. To this end, take the inner product with $\delta_{t\cdot}u$ over the region $\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}$, consisting of the grid locations $l = 1, \ldots, N_r$, $m = 0, \ldots, N_\theta - 1$, and employ the summation by parts identity (10.27):

$$\begin{aligned}\langle \delta_{t\cdot}u, \delta_{tt}u\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} &= \gamma^2 \langle \delta_{t\cdot}u, \delta_{\Delta\circ}u\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} \\ &= -\gamma^2 \left\langle \delta_{t\cdot}\delta_{r-}u, \frac{\mu_{r-}r}{r}\delta_{r-}u\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} - \gamma^2 \left\langle \delta_{t\cdot}\delta_{\theta-}u, \frac{1}{r^2}\delta_{\theta-}u\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} \\ &\quad + \mathfrak{b} - \frac{\gamma^2\pi h_r^2}{4}\delta_{t\cdot}u_{0,0}\delta_{\Delta\circ}u_{0,0}\end{aligned}$$

This may be written, as usual, as $\delta_{t+}\mathfrak{h} = \mathfrak{b}$, where $\mathfrak{h} = \mathfrak{t} + \mathfrak{v}$, and

$$\begin{aligned}\mathfrak{t} &= \frac{1}{2}\|\delta_{t-}u\|^2_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} + \frac{\pi h_r^2}{8}\left(\delta_{t-}u_{0,0}\right)^2 \\ \mathfrak{v} &= \frac{\gamma^2}{2}\left\langle \delta_{r-}u, \frac{\mu_{r-}r}{r}e_{t-}\delta_{r-}u\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} + \frac{\gamma^2}{2}\left\langle \delta_{\theta-}u, \frac{1}{r^2}e_{t-}\delta_{\theta-}u\right\rangle_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} \\ \mathfrak{b} &= \gamma^2\{\delta_{t\cdot}u, \mu_{r+}r\delta_{r+}u\}_{(N_r,[0,N_\theta-1])}\end{aligned}$$

Notice in particular the appearance of a component of the kinetic energy at the central grid location. Clearly, $\mathfrak{t}$ is non-negative, and for stability it suffices to bound $\mathfrak{v}$ in terms of it. To this end, one may write, using the bounds (10.28),

$$\begin{aligned}\mathfrak{v} &\geq -\frac{\gamma^2k^2}{8}\|\sqrt{mu_{r-}r/r}\delta_{r-}\delta_{t-}u\|^2_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} - \frac{\gamma^2k^2}{8}\|(1/r)\delta_{\theta-}\delta_{t-}u\|^2_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} \\ &\geq -\frac{\gamma^2k^2}{2h_r^2}\|\delta_{t-}u\|^2_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}} - \frac{\pi\gamma^2k^2}{4}\left(\delta_{t-}u_{0,0}\right)^2 - \frac{\gamma^2k^2}{2h_\theta^2}\|(1/r)\delta_{t-}u\|^2_{\overline{\mathbb{U}}^2_{\circ,N_r,N_\theta}}\end{aligned}$$

Now that $\mathfrak{v}$ has been bounded in terms of the grid function $\delta_{t-}u$, it is possible to extract the following conditions for non-negativity of $\mathfrak{h}$:

$$\frac{\gamma^2k^2}{h_r^2}\left(1 + \frac{1}{h_\theta^2}\right) \leq 1 \qquad\qquad h_r \geq \sqrt{2}\gamma k \tag{11.26}$$

The first condition arises from examination of the grid function over interior points, and the second for the central point; normally, the first condition above is much stronger than the second. These serve as stability conditions for scheme (11.25), as long as lossless or dissipative boundary conditions are applied; notice that $\mathfrak{b}$ vanishes when

$$u_{N_r,m} = 0 \qquad \text{or} \qquad \delta_{r+}u_{N_r,m} = 0$$

These conditions correspond to Dirichlet and Neumann conditions at the outer edge of the circle.

In general, h_r and h_θ are independent in the above scheme, and there is thus an extra degree of freedom when these quantities are subject to the stability condition (11.26). For some guidance as to how to proceed in choosing these, see Problem 11.16.

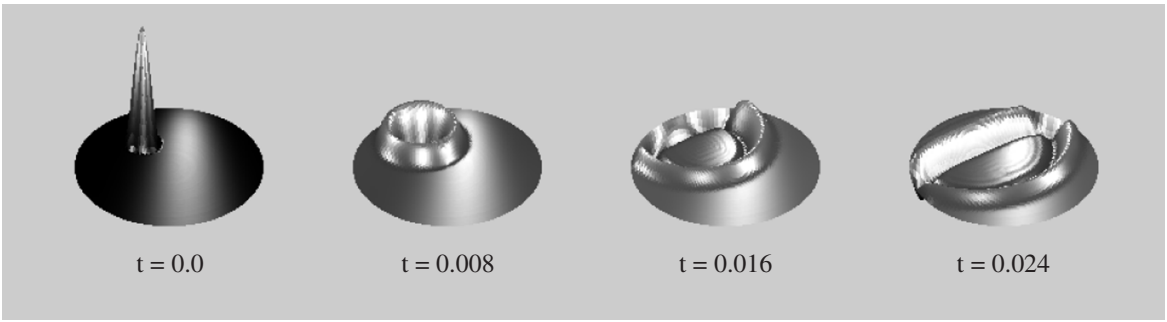


Figure 11.7 Snapshots of the time evolution of the solution to the 2D wave equation, with $\gamma = 40$, defined over a unit circle with fixed boundary conditions, at times as indicated. Scheme (11.25) is employed, at a sample rate of 44 100 Hz, and is initialized with a raised cosine distribution.

Numerical bandwidth

The behavior of this scheme is, at first sight, as expected—see Figure 11.7, which illustrates the time evolution of a plucked initial condition. Upon inspection of a typical spectral output, however, it is clear that all is not well, as in Figure 11.8(a). The output is severely band-limited, in some cases to as low as one-eighth the sample rate. Furthermore, the components which do appear suffer from extreme numerical dispersion. This scheme, though it has been employed by various authors in an analysis setting, is probably not suitable for synthesis, except at extremely high sample rates—clearly, more work at the stage of scheme design is necessary.

11.7.2 A parameterized scheme

Given the difficulties mentioned above, one might wonder whether one can do better with a parameterized implicit scheme—and indeed one can. Here is a simple form:

$$\left(1 + \gamma^2 k^2 \alpha \delta_{\Delta\circ}\right) \delta_{tt} u = \gamma^2 \delta_{\Delta\circ} u \tag{11.27}$$

The new free parameter is α, and the scheme reduces to (11.25) when $\alpha = 0$. In vector–matrix form, the scheme update looks like

$$\mathbf{A}\mathbf{u}^{n+1} + \mathbf{B}\mathbf{u}^{n} + \mathbf{A}\mathbf{u}^{n-1} = \mathbf{0}$$

where

$$\mathbf{A} = \mathbf{I} + \gamma^2 k^2 \alpha \mathbf{D}_{\Delta\circ} \qquad \mathbf{B} = -2\mathbf{I} - (2\alpha + 1)\gamma^2 k^2 \mathbf{D}_{\Delta\circ}$$

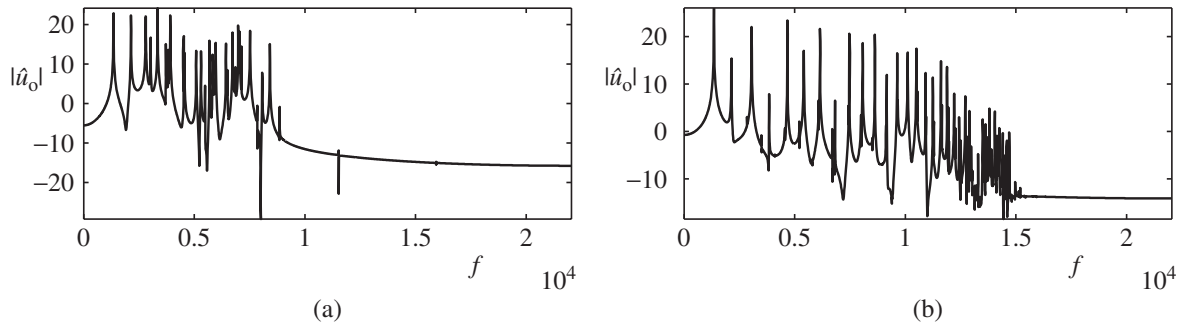


Figure 11.8 Typical output spectra for radial finite difference schemes for the 2D wave equation defined over the unit area circle, with $\gamma = 2000$, under fixed boundary conditions at the outer edge. In (a), scheme (11.25) is employed, and in (b), the parameterized scheme (11.27), with $\alpha = -0.248$. In both cases, the sample rate is 44 100 Hz and $h_\theta / h_r \approxeq 6$.

Here, $\mathbf{u}^n$ is a vector consisting of the value $u^n_{0,0}$ followed by a concatenation of the successive concentric rings of the grid function $u^n_{l,m}$ for $l = 1, 2, \ldots$ and $\mathbf{D}_{\Delta\circ}$ is the Laplacian operator in matrix form.

The stability condition for scheme (11.27) must be adjusted from (11.26) to

$$\frac{4\,(\alpha + 1/4)\,\gamma^2 k^2}{h_r^2}\left(1 + \frac{1}{h_\theta^2}\right) \leq 1 \tag{11.28}$$

when $\alpha \geq -1/4$ (otherwise the scheme is unconditionally stable).

For certain ranges of the parameter α, particularly when α is slightly above $-1/4$, this scheme exhibits markedly better performance, in terms of bandwidth, and numerical dispersion, though even under optimal conditions it performs less well than schemes in Cartesian coordinates—see Figure 11.8(b). For some exploration of this family of difference schemes, see Programming Exercise 11.9.

11.8 Comparative study II

The case of the 2D wave equation is somewhat more representative of the relative merits of the various synthesis methods than its counterpart in 1D. This section continues the previous discussion of computational issues associated with various different synthesis methods with regard to the 1D wave equation in Section 6.6.

The most important observation that one can make is that the efficiency advantage of digital waveguides does not extend to 2D—indeed, any solution produced by a digital waveguide mesh structure can always be more efficiently calculated using an associated finite difference scheme. Similarly, lumped network representations also can be viewed, as in 1D, as finite difference schemes.[2] Thus the main comparison here will be between difference schemes and modal methods, which remain distinct in the 2D case.

With regard to accuracy, modal methods are, under fairly strict conditions on problem geometry and boundary conditions, and if properly employed, extremely accurate. Finite difference methods suffer from a greater degree of numerical dispersion (and thus mode detuning) than in the 1D case, where an exact finite difference (FD) solution is available. In addition, there is the added issue of numerical anisotropy. A more interesting question, and one which is too involved to address here, is the perceptual relevance of such dispersion effects. More elaborate schemes may be designed which minimize such dispersive effects, at the expense of higher computational cost and algorithm complexity.

In terms of memory use, modal methods and difference schemes are roughly comparable. For a unit area domain, and for a given wave speed γ and sampling rate f_s, the number of memory locations necessary to store state in a modal method will be twice the number of modes in the range $[0, f_s/2]$, or, from (11.9), $\pi f_s^2/2\gamma^2$. For a typical difference scheme, the stability condition leads to a minimum grid spacing which again determines the number of memory locations. For scheme (11.10), for example, which is a two-step scheme, the memory requirement will be f_s^2/γ^2, which is on the same order as for a modal method (less, in fact, though this apparent advantage is illusory once numerical dispersion effects are taken into account). As in 1D, however, if the modal shapes are not expressible in closed form, then one is again faced with the extra tasks of (1) computing the mode shapes, through the solution of a potentially large eigenvalue problem and reintroducing numerical inaccuracy, and possibly (2) storing them. For even moderately low

[2] In fact, it is generally only easy to associate mesh structures and lumped networks with simple explicit schemes such as (11.10); making this association with more refined designs, such as implicit schemes, is very strenuous, and thus there is a real advantage to a finite difference framework, from a design perspective.

values of γ, and at audio sample rates, this is, in 2D, a daunting task, as full storage of $\pi f_s^2/2\gamma^2$ modal shapes with sufficient spatial resolution amounts to storage on the order of f_s^4/γ^4 units, which can be very large indeed. On the other hand, in a modal approach, it is relatively easy to discard higher modal frequencies and thus reduce computational cost; in an FD scheme, this can be done by increasing the grid spacing away from that prescribed by the stability bound, at the expense of introducing serious dispersion effects. In either case, however, the best way to reduce computational complexity is to simply work at a reduced sample rate, in which case the two methods are again on an equal footing in terms of computational cost.

The operation count/time step scales, as in 1D, with the number of degrees of freedom for a given γ and f_s. Calling this number N (as mentioned above, it is comparable for FD schemes and modal methods). A modal method requires, in the lossless case, $2N$ additions/multiplications/time step, whereas for an FD scheme, the number of operations depends on the sparsity of the scheme, as αN, for some (low) scheme-dependent parameter α. For the scheme (11.10), $\alpha = 7$, but it will be larger for interpolated explicit schemes such as (11.16), and even larger if an implicit method such as (11.18) is used.

The issues of precomputation, stability, and generating output may be addressed in a manner similar to those of the 1D case.

For more general comments on the relative merits of these methods, see the closing remarks in Section 14.1.

11.9 Problems

Problem 11.1 *Consider the 2D wave equation, defined over the half plane $\mathbb{R}^{2,x+}$, and recall the expression for the energy balance given in* (11.5) *and* (11.6)*:*

$$\frac{d\mathfrak{H}}{dt} = -\gamma^2 \{u_x, u_t\}_{(0,\mathbb{R})}$$

Suppose the boundary condition, at $x = 0$, is given as follows:

$$u_t = \alpha u_x + \phi u_{xt} + \epsilon m \qquad \text{with} \qquad m_t = u_x(0, y, t)$$

and where the constants α, ϕ, and ϵ are non-negative. Such an energy-storing condition may be used in order to model effects of mass, stiffness, and loss at a wall in the context of room acoustics [208]. Show that the above condition implies that the energy balance becomes

$$\frac{d}{dt}(\mathfrak{H} + \mathfrak{H}_b) = -\mathfrak{Q}$$

where $\mathfrak{H}_b \geq 0$ and $\mathfrak{Q} \geq 0$. Thus the total energy $\mathfrak{H} + \mathfrak{H}_b$ remains non-negative and monotonically decreasing.

Problem 11.2 *Extend the energy analysis of the 2D wave equation to the case for which the domain $\mathcal{D}$ is a unit area rectangle $\mathbb{U}_\epsilon^2 = [0, \sqrt{\epsilon}] \times [0, 1/\sqrt{\epsilon}]$. Show that the energy balance* (11.5) *holds as before, where the boundary term $\mathfrak{B}$ is given by*

$$\mathfrak{B} = \gamma^2 \left(\{u_x, u_t\}_{(\sqrt{\epsilon},[0,1/\sqrt{\epsilon}])} - \{u_x, u_t\}_{(0,[0,1/\sqrt{\epsilon}])} + \{u_y, u_t\}_{([0,\sqrt{\epsilon}],1/\sqrt{\epsilon})} - \{u_y, u_t\}_{([0,\sqrt{\epsilon}],0)} \right)$$

Problem 11.3 *Considering the 2D wave equation defined over a square of unit area, under fixed boundary conditions, the frequency of mode (p, q) is given by* (11.8) *or, in Hertz,*

$$f_{p,q} = \frac{\gamma}{2}\sqrt{p^2 + q^2}$$

for $p, q > 0$.

(a) Show that $f_{p,q}$ corresponds to the distance between the origin and the point of coordinates $(p\gamma/2, q\gamma/2)$ in a plane.

(b) Find the density (per unit area) of such points in the plane.

(c) The total number of modes of frequency less than or equal to $f_s/2$ may be represented as the number of modal points contained within a quarter circle of radius $f_s/2$ in the plane. Given the expression obtained above for the density of such points, find the approximate value for $N_m(f_s/2)$ given in (11.9). *(N_m is twice the number of modes over this band.)*

(d) Show that this expression remains unchanged, in the limit of high f_s, if the 2D wave equation is defined instead over a arbitrary rectangle of unit area, i.e., over $\mathbb{U}_\epsilon^2$.

(e) Show that this expression remains unchanged, in the limit of high f_s, in the case of free boundary conditions at all edges of the rectangular domain.

Problem 11.4 (The 3D wave equation I) *Most key components of musical instruments, such as strings, bars, plates, membranes, and acoustic tubes, are well modeled in 1D or 2D. There are, however, occasions when one might be interested in full 3D modeling—this comes up when one is faced with, say, the interior of an instrument body, such as an acoustic guitar or kettledrum, whose dimensions are not small in any coordinate direction. Another application, slightly removed from synthesis, is room acoustics simulation, of interest in developing physical artificial reverberation algorithms. The equation to be solved in these cases is invariably the 3D wave equation, written in terms of time t, coordinates x, y, and z, and wave speed c as*

$$u_{tt} = c^2 \Delta u = c^2 \left(u_{xx} + u_{yy} + u_{zz} \right)$$

or, when spatially scaled with reference to a characteristic length L,

$$u_{tt} = \gamma^2 \Delta u \tag{11.29}$$

with $\gamma = c/L$.

(a) Supposing that the 3D wave equation is defined over a unit cube, under fixed boundary conditions, show that the modal frequencies $\omega_{p,q,r}$ are given by

$$\omega_{p,q,r} = \pi \gamma \sqrt{p^2 + q^2 + r^2}$$

(b) Extend the analysis of Problem 11.3, and show that an expression for the number of degrees of freedom (or twice the number of modes of frequency less than $f_s/2$) is given by

$$N_m(f_s/2) = \frac{\pi f_s^3}{3\gamma^3}$$

(c) Given a typical audio range (i.e., choose $f_s = 44\,100$ Hz), and for a wave speed $c = 340$ m/s, determine the number of modes necessary to describe the behavior of an object such as a violin, a kettledrum, or a large concert hall. (For each, estimate a characteristic length as $L = V^{1/3}$, where V is the total volume enclosed, and thus obtain a rough estimate of $\gamma = c/L$.) Comment on the feasibility of performing simulations using modal methods for each of these cases, keeping in mind the memory available in RAM on typical computers (look this up for your own machine).

Problem 11.5 (Kettledrum I) *Drum modeling always takes as its starting point a membrane model, described by a variant of the 2D wave equation (perhaps involving stiffness—see the next chapter), coupled to a resonator, which is effectively a closed cavity. A detailed model necessarily requires the solution of the 3D wave equation in the body of the resonator, through either a time domain method, or a modal decomposition. (One such study has been carried out, in the case of the kettledrum, by Rhaouti et al. [283].)*

Here is a very crude model, proposed early on by Morse [243]. For simplicity, assume an $L \times L$ square membrane, with wave speed c_M, under fixed boundary conditions, coupled to an enclosed volume of air, of density ρ, volume V_0, with wave speed c_0, and under tension T_0 per unit length applied at the edge. In scaled form, the equation of motion of the membrane is

$$u_{tt} = \gamma^2 \Delta u - \gamma^2 d^2 \int_0^1 \int_0^1 u dx dy \qquad \text{for} \quad (x, y) \in \mathbb{U}^2$$

where $\gamma = c_M/L$, and where $d^2 = \rho c_0^2 L^4/(T_0 V_0)$ is a dimensionless parameter. Such a model, in that the effect of the cavity is averaged over the membrane, is reminiscent of the Kirchhoff–Carrier string model (see Section 8.1), though here the averaging is linear. One should only expect it to hold for low frequencies.

(a) Assuming a modal solution of the form $u(x, y, t) = e^{j\omega t} \sin(p\pi x) \sin(q\pi y)$, show that the modal frequencies $\omega_{p,q}$ are given by

$$\omega_{p,q}^2 = \gamma^2 \pi^2 \left(p^2 + q^2\right) + \frac{\gamma^2 d^2}{pq\pi^2} \left((-1)^p - 1\right) \left((-1)^q - 1\right), \qquad p, q \geq 1 \tag{11.30}$$

Which modal frequencies are altered with respect to those of the uncoupled wave equation (i.e., when $d = 0$)? Are they raised or lowered?

(b) Show that an expression for the total energy of the membrane coupled to the cavity is given by

$$\mathfrak{H} = \frac{1}{2} \|u_t\|_{\mathbb{U}^2}^2 + \frac{\gamma^2}{2} \left(\|u_x\|_{\mathbb{U}^2}^2 + \|u_y\|_{\mathbb{U}^2}^2\right) + \frac{\gamma^2 d^2}{2} \left(\langle u, 1 \rangle_{\mathbb{U}^2}\right)^2 \tag{11.31}$$

which remains non-negative.

Problem 11.6 *Show that scheme* (11.10)*, operating over the infinite domain $\mathcal{D} = \mathbb{Z}^2$, and at the stability limit $\lambda = 1/\sqrt{2}$, may be decomposed into two separate schemes, one operating over values of the grid function $u_{l,m}^n$ for even values of $n + l + m$, and one for odd values. If the scheme is restricted to operate over a finite domain, discuss the effects of various boundary conditions on the ability to obtain such a decomposition.*

Problem 11.7 *Consider scheme* (11.10)*, but with h_x and h_y not, in general, equal.*

(a) Show, from von Neumann analysis, that the stability condition relating k, h_x, and h_y becomes

$$\gamma^2 k^2 \leq \frac{h_x^2 h_y^2}{h_x^2 + h_y^2} \tag{11.32}$$

(b) Write the update explicitly in terms of the grid function $u_{l,m}^n$, in a manner similar to that shown in (11.11)*. How many additions/multiplications are necessary in order to perform the update at a given grid point? Is there a particular choice of k, h_x, and h_y such that computational complexity may be reduced?*

Problem 11.8 (The 3D wave equation II) *The direct extension of scheme* (11.10) *for the 2D wave equation to the 3D wave equation (11.29) is of the following form:*

$$\delta_{tt} u = \gamma^2 \left(\delta_{xx} u + \delta_{yy} u + \delta_{zz} u\right)$$

Here, $u = u_{l,m,p}^n$ is a 3D grid function representing an approximation to $u(x, y, z, t)$ at $x = lh$, $y = mh$, $z = ph$, $t = nk$. for a time step t and a grid spacing h (assumed uniform in all directions).

The spatial difference operators δ_{xx} and δ_{yy} are as in the 2D case, and δ_{zz} is defined, in terms of its action on the grid function u, as

$$\delta_{zz} u^n_{l,m,p} = \frac{1}{h^2}\left(u^n_{l,m,p+1} - 2u^n_{l,m,p} + u^n_{l,m,p-1}\right) \tag{11.33}$$

(a) Using an extension of either von Neumann analysis or energy methods, show that the stability condition for the scheme will be

$$\frac{\gamma k}{h} \leq \frac{1}{\sqrt{3}}$$

(b) Assuming that the 3D wave equation is defined over a unit volume, and that the stability condition above is satisfied with equality, how does the number of degrees of freedom (twice the number of grid points necessary to fill the domain) depend on γ and the sample rate $f_s = 1/k$? Compare your answer to your calculation of the number of modes in Problem 11.4.

Problem 11.9 (Kettledrum II) *Consider the following explicit finite difference scheme for the coupled membrane/cavity system given in Problem 11.5:*

$$\delta_{tt} u = \gamma^2 \delta_{\Delta\boxplus} u - \gamma^2 d^2 \langle u, 1\rangle_{\mathbb{U}^2_{N,N}} \qquad \text{over} \qquad \mathbb{U}^2_{N,N}$$

where "1" refers to a grid function taking the value unity over the entire domain. Assume fixed boundary conditions, i.e., the grid function u takes on the value zero at the edges of the domain $\mathbb{U}^2_{N,N}$.

(a) Find an expression for the conserved energy of this scheme, by taking an inner product of the scheme with $\delta_{t\cdot}u$ over $\mathbb{U}^2_{N,N}$. It should correspond to (11.31) *for the model system.*

(b) By ensuring non-negativity of this expression, show that the CFL condition becomes

$$h \geq \gamma k \sqrt{\frac{2}{1 - d^2\gamma^2 k^2/4}} \tag{11.34}$$

For what values of d will this scheme allow such a stability condition? How does the number of degrees of freedom (i.e., the grid size for a given γ and time step k) change with d? Does this coincide with your analysis of the deviation in modal frequencies with d, in Problem 11.5?

Problem 11.10 *Consider scheme* (11.10) *for the 2D wave equation, defined over the quarter space $\mathcal{D} = \mathbb{Z}^{2,x+y+}$. Assume $h_x = h_y = h$. Instead of the first-order Neumann boundary conditions given as the second of each pair in* (11.15)*, consider the following centered (second-order) conditions:*

$$\delta_{x\cdot} u_{0,m\geq 0} = 0 \qquad \delta_{y\cdot} u_{l\geq 0,0} = 0$$

Such conditions have been introduced earlier in Problem 10.5. Show numerical stability, using energy methods. Recall the use of the primed inner product in the analogous case in 1D, as described on page 138. Here, consider an inner product of the form

$$\langle f, g\rangle'_{\mathbb{Z}^{2,x+y+}} = \sum_{l=1}^{\infty}\sum_{m=1}^{\infty} h^2 f_{l,m} g_{l,m} + \frac{h^2}{2}\sum_{l=1}^{\infty} f_{l,0} g_{l,0} + \frac{h^2}{2}\sum_{m=1}^{\infty} f_{0,m} g_{0,m} + \frac{h^2}{4} f_{0,0} g_{0,0}$$

Problem 11.11 *For the parameterized explicit scheme* (11.16) *for the 2D wave equation:*

(a) Write the update explicitly in terms of the grid function $u^n_{l,m}$, in a manner similar to that shown in (11.11).

(b) For a given value of α, the scheme free parameter, what is the condition on λ such that the scheme update no longer depends on the value of the grid function $u^n_{l,m}$ at the "center point?" Does this condition conflict with the stability condition (11.17) *for the scheme? (This case has received some attention in the literature, as it corresponds to the so-called "interpolated digital waveguide mesh" [312].)*

(c) Discuss the significance and ramifications of a choice of λ away from the stability condition, with reference to Rule of Thumb #1, given on page 136.

Problem 11.12 *Show that the characteristic polynomial corresponding to the "nine-point" explicit scheme* (11.16) *is given by*

$$z - 2 + 4\lambda^2 \left(p_x + p_y - 2(1-\alpha)p_x p_y\right) + z^{-1}$$

where p_x and p_y are as defined in terms of wavenumber components as in (10.13)*, and take on values between 0 and 1. The stability condition is thus*

$$0 \leq \lambda^2 \left(p_x + p_y - 2(1-\alpha)p_x p_y\right) \leq 1$$

Show that these conditions lead immediately to the stability conditions given in (11.17).

Hint: Begin with the left inequality, and derive a condition on α alone, then use the right inequality to find the condition on λ. (Your result from Problem 10.2 will be of great use here.)

Problem 11.13 *Show that the characteristic polynomial corresponding to the compact implicit scheme* (11.18) *is given by*

$$z - 2 + \frac{4\lambda^2 \left(p_x + p_y - 2(1-\alpha)p_x p_y\right)}{1 + 2(1-\theta)\lambda^2 \left(p_x + p_y - 2(1-\alpha)p_x p_y\right)} + z^{-1} = 0$$

where p_x and p_y are again as defined in (10.13)*, and take on values between 0 and 1. The stability condition is thus*

$$0 \leq \frac{\lambda^2 \left(p_x + p_y - 2(1-\alpha)p_x p_y\right)}{1 + 2(1-\theta)\lambda^2 \left(p_x + p_y - 2(1-\alpha)p_x p_y\right)} \leq 1$$

Show that these conditions lead immediately to the stability conditions given in (11.19).

This is much trickier than the above problem, in that now the function to be bounded is rational rather than polynomial. There are essentially three steps: (1) Find conditions under which the numerator of the expression above is non-negative. (2) Given this condition, find conditions under which the denominator is non-negative. (3) Given the conditions derived in (1) and (2) here, find conditions under which the numerator is less than or equal to the denominator. Your bounds on λ may overlap, depending on the choice of θ and α.

Problem 11.14 *Consider the following family of implicit finite difference schemes for the 2D wave equation, generalizing scheme* (11.18)*:*

$$\left(1 + \theta\gamma^2 k^2 \delta_{\Delta\alpha_1}\right) \delta_{tt} u = \gamma^2 \delta_{\Delta\alpha_2} u$$

This scheme now depends on the parameters θ, α_1, and α_2—the two nine-point approximations to the Laplacian are distinct. Using von Neumann analysis, find a stability condition for this scheme of the form $\lambda \leq \lambda^*(\theta, \alpha_1, \alpha_2)$.

Problem 11.15 *Beginning from the definition of the grid variable $u^{n+1}_{l,m}$ in terms of wave variables, from* (11.21)*, use the scattering operation* (11.22) *and the shifting operation* (11.23) *in order to*

show that the waveguide mesh calculates solutions to the finite difference scheme (11.10)*, under the special choice of* $\lambda = \gamma k/h = 1/\sqrt{2}$.

Problem 11.16 *The stability condition for scheme* (11.25) *for the wave equation in radial coordinates, relating the radial and angular grid spacings* h_r *and* h_θ *to the time step* k*, is given in* (11.26)*. In order to set the grid spacings for a given sample rate, it is useful to set, beforehand, a parameter* $q = h_\theta/h_r$*. Show that for a given* q*, a bound on* h_r *is then*

$$h_r^2 \geq \frac{\gamma^2 k^2}{2}\left(1 + \sqrt{1 + \frac{4}{q^2\gamma^2 k^2}}\right)$$

Given that, in an implementation, the grid spacings must be quantized as $h_r = 1/N_r$ *and* $h_\theta = 2\pi/N_\theta$*, for integer* N_r *and* N_θ*, can you find a stability condition in terms of* N_θ *and* N_r*?*

11.10 Programming exercises

Programming Exercise 11.1 *Modify the code implementation of scheme* (11.10) *for the 2D wave equation given in Section A.11, such that the grid spacings* h_x *and* h_y *are not the same. Again, the problem is assumed defined over the unit area rectangle* $\mathbb{U}_\epsilon^2$*, and with Dirichlet conditions on all four sides. The main question here is the following: given a sample rate* $f_s = 1/k$*, how are* h_x *and* h_y *to be chosen such that the bound* (11.32) *is satisfied as close to equality as possible, given the further restriction that* $h_x = \sqrt{\epsilon}/N_x$*,* $h_y = 1/\sqrt{\epsilon}N_y$*, for a given* ϵ*, and for integer* N_x *and* N_y*? Once you have found a solution to this problem, and made the necessary changes to the rest of the code, try comparing the output of your new scheme to that of the scheme with equal grid spacing, for a variety of different choices of the aspect ratio* ϵ *and the wave speed* γ*. Under what conditions are these perceptually distinct?*

Programming Exercise 11.2 *Modify the code implementation of scheme* (11.10) *for the 2D wave equation given in Section A.11, such that the output interpolation location is time varying (i.e., moving). One simple way of specifying a trajectory is as follows: for a rectangular domain of aspect ratio* ϵ*, and thus side lengths* $\sqrt{\epsilon}$ *and* $1/\sqrt{\epsilon}$*, let the continuous output coordinates* x_o *and* y_o *be defined as*

$$x_o(t) = \frac{\sqrt{\epsilon}}{2} + A\cos(2\pi f_o t) \qquad y_o(t) = \frac{1}{2\sqrt{\epsilon}} + A\sin(2\pi f_o t)$$

Here, f_o *is a scan frequency, typically on the order of 1 Hz or less, and* A *specifies the radius of a circle along which the output location travels (how large can* A *be?). (This means of obtaining output is similar to so-called scanned synthesis [401]—see also Programming Exercise 6.6, which deals with the 1D case.)*

Employ bilinear interpolation at each sampling instant (see Section 10.2.1 and Programming Exercise 10.1). For which range of values of γ *and* f_s *do you hear truncation effects? Extend your code such that the output is stereo, with each channel drawn from a separate output location, each specified by a given scan frequency and circle radius.*

Programming Exercise 11.3 *Modify the code implementation in Section A.11 such that Neumann conditions are enforced—try both the first-order accurate conditions given in* (10.20) *and the second-order accurate conditions given in Problem 10.5. Because the solution is now free to drift, you will need to apply some sort of DC blocking filter to your output signal* u_o^n*, if it is drawn directly from values computed on the grid. A simple one–zero filter output* v_o^n *is given by*

$v_o^n = (u_o^n - u_o^{n-1})/k$, and is equivalent to reading a velocity. Can you hear the difference in the resulting sound between the two types of boundary condition?

Programming Exercise 11.4 (Artificial reverberation in 2D) *Now that you have a code implementation of the 2D wave equation under Neumann boundary conditions, it is possible to use it as a crude physical model of a 2D "room," in order to apply artificial reverberation to an input audio signal. The finite difference scheme will be of the form*

$$\delta_{tt} u = \gamma^2 \delta_{\Delta\boxplus} u + J(x_i, y_i) F$$

where $J(x_i, y_i)$ is an input spreading operator acting at the desired source position (x_i, y_i), as described in Section 10.2.1. Be careful to take these coordinates relative to the side lengths of the domain. In order to do this, first remove the initial condition from the code. Then, read in a mono soundfile (read about Matlab functions for doing this, such as, for example, `wavread`*), which will become $F = F^n$ in the above recursion.*

The parameter $\gamma = c/L$ may be used in order to scale the "size" of the virtual reverberant room—do not be surprised if, for particularly low values of γ, the calculation takes quite a while, or if you run into an "out of memory" error. This is a big calculation! In fact, it will be essential for you to use a relatively low audio sample rate (such as 8000 Hz) in order to get the effect of a reasonably sized room in a short calculation. Make sure that your input soundfile is thus properly downsampled before beginning! Also, make sure that the input signal has a zero DC offset before processing—otherwise, your "room" will drift away (think of it as a membrane under free conditions).

You might also wish to generalize the code such that it generates stereo output, by reading from separate locations over the grid.

Programming Exercise 11.5 *Modify the code implementation of scheme* (11.10) *for the 2D wave equation given in Section A.11, such that it simulates a membrane coupled to a cavity, as described in Problems 11.5 and 11.9. (This should be a very minor operation!) Make sure that for a given k, γ, and d, h is chosen according to the more general stability condition* (11.34). *Verify, by taking the Fourier transform of the response of the membrane to an impulsive excitation, that the modal frequencies deviate from those of the uncoupled case according to* (11.30).

Programming Exercise 11.6 *For the scheme* (11.10) *with $h_x = h_y$, defined over the square $\mathbb{U}^2_{N,N}$, calculate numerical modal frequencies using the first-order accurate Neumann conditions given as the second of each pair in* (11.15), *applied at all four edges of the domain, and the second-order accurate conditions given in Problem 11.10. In either case, the scheme, in vector–matrix form, will look like*

$$\mathbf{u}^{n+1} = 2\mathbf{u}^n - \mathbf{u}^{n-1} + \gamma^2 k^2 \mathbf{D}_{\Delta\boxplus} \mathbf{u}^n$$

where $\mathbf{u}^n$ is the grid function $u^n_{l,m}$ rewritten as a vector (see Section 10.2.4), and where $\mathbf{D}_{\Delta\boxplus}$ is the matrix form of the Laplacian operator, incorporating the particular boundary conditions. The modal frequencies, in Hertz, will be given by $f = (1/\pi k)\sin^{-1}(\gamma k\sqrt{-\mathrm{eig}(\mathbf{D}_{\Delta\boxplus})}/2)$. Assume the sample rate to be $f_s = 44\,100$ Hz and $\gamma = 1000$. Produce a sorted list of these frequencies for both types of boundary condition, and compare the first few (say, 20) to the exact frequencies, which are given by

$$f_{p,q} = \frac{\gamma}{2}\sqrt{p^2 + q^2} \qquad \text{for} \qquad p, q = 0, \ldots$$

Programming Exercise 11.7 *Implement the two-parameter scheme* (11.18) *for the 2D wave equation, defined over a rectangular region $\mathbb{U}^2_\epsilon$, according to the vector–matrix update form* (11.20). *Assume Dirichlet boundary conditions. In order to do this, you will need a matrix form*

of the operator $\mathbf{D}_{\Delta\alpha}$*—see Problem 10.4 and Programming Exercise 10.4. In order to perform the update, make use of the standard linear system updating package in Matlab. Compare the computing time required to that of the explicit scheme, which is a special case of scheme* (11.18) *with* $\alpha = \theta = 1$.

In addition, compute the numerical modal frequencies of scheme (11.18) *directly as*

$$f = \frac{1}{2\pi k} \cos^{-1}\left(-\text{eig}(\mathbf{A}^{-1}\mathbf{B})\right)$$

Programming Exercise 11.8 *Create a Matlab implementation of modal synthesis for the 2D wave equation, defined over the unit square* $\mathbb{U}^2$*, under fixed or Dirichlet boundary conditions, and for a plucked initial condition of the form of a 2D raised cosine. Here, the update is as in* (11.24)*, with frequencies* ω *as given in* (11.8)*. In fact, the update is very straightforward; the determination of the initializing values for the state* Φ_m *is more difficult (read about the 2D FFT function* `fft2` *in Matlab).*

Programming Exercise 11.9 *Create a Matlab script which calculates a finite difference solution to the 2D wave equation, defined over the unit circle, using the family of schemes* (11.27)*. Make use of a fixed (Dirichlet) condition at the outer edge of the circle. As this is an implicit scheme, a vector–matrix form, requiring linear system solutions, is essential—you will need to arrange the grid function as a vector* $\mathbf{u}$*, and generate the matrix form of the operator* $\delta_{\Delta\circ}$ *(see Programming Exercise 10.6). You will also need to ensure that you have satisfied the stability condition* (11.28) *as close to equality as possible. In order to do this, rewrite this condition in terms of* h_r *and* $q = h_\theta / h_r$*, as in Problem 11.16.*

For a given value of γ*, such as* $\gamma = 2000$*, perform a study of output bandwidth as a function of the free parameter* α*, as well as the ratio* $q = h_\theta / h_r$*. Initialize the scheme with a sharply peaked pluck-like distribution, so that you will excite all the modes sufficiently, and plot an output spectrum, from which the bandwidth should be easily observable—ignore any isolated spurious modes which appear high in the audio spectrum.*