were huge oscillations. But with ADAPT=.FALSE.,CRANKN=.FALSE., and NSTEPS=1000, which means 1000 steps are taken with a simple backward Euler method, we got the reasonable results shown in Figure 1.2. The backward Euler method is actually ideally suited for differential-algebraic systems (except of course that it is only $O(dt)$ accurate!) and should always be selected when solving mixed time-dependent and steady-state PDEs with PDE2D.[1]

Figure 1.2a shows a pendulum whose oscillations are dying out, but there was no damping included in model (1.2); the total energy in this model should be constant. The damping is due to the low-order backward Euler method's well-known tendency to dissipate energy, and rerunning with NSTEPS=100 000 produced much less damping, as seen in Figure 1.2b.

1.3 Problems

1 (Effect of gravity on spring) Suppose we add gravity as an external force to Eq. (1.1), that is, suppose $my'' = -ky - by' - mg + f(t)$. That will displace the mass downward at rest a distance mg/k (force divided by the spring constant.) Now define $z \equiv y + mg/k$, so the new rest position $y = -mg/k$ corresponds to $z = 0$. Show that Eq. (1.1) becomes $mz'' = -kz - bz' + f(t)$, so this equation really does take gravity into account, as long as $z = 0$ ($y = 0$ in the original equation) means the rest height with gravity on.

2 (Resonance in spring) Solve Eq. (1.1) with $m = 1, b = 0, k = 16, f(t) = sin(\omega t), y(0) = 0, y'(0) = 0$. Solve first with $\omega = 3$, then with $\omega = 4$, and plot the solution $y(t)$ as a function of time (see Figure 1.3). Use the PDE2D GUI ("pde2d_gui *name*") to create your program and "runpde2d *name*" to run the program.

Equation (1.1), with external force $f(t) = sin(\omega t)$, can be solved analytically. If $b^2 < 4mk$, the general solution is

$$y(t) = C_1 e^{-\alpha t} sin(\beta t) + C_2 e^{-\alpha t} cos(\beta t) + \frac{sin(\omega t - \phi)}{\sqrt{(k - m\omega^2)^2 + (b\omega)^2}}$$

where $\alpha = b/(2m)$, $\beta = \sqrt{4mk - b^2}/(2m)$, $\phi = tan^{-1}[b\omega/(k - m\omega^2)]$. From this we see that if the frictional coefficient b is small, and ω is close to the resonant frequency $\sqrt{k/m}$ ($= 4$ in this problem), the solution will have an oscillating term of frequency ω, with a large amplitude. If, as in your problem, $b = 0$ and $\omega = \sqrt{k/m}$, the denominator in the analytical solution given above is zero, so it is no longer a valid solution. What is the general solution then?

1 For a system of equations $c_i u_i' = f_i(t, u_1, ..., u_N)$, backward Euler is $c_i[u_i^{n+1} - u_i^n]/dt = f_i(t^{n+1}, u_1^{n+1}, ..., u_N^{n+1})$. If the i-th equation is algebraic, $c_i = 0$, and this becomes simply $0 = f_i(t^{n+1}, u_1^{n+1}, ..., u_N^{n+1})$, which means the algebraic equation is enforced exactly every step.

This problem has an analytical solution, $u_{true}(x) = -0.1(x^4/24 - Lx^3/6 + L^2x^2/4)$, and with $NXGRID = 10$ gridpoints, using the collocation method, PDE2D calculates the L_1 norm of the error, that is, the integral of $|u - u_{true}|$, as about 10^{-6}. When the Galerkin method is used, with only $NXGRID = 3$ fourth-degree elements, the integral is close to roundoff error, 10^{-11}, because the exact solution is a fourth-degree polynomial.

Example 2.2 (Plate with point loading) Next we solved the plate bending problem (2.2) in the unit square, with $D(x,y) = 1$ and $q(x,y) = -\delta(x - 0.5, y - 0.5)$, where δ is the Dirac delta function, so there is a load only at the midpoint of the square plate. We have to use the Galerkin method because of the point loading. Again, we have to reduce this fourth-order equation to a system of two second-order equations:

$$\nabla^2 u = M$$
$$\nabla^2 M = -\delta(x - 0.5, y - 0.5)$$

We assume clamped boundary conditions ($u = \frac{\partial u}{\partial n} = 0$) at $x = 0$ and $y = 0$ and simply supported boundary conditions ($u = M = 0$) at $x = 1$ and $y = 1$.

Using a property of the delta function, and two integrations by parts (formula (A.3)), and the fact that the boundary integrals disappear because of the boundary conditions, we see that

$$u(0.5, 0.5) = \iint_\Omega u\,\delta(x - 0.5, y - 0.5)\,dA = -\iint_\Omega u\nabla^2 M\,dA =$$
$$-\int_{\partial\Omega} u\frac{\partial M}{\partial n}ds + \int_{\partial\Omega} M\frac{\partial u}{\partial n}ds - \iint_\Omega M\nabla^2 u\,dA = \iint_\Omega -M^2\,dA$$

Thus to check the PDE2D solution, we compare the value of the solution at the midpoint with the PDE2D-calculated integral of $-M^2$, and the agreement is very good (-0.007426). A surface plot of the plate is shown in Figure 2.1. (Note: The minimum value as shown in this plot does not occur exactly at the midpoint.)

2.4 Problems

1 (Beam problem, collocation method) Solve the beam problem (2.1) with $D(x) = 1 + (x-1)^2$, $q(x) = -e^{-(x-1)^2}$, $L = 2$, and clamped boundary conditions, $U = 0$, $U' = 0$ at both ends. Note that parameters and variables must begin with a letter in the range $A - H$ or $O - Z$; otherwise, they will be integers, so, for example, do not use M for the moment, use, say, RM. Output the integral of $U(x)$ over the beam, and plot the beam height (see Figure 2.2).

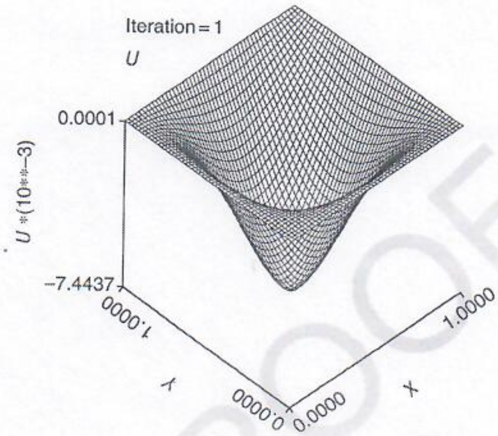

Figure 2.1 Example 2.2 solution.

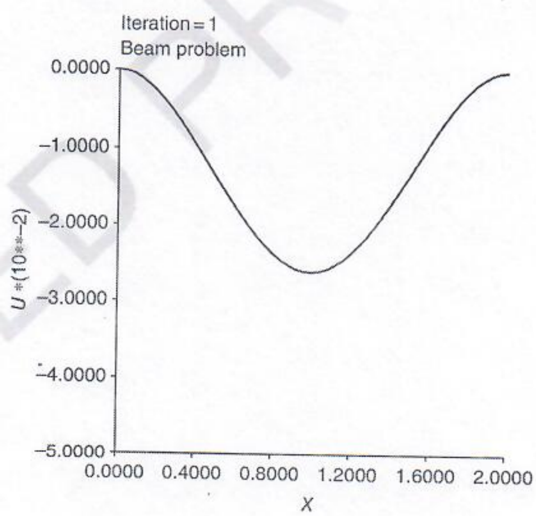

Figure 2.2 Solution for beam Problems 1 and 2.

Use the PDE2D GUI ("pde2d_gui *name*"); that means the collocation method will be used.

2 (Beam problem, Galerkin method) Repeat Problem 1 using the Galerkin method. For this you must use the interactive driver ("pde2d *name*"). Notice that the format for the PDEs and (especially) the boundary conditions are very different than for the collocation method, so pay close attention to the documentation, especially to the hint on how to handle "mixed" (e.g. one

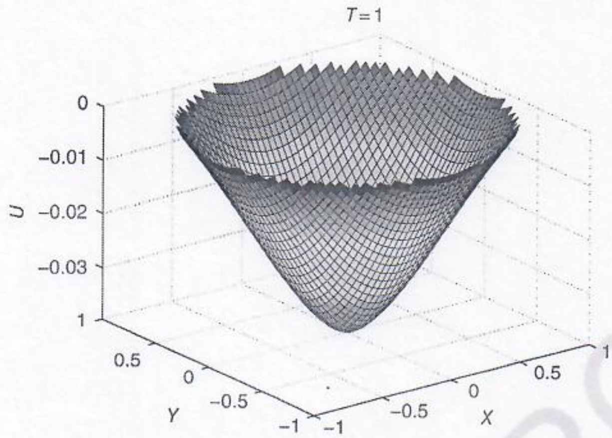

Figure 2.3 MATLAB plot of Problem 3a solution.

fixed, one free) boundary conditions, which must be treated as free conditions. If $A1 = U_x, A2 = RM_x$, you can set $GB1 = 0, GB2 = \text{zero}(U)$, which means $U_x N_x = 0, RM_x N_x = \beta U$, where β is a large number. (Setting $GB2 = \text{zero}(U)$ is exactly the same as $GB2 = \beta U$, except that PDE2D chooses an appropriate large β for you.) Again, output the integral of $U(x)$ to confirm you are getting the same solution as using the collocation method.

3 (Round plate with point loading, Galerkin method)

a) Solve the plate bending problem (2.2) using the PDE2D Galerkin method, with $D(x, y) = 1, q(x, y) = -\delta(x, y)$, in the unit disk; thus, there is a point load at the center of the disk. Use simply supported boundary conditions, $u = M = 0$, where $M \equiv \nabla^2 u$. Use initial triangulation generation option INTRI=2, with X=P*cos(Q), Y=P*sin(Q). Then IPARC(1) = 0 since P=0 is just a point, and IQARC(1)=IQARC(2)=1000, since the two edges Q=0 and Q= 2π coincide. Make a surface or contour plot of u (see Figure 2.3).

   As a check on the solution, let PDE2D calculate the boundary integral of $\frac{\partial M}{\partial n}$. Integrate both sides of (2.2), and use the divergence theorem to evaluate the integral of $\nabla^2 M$ (formula (A.1b)), and you will see what this integral should equal.

b) Find the analytical solution to Problem 3a as follows. Clearly u and M will be functions of $r = \sqrt{x^2 + y^2}$ only, so we can write the Laplacians in (2.2) in polar coordinate form as (cf. Problem 4d of Chapter 5):

$$\frac{1}{r}(ru_r)_r = M$$
$$\frac{1}{r}(rM_r)_r = -\delta(x, y)$$

Since $\delta(x, y) = 0$ for $0 < r \leq 1, (rM_r)_r = 0$, so $M = C\ ln(r) + D$. The boundary condition $M(1) = 0$ implies $D = 0$, so $M = C\ ln(r)$. To find C, use

$$-1 = \iint_R -\delta(x, y)\ dx\ dy = \iint_R \nabla^2 M\ dx\ dy =$$
$$\int_{\partial R} \frac{\partial M}{\partial n} ds = 2\pi r\ M_r = 2\pi\ C.$$

where R is the disk $x^2 + y^2 \leq r^2$, for arbitrary $0 < r \leq 1$, and we have used the fact that $\frac{\partial M}{\partial n} = M_r$ is constant on the boundary ∂R and used the divergence theorem (Appendix A, formula (A.1b)) to replace the area integral with a boundary integral.

Now continue with $(ru_r)_r = rM$, integrate this twice and find the two constants from the boundary conditions $u(1) = 0$ and $u(0) < \infty$. (Hint: $\int r\ ln(r)\ dr = \frac{r^2 ln(r)}{2} - \frac{r^2}{4}$.) Edit your program from Problem 3a to calculate the integral of the absolute value of the error (don't forget to set NINT=1), and rerun.

4 (Round plate with point loading, collocation method) The collocation method does not allow point loading, but you can approximate $\delta(x, y)$ by the bivariate normal function $e^{-(x^2+y^2)/(2\sigma^2)}/(2\pi\sigma^2)$, when the standard deviation σ is small, because then this function has a very large, sharp peak near the origin, and an integral of one. Resolve Problem 3a using the collocation method, with $\delta(x, y)$ replaced by this approximation, and use ITRANS=1, that is, X=p1*cos(p2), Y=p1*sin(p2). At p1=0 there is no boundary condition (enter NONE), and there are periodic boundary conditions at p2= $0, 2\pi$. Now it is essential to use a nonuniform grid, with several gridlines between $p1 = 0$ and, say, $p1 = 5\sigma$.

Again compute the boundary integral of $\frac{\partial M}{\partial n}$ and the integral of the absolute value of the error, using your exact solution from Problem 3b.

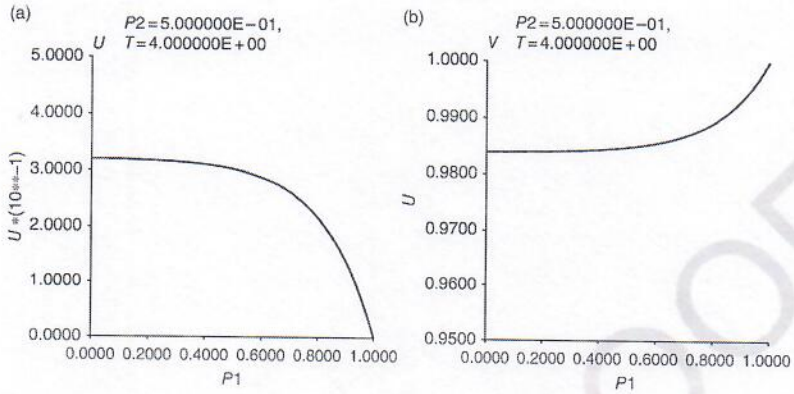

Figure 3.11 Cross sections of E_2 and E at steady state, Problem 1. (a) E_2 density vs. X and (b) E density vs. X.

Redo the problem with no-flux boundary conditions on all four sides. Now when a steady state is reached, u and v will be constant throughout the square. Predict the final values of u, v before running your program, using the facts that the rate of molecule formation now matches the rate of molecule breakup and that the total weight of atoms plus molecules of E in the region cannot change since now there is no boundary flux.

2 (2D heat convection) Solve the heat convection equation ((3.3) with no conduction or sources, $\kappa = q = 0, \rho\ C_p = 1$):

$$T_t = -\nabla \cdot (T\ \mathbf{v}) \tag{3.6}$$

or, if the velocity flow is incompressible, $\nabla \cdot \mathbf{v} = 0$

$$T_t = -\mathbf{v} \cdot \nabla T$$

in three-fourths of a unit circle, $0 \le r \le 1, \pi/2 \le \theta \le 2\pi$, where the incompressible velocity flow is $\mathbf{v} = (y, -x)$. The region and fluid velocity are shown in Figure 3.12. The initial condition is $T(x, y, 0) = 80$, and at the inlet boundary, $\theta = 2\pi$, you should specify $T = 100$ with no boundary conditions on the rest of the boundary. If you use the collocation method, just input NONE for these boundary conditions; if you use Galerkin, A and B should be 0, so $GB = 0$ means $0 = 0$, or no boundary condition. CRANKN=.TRUE. is not a good idea for this problem, as it will cause the solution to oscillate.

Since there is no heat conduction, the "heat front" at $\theta = 2\pi$ will simply move with the flow, and at $t = \pi/2$ one-third of the region will be "hot" ($T = 100$), while the rest will be cold ($T = 80$). If you resolve with the

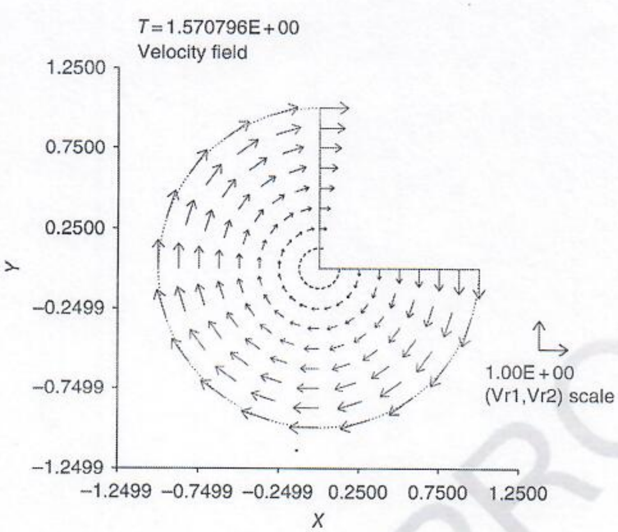

Figure 3.12 Fluid velocity field, Problem 2.

boundary condition $T = 100$ at the outlet boundary ($\theta = \pi/2$) and no boundary condition at the inlet, the solution will, not surprisingly, go unstable. Problem 1a of chapter 3 of Sewell (2015) shows that (3.6) has a unique solution if the temperature is specified on the inlet boundary only, that means, if you want to predict future temperatures, you need to know the temperatures upwind, not downwind, from where you are!

Resolve (3.3) with $\kappa = 0.01$, and you will see that heat conduction smooths out the solution (Figure 3.13b); the heat front will no longer be as sharp. Figure 3.13 shows why a small amount of "artificial diffusion" is sometimes added to a convection-only problem, to make it more tractable.

3 (Shock wave) Use either the collocation or Galerkin algorithm to solve the 1D Burgers' equation:

$$
\begin{aligned}
Y_t &= Y\,Y_x \\
Y(x,0) &= x \\
Y(-1,t) &= -1 \\
Y(1,t) &= 1
\end{aligned}
$$

This nonlinear problem can be looked at as a diffusion/convection problem (3.2) with no diffusion and convection velocity $v = -Y$ dependent on the solution. As we observed in Problem 2, a first-order convection problem

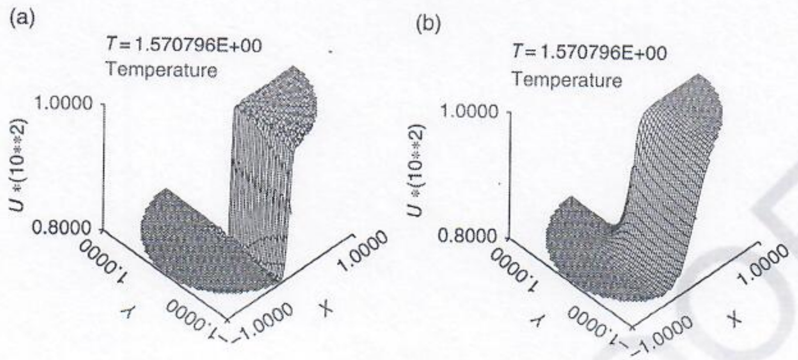

Figure 3.13 Temperature at $t = \pi/2$, Problem 2. (a) $\kappa = 0$ and (b) $\kappa = 0.01$.

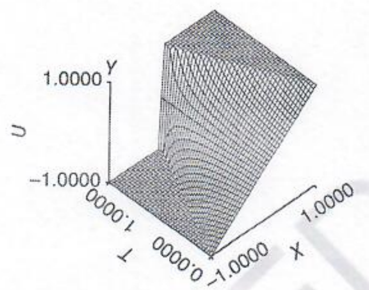

Figure 3.14 Shock wave formation, Problem 3.

needs boundary conditions on the inlet and not the outlet. At the left endpoint, $x = -1$, the velocity is held at $v = -Y = 1$, so the flow is inward; at the right endpoint $x = 1$, the velocity is $v = -Y = -1$, so the flow is inward there also, so it seems appropriate to specify boundary conditions at both ends. If you solve this problem, everything is fine until $t = 1$, and then the solution goes unstable; if you request an adaptive time step, PDE2D will take extremely small steps.

From the solution displayed in Figure 3.14, you can see what the problem is. If you think of Y as temperature, there is a flow of cold air ($Y = -1$) coming in from the left with velocity 1 and a flow of hot air ($Y = 1$) coming from the right with the same velocity, and they collide at $x = 0$ and form a discontinuous shock wave. Sometimes people encounter such behavior in nonlinear PDEs and think they just need a better numerical algorithm, but no PDE solver can get past $t = 1$ successfully on this problem, as it is the solution itself that goes unstable.

Retry the problem after adding 0.01 Y_{xx} to the right-hand side of the PDE, and you will be able to integrate past $t = 1$ to a steady state. Adding

2 (Stresses at corner) Suppose the L-shaped block shown in Figure 5.6 (there are edges at $x = 0, 1, 2$ and $y = 0, 1, 2$) is attached ($U = V = 0$) at the base ($y = 0$), and a unit boundary force to the left ($GB1 = -1, GB2 = 0$) is applied along the top ($y = 2$), and there are zero boundary forces on the other edges. Solve (use Galerkin) the 2D elasticity problem (5.6) (without the time derivatives) with plane strain relations (5.7), with $E = 10^6$, $\nu = 0.1$. Plot the displacement (U, V) field and the stress field. Since stresses will be high near the corner $(1, 1)$, you should define TRIDEN to be large near this point, so the final triangulation will be dense there, as in Figure 5.6a.

3 (Axisymmetric elasticity problem) The axisymmetric equations, derived in Problem 4, are another way to reduce the 3D elasticity equations (5.2)/(5.4) to two dimensions. This means the 3D region and the forces and boundary conditions are such that if we convert the equations to cylindrical coordinates r, θ, z, the solution will not depend on the angle θ, only on r and z:

$$\begin{aligned}\rho U_{tt} &= (\sigma_{rr})_r + (\sigma_{rz})_z + \frac{E(U_r - U/r)}{(1+\nu)r} + f_r \\ \rho W_{tt} &= (\sigma_{zr})_r + (\sigma_{zz})_z + \frac{E(U_z + W_r)}{2(1+\nu)r} + f_z\end{aligned} \tag{5.10}$$

with

$$\begin{aligned}\sigma_{rr} &= E\frac{(1-\nu)U_r + \nu(W_z + U/r)}{(1+\nu)(1-2\nu)} \\ \sigma_{zz} &= E\frac{\nu(U_r + U/r) + (1-\nu)W_z}{(1+\nu)(1-2\nu)} \\ \sigma_{rz} = \sigma_{zr} &= E\frac{U_z + W_r}{2(1+\nu)}\end{aligned} \tag{5.11}$$

Here U and W are the displacements in the r and z directions, and f_r, f_z are the external forces in the r and z directions. On free boundaries, the boundary conditions have the form

$$\begin{aligned}\sigma_{rr}N_r + \sigma_{rz}N_z &= g_r \\ \sigma_{zr}N_r + \sigma_{zz}N_z &= g_z\end{aligned} \tag{5.12}$$

where (g_r, g_z) is the boundary force vector.

Solve the steady-state axisymmetric equations in the notched cylinder of height 2 and radius 2, shown in Figure 5.7 (the vertices of the notch are $(2, 0.9), (1.5, 1.0), (2, 1.1)$). Rotate this figure with the left edge as the axis to imagine the original 3D figure. The cylinder is glued to the floor at the bottom ($U = W = 0$), and there is an upward force at the top: $(g_r, g_z) = (0, 1)$. At $r = 0$ we have symmetry boundary conditions, $U = W_r = 0$ (see Problem 1e of Chapter 6), and thus also $U_z = 0$ and $\sigma_{rz} = 0$. See Problem 2 of Chapter 2 for help on how to handle mixed boundary conditions like these using

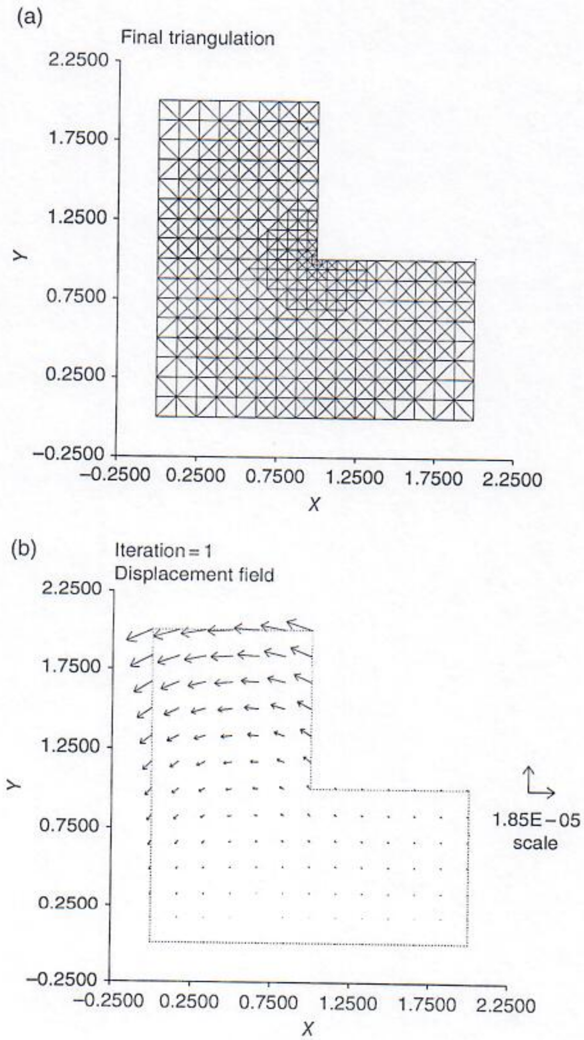

Figure 5.6 L-shaped block, Problem 2. (a) Graded final triangulation and (b) displacement field.

we call this a "symmetry" boundary. Argue that a symmetry boundary condition is the same as a free-slip condition. (Hint: If p is the normal component of velocity/displacement, and q is a tangential component, what would happen if either p or $\frac{\partial q}{\partial n}$ were nonzero?)

2 (Time-dependent problem using stream function) Solve, using the Galerkin method, the time-dependent stream function equations (6.6) in the region of Examples 6.1–6.3, starting with initial conditions $U = V = \phi = \omega = 0$, but with free-slip conditions ($\phi = \omega = 0$) on the bottom and sides this time but still no-slip conditions (see Problems 1a and b) on the V-shaped top. Use $\rho = 10, \mu = 0.1, \mathbf{f} = (y, -x)$. Plot the velocity vector $(\phi_y, -\phi_x)$ out to about $t = 40$, by which time a steady state (Figure 6.8) will have been reached. (Hint: To plot $(\phi_y, -\phi_x)$ you can set APRINT(1)=PHIy, BPRINT(1)=-PHIx, assuming ϕ is named "PHI," then plot $(A1, B1)$.)

3 (Fluid flow with free-slip boundary)

a) Use the collocation method to solve the steady-state 2D fluid flow equations (6.5), with penalty method formulation, and $\mu = 0.1, \rho = 10, \mathbf{f} = (-y, x)$, in the trapezoid of Figure 6.9a. This means set $P_x = -\alpha(U_{xx} + V_{yx})$ and $P_y = -\alpha(U_{xy} + V_{yy})$. This trapezoid can be parameterized by $x = p1, y = p2 * (1 + p1), 0 \le p1 \le 1, 0 \le p2 \le 1$, and boundary conditions are no slip on the left and right sides and free slip on the top (see Problem 1d) and bottom. Take $\alpha = 10^5\mu$; higher values make the problem ill conditioned. Calculate the integral of $U^2 + V^2$, which should be about 0.0131. Pressure plots may be noisy even when the velocity field is accurate.

b) Resolve the problem of part (a) using the Galerkin method and the steady-state equations (6.7), with penalty method formulation. Again calculate the integral of $U^2 + V^2$, to compare with part (a), and plot the pressure $P = -\alpha(U_x + V_y)$ (Figure 6.9b). The free-slip boundary conditions on the top can be written

$$GB1 = \text{zero}(U * NORMx + V * NORMy)$$
$$GB2 = \text{zero}(Uy - Vx)$$

The penalty parameter α can be much larger for the Galerkin method before the problem becomes too ill conditioned.

c) Resolve the problem of part (a) using the collocation method and the steady-state stream function equations (6.6). Recall that in Problem 1a it was shown that free-slip conditions for the stream function approach can be modeled by $\phi = \omega = 0$ and in Problem 1b it was shown that no-slip conditions are $\frac{\partial \phi}{\partial n} = \phi = 0$.

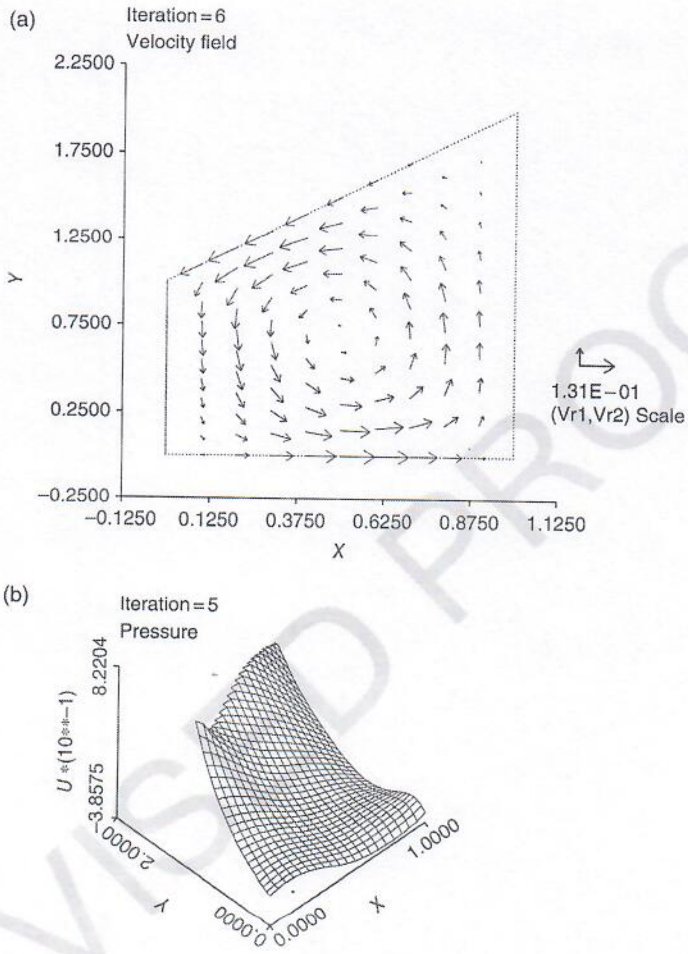

Figure 6.9 Penalty method formulation, Problems 3a and b. (a) Velocity field, using collocation method and (b) pressure, using Galerkin method.

Plot the streamlines (Figure 6.10a) and velocity field $(\phi_y, -\phi_x)$, and calculate the integral of $U^2 + V^2 = \phi_y^2 + \phi_x^2$ to compare with parts (a) and (b).

4 (Fluid flow with inlet and outlet) Solve the 2D steady-state equations (6.7) with penalty method (Galerkin method) in the region of Examples 6.1–6.3,

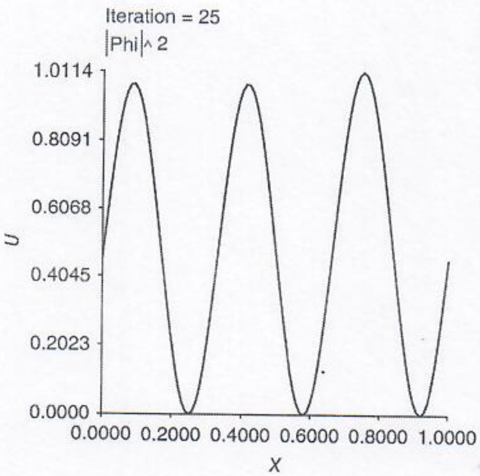

Figure 7.9 $|\phi|^2$ for third eigenfunction, with $b = 0.5$, Problem 3b.

3 (Kronig–Penney model)

a) When $a = 1, b = 0.01, P = 1.5$, and $k = \pi$, it appears from Figure 7.3 that the third eigenvalue of the Kronig–Penney problem (7.5) is about $\lambda = 90$. Use a numerical method (Newton's method, for example) to solve the algebraic equation (7.6), where it is assumed that b is small, for a value of λ near 90.

b) Use PDE2D to find the eigenvalue of (7.5) closest to $\lambda = 90$ (set EV0R=90) and the corresponding eigenfunction, with $a = 1, b = 0.01, P = 1.5$, and $k = \pi$, and make a plot of probability $|\phi|^2 = UR^2 + UI^2$. Your eigenvalue should be close to the value from part (a), since b is small. Repeat with $b = 0.5$ (Figure 7.9).

4 (Eigenvalues in cone) Find the smallest eigenvalue of

$$W_{xx} + W_{yy} + W_{zz} = \lambda W \quad \text{in the cone } 0 \le z \le 1 - r,$$
$$W = 0 \quad \text{on the boundary.}$$

You will need to set ITRANS=−3 and define an appropriate parameterization of the cone. (Answer: $\lambda = -36.92$.)

Set NPROB=12 and EV0R = -30-5*IPROB, and make 12 runs to find the eigenvalues closest to $-35, -40, \ldots - 90$. This "fishing expedition" should net 4 eigenvalues in this range. Look at the eigenfunction plots for each to see which are axisymmetric. One MATLAB cross-sectional plot for each of the first two eigenfunctions is shown in Figure 7.10, from which it appears that the first is axisymmetric while the second is not.

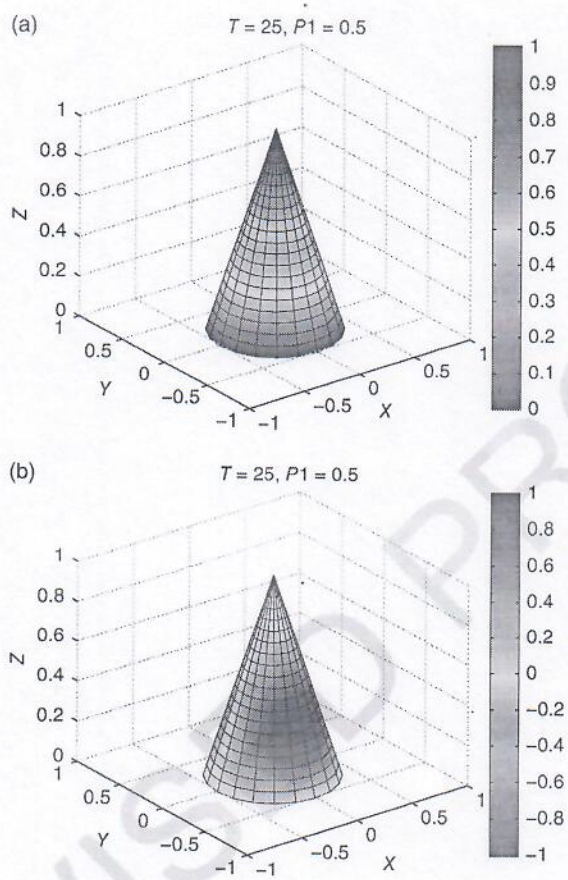

Figure 7.10 Eigenfunction cross sections, Problem 4. (a) $\lambda = -36.92$ and (b) $\lambda = -62.30$.

5 (Axisymmetric cone eigenvalues, Galerkin method) Two of the four eigenvalues found in Problem 4 have eigenfunctions that are functions of r and z only, so these two (but not all eigenvalues of the 3D problem) can also be found by solving the axisymmetric equation:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial W}{\partial r}\right) + \frac{\partial}{\partial z}\left(\frac{\partial W}{\partial z}\right) = \lambda W \qquad 0 \le r \le 1, \qquad 0 \le z \le 1 - r.$$

Multiply this equation through by r, and find the smallest eigenvalue, using the Galerkin method. Note that since $A = rW_r$ and $B = rW_z$