

UCB Math 228B, Spring 2015: Problem Set 5

Due April 16

In this problem set, you will study two extensions of the simple piece-wise linear Poisson solver `fempoi` that we discussed in class. First, you will extend the equations to solve the Helmholtz equation, for simulation of wave propagation in waveguides. Next, you will extend the Poisson solve to use quadratic elements instead of linear.

Time-harmonic Waveguide Simulations

1. Consider the following 2-D Helmholtz problem, for a given wave number k with normalized propagation velocity, and so-called Sommerfeld radiation conditions at the in/out boundaries:

$$-\nabla^2 u - k^2 u = 0, \quad \text{in } \Omega, \quad (1)$$

$$\mathbf{n} \cdot \nabla u = 0, \quad \text{on } \Gamma_{\text{wall}} \quad (2)$$

$$\mathbf{n} \cdot \nabla u + iku = 0, \quad \text{on } \Gamma_{\text{out}} \quad (3)$$

$$\mathbf{n} \cdot \nabla u + iku = 2ik, \quad \text{on } \Gamma_{\text{in}} \quad (4)$$

Here, the domain boundary $\Gamma = \partial\Omega$ is decomposed into the three parts $\Gamma = \Gamma_{\text{wall}} \cup \Gamma_{\text{out}} \cup \Gamma_{\text{in}}$. For a computed solution u , we will also calculate its intensity at the output boundaries:

$$H(u) = \int_{\Gamma_{\text{out}}} |u|^2 ds, \quad (5)$$

where $|\cdot|$ is the complex absolute value.

- a) Derive a Galerkin finite element formulation for (1)-(4), for an appropriate space V_h of continuous piece-wise linear functions.
- b) Show that the discretized system can be written in the form $A\mathbf{u} = \mathbf{b}$, where

$$A = K - k^2 M + ik(B_{\text{in}} + B_{\text{out}}) \quad \text{and} \quad \mathbf{b} = 2ik\mathbf{b}_{\text{in}} \quad (6)$$

for real matrices $K, M, B_{\text{in}}, B_{\text{out}}$ and a vector \mathbf{b}_{in} , which do not depend on the wave number k . Give explicit expressions for the matrix/vector entries, involving the basis functions $\varphi_i(\mathbf{x})$ for the space V_h .

- c) Show that the transmitted intensity (5) for a finite element solution \mathbf{u} can be calculated as $H(\mathbf{u}) = \mathbf{u}^H B_{\text{out}} \mathbf{u}$.
2. For the implementation, consider a model test problem with wave number $k = 6$ and a straight channel domain of dimensions 5×1 :

$$\Omega = \{0 \leq x \leq 5, 0 \leq y \leq 1\} \quad (7)$$

$$\Gamma_{\text{in}} = \{x = 0, 0 \leq y \leq 1\} \quad (8)$$

$$\Gamma_{\text{out}} = \{x = 5, 0 \leq y \leq 1\} \quad (9)$$

$$\Gamma_{\text{wall}} = \{0 \leq x \leq 5, y = 0 \text{ or } y = 1\} \quad (10)$$

- a) Show that an exact solution to the Helmholtz problem (1)-(4) for the domain (7)-(10) is given by $u_{\text{exact}}(x, y) = e^{-ikx}$.

- b) Write a function `waveguide_edges` which for a given triangular mesh `p`, `t` identifies the boundary edges corresponding to the wall, in, and out boundaries, respectively:

Inputs : `p`, `t`

Outputs : `ein`, `eout`, `ewall`

Use the function `boundary_edges` on the course web page to find all mesh edges, then assume that the in-boundary consists of all vertical edges with $x = 0$, and that the out-boundary consist of all vertical edges with $x = 5$.

- c) Write a function `femhelmholtz` that computes the matrices $K, M, B_{\text{in}}, B_{\text{out}}$ and the vector b_{in} for a given mesh:

Inputs : `p`, `t`, `ein`, `eout`

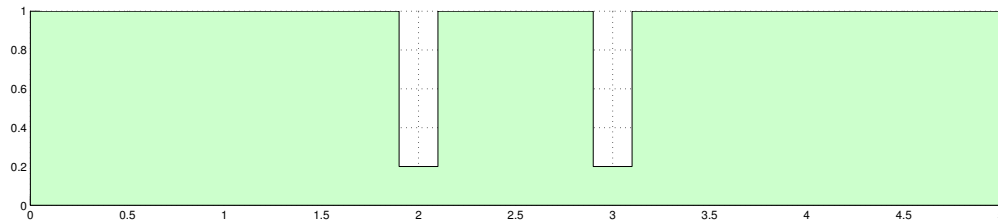
Outputs : `K`, `M`, `Bin`, `Bout`, `bin`

- d) Solve the discretized problem on the meshes generated by the parameters

```
pv = np.array([[0,0], [5,0], [5,1], [0,1], [0,0]])
hmax = 0.3
```

and `nref` ranging from 1 to 4. Compute max-norm errors using the exact solution u_{exact} , plot errors vs. mesh size in a log-log plot, and estimate the order of convergence.

3. Finally, you will use your Helmholtz solver to compute a frequency response for a waveguide with two slits, see figure below. The waveguide is again of dimensions 5-by-1, and the slits are 0.2 units wide and 0.8 units deep, centered at $x = 2$ and $x = 3$.



- a) Create a mesh for the domain using `pmesh`, with `hmax` = 0.2 and `nref` = 2.
- b) To look for resonance phenomena around $k \approx 2\pi$, solve for a range of wave numbers k between $k = 6$ and $k = 6.5$, in steps of $\Delta k = 0.01$. For each k , solve the problem and calculate $H(\mathbf{u})$. Plot H vs. k in a semi-log plot.
- c) Plot two of your solutions using `tplot` of the real part, for the wave numbers k corresponding to the smallest and the largest value of $H(\mathbf{u})$.

Quadratic Elements for Poisson's equation

4. Here you will modify the `fempoi` function to use quadratic elements instead of linear. Some details of the implementation:

- Assume that all triangles are straight-sided
- Use the following second-order accurate integration rule for a triangle T_k with corner nodes $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ and area A_k :

$$\int_{T_k} f(\mathbf{x}) d\mathbf{x} \approx A_k \sum_{i=1}^3 w_i^g f(\mathbf{x}_i^g), \quad w_i^g = 1/3, \quad \mathbf{x}_i^g = \sum_{j=1}^3 (\mathbf{x}_j/6 + \delta_{ij}\mathbf{x}_j/2), \quad i = 1, 2, 3$$

- Note that `tpplot` only knows how to plot linear functions, but to plot a quadratic function you can at least use it to draw linear functions between the corners nodes.

- a) First, a given mesh `p,t` has to be extended for quadratic functions by adding edge midpoints. Write a function `p2mesh` with:

Inputs : `p, t`

Outputs : `p2, t2, e2`

which produces the arrays:

`p2` : $N \times 2$, node coordinates of original mesh nodes and the new midpoints

`t2` : $T \times 6$, local-to-global mapping for the T triangular elements

`e2` : $E \times 1$, indices of boundary nodes, both from original mesh nodes and the midpoints

- b) Write a function `fempoi2` with

Inputs : `p2, t2, e2`

Outputs : `u, A, b`

which solves the Poisson problem using quadratic elements, for the mesh produced by your function `p2mesh` above.

- c) Do a convergence study as follows: Solve the problem on meshes given by

```
pv = np.array([[0,0], [1,0], [1,1], [0,1], [0,0]])
hmax = 0.3
```

with homogeneous Dirichlet conditions on all boundaries and `nref` ranging from 0 to 4. Consider the finest mesh (`nref` = 4) the true solution. Calculate approximate max-norms for all the other solutions (`nref` = 0 to 3) as the maximum error at the set of nodes in the coarsest mesh (`nref` = 0). Plot these errors vs. mesh size in a log-log plot, and estimate the slope.