FEMLAB users manual

Peter Hansbo & Mårten Levenstam

Department of Mathematics, Chalmers University of Technology,
S-412 96 Göteborg, Sweden

Contents

Chapter 1

Problem types solved by FEMLAB

Our goal is to implement adaptive solvers for a wide variety of partial differential equations. To start with, we have implemented a solver for two-dimensional convection-diffusion problems of the type

$$\begin{aligned} -\nabla \cdot (\mathbf{K}\,\nabla u) + \beta \cdot \nabla u + \alpha\, u &= f \quad \text{in}\,\Omega, \\ \boldsymbol{n} \cdot \mathbf{K}\,\nabla u + \gamma\, u &= g \quad \text{on}\,\Gamma_{\mathrm{N}}, \\ u &= h \quad \text{on}\,\Gamma_{\mathrm{D}}, \end{aligned}$$

where $\mathbf{K}$ is a symmetric, positive definite, 2×2-matrix containing material coefficients, α is an absorption coefficient, β is a convective velocity, and γ is a heat transfer coefficient. All of these coefficients may be variable. Furthermore, $\boldsymbol{n}$ is the outward pointing unit normal to the boundary, Γ_{D} is the Dirichlet part of the boundary where the solution is prescribed, and Γ_{N} is the Neumann part of the boundary where the normal flux is prescribed (we also allow for Robin boundary conditions, where the normal flux depends on the solution, also known as *convective* boundary conditions). This problem is solved with h-adaptivity using root-mean-square (L_2), maximum (L_∞), or energy norm control of the error. Thus, the program aims at producing solutions where

$$\|u - u^h\| \leq \text{TOL}\ \|u\|\,,$$

where TOL is a tolerance given by the user, and $\|\cdot\|$ is the chosen norm. In order to avoid too many degrees of freedom, we require the user to also supply the maximum number of refinement levels, where the initial mesh represents level zero.

In order to have accurate control of the error in L_2 and L_∞, an evaluation of a stability constant related to a continuous dual problem is also performed according to the theory exposed in [1].

Chapter 2

Interactive X-windows input and output

In order to simplify the input of data, an X-windows based interactive mesh generator and input data preparation module has been implemented. It is activated by giving the command 'Femlab', which activates an X11-based data I/O window.

2.1 Data input

The X11–window has five command buttons on the top left–hand side:

1. **File**
2. **Geometry**
3. **Mesh**
4. **Equations**
5. **View Result**

The numbered buttons on the right–hand side refers to viewing of post–processed data, and will be discussed in Section 2.2 together with the '**View Result**' button.

Only highlighted command buttons are active; the dimmed buttons will become activated when the right chain of commands has been given.

2.1.1 The 'File' button

Pressing the '**File**' button reveals a roll–down menu with four alternatives:

1. **Load…**
2. **Save…**
3. **Print…**
4. **Print to file…**
5. **Quit…**

The '**Load**'/'**Save**' commands work are tied to each other as follows. At any time during the mesh generation or after solving, the user may launch the '**Save**'–command which prompts for a filename. Wen loading this file using '**Load**', the program will return to exactly the state it was at '**Save**'. This means that the user may leave the input data unfinished and return at a later time to continue where he left off.

The '**Print**' command prints the graphics displayed on the screen to a printer (using the PostScript format). A dialogue window lets the user decide where to direct the output.

The '**Print to file**' command saves the graphics displayed on the screen in a file using postscript. The

The '**Quit**' command exits the program without saving.

2.1.2 The 'Geometry' button

Pressing the '**Geometry**' button reveals a roll–down menu with two alternatives:

1. **Define scale…**
2. **Define boundaries…**

Only the '**Define scale**' commands are highlighted initially. When launching this command, a command box pops up, prompting for the domain size. These figures will set the physical dimensions of the input window (or a close approximation).

Once the scale has been set, another command box pops up, showing the type of geometrical objects that can be chosen to define the boundaries. These are, from top to bottom,

- NACA wing profiles
- Circles
- Ellipses
- Rectangles

- General polygons

The following two items in the command box may be used to move points and to delete given boundaries, respectively; the last two are used for zooming.

Use the left button to click out the boundaries. The general polygon must be closed by clicking at the initial point. Use the left mouse–button to insert points. Make a closed loop of straight segments to complete a boundary; the direction does not matter. To close the loop click once again at the initial point. (NOTE: when closing the loop, take care to position the arrow close to the initial point. If this is not done, one more segment will be created. The mesh generator will assume that the points given by the user to define the boundaries are to be used as nodes, and it cannot handle a situation where the smallest boundary segment is substantially smaller than the local meshsize.)

The other geometrical figures are defined by the point at which you click (centroid for the circle and ellipse, left upper corner for the rectangle and leading edge for the NACA profile. Clicking again on the left mouse button fixes the boundary of the geometrical object.

Note that you must choose type of boundary:

- An outer boundary is exterior to the domain.
- An interior boundary is interior to the domain.
- An internal boundary is embedded in the domain.

The internal boundaries may represent for instance borders between different material types.

WARNING: Do not place a geometrical object too close to a large boundary segment defined by a polygon or a rectangle! The mesh generator is forced to make the elements at least as small as the segment size on the geometrical object. If the given meshsize is considerably larger, problems will occur.

2.1.3 The 'Mesh' button

Pressing the '**Mesh**' button reveals a roll–down menu with two alternatives:

1. **Define mesh size...**
2. **Make mesh**

The '**Define mesh size**' command prompts for the local mesh size. This can be given different values at different points in the domain. Use the left mouse–button to give the location of the mesh size currently displayed in the dialogue window. Interpolation between different points is performed using Shepard's distance–weighting [2].