Numerical Solution of PDEs Using the Finite Element Method

Exercise 5: Local refinement, hanging nodes and the ConstraintMatrix

Jean-Paul Pelteret (jean-paul.pelteret@fau.de)
Luca Heltai (luca.heltai@sissa.it)

20 March 2018

Some useful resources

https://www.dealii.org/8.5.1/doxygen/deal.II/step_4.html
https://www.dealii.org/8.5.1/doxygen/deal.II/step_5.html
https://www.dealii.org/8.5.1/doxygen/deal.II/namespaceGridGenerator.html
https://www.dealii.org/8.5.1/doxygen/deal.II/namespaceVectorTools.html
https://www.dealii.org/8.5.1/doxygen/deal.II/namespaceDoFTools.html
https://www.dealii.org/8.5.1/doxygen/deal.II/group__constraints.html
https://www.dealii.org/8.5.1/doxygen/deal.II/step_49.html

---

1. Using `step-5` (or your previously modified version of `step-3`) as a base:
   (a) Solve Laplace's equation

$$\begin{aligned} -\Delta u(\mathbf{x}) = 0 \quad &\text{in} \quad \Omega, \quad \text{with} \\ u(\mathbf{x}) = 0 \quad &\text{on} \quad \partial\Omega_1, \quad \text{and} \\ u(\mathbf{x}) = 1 \quad &\text{on} \quad \partial\Omega_2 \end{aligned}$$

   on an L-shaped domain with length and width of dimension 1. Here, Ω_2 denotes one of the end-edges of the L, and $\partial\Omega_1 = \partial\Omega \backslash \partial\Omega_2$.
   (b) Starting from a coarse grid, perform successive global refinements to increase the accuracy of the solution and to locate the singularity.
   (c) Now switch from using global refinement to using local refinement in the vicinity of the singularity. To accomplish this, you'll need to build the hanging node constraints.
   (d) Build the Dirichlet boundary directly into a global `ConstraintMatrix`, along with the existing hanging node constraints. With this change, several parts of your code need to be modified:
      i. The constraints need to be built in the `setup` function.
      Tip: You can use the `VectorTools::interpolate` function here.
      ii. The `ConstraintMatrix` should be used to distribute local cell and vector contributions to their global counterparts.
      iii. The constraints must be distributed to the solution.

2. Additional tasks
   (a) Following (c), what happens if you "forget" to distribute the (hanging node) constraints after solving the linear system?
   (b) Instead of using the `VectorTools::interpolate` function, construct the Dirichlet contributions to the `ConstraintMatrix` manually.
   Tip: There are two ways to accomplish this: (1) Use the tools provided in the `DoFTools` namespace, or (2) loop over cell faces and interrogate them for global DoFs that have support there.
   (c) Experiment with some of the other grids in the `GridGenerator` namespace, such as `GridGenerator::hyper_cross` and `GridGenerator::cheese`, or create your own by using some of the grid modification tools discussed in `step-49`. In each case, find ways to efficiently increase the cell density in the location of singularities.

(d) Using your existing code, replicate the study performed in `step-5`. Note that the governing equation has changed slightly. Can you further improve the accuracy of the result in `step-5` using Manifolds?
Tip: Look to the discussion "A better mesh" in the "Possibilities for extensions" section of `step-6`.