Faculty of Electrical Eng, Mathematics and Computer Science

Lab Assignment 1
AP3001-FE
Finite-element analysis for Applied Physics
Kristof Cools

1 Problem Statement

A megatsunami is created when a large land mass slides into the ocean. To understand its potential to cause harm we wish to model this phenomenon. In this lab assignment you will:

- From the wave equation that models propagation of wave through the ocean, you will derive a weak formulation and Galerkin discretisation
- You will encounter various boundary conditions: encounters of waves with coastal lines will be modelled by a simple Dirichlet boundary condition.
- The domain of simulation (essentially the total surface of planet earth) needs to closed of. You can do this in various ways, that results in various levels of realism and accuracy. You can opt to enforce an Robin type absorbing boundary condition, a toroidal or spherical periodic boundary condition, or you can choose to model the earth's surface as an actual sphere, removing the need to bound to domain of simulation.
- You will use an open source meshing tool (gmsh) to create a discrete geometry that approximates the actual map of the world.
- Your will report on your work and findings in a final report.

Let Ω denote the part of the earths surface covered by oceans. The pressure u associated to wave is a solution to the wave equation:

$$-\operatorname{div}\operatorname{grad} u - k^2 u = f \quad \text{in } \Omega \tag{1}$$

Here k is the wave number, which is related to the wavelength by $k = 2\pi/\lambda$. Assume λ to be 4,000 km. The right hand side is related to the source of the wave and can be chosen to be a quickly decaying bell function centered around the epicentre.
The domain Ω is bounded on one hand by coastal lines Γ_1 and on the other hand by the end of the simulation domain (think of it as the edge of the world) Γ_2, such that $\partial\Omega = \bar{\Gamma}_1 \cap \bar{\Gamma}_2$.
We will model coastal lines Γ_1 by enforcing a homogeneous Dirichlet boundary condition:

$$u = 0 \quad \text{on } \Gamma_1 \tag{2}$$

In a first approximation, we will define an arbitrary line to act as the end of the world (a rectangular of circular line enclosing all continents) where we wish to enforce an absorbing boundary condition. Such a condition will cause any waves that impinge on this boundary to just keep going, without creating spurious reflections that would give away the presence of this artificial line.
Consider a plane wave solution in a one-dimensional setting:

$$u(x) = e^{-ikx} \tag{3}$$

In addition to being a solution to the wave equation, this function also solves $\partial u/\partial x + iku = 0$. Also note that the left travelling wave e^{+ikx} does not fulfill this condition. This suggests the following absorbing boundary condition to be applied:

$$\frac{\partial u}{\partial n} + iku = 0, \quad \text{on } \Gamma_2 \tag{4}$$

2 Assigment

In order to solve this problem, one needs to consider the following questions:

1. Give the weak formulation of the problem (partial differential equation + boundary conditions).
2. Give the Galerkin equations (the system of linear equations). In particular, pay attention to the form of the trial function. Which nodes of the mesh need to be associated to basis functions?
3. Give the element matrix and element vector for the internal two-dimensional triangular elements.
4. Give the element matrix and element vector for the one-dimensional boundary elements.
5. Program the finite-element code (elementmatrix, elementvector, boundaryelementmatrix), assemble these local contirbutions in the global finite element matrixm and solve the resulting linear system.
6. Create better meshes that (i) resemble more the outlines of the continents, and (ii) have a smaller mesh size, resulting in a better approximation of the solution on the mesh.

Obviously the behaviour at the boundary is not realistic. In addition we have not taken into account the curvature of the earth. To improve upon this situation, choose one of the following two avenues to increase the model accuracy and realism in a second modelling cycle:

1. Adapt the boundary condition on Γ_1 to be periodic: waves disappearing of the left of the map should reappear on the right:

   $$u(x, c) = u(x, d) \tag{5}$$

   with c, d the y-coordinates corresponding to the left and right boundary. Similarly, implement the boundary condition

   $$u(a, y) = u(b, y) \tag{6}$$

   to allow waves disappearing at the top/bottom of the simulation domain to reappear on the other side. Is this the behaviour you expect on a spherical surface? If not, what geometry are you modelling? Carefully list the natural and/or essential boundary conditions you have introduced.
2. Use gmsh to create a spherical surface and mesh that surface suboordinate to the contours of the oceans. This extension if suited more for those of you interested in CAD. Some research in the required tools and methods is required. Note that the provided library CompScienceMeshes only reads gmsh file format 2. Take this into account when exporting your mesh.

3 Notes

- The provided example solver the Laplace equation with Dirichlet conditions on the complete boundary. You can use this example as a starting point. Take not on how the Dirichlet boundary condition is implemented; Dirichlet boundary vertices are ignored during the assembly procedure. This requires a renumbering of the unknowns and a smart index lookup approach. The approach offered here is certainly not the only/best approach.

- The example is written in Julia. To run, navigate on the command line to the example directory, start julia, and run the example by executing `include(‘‘tsunami.jl’’)`. The first time you do this, this can take some time, because the example dependencies will be downloaded, installed, and compiled. Later executions should run much faster. You can either build on top of this example, or code your solution in your own Julia, Python, or Matlab package.
- Looking at the absorbing boundary condition it is clear that the matrices and the solution for u are complex valued. The field u should be interpreted as a field of phasors. Even though this type of problems was not discussed in the theory, Galerkin's method can still be applied.

4 Report

You are allowed to work in groups of two.
Your final report can be up to 10 pages. It should contain the following sections:

- Introduction and Problem Statement
- Mathematical Model
- Numerical Model
- Discussion of implementation, geometry generation, and results
- Conclusion, recommendations, and future directions

The report together with the code can be uploaded to BrightSpace.