

Homework 4: Partial Differential Equations (PDEs): relaxation and leapfrog methods

C.F. Gauss
Department of Physics, Astronomy, and Applied Physics,
Rensselaer Polytechnic Institute
`gauss@rpi.edu`

04/01/2012

0.1 Assignment

This homework assignment focuses on numerical methods to solve PDE equations.

1. Elliptical PDE: relaxation methods. We have solved a number of cases involving Poisson or Laplace equations during the lectures. Here, I want you to write a program to realize this relaxation, and provide a detailed report of your findings. Your code will include :
 - (a) Simple relaxation method (Jacobi)
 - (b) Gauss-Seidel Method
 - (c) Over- and Under- relaxation method.

Your work will include the following items.

- (a) Calculate the potential in the system depicted on the figure. Study the effect of grid size (test a few different values for separation and thickness).
 - (b) Plot equipotential lines and compute the electric field and discuss the physics.
 - (c) Find the optimal value of the over-relaxation parameter (make a plot of the number of iterations needed to reach a set precision versus relaxation parameter). Make a convincing case when presenting your findings. Here you will use a 128×128 grid.
2. Hyperbolic or Parabolic equations: select a problem of your choice where the leapfrog method can be used. Implement the leapfrog method, discuss the physics of your problem and comment on the numerical results and numerical stability. You are allowed to use a problem discussed in class, provided you consider different boundary/initial conditions.
3. (optional) This part of the assignment is for extra credits (up to 2). As you will soon realize, the speed at which the relaxation method converges depends critically on the quality of starting guess (you could test that for the first problem above, by artificially setting the starting potentials at various values, clearly out of the range of the solution; for instance, putting all the unknown values at -100V should be worse than setting them all between 0 and 10). A good approach to accelerate convergence is by using a multigrid approach (MGA). In the MGA, a coarse grid is first used to perform the relaxation. The converged result is then interpolated into a finer grid, and relaxation is then performed using that new grid. You can use interpolation of your choice to move from one grid to the next (you can simply copy the value at one point of the old grid to the new four points defined by the new grid, or you can use linear or cubic interpolation). Here, I suggest you use a 64×64 grid as a coarse grid and the 128×128 grid as a fine grid.

For those students who want to meet a challenge, I suggest to write a more general code that will use any number of grids (for example, up to 7 grids for the 128×128 finest grid), by successively interpolate into a finer and finer grid. For this second part, the student will choose a Laplace/Poisson problem of their choice. It is important to show and discuss the improvements of the MGA compared to the usual one (i.e. plot the execution time as function of grid size).

Make sure your report is self-contained with sufficient details and clear plots. Feel free to add listings of your codes (or use pseudo-code).

Important Note: One point will be subtracted for each missing or incomplete figure caption or missing axis labels.

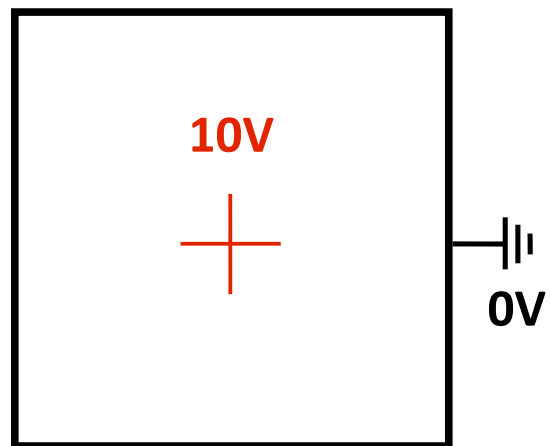


Figure 1: Setup for question 1(a). The central cross is set at 10V and the boundary is grounders (0V).