

Chapter 1

Verification

To ensure the correctness of the solver we have used a few different methods to improve our confidence in it. The first layer is several unittests built to verify the small modular parts of the model where it is possible, for example restricting a predefined grid and checking that the coarser grid is correct, and these tests are described in chapter 2

The multigrid method has several different steps in the algorithm, as a developmental help and to ensure that the program works correctly during as many different conditions as possible we want to test the whole code, as well as the constituent parts where possible. The method is quite modular and several parts of it can be tested alone. The GS-RB, used for smoothing, can be independently tested, since on its own it converges to a solution, just at a higher computational cost than the multigrid method. To test it we will use an initial density field with a length between the grid steps that results in an exact answer. The restriction and prolongation operators can also be tested to a degree by checking that they preserve a constant grid through several grid level changes.

NB! Rewrite (see if something is usable) below

The multigrid method has several different steps in the algorithm, as a developmental help and to ensure that the program works correctly during as many different conditions as possible we want to test the whole code, as well as the constituent parts where possible. The method is quite modular and several parts of it can be tested alone. The GS-RB, used for smoothing, can be independently tested, since on its own it converges to a solution, just at a higher computational cost than the multigrid method. To test it we will use an initial density field with a length between the grid steps that results in an exact answer. The restriction and prolongation operators can also be tested to a degree by checking that they preserve a constant grid through several grid level changes.

1.0.1 The Multigrid method

We use both of the aforementioned tests to check that the whole multigrid method works as intended. Since a constant source term will give a trivial solution of the potential, $\phi(x, y, z) = \mathbf{0}$, we use that as a test. In addition we also test that it converges on a sinusoidal source term as we did the smoother.

1.0.2 Prolongation and Restriction

The prolongation and restriction operators with the earlier proposed stencils will average out the grid points when applied. So the idea here is to set up a system with a constant charge density, $\rho(\mathbf{r}) = C$, and then apply a restriction. After performing the restriction we can check that the grid points values are preserved. Then we can do the same with the prolongation. While this does not completely verify that the operators work as wanted, it gives an indication that we have not lost any grid points and the total mass of the charge density should be conserved.

1.0.3 Smoothers

The iterative method GS-RB used for the pre- and postsmoothing of the grid in our implementation of the multigrid method is also a direct solver. So we can test it, or most other smoothers, by testing them on a small system where the problem has an analytical solution. Then we can let them run for a while and ensure that they are converging towards the solution. If we let the source term be sinusoidal in one direction, and constant in the other directions it has an easy solution given below

$$\nabla^2 \phi(x, y, z) = \sin(x) \tag{1.1}$$

This has a solution when $\phi(x, y, z) = -\sin(x)$ and we can test that the solver converges to the solution. If we let the source term be constant in the x direction and instead vary in the other directions we can get verify that the solver works in all three directions independently.

Chapter 2

Unit Tests