Assignment 2: Partial Differential Equations

G. Flatters
Department of Physics, University of Bristol.
(Dated: October 20, 2020)

This assignment explores two methods for solving partial differential equations; Jacobi and Gauss-Seidel. Evidence has been found to support theoretical models that predict the number of iterations that these two methods require, for a given grid density and convergence condition. The Gauss-Seidel method is then used to find the potential and electric fields of a parallel plate capacitor. These results are compared to the infinite plate solution and its properties. Finally, the diffusion equation is solved for an iron poker using the implicit difference method, and by solving a set of linear equations to evolve the system in time.

LAPLACE'S EQUATION

Relaxation methods are especially important in the solution of linear systems used to model elliptic partial differential equations, such as Laplace's equation (2D):

$$\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} = 0 \tag{1}$$

where *V* is the potential, a smooth real valued function. If we define a two dimensional grid of equally spaced nodes (x_i, y_j) then the equation can be represented as a finite difference function:

$$V^{n+1}(x_i, y_j) = \frac{1}{4}V^n(x_{i-1}, y_j) + V^n(x_{i+1}, y_j) + V^n(x_i, y_{j-1}) + V^n(x_i, y_{j+1}) \tag{2}$$

where V^n is an initial guess potential. Each node on the grid can be approximated by finding the average of its four closest neighbours. The grid is then updated with its new candidate function V^{n+1}, this will hopefully be a better guess of the solution than V^n for a given point. If this new function is not a solution then a 'residue' will be left over (i.e the left hand side of equation 1 will not be equal to zero). Each time a new candidate potential is substituted into a given node the question "is the residue zero?" is asked. This method is repeated - candidate potentials are continuously 'relaxed' until the residue is equal to zero, at which point the solution to the Laplace equation will have been found for that node.

Laplace's equation only has a unique solution if boundary conditions are set. This is done by fixing boundary nodes to known values of potential. The algorithm is carried out for all nodes on the grid that are not boundary nodes. The number of iterations required to find a solution is not *a priori* so a convergence condition must be defined. This is done by stating that if the the change between successive iterations is below a set tolerance level then halt iterations.

There are different methods for carrying out the relaxation algorithm. The Jacobi method computes a new value for every node before updating the full grid. This requires two copies of the grid, one with the old values and one with the new values. This method is not practical because it converges too slowly. A faster method is the Gauss-Seidel method. This makes use of updated values of *V* on the right hand side of equation 2 as soon as they become available. This means that the averaging is done 'in place', rather than being copied from an earlier step like the Jacobi method is. However, this method is also slowly converging when compared to modern methods [1].

THE DIFFUSION EQUATION

Solving the diffusion equation (3) is conceptually similar to solving the Laplace equation using relaxation; an initial distribution for a time dependent system will be relaxed to an equilibrium solution as *t* approaches infinity.

$$\frac{\partial \phi}{\partial t} = \alpha\left(\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2}\right) \tag{3}$$

The forward-time central-space (FTCS) method is based on central difference in space and the forward Euler method in time [2]. This method for two dimensional equations is stable if and only if the following condition is satisfied [3]:

$$\frac{\alpha \delta t}{h^2} \leq \frac{1}{4} \tag{4}$$

where *h* is the grid spacing. The time step Δt is subject to restrictions due to this stability condition. A major limitation of this method is that for problems with relatively large diffusivity (α), the time step restriction can be too severe. A stable alternative is obtained by using the equivalent backward-time equation [4] to go from time $t + \delta t$ to time *t*:

$$\frac{\phi'(x_{i,t+\delta t}) - \phi(x_{i,t})}{\delta t} = \frac{\alpha}{h^2}[\phi'(x_{i-1,t+\delta t}) + \phi'(x_{i+1,t+\delta t}) - 2\phi'(x_{i,t+\delta t})] \tag{5}$$

this method converges for all values of $\frac{\alpha \delta t}{h^2}$ [5], which means there are no restrictions on the time step. The disadvantage is that it requires solving a linear set of equations. These can be arranged as a matrix equation and solved by inverting a matrix of prefactors, *P*, such that:

$$\phi(t+\delta t) = \phi(t)P^{-1} \tag{6}$$

where $\phi(t)$ runs over all nodes in the rod at time *t*.

RATE OF CONVERGENCE

The Jacobi method uses an *iteration matrix* that maps one set of *V*'s into the next. This iteration matrix has eigenvalues, each one representing the extent to which the amplitude of a particular eigenmode of undesired residual is suppressed during one iteration. These factors must all have modulus < 1 for relaxation to work at all. The rate of convergence of the method is limited by the rate of the slowest decaying eigenmode (i.e. the factor with the largest modulus). The modulus of this largest factor is called the *spectral radius* of the relaxation operator (ρ_s). In general, the spectral radius goes asymptotically to 1 as the grid size *J* increases, so that more iterations are required. The estimate for the number of iterations *r* required to reduce the overall error by a factor of 10^{-p} is described by equation 7 [1].

$$r \simeq \frac{2pJ^2\ ln(10)}{(\pi^2)} \simeq \frac{1}{2}pJ^2 \tag{7}$$

Therefore, the number of iterations is proportional to the number of node points J^2. It is found that for the Gauss-Seidel method, the spectral radius is just the square of the spectral radius of the Jacobi method. Equation 8 [1] describes *r* for the Gauss-Seidel method.

$$r \simeq \frac{pJ^2\ ln(10)}{(\pi^2)} \simeq \frac{1}{4}pJ^2 \tag{8}$$

The Gauss-Seidel method has a factor of two improvement in the number of iterations over the Jacobi method. The proportionality with J^2 demonstrates the impracticality of both methods when dealing with values of *J* of the order 10^2 and greater.

RESULTS

Convergence Condition and Grid Density

The Jacobi and Gauss-Seidel routines were tested in two different ways. Firstly, the number of iterations required for varying convergence conditions were measured, as seen in figures 1 and 2. The x-axes are a logarithmic scale, this is done so that the gradients given are in accordance with the theory (equations 7 and 8). The ratio of the Jacobi method gradient over the Gauss-Seidel method gradient is 1.99 ± 0.05. This is exactly as the theory predicts - the Gauss-Seidel method has a factor of two improvement in the number of iterations over the Jacobi method.

The maximum value of *p* that data can be taken for is 15, data becomes unreliable above this because 1^{-16} is the relative quantisation step of a floating point number (float64) [6]. When *p* exceeds this value the iterations required becomes constant for increasing *p* which skews the line of best fit.

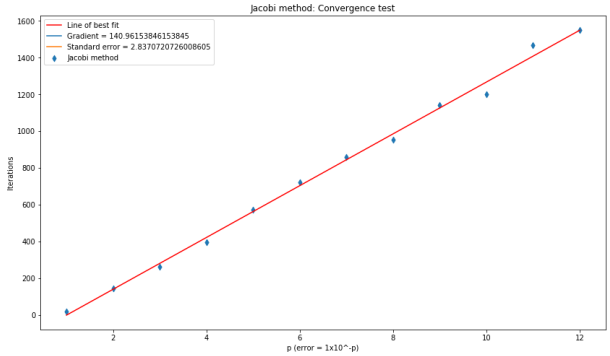


FIG. 1: Jacobi Method: number of iterations vs convergence condition. X-axis is a logarithmic plot. Gradient = 141 ± 3.

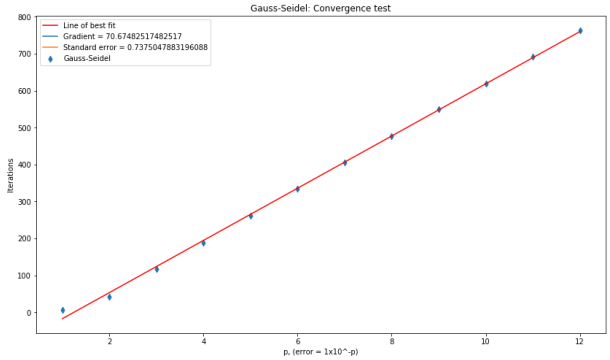


FIG. 2: Gauss-Seidel Method: number of iterations vs convergence condition. X-axis is a logarithmic plot. Gradient = 70.7 ± 0.7.

The second test was of the number of iterations required for varying *J* (grid dimensions are $J \times J$), shown in figures 3 and 4. This data is plotted against the model fits as seen in equations 7 and 8 for the respective methods. Support for the theory is found in the low standard deviations of 0.0133 for the Jacobi method and 0.0137 for the Gauss-Seidel method, suggesting that in both cases the models are a very good fit. Data was taken for values of *J* ten and greater, this is because below this value the grid is too sparse to provide useful data. Obviously, more data could be taken at larger values of *J* at the expense of time.

The results of these tests give me great confidence that my routines are running as intended and are suitable for solving the following physics problem.

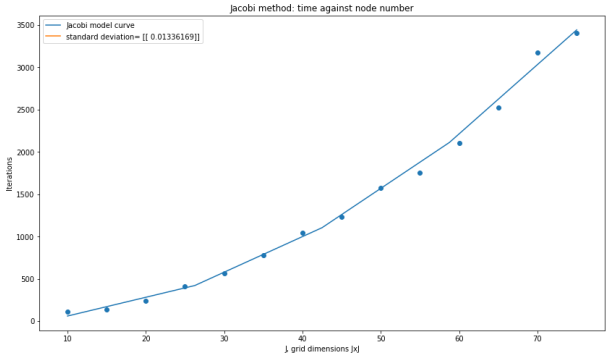


FIG. 3: Jacobi Method: number of iterations vs J with model fitting. Standard deviation = 0.0133.

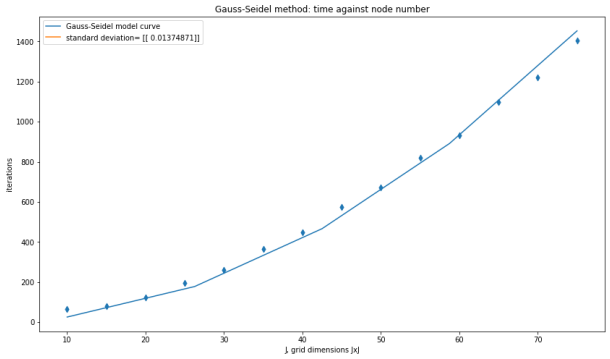
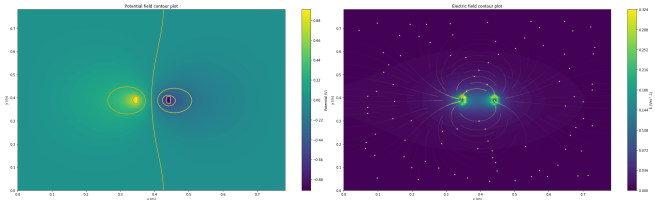
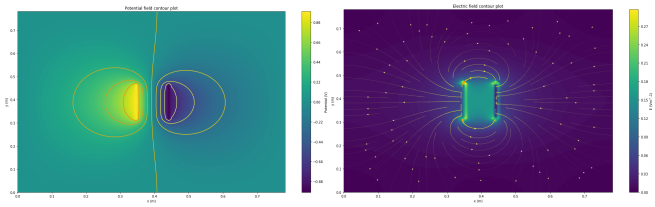


FIG. 4: Gauss-Seidel Method: number of iterations vs J with model fitting. Standard deviation = 0.0137.



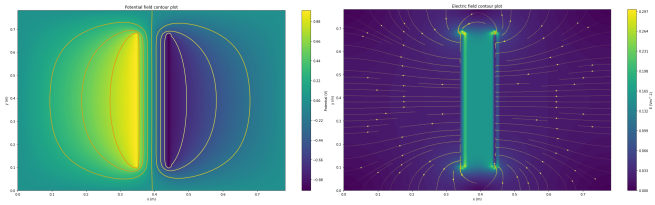
(a) Potential Field (b) Electric Field

FIG. 5: Parallel plate capacitor, a = 5m, d = 12m.



(a) Potential Field (b) Electric Field

FIG. 6: Parallel plate capacitor, a = 20m, d = 12m.



(a) Potential Field (b) Electric Field

FIG. 7: Parallel plate capacitor, a = 70m, d = 12m.

Physics Problem One

The Gauss-Seidel method was used to calculate the potential and electric fields in and around a parallel plate capacitor. Figures 5 to 7 show these fields for varying capacitor width *a*, and constant plate separation *d*. When $\frac{a}{d}$ is small, as in figure 5b, the capacitor resembles a dipole. As this ratio gets larger (figure 7), the field configuration approaches the infinite plate solution. This is where $E = \frac{V}{d}$ between the plates and $E = 0$ elsewhere. Figure 7a demonstrates clearly that between the two plates, the potential field has a constant gradient. Additionally, figure 7b shows that between the two plates the electric field is linear and constant, as expected. At the ends of the plates it can be seen that the infinite plate solution begins to break down, the field lines begin to curve away from the region of constant *E*.

Physics Problem Two

In this problem the diffusion equation (3) was solved for an iron poker. This was done for two different boundary conditions. For case one, one end of the poker is in a furnace of temperature *1273.15* K and the other end experiences no heat loss (figure 8). In case two, one end of the poker is in the furnace and the other is immersed in a block of ice (*273.15* K). Heat loss through the edges of the rod is ignored throughout.

Figure 8 shows the results of case one. As time advances, heat gradually dissipates through the poker. Because there is no heat lost from the system, the heat of the rod continues to build up until all of the rod is at *1273.15* K. Within a given temperature error tolerance the poker can reach equilibrium in a finite number of iterations.

Figure 9 shows the results of case two. The equilibrium point for this system is a linear distribution of temperature between the two ends of the poker. In the previous case 'true

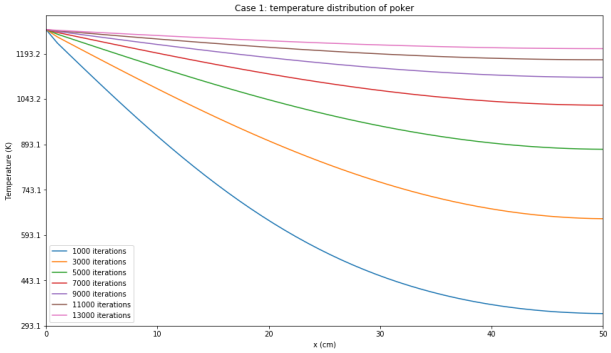


FIG. 8: Case one: temperature distribution of the poker evolving in time. Absolute temperature error set at 1×10^{-2}.

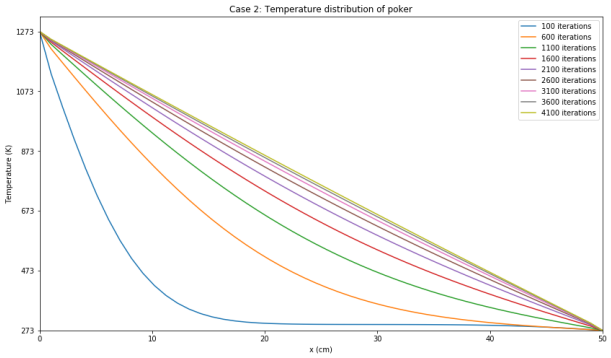


FIG. 9: Case two: temperature distribution of the poker evolving in time. Absolute temperature error set at 1×10^{-2}.

equilibrium' could not be reached in a finite number of iterations. However, the introduction of the second boundary condition in this case means that true equilibrium can be reached.

In this problem two sets of boundary conditions were imposed. The first of these were Dirichlet boundary conditions - the temperatures that were constant in time on either end of the poker. These were imposed by setting them after every iteration. The second boundary conditions were the Neumann boundary conditions. Every diagonal term in the matrix of prefactors is equal to $1 + \frac{2\alpha\delta t}{h^2}$, the factor of two is due the two nearest neighbours being taken into account. Because the problem specified that there should be no heat loss through the sides, only one neighbour was taken into account when considering the first and last nodes. This meant using a prefactor of $1 + \frac{\alpha\delta t}{h^2}$.

CONCLUSIONS

In conclusion, evidence has been found to support the hypothesis that the Gauss-Seidel method has a factor of two improvement in the number of iterations over the Jacobi method. It has also been shown that the proportionality of the number of iterations with J^2 means these methods are impractical for large grid densities. Finite different equations have been used to solve both Laplace's equation and the diffusion equation. The Laplace equation was applied to a parallel plate capacitor. It has been shown that as the ratio $\frac{a}{d}$ gets larger, the field configuration approaches the infinite plate solution. Solutions to the diffusion equation for an iron poker have been found for two different cases. It has been found that for equilibrium to be achieved a finite amount of iterations, there must be a boundary condition applied to either end of the poker.

[1] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[2] Patrick J Roache. *Computational fluid dynamics*. Number BOOK. Hermosa publishers, 1972.

[3] Dale Anderson, John C Tannehill, and Richard H Pletcher. *Computational fluid mechanics and heat transfer*. CRC Press, 2016.

[4] Diego A Murio. Implicit finite difference approximation for time fractional diffusion equations. *Computers & Mathematics with Applications*, 56(4):1138–1145, 2008.

[5] Joe D Hoffman and Steven Frankel. *Numerical methods for engineers and scientists*. CRC press, 2001.

[6] Christian Hill. *Learning scientific programming with Python*. Cambridge University Press, 2016.