Week 4- Lecture 1 and 2

Computational Methods II (Elliptic)

Dr. Farzad Ismail

School of Aerospace and Mechanical Engineering
Universiti Sains Malaysia
Nibong Tebal 14300 Pulau Pinang

Overview

- We already know the nature of hyperbolic and parabolic PDE's.
- Now we will focus on elliptic PDE.
- The elliptic PDE is a type of PDE for solving incompressible flow.

Overview (cont'd)

- Elliptic PDE's are always smooth
- Easier to obtain accurate solutions compared to hyperbolic problems, the challenge is in getting solutions <u>efficiently</u>.
- Unlike hyperbolic and parabolic problems, information is transmitted everywhere

Overview (cont'd)

- The model problem that will be discussed is

$$u_{xx} + u_{yy} = f(x, y)$$

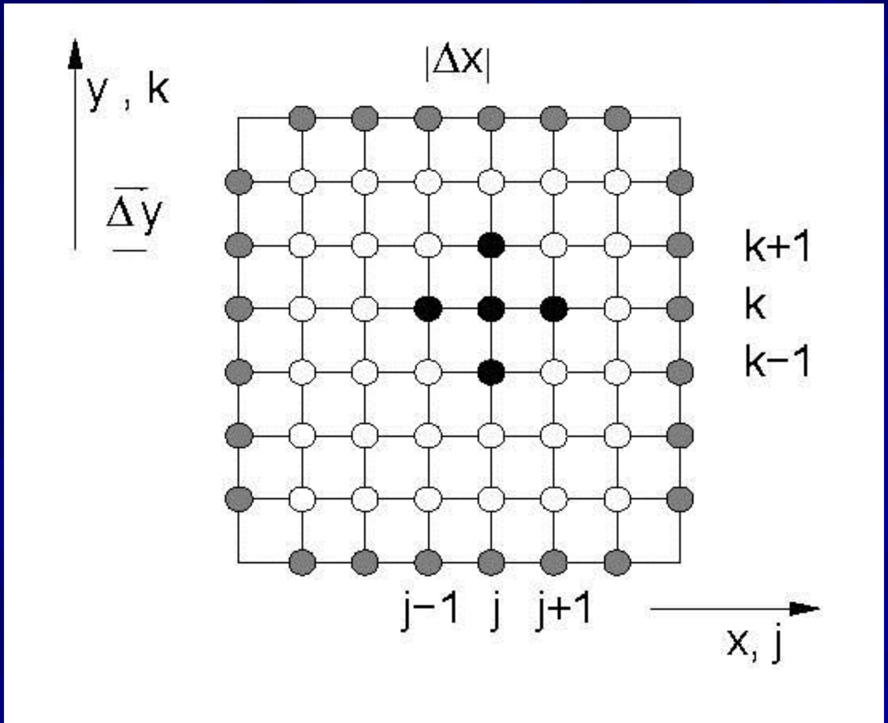
Poisson Eqn

$$u_{xx} + u_{yy} = 0$$

Laplace Equation

- Note that the problem is now 2D but with no time dependence

Overview (cont'd)



$$u_{xx} \approx \frac{u^n_{j+1,k} - 2u^n_{j,k} + u^n_{j-1,k}}{\Delta x^2}$$

$$u_{yy} \approx \frac{u^n_{j,k+1} - 2u^n_{j,k} + u^n_{j,k-1}}{\Delta y^2}$$

Overview (cont'd)

- For special case where the grid sizes in x and y is identical

$$\frac{1}{4}(u_{j-1,k} + u_{j+1,k} + u_{j,k-1} + u_{j,k+1}) - u_{j,k} = r_{j,k} = 0$$

$r_{j,k}$ is the residual associated with cell (j,k)

- Solve for u(j,k) such that r(j,k)=0, yielding a system of matrix Mu = 0
- Assume M intervals for each direction, Gaussian elimination method needs $O(M^6)$ operations in 2D

Iteration Method

- Solving it directly is to expensive, usually an iteration method is employed.

- The simplest iteration method is the point Jacobi method – based on solving the steady state of parabolic pde

$$u_t = u_{xx} + u_{yy}$$

$$u^{n+1}_{j,k} = u^n_{j,k} + \omega[\frac{1}{4}(u^n_{j-1,k} + u^n_{j+1,k} + u^n_{j,k-1} + u^n_{j,k+1}) - u^n_{j,k}]$$

$$u^{n+1}_{j,k} = u^n_{j,k} + \omega r^n_{j,k}$$

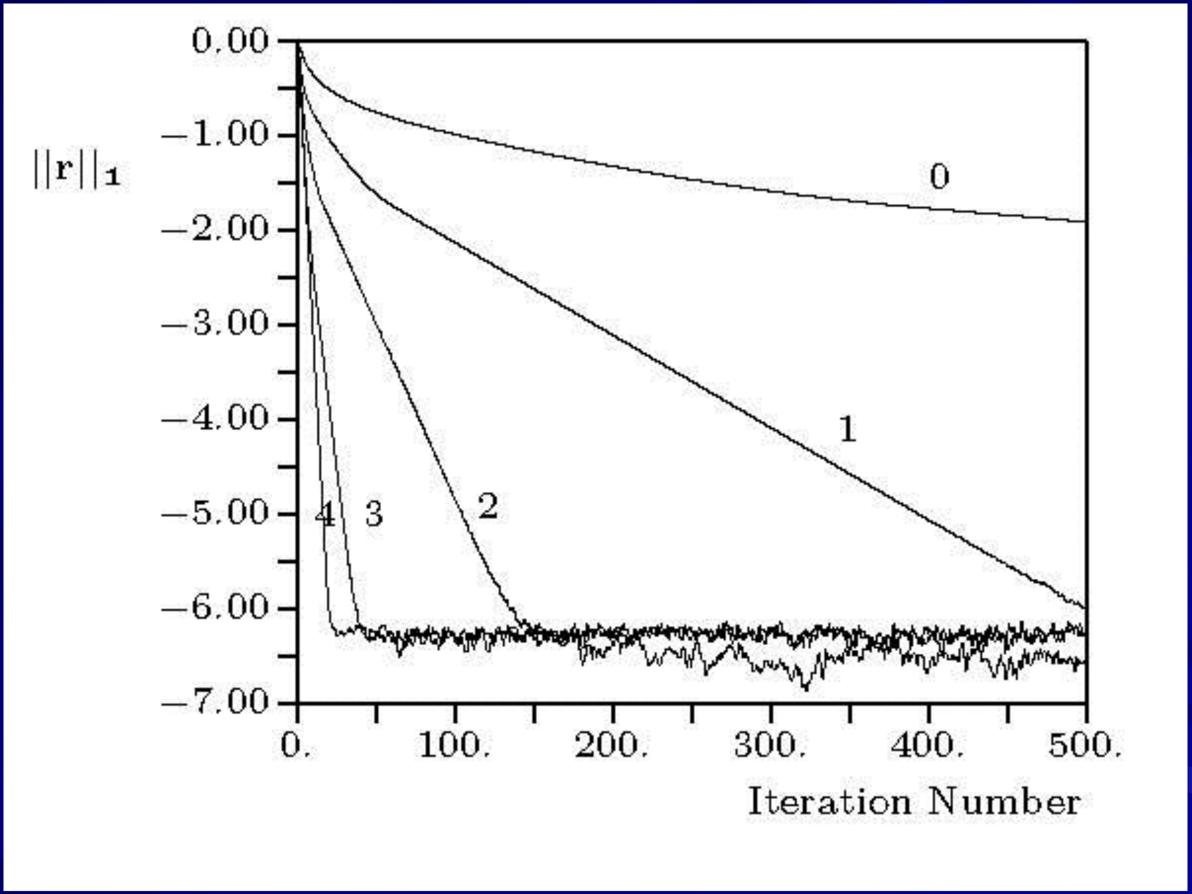
- Solve iteratively for $u^{n+1}_{j,k}$ until

$$u^{n+1}_{j,k} - u^n_{j,k} \approx 0$$

Iteration Method (cont'd)

- The superscript n denotes pseudo-time, not physical time
- ω is the relaxation factor, analogous to μ
- The first thought is to achieve steady-state as quickly as possible, hence choosing a largest ω consistent with stability
- Not a good idea since we need to know what is the best choice for ω

Typical Residual Plot



Residual Pattern

- Plot of convergence history
- Has three distinct phases
- First phase, the residual decays very rapidly
- Second phase decays linearly on the log-plot (or exponentially in real plot)
- Third phase is ‘noise’, since randomness has set in
  - after a while residuals become so small that they are of the order of round-off errors

Residual Pattern (cont’d)

- Sometimes the residual plot would ‘hang’ up
- Usually due to a ‘bug’ in the code
- To save time, apply a ‘stopping’ criterion to residual
- But not easy to do so, need to understand errors

Analyzing the Errors

- We want know how the iteration errors decay.
- Expand the solution as

$$u^n_{j,k} = u^\infty_{j,k} + \epsilon^n_{j,k}$$

- The first term on the RHS is the solution after infinite number of iterations
- The second term on the RHS is the error between the solution at iterative level n and after infinite iterations
- Get the best solution if error is removed

Analyzing the Errors (cont'd)

- Substitute the error relation into the iteration method.

$$u^{\infty}_{j,k} + \epsilon^{n+1}_{j,k} = u^{\infty}_{j,k} + \epsilon^{n}_{j,k} + \omega[\frac{1}{4}(u^{\infty}_{j-1,k} + u^{\infty}_{j+1,k} + u^{\infty}_{j,k-1} + u^{\infty}_{j,k+1}) - u^{\infty}_{j,k}$$
$$+ \frac{1}{4}(\epsilon^{n}_{j-1,k} + \epsilon^{n}_{j+1,k} + \epsilon^{n}_{j,k-1} + \epsilon^{n}_{j,k+1}) - \epsilon^{n}_{j,k}]$$

- Converge solution gives zero residual, hence

$$\epsilon^{n+1}_{j,k} = \epsilon^{n}_{j,k} + \omega[\frac{1}{4}(\epsilon^{n}_{j-1,k} + \epsilon^{n}_{j+1,k} + \epsilon^{n}_{j,k-1} + \epsilon^{n}_{j,k+1}) - \epsilon^{n}_{j,k}]$$

- Iterations for error

Analyzing the Errors (cont’d)

- Shows that error itself follow an evolutionary pattern
- This is true for all elliptic problems
- Remarkably, we can determine how the error would behave even if we do not know the solution

Analyzing the Errors using VA

- In 2D, von Neumann analysis (VA)

$$\epsilon^n_{j,k} = \mathrm{Real}(g^n \exp(i[j\theta_x + k\theta_y]))$$

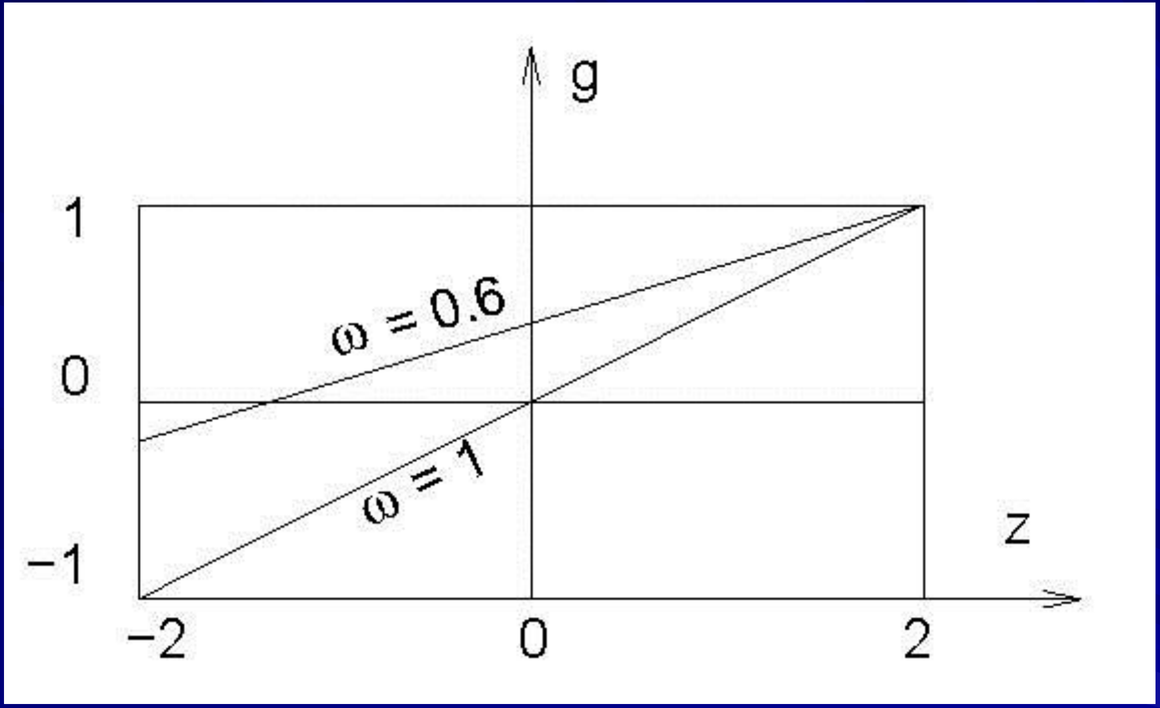
- Insert this into the error equation

$$\epsilon^{n+1}_{j,k} = \epsilon^n_{j,k} + \omega[\frac{1}{4}(\epsilon^n_{j-1,k} + \epsilon^n_{j+1,k} + \epsilon^n_{j,k-1} + \epsilon^n_{j,k+1}) - \epsilon^n_{j,k}]$$

- Yields the amplification factor for the errors

$$g = 1 - \frac{\omega}{2}[2 - (cos\theta_x + cos\theta_y)]$$

Amplification factor for Point Jacobi Method



$$z = (cos\theta_x + cos\theta_y)]$$

$$g = 1 - \frac{\omega}{2}[2 - z]$$

What does the figure tell you ???