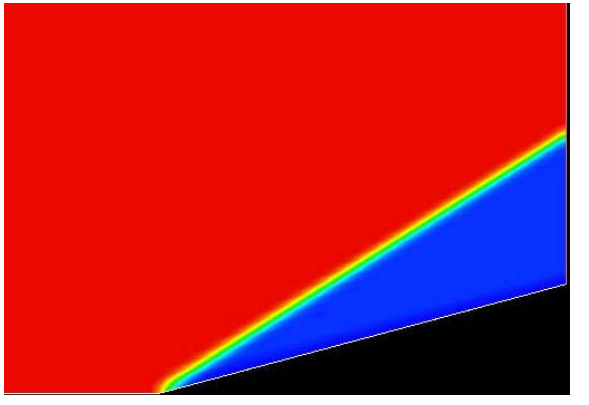
MAE 4230/5230: Introduction to CFD

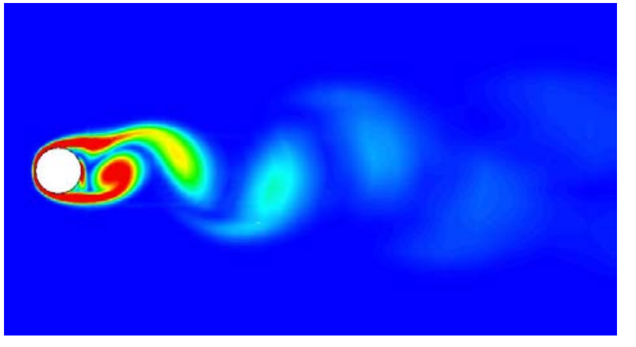
Rajesh Bhaskaran

Sibley School of Mechanical &
Aerospace Engineering

Cornell University

Ithaca, New York



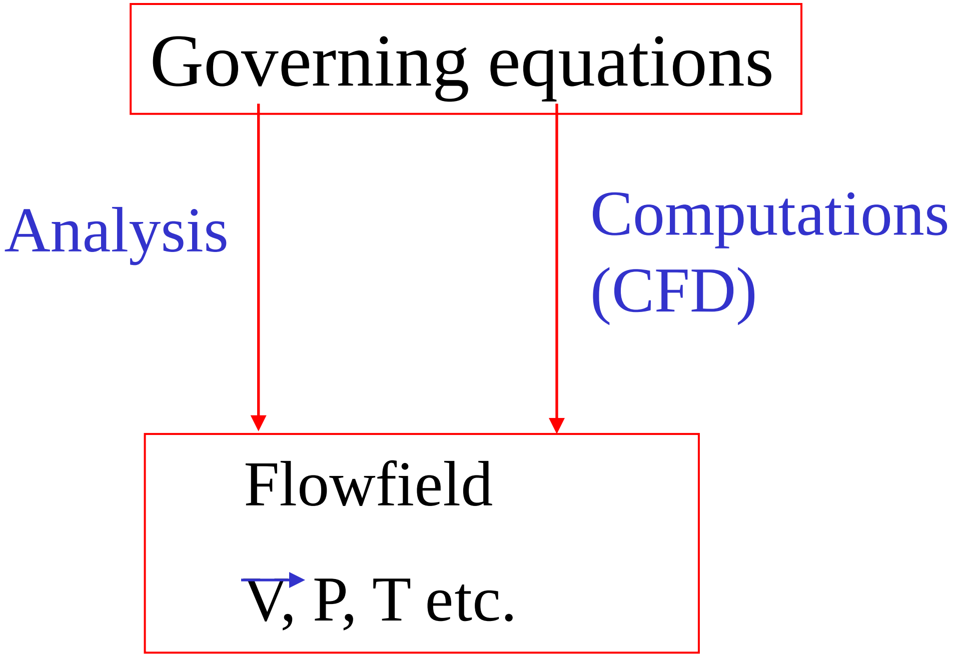


My Coordinates

- Dr. Rajesh Bhaskaran
  Swanson Director of Engineering Simulation
  Sibley School of Mechanical & Aerospace Engineering
- E-mail: bhaskaran@cornell.edu
- Office: 102 Rhodes
- Office hours:
  - Come with questions about FLUENT
  - Held in Swanson Lab (163 Rhodes)
  - Time to be announced

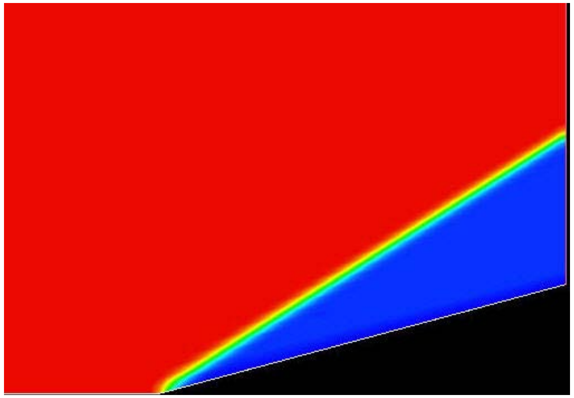
Introduction to CFD

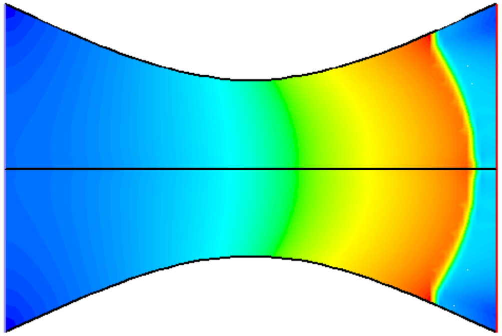
Intro to CFD handout on blackboard



Introduction to CFD

- Approach:
  - Go through a series of case studies in the use of CFD to analyze flow problems
  - Case studies to be performed using FLUENT
- Goals of the CFD case studies:
  - Build an understanding of the foundations of CFD.
  - Use hands-on learning to develop better physical feel for fluid flows and reinforce theory.





Introduction to CFD

- Before embarking on the CFD case studies, need to understand the rudiments of the CFD solution procedure operating under the hood of the software.
- In order to understand the CFD solution procedure, we will apply it to a simple model problem.
- Notes will be posted on course website.

Governing Equations for a Fluid

Continuity:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0$$

X – Momentum:

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial(\rho uw)}{\partial z} = -\frac{\partial p}{\partial x} + \frac{1}{Re_r}\left[\frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z}\right]$$

Y – Momentum:

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho v^2)}{\partial y} + \frac{\partial(\rho vw)}{\partial z} = -\frac{\partial p}{\partial y} + \frac{1}{Re_r}\left[\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z}\right]$$

Z – Momentum

$$\frac{\partial(\rho w)}{\partial t} + \frac{\partial(\rho uw)}{\partial x} + \frac{\partial(\rho vw)}{\partial y} + \frac{\partial(\rho w^2)}{\partial z} = -\frac{\partial p}{\partial y} + \frac{1}{Re_r}\left[\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z}\right]$$

Energy:

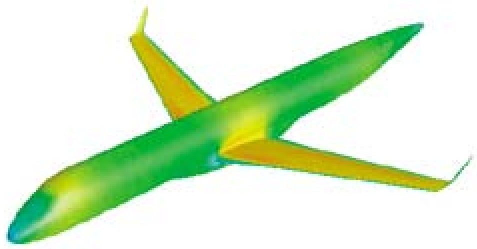
$$\frac{\partial(E_T)}{\partial t} + \frac{\partial(uE_T)}{\partial x} + \frac{\partial(vE_T)}{\partial y} + \frac{\partial(wE_T)}{\partial z} = -\frac{\partial(up)}{\partial x} - \frac{\partial(vp)}{\partial y} - \frac{\partial(wp)}{\partial z} - \frac{1}{Re_r Pr_r}\left[\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z}\right]$$
$$+ \frac{1}{Re_r}\left[\frac{\partial}{\partial x}(u\,\tau_{xx} + v\,\tau_{xy} + w\,\tau_{xz}) + \frac{\partial}{\partial y}(u\,\tau_{xy} + v\,\tau_{yy} + w\,\tau_{yz}) + \frac{\partial}{\partial z}(u\,\tau_{xz} + v\,\tau_{yz} + w\,\tau_{zz})\right]$$

Introduction to CFD

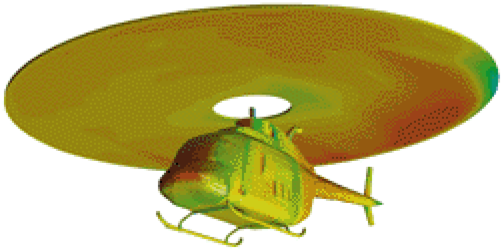
- Not possible to solve the governing equations analytically for most engineering problems.
- However, it is possible to obtain *approximate* computer-based solutions for many engineering problems.
- This is the subject matter of CFD.

CFD Applications

Pressure distribution for airplane configuration

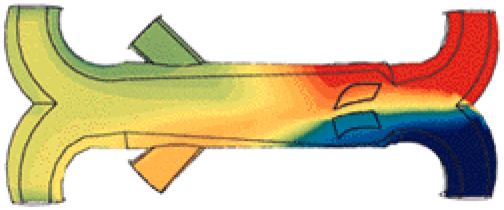


Pressure distribution for helicopter configuration



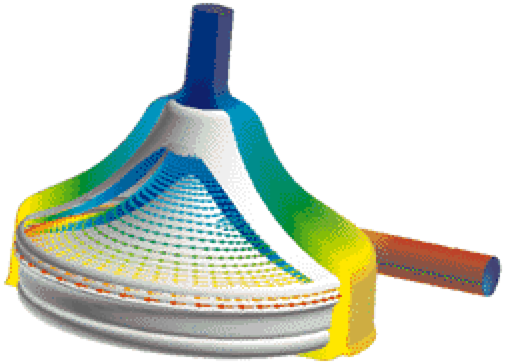
CFD Applications

Temperature distribution in a mixing manifold
(Boeing 767)

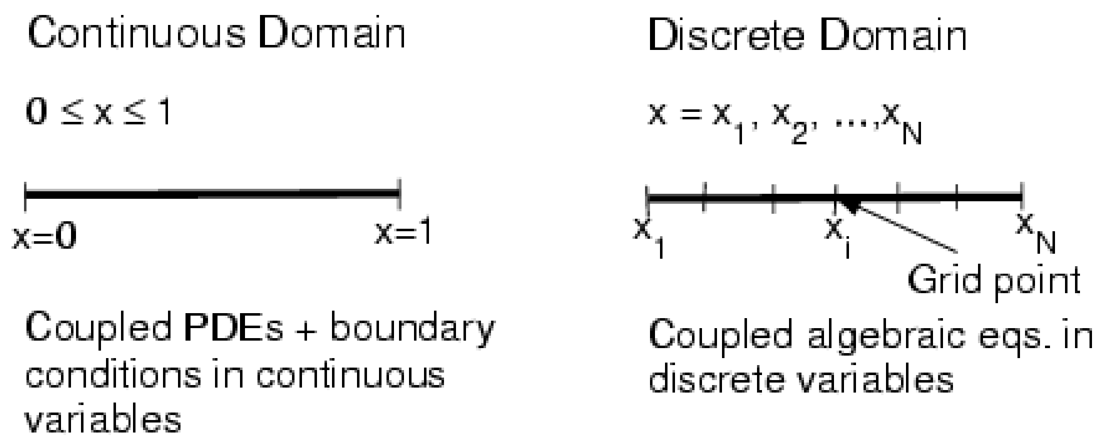


CFD Applications

Pressure contours and velocity vectors in a blood pump



Strategy of CFD



Example: Finite-Difference Approximation for du/dx

$$\left(\frac{du}{dx}\right)_i = \frac{u_i - u_{i-1}}{\Delta x} + O(\Delta x)$$

Numerical Solution of Model Equation

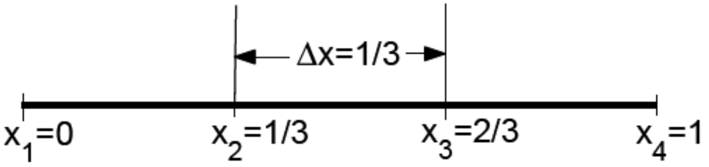
- Model equation (with m=1):

$$\frac{du}{dx} + u^m = 0; \quad 0 \le x \le 1; \quad u(0) = 1$$

- Finite-difference approximation:

$$\frac{u_i - u_{i-1}}{\Delta x} + u_i = O(\Delta x)$$

$$-u_{i-1} + (1 + \Delta x)u_i = 0$$



Numerical Solution of Model Equation

- System of four simultaneous algebraic equations:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1+\Delta x & 0 & 0 \\ 0 & -1 & 1+\Delta x & 0 \\ 0 & 0 & -1 & 1+\Delta x \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

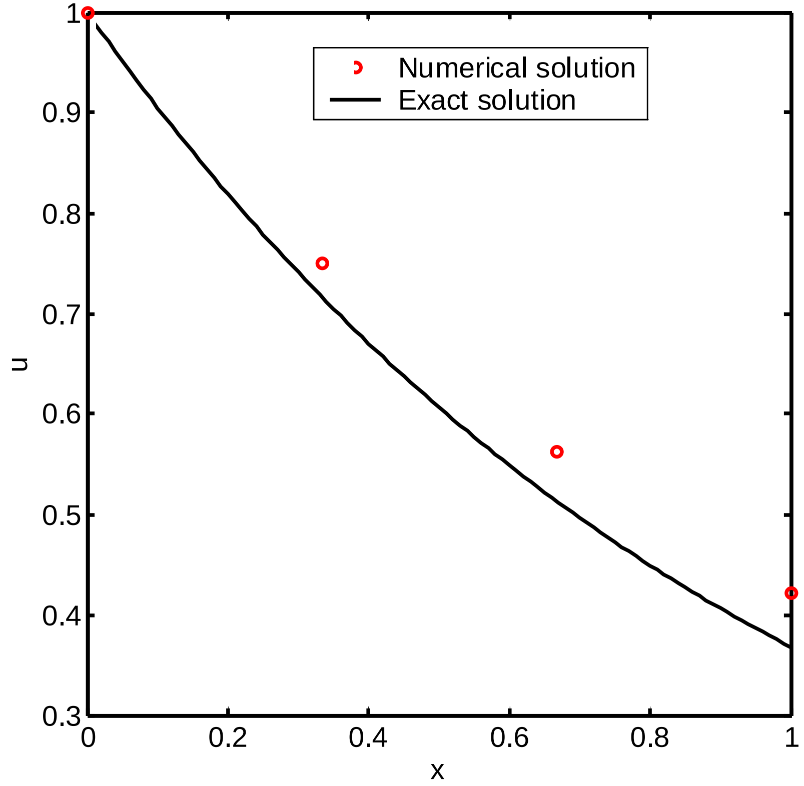
- Discrete solution:

$$u_1 = 1 \quad u_2 = 3/4 \quad u_3 = 9/16 \quad u_4 = 27/64$$

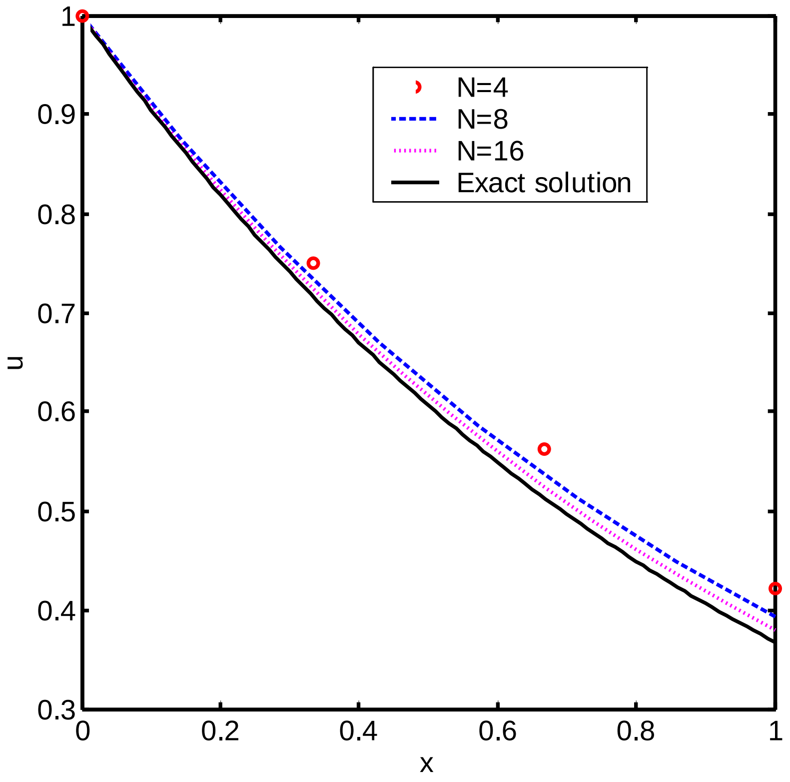
- Exact solution:

$$u_{exact} = \exp(-x)$$

1D Solution on 4-point grid



Grid Convergence of 1D Solution



Error on Different Grids

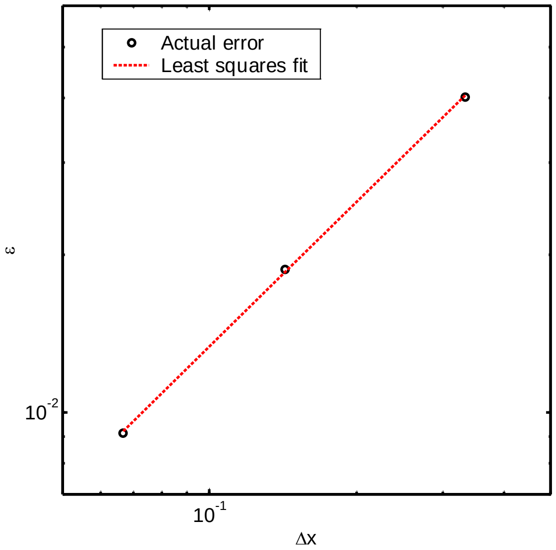
- Would like to know the error introduced by discretization on a given grid.
- In general, not possible to determine the actual values of the discretization error.
- However, we can estimate the *rate* at which error would decrease on refining the grid.
- One measure of error:

$$\epsilon = \sqrt{\frac{\sum_{i=1}^{N} \left(u_i - u_{i,exact}\right)^2}{N}}$$

Error on Different Grids

$\varepsilon = C \, \Delta x^{\alpha}$

$\alpha = 0.92$ from least squares fit



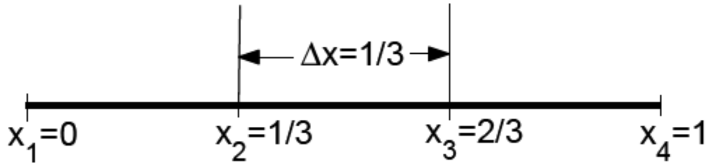
Numerical Solution: Second-Order Accuracy

- Model equation (with m=1):

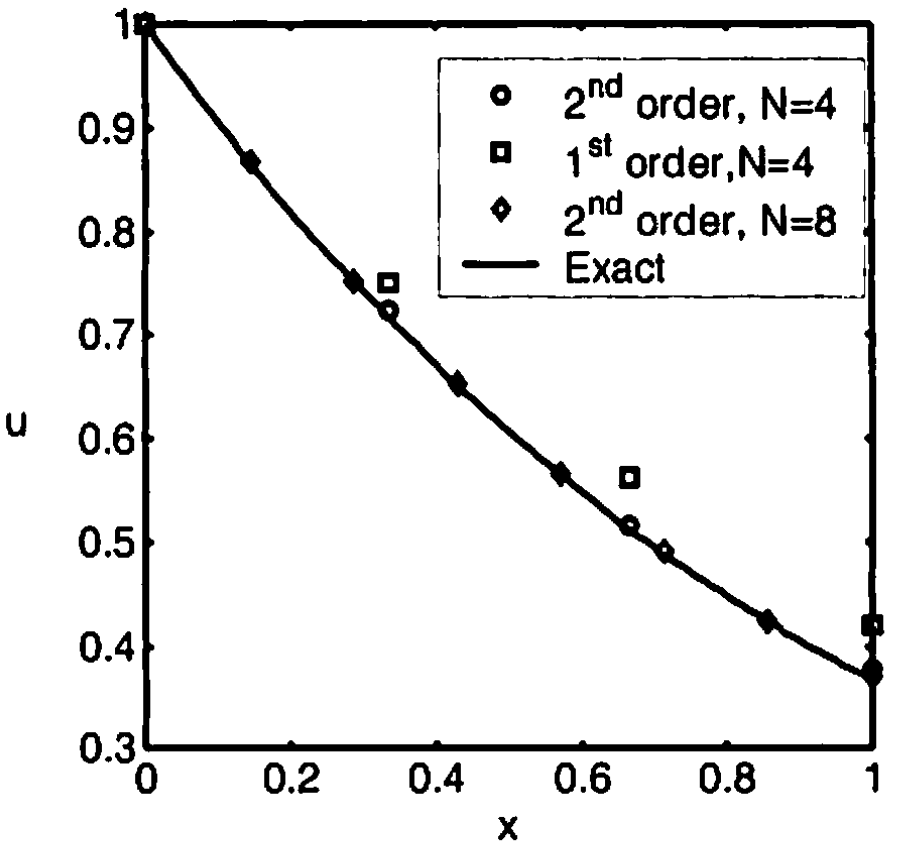
$$\frac{du}{dx} + u^m = 0; \quad 0 \leq x \leq 1; \quad u(0) = 1$$

- Second-order finite-difference approximation:

$$\left(\frac{du}{dx}\right)_i = \frac{u_{i+1} - u_{i-1}}{2\Delta x} + O(\Delta x^2)$$

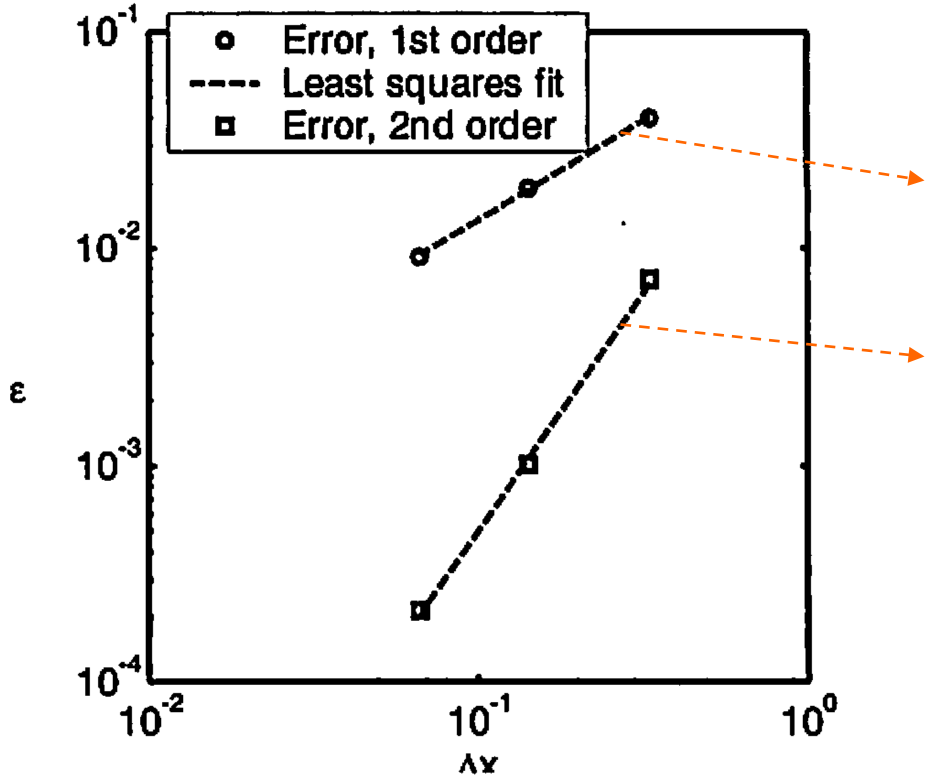


Comparison of First and Second-Order Solutions



Error on Different Grids

$\varepsilon = C\,\Delta x^{\alpha}$



$\alpha = 0.92$

$\alpha = 2.18$

Dealing with Nonlinearity

- Model non-linear equation

$$\frac{du}{dx} + u^2 = 0; \quad 0 \le x \le 1; \quad u(0) = 1$$

- Finite-difference approximation

$$\frac{u_i - u_{i-1}}{\Delta x} + u_i^2 = O(\Delta x)$$

- Linearize about guess value u_g

$$\frac{u_i - u_{i-1}}{\Delta x} + 2u_{g_i} u_i - u_{g_i}^2 = 0$$

- Linearization error = O[(u- u_g)^2]

Dealing with Nonlinearity

- Matrix system on four-point grid

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1+2\Delta x\, u_{g_2} & 0 & 0 \\ 0 & -1 & 1+2\Delta x\, u_{g_3} & 0 \\ 0 & 0 & -1 & 1+2\Delta x\, u_{g_4} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 1 \\ \Delta x\, u_{g_2}^2 \\ \Delta x\, u_{g_3}^2 \\ \Delta x\, u_{g_4}^2 \end{bmatrix}$$

- Iterate until $|u - u_g| / |u| <$ Tolerance

$$\begin{aligned} &\text{Iteration 1:} \quad u_g^{(1)} = \text{Initial guess} \\ &\text{Iteration 2:} \quad u_g^{(2)} = u^{(1)} \\ &\vdots \\ &\text{Iteration l:} \quad u_g^{(l)} = u^{(l-1)} \end{aligned}$$

- The difference $|u - u_g| / |u|$ is called the Residual

Dealing with Nonlinearity

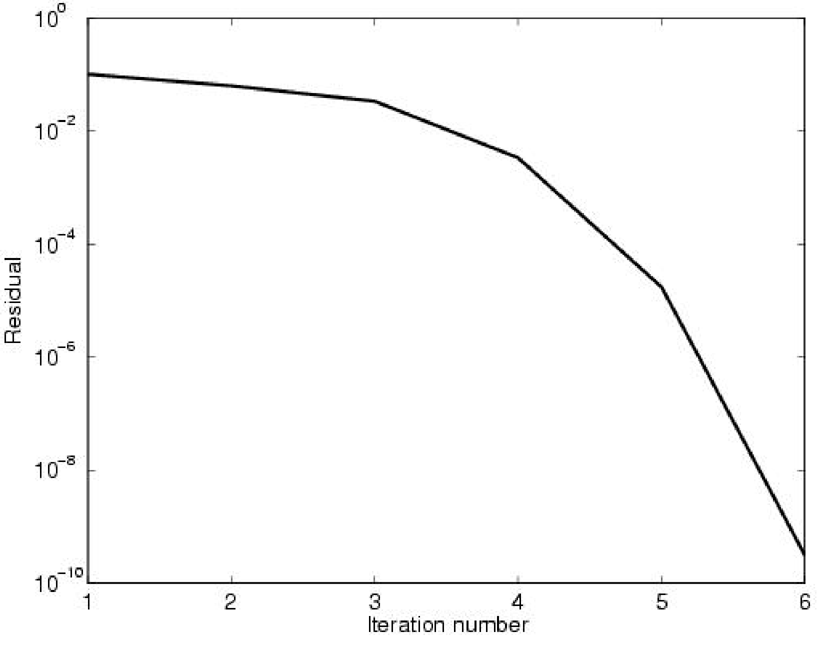
- Unscaled residual:

$$R \equiv \sqrt{\frac{\sum_{i=1}^{N} \left(u_i - u_{g_i}\right)^2}{N}}$$

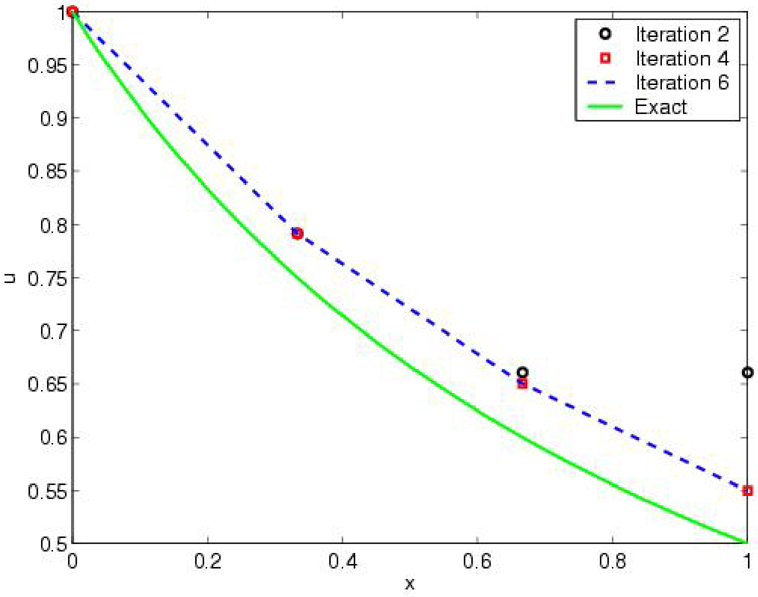
- Scaled residual:

$$R = \left(\sqrt{\frac{\sum_{i=1}^{N} \left(u_i - u_{g_i}\right)^2}{N}}\right) \left(\frac{N}{\sum_{i=1}^{N} |u|_i}\right) = \frac{\sqrt{N \sum_{i=1}^{N} \left(u_i - u_{g_i}\right)^2}}{\sum_{i=1}^{N} |u|_i}$$

Dealing with Nonlinearity



Linearization Example



Matrix inversion

- In each iteration, one can:
  1. Form the matrix and invert
  2. Sweep across the mesh updating each point in turn
     - Use guess value for those values in difference eq. that have not been updated
     - No need to form the matrix
     - Easier to code and faster
- See Intro to CFD handout on BB for details