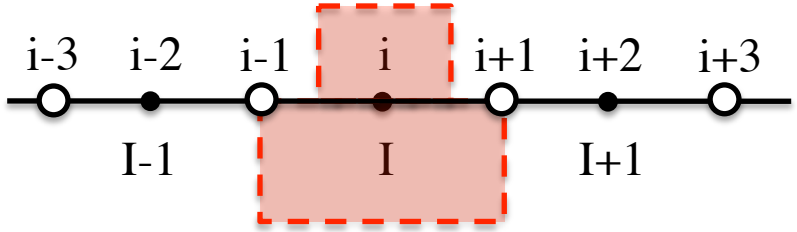
Computational Fluid Dynamics: Lectures 10 (ME EN 6720)

Prof. Rob Stoll
Department of Mechanical Engineering
University of Utah

Spring 2014

Multigrid Methods

- One of the drawbacks for point iterative schemes (and motivation for the previous block iterative schemes) was the slow propagation of BCs for B.V.P.s
- BC propagation can be especially painful for fine grid simulations
- One solution for this: use two grids, a coarse resolution one to propagate BCs and a second grid at the desired resolution to find the solution at our node points.



- For this method to work we will have to be able to translate between the two grids
- Why should this work?

  -For many methods (G-S, etc.) error (ε^k and ρ^k) as a function of space varies smoothly

  -Methods such as G-S are not only faster per iteration (cost is ¼ for 2D) they also converge in fewer total iterations (for G-S in 2D on a grid twice as coarse it takes ¼ as many iterations).

Multigrid Methods

- Normally the coarse grid is every other point as depicted on the previous slide
- **Example:** 1-D equation: $\frac{d^2\phi}{dx^2} = f_i$

-On a uniform grid using a CD scheme we can write: $\frac{1}{(\Delta x)^2}\left(\phi_{i-1} - 2\phi_i + \phi_{i+1}\right) = f_i$ (*)

-After k iterations we have an approximate solution with in a residual value ρ^k of

$$\frac{1}{(\Delta x)^2}\left(\phi_{i-1}^k - 2\phi_i^k + \phi_{i+1}^k\right) = f_i - \rho_i^k$$

-Subtracting this equation from (*) we get the following equation for the iteration error

$$\frac{1}{(\Delta x)^2}\left(\varepsilon_{i-1}^k - 2\varepsilon_i^k + \varepsilon_{i+1}^k\right) = \rho_i^k \qquad (**)$$

-On a grid that is twice as coarse we can write this equation as:

$$\frac{1}{4(\Delta x)^2}\left(\varepsilon_{i-2}^k - 2\varepsilon_i^k + \varepsilon_{i+2}^k\right) = \frac{1}{4}\left(\rho_{i+1}^k + 2\rho_i^k + \rho_{i-1}^k\right)$$

smoothing (average) note we could also use injection ($I=i$ at coincident points)

-Using our coarse grid variables, we can write this as:

$$\frac{1}{(\Delta X)^2}\left(\varepsilon_{I-1}^k - 2\varepsilon_I^k + \varepsilon_{I+1}^k\right) = \bar{\rho}_I^k \qquad (***)$$

where we have used the idea that our residuals are translated between the two grids (coarse and fine) using a smoothing operator.

Multigrid Methods

-From these error equations (** and ***) we can see that the residuals can be translated between the two grids by aggregation (averaging or injecting) or interpolation (linear etc)

-The goal is to find the iteration error (or corrections) on the coarse grid subject to the aggregated fine grid residuals that can be used to improve the fine grid solution.

-Out fine grid solution can then be written as:

$$\frac{1}{(\Delta x)^2}\left(\phi_{i-1}^k - 2\phi_i^k + \phi_{i+1}^k\right) + \varepsilon_i^k = f_i$$

The steps for the multigrid method are:

1. On the fine grid perform iterations with a 'smooth' method (usually only 4-5 iterations)
2. Compute the fine grid residuals
3. Restrict (or aggregate) the fine grid residuals to the coarse grid
4. Solve equation *** on the coarse grid to find ε_I^k
5. Interpolate ε_I^k values to the fine grid ε_i^k values. At points that coincide, the exact values are used and at in-between points linear interpolation is used.
6. Update the fine grid ϕ_i^k values using the interpolated iteration error and then repeat until convergence

Multigrid Methods

-An obvious extension is to continue this process (grid coarsening)

- we can nest further coarse grids within each coarse grid until we run out of points to aggregate (or coarsen)
- This can further improve the total cost of developing a converged solution (especially for 3D simulations)
- For the multigrid method the **number of iterations** at the finest grid level is basically **independent of the number of grid points**
- multigrid method will work with any 'smooth' method, examples include G-S and SIP (we can also use CGM and its family of methods).
- the 'smooth' requirement rules out combining SOR and multigrid in the same scheme.

Convergence

- In all iterative methods we have to choose a criteria for **when to quit**.
- Typical criteria are based on estimates of a norm of the iteration error where the iteration error can be interpreted as the difference between two successive estimates: ε_i^k

$$\varepsilon_i^k = \phi_i^{k+1} - \phi_i^k$$

- two criteria are the most common (with variations of the 1st the most popular):

1. L_1 norm: $\sum_{i=1}^{N} |\varepsilon_i| = \sum_{i=1}^{N} |\phi_i^{k+1} - \phi_i^k|$
2. L_2 norm: $\left[\sum_{i=1}^{N} |\varepsilon_i|^2\right]^{\frac{1}{2}} = \left[\sum_{i=1}^{N} |\phi_i^{k+1} - \phi_i^k|\right]^{\frac{1}{2}}$
3. L_∞ norm: $\left[\max |\varepsilon_i|\right] = \left[\max |\phi_i^{k+1} - \phi_i^k|\right]$

- Note we can also use normalized versions of these by dividing our norms by the total number of points
- Usually we stop when we have reduced the error by 3-4 orders of magnitude from its initial value.