SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF AEROSPACE ENGINEERING

19ASB304 – COMPUTATIONAL FLUID DYNAMICS FOR AEROSPACE APPLICATIONS

III YEAR VI SEM

UNIT-II DISCRETIZATION

TOPIC: Boundary layer equations and methods of solution

NAME: Mr.N.Venkatesh., M.Tech
Assistant Professor
Aerospace Engineering
SNS College of Technology

Overview

- Structured vs. Unstructured meshing approaches
- Development of an efficient unstructured grid solver
  - Discretization
  - Multigrid solution
  - Parallelization
- Examples of unstructured mesh CFD capabilities
  - Large scale high-lift case
  - Typical transonic design study
- Areas of current research
  - Adaptive mesh refinement
  - Moving and overlapping meshes

CFD Perspective on Meshing Technology

- CFD Initiated in Structured Grid Context
  - Transfinite Interpolation
  - Elliptic Grid Generation
  - Hyperbolic Grid Generation
- Smooth, Orthogonal Structured Grids
- Relatively Simple Geometries

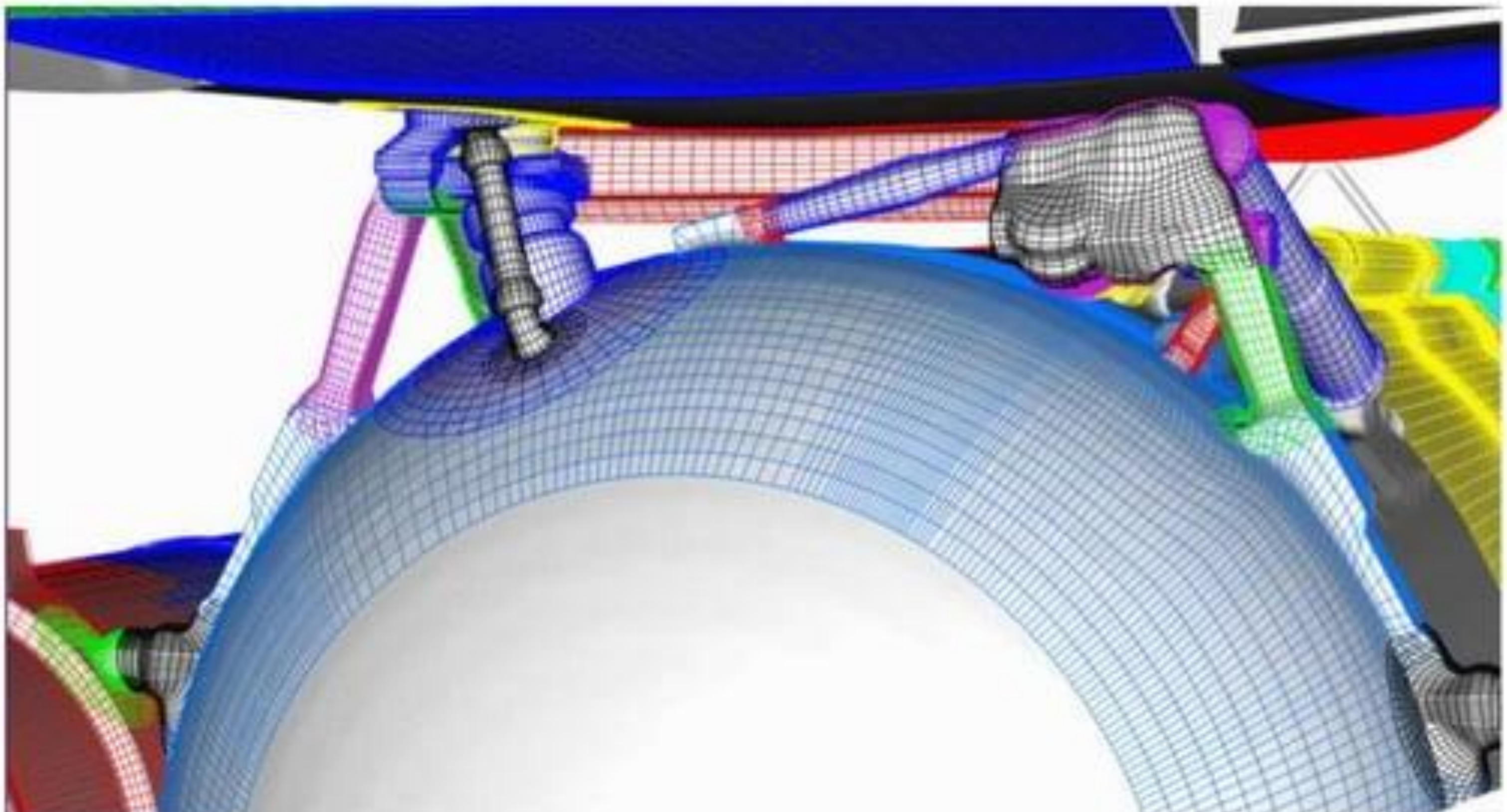
CFD Perspective on Meshing Technology

- Sophisticated Multiblock Structured Grid Techniques for Complex Geometries

Engine Nacelle Multiblock Grid by commercial software TrueGrid.

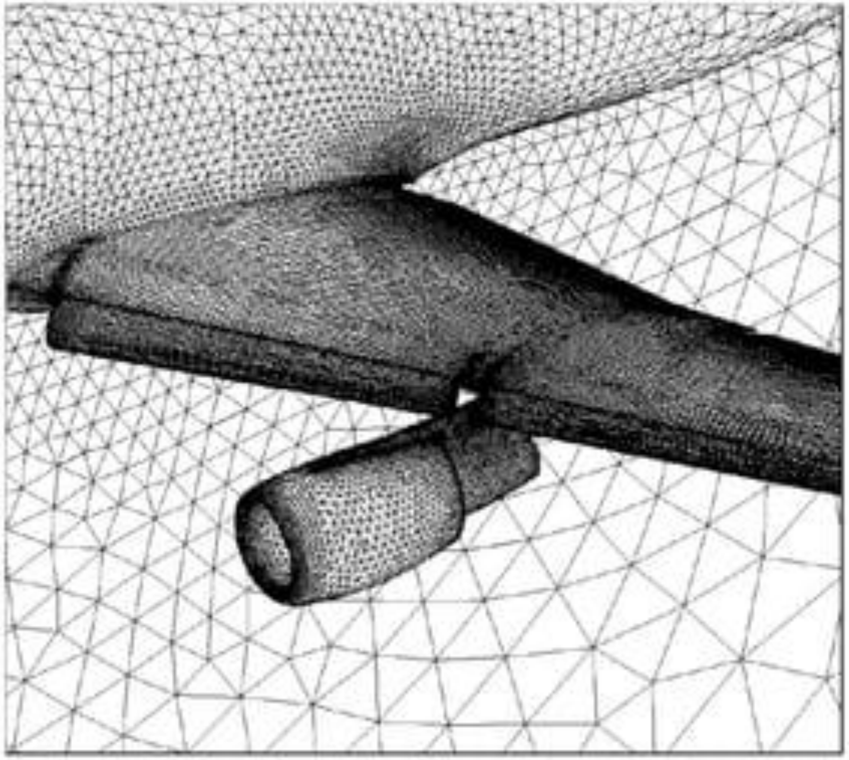
CFD Perspective on Meshing Technology

- Sophisticated Overlapping Structured Grid Techniques for Complex Geometries



Overlapping grid system on space shuttle (Slotnick, Kandula and Buning 1994)

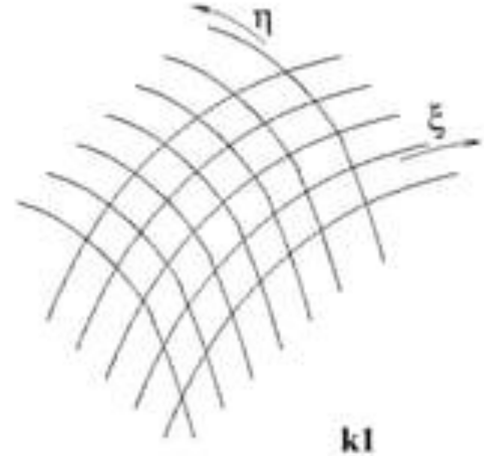
Unstructured Grid Alternative



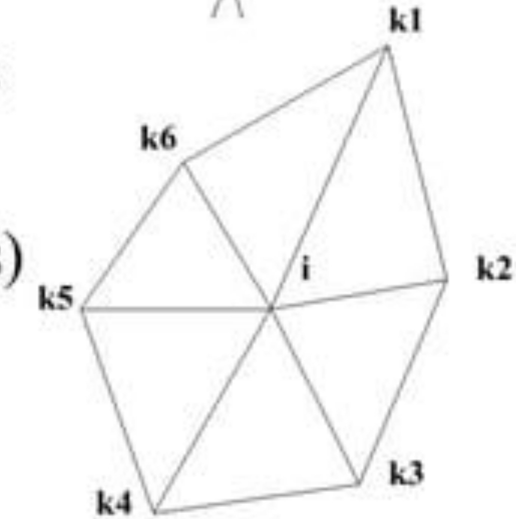
- Connectivity stored explicitly
- Single Homogeneous Data Structure

Characteristics of Both Approaches

- Structured Grids
  - Logically rectangular
  - Support dimensional splitting algorithms
  - Banded matrices
  - Blocked or overlapped for complex geometries



- Unstructured grids
  - Lists of cell connectivity, graphs (edge,vertices)
  - Alternate discretizations/solution strategies
  - Sparse Matrices
  - Complex Geometries, Adaptive Meshing
  - More Efficient Parallelization

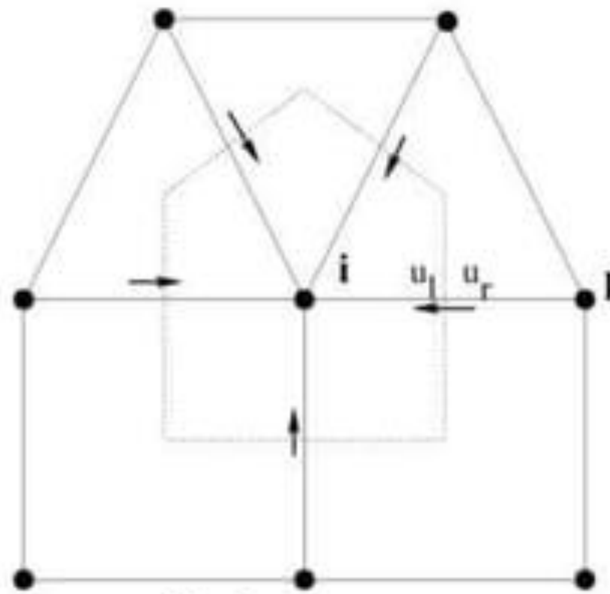


Discretization

- Governing Equations: Reynolds Averaged Navier-Stokes Equations
  - Conservation of Mass, Momentum and Energy
  - Single Equation turbulence model (Spalart-Allmaras)
    - Convection-Difusion – Production
- Vertex-Based Discretization
  - 2^{nd} order upwind finite-volume scheme
  - 6 variables per grid point
  - Flow equations fully coupled (5x5)
  - Turbulence equation uncoupled

Spatial Discretization

- Mixed Element Meshes
  - Tetrahedra, Prisms, Pyramids, Hexahedra
- Control Volume Based on Median Duals
  - Fluxes based on edges
    - $\mathbf{F}_{\mathrm{ik}} = f(\mathbf{u}_{\mathrm{left}}, \mathbf{u}_{\mathrm{right}})$
    - $\mathbf{u}_{\mathrm{left}} = \mathbf{u}_{\mathrm{i}}, \mathbf{u}_{\mathrm{right}} = \mathbf{u}_{\mathrm{k}}$: *1st order accurate*
    - $\mathbf{u}_{\mathrm{left}} = \mathbf{u}_{\mathrm{i}} + \frac{1}{2}\nabla \mathbf{u}_{\mathrm{i}}.\mathbf{r}_{\mathrm{ik}}$
    - $\mathbf{u}_{\mathrm{right}} = \mathbf{u}_{\mathrm{k}} + \frac{1}{2}\nabla \mathbf{u}_{\mathrm{k}}.\mathbf{r}_{\mathrm{ki}}$: *2nd order accurate*
    - ∇u_i *evaluated as contour integral around CV*



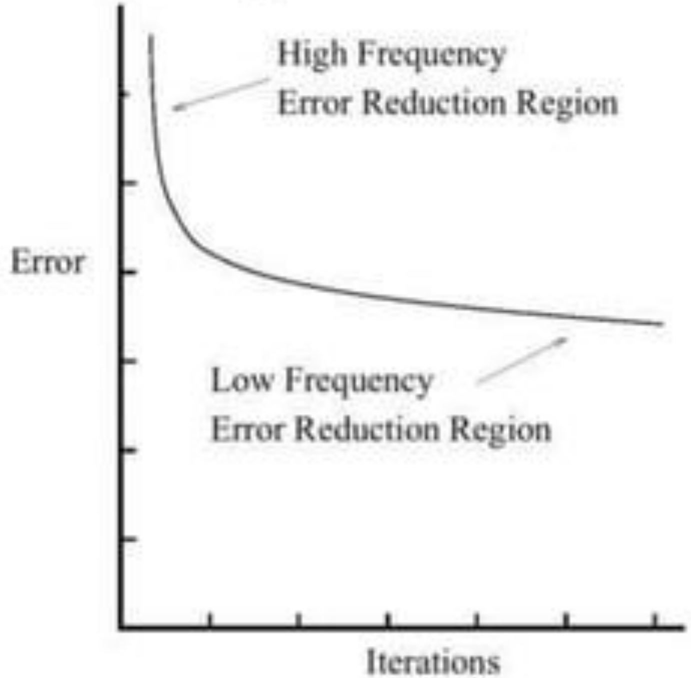
  - Single edge-based data-structure represents all element types

Spatially Discretized Equations

$$\frac{du}{dt} + \mathbf{R}(\mathrm{u}) = 0$$

- Integrate to Steady-state
- Explicit: $u^{n+1} = u^n - \Delta t\, \mathbf{R}(\mathrm{u^n})$
  - Simple, Slow: Local procedure
- Implicit $(\frac{I}{\Delta t} + \frac{\partial \mathbf{R}}{\partial u})(u^{n+1} - u^n) = -\Delta t\, \mathbf{R}(\mathrm{u^n})$
  - Large Memory Requirements
- Matrix Free Implicit: $\frac{\partial \mathbf{R}}{\partial \mathrm{u}} \Delta u = \frac{\mathbf{R}(\mathrm{u} + \epsilon \Delta \mathrm{u}) \quad \mathbf{R}(\mathrm{u})}{\epsilon}$
  - Most effective with matrix preconditioner
- Multigrid Methods

Multigrid Methods



- High-frequency (local) error rapidly reduced by explicit methods
- Low-Frequence (global) error converges slowly
- On coarser grid:
  - Low-frequency viewed as high frequency

Multigrid Correction Scheme (Linear Problems)

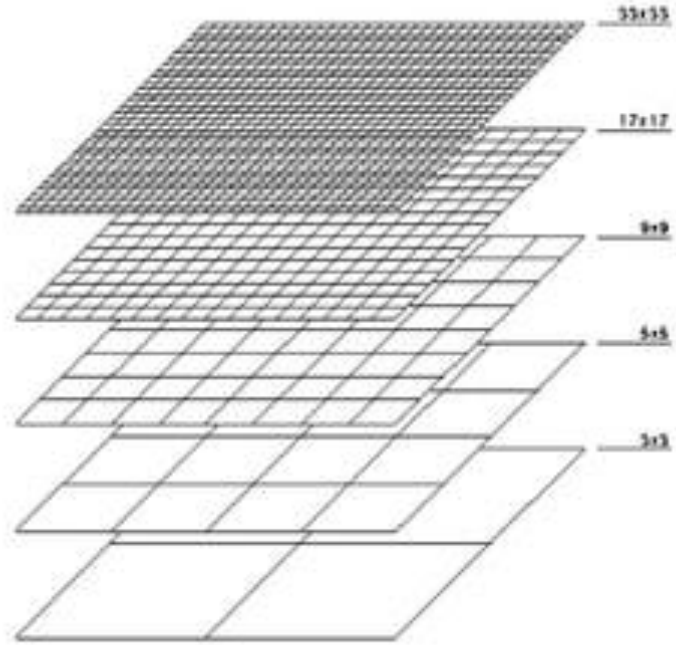
$$L_h u_h = f_h$$

$$L_h \overline{u}_h - f_h = r_h$$

$$u_h = \overline{u}_h + v_h$$

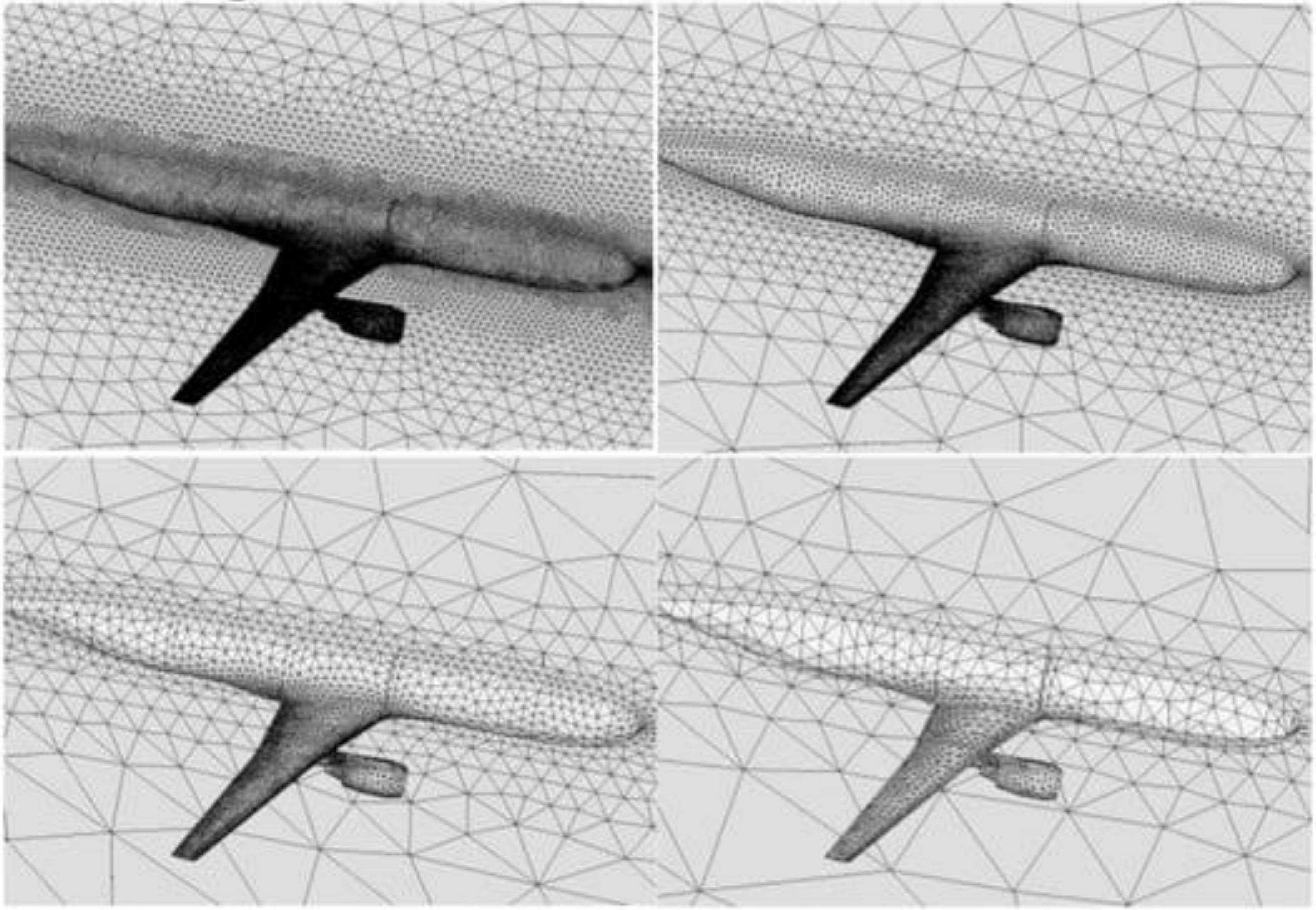
$$L_h u_h - L_h \overline{u}_h = -r_h$$

$$L_h v_h = -r_h$$



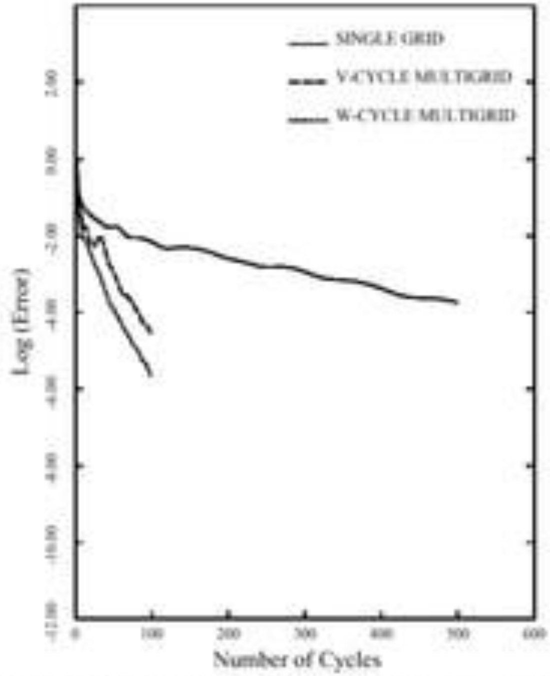
- *On Coarse Grid:* $L_H v_H = -I_h^H r_h$
- *Correct Fine Grid as:* $\overline{u}_h^{new} = \overline{u}_h + I_H^h v_H$

Multigrid for Unstructured Meshes



- Finest grid: 804,000 points, 4.5M tetrahedra
- Four level Multigrid sequence

Geometric Multigrid



- Order of magnitude increase in convergence
- Convergence rate equivalent to structured grid schemes
- Independent of grid size: O(N)

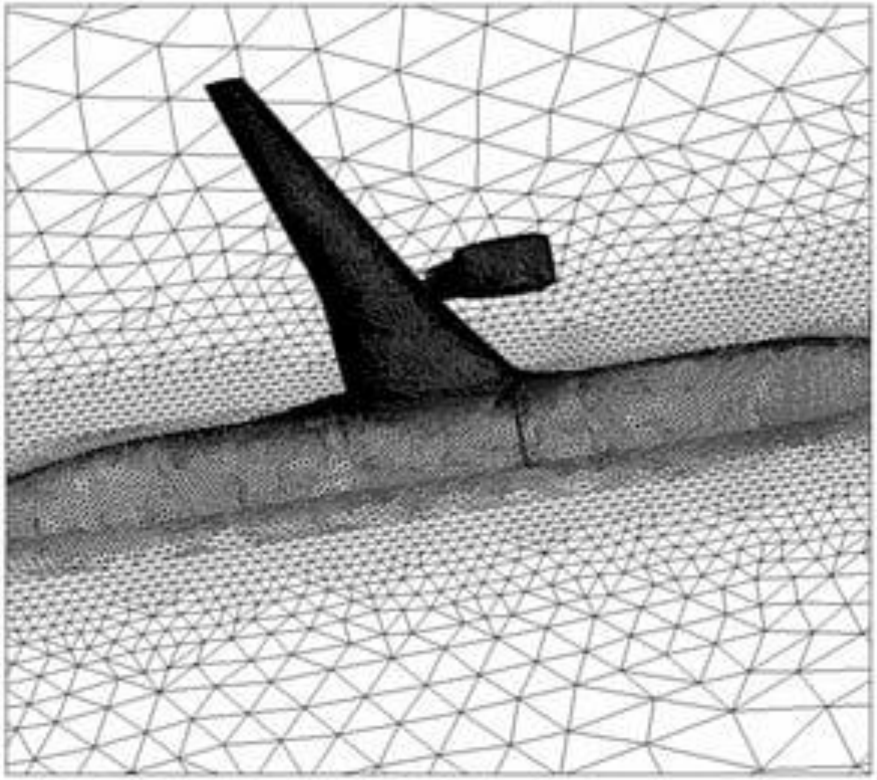
Agglomeration vs. Geometric Multigrid

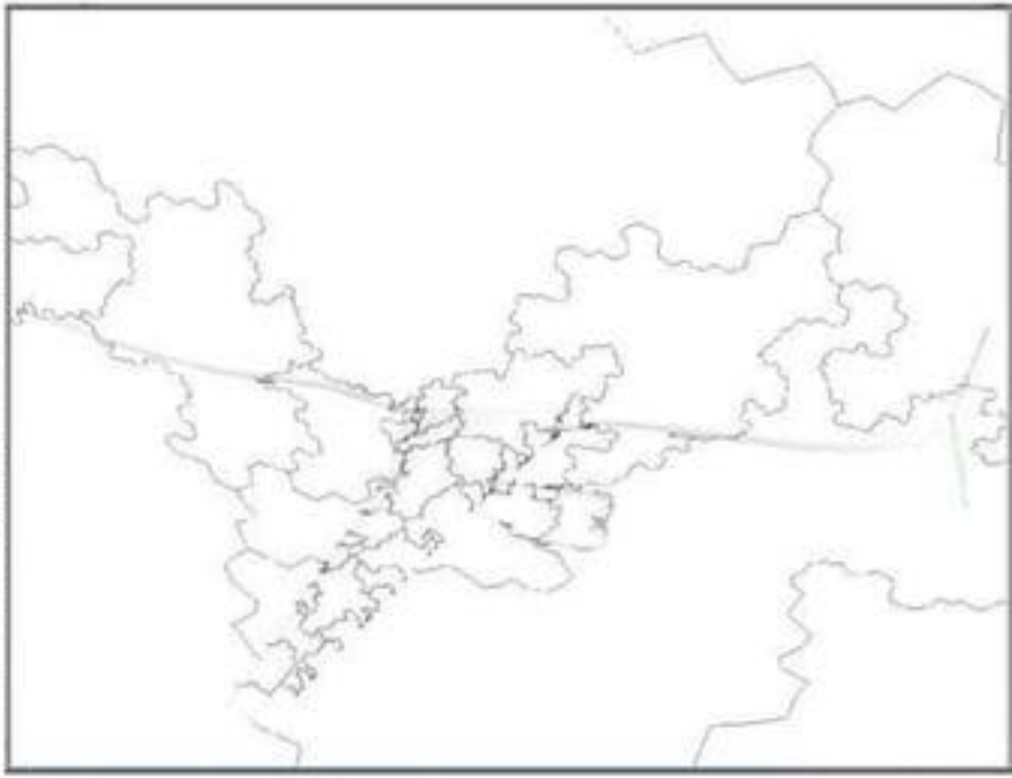
- Multigrid methods:
  - Time step on coarse grids to accelerate solution on fine grid
- Geometric multigrid
  - Coarse grid levels constructed manually
  - Cumbersome in production environment
- Agglomeration Multigrid
  - Automate coarse level construction
  - Algebraic nature: summing fine grid equations
  - Graph based algorithm

Agglomeration Multigrid

- Agglomeration Multigrid solvers for unstructured meshes
  - Coarse level meshes constructed by agglomerating fine grid cells/equations

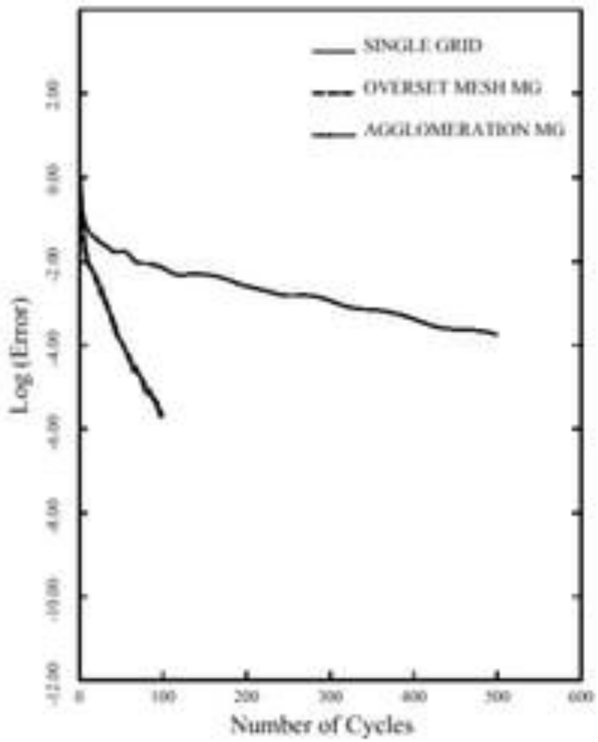
Agglomeration Multigrid





- Automated Graph-Based Coarsening Algorithm
- Coarse Levels are Graphs
- Coarse Level Operator by Galerkin Projection
- Grid independent convergence rates (order of magnitude improvement)

Agglomeration MG for Euler Equations

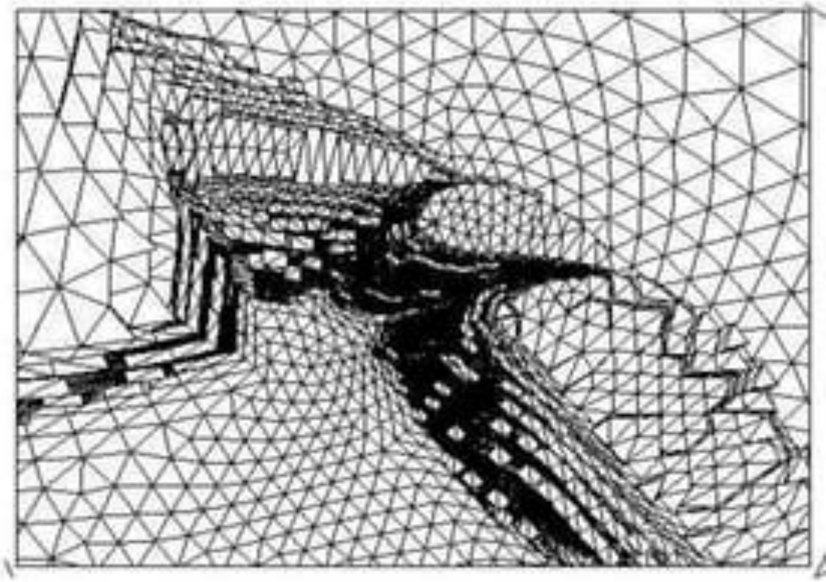


- Convergence rate similar to geometric MG
- Completely automatic

Anisotropy Induced Stiffness

- Convergence rates for RANS (viscous) problems much slower then inviscid flows

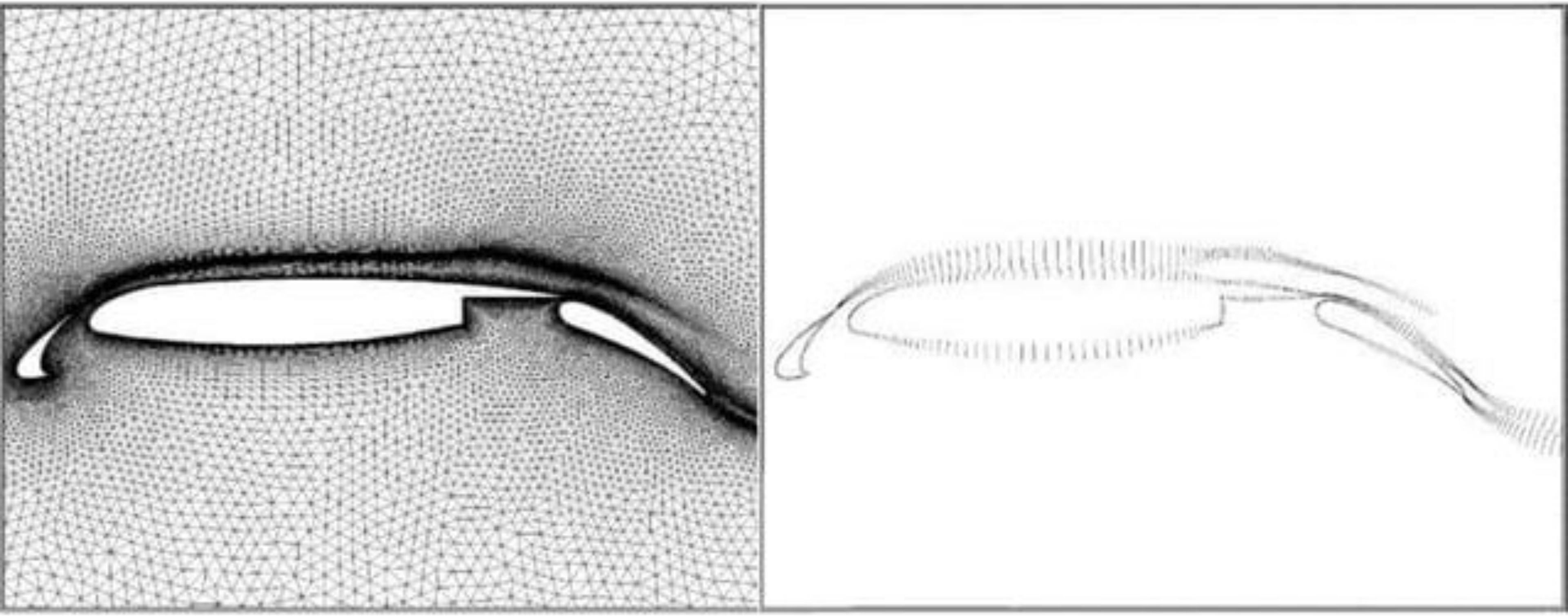
  - Mainly due to grid stretching
  - Thin boundary and wake regions
  - Mixed element (prism-tet) grids



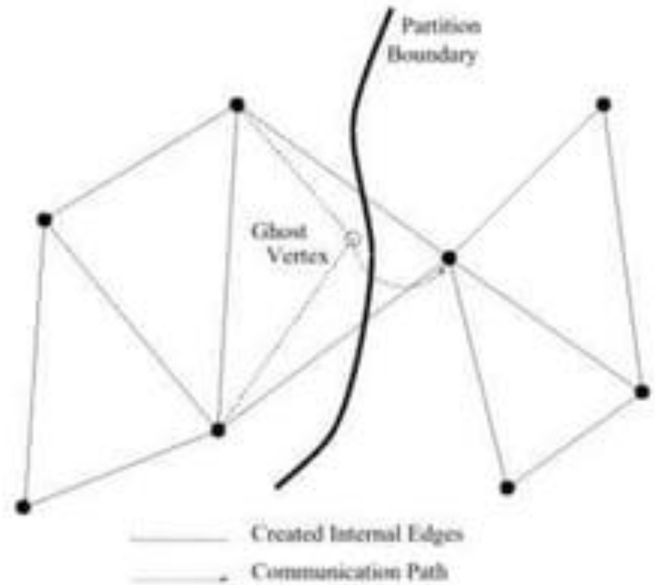
- Use directional solver to relieve stiffness
  - Line solver in anisotropic regions

Directional Solver for Navier-Stokes Problems

- Line Solvers for Anisotropic Problems
  - Lines Constructed in Mesh using weighted graph algorithm
  - Strong Connections Assigned Large Graph Weight
  - (Block) Tridiagonal Line Solver similar to structured grids



Implementation on Parallel Computers



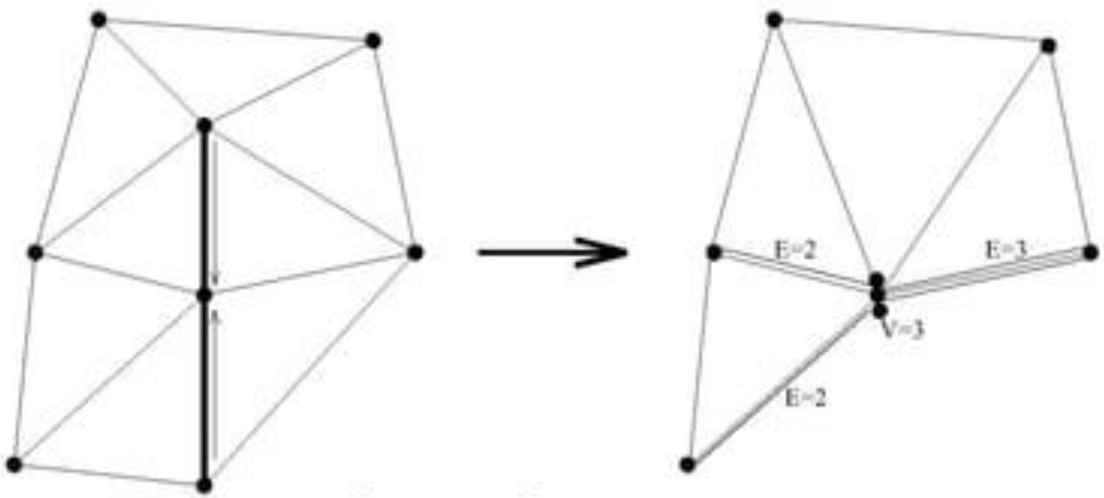
- Intersected edges resolved by ghost vertices
- Generates communication between original and ghost vertex
  - Handled using MPI and/or OpenMP
  - Portable, Distributed and Shared Memory Architectures
  - Local reordering within partition for cache-locality

Partitioning

- Graph partitioning must minimize number of cut edges to minimize communication
- Standard graph based partitioners: Metis, Chaco, Jostle
  - Require only weighted graph description of grid
    - Edges, vertices and weights taken as unity
  - Ideal for edge data-structure
- Line Solver Inherently sequential
  - Partition around line using weigted graphs

Partitioning

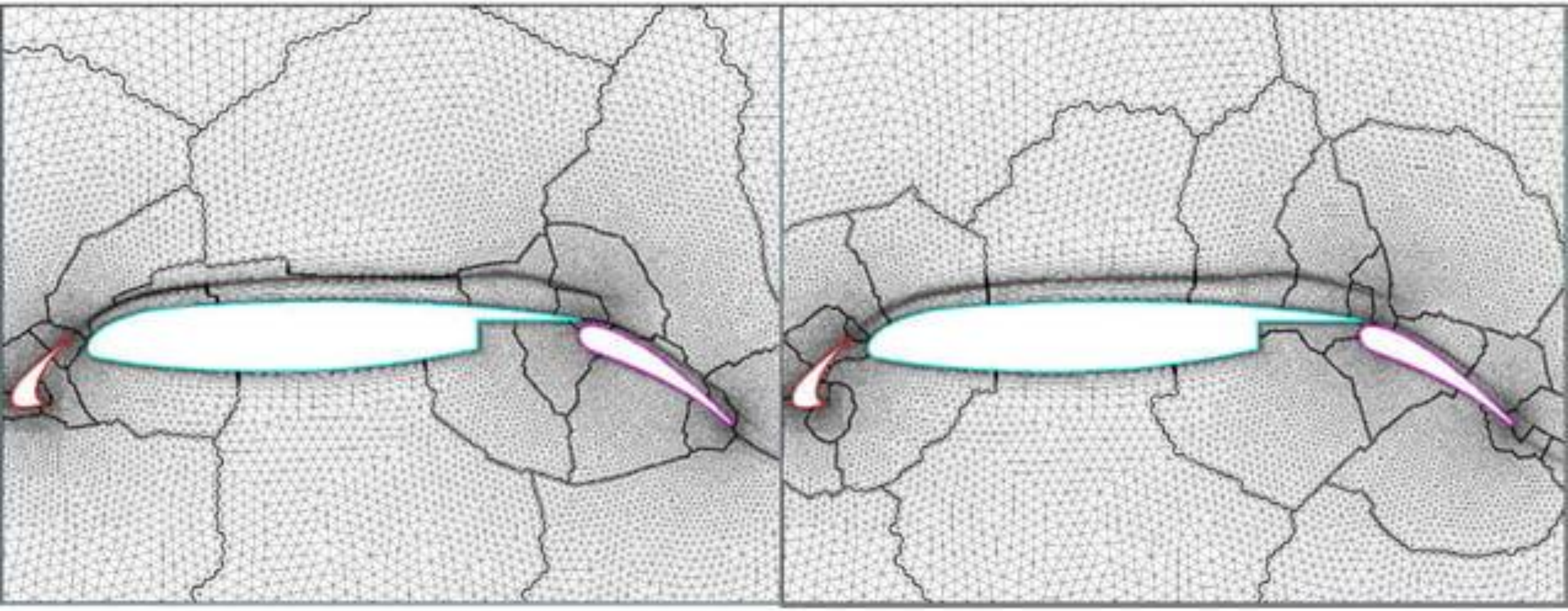
- Contract graph along implicit lines
- Weight edges and vertices



- Partition contracted graph
- Decontract graph
  - Guaranteed lines never broken
  - Possible small increase in imbalance/cut edges

Partitioning Example

- 32-way partition of 30,562 point 2D grid



- Unweighted partition: 2.6% edges cut, 2.7% lines cut
- Weigted partition: 3.2% edges cut, 0% lines cut