THE UNIVERSITY of TENNESSEE at CHATTANOOGA

SIMCENTER NATIONAL CENTER for COMPUTATIONAL ENGINEERING

An Adaptive Streamline/Upwind Petrov Galerkin Overset Grid Scheme for the Navier-Stokes Equations with Moving Domains

Chao Liu, Behzad R. Ahrabi, James C. Newman III, W. Kyle Anderson
SimCenter: National Center for Computational Engineering
University of Tennessee at Chattanooga
12th Symposium on Overset Composite Grids and Solution Technology

Motivation

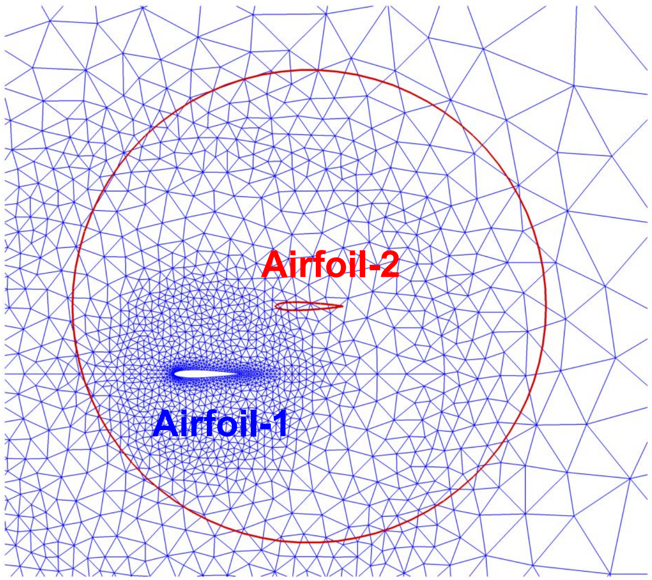
- Advantages of finite elements
  - Extendable to high-order accuracy
  - Stencil is contained inside the element
- Benefits for overset grid schemes
  - Minimal grid overlapping required
  - Facilitates hole cutting
  - Curved geometry poses minimal difficulties

Outline

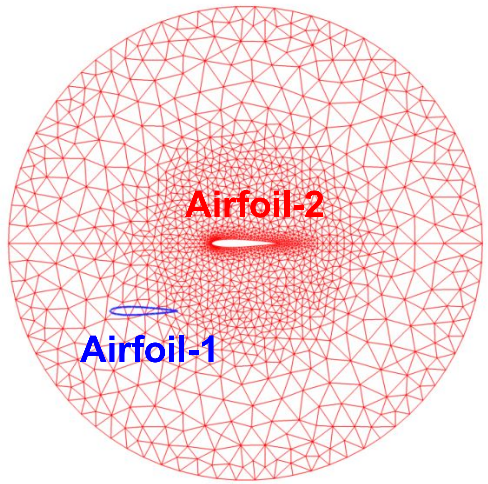
- Hole cutting
- Governing equations
- Overset methodology
- Overset results
- Adaptive overset
- Conclusion

Hole Cutting

- Hole cutting includes two steps
  - Identify invalid cells
  - Selection among valid cells



Grid-1

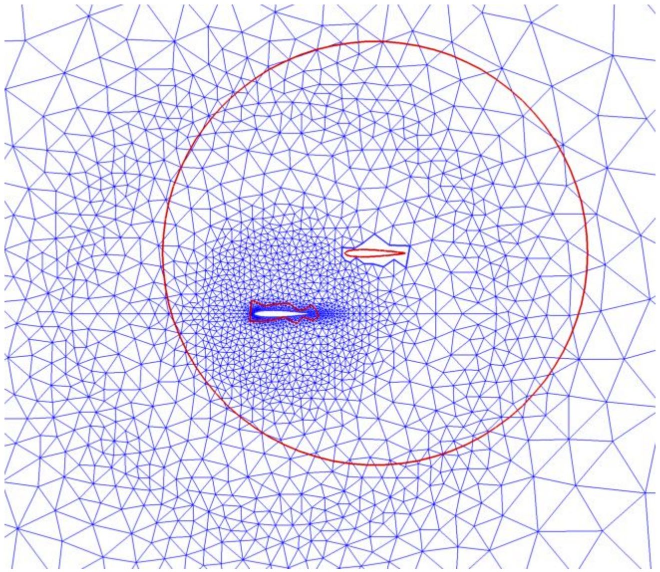


Grid-2

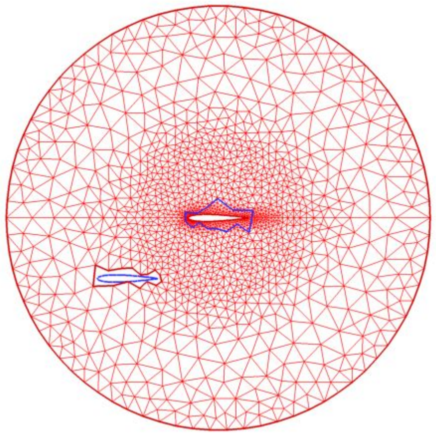
Example of 2 airfoil overset grids

Identify Invalid Cells

- On Grid-1, determine location of Airfoil-2. Cells in Grid-1 that intrude or lie inside of Airfoil-2 are invalid, and need to be removed from domain. Repeat procedure on Grid-2 for Airfoil-1.
- Direct wall cut is used to identify invalid cells



Grid 1



Grid 2

Grids after direct wall cut (all invalid cells removed)

Select Among Valid Cells

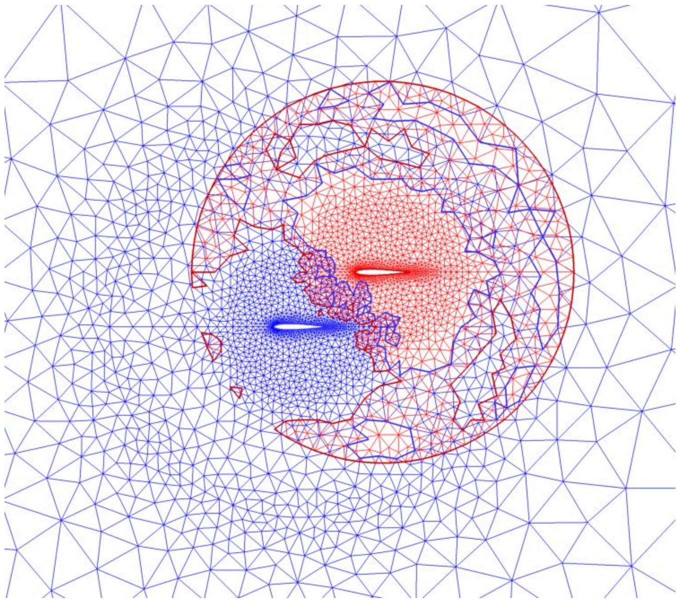
- To minimize grid overlapping, among the valid cells, certain cells are selected for simulation, the remainder are removed.
  - Mesh quality
  - Automation
  - Parallel
- No definitive selection process. Two approaches are explored:
  - Existing Implicit Hole Cutting (IHC) method
  - Novel Elliptic Hole Cutting (EHC) method

IHC

- Developed by Lee & Baeder, 2003
- A cell select process based on *cell-quality*
  - Each grid node is viewed as a sampling point
  - For each sampling point, all candidate donors are identified
  - Only the candidate with highest *cell-quality* is actived

- *cell-quality* is a grid metric (inverse of cell volume, aspect ratio...)
- User control is optional
  - Not needed in getting valid overset grids
  - In some cases, it's needed to make grid "continuous"
- All nodes in overlapping region need to be searched (expensive in parallel implementation)

IHC



Mesh after original IHC

- *Cell-quality* defined as the inverse of cell volume
- Smallest cells are selected across the whole domain
- High *cell-quality* does not gurantee a high-quality overset mesh. "Continuity" of cell selection is often more important

Elliptic Hole Cutting

- New approach. AIAA paper 2014-2980
- Solve a Poisson equation on each grid. Select the cells with the highest pseudo temperature.

$$\nabla^2 T = f$$

  - Assign high *T* to nodes you really want
  - Assign low *T* to nodes you really don't want

  $\}$ boundary conditions

  - Let Poisson solver take care the rest of the nodes

- No need to solve the exact Poisson problems
- No need for the solutions to fully converge

Elliptic Hole Cutting

- Choices of BCs
  - Choice we have been using
    - Invalid nodes are set to min value (T= -1)
    - Nodes on wall, and nodes in non-overlap regions are set to max value (T= 1)
    - The rest of the boundaries are treated as adiabatic wall (T_n= 0)
  - Approximate distance function
    - Boundary nodes are set to T = - *distance_to_wall*
  - Other choices of BCs possible

Elliptic Hole Cutting

- Choices of source term
  - In favor of *cell-quality*

$$f = f_{\min} + \frac{c - c_{\min}}{c_{\max} - c_{\min}} \left(f_{\max} - f_{\min} \right)$$

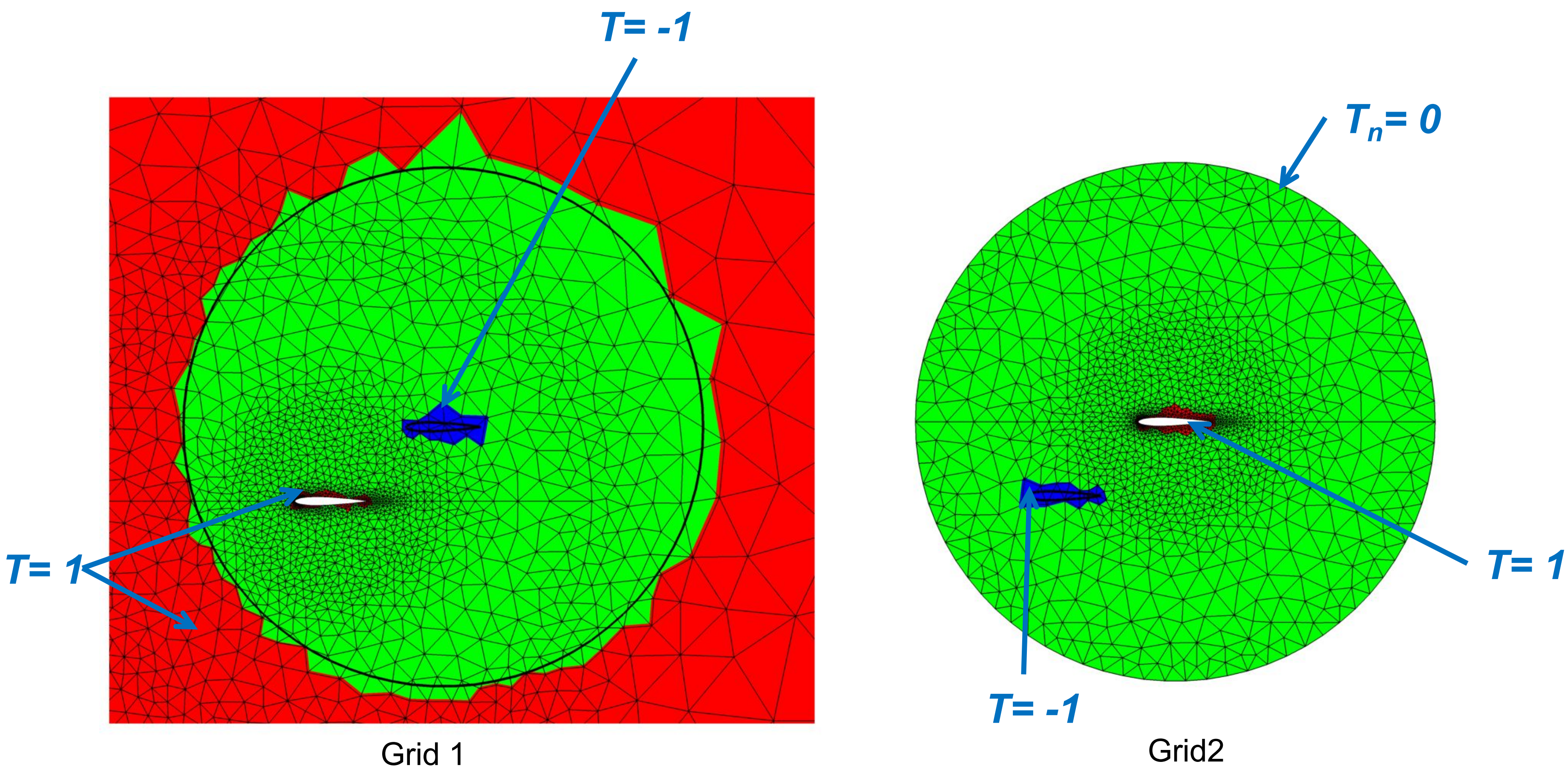
where *c* is *cell-quality*

  - In favor of specific grids

$$f = \begin{cases} f_{\max} & \text{for prefered grids} \\ f_{\min} & \text{for other grids} \end{cases}$$

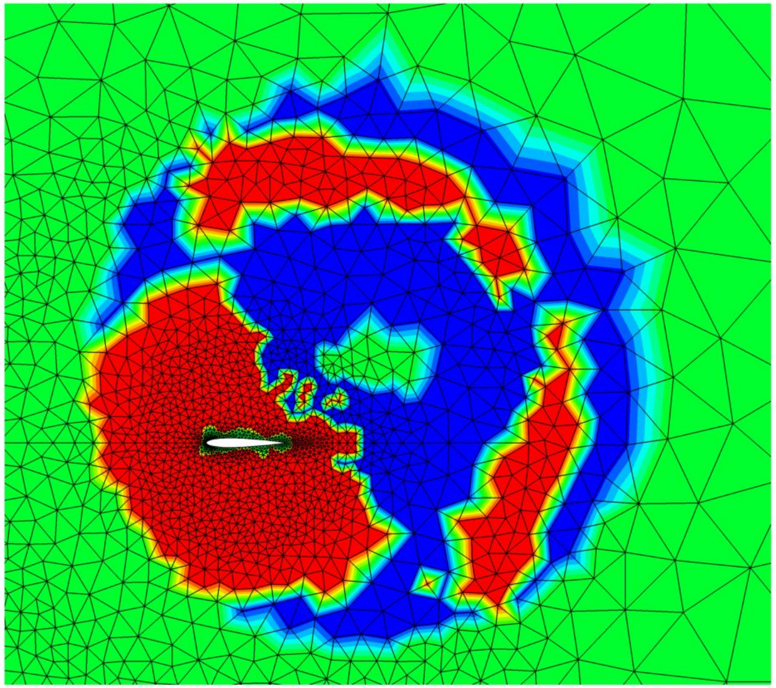
  - Other choices of source term possible

Elliptic Hole Cutting

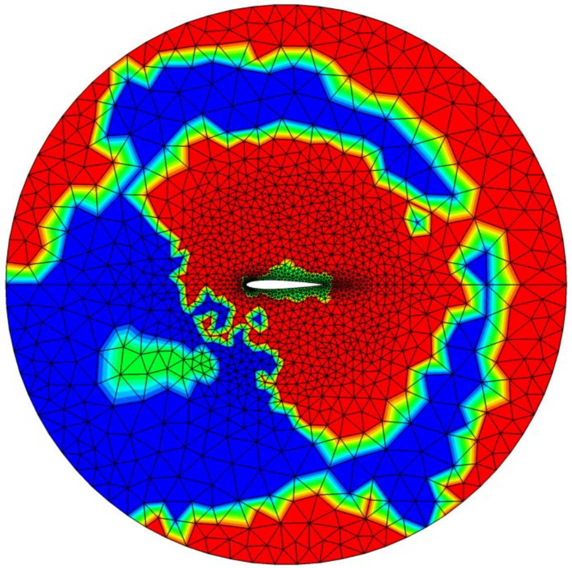


Boundary conditions for Poisson equations on each grid

Elliptic Hole Cutting



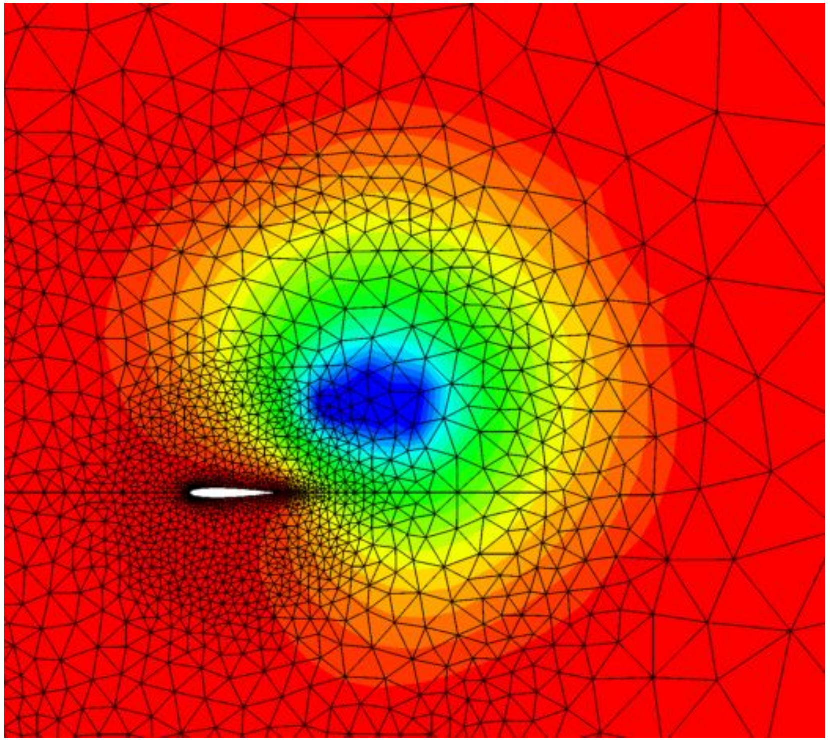
Grid 1



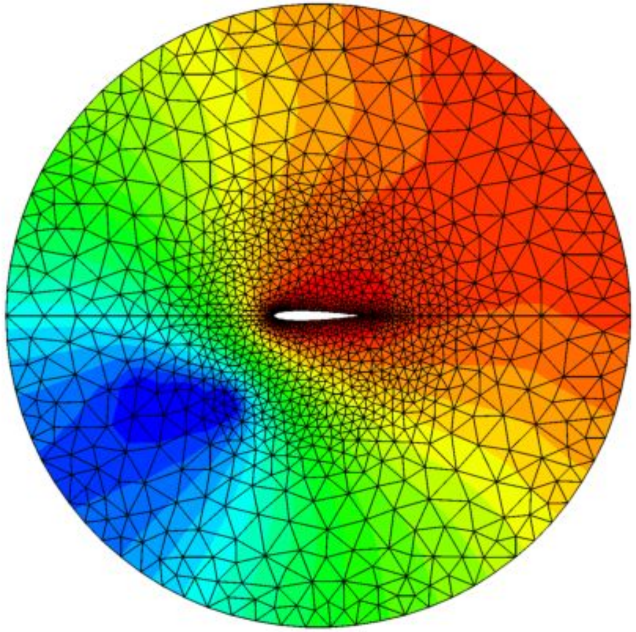
Grid2

Source term for the Poisson problems in favor of cell-quality

Elliptic Hole Cutting



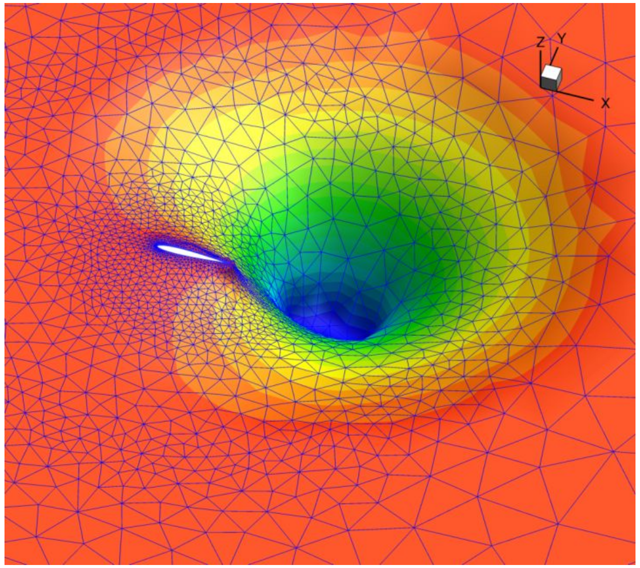
Grid 1



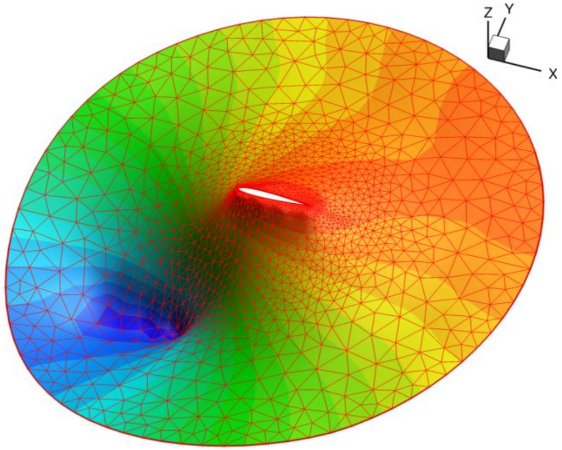
Grid 2

Solution of Poisson problems

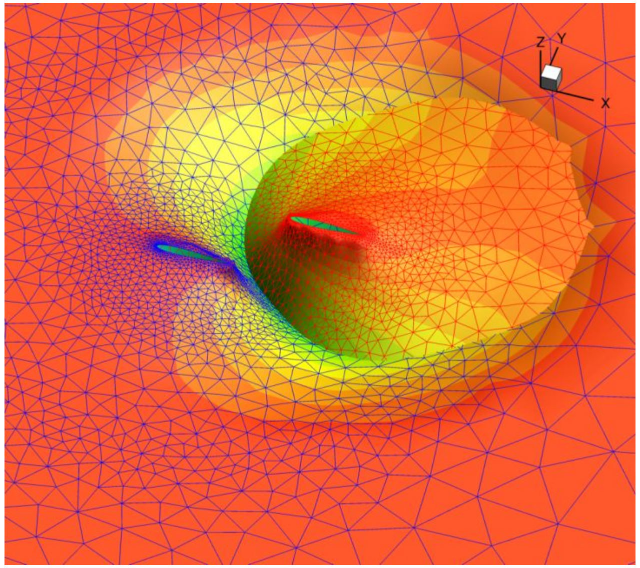
Elliptic Hole Cutting



Grid 1



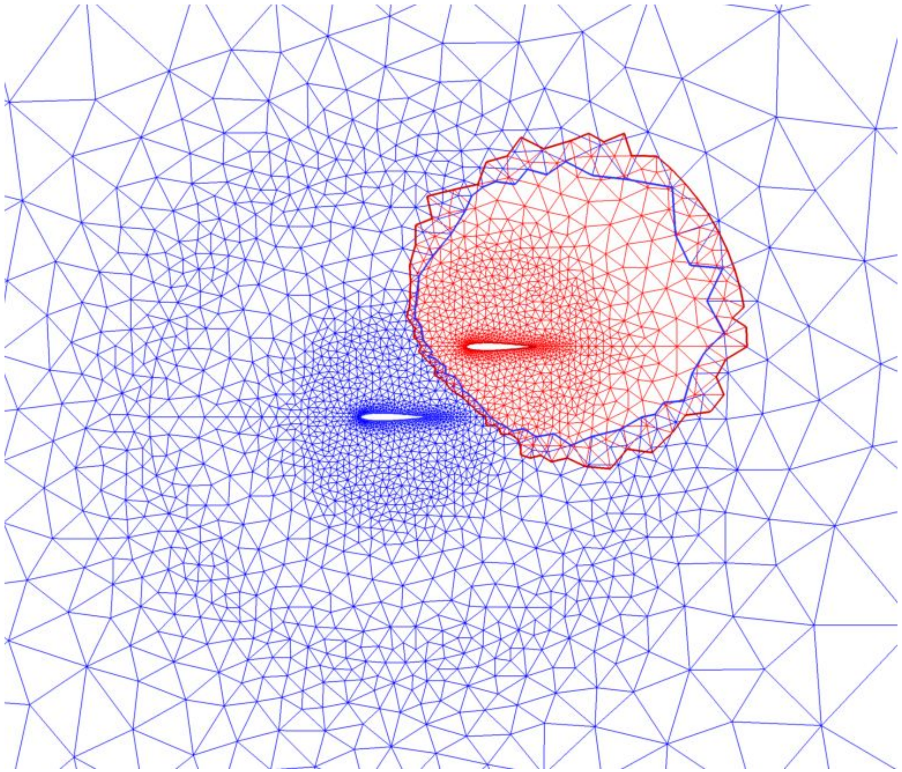
Grid 2



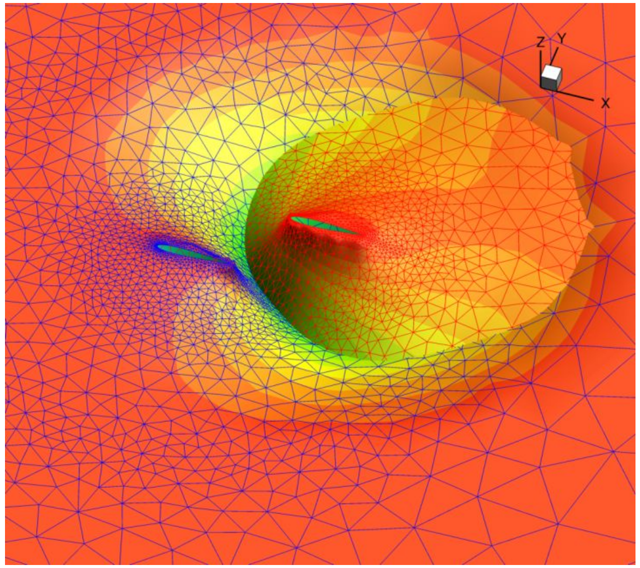
Grid 1 and 2

3D view of Poisson solution

Elliptic Hole Cutting

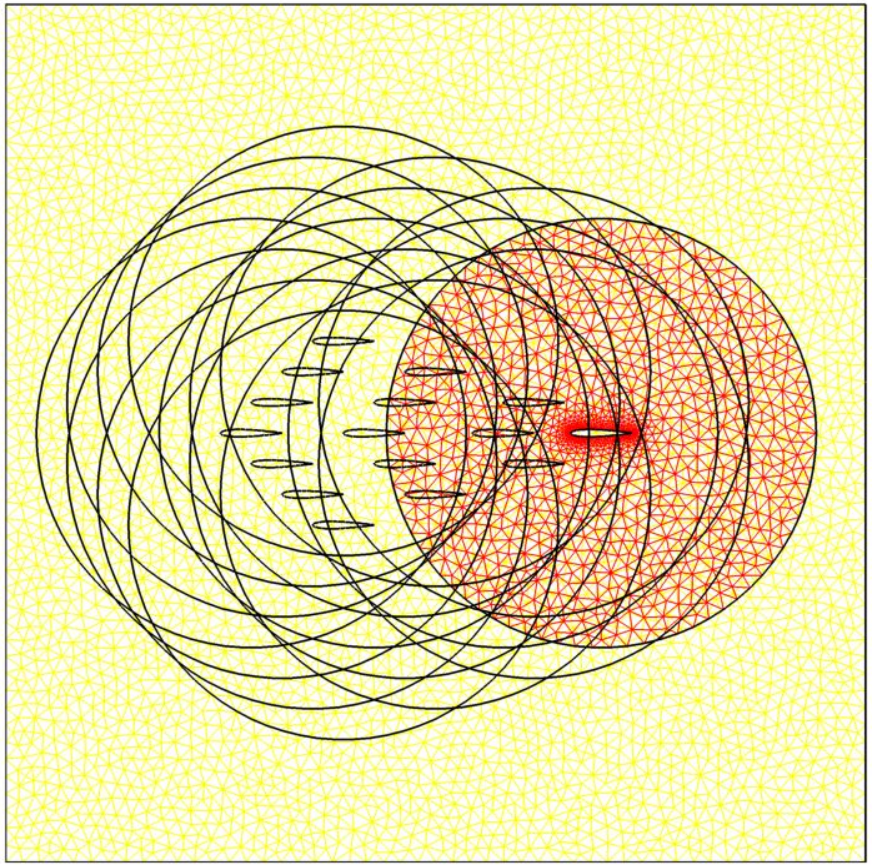


Final mesh



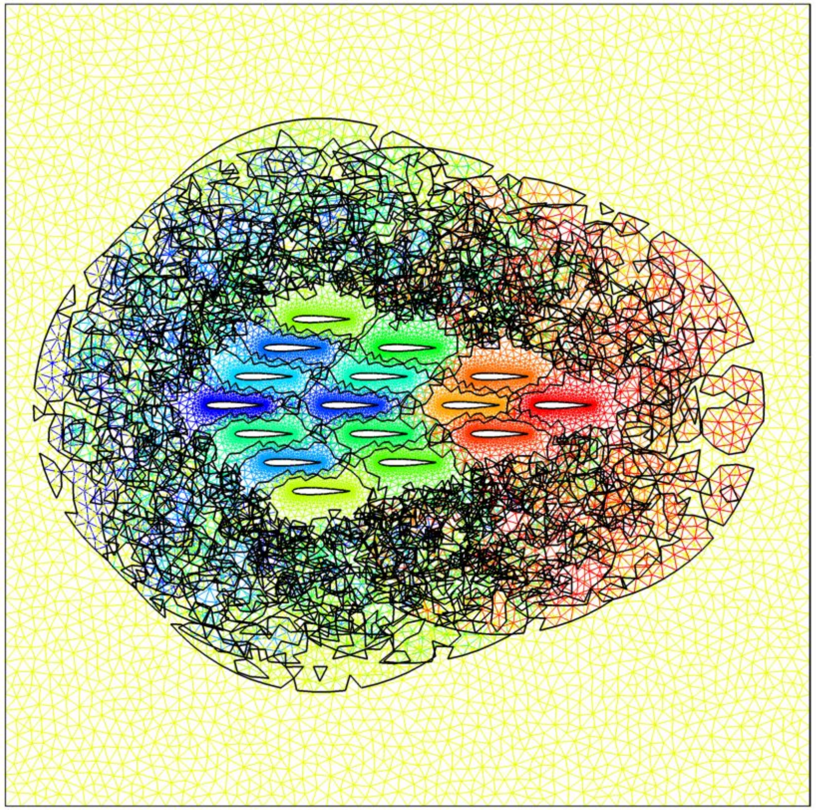
3D view of Poisson solution

Comparison of Hole Cutting



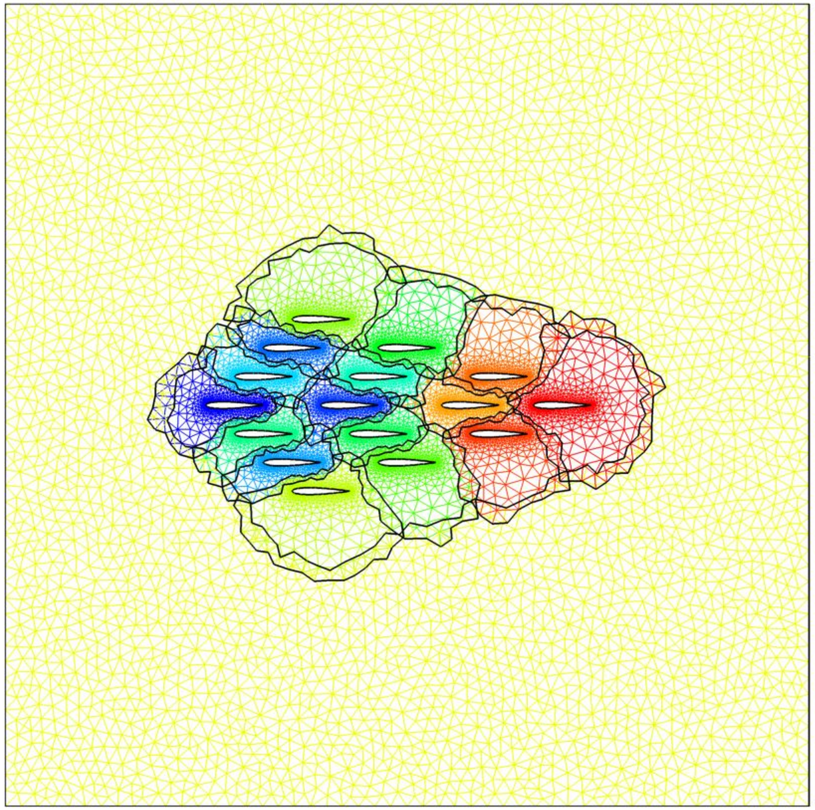
16 airfoil-grids overlapping on a background grid

Comparison of Hole Cutting

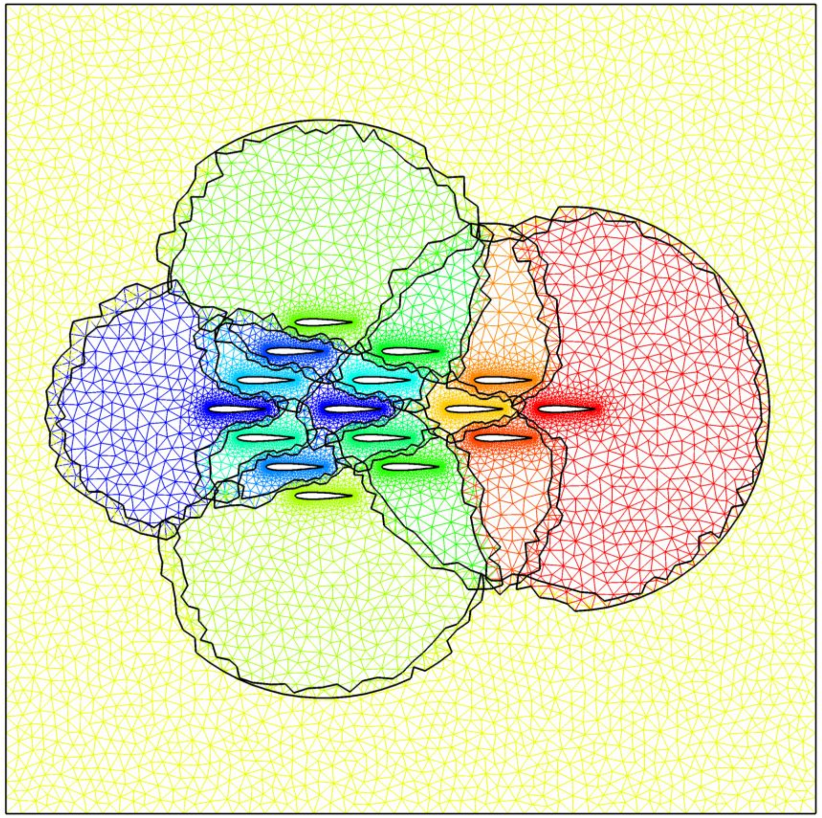


Implicit Hole Cutting

Comparison of Hole Cutting



In favor of cell quality

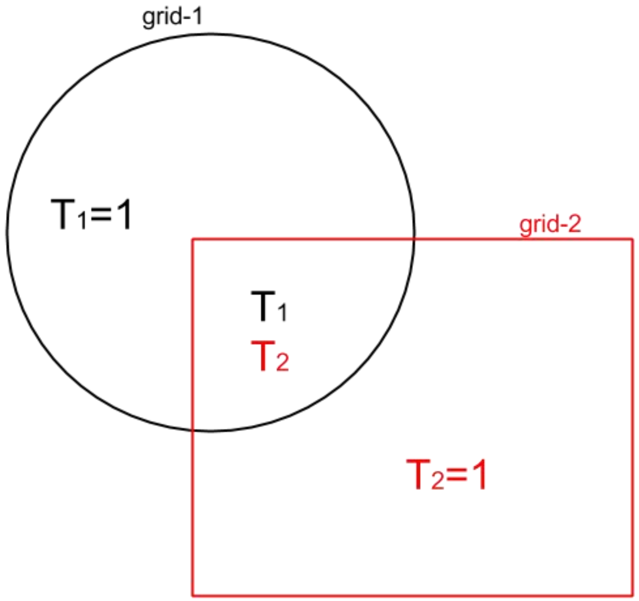


In favor of airfoil grids

Elliptic Hole Cutting using different source terms

Consideration for Parallel

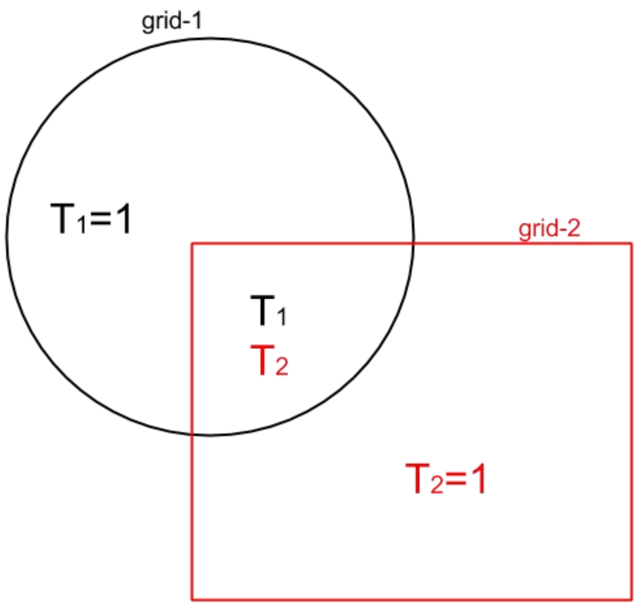
- T_1 and T_2 need to be compared at the same location (at same node from same grid)
- If we want to compare T_1 and T_2 on grid-1, we can:
  - Interpolate T_2 from grid-2 to grid-1; or
  - Solve for T_2 on grid-1

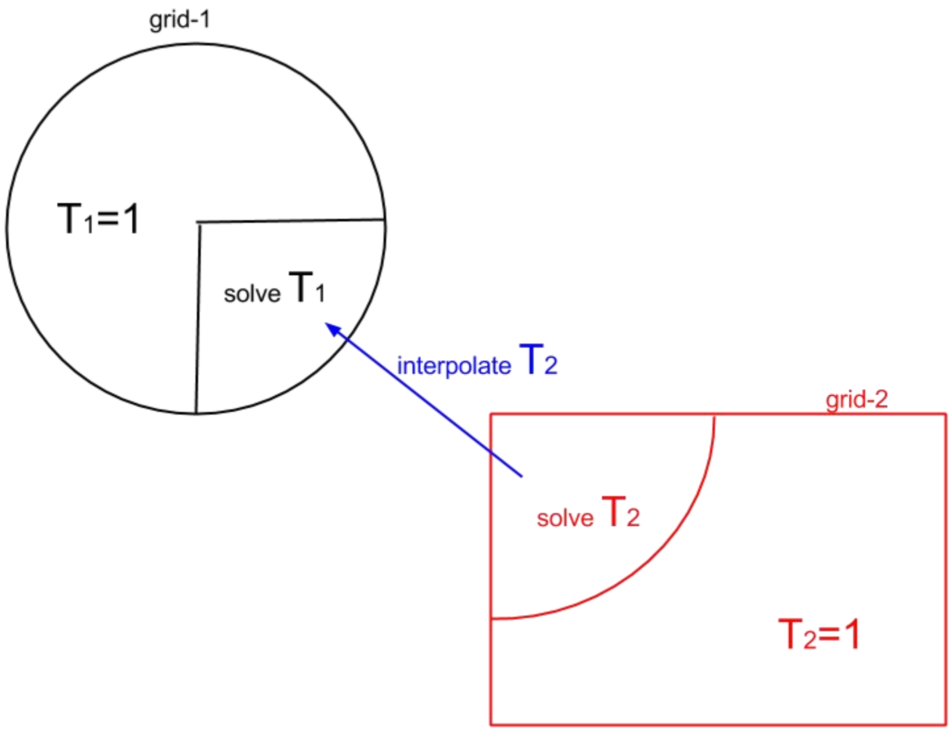


Consideration for Parallel

Interpolate T_2 from grid-2 to grid-1

- Every node of grid-1 in the overlapped region needs to be searched on grid-2
- Lots of communication in parallel implementation

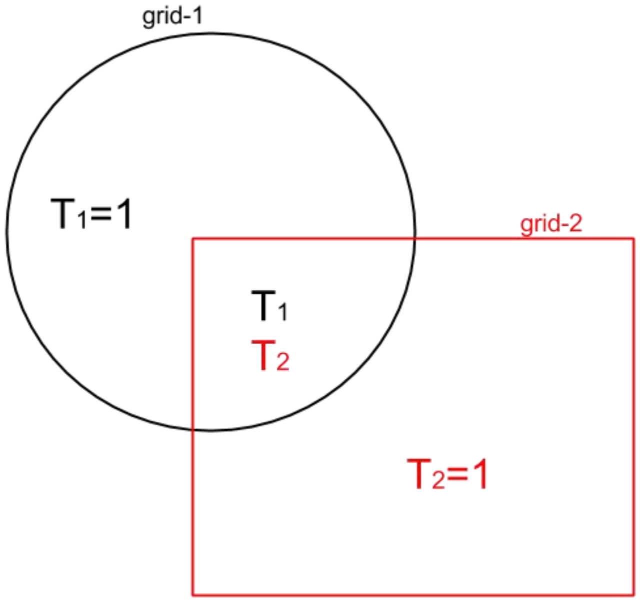


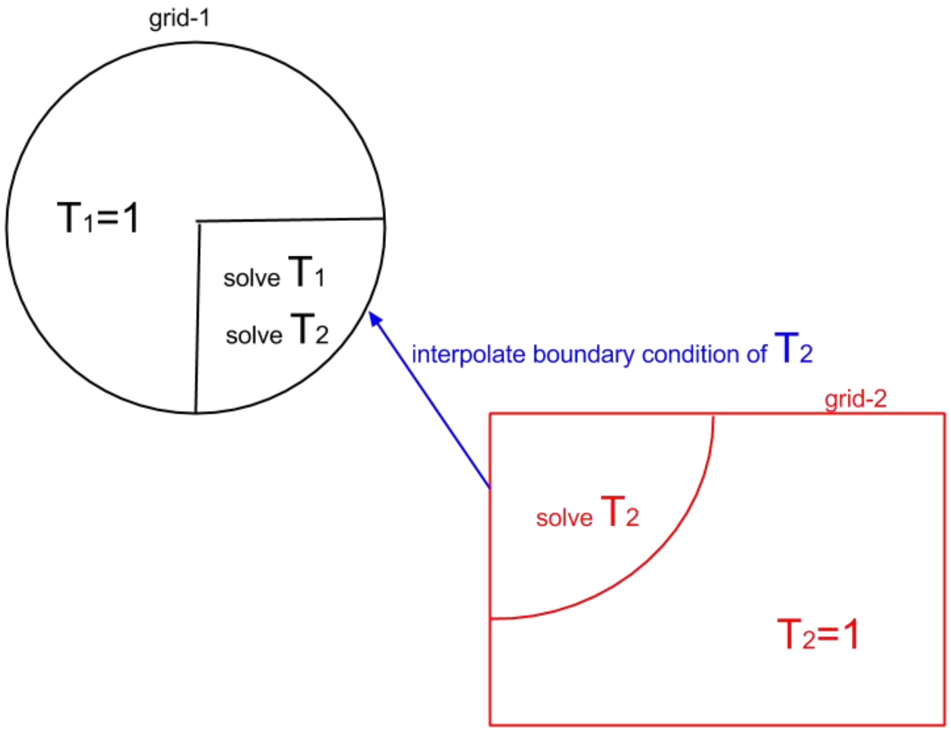


Consideration for Parallel

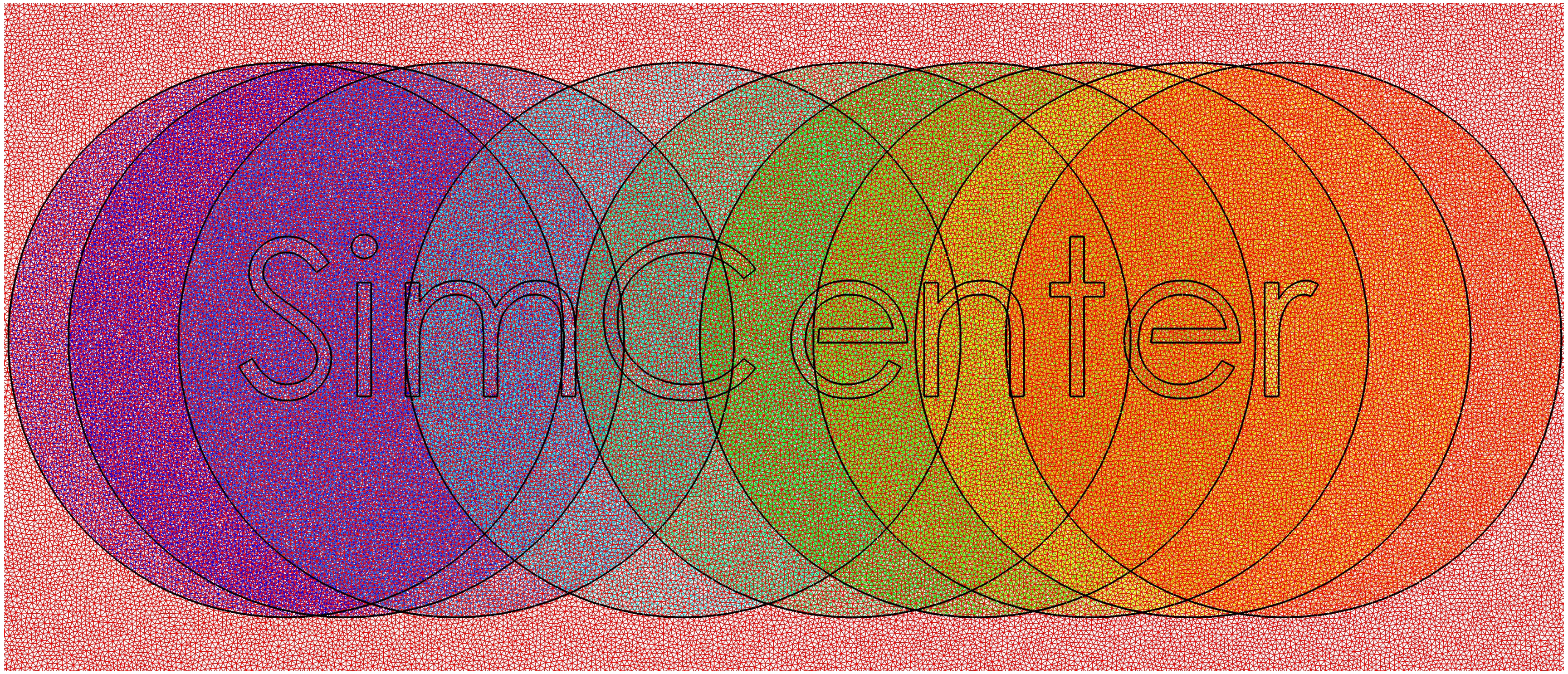
Solve for T_2 on grid-1

- Only the nodes of grid-1 that are on boundaries of the overlapped region need to be searched on grid-2
- Much less communication in parallel implementation, ideal for parallel implementation



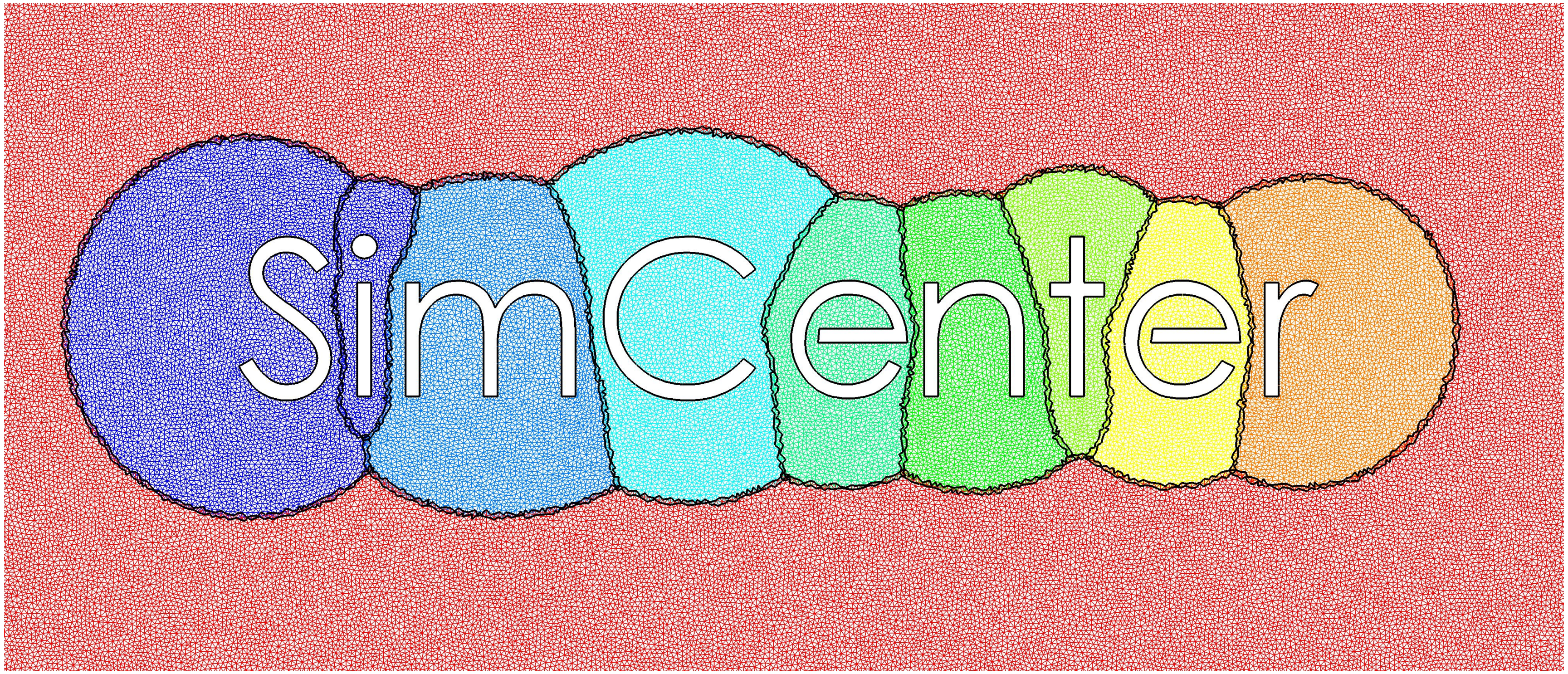


Elliptic Hole Cutting



“SimCenter” grids before hole cutting

Elliptic Hole Cutting



"SimCenter" grids after elliptic hole cutting

Advantages of Elliptic Hole Cutting

- Mesh quality:
  - The "continuity" of cell selection is guaranteed by the smoothness of the Poisson solutions
- Automation:
  - User input is not necessary
  - Yet, user still have the freedom to influence cell selection process indirectly (through source terms, or boundary conditions) or directly (by modifying Poisson solution)
- Parallel:
  - Poisson solver can easily be parallel
  - Limited searching keeps communication cost down
- Flexibility
  - Approximated distance funtion
  - other choices possible

Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
- Adaptive overset
- Conclusion

Governing Equations

- Weighted intergral form of compressible Navier-Stokes equations with Spalart-Allmaras turbulence model

$$\int_{\Omega} \varphi \left[\frac{\partial Q}{\partial t} + \nabla \cdot \left(\overline{\mathbf{F}}_e(Q) - \mathbf{F}_v(Q, \nabla Q) \right) - S(Q, \nabla Q) \right] d\Omega = 0$$

- Convective flux on dynamic grids

$$\overline{\mathbf{F}}_e = \mathbf{F}_e - \mathbf{V}_g Q$$

- SUPG used in defining weighting function

$$\varphi = [N] + [P]$$

- Utilizing integration by parts the weak form becomes

$$\frac{\partial}{\partial t}\int_{\Omega} N \mathbf{Q} d\Omega - \int_{\Omega} \nabla N \cdot \left(\overline{\mathbf{F}}_e - \mathbf{F}_v \right) d\Omega + \underline{\oint_{\Gamma} N \left(\overline{\mathbf{F}}_e - \mathbf{F}_v \right) \cdot \mathbf{n} \, d\Gamma}$$

Boundary terms

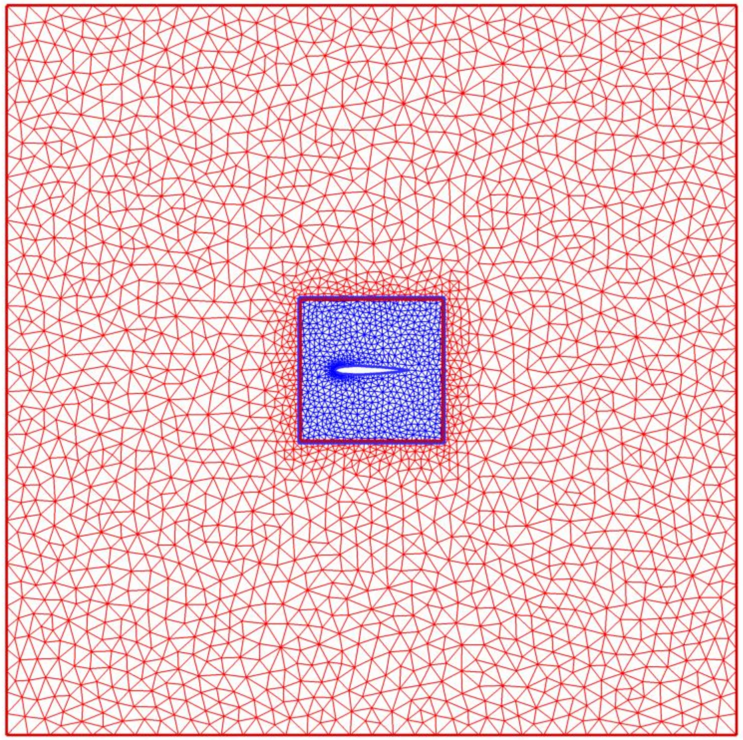
$$- \int_{\Omega} N S d\Omega + \frac{\partial}{\partial t}\int_{\Omega} [P] \mathbf{Q} d\Omega + \int_{\Omega} [P] \left(\nabla \cdot \left(\overline{\mathbf{F}}_e - \mathbf{F}_v \right) - S \right) d\Omega = 0$$

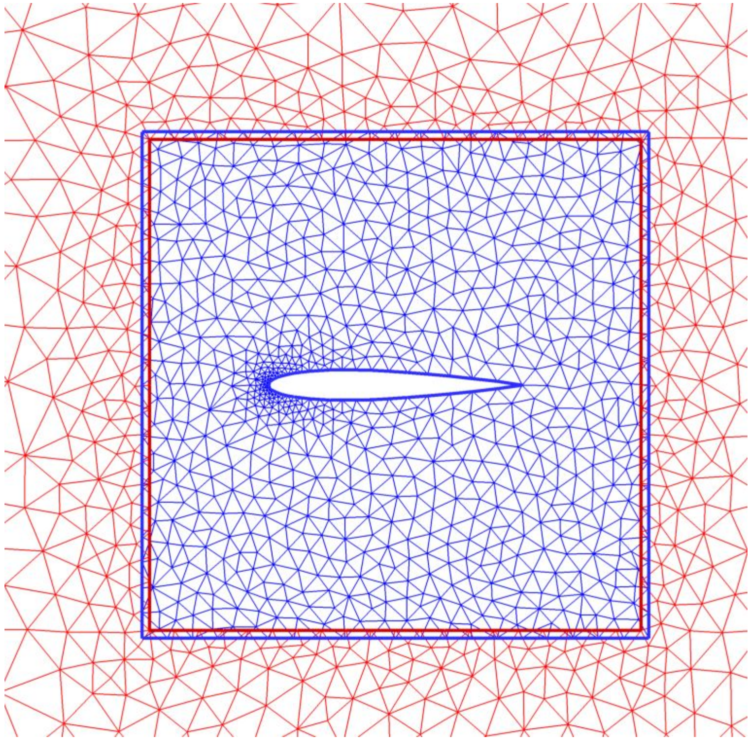
Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
- Adaptive overset
- Conclusion

Overset Methodology

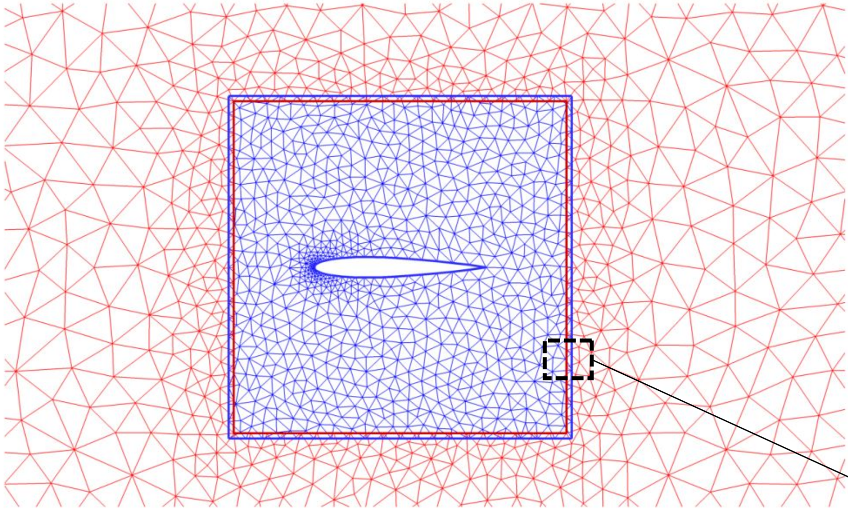
- Overset problems appear as boundary conditions



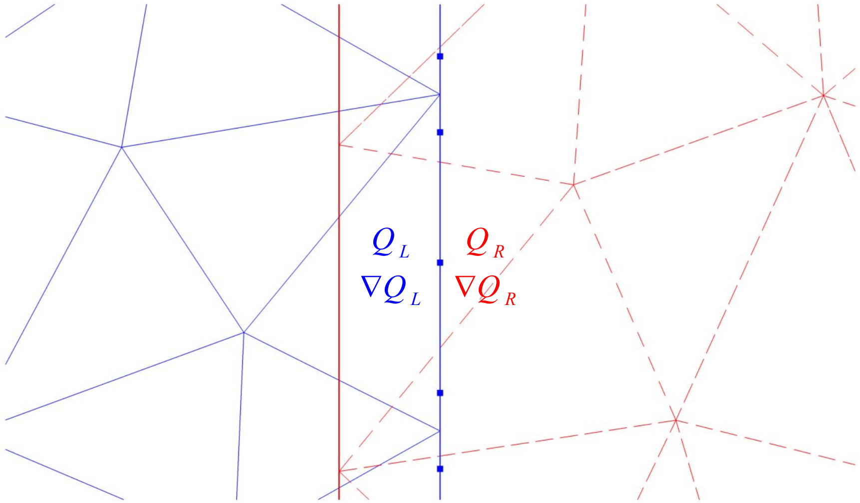


Example of overset problem of an airfoil

Discretization



$Q_L, \nabla Q_L$ are obtained locally
$Q_R, \nabla Q_R$ are interpolated from donor cell



Convective flux viewed as Riemann problem

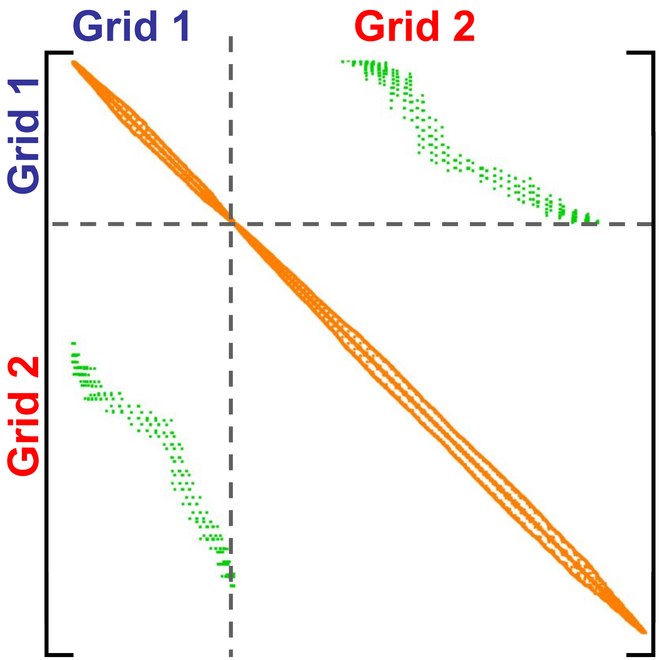
$$\overline{\mathbf{F}}_e \cdot \mathbf{n} = \overline{\mathbf{F}}_e^{+}\left(\mathbf{Q}_L\right)\cdot\mathbf{n} + \overline{\mathbf{F}}_e^{-}\left(\mathbf{Q}_R\right)\cdot\mathbf{n} \quad \text{van Leer flux}$$

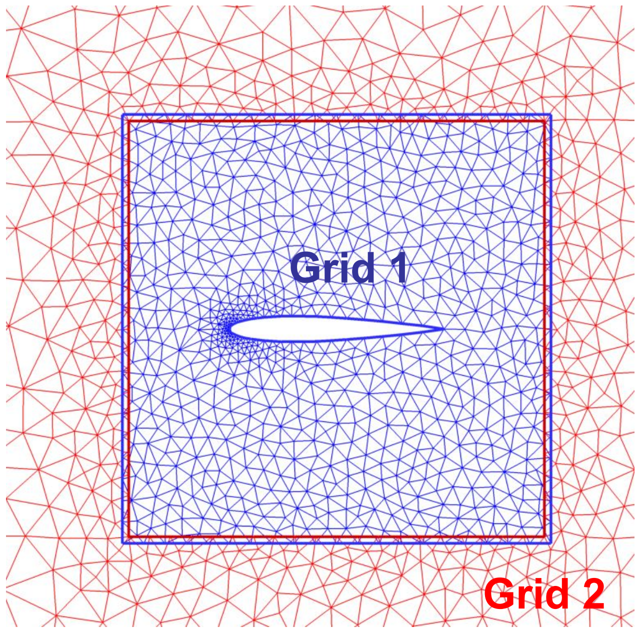
$$\mathbf{F}_v \cdot \mathbf{n} = \frac{1}{2}\left(\mathbf{F}_v\left(\mathbf{Q}_L, \nabla\mathbf{Q}_L\right)\cdot\mathbf{n} + \mathbf{F}_v\left(\mathbf{Q}_R, \nabla\mathbf{Q}_R\right)\cdot\mathbf{n}\right)$$

Linearization

- Jacobian matrix has two components stored separately
  - $\tilde{A}$ Intra-grid dependency, its structure does not change
  - O Inter-grid dependency, its structure changes with dynamic grids

$$A = \tilde{A} + O =$$





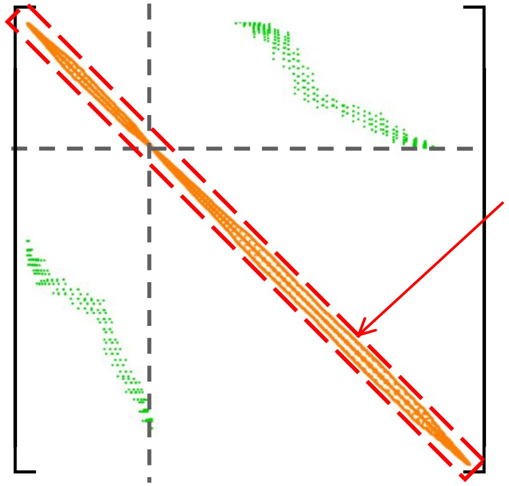
Solution Procedure

- Discrete-Newton relaxation to converge time-residual
  - Both intra-grid and inter-grid dependency are used, resulting in an implicit treatment of the overset boundaries
- GMRES with ILU(k) preconditioning to solve linear system
  - Preconditioner is modified for overset problems for improved convergence of GMRES

Original GMRES Preconditioner

- Jacobian matrix has large bandwidth due to O
  - Reordering would be expensive: not practical for parallel implementation

$$A = \tilde{A} + O = \begin{bmatrix} \tilde{A}_1 & O_{12} \\ O_{21} & \tilde{A}_2 \end{bmatrix} =$$



ILU(k) Decomposition inside here

$\tilde{A} \approx LU$

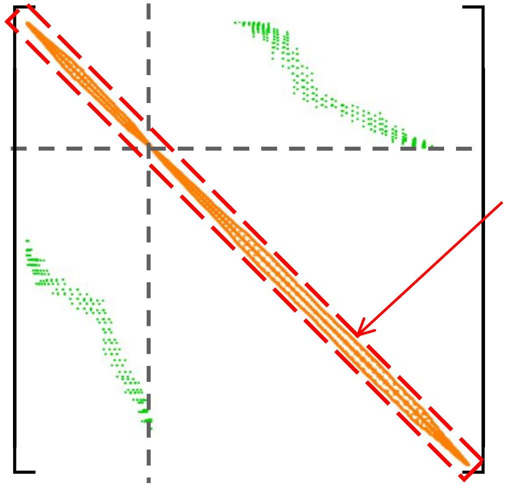
- ILU(k) only considers entries within a specified bandwidth (intra-grid): preconditioner may not be satisfactory

$$A_{pre} = LU = \begin{bmatrix} L_1 & \\ & L_2 \end{bmatrix} \begin{bmatrix} U_1 & \\ & U_2 \end{bmatrix} \approx \begin{bmatrix} \tilde{A}_1 & \\ & \tilde{A}_2 \end{bmatrix}$$

O is completely ignored

Modified GMRES Preconditioner

$$A = \tilde{A} + O = \begin{bmatrix} \tilde{A}_1 & O_{12} \\ O_{21} & \tilde{A}_2 \end{bmatrix} =$$



ILU(k) Decomposition inside here

$\tilde{A} \approx LU$

A modification for overset grids may be implemented as

$$A_{pre} = L\left(U + L^{-1}O_U\right)$$

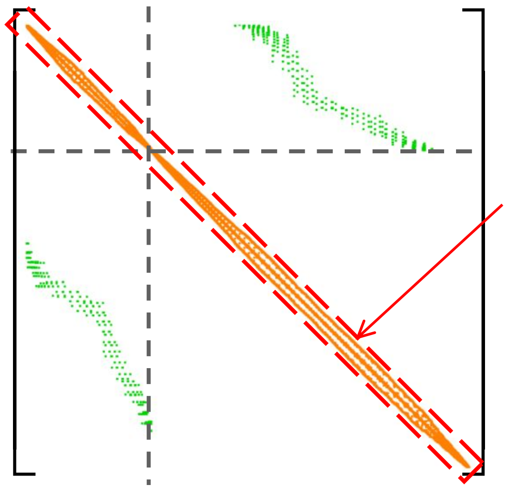
$$= \begin{bmatrix} L_1 & \\ & L_2 \end{bmatrix} \begin{bmatrix} U_1 & L_1^{-1}O_{12} \\ & U_2 \end{bmatrix} \approx \begin{bmatrix} \tilde{A}_1 & O_{12} \\ & \tilde{A}_2 \end{bmatrix}$$

LU-decomposition of A_{pre} readily known implicitly

O_{12} is considered

Modified GMRES Preconditioner

$$A = \tilde{A} + O = \begin{bmatrix} \tilde{A}_1 & O_{12} \\ O_{21} & \tilde{A}_2 \end{bmatrix} =$$



ILU(k) Decomposition inside here

$$\tilde{A} \approx LU$$

Another modification

$$A_{pre} = \left(L + O_L U^{-1}\right)\left(U + L^{-1} O_U\right)$$

LU-decomposition of A_{pre} readily known implicitly

$$= \begin{bmatrix} L_1 & \\ O_{21} U_1^{-1} & L_2 \end{bmatrix} \begin{bmatrix} U_1 & L_1^{-1} O_{12} \\ & U_2 \end{bmatrix} \approx \begin{bmatrix} \tilde{A}_1 & O_{12} \\ O_{21} & \tilde{A}_2 + \underline{O_{21} \tilde{A}_2^{-1} O_{12}} \end{bmatrix}$$

O_{12}, O_{21} are considered, but extra term is added

Outline

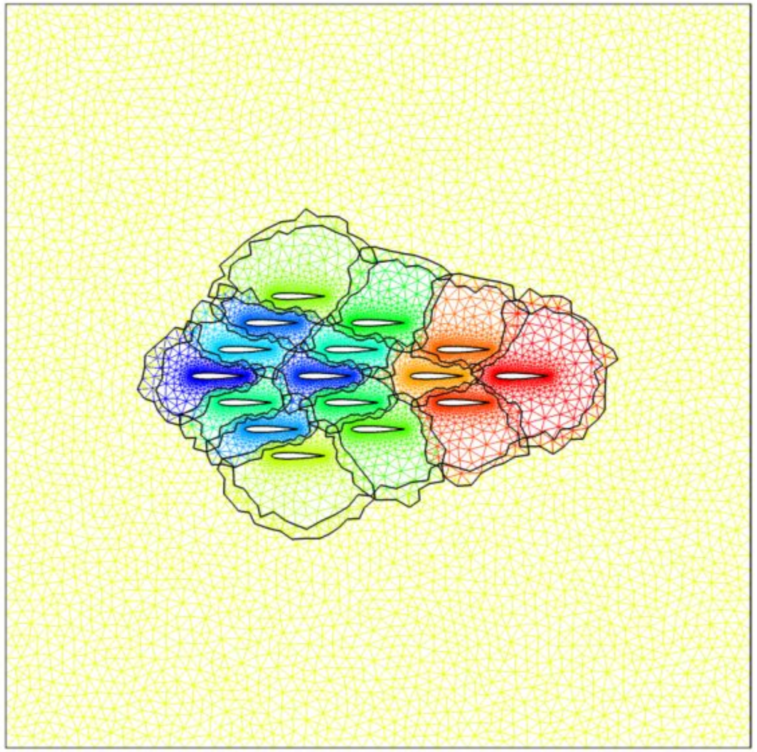
- Hole cutting
- Governing equations
- Overset methodology
- Overset results
  - Modified preconditioner
  - Manufactured solutions
  - Steady turbulent
  - Unsteady moving boundary
  - Relative motion between two bodies
- Adaptive overset
- Conclusion

Modified GMRES Preconditioners

Steady inviscid flow, P1 elements
Free stream condition $M = 0.2, \alpha = 2^\circ$

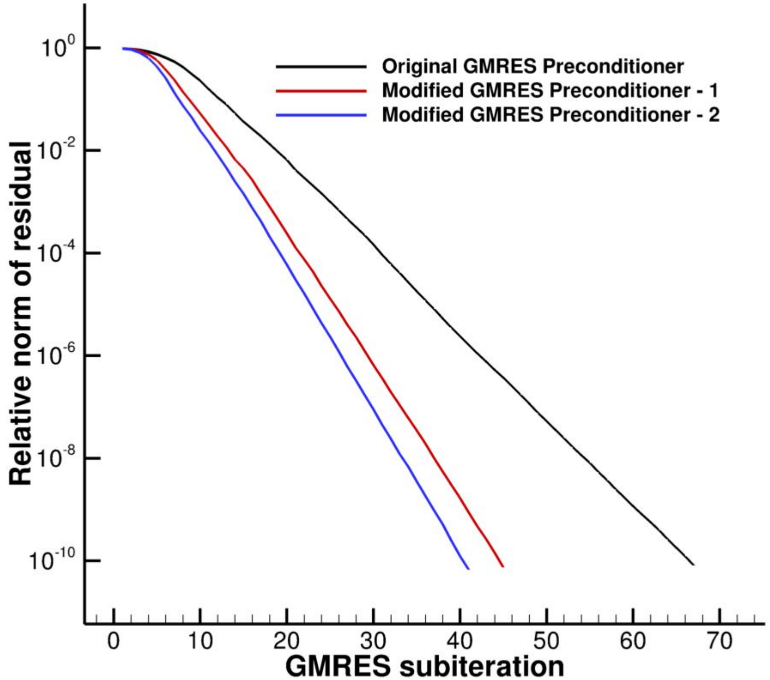
CFL=100

One discrete-Newton step performed



Mesh used for comparison

Comparison of GMRES Preconditioners



Convergence history of GMRES subiteration
using different preconditioner

1St modified version is used in current study

Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
  - Modified preconditioner
  - Manufactured solutions
  - Steady turbulent
  - Unsteady moving boundary
  - Relative motion between two bodies
- Adaptive overset
- Conclusion

Manufactured Solutions

- The Method of Manufactured Solution (MMS) is a general procedure for generating nontrivial exact solutions to PDEs

- Accuracy of the SUPG overset scheme is assessed using MMS based on a comprehensive set of guidelines

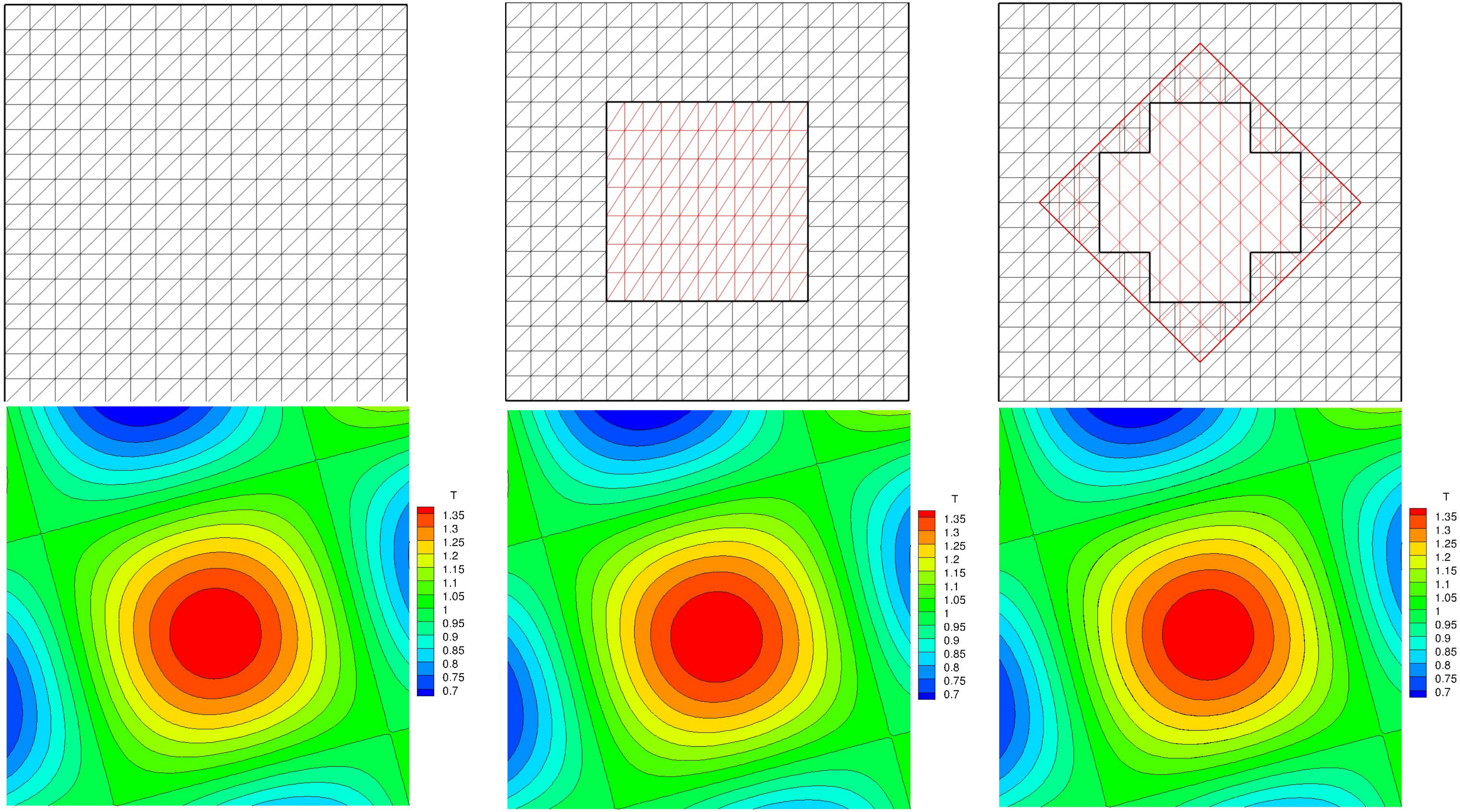
Manufactured Solutions

- MMS for both inviscid and laminar (Re=100) equations are performed to assess accuracy
- The following trigonometric functions are used to derive forcing functions and boundary conditions

$$\rho = \rho_o \{1 + 0.2\cos[\pi(c_1 x - s_1 y)] + 0.2\cos[\pi(c_1 x + s_1 y)]\}$$
$$u = u_o \{1 + 0.2\cos[\pi(c_2 x - s_2 y + 0.1)] + 0.2\cos[\pi(c_2 x + s_2 y + 0.1)]\}$$
$$v = v_o \{1 + 0.2\cos[\pi(c_3 x - s_3 y - 0.1)] + 0.2\cos[\pi(c_3 x + s_3 y + 0.1)]\}$$
$$T = T_o \{1 + 0.2\cos[\pi(c_4 x - s_4 y - 0.1)] + 0.2\cos[\pi(c_4 x + s_4 y - 0.1)]\}$$

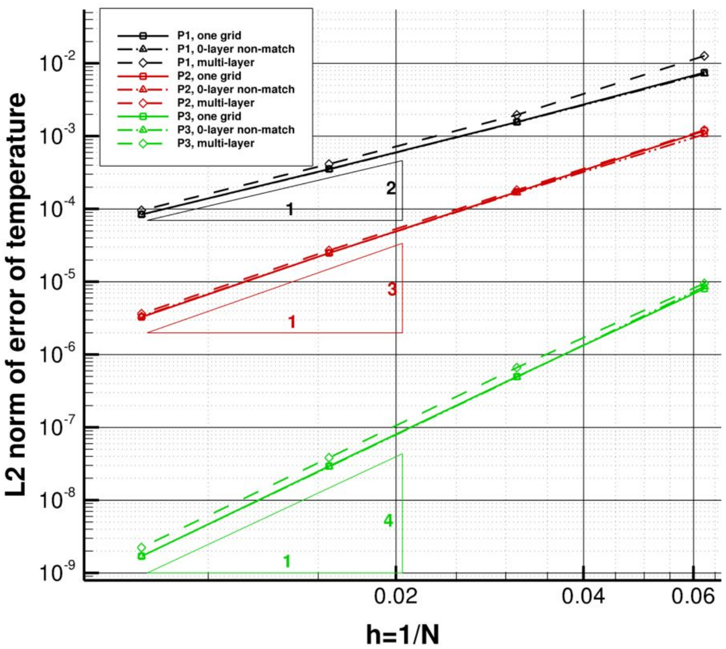
  - ρ_o, u_o, v_o, T_o correspond to the free stream condition of $M = 0.2, \alpha = 15^\circ$
  - c_i, s_i correspond to cosine and sine of 0°, 40°, 80°, and 120°

Manufactured Solutions

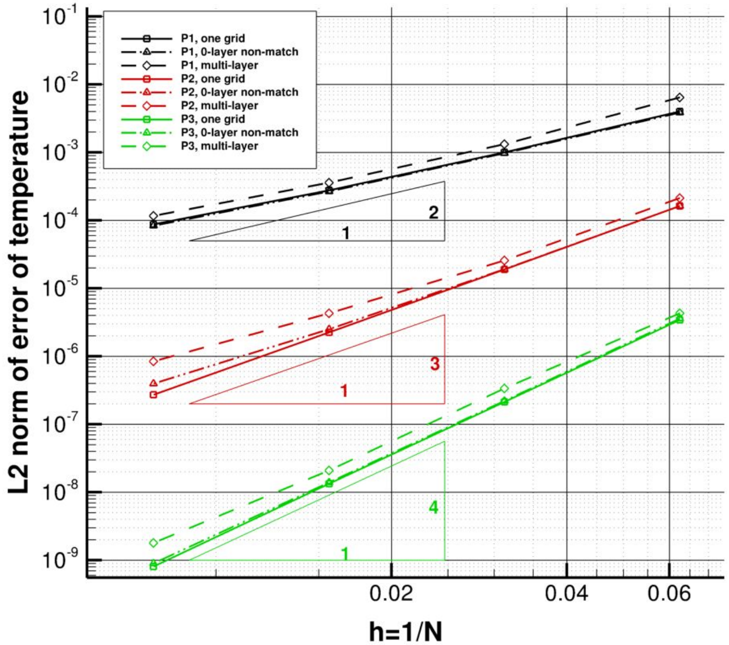


Temperature on coarsest meshes, laminar, P3 elements

Manufactured Solutions



Temperature, invisicd



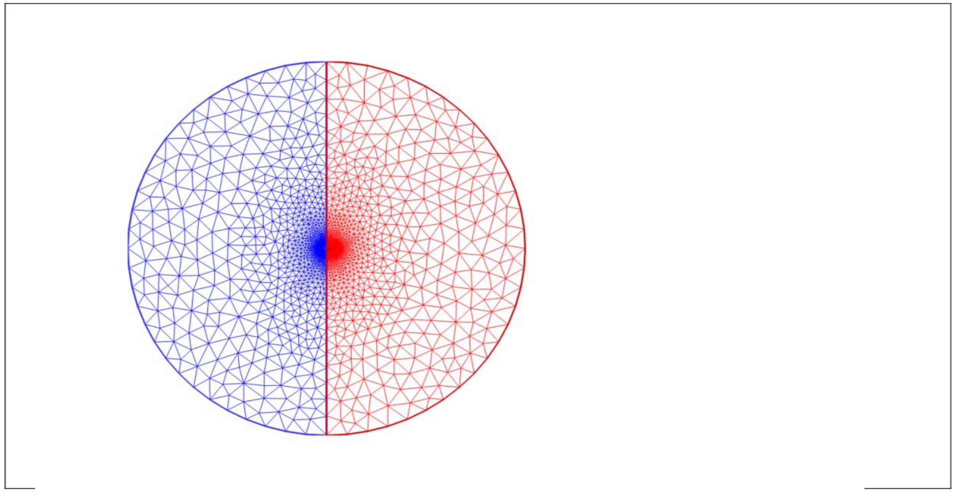
Temperature, laminar

Order of accuracy for inviscid and laminar flow

Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
  - Modified preconditioner
  - Manufactured solutions
  - Steady turbulent
  - Unsteady moving boundary
  - Relative motion between two bodies
- Adaptive overset
- Conclusion

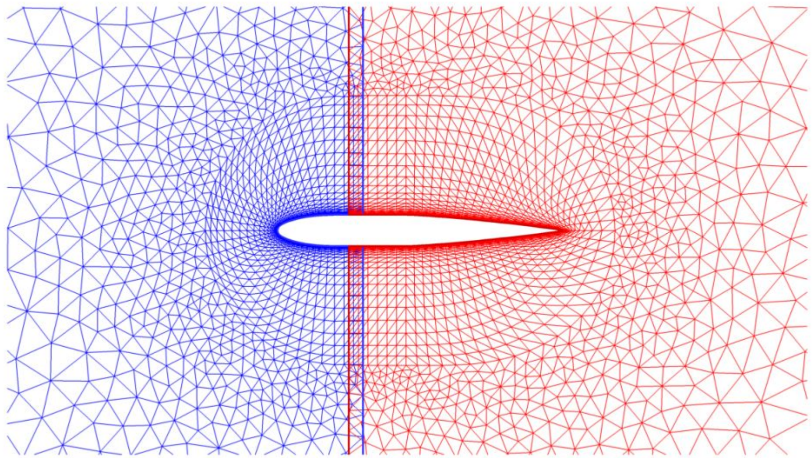
Steady Turbulent Flow



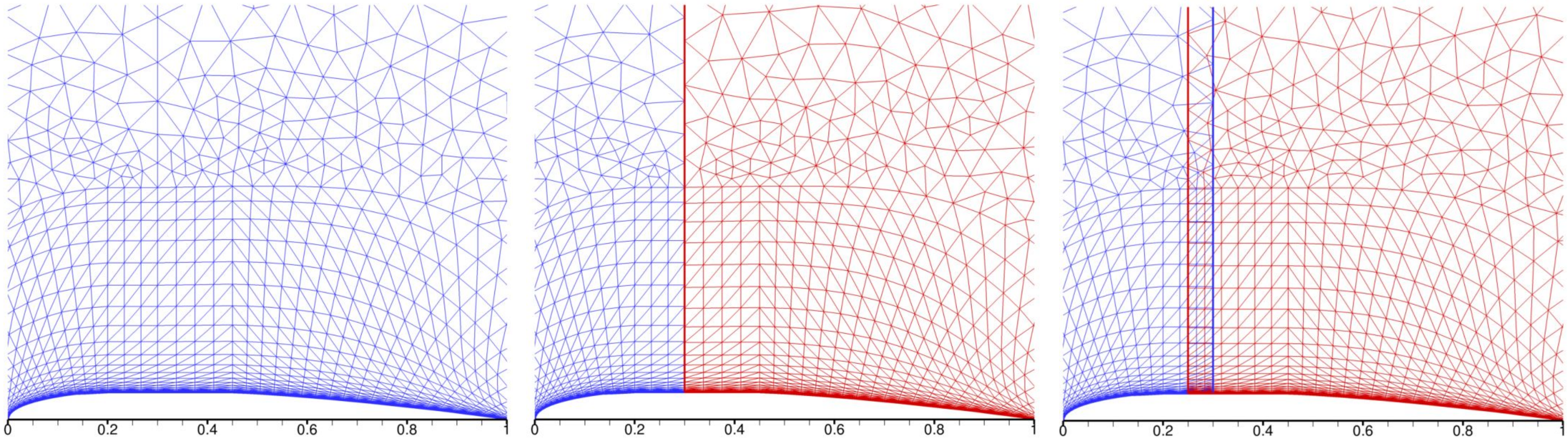
Free stream condition

$M_\infty = 0.2, \alpha_\infty = 2^\circ, \mathrm{Re} = 10^6$

Spalart-Allmaras turbulent model
y+ of wall spacing is 1



Steady Turbulent Flow



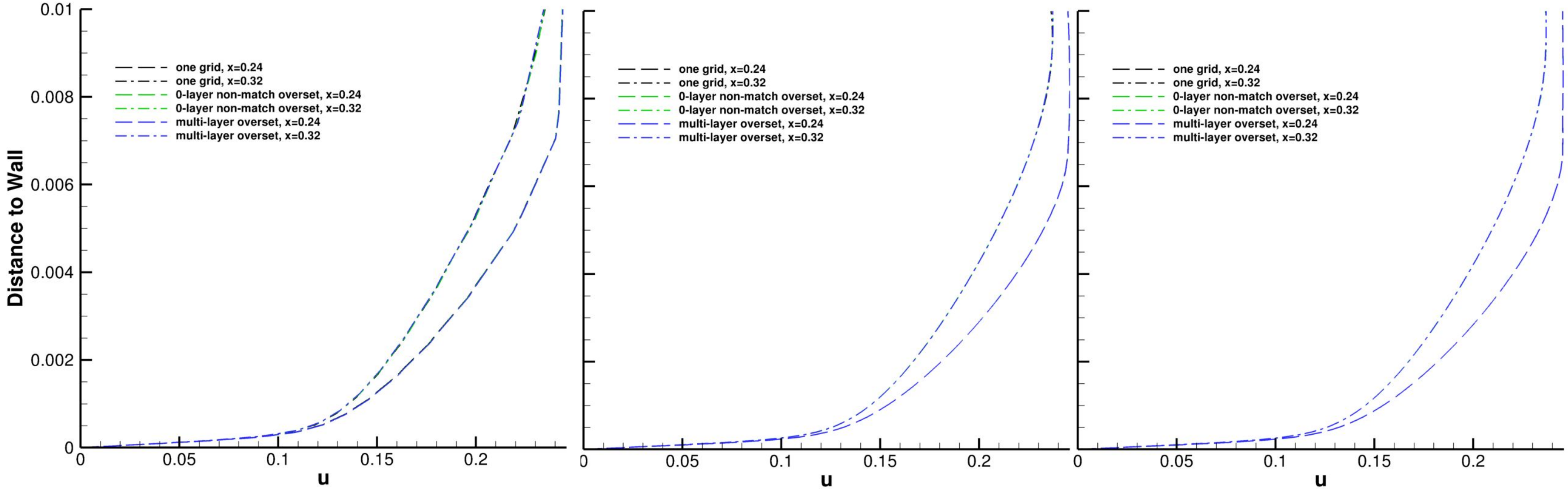
Single grid

Zero-layer non-matched overset grid

Multi-layer overlapping overset grid

Grids used in simulations

Steady Turbulent Flow



P1 elements

P2 elements

P3 elements

X-velocity profile at x=0.24 and 0.32

Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
  - Modified preconditioner
  - Manufactured solutions
  - Steady turbulent
  - Unsteady moving boundary
    - Sinusoidally oscillating airfoil
    - Sinusoidally pitching and plunging airfoil
  - Relative motion between two bodies
- Adaptive overset
- Conclusion

Sinusoidally Oscillating Airfoil

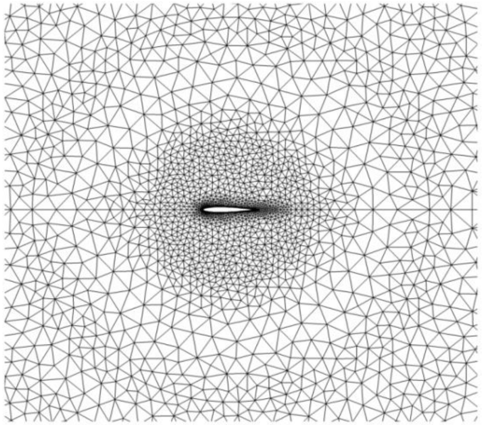
- Benchmark case for dynamic mesh code validation
- Free stream $M_\infty = 0.6, \alpha_\infty = 0^\circ$
- NACA0012 airfoil pitch about its quarter chord

$$\alpha(t) = \alpha_m + \alpha_o \sin(2kM_\infty t)$$

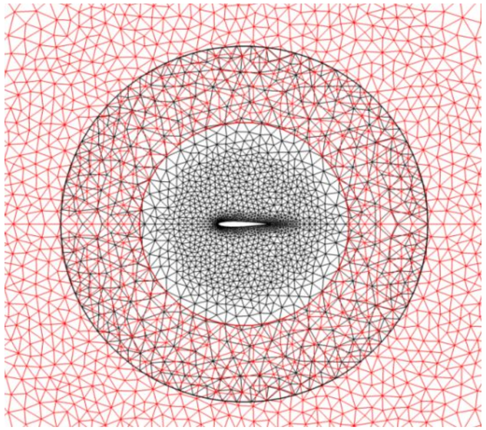
$$\text{where } \alpha_m = 2.89^\circ, \alpha_0 = 2.41^\circ, k = 0.0808$$

Sinusoidally Oscillating Airfoil

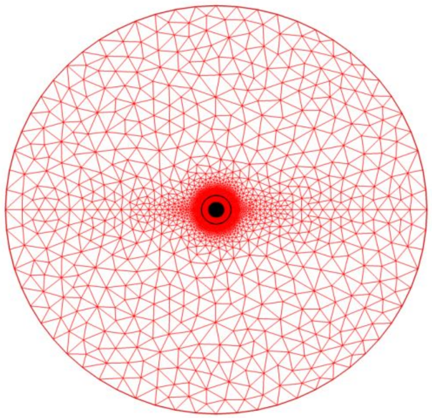
- Inviscid. P1 elements
- Multiple layers of overlap, grids generated a priori
- Grid moves as a rigid body. Analytical grid velocities are used
- For overset simulation, background grid is stationary, only airfoil grid is moving



Single grid

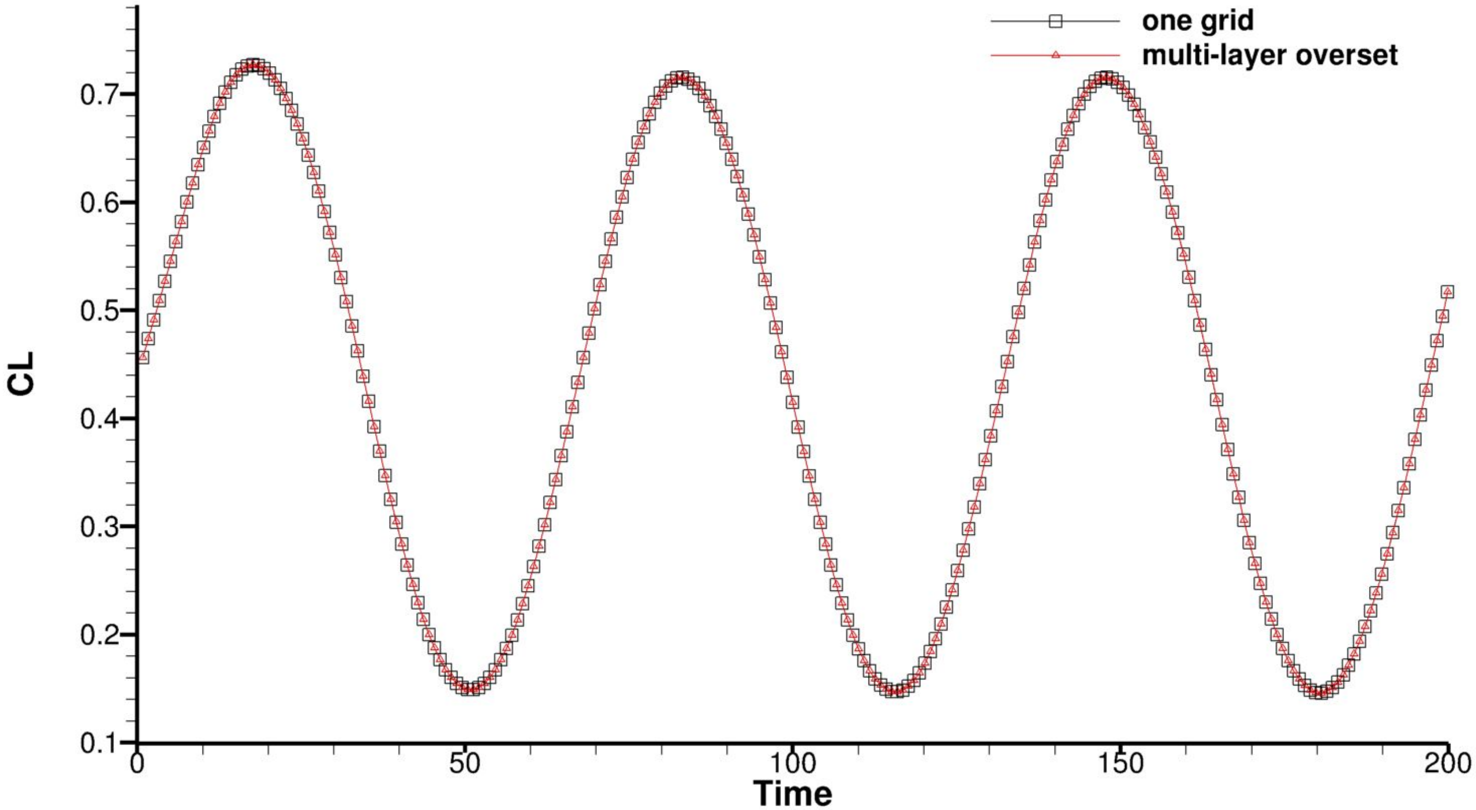


Overset grids



Global view of overset grids

Sinusoidally Oscillating Airfoil



Time history of coefficient of lift

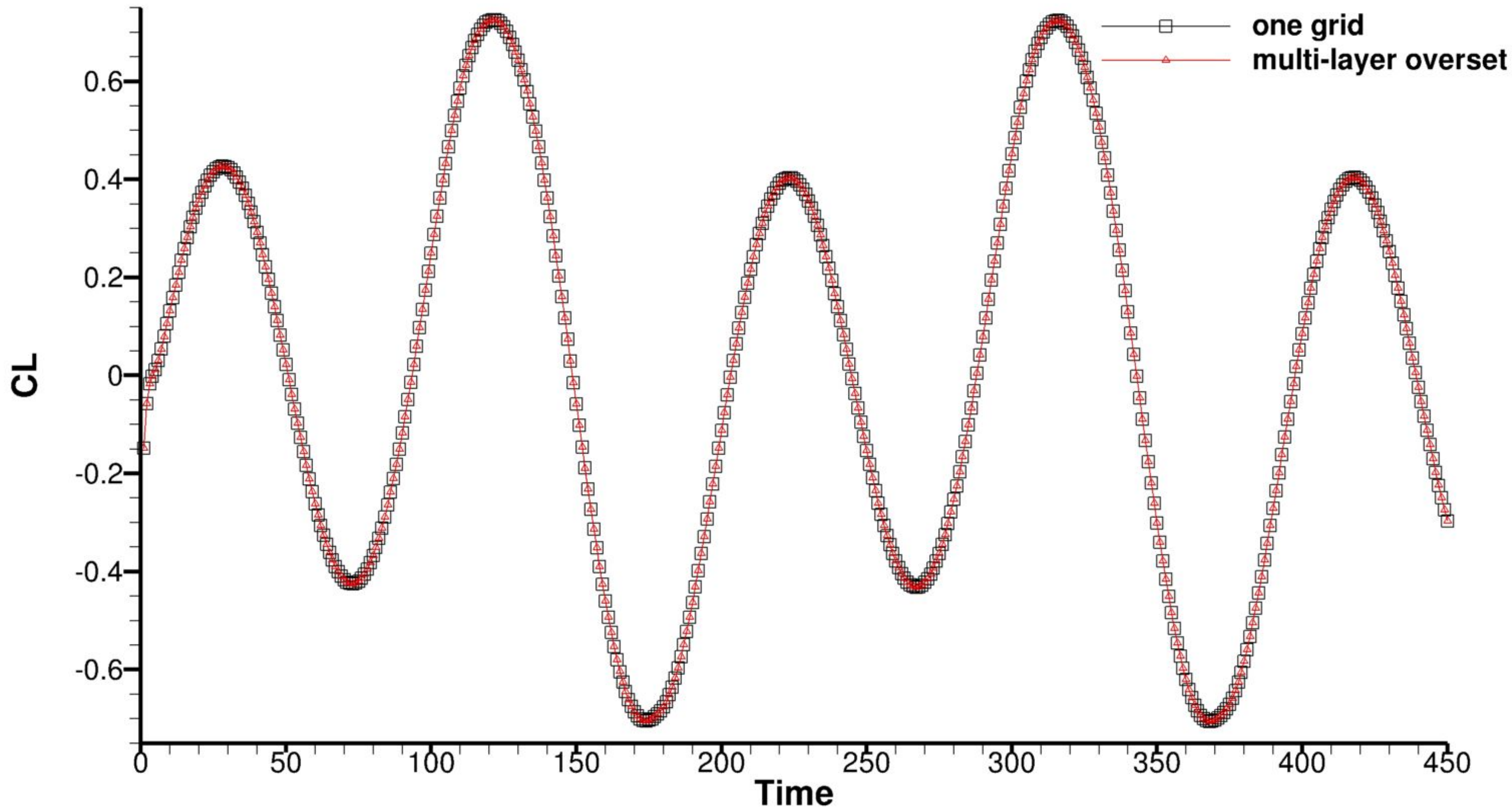
Sinusoidal Pitch and Plunge Airfoil

- Free stream $M_\infty = 0.4, \alpha_\infty = 0^\circ$
- NACA0012 Airfoil pitch about its quarter chord, and plunge

$$\begin{cases} \alpha(t) = & \alpha_m + \alpha_o \sin(2kM_\infty t) \\ h(t) = & h_0 \sin(kM_\infty t) \end{cases}$$

where $\alpha_m = 0^\circ, \alpha_0 = 5^\circ, k = 0.0808, h_0 = 0.4c,\quad c$ is the chord length

Sinusoidal Pitch and Plunge Airfoil



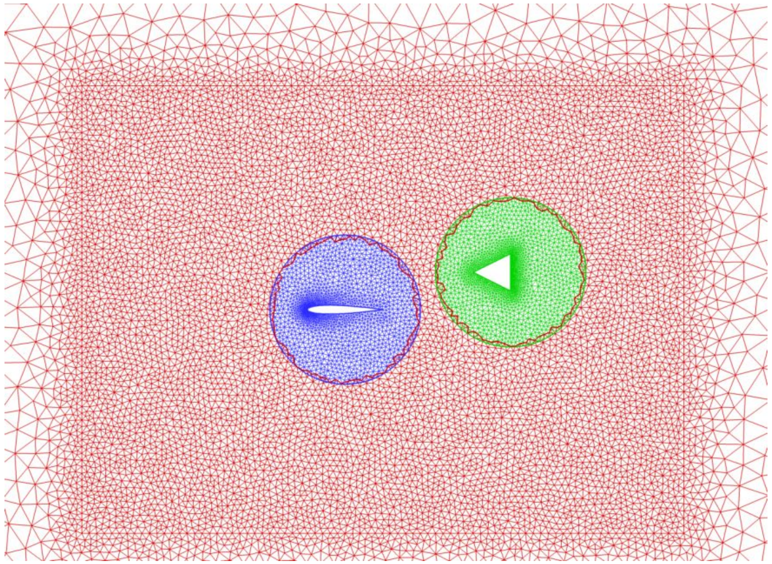
Time history of coefficient of lift

Outline

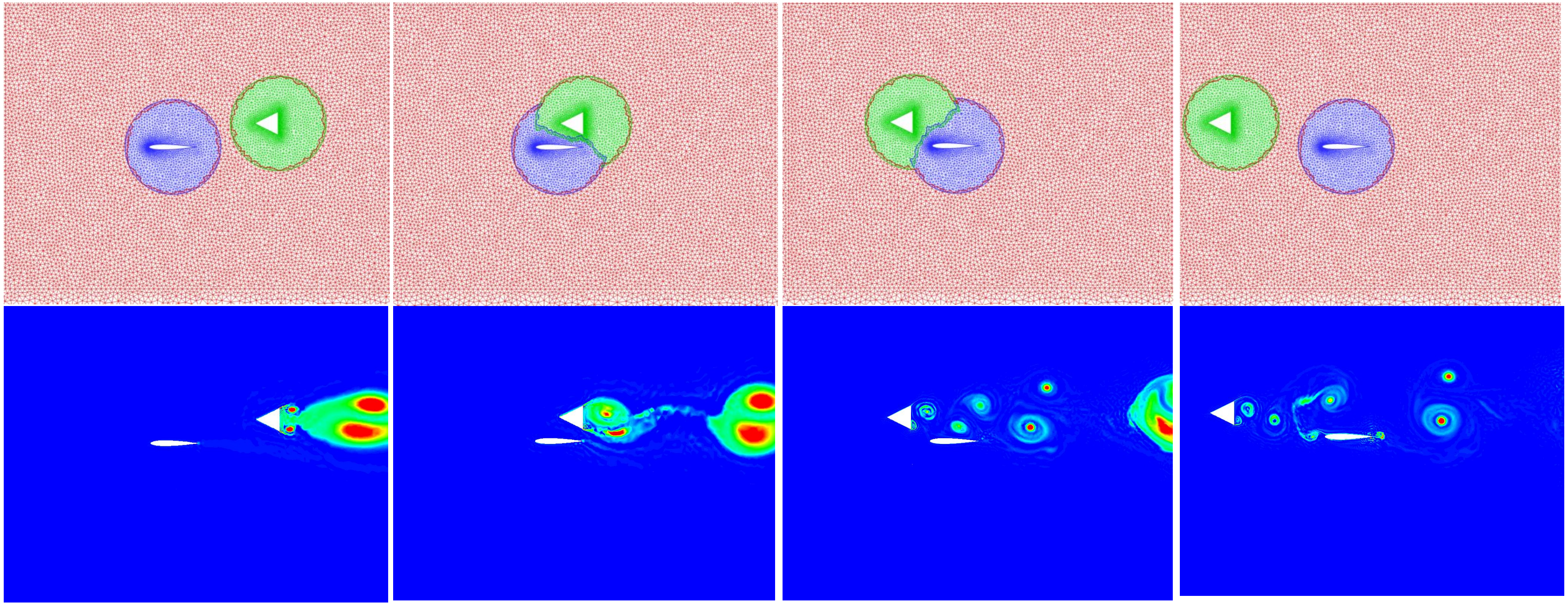
- Hole cutting
- Governing equations
- Overset methodology
- Overset results
  - Modified preconditioner
  - Manufactured solutions
  - Steady turbulent
  - Unsteady moving boundary
  - Relative motion between two bodies
- Adaptive overset
- Conclusion

Relative Motion Between Two Bodies

- Inviscid simulation
- Demonstration of dynamic hole cutting
- Free stream $M_{\infty} = 0.1, \alpha_{\infty} = 0^{\circ}$
- Airfoil is stationary. Triangle wedge moves upstream at $M = 0.1$
- Non-dimensional chord length = 1
- Non-dimensional time step = 0.05
- Modified IHC is used



Relative Motion Between Two Bodies



t=3 t=20 t=34 t=43

Grids (after hole cutting) and entropy contour, P_2 elements

Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
- Adaptive overset
  - Adaptation methodology
  - Preliminary results
- Conclusion

Adaptation Methodology

- Ahrabi, B.R., Anderson, W.K., Newman III, J.C., "High-Order Finite-Element Method and Dynamic Adaptation for Two-Dimensional Laminar and Turbulent Navier-Stokes," 32nd AIAA Applied Aerodynamics Conference, June 2014, AIAA Paper 2014-2983.
  - Dynamic hp-adaptation
  - Adjoint-based (steady), and featured-based (steady & unsteady)
  - Weight function is contiunous across cell interface (no need to calculate the flux)
  - Efficient handling of hanging nodes
    - Implemented simply by adding a static condensation step to every continuous Galerkin method
  - Discretization is conservative

Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
- Adaptive overset
  - Adaptation methodology
  - Preliminary results
- Conclusion

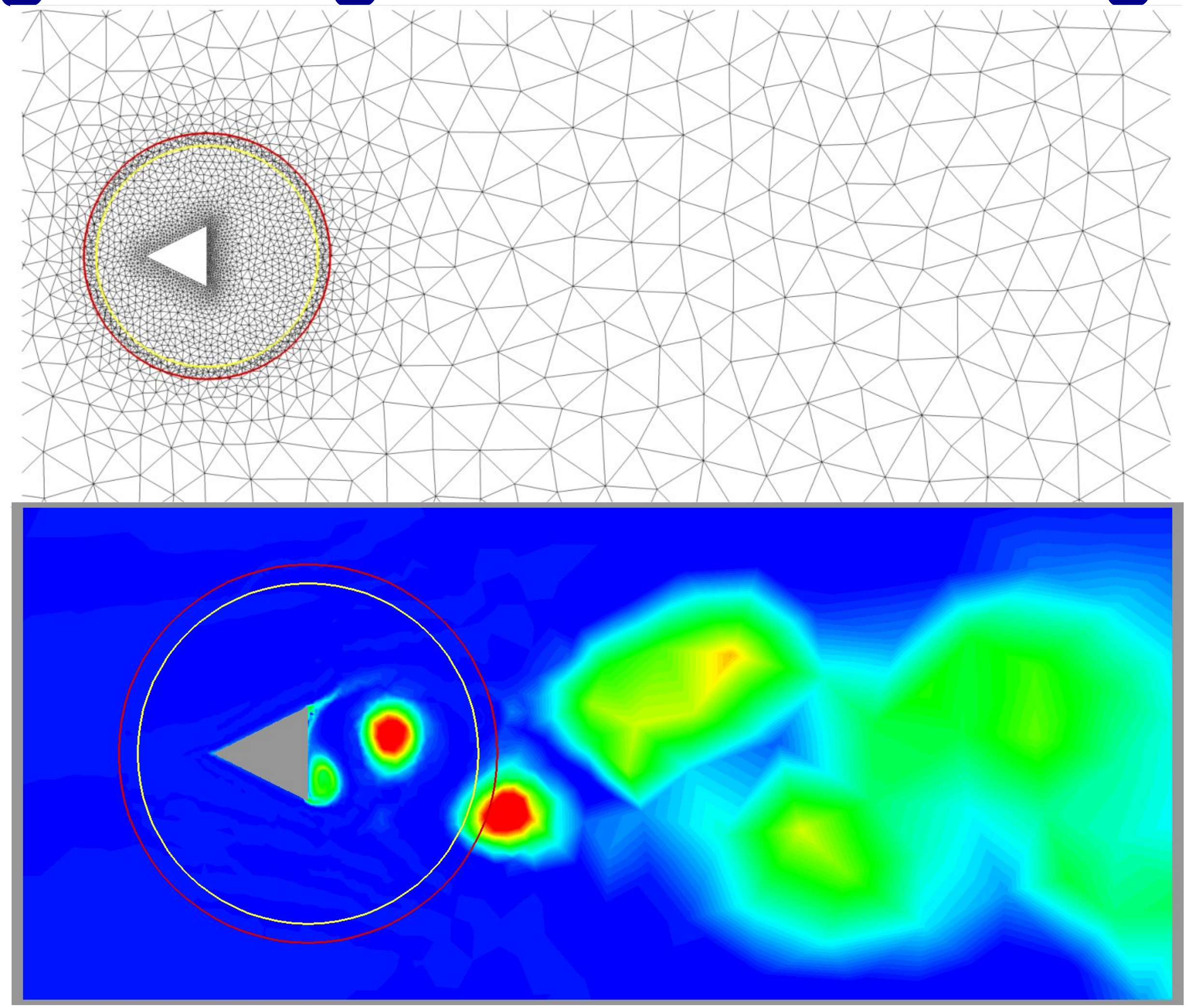
Triangle Wedge Vortex Shedding

Inviscid

M=0.2

P_1 element

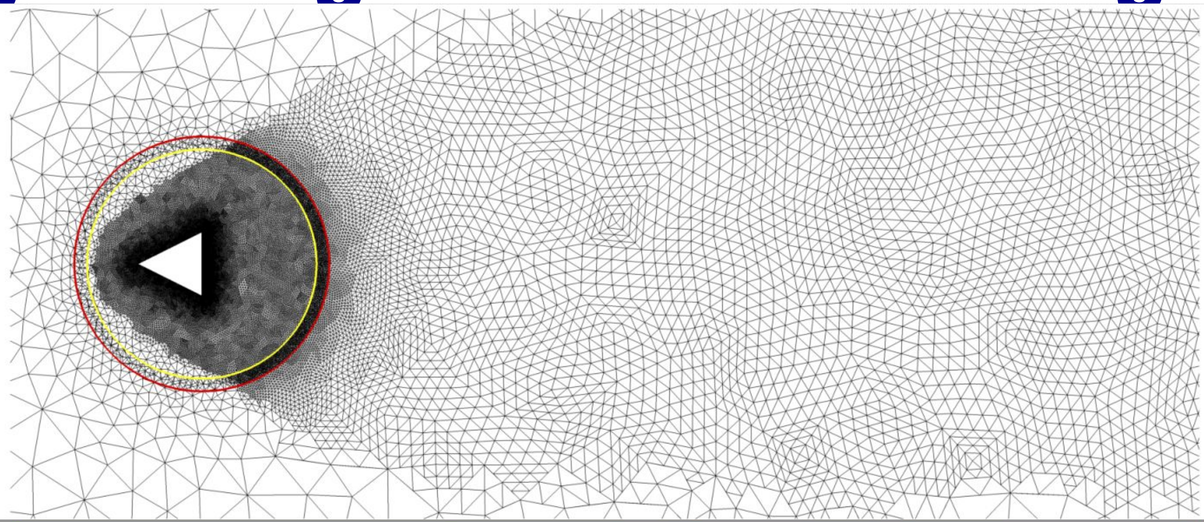
No adaptation

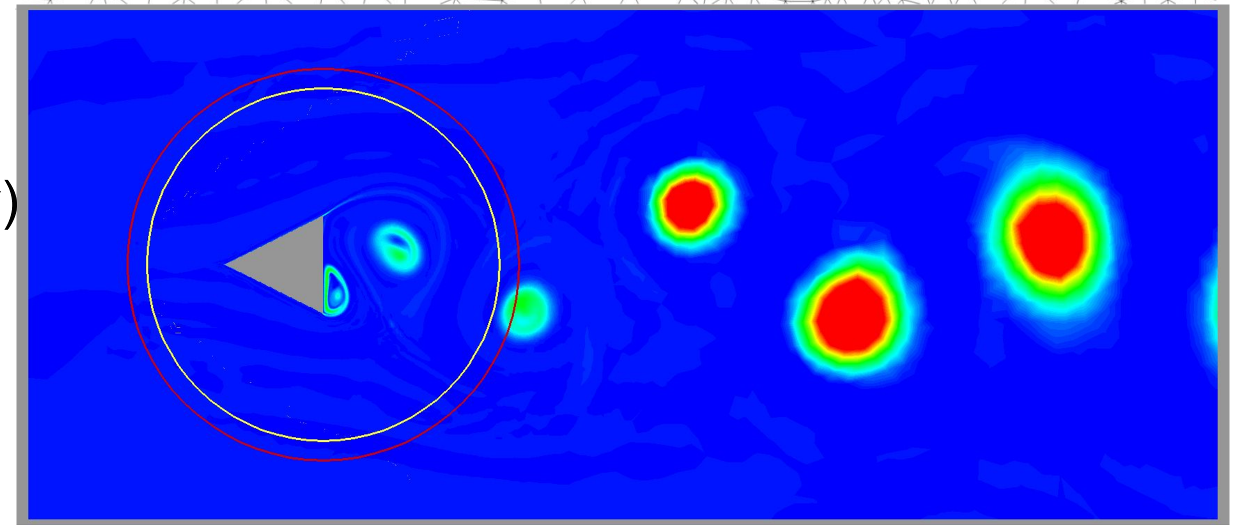


No adaptation

Triangle Wedge Vortex Shedding

- Inviscid
- M=0.2
- P_1 element
- Feature based
    - H-adaptation
    - (refinement only)





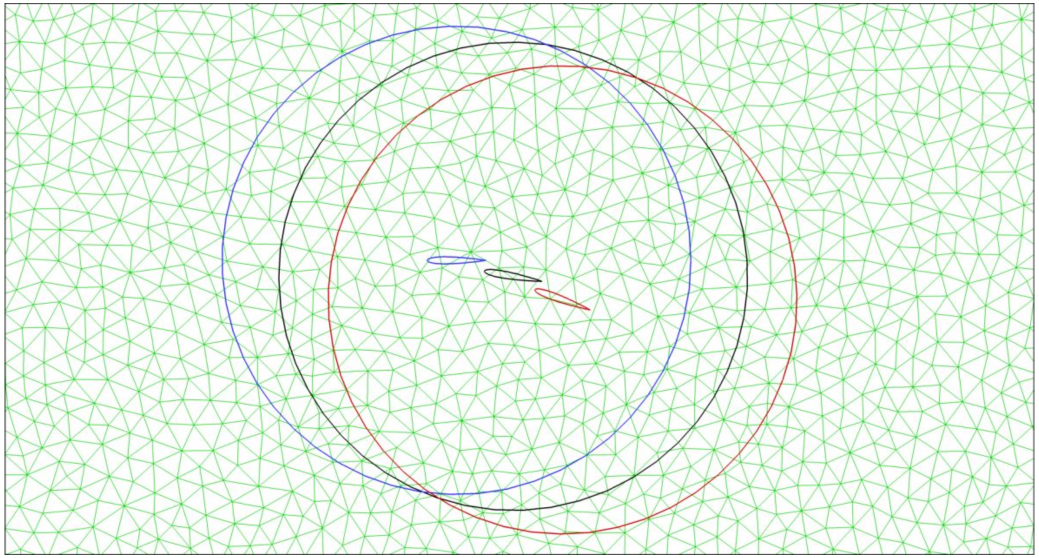
With h-adaptation (refinement only)

Multiple Airfoils

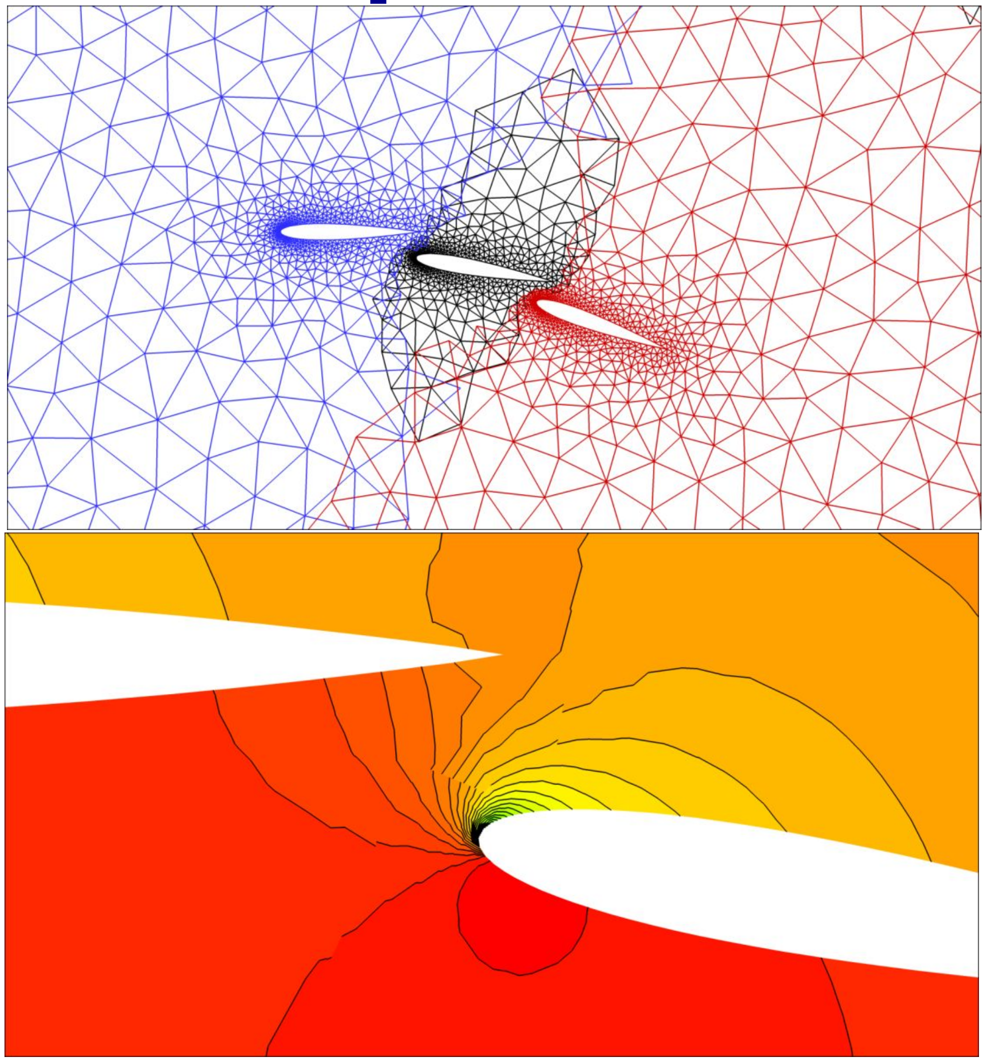
Inviscid, steady

M=0.2

P_1 element

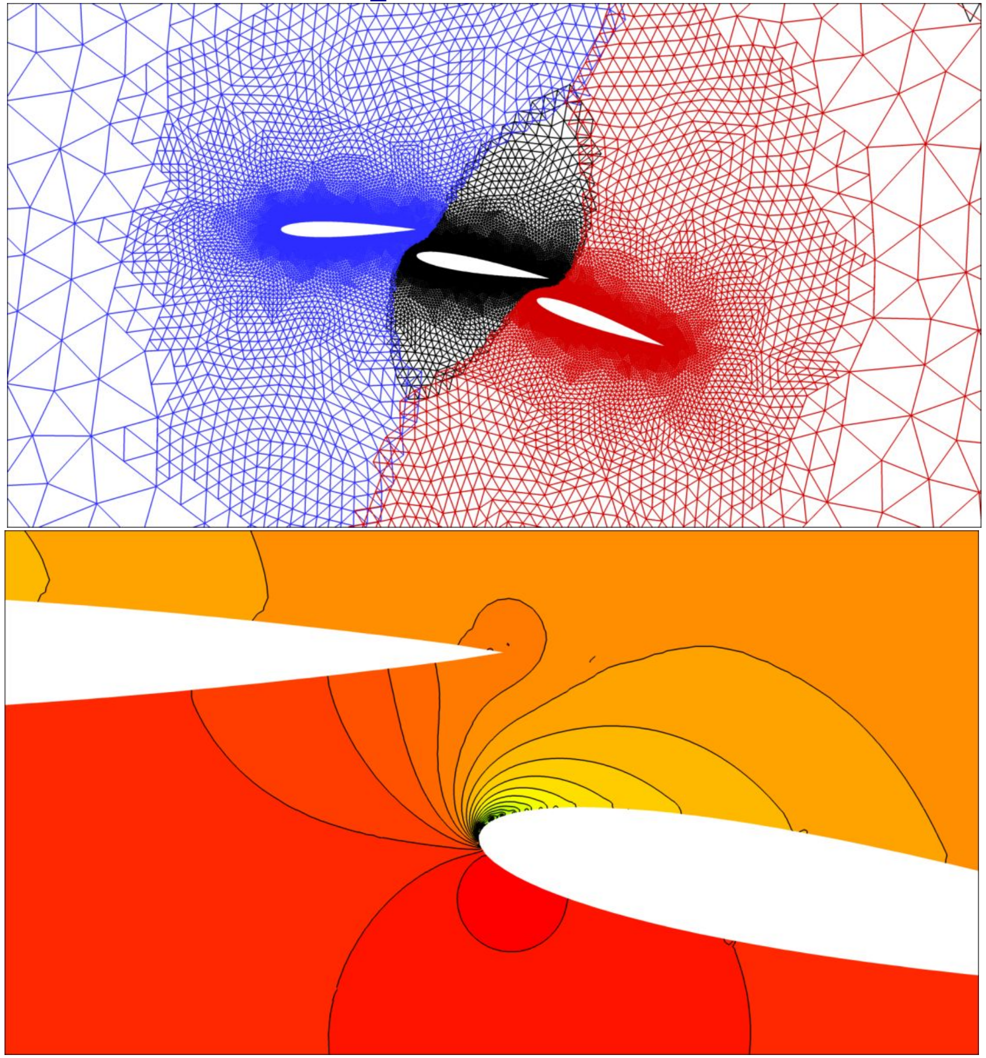


Multiple Airfoils



No adaptation

Multiple Airfoils



H-refinement in specified region

Outline

- Hole cutting
- Governing equations
- Overset methodology
- Overset results
- Adaptive overset
- Conclusion

Conclusion

- Development of a novel hole cutting procedure: Elliptic Hole Cutting
- Development of modified preconditioners for overset grid computations
- Demonstrated that the design order of accuracy of the method is retained using the method of manufactured solutions
- Demonstrated the method for steady-turbulent and for dynamic moving boundary simulations
- First implementation of a high-order SUPG overset grid scheme
- Demonstrated the potential of using adaptation in overset scheme
- Prototyping in 2D complete. Extension to 3D underway