SOLUTION ADAPTATION IN EMBEDDED BOUNDARY METHODS: ADAPTIVE MESH REFINEMENT VS. ADAPTIVE REMESHING

D. Huang\textsuperscript{3}, R. Borker\textsuperscript{1}, P. Avery\textsuperscript{1} and C. Farhat\textsuperscript{1,2,3}

\textsuperscript{1}Aeronautics and Astronautics
\textsuperscript{2}Mechanical Engineering
\textsuperscript{3}Institute for Computational and Mathematical Engineering
Stanford University

E. Hachem and A. Larcher
CFL Computing and Fluids Research group
CEMEF - Mines ParisTech
PROBLEMS OF INTEREST

- Viscous Fluid-Structure Interaction (FSI) problems

- Large motions and/or deformations, topological changes
**Computational Frameworks for FSI**

**Arbitrary Lagrangian Eulerian (ALE)**
- Stable, accurate for small motions and/or deformations
- Lacks robustness for large motions and/or deformations, and topological changes: mesh entanglement, re-meshing

**Embedded/Immersed Boundary Method (EBM)**
- Robust for large motions or deformations; simpler mesh generation procedure
- Viscous flows: does not track boundary layers, complicates interface treatment
Adaptive Mesh Refinement (AMR) vs Adaptive Remeshing (AR)

**AMR for structured meshes**
- Local grid/patch-based adaptation
- Non-conforming meshes (hanging nodes)

*Iaccarino (2004), Vanella (2014)*

**AR for unstructured meshes**
- Conforming meshes
- Adaptation via re-meshing

*Hachem (2013), Abgrall (2014)*
OBJECTIVE

- Comparison between AMR and AR for
  - tracking and resolving boundary layers
  - resolving all flow features of interest

in the context of an EBM for highly nonlinear FSI problems
AMR VS. AR

- **AMR (isotropic, because the context is that of an EBM)**
  - tetrahedral meshes
  - newest vertex bisection algorithm for edge refinement/coarsening \(\rightarrow\) mesh conformity is preserved

- **AR (anisotropic)**
  - defines anisotropic directions
  - generates stretched elements (large aspect ratios)
AMR – CRITERIA

- Tracking boundary layers and keeping them well resolved
  - edge selection criterion: wall distance

\[
|W^m - W_h^m|_{\infty,k} \leq c |W^m - \Pi_h W^m|_{\infty,k} \leq c_d \max_e |e^T H e|, \quad m = 1, \ldots, n
\]

- Tracking flow features and resolving them
  - edge selection criterion: Hessian-based error indicator

Cea’s lemma (elliptic)

Taylor series scheme-independent (curvature)
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)

---

initial edges

initial nodes
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)

step 2: mark reference edges of the elements associated with marked edges

- initial edges
- initial nodes
- element ‘reference edge’
- edges marked for refinement
- reference edges marked for refinement
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)

step 3: refine marked reference edges
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)

step 4: bisect marked edges
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)

step 1: mark edges for coarsening

- initial edges
- initial nodes
- element ‘reference edge’
- edges marked for coarsening
- new nodes (level 1)
- new nodes (level 2)
Adaptive mesh refinement
- local coarsening and refinement (newest vertex bisection)

step 2: delete only newest nodes (highest level) and repeat
AMR – PROCEDURE

- Adaptive mesh refinement
  - local coarsening and refinement (newest vertex bisection)

repeat step2 till all coarsening is completed
Adaptive remeshing

- Hessian-based error indicator along the edges

\[ |(\nabla \Pi_h u - \nabla u(X^i)) \cdot X^{ij}| \leq \max_{Y \in \{X^i, X^j\}} |X^{ijT} H(Y) X^{ij}| \]
\[ = e_{ij} \]
- adapted mesh metric associated with each node

\[ M^i = \left( \frac{d}{\Gamma(i)} \sum_{j \in \Gamma(i)} s_{ij}^2 X^{ij} \otimes X^{ij} \right)^{-1} \]

\[ s_{ij} = 1: \text{no stretching}; \quad s_{ij} = C e_{ij}^{-1/2}: \text{error-based stretching} \]
Adaptive remeshing

- generate an optimal mesh based on the nodal-based metric $M^i$ using the local remeshing library MTC*

*Coupez et al. "Parallel meshing and remeshing."
Laminar flow over a collapsing cylinder

- $D = 1$; $\frac{L}{D} = 2$; $f = 1/2\pi$
- $M_\infty = 0.2$; $Re = 100$
Prescribed structural motion (displacement)

\[
[d_x, d_y] = \left( \frac{|x|+|y|}{\sqrt{2}} - \sqrt{x^2+y^2} \right) \left[ \frac{-\text{sign}(x)}{\sqrt{2}}, \frac{-\text{sign}(y)}{\sqrt{2}} \right] \left( 1 - \cos(2\pi ft) \right)
\]
AMR VS. AR

$C_d = 1.05$

$C_d = 1.1$
AMR VS. AR

AMR

AR
- both methods deliver similar peak drag coefficients
- both method captures multiple modes
- phase discrepancies exist
- AR exhibits smoothness issues
AMR VS. AR

- both methods deliver similar peak drag coefficients
- both method captures multiple modes
- phase discrepancies exist
- AR smoothness issues are attenuated
AMR VS. AR

Efficiency comparison

- **AMR**
  - 48 cores
  - adaptation frequency = 1
  - total time = 14.1h
    - o AMR = 2.8h
    - o balancing = 0.4h

- **AR**
  - 64 cores
  - Adaptation frequency = 5
  - total time = 19.15h
    - o AR time = 9.58h
    - o balancing = 0.65h
AMR VS. AR

Simulations without AMR/AR on uniform meshes of similar sizes

AMR/AR improve prediction of drag coefficient
AMR for EBM (FIVER)

+ more computationally efficient
+ population of new nodal values is Total Variation Diminishing (TVD)
+ preserves the element quality of the initial mesh

- more elements than AR due to isotropy, but still faster than anisotropic AR
- wall model is needed for high Reynolds number simulations
CONCLUSIONS

AR for EBM

- fewer elements but still slower than AMR
- large aspect ratios, which is favorable for RANS computations, but still slower than AMR (at least current implementation)
  - massive interpolations incurred by remeshing lead to spurious oscillations