Overview of Capabilities of Suggar++ With Emphasis on Recent Improvements

Ralph Noack
David Boger

Penn State University Applied Research Laboratory
• Brief overview of capabilities
• Details on what is new
  – Integrated USURP
  – Periodic passage: Turbine blade row
  – 650 million grid point grid system
  – High-order overset interpolation
• Summary
BRIEF
SUGGAR++ OVERVIEW
Suggar++ Overview

• Built upon experience with SUGGAR
  – Complete rewrite
  – Improved algorithms

• Significantly better than SUGGAR
  – Performance: memory and speed
  – New capabilities

• Integrated with new Pointwise OGA capability
Suggar++ Overview

• Node- and/or cell-centered assembly

• Grid types
  – Structured: Curvilinear and Cartesian
  – Unstructured
    • Tet, Mixed element, Octree
    • General polyhedral to be added in future
• Hole cutting
  – Direct cut, analytic, octree, manual

• Integrated surface assembly
  – Structured and/or mixed element grids

• Overlap minimization using general Donor Suitability Function
  – Including distance-to-wall

• Support for arbitrary structured solver stencil
Suggar++ Overview

- Hybrid parallel execution
  - Threads and/or MPI
  - Decomposition of structured or unstructured grids

- Unstructured grid refinement

- Designed for moving body problems

- Link into flow solver for integrated dynamic OGA
  - Dynamic Groups: hide OGA execution time
Suggar++ Overview

- Integrated USURP to support F&M integration
- Periodic passages
- High-order support
  - Arbitrary number of fringes
  - High-order interpolation for structured grids
INTEGRATED USURP CAPABILITY
Integrated USURP capability

• Similar but not identical to the USURP utility
  – Different coding
  – Uses CLIPPER for polygon clipping
    • More robust than GPC used in USURP
  – Triangulation routines are different than USURP

• Panel weights
  – Included in DCI file: Can be retrieved via DiRTlib
  – Written to files

• Can create zipper grid
  – Not sufficiently robust
Suggar++ USURP output for WingBody

Triangulated clipped panels

Clipped and unclipped panels
Suggar++ USURP output for WingBody

Zipper grid:
Triangles contain only points in the original grid
PERIODIC PASSAGE
CAPABILITY
Periodic Passage

- Simulate one blade row out of full wheel
- Typically non-planar periodic boundary surfaces
- Flow solver uses periodic boundary conditions to simulate effect of all blades
Periodic passage
Periodic Passage: Complexities For Overset

• Donor stencil member can be across periodic boundary
  – Virtual donor member grid: flow solver must transform velocities

• Grid can extend into neighboring passage
  – Donor search must extend into virtual grid
    • Not yet implemented

• DCI must be consistent across boundaries
Periodic Passage: Complexities For Overset

• Grids that overlap on passage boundary will have different representations of passage boundary
  – Treatment to prevent flood fill leaks along passage boundary
  – May need surface assembly to keep points inside passage
    • Ignored to be consistent with full wheel/non-periodic grid system
SUGGAR++ EXECUTION OF VERY LARGE GRID SYSTEM
Suggar++ Execution of Very Large Grid System

- Athena grid system obtained from Pablo Carrica (U. Iowa/IIHR)
  - Appended, propeller, moving rudder, refinement grids
  - Plane of Symmetry
  - Provided Medium and Fine grid systems
Set Of Grid Systems

- Started with Medium and Fine grid systems
- Mirrored and refined to get larger systems
  - Refined by factor of 2 in each direction

<table>
<thead>
<tr>
<th>Case</th>
<th>Number of grid points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Medium</td>
<td>14,374,923</td>
</tr>
<tr>
<td>Medium-mirrored</td>
<td>28,749,846</td>
</tr>
<tr>
<td>Medium-mirrored-refined</td>
<td>225,434,040</td>
</tr>
<tr>
<td>Fine</td>
<td>40,871,388</td>
</tr>
<tr>
<td>Fine-mirrored</td>
<td>81,742,776</td>
</tr>
<tr>
<td>Fine-mirrored-refined and partitioned</td>
<td>653,279,190</td>
</tr>
</tbody>
</table>
Memory Usage
Fine Grid: ~40 Million points

Suggar++ using significantly less memory
Used 20 nodes and only one thread/node
  - Limit of available nodes
  - Used only one thread to limit memory usage
  - Max memory used was 22 GB
    - Nodes had 24 GB memory
    - Probably paging with degraded performance

Time and resources did not permit a detail performance analysis
  - Successful feasibility demonstration
HIGH ORDER OVERSET INTERPOLATION
High Order Overset Interpolation

• Suggar++ has had the ability for an arbitrary number of fringe levels from the beginning

• Added high-order interpolation using Lagragian polynomials
  – Using Lagrangian Polynomial approach of Scott Sherer
  – Implemented for Node- and Cell-Centered assemblies
High-Order Overset Interpolation
Examination of Accuracy

• Suggar++ supports a Monitor Grid
  – A set of points where all points/cells are fringes
  – Useful for interpolating solution onto a new grid
  – Examine solution at specific points

• Examine order of accuracy of high order interpolation
  – Background/donor grid system
    • Series of grids refined by factor of 2 in each direction
  – Dependent variables will be analytic functions
    • Linear
    • Polynomial of varying order
    • Exponential, analytic vortex
  – Monitor grid to obtain interpolated values at set of locations
  – Find the maximum deviation between the interpolated and exact
• Single uniform Cartesian donor grid
• Monitor grid is a randomly perturbed set of points
  – To prevent fortuitous alignment
• Only show results for analytic vortex
• Convergence slope is computed using gnuplot
• Inverse distance weighting is first order
• Least square, dual hex, trilinear are 2\textsuperscript{nd} order
- 6 point Lagrangian polynomial is 6th order
Block-to-Block Grid Systems

- Block-to-Block grid systems arise from
  - Grid generation
  - Grid decomposition/splitting

- How does this affect high-order interpolation?
Consider A Single Grid
Decompose Into Block-to-Block Grid System
Cell-Centered High-Order

• Cell-centered high-order interpolation uses dual grid
  – Structured grid connecting cell centers
Cell-Centered Dual Grid

- Dual grid does not cover volume
- Will require extrapolation
• **Analytic Vortex**  
  – 6th order Lagrangian polynomial  
  – Include Extrapolation: < 0 , > 20

• **Results:**  
  – Within range: oscillates about exact  
  – Extrapolation: can produce large deviations

Difference Between Exact and Polynomial Values
• Extend grid into neighboring block-to-block grids
• First into block-face neighbors
Use “Extended Grid“

- Extend grid into neighboring block-to-block grids
- Next across edges and corners
6th Order Interpolation
Extended Grid Results

• Extended grid yields same results as single grid
Real World: Turbine Blade

- Single blade
  - Periodic passage
- Cell-centered Block-to-block grids
- Passage boundaries are not displayed
Real World: Turbine Blade

- Small tip gap
Real World: Turbine Blade

- Unstructured B2B connection
- Primary grid plane
- Extended grid planes
  - Connects cell centers
Simplified Grid System

- Cannot fully extend grid
- Leaves void where extrapolation is required
Void in Extended Grid Reduces Accuracy

Order=ho6 $F=5.0 \times \text{pow}(M_E, r^r/16.0)$

- cc
- slope=-2.03
- cc-single-grid

Plot showing Max Deviation vs. N Points with a logarithmic scale.
• Surface assembly required to “shift/project” fringes into donor grid

• Bilinear surface representation not appropriate for high-order interpolation
  – Must use high-order polynomial representation

• Cell-centered assembly must use dual surface

• Viscous spacing + stretching + high-order
  – Sensitive and difficult problem
  – Not happy with the current results
SUMMARY
Summary

- Suggar++ is the most general overset grid assembler available
- Significant improvement over SUGGAR
- New capabilities were presented
  - Integrated USURP
  - Periodic passage
  - High-order interpolation
- Demonstrated execution for very large grid system
• Presented difficulties with high-order interpolation for cell-centered assemblies
  – Dual grid does not cover domain
  – Extended grid can recover single grid results
  – Block-to-Block topologies can leave regions without extended grid
    • Adversely affects interpolation accuracy

• Recommend node-centered approach for high-order overset solutions
Acknowledgments

- Funding provided by
  - NASA Grant No. NNX10AQ74G
    - Technical Officer: Molly White
  - ONR Grant N00014-12-1-0315
    - Technical Officer: Dr. Patrick Purtell
Commercial distribution and support for Suggar++ provided by

Celeritas Simulation Technology, LLC

http://www.CeleritasSimTech.com

Exportable under an EAR-99 license