Direct Generation of 3D Overset Grids from Solid Models

John F. Dannenhoffer, III
Syracuse University
Robert Haimes
Massachusetts Institute of Technology

10th Symposium on Overset Composite Grids and Solution Technology
September 21, 2010

Overview

- Background
- Solid Model / Overset Grid Duality
- Overall Strategy
- Example: JMR3
- Summary / Next Steps

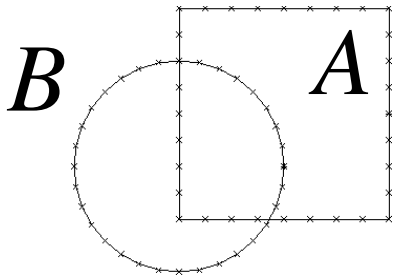
Speed Improvement Opportunities

- **Bottom-up approach**
  - currently implemented in OverGrid
  - user decomposes configuration
  - extensive toolkit for performing low-level operations
  - script library for “replaying” grid generation process
  - very labor intensive

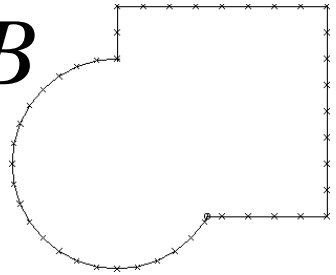
- **Top-down approach**
  - currently prototyped in OvrCad
  - decomposition is based upon solid models

Solid Models

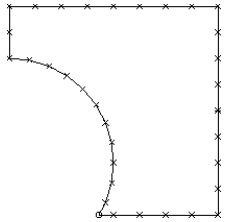
- **Basis of all modern CAD systems**
- **Any solid configuration can be made by Boolean combinations of primitive solids**
- **Primitives**
  - Simple: bricks, wedges, pyramids, cylinders, spheres , cones, tori, …
  - Higher-order: extrusions, rotations, sweeps, …
- **Boolean operations**
  - Union, difference, intersect
- **Transformations**
  - Scale, rotate, translate



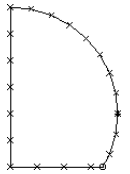
$A \cup B$



$A - B$



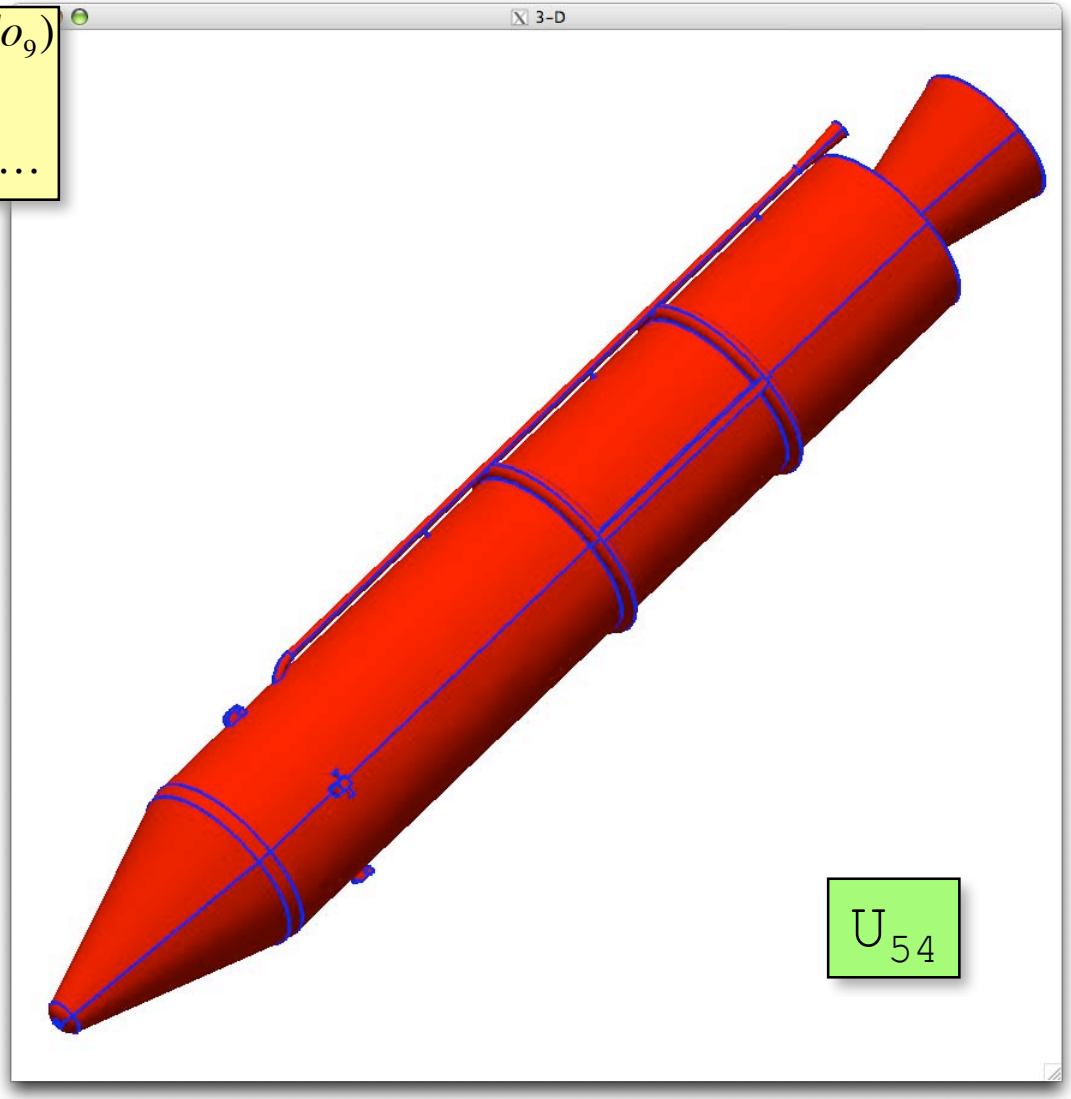
$A \cap B$



Solid Model Build-up of Rocket Config.

$$R_7(R_4(Co_1 \cap_3 Sp_2) \cup_6 Cy_5) \cup_{11} (Co_8 -_{10} Co_9)$$
$$\cup_{13} To_{12} \cup_{15} To_{14} \cup_{17} Bx_{16}$$
$$\cup_{31} (To_{18} \cap_{20} Bx_{19} \cup_{22} Cy_{21} \cup_{24} Cy_{23} \cdots) \quad \cdots$$

```
U17
. U15
. . U13
. . . U11
. . . . R7
. . . . . U6
. . . . . . R4
. . . . . . . I3
. . . . . . . . Co1
. . . . . . . . Sp2
. . . . . . . Cy5
. . . . -10
. . . . . Co8
. . . . . Co9
. . . To12
. . To14
. Bx16
```



Solid Model / Overset Grid Duality

- **New idea: Exploit the duality between solid model creation and overset grid generation**
  - for each primitive in feature tree
    $\Rightarrow$ generate one or more grids
  - for each boolean operation in feature tree
    $\Rightarrow$ perform overset grid transfers

Overall Strategy - 1

- **Start with solid model feature tree**

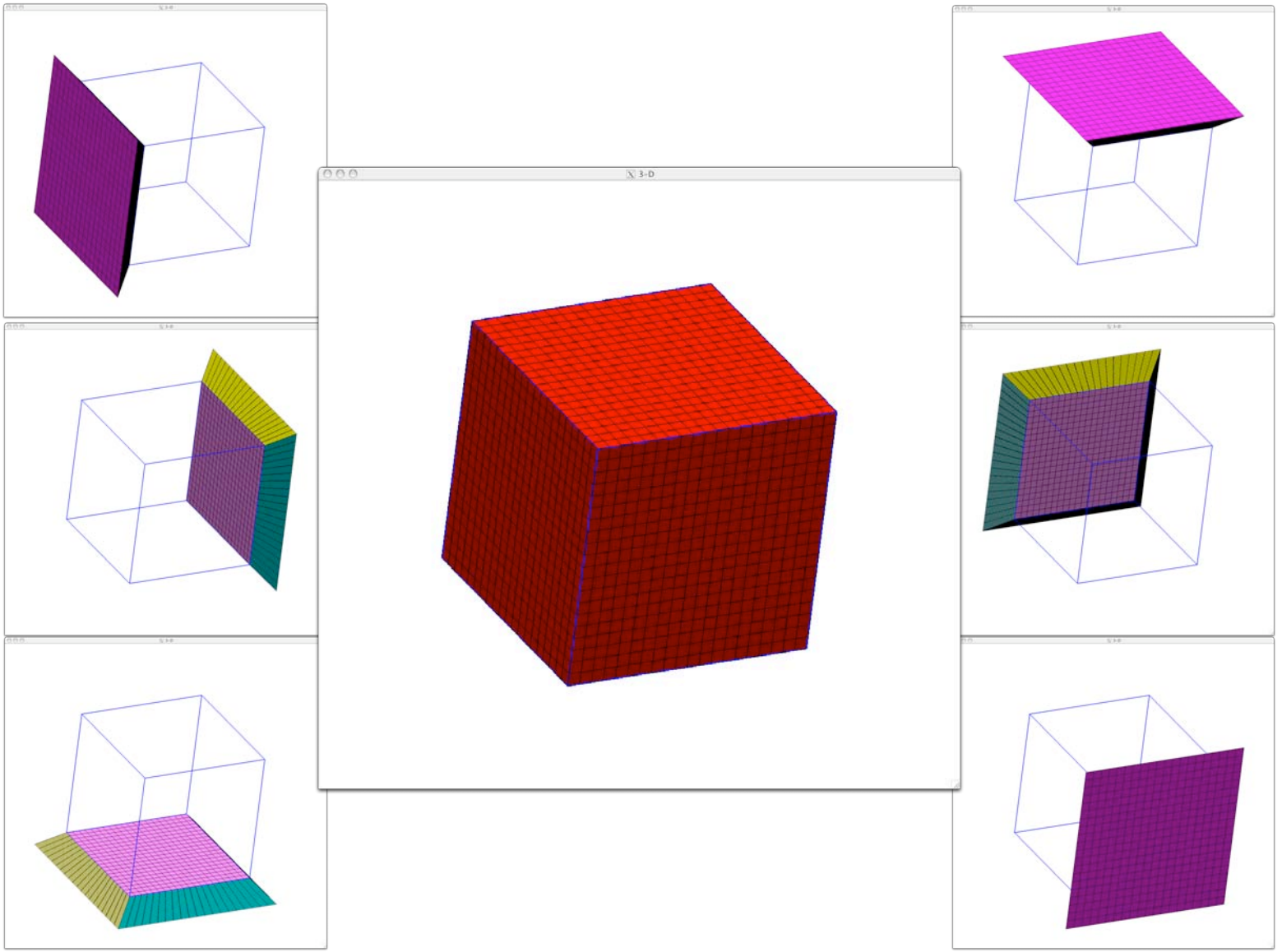
$$B_1 - [C_1 \cup ((B_2 - C_2) \cap C_3]$$

- **Determine if grid should be generated inside or outside each primitive**

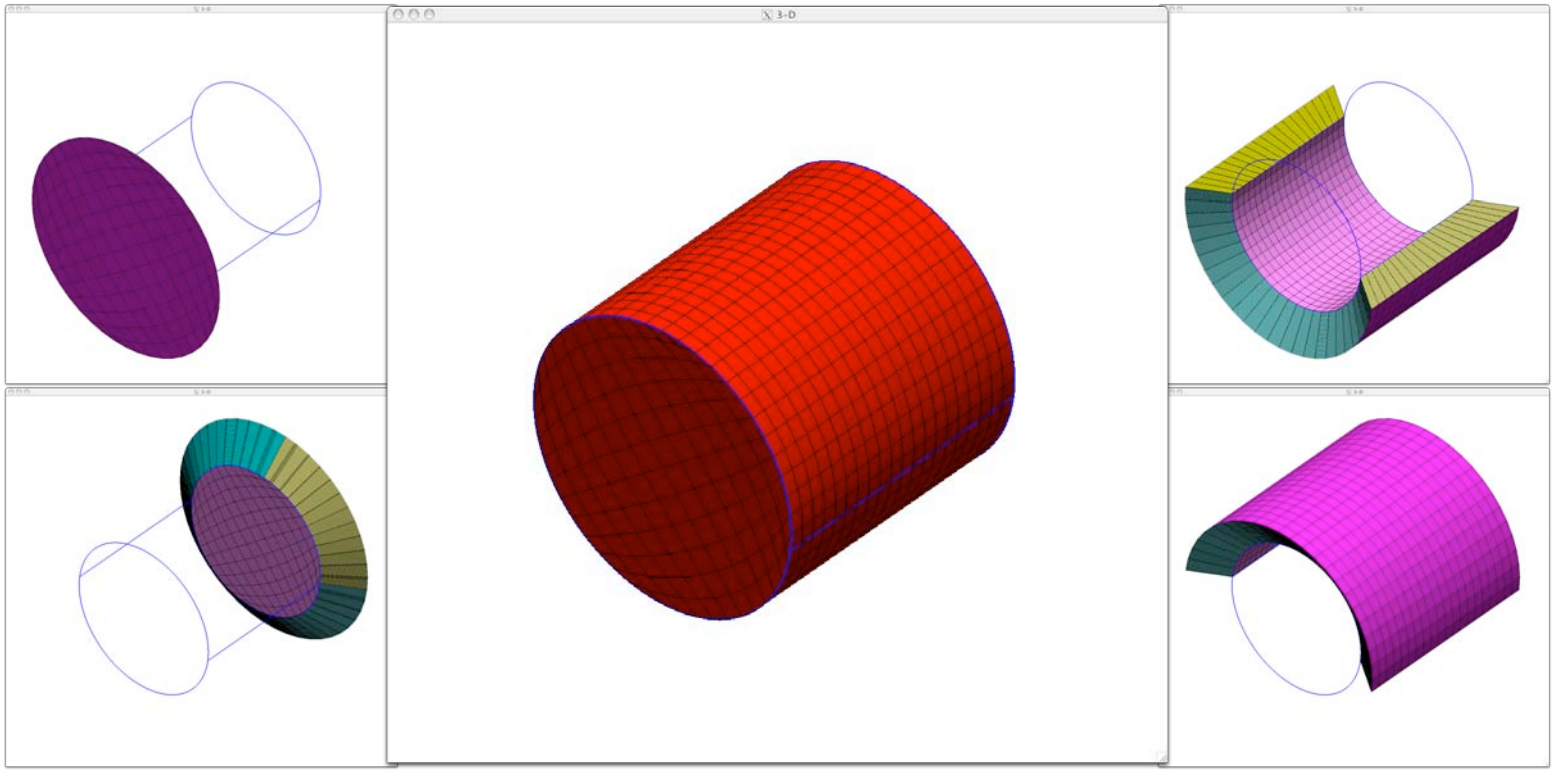
in out out in out

- **Create body-fitted grid(s) for each primitive solid**
  - only generate grids for “active” faces

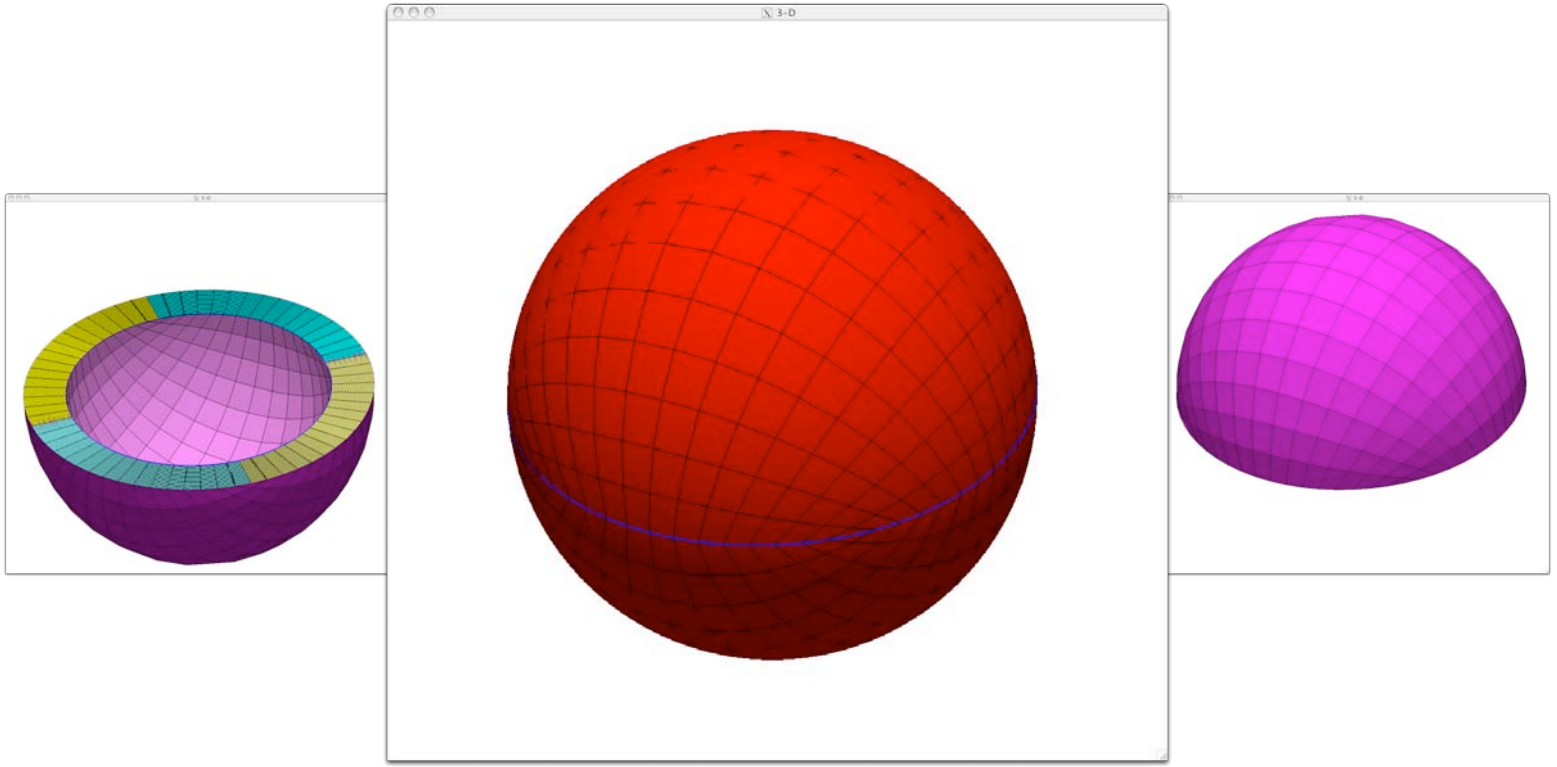
Basic Grids: Box



Basic Grids: Cylinder

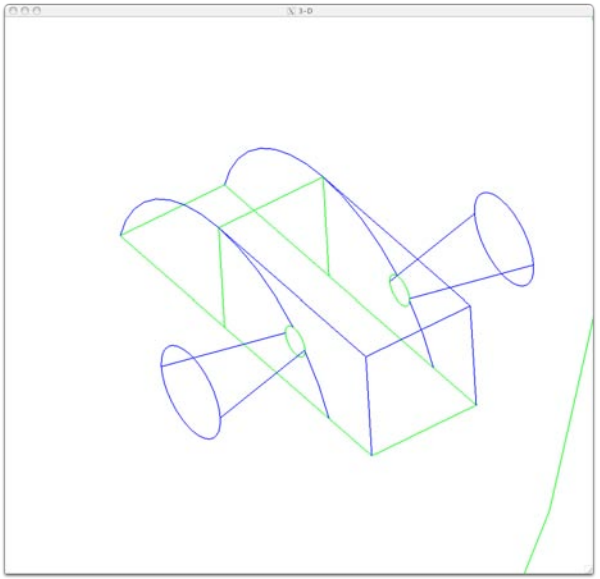


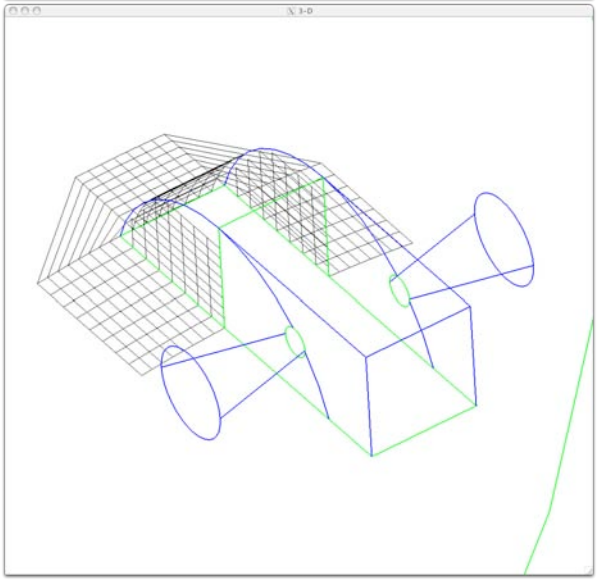
Basic Grids: Sphere



Overall Strategy - 2

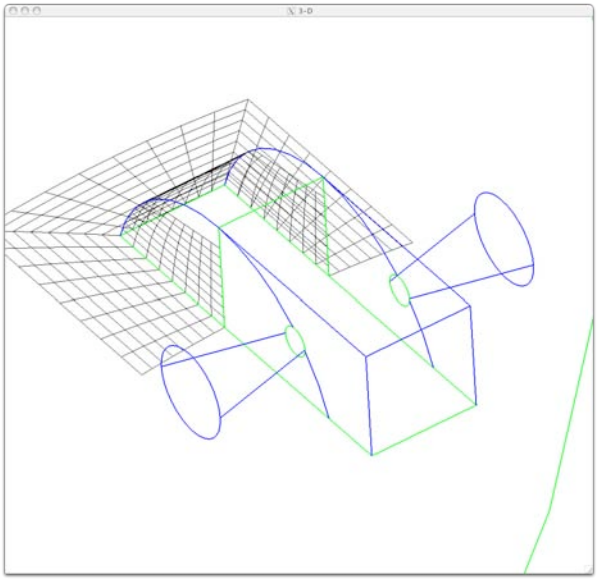
- **Create collar grids for every “boolean” edge**
  - Find string of edges between 3-way joints and nodes with >4 edges
  - Generate grid on adjoining faces

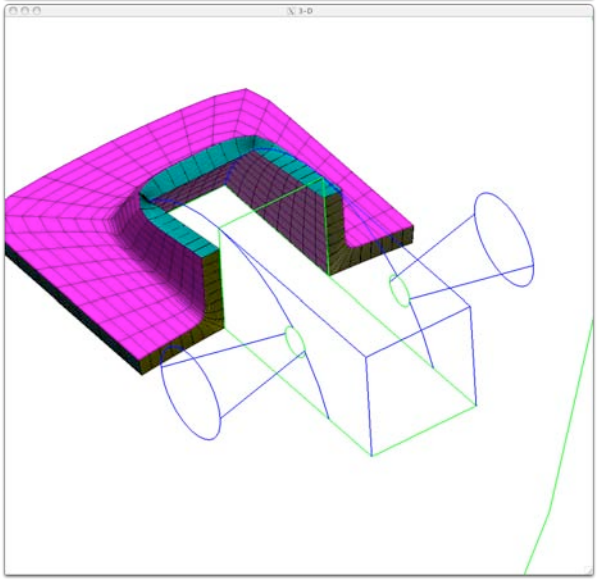




Overall Strategy - 3

- **Create collar grids for every “boolean” edge**
  - Modify the grid to coincide with incident edges
  - Generate field grid using HYPGEN





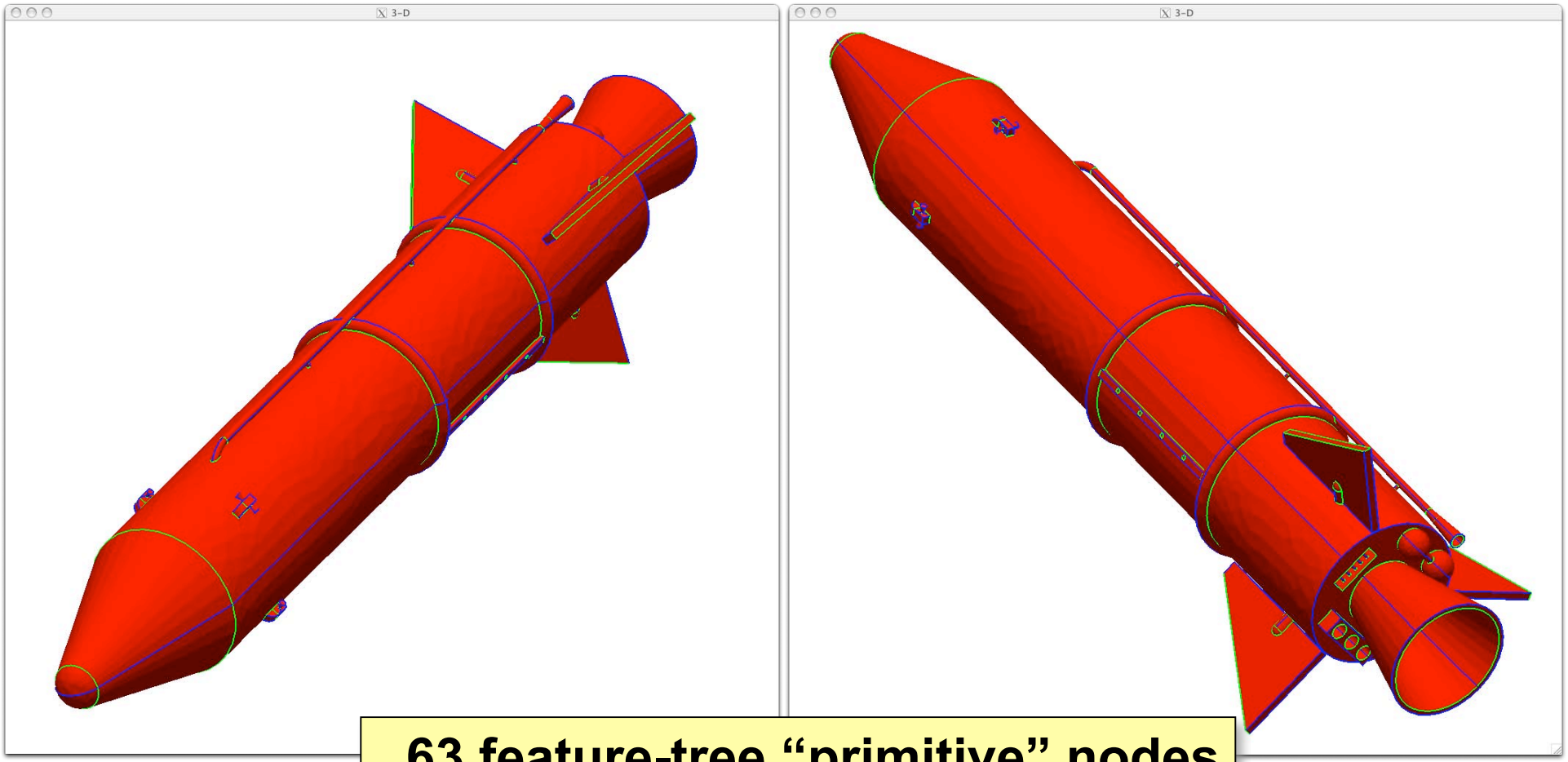
Overall Strategy - 4

- **Remove inactive grids**
  - Any primitive-based grid that has no “boundary” points is “inactive” and removed
- **Cut holes & create interpolation stencils**
  - Use standard “hole cutting” package (e.g., PEGASUS, SUGGAR, …)
- **Solve flow problem**
  - Use standard “overset flow solver” (e.g., OVERFLOW, …)

Overall Strategy - 5

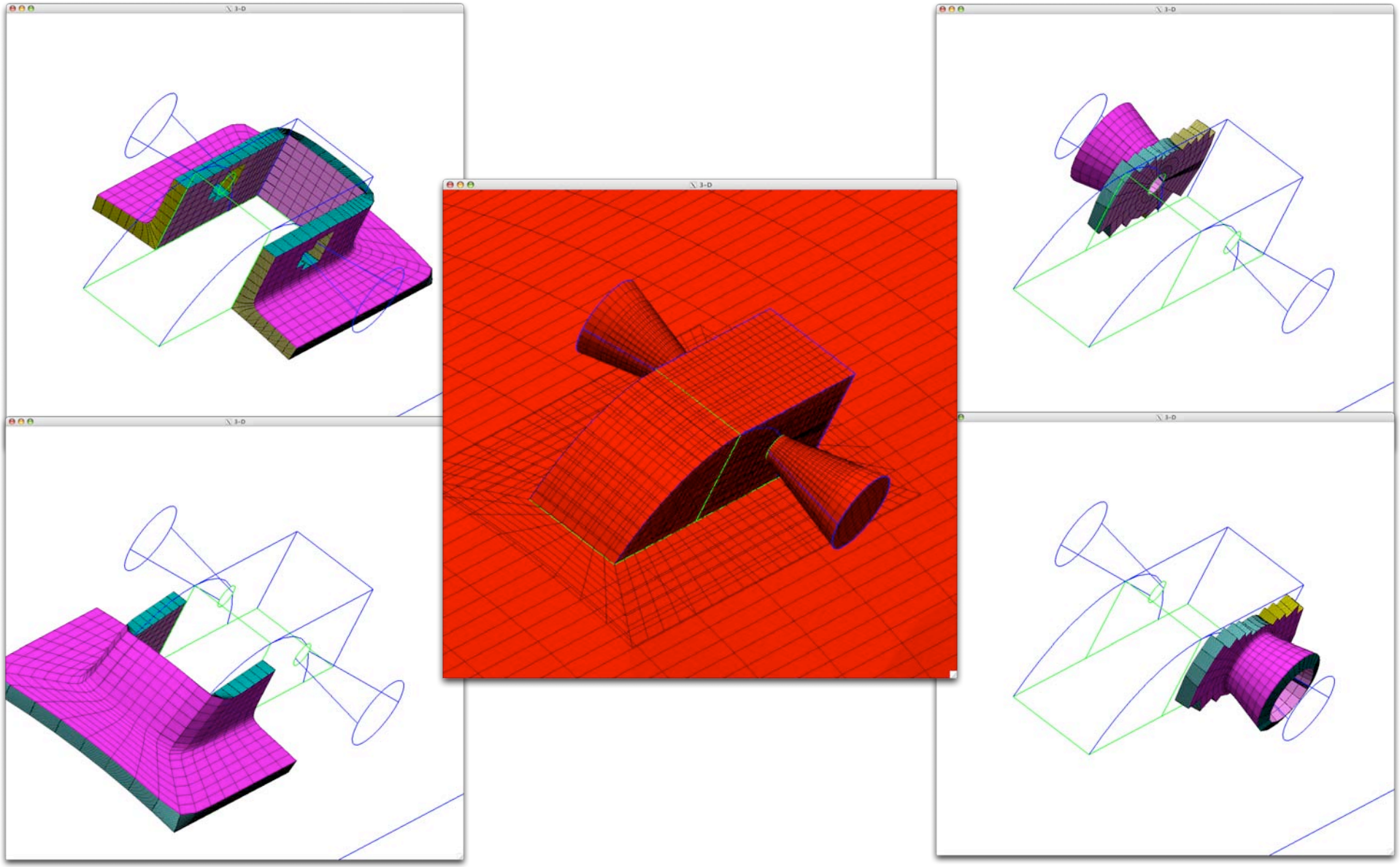
- **Cut holes & create interpolation stencils**
  - Use standard “hole cutting” package (e.g., PEGASUS, SUGGAR, …)
- **Solve flow problem**
  - Use standard “overset flow solver” (e.g., OVERFLOW, …)
- **Post-processes and visualize**

Example: JMR3

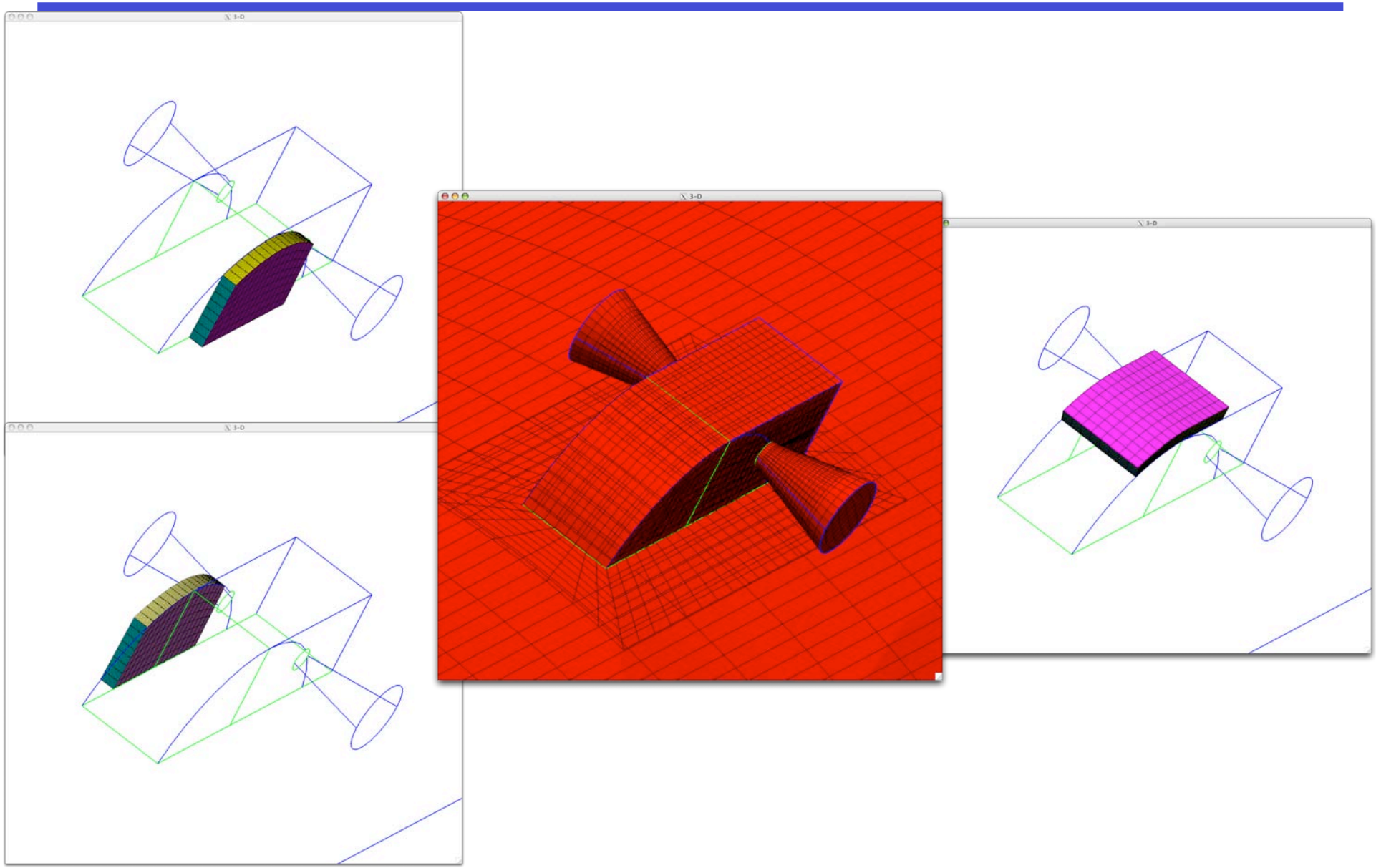


63 feature-tree “primitive” nodes
62 feature-tree “boolean” nodes
505 edges, 202 faces

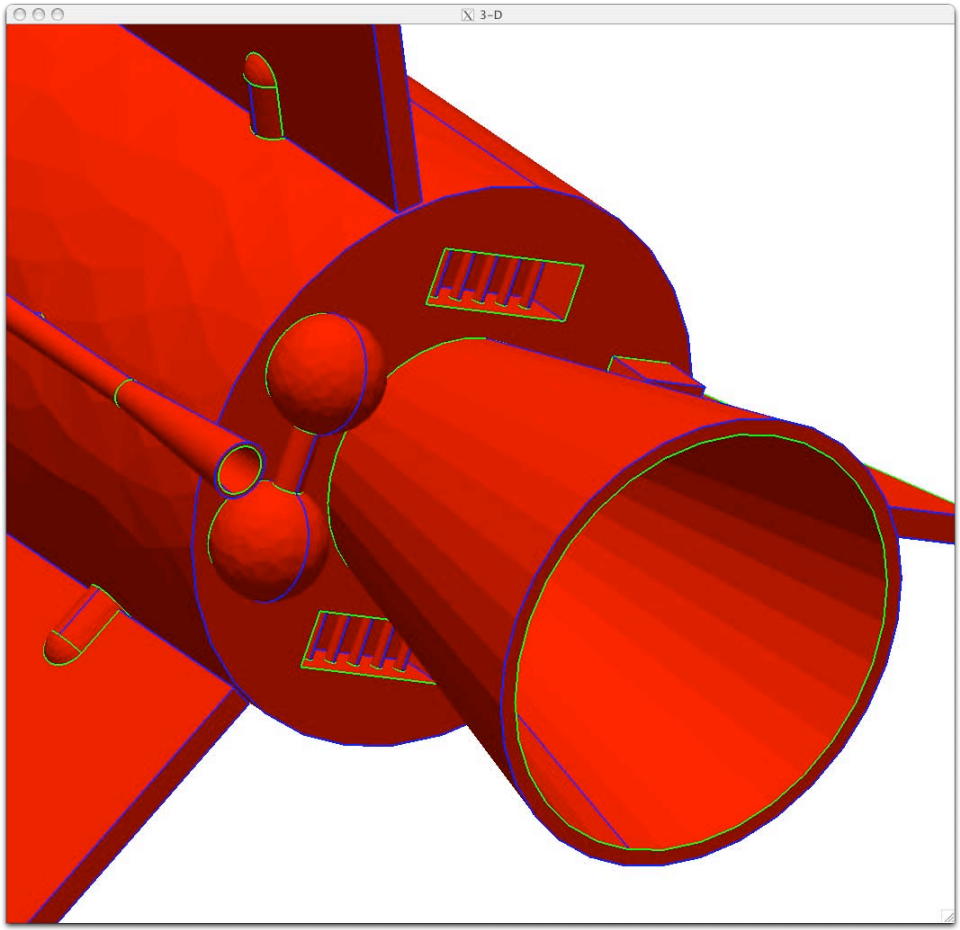
JMR3: Collar Grids Near Thruster 2

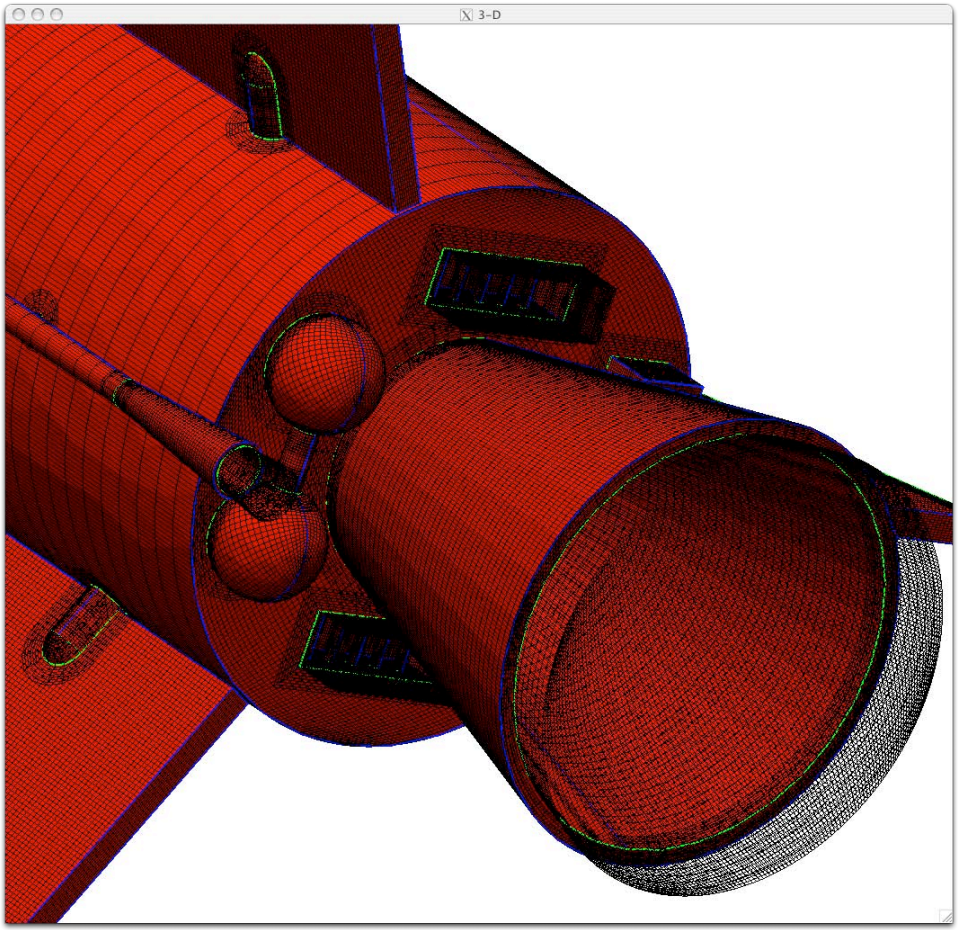


JMR3: Collar Grids Near Thruster 2

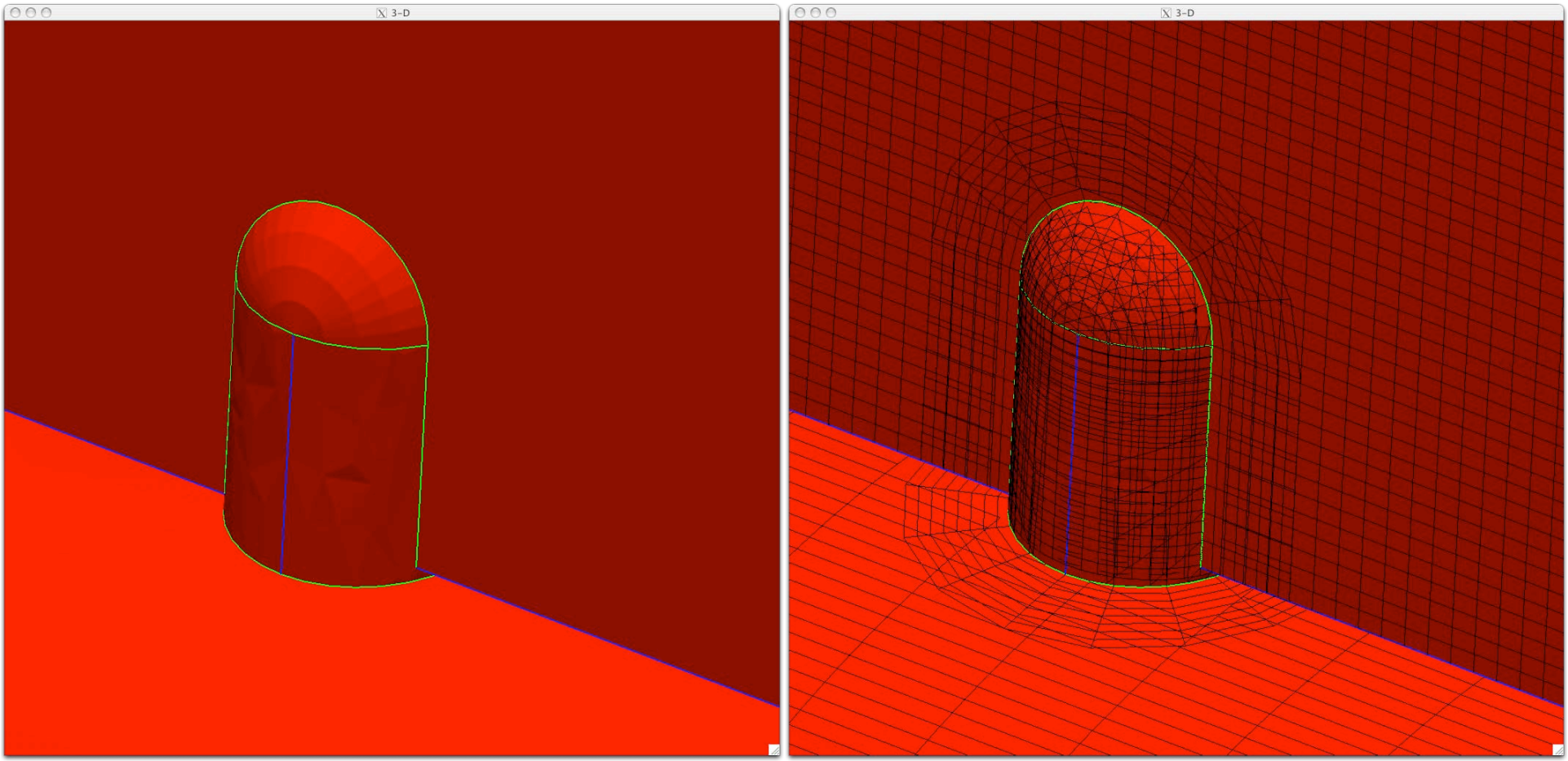


JMR3: Configuration and Grids Near Base





JMR3: Configuration and Grids Near Fin



JMR3: Grid Summary

Number of “basic” grids	136
Number of “inactive” faces	76
Number of “collar” grids	116
Number of “surface” points	269,133 (2%)
Number of “field” points	7,130,014 (61%)
Number of “fringe” points	1,042,024 (8%)
Number of “hole” points	3,183,447 (27%)

JMR3: Timing Summary

Phase	CPU (sec on MACbook PRO)
Read feature tree	0.00
Build BRep and tessellate	16.99
Generate basic grids	11.75
Build collar grids	23.30
Cut holes	38.68
Generate interpolation stencils	N/A

Summary

- **Duality between solid models and overset grids can be exploited for "automatic" grid generation**
  - Can be executed directly from solid modeler's feature tree
  - Eliminates need for user to decompose configuration
  - Grid generation time can be significantly decreased
- **Technique naturally supports design optimization**
  - Parametric optimization (no topology change)
  - Feature optimization (with topology changes)
- **Next steps**
  - Fillets/rounds, extrusions, revolves, sweeps
  - Parametric editing and sensitivities