

Aerothermodynamic CFD Analysis

Kyle Thompson

NASA Langley Research Center

August 29, 2019

Outline

- Learning Objectives
- (very brief) CFD Fundamentals
- Simulation requirements and workflows
 - LAURA (Aerothermodynamics)
 - FUN3D (Aerodynamics)
- Upcoming features (if we have time):
 - Radiative Heating Uncertainty Quantification
 - Sketch-to-Solution

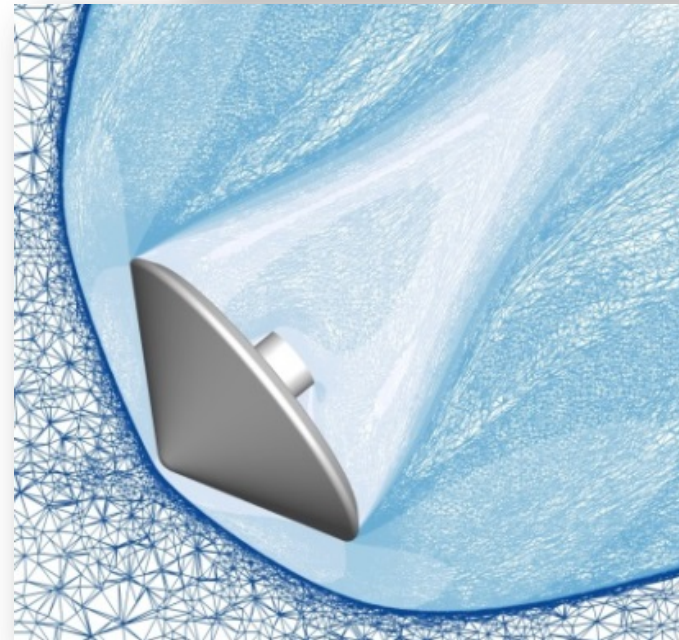
Learning Objectives

- We will cover:
 - How to request LAURA or FUN3D
 - Required/optional input files for LAURA and FUN3D
 - Where to find additional documentation for both codes
 - Troubleshooting
- We will not cover:
 - How to install LAURA or FUN3D (covered in users manuals)
 - All options and best practices (covered in users manuals and tutorials)
 - Optimal meshing strategies (covered in previous presentation)

CFD Fundamentals

- Computational Fluid Dynamics
 - Consists of conservation laws
 - Navier-Stokes equations
 - Global flux balance of mass, momentum, and energy
- Equations can't be solved analytically, so algorithms developed to compute solutions numerically
 - Finite-volume method
 - Finite-difference method
 - Finite-element method
 - Many others

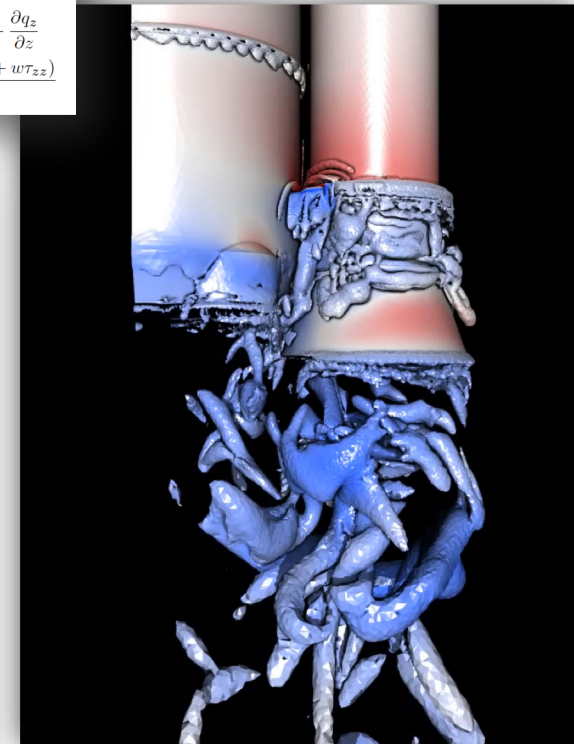
*Courtesy Beth Lee-Rausch,
Bob Biedron, and Bil Kleb*



$$\begin{aligned} \text{Mass Conservation :} \\ \frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} + \frac{\partial (\rho w)}{\partial z} = 0 \\ \text{Momentum Conservation :} \\ \frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u^2)}{\partial x} + \frac{\partial (\rho uv)}{\partial y} + \frac{\partial (\rho uw)}{\partial z} = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} \\ \frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho vu)}{\partial x} + \frac{\partial (\rho v^2)}{\partial y} + \frac{\partial (\rho vw)}{\partial z} = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z} \\ \frac{\partial (\rho w)}{\partial t} + \frac{\partial (\rho wu)}{\partial x} + \frac{\partial (\rho wv)}{\partial y} + \frac{\partial (\rho w^2)}{\partial z} = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{zx}}{\partial x} + \frac{\partial \tau_{zy}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} \\ \text{Total Energy Conservation :} \\ \frac{\partial \rho E}{\partial t} + \frac{\partial (\rho E + p) u}{\partial x} + \frac{\partial (\rho E + p) v}{\partial y} + \frac{\partial (\rho E + p) w}{\partial z} = \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} \\ + \frac{\partial (u \tau_{xx} + v \tau_{xy} + w \tau_{xz})}{\partial x} + \frac{\partial (u \tau_{yx} + v \tau_{yy} + w \tau_{yz})}{\partial y} + \frac{\partial (u \tau_{zx} + v \tau_{zy} + w \tau_{zz})}{\partial z} \end{aligned}$$



*Courtesy
Greg Brauckmann,
Steve Alter, Bil Kleb*



Reacting flows

- Navier-Stokes equations can be extended to accommodate flows involving a mixture of heterogeneous gases
 - Mixture model: collection of thermally perfect gases
 - Chemical reactions between species accounted for in source term
 - Thermodynamic nonequilibrium accounted for through tracking multiple temperatures (translational-rotational, vibrational, electronic)
- Number of equations increases from 5 to:
 - 3 momentum equations
 - number of species equations
 - number of temperatures equations
- Solving more equations increases cost of simulations

Mass Conservation :

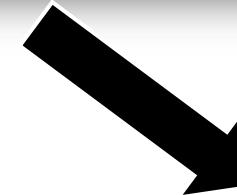
$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} + \frac{\partial (\rho w)}{\partial z} = 0$$

Momentum Conservation :

$$\begin{aligned} \frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u^2)}{\partial x} + \frac{\partial (\rho uv)}{\partial y} + \frac{\partial (\rho uw)}{\partial z} &= -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} \\ \frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho vu)}{\partial x} + \frac{\partial (\rho v^2)}{\partial y} + \frac{\partial (\rho vw)}{\partial z} &= -\frac{\partial p}{\partial y} + \frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z} \\ \frac{\partial (\rho w)}{\partial t} + \frac{\partial (\rho wu)}{\partial x} + \frac{\partial (\rho wv)}{\partial y} + \frac{\partial (\rho w^2)}{\partial z} &= -\frac{\partial p}{\partial z} + \frac{\partial \tau_{zx}}{\partial x} + \frac{\partial \tau_{zy}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} \end{aligned}$$

Total Energy Conservation :

$$\begin{aligned} \frac{\partial \rho E}{\partial t} + \frac{\partial (\rho E + p) u}{\partial x} + \frac{\partial (\rho E + p) v}{\partial y} + \frac{\partial (\rho E + p) w}{\partial z} &= \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} \\ + \frac{\partial (u \tau_{xx} + v \tau_{xy} + w \tau_{xz})}{\partial x} + \frac{\partial (u \tau_{yx} + v \tau_{yy} + w \tau_{yz})}{\partial y} + \frac{\partial (u \tau_{zx} + v \tau_{zy} + w \tau_{zz})}{\partial z} \end{aligned}$$



Species Conservation :

$$\begin{aligned} \frac{\partial \rho_1}{\partial t} + \frac{\partial (\rho_1 u)}{\partial x} + \frac{\partial (\rho_1 v)}{\partial y} + \frac{\partial (\rho_1 w)}{\partial z} &= \frac{\partial D_m}{\partial x} + \frac{\partial D_m}{\partial y} + \frac{\partial D_m}{\partial z} + \omega_1 \\ &\vdots \\ \frac{\partial \rho_N}{\partial t} + \frac{\partial (\rho_N u)}{\partial x} + \frac{\partial (\rho_N v)}{\partial y} + \frac{\partial (\rho_N w)}{\partial z} &= \frac{\partial D_m}{\partial x} + \frac{\partial D_m}{\partial y} + \frac{\partial D_m}{\partial z} + \omega_N \end{aligned}$$

Mixture Momentum Conservation :

$$\begin{aligned} \frac{\partial (\rho u)}{\partial t} + \frac{\partial (\rho u^2)}{\partial x} + \frac{\partial (\rho uv)}{\partial y} + \frac{\partial (\rho uw)}{\partial z} &= -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} \\ \frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho vu)}{\partial x} + \frac{\partial (\rho v^2)}{\partial y} + \frac{\partial (\rho vw)}{\partial z} &= -\frac{\partial p}{\partial y} + \frac{\partial \tau_{yx}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z} \\ \frac{\partial (\rho w)}{\partial t} + \frac{\partial (\rho wu)}{\partial x} + \frac{\partial (\rho wv)}{\partial y} + \frac{\partial (\rho w^2)}{\partial z} &= -\frac{\partial p}{\partial z} + \frac{\partial \tau_{zx}}{\partial x} + \frac{\partial \tau_{zy}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} \end{aligned}$$

Total Energy Conservation :

$$\begin{aligned} \frac{\partial \rho E}{\partial t} + \frac{\partial (\rho E + p) u}{\partial x} + \frac{\partial (\rho E + p) v}{\partial y} + \frac{\partial (\rho E + p) w}{\partial z} &= \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} \\ + \frac{\partial (u \tau_{xx} + v \tau_{xy} + w \tau_{xz})}{\partial x} + \frac{\partial (u \tau_{yx} + v \tau_{yy} + w \tau_{yz})}{\partial y} + \frac{\partial (u \tau_{zx} + v \tau_{zy} + w \tau_{zz})}{\partial z} \\ &+ \frac{\partial D_e}{\partial x} + \frac{\partial D_e}{\partial y} + \frac{\partial D_e}{\partial z} - Q_{rad} \end{aligned}$$

Vibrational Energy Conservation :

$$\begin{aligned} \frac{\partial \rho E}{\partial t} + \frac{\partial (\rho e_v) u}{\partial x} + \frac{\partial (\rho e_v) v}{\partial y} + \frac{\partial (\rho e_v) w}{\partial z} &= \frac{\partial q_x^v}{\partial x} + \frac{\partial q_y^v}{\partial y} + \frac{\partial q_z^v}{\partial z} \\ + \frac{\partial (u \tau_{xx} + v \tau_{xy} + w \tau_{xz})}{\partial x} + \frac{\partial (u \tau_{yx} + v \tau_{yy} + w \tau_{yz})}{\partial y} + \frac{\partial (u \tau_{zx} + v \tau_{zy} + w \tau_{zz})}{\partial z} \\ &+ \frac{\partial D_v}{\partial x} + \frac{\partial D_v}{\partial y} + \frac{\partial D_v}{\partial z} + \omega_v \end{aligned}$$

LAURA



- LAURA = Langley Aerothermodynamic Upwind Relaxation Algorithm
 - Upwind = finite-volume method using upwinding flux method
 - Relaxation = point-implicit relaxation algorithm used as core nonlinear solver
- Numerics:
 - 2nd-order finite-volume method
 - Roe Flux Difference Splitting method
 - Symmetric Total Variation Diminishing (STVD) method with minmod limiter
- Designed for Entry, Descent, and Landing applications
 - Finite-rate chemistry
 - 2-temperature model (translation-rotational and vibration temperatures)

How do I request LAURA?

- NASA Software Catalog:

<https://software.nasa.gov/>



The screenshot shows the NASA Software Catalog homepage. At the top is a dark blue header with the NASA Technology Transfer Program logo and the tagline "BRINGING NASA TECHNOLOGY DOWN TO EARTH". Below this is a large banner with the text "NASA SOFTWARE" in white and blue, accompanied by a background image of a person in a white protective suit holding a stack of software manuals. To the right of the banner, there is a snippet of code in a monospaced font.

Below the banner, a light blue section contains the text "972 AVAILABLE PROGRAMS AND COUNTING, CHECK OUT [RECENT RELEASES](#)." followed by a "Featured Software" section. This section features a video player with a play button and the text "Download NASA Software for Free!". Below the video player is a search bar with the text "Search for NASA software" and a "Search" button. To the right of the search bar is a link to "Advanced Search".

On the right side of the page, there is a section titled "Introducing the 2019-2020 NASA Software Catalog" with a video player and a description: "The 2019-2020 NASA Software Catalog offers hundreds of new software programs you can download for free to use in a wide variety of technical applications." Below this is a link to "Download the PDF" and a note: "Documents in PDF format require the [Adobe Acrobat Reader](#)."

Input files for LAURA

- Red = must have in working directory.
- Blue = will use copy in working directory if present; else will use copy in [install-prefix]/share/physics_modules.*
- Green = required in working directory if special options are requested.
- *If you receive an error message that a file is not found, then it is likely that the installation is flawed. You can copy the file from PHYSICS_MODULES into the working directory.

Filename	Content
REQUIRED FILES:	
laura.g*	PLOT3D grid
laura_bound_data	Grid block face boundary conditions
laura_namelist_data	Simulation configuration
tdata	Gas model
SIMULATION DEPENDENT FILES:	
assign_tasks	Sweep and relaxation directions
hara_namelist_data	Radiation mechanisms
jdata	Jet chamber conditions
kinetic_data	Specie reactants and products
laura.rst†	Flowfield solution for restart
laura.trn	Transition location and length
laura_trajectory_data	Trajectory points
laura_vis_data	Viscous term treatment
species_thermo_data	Specie thermodynamics
species_transp_data	Collision cross-sections
species_transp_data0	High-order collision cross-sections
surface_property_data	Thermochemical surface properties

LAURA Users Manual

- Distributed with LAURA source code

NASA/TM-2013-217800 (Corrected Copy)

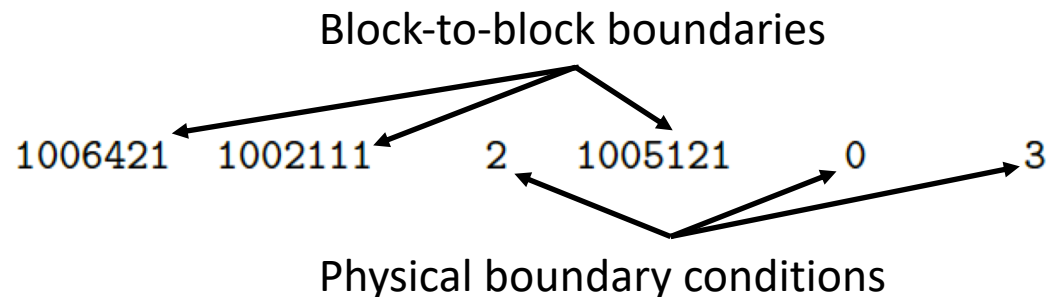


LAURA Users Manual: 5.5-65135

*Alireza Mazaheri, Peter A. Gnoffo, Christopher O. Johnston, and Bil Kleb
Langley Research Center, Hampton, Virginia*

Boundary Conditions – **laura_bound_data**

- Recommend running bounds utility to generate **laura_bound_data**
- Details in manual Sec. 5.3
- Sample line in **laura_bound_data** that defines block face boundaries



-9,...,0: Solid surface boundary. Up to ten different solid surface boundaries may be specified. Thermochemical properties of solid surfaces that are different than type 0, which are specified in `laura_namelist_data`, are defined in `surface_property_data` file—see Section 5.18 on page 55.

1: Outflow boundary (extrapolation).

2: Symmetry boundary across $y = \text{constant}$.

3: Farfield/Freestream boundary.

4: Symmetry boundary across $x = \text{constant}$ or $z = \text{constant}$.

5: Reflection boundary across $j = 1$ symmetry (valid for axi-symmetric and/or 2D grids).

6: Venting boundary. (See Section 5.4.18 on page 37 for more details.)

7: Reflection boundary across i face singularity with periodic j boundary.

8: Characteristic (also known as subsonic) boundary condition.

Flow Conditions/Numerics – **laura_namelist_data**

- Flow conditions and numerics options set in **laura_namelist_data**
 - Fortran 95 namelist syntax
- Freestream flow conditions expected in SI units
- Details in manual section 5.4

```
&laura_namelist
  velocity_ref = 5000.0 ! Free stream velocity, m/s
  density_ref  = 0.023  ! reference density, kg/s
  tref         = 250.0  ! Free stream temperature, K
  alpha        = 25.0   ! Angle-of-attack in xz plane, degrees
[ variable    = value  ! Optional comment ]
/
```


Thermochemical Model - **tdata**

- Details in manual section 5.5
 - LAURA can accommodate arbitrary number of species

Ideal gas assumption
($P = \rho R T$)

```
perfect_gas
&species_properties
  gamma = 1.4
  mol_wt = 28.8
  suther1 = 0.1458205E-05
  suther2 = 110.333333
  prand = 0.72
```

/

5-species
one-temperature model

```
one
N2 .767
N
O2 .233
O
NO
```

9-species
two-temperature model

```
two
N2 .767
N
O2 .233
O
NO
O2+
O+
NO+
e-
```

free-energy minimization

```
FEM
N2 .767
N
O2 .233
O
NO
O2+
O+
NO+
e-
```

Thermodynamics Data - `species_thermo_data`

- Section 5.15 of manual
- Uses for this file:
 - Add pseudo species to describe combustion products in an approximate way to save resources.
 - Provide parametric variation to thermal relaxation models.
 - Value of $C_p(T)$ frozen at curve fit extrema is not an acceptable option.

```
N
&species_properties
elec_impct_ion = 4.196 ! 14.53 ! 4.196 from excited state
sigma = 5.e-20, 0., 0.
mol_wt = 14.
/
3
0.00000000E+00 0.00000000E+00 0.25000000E+01 0.00000000E+00
0.00000000E+00 0.00000000E+00 0.00000000E+00 0.00000000E+00
0.56104638E+05 0.41939093E+01 200.000 1000.000
0.88765014E+05 -0.10712315E+03 0.23621883E+01 0.29167201E-03
-0.17295151E-06 0.40126579E-10 -0.26772276E-14 0.00000000E+00
0.56973513E+05 0.48652358E+01 1000.000 6000.000
0.54751811E+09 -0.31075750E+06 0.69167827E+02 -0.68479881E-02
0.38275724E-06 -0.10983677E-10 0.12779860E-15 0.00000000E+00
0.25505856E+07 -0.58487697E+03 6000.000 20000.000
```

Collision Cross Section Data - **species_transp_data**

- Section 5.16 of manual
- Original LAURA log-linear fit between 2000K and 4000K
 - $\log_{10}(\pi\Omega^{(1,1)})$ and $\log_{10}(\pi\Omega^{(2,2)})$ in m^2
- Supplanted by high-order fit data if available (next slide)
- The majority of simulations do not require changes to this file – exceptions include:
 - Need quick look at new species transport property model.

```
Ar Ar
-14.6017 -14.6502 -14.5501 -14.6028 ! trr132+kestin et al
Ar CH4
-14.5827 -14.6319 -14.5328 -14.5806 ! kestin et al
```

Collision Cross Section Data - [species_transp_data_0](#)

- Section 5.17 of manual
- Sources
 - Wright, M.: Recommended Collision Integrals for Transport Property Computations Part 1: Air Species. AIAA J., vol. 43, no. 12, 2005, pp. 2558-2564.
 - Wright, M.: Recommended Collision Integrals for Transport Property Computations Part 2: Mars and Venus Entries. AIAA J., vol. 45, no. 1, 2005, pp. 281-288.
- Supersedes log-linear fit data if available (previous slide)
- The majority of simulations do not require changes to this file.
- The data file contains all original sources and details of the curve fits.

Chemical Kinetic Data - kinetic_data

- Section 5.10 of manual
- Advantageous to keep local copy for parametric studies involving thermochemical nonequilibrium.

```
O2 + M <=> 2O + M                                ! JTHT Vol 15 No 1 pp76-90 2001 (Ref 1)
2.000e+21 -1.50 5.936e+04
teff1 = 2
exp1 = 0.7
t_eff_min = 1000.
t_eff_max = 50000.
C = 5.0
O = 5.0
N = 5.0
H = 5.0
Si = 5.0
e- = 0.

N + e- <=> N+ + e- + e-                            ! Ref 1
2.500e+34 -3.82 1.682e+05
teff1 = 3
teff2 = 3
```

LAURA Output Files

- Solution output in PLOT3D file format
 - Flowfield -> laura.q
 - Surface -> laura_surface.q
- Integrated aerodynamic quantities in laura.aero
- Restart file and adapted grid written with “_new” suffix to avoid overwriting initial files
 - These need to be renamed as laura.* upon restart

Filename	Content	Format
laura.aero	Aerodynamic forces, moments, and coefficients	ASCII
laura.g.fvbnd	Boundary types in laura.g	FieldView™
laura.nam	Variables names in laura.q	Tecplot™
laura.q	Flowfield solution	PLOT3D
laura_blayer.dat	Wall and Boundary layer edge data	Tecplot™
laura_conv.out	Time step, CPU time, and residuals	ASCII
laura_new.g	Volume grid	PLOT3D
laura_new.rst	Flowfield solution for restart	Binary
laura_surface.g	Surface grid	PLOT3D
laura_surface.nam	Variables names in laura_surface.q	Tecplot™
laura_surface.q	Surface solution	PLOT3D

LAURA Nondimensionalization

- BEWARE! Some output in laura.q is nondimensionalized (section 6.4)
 - If the *.nam file does not have units attached to the variable name, verify the normalization in the manual

Variables	Definition	Normalized by
$c_{N,N2,...}$	Species mass fraction	-
u, v, w	Velocity components	U_∞
E	Total energy per unit mass	U_∞^2
e_j	Energy mode j per unit mass	U_∞^2
T	Translational temperature	-
T_v	Vibrational temperature	-
ρ	Density	ρ_∞
M_w	Molecular weight	-
p	Pressure	$\rho_\infty U_\infty^2$
c	Sonic velocity	U_∞
e	Static energy per unit mass	U_∞^2
e_v	Vibrational energy per unit mass	U_∞^2

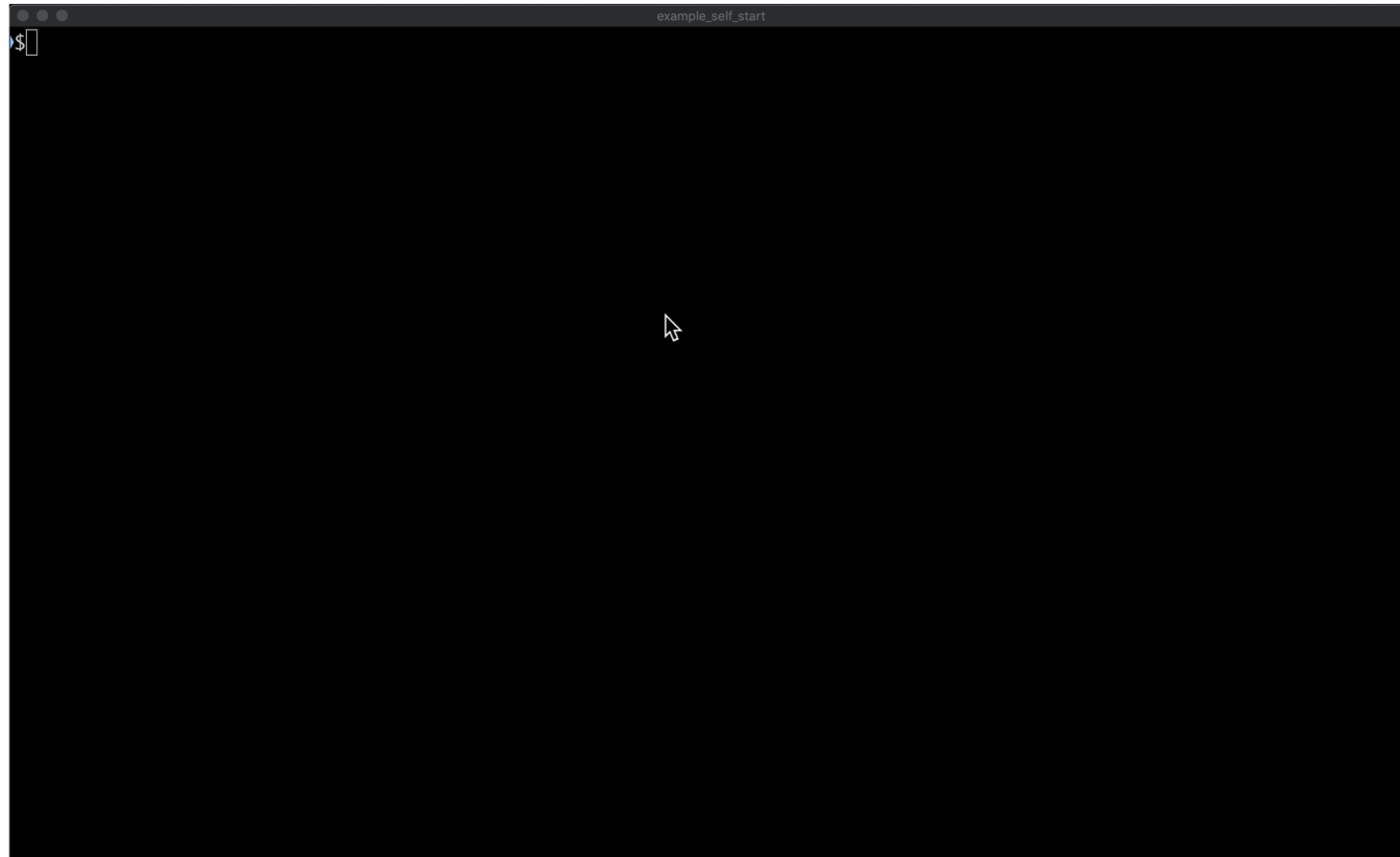
Getting started with LAURA

- LAURA comes with `self_start` utility
 - Generates structured grid for parameterized conic geometries
 - Generates input file (`laura_namelist_data`)
 - Generates boundary conditions file (`laura_bounds_data`)
- `self_start` prompts user for grid inputs
 - User is responsible for modifying `laura_namelist_data` for inputs prior to running `laura_mpi`
- **Demo:**
 - **compute surface heating and pressure on 1m radius axisymmetric hemisphere mesh**

LAURA Workflow

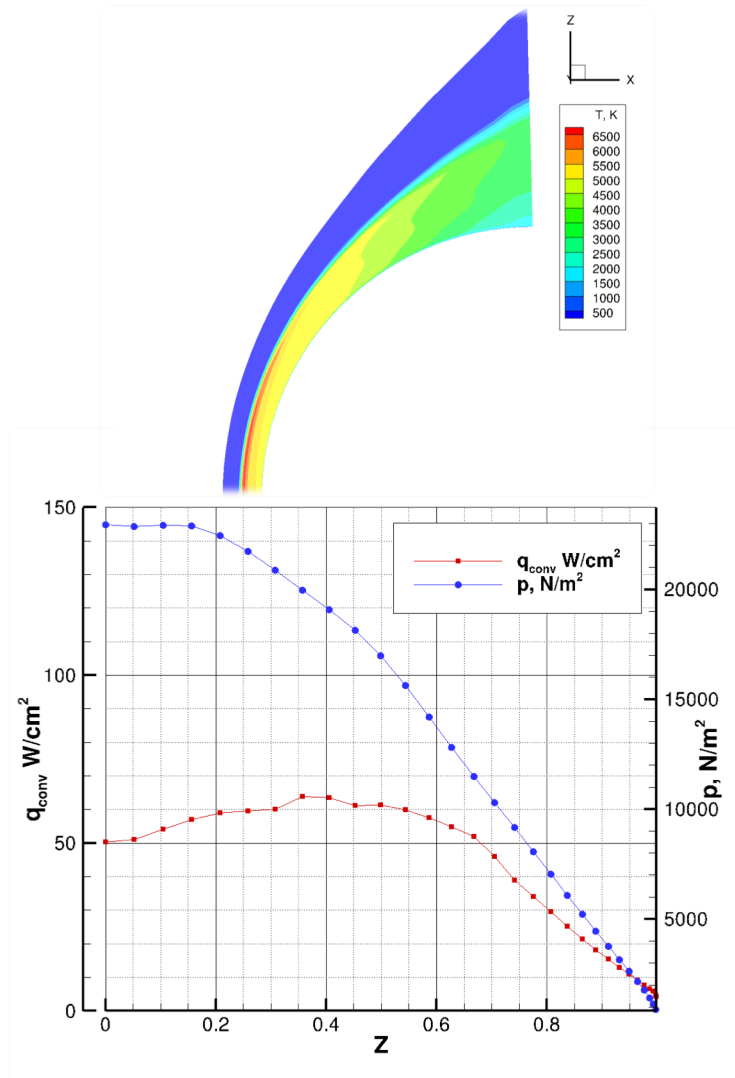
1. Run `self_start` utility -> `laura.g`, `laura_namelist_data`, `laura_bounds_data`
2. Set conditions -> `laura_namelist_data`
3. Set gas model -> `tdata`
4. Run `laura_mpi`
5. Visualize volume solution -> `laura.q`, `laura_new.g`, `laura.nam`
6. Visualize surface solution -> `laura_surface.q`
7. Restart -> `laura_new.g`, `laura_new.rst`
8. Verify convergence (Both error norms and heating)

LAURA Workflow

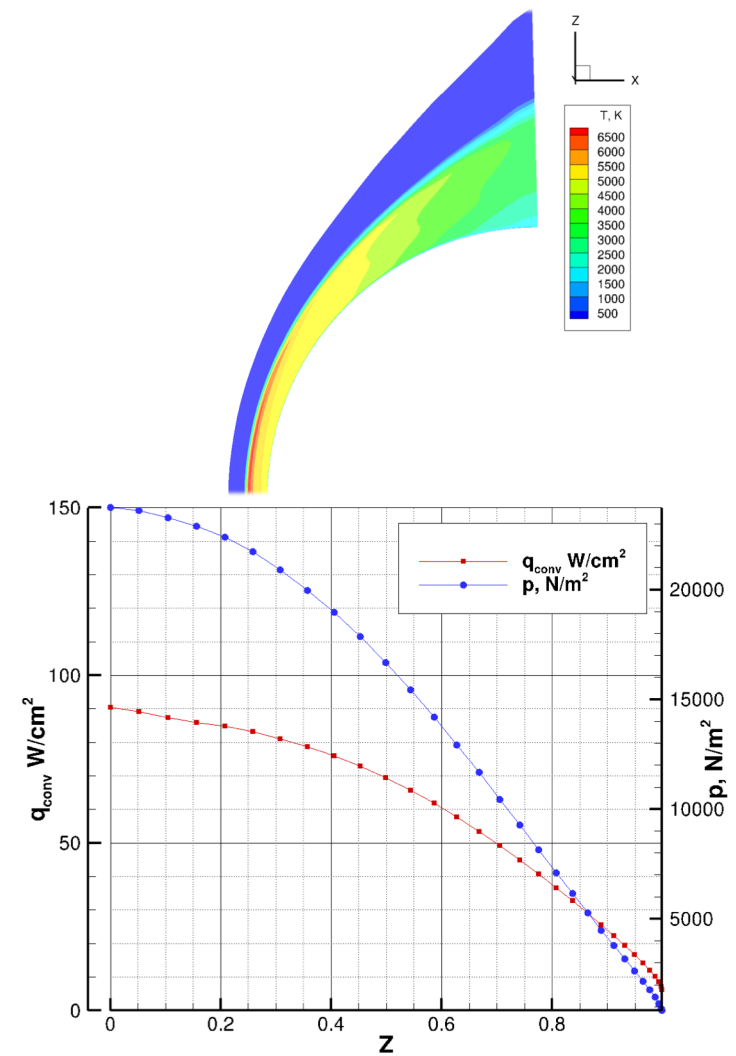


Effect of Grid Adaptation on Solution

Initial Adaptation



Final Adaptation



Going further with LAURA

- More sample cases included in section 8 of manual

8	Sample Cases	68
8.1	Sphere: 5-species Air, Thermo-chemical Nonequilibrium	68
8.2	Coupled radiation procedure	73
8.3	Unspecified ablation procedure - Coupled	74
8.4	Unspecified ablation procedure - Uncoupled	75

- Radiation options for HARA in section 5.8 of manual
- LAURA utilities (section 7)
 - coarsen/refine structured meshes with coarsen/prologate utilities

FUN3D



- Supports mixed element grid
 - Supported element types: tetrahedra, prisms, pyramids, hexahedra
- 2nd-order node-centered finite-volume algorithm
 - Wide variety of flux functions (Roe, Vanleer, LDFSS, etc.)
 - Supports automated mesh adaptation with Refine
 - Supports variety of turbulence models (SA, SST, etc.)
- Hypersonic path of FUN3D designed for Entry, Descent, and Landing applications, based on LAURA physics libraries
 - Finite-rate chemistry
 - 2-temperature model (translation-rotational and vibration temperatures)

FUN3D Website

- <https://fun3d.larc.nasa.gov>
- Details
 - FUN3D Applications
 - Training materials
 - Users manual
 - Code request forms (when NASA software catalog is down)

The image shows a screenshot of the FUN3D website and a separate image of the FUN3D Manual: 13.5 cover. The website header features the NASA logo and the FUN3D logo with the text "Fully Unstructured Navier-Stokes". Below the header, it states "Current Release: 13.5-d36c511" and "FUN3D is a Computational Fluid Dynamics (CFD) suite of tools actively developed at NASA that benefits Aeronautics, Space Technology, and Exploration by modeling fluid flow." The main content area is divided into two columns. The left column contains a "SITE CONTENTS" menu with sections for "SOFTWARE OVERVIEW AND REFERENCES" and "TRAINING WORKSHOPS". The right column displays "Up 1 Next (C2: Reference Information)" and "1. OVERVIEW AND GETTING STARTED", with a sub-section "1.1. BACKGROUND". The background section text describes the history of FUN3D, its purpose, and its applications. Below the website screenshot is the cover of the "FUN3D Manual: 13.5". The cover features the NASA logo, the title "FUN3D Manual: 13.5", and a list of authors: Robert T. Biedron, Jan-René Carlson, Joseph M. Derlaga, Peter A. Gnoffo, Dana P. Hammond, William T. Jones, Bil Kleb, Elizabeth M. Lee-Rausch, Eric J. Nielsen, Michael A. Park, Christopher L. Rumsey, James L. Thomas, Kyle B. Thompson, and William A. Wood. The cover also mentions "Langley Research Center, Hampton, Virginia".

Website Header:

- NASA logo
- FUN3D logo: Fully Unstructured Navier-Stokes
- Current Release: 13.5-d36c511
- FUN3D is a Computational Fluid Dynamics (CFD) suite of tools actively developed at NASA that benefits Aeronautics, Space Technology, and Exploration by modeling fluid flow.

Website Content:

- SITE CONTENTS**
- SOFTWARE OVERVIEW AND REFERENCES**
- 1. OVERVIEW AND GETTING STARTED**
- 1.1. BACKGROUND**

Manual Cover:

- NASA logo
- NASA/TM-2019-220271
- FUN3D Manual: 13.5**
- Authors: Robert T. Biedron, Jan-René Carlson, Joseph M. Derlaga, Peter A. Gnoffo, Dana P. Hammond, William T. Jones, Bil Kleb, Elizabeth M. Lee-Rausch, Eric J. Nielsen, Michael A. Park, Christopher L. Rumsey, James L. Thomas, Kyle B. Thompson, and William A. Wood
- Langley Research Center, Hampton, Virginia

Additional Reference Material

- U.S. persons who have been approved to receive the high-energy-physics options in FUN3D may also obtain NASA/TM-2014-218658 for detailed examples.

NASA/TM-2014-218658



A Tutorial for the Generic Gas Path in Fun3D

*Peter A. Gnoffo, William A. Wood, William L. Kleb, Stephen J. Alter and Christopher E. Glass
Langley Research Center, Hampton, Virginia*

Contents

1	Introduction	7
2	Fun3D Input Overview	8
3	Fun3D Input: fun3d.nml	8
3.1	Project Namelist	10
3.2	Raw Grid Namelist	11
3.3	Governing Equations Namelist	11
3.4	Reference Properties Namelist	12
3.5	Force and Moment Integration Namelist	13
3.6	Inviscid Flux Namelist	13
3.6.1	<i>Inviscid Flux Reconstruction Algorithms</i>	14
3.6.2	<i>Eigenvalue Limiting</i>	15
3.6.3	<i>First Order Iterations</i>	16
3.6.4	<i>First or Second Order Turbulent Flux</i>	16
3.6.5	<i>Pole Gradient</i>	17
3.7	Non-linear Solver Namelist	17
3.8	Linear Solver Namelist	18
3.9	Boundary Conditions Namelist	18
3.10	Turbulent Diffusion Model Namelist	19
3.11	Output Boundary Namelist	20
3.12	Grid Adaptation Namelist	21
4	Fun3D Input: tdata	22
4.1	Perfect Gas	22
4.2	Equilibrium Gas	23
4.3	Mixture of Thermally Perfect Gases	23
5	Fun3D Input: <i>project_root_name.mapbc</i>	24

6	Test Examples	25
6.1	Preliminaries	25
6.2	Perfect Gas	26
6.2.1	air_pg_tlms_5kps_axi_sphere	26
6.2.2	air_pg_tlms_5kps_axi_sphere_makeTets	29
6.2.3	air_pg_tlms_1kps_8deg_sphere_cone_lineadapt	31
6.2.4	air_pg_tlms_M6_axi_cev	34
6.2.5	cev-Tunnel9-Laminar	38
6.2.6	cev-Tunnel9-CS	41
6.3	Equilibrium Air	43
6.3.1	equilibrium_air-FEM	43
6.3.2	equilibrium_air-table_lookups	47
6.4	Reacting Gases	49
6.4.1	air_5sp_2t_tlms_5kps_axi_sphere	50
6.4.2	FullCEV-3km-5sp	52
6.4.3	cev-6km-5sp-1T	58
6.4.4	cev-6km-5sp-1T_makeTets	60
6.4.5	cev-10km-11sp-2t	61
6.4.6	air_11sp_2t_11kps_axi_stardust	66
6.4.7	mars_1sp_co2	68
6.4.8	titan_18sp_2T_6kps_axi_70deg_scone	71

6.5	Turbulence Modeling	76
6.5.1	turb_wilcox_2006	76
6.6	Internal Flow	79
6.6.1	Plenum_Nozzle	79
6.7	Radiation	84
6.7.1	coupled_radiation	84
6.7.2	coupled_radiation_makeTets	85
6.8	Ablation	86
6.8.1	coupled_radiation_and_blowing	87
6.8.2	coupled_ablation-computed_rates	90
6.8.2.1	<i>Solution Start-Up</i>	92
6.8.2.2	<i>Solution Progression</i>	92
6.8.2.3	<i>Solution Completion</i>	92
7	Summary	94
8	Appendix A: Convert LAURA Input to Fun3D input	95
9	Appendix B: Gas Physics Modules	96
9.1	Thermodynamics	97
9.1.1	Initialization	97
9.1.2	Thermodynamics Argument List	98
9.2	Transport Properties	102
9.2.1	Initialization	102
9.2.2	Transport Property Argument List	102
9.3	Chemical Kinetics	104
9.3.1	Initialization	104
9.3.2	Chemical Kinetics Argument List	104
9.4	Thermal Relaxation	105
9.4.1	Initialization	106
9.4.2	Thermal Relaxation Argument List	106

Governing Equations – fun3d.nml

- Detail in manual Sec. B.4.6
 - `chemical_kinetics`: 'finite-rate', 'frozen'
 - `thermal_energy_model`: 'non-equilib', 'frozen'
 - `gas_radiation`: 'off', 'uncoupled', 'coupled'
 - `rad_use_impl_lines`: .false., .true.
 - `multi_component_diff`: .false., .true.
 - `cpiv_min_factor`: 0.0001
 - `augment_kinetics_limiting`: .false., .true.
 - `implicit_rate_limiting`: .true., .false.

Boundary Conditions – fun3d.nml

- Detail in manual Sec. B.4.18

```
wall_temperature(:) = 1.  
wall_temp_flag(:) = .false.  
wall_radeq_flag(:) = .false.  
wall_emissivity(:) = 0.8  
wall_emissivity_b(:) = 0.  
wall_emissivity_c(:) = 0.  
wall_emissivity_d(:) = 0.  
wall_temp_relax(:) = 0.001  
wall_catalysis_model(:) = 'super-catalytic'  
catalytic_efficiency_o(:) = 0.  
catalytic_efficiency_n(:) = 0.  
plenum_t0(:) = 1000.  
plenum_p0(:) = 1000.  
plenum_id(:) = 0  
fixed_in_id(:) = 0  
fixed_in_rho(:) = 0.  
fixed_in_uvw(:,1:3) = 0.  
fixed_in_t(:) = 0.  
fixed_in_tv(:) = 0.  
fixed_in_turb(:,1:7) = 0.
```

Multiple Inflow Boundaries

tdata

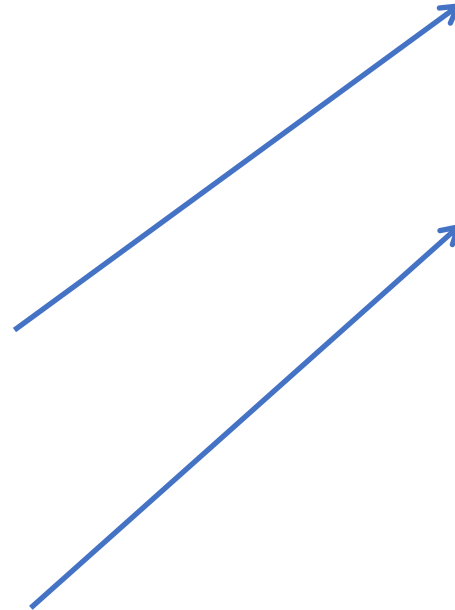
one
N2 1.0
O2
H2
O
H
OH
H2O

N2 0.7686
O2 0.2314

H2 1.0000
O 0.0000
H 0.0000
OH 0.0000
H2O 0.0000

fun3d.nml

&boundary_conditions
 wall_temperature(1) = 1305.0
 wall_temp_flag(1) = .true.
 wall_catalysis_model(1) = 'non-catalytic'
 plenum_id(2) = 2
 plenum_t0(2) = 1000.0
 plenum_p0(2) = 3800.0
 .
 .
 plenum_id(6) = 3
 plenum_t0(6) = 900.0
 plenum_p0(6) = 16000.0



Inviscid_flux_method – fun3d.nml

- Detail in manual Sec. B.4.9

```
&inviscid_flux_method  
flux_construction = 'stvd' ! 'roe' ! 'multidm'  
flux_limiter = 'minmod_gg' ! 'hvanalbada'  
gen_turb_advectn_ho = .false.
```

```
multidm_option = 1  
fixed_direction = .true.
```

Some improvement of heating on tetrahedral grids.

```
adptv_entropy_fix = .true.  
rhs_u_eigenvalue_coef = 0.5  
lhs_u_eigenvalue_coef = 1.0  
rhs_a_eigenvalue_coef = 0.5  
lhs_a_eigenvalue_coef = 1.0
```

Eigenvalue limiting controls to suppress carbuncle but enable accurate heating on prismatic grids.

```
re_min_vswch = 5.  
re_max_vswch = 50.
```

Eigenvalue limiting in boundary layer. Typically smaller values on startup

```
pole_gradient = .false.  
/
```

Turbulent Diffusion Model – fun3d.nml

- Detail in manual Sec. B.4.10
- Algebraic models available if implicit lines span domain.
 - *baldwin-lomax*
 - *cebeci-smith*
- Multi-equation models available in the generic gas path
 - *sa-catris*
 - *kw-sst*
 - *kw-sst2003*
 - *wilcox-kw98*
 - *wilcox-kw06*

Boundary Output Variables– fun3d.nml

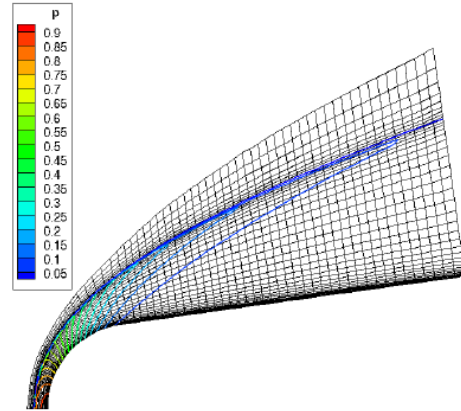
- Detail in manual Sec. B.4.27
- The units of convective heating (*heating*) and radiative heating (*q_rad*) are W/cm².
- The units of shear in the x, y, and z directions (*shear_x*, *shear_y*, and *shear_z*, respectively) are in N/m².
- The units of temperature (*tt*, *tv*) are in K.
- All other variables are non-dimensionalized by reference density, velocity, and length - consistent with units used to demonstrate the Mach number independence principle.

Variable	Non-dimensionalization
ρ	ρ_{ref}
u	V_{ref}
v	V_{ref}
w	V_{ref}
a	V_{ref}
p	$(\rho_{ref} V_{ref}^2)$
e	V_{ref}^2
h	V_{ref}^2
μ	$(\rho_{ref} V_{ref} L_{ref})$

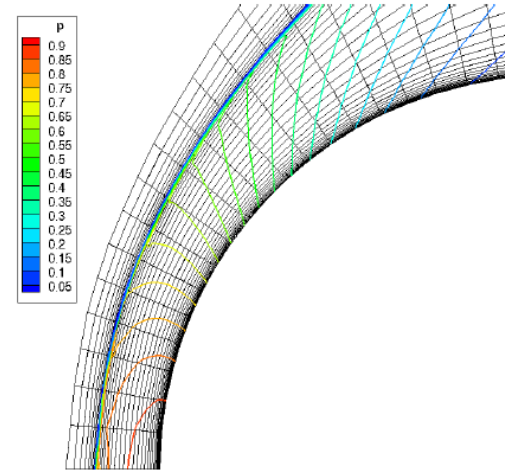
Grid Adaptation – fun3d.nml

- Detail in manual Sec. B.4.34
- Line adaptation available if implicit lines span domain.

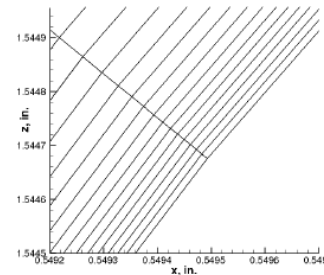
```
&adapt_mechanics  
ladapt_fsh = 0.8  
ladapt_fstr = 0.75  
ladapt_fctrjmp = 1.05  
ladapt_re_cell = 1.  
ladapt_beta_grd = 0.  
ladapt_ep0_grd = 0.  
ladapt_jumpflag = 2  
ladapt_freq = 200  
ladapt_max = 1000  
/
```



(a) Global view.



(b) Detailed view.



(c) Near wall view.

Getting started with FUN3D

- FUN3D supports variety of grid formats (AFLR3, CGNS, VGRID, etc.)
 - No grid generation software distributed with FUN3D
- See *doc/quick_start* included in distribution directory to verify perfect-gas simulation is working
- **Demo:**
 - **compute surface pressure on unstructured ONERA M6 wing mesh**

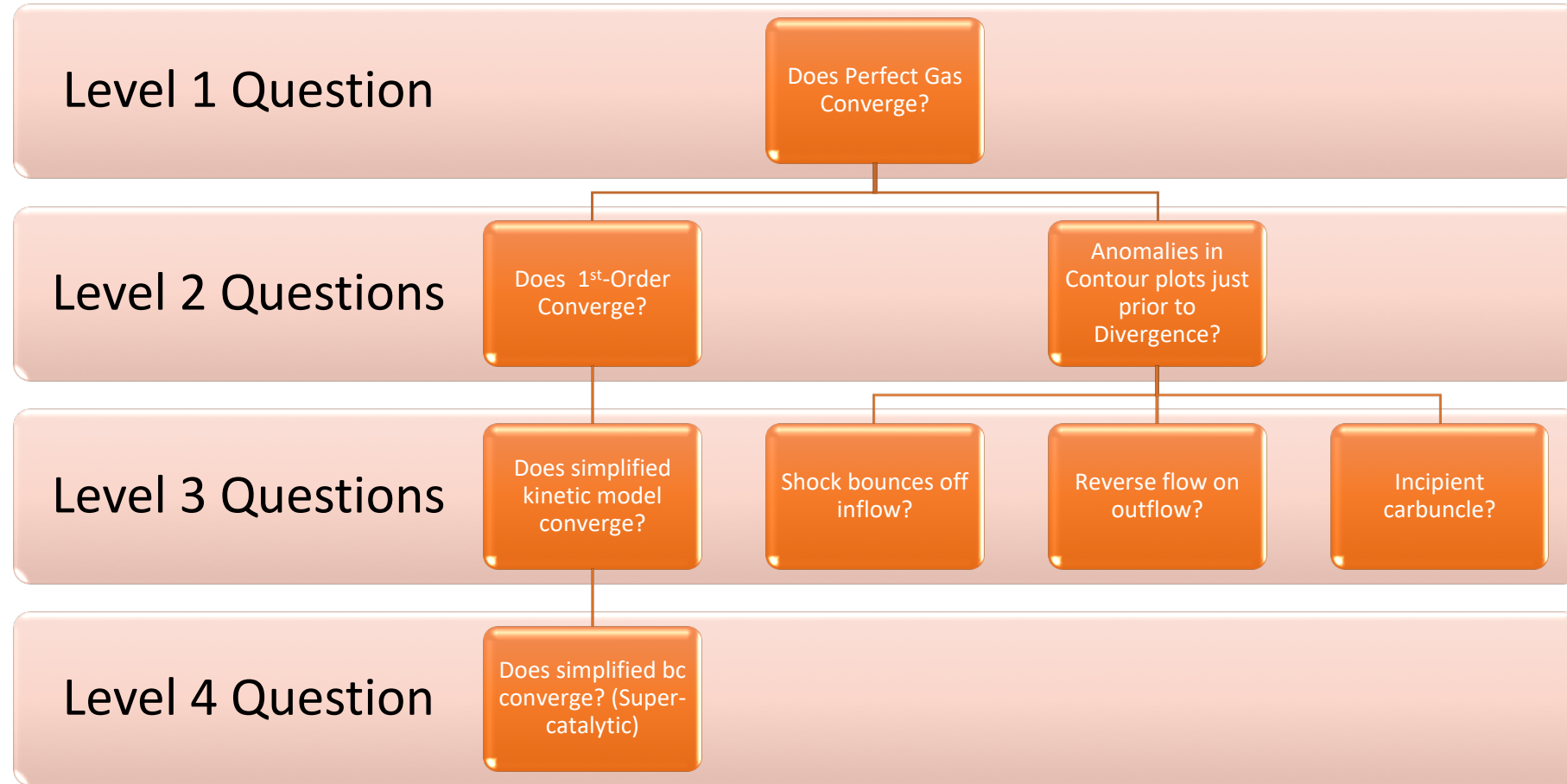
Format	Grid files	BC File
AFLR3	.ugrid	.mapbc
FAST	.fgrid	.mapbc
FieldView	.fvgrid_fmt	.mapbc
	.fvgrid_unf	.mapbc
FUN2D	.faces	.mapbc
VGRID	.cogsg, .bc	.mapbc*
FELISA	.gri, .fro	.bco

* Same suffix, but GridTool format.

```
$ls
fun3d.nml
inv_wing.fgrid
inv_wing.mapbc
inv_wing_hist.lay
inv_wing_tec_boundary.lay
$
```

I

Troubleshooting



Possible Remedies

- Level 0
 - Consult NASA-TM-2014-218658 for a test case most like your case for guidance in selecting namelist parameters.
- Level 1
 - If perfect gas fails to converge, then there is likely a problem with grid quality or boundary conditions. Capturing surface contour plots, contour plots in the symmetry plane, or contour plots in cut planes through “challenging” topological features will often reveal problems with the grid or boundary conditions. Plot streamlines to confirm consistency with boundary conditions.
- Level 2
 - If first-order fails to converge and Level 1 checks pass, then the problem is likely one or more of the following conditions: overaggressive cfl ramping, inadequate eigenvalue limiting, temperature transient causes instability in the kinetic model.
- Level 3
 - If simplified kinetic model works where more complete model fails then additional limiting on max / min temperatures may be required for offending reactions. Proceed with caution that real physics isn't compromised!
- Level 4
 - If simple (Dirichlet) bc converges, then a more conservative (smaller) relaxation factor may be required for the radiative equilibrium wall boundary (wall_temp_relax(:)).

Warnings

64 0.226705883950811E-05 0.80561E-05 0.2348E-01 0.5379E-02 0.1232E+00 3
Lift -0.189740078163752E-02 Drag 0.679775300136714E-02
INFO: Temperature below 1.0000 4 times. Reset. Min: 0.50000
INFO: Temperature below 1.0000 6 times. Reset. Min: 0.50000
INFO: Temperature below 1.0000 9 times. Reset. Min: 0.50000

- Such warnings (and related involving density) are not necessarily a problem.
- Usually, they are indicative of an undershoot or overshoot across a captured shock as its position evolves.
- Often, they disappear as the solution converges.
- Occasionally, they persist indicating a shortcoming of the limiter.
- If the number of occurrences keeps growing, the solution is diverging.

FUN3D Summary

- Generic gas path simulations require additional data files available in the PHYSICS_MODULES directory.
- Recommended flux reconstruction and limiting options are documented in the tutorial.
- Not all options for the compressible gas path are available for the generic gas path.
 - Not all of the turbulence models are engaged in the generic gas path.
 - Not all of the boundary conditions are engaged in the generic gas path.
 - Not all of the flux reconstruction option are engaged in the generic gas path.
 - Usually, engagement will only require accommodation of correct equation of state or speed of sound.
- Prismatic grids orthogonal to the boundary layer, free shear layers, and captured shocks are HIGHLY recommended if accurate simulation of surface heating and shear is required.
 - The *multidm* option can be engaged for an all-tetrahedral grid to improve simulation quality, but this option is not as effective as a good grid.
 - The utility *aflr3_make_tets* will convert a mixed-element grid into tetrahedra.

Radiative Heating Uncertainty Quantification (LAURA/HARA)

Advanced Ray-Tracing

JOURNAL OF SPACECRAFT AND ROCKETS



- Last year's work for assessing tangent-slab accuracy for coupled radiation resulted in ray-tracing and coupling infrastructure improvements.

— Published Feb. 27, 2018

- These improvements include:
 - More robust ray-tracing.
 - Better choice of ray directions.
 - Better memory utilization.
 - New approach for visualizing the radiation.
- This makes larger scale ray-tracing or coupled radiation problems feasible if the time per line-of-sight evaluation is reduced by a factor of ~10.

Impact of Non-Tangent-Slab Radiative Transport on Flowfield–Radiation Coupling

Christopher O. Johnston^a and Alireza Mazaheri^b
NASA Langley Research Center, Hampton, Virginia 23681

DOI: 10.2514/6.2017-1371

For coupled flowfield–radiation simulations, required for Earth entry at velocities greater than 10 km/s, tangent-slab radiative transport is shown to sufficiently model the divergence of the radiative flux (this term couples the radiative energy to the flowfield energy equations). However, as shown in previous studies, the tangent-slab approximation does not sufficiently model the radiative flux reaching the surface (i.e., the radiative heating). These conclusions are reached through the development of a detailed ray-tracing approach capable of computing both the divergence of the radiative flux and the radiative heating. This approach is orders of magnitude more computationally expensive than the tangent-slab approximation. The ability of the tangent-slab approach to accurately model the divergence of the radiative flux, but not the radiative heating, is shown to be the result of a cancellation of errors during the angular integration. The current work shows that combining the tangent-slab approximation for the divergence of the radiative flux with the ray-tracing approach for the radiative heating (as a final step) provides sufficiently accurate results (radiative heating values within 1%), in an efficient manner, for coupled flowfield–radiation problems.

Nomenclature

E_2	=	second-order exponential integral, dimensionless
E_3	=	third-order exponential integral, dimensionless
I_ν	=	radiative intensity, $W \cdot s/cm^2/sr$
$\bar{I}_\nu$	=	radiative intensity integrated over θ , $W \cdot s/cm^2/radian$
J_ν	=	emission coefficient, $W \cdot s/cm^2/sr$
N_θ	=	number of θ values for a given ϕ
$q_{rad,\nu}$	=	radiative heating, $W \cdot s/cm^2$
q_ν^+	=	shock-directed radiative flux, $W \cdot s/cm^2$
q_ν^-	=	wall-directed radiative flux, equal to $q_{rad,\nu}$ when evaluated at the surface, $W \cdot s/cm^2$
R_ν	=	blackbody radiative flux, $W \cdot s/cm^2$
$S(z)$	=	divergence of the radiative flux, W/cm^3
$S^{\phi,\theta}$	=	directionally dependent divergence of the radiative flux, $W/cm^3/sr$
$\bar{S}^{\phi}$	=	directionally dependent divergence of the radiative flux integrated over θ , $W/cm^3/rad$
s	=	distance along a ray, cm
z	=	distance along body normal line or normal grid line, cm
κ_ν	=	absorption coefficient, cm^{-1}
ν	=	frequency, s^{-1}
ϕ	=	angle defined in Fig. 1
θ	=	angle defined in Fig. 1
τ_ν	=	optical thickness, dimensionless

Subscripts

k	=	grid index for direction normal to the body
ν	=	indicates frequency dependence

Superscript

TS	=	indicates tangent slab
----	---	------------------------

I. Introduction

FOR Earth-entry velocities greater than 10 km/s, the radiative emission and absorption from the shock layer gas have a noticeable impact on the flowfield energy equations [1,2]. Because of this, accurate simulations of the flowfield temperatures and species number densities, and the resulting radiative heating, require coupling between the radiative transfer and flowfield computations. The radiative energy source term, or divergence of the radiative flux, couples the radiative energy to the flowfield energy equation [3]. This term represents the difference between the emitted and absorbed energy at a point in the flow. While the emitted energy (in a given direction) is simply the emission coefficient, which depends only on the local conditions, the absorbed energy depends on the incoming radiative intensity, which is a function of the surrounding flowfield. The component of the divergence of the radiative flux resulting from a direction defined in Fig. 1 by ϕ and θ is written for a point z in the flowfield as follows:

$$S^{\phi,\theta}(z) = \int_0^\infty j_\nu(z) d\nu - \int_0^\infty \kappa_\nu(z) I_\nu(z, \phi, \theta) d\nu \quad (1)$$

where j_ν is the emission coefficient, κ_ν is the absorption coefficient, and I_ν is the incoming radiative intensity, with the ν subscript on each indicating its frequency dependence. To obtain the divergence of the radiative flux applicable to the flowfield energy equations, this term is integrated over the solid angle:

$$S(z) = \int_0^{2\pi} \int_0^\pi S^{\phi,\theta}(z) \sin \phi d\phi d\theta \quad (2)$$

The second term on the right hand side of Eq. (1) represents the primary difficulty in evaluating $S(z)$, because it requires the computation of the radiative intensity for all directions surrounding point z . In practical terms, this evaluation is difficult because of both the computational expense of computing the radiative intensity at numerous (greater than 400) ϕ and θ values for every flowfield point, as well as the requirement of a robust interpolation capability for extracting the flowfield properties required for the radiative intensity computations. In fact, to the authors' knowledge, application of this rigorous approach for the computation of $S(z)$, referred to as ray tracing, has been reported in the literature only for an Earth-entry shock layer by Elbert and Cinnella [4,5], Feldick and Modest [6,7], and Karl et al. [8], whereas Potter [9] has performed these simulations for Mars entry. For the Elbert and Cinnella study, the computational limitations at the time of their study limited them to a gray-gas radiation model, which is unable to capture the details of atomic lines. The study by Feldick and Modest showed the important result that

Presented as Paper 2017-1371 at the 55th AIAA Aerospace Sciences Meeting, Grapevine, TX, 9–13 January 2017; received 31 August 2017; revision received 21 December 2017; accepted for publication 3 January 2018; published online 26 February 2018. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the ISSN 0022-4650 (print) or 1533-6794 (online) to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

^aAerospace Engineer, Aerothermodynamics Branch; christopher.o.johnston@nasa.gov. Member AIAA.

^bAerospace Engineer, Aerothermodynamics Branch; ali.r.mazaheri@nasa.gov. Member AIAA.

Start with converged LAURA flowfield solution



Obtain LAURA surface and blayer
files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

Run HARA UQ:

User just switches laura executable with hara_uq executable

- increase # of processors
- no other user input required
- compiled with LAURA
- included in latest LAURA release
- utilizes GPUs, opacity-binning, advanced ray-tracing

Obtain LAURA surface and blayer files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

Obtain LAURA surface and blayer
files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

Flowfield may include coupled radiation and/or coupled ablation.

Obtain LAURA surface and blayer files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

Obtain LAURA surface and blayer
files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

Both LAURA flowfield and HARA radiation modeling parameters treated in analysis:

- Flowfield kinetics
- Ionization potentials
- Atomic line strengths
- Atomic line broadening
- Molecular band strengths
- Non-Boltzmann rates
- Diffusion cross sections

Obtain LAURA surface and blayer files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

Both LAURA flowfield and HARA radiation modeling parameters treated in analysis:

- Flowfield kinetics
- Ionization potentials
- Atomic line strengths
- Atomic line broadening
- Molecular band strengths
- Non-Boltzmann rates
- Diffusion cross sections

Leverages uncertainty approaches for radiation developed by Kleb et al. (JSR, No. 3, 2012) and West et al. (JTHT, No. 2, 2017).

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

- Reduce # of cases by checking each flowfield for noticeable impact. Ignore if no impact

Obtain LAURA surface and blayer files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

- Reduce # of cases by checking each flowfield for noticeable impact. Ignore if no impact

Evaluates small number of flowfield iterations for each perturbed rate to check impact.

Reducing the number of rates reduces the number of flowfield/radiation evaluations required for the uncertainty analysis.

Obtain LAURA surface and blayer files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

- Reduce # of cases by checking each flowfield for noticeable impact. Ignore if no impact

Run simultaneous flowfield cases

Case A

Case B

Case C



Loop through
all cases

Obtain LAURA surface and blayer
files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

- Reduce # of cases by checking each flowfield for noticeable impact. Ignore if no impact

Run simultaneous flowfield cases

Case A

Case B

Case C



Loop through
all cases

The MPI for the LAURA flowfield simulations is arranged within `hara_uq` to allow multiple flowfield cases to run simultaneously.

3 simultaneous cases are shown here. This number is only limited by the # of processors allocated for the job.

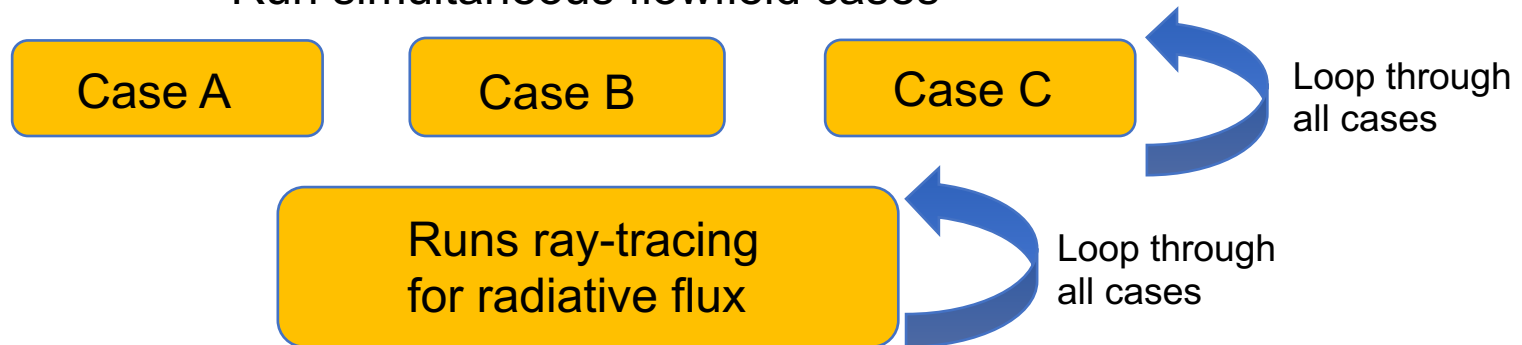
Code automatically assigns 1 processor per grid block.

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

- Reduce # of cases by checking each flowfield for noticeable impact. Ignore if no impact

Run simultaneous flowfield cases



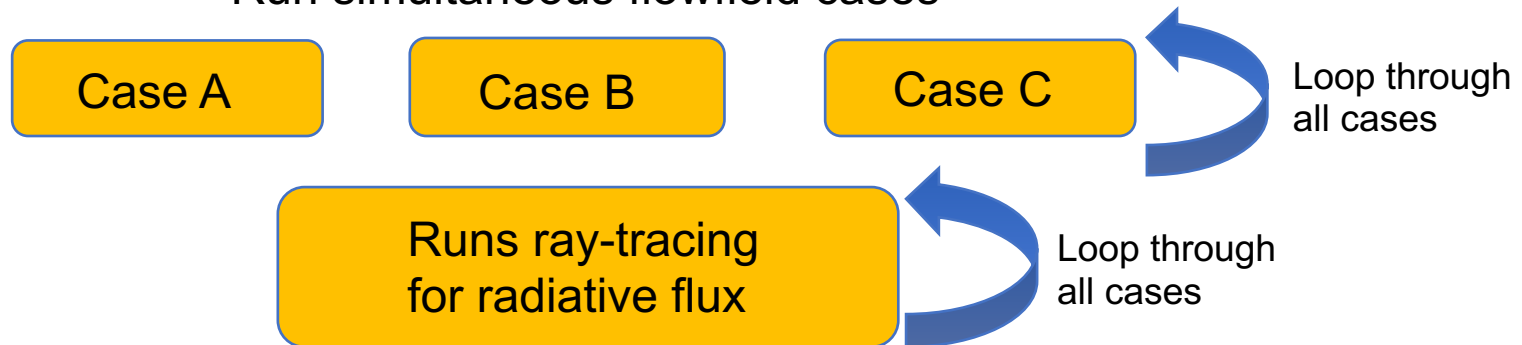
Obtain LAURA surface and blayer files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

- Reduce # of cases by checking each flowfield for noticeable impact. Ignore if no impact

Run simultaneous flowfield cases



The radiative flux over the entire surface computed using ray-tracing for all combination of parameters.

This step leverages the recent HARA advancements (GPU utilization, opacity binning, and improved ray-tracing)

**Computes
Parametric
Uncertainty**
 σ_{param}

Start with converged LAURA flowfield solution

- Find radiation modeling parameter uncertainties
- Find flowfield modeling parameter uncertainties
- Determine # of cases to run

- Reduce # of cases by checking each flowfield for noticeable impact. Ignore if no impact

Run simultaneous flowfield cases

Case A

Case B

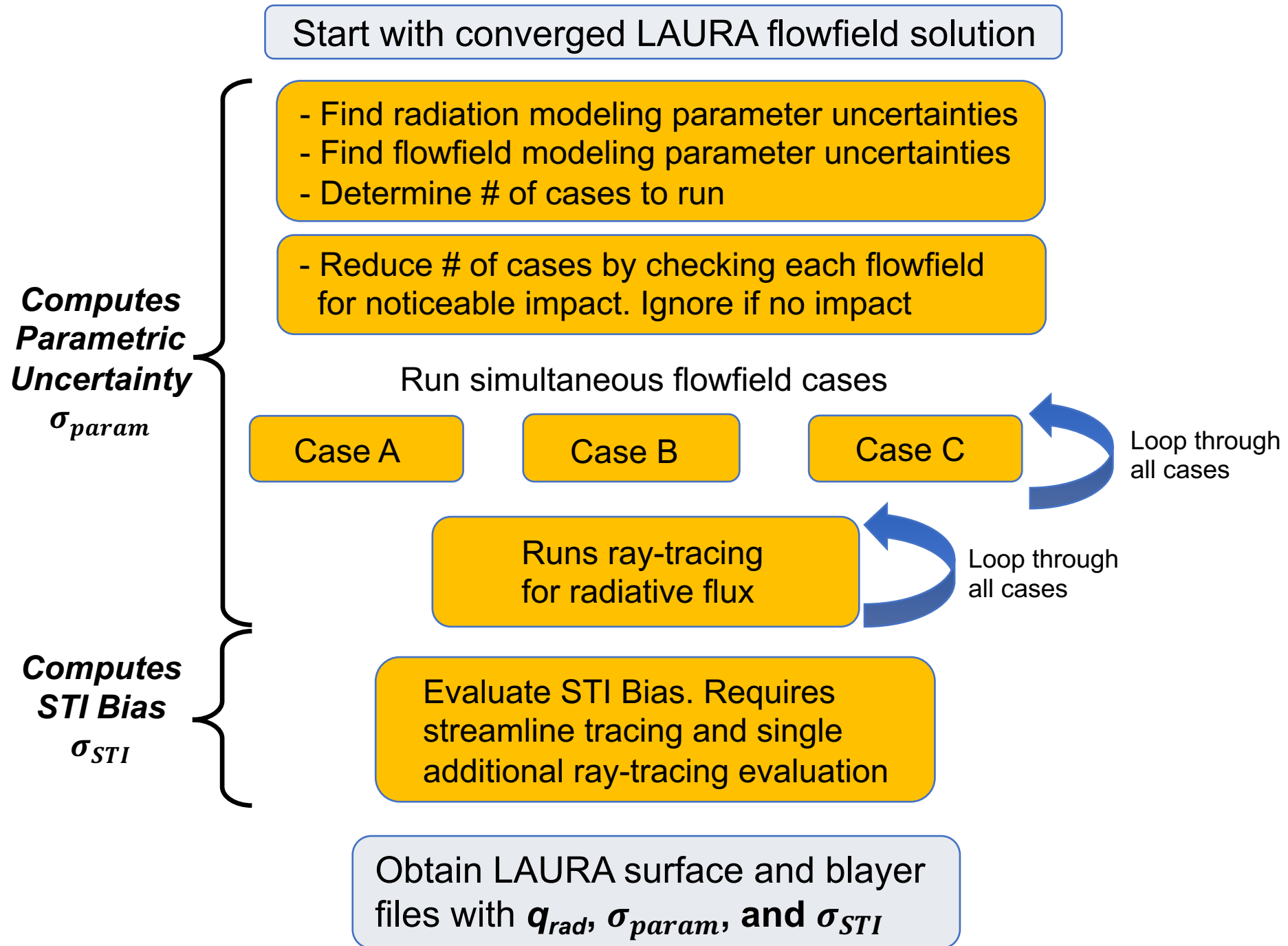
Case C

Loop through
all cases

Runs ray-tracing
for radiative flux

Loop through
all cases

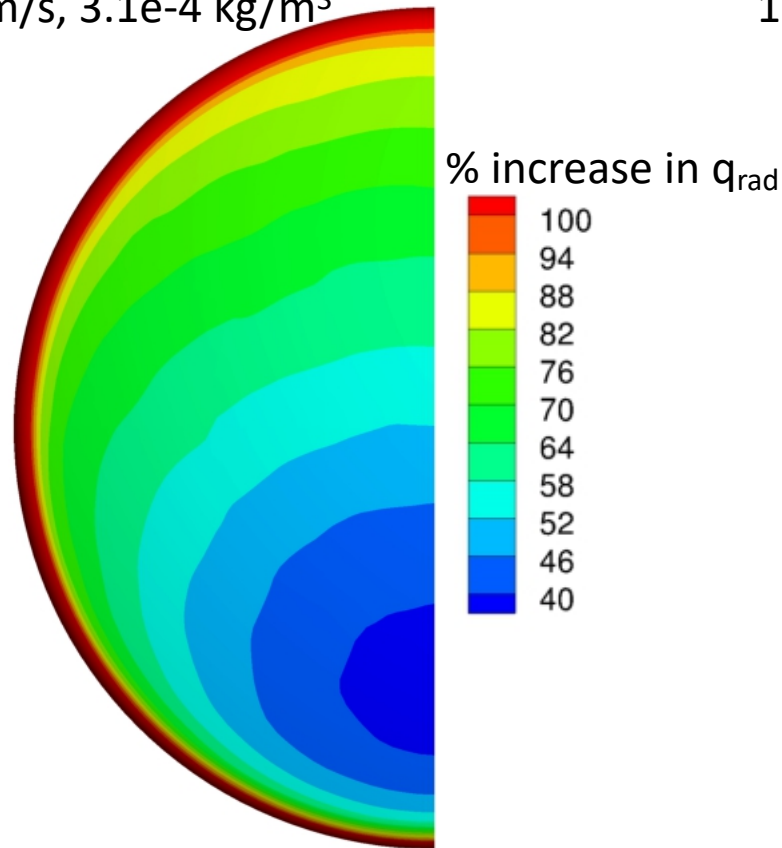
Obtain LAURA surface and blayer
files with $\mathbf{q}_{rad}$, σ_{param} , and σ_{STI}



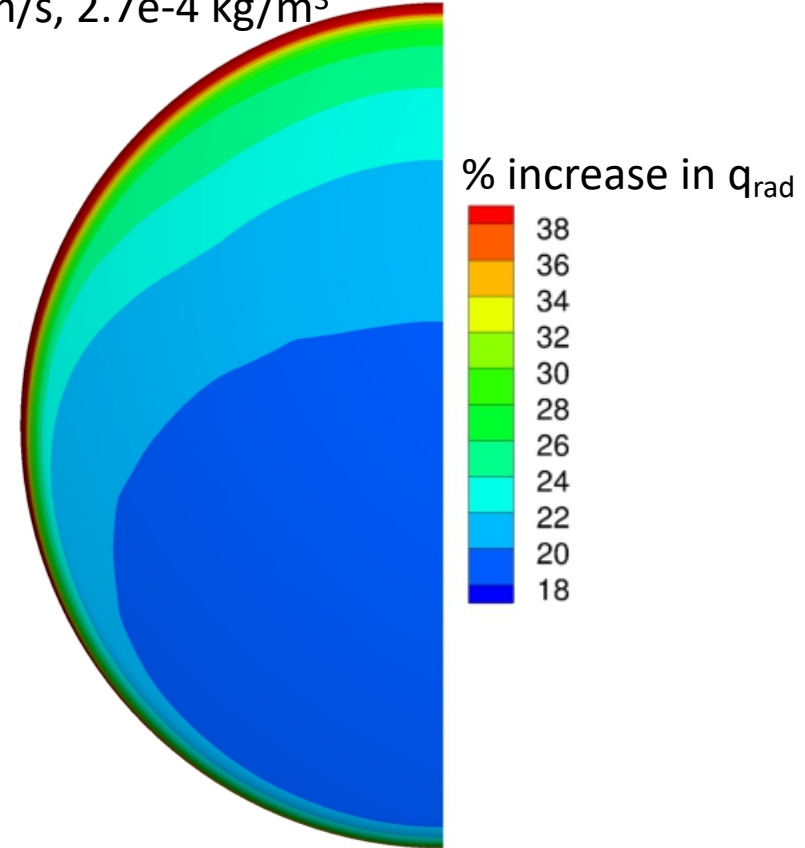
Parametric Uncertainty (σ_{param})

Examples: Orion

9 km/s, $3.1\text{e-}4 \text{ kg/m}^3$



10.5 km/s, $2.7\text{e-}4 \text{ kg/m}^3$

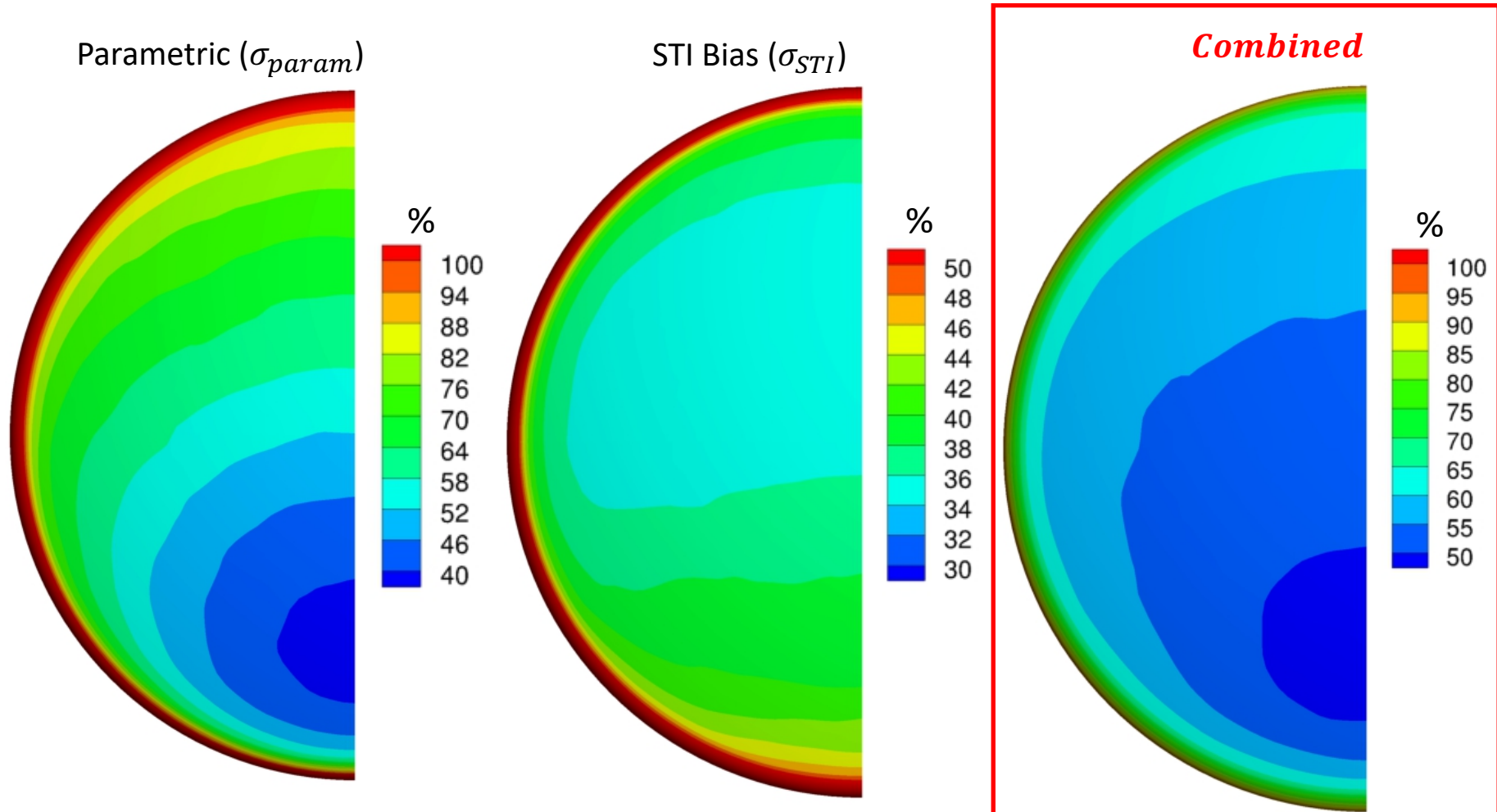


- Uncertainties increase significantly near shoulder, which was not previously captured by stagnation-line analyses.

Combined Final Margin

- Applying approach of Cruden et al. (JTHT, Vol. 32, 2018) to combine the parametric uncertainty and STI bias:

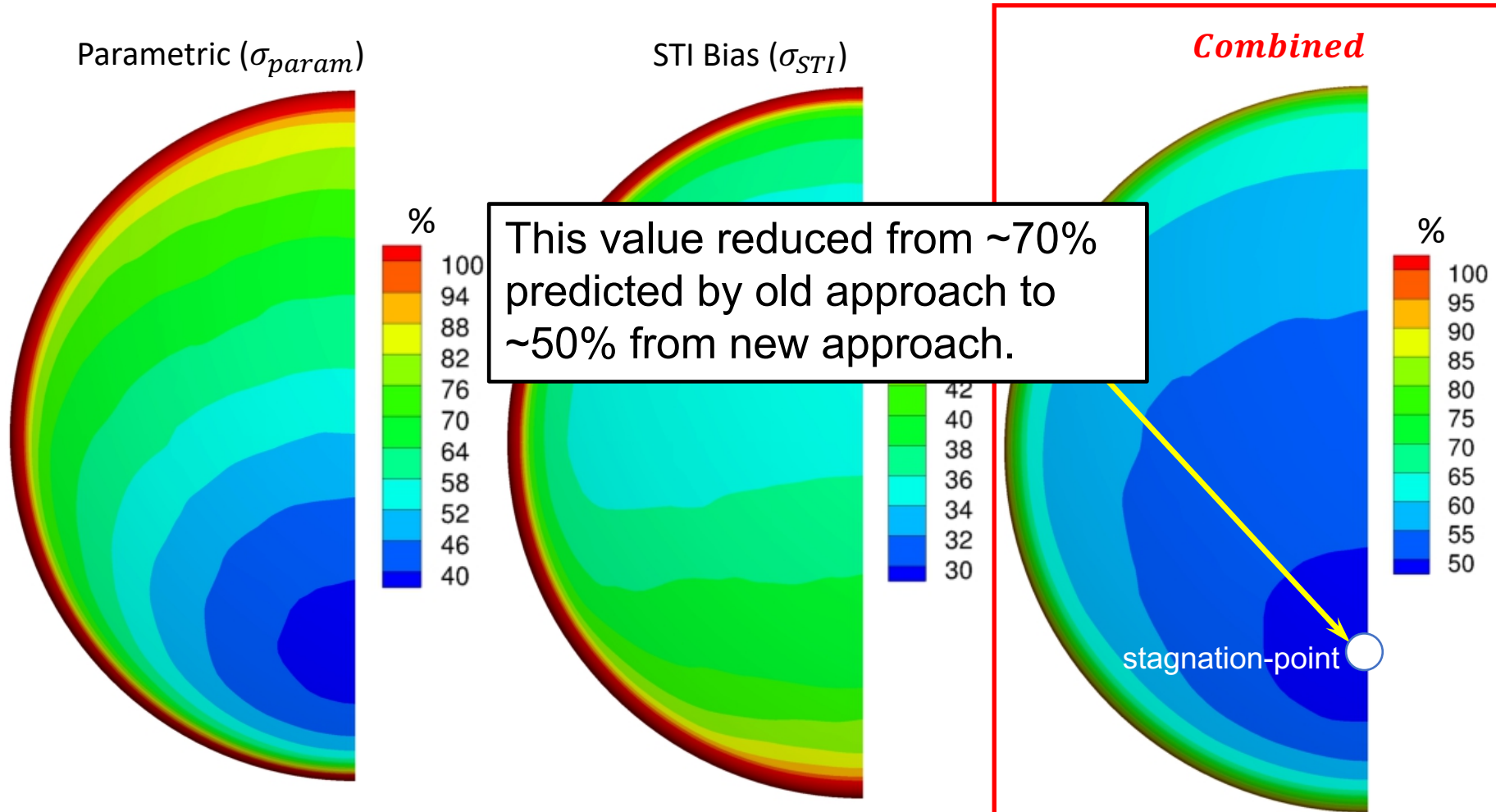
Orion: 9 km/s, 3.1e-4 kg/m³



Combined Final Margin

- Applying approach of Cruden et al. (JTHT, Vol. 32, 2018) to combine the parametric uncertainty and STI bias:

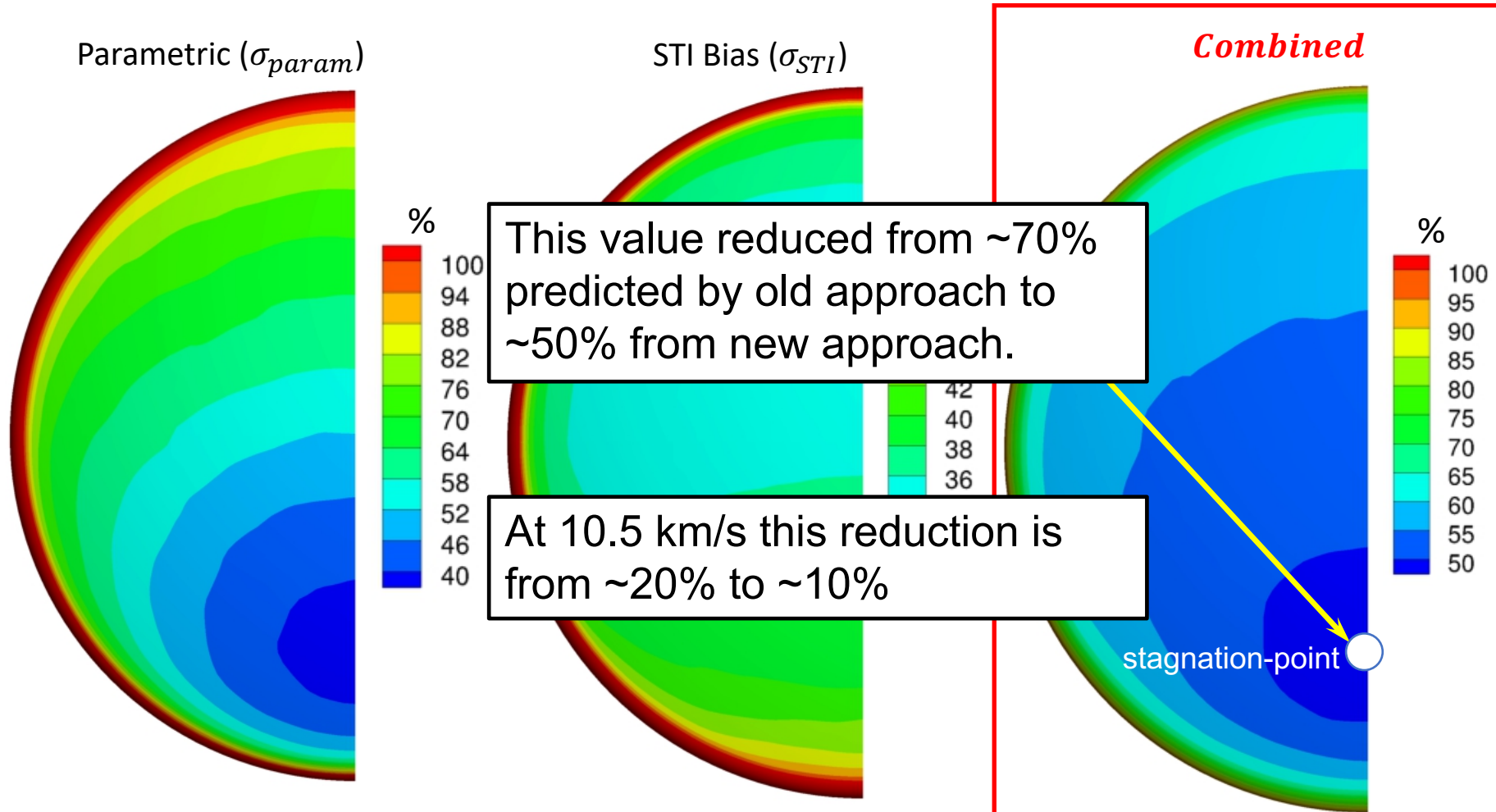
Orion: 9 km/s, 3.1e-4 kg/m³



Combined Final Margin

- Applying approach of Cruden et al. (JTHT, Vol. 32, 2018) to combine the parametric uncertainty and STI bias:

Orion: 9 km/s, 3.1e-4 kg/m³

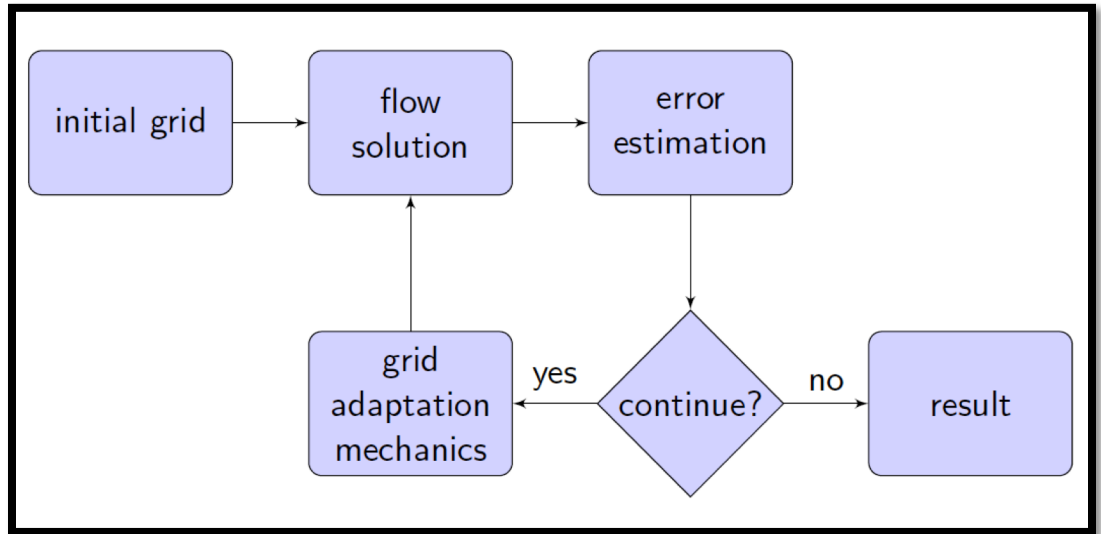


Sketch-to-Solution

Sketch-to-Solution

“Ultimately the mesh generation process should be invisible to the CFD user or engineer.” —*CFD Vision 2030 Study*

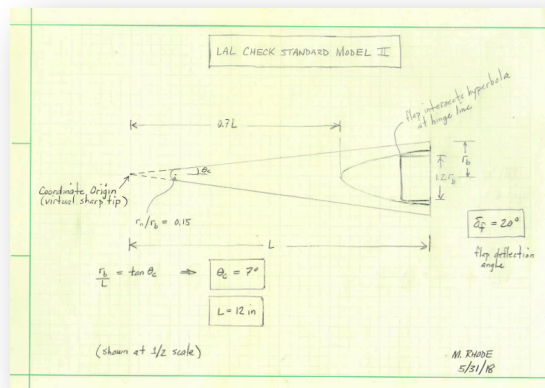
- Fully automated mesh generation/adaptation process
 - User only supplies the geometry
 - Flow solution from FUN3D
 - Mesh is adapted using the *refine* library



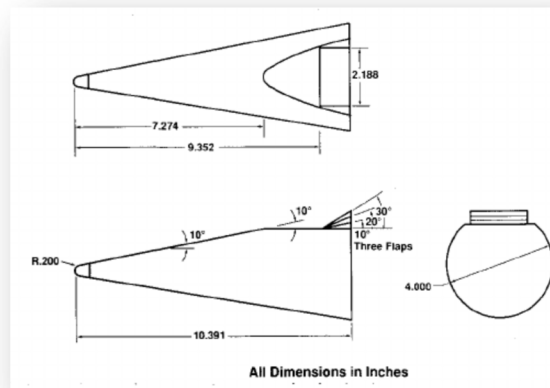
¹<https://github.com/nasa/refine>

Geometry Specification

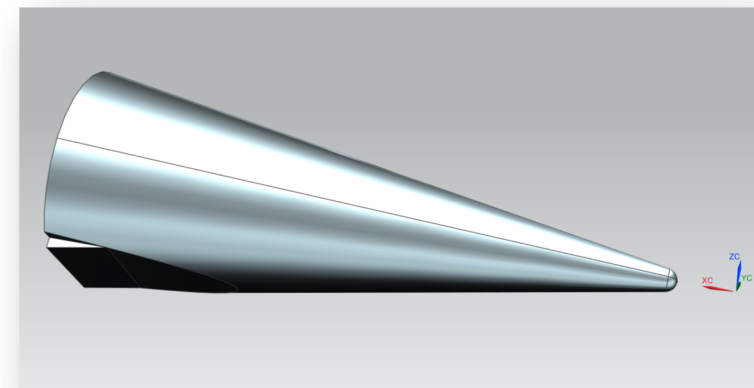
Hand-drawn sketch



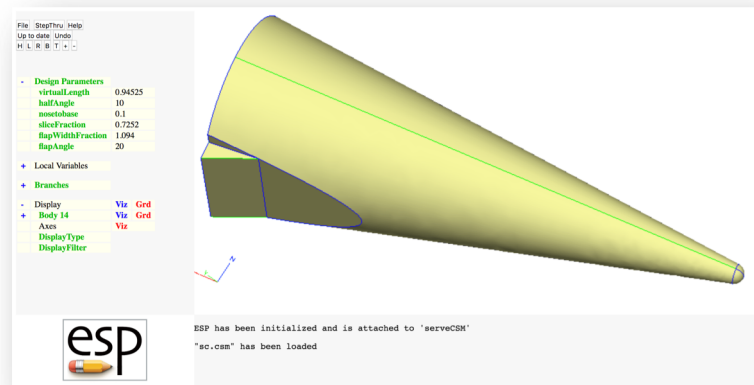
Publication



STEP file



Create geometry model with Engineering Sketch Pad¹

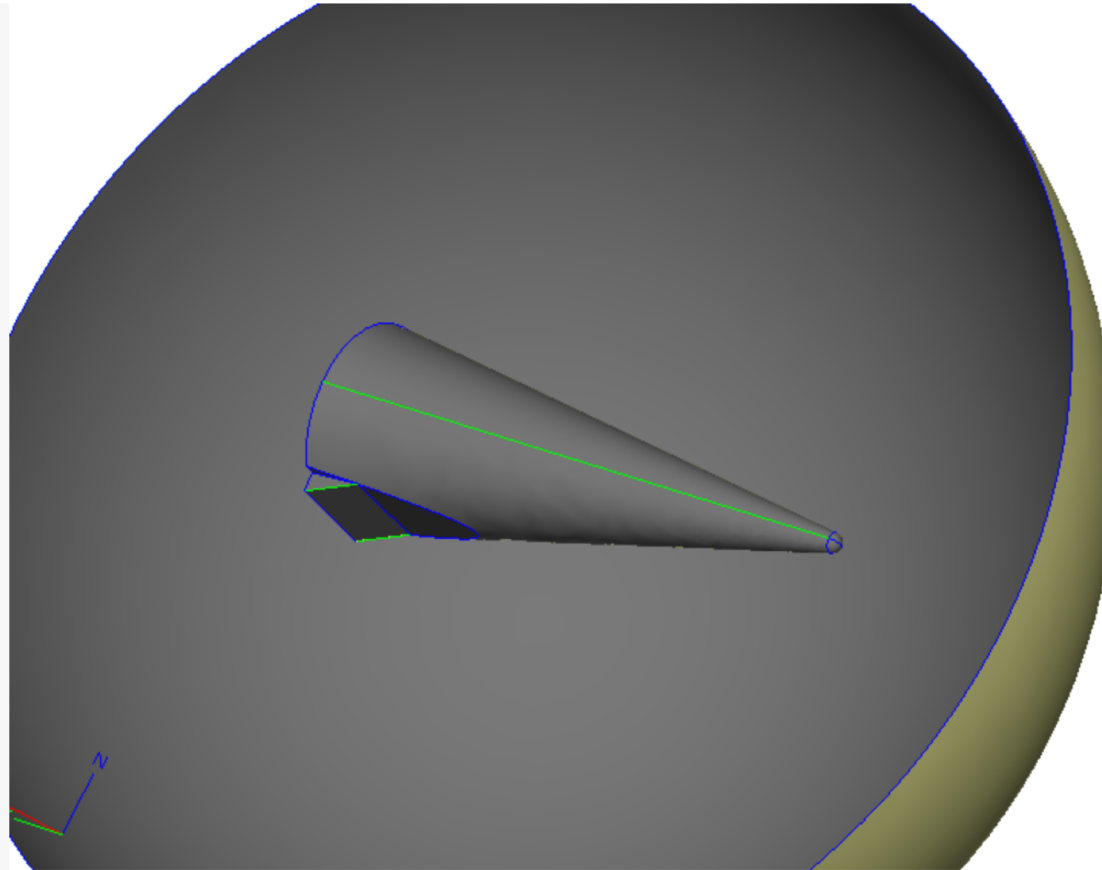


¹Dannenhoffer, AIAA 2013-0701.

Create Computational Domain and Assign BCs

File StepThru Help
Up to date Undo
H L R B T + -

- Design Parameters
 - virtualLength 0.94525
 - halfAngle 10
 - nosetobase 0.1
 - sliceFraction 0.7252
 - flapWidthFraction 1.094
 - flapAngle 20
- + Local Variables
- + Branches
 - Display Viz Grd
 - + Body 16 Viz Grd
 - Axes Viz
 - DisplayType
 - DisplayFilter



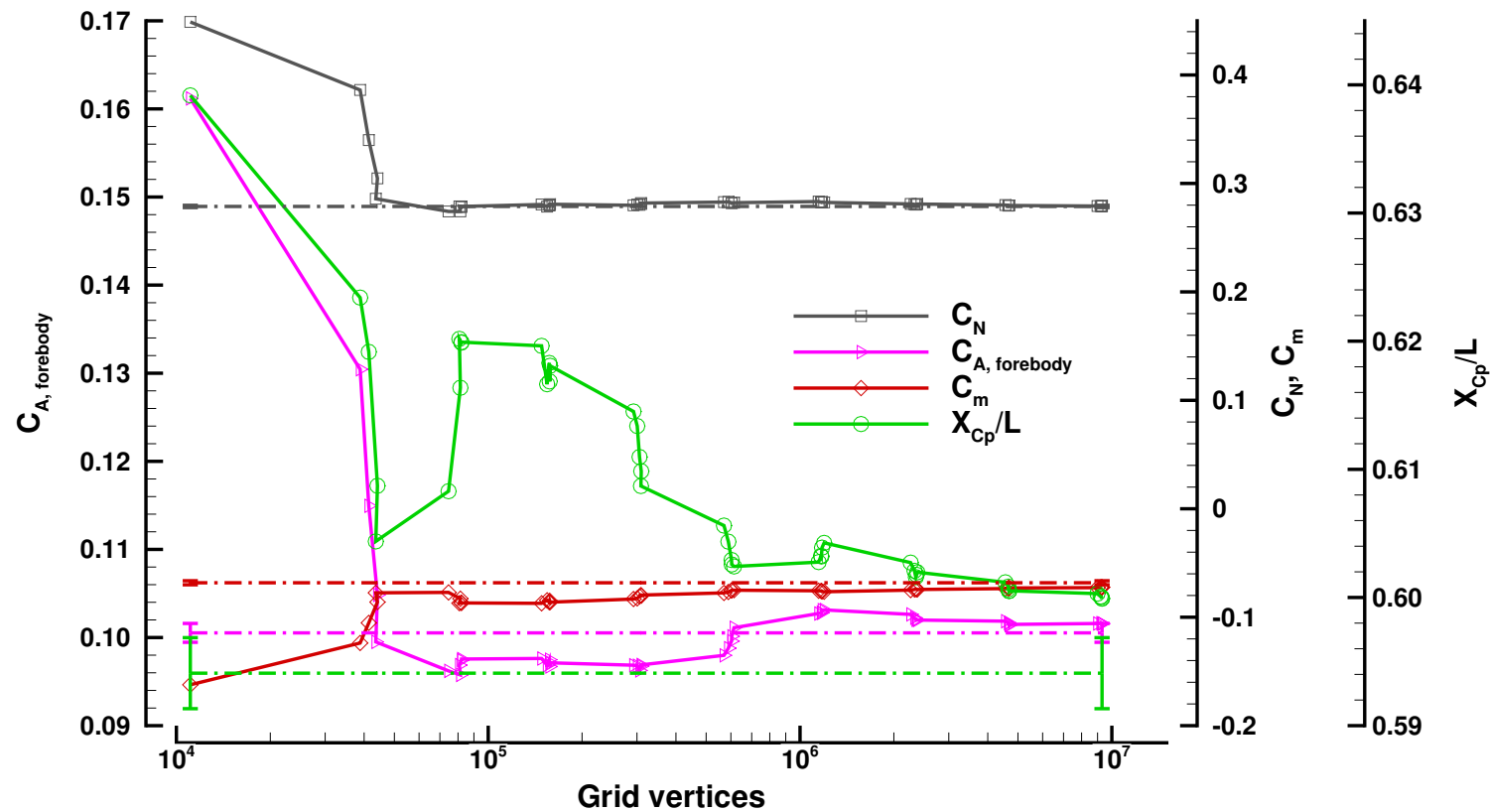
Entire build complete, which generated 1 Body(s)

Toggling visibility of Body 16 Face 2

```
Picked: Body 16 Face 1
_body= 1 0
_brch= 5 0
_hist= 1 16
_faceID= 1 0 1
bc= farfield
```

Results

- Mach 8, 10° AoA, flapless case
 - Experimental measurements⁹ shown as dash-dot lines with 2σ uncertainty bars.



⁹Oberkampf and Aeschliman, AIAAJ 30(8), 1992.

Sketch-to-Solution Details

- Details of workflow and other cases in AIAA 2019-2948

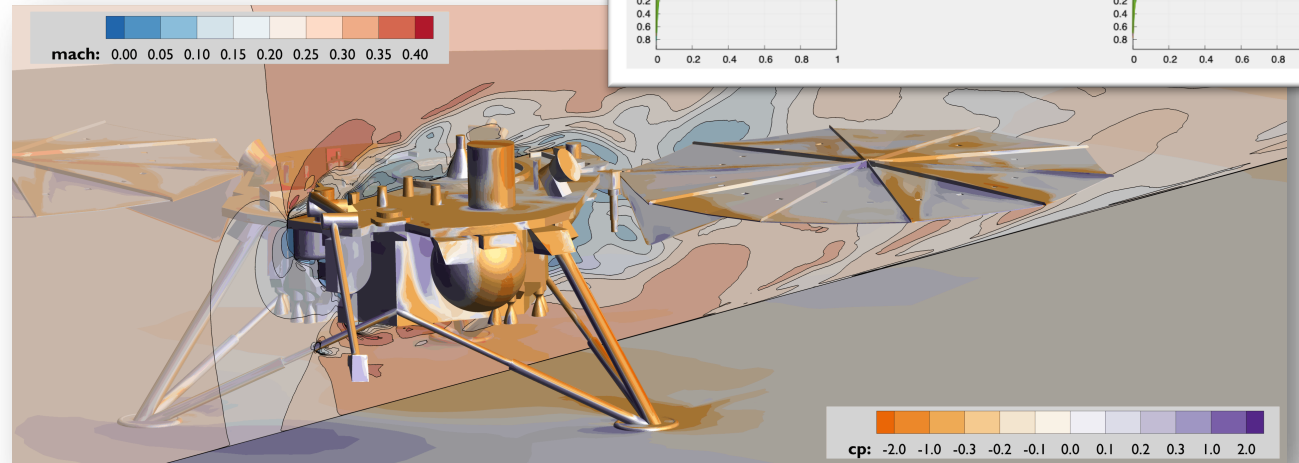
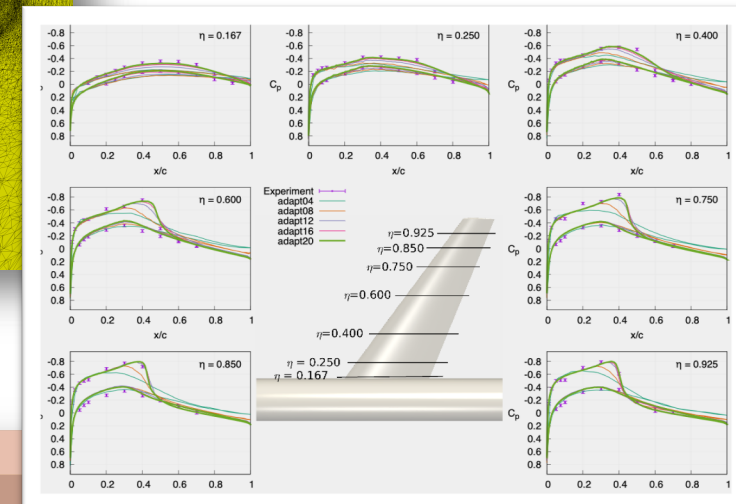
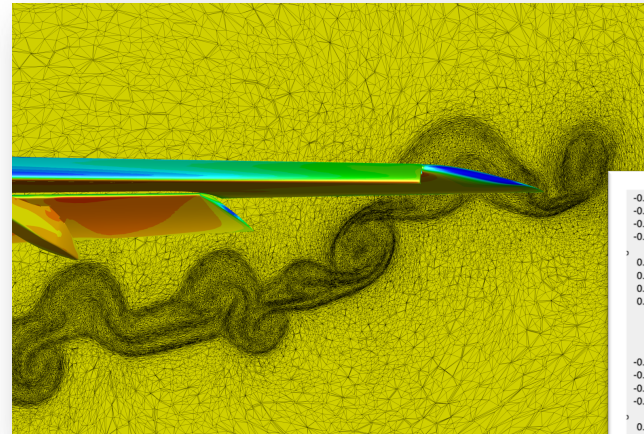
Sketch-to-Solution: An Exploration of Viscous CFD with Automatic Grids

Bil Kleb,^{*} Michael A. Park,[†] William A. Wood,[‡] Karen L. Bibb,[§] and Kyle B. Thompson[¶]
NASA Langley Research Center, Hampton, VA 23681, USA

Reynaldo J. Gomez III^{||}
NASA Johnson Space Center, Houston, TX 77058, USA

Steffan H. Tesch^{**}
Arizona State University, Tempe, AZ 85287, USA

Numerical simulation of the Reynolds-averaged Navier–Stokes (RANS) equations has become a critical tool for the design of aerospace vehicles. However, the issues that affect the grid convergence of three dimensional RANS solutions are not completely understood, as documented in the AIAA Drag Prediction Workshop series. Grid adaption methods have the potential for increasing the automation and discretization error control of RANS solutions to impact the aerospace design and certification process. The realization of the CFD Vision 2030 Study includes automated management of errors and uncertainties of physics-based, predictive modeling that can set the stage for ensuring a vehicle is in compliance with a regulation or specification by analysis). For example, the Cart3D inviscid analysis package has automated Cartesian cut-cell gridding with output-based error control. Fueled by recent advances in the fields of anisotropic grid adaptation, error estimation, and geometry modeling, a similar work flow is explored for viscous CFD simulations; where a CFD application engineer provides geometry, boundary conditions, and flow parameters, and the sketch-to-solution process yields a CFD simulation through automatic, error-based, grid adaptation.



Concluding remarks

- LAURA/FUN3D are CFD tools that enable high-fidelity simulation capabilities tailored for Entry, Descent, and Landing applications
- LAURA primarily used for Aerothermodynamic predictions
- FUN3D primarily used for Aerodynamic predictions
- Both codes contain core thermo-chemical nonequilibrium capabilities

Questions?

fun3d-support@lists.nasa.gov

laura-support@lists.nasa.gov

(Both are civil-servant only)