SAND2016-4803 C

Solution Verification for Field Variables

V. Gregory Weirs

Sandia National Laboratories

Exceptional service in the national interest

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

Verification is about numerical error

Code Verification

- Goal: software quality and algorithmic improvement
- Have exact solution, so can compute exact error
- Hard estimates of convergence properties
- Metrics defined by numerical analysis

Solution Verification

- Goal: Estimate numerical error for problems with unknown solutions ("real" problems)
- Soft estimates of numerical error
- Metrics defined by analyst
- Also called "calculation verification"

Motivation

This work is about bridging the gap between code verification and solution verification:

- Provide appropriate confidence in solution verification results (error estimates)
- Better understanding of the weak points in the solution verification process

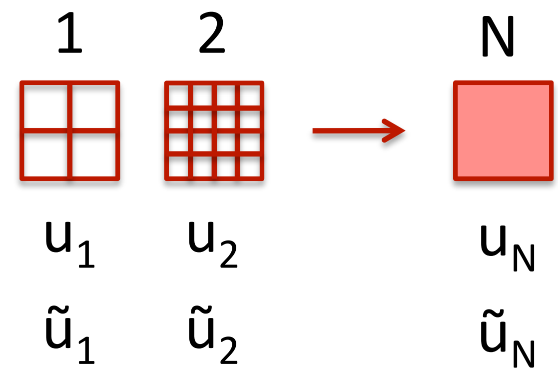
This is work in progress so results are more suggestive than definitive.

Sierra Aero – compressible flow code

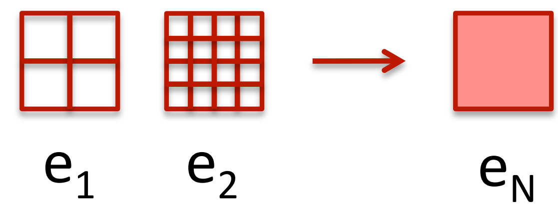
- Solves compressible Euler and Navier-Stokes equations, including RANS turbulence models
- Demonstrated parallel scaling to tens of thousands of cores
- 2D and 3D unstructured meshes, several element types
- Numerical method:
  - Edge-based finite elements a la Barth
  - Nodal values are interpreted as point values or cell averages of a dual mesh
  - TVD methods compute fluxes normal to cell edges
  - First or second order spatial accuracy
  - Implicit or explicit time advancement methods

Review of code verification

1. Exact and approximate values of field variables on suitably refined meshes



2. Exact errors on each mesh

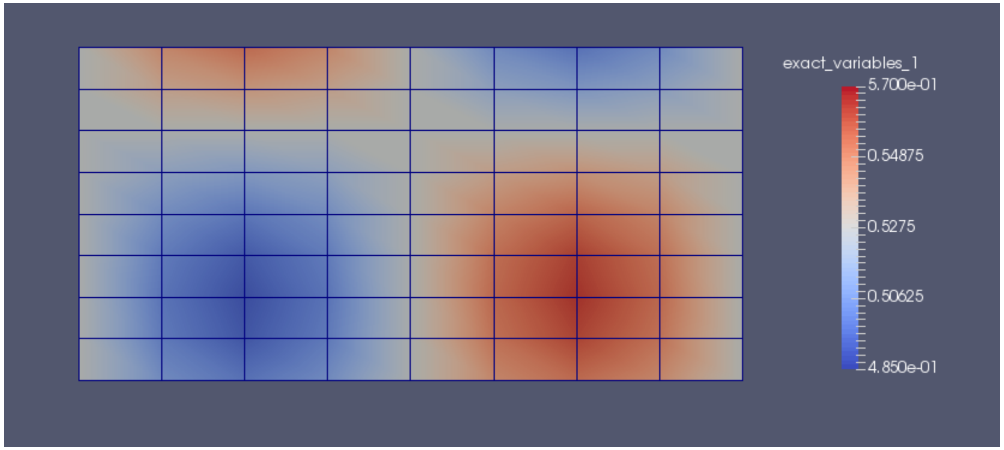


3. "Exact" global error norm on each mesh

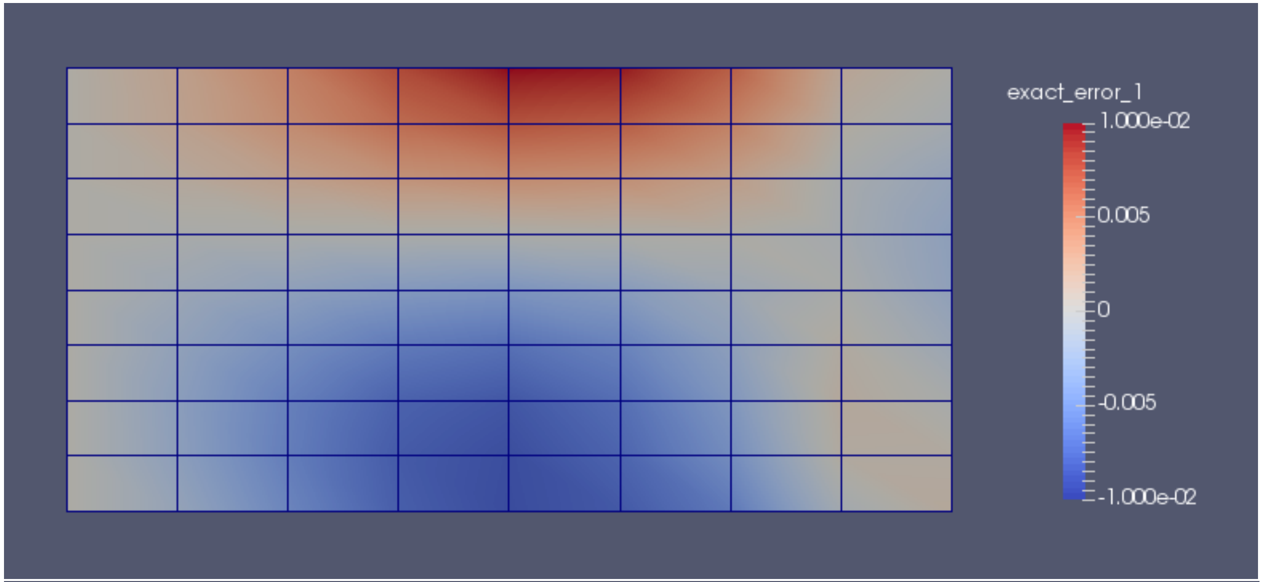
$||e_1||, \ldots, ||e_N||$

Euler Box manufactured solution

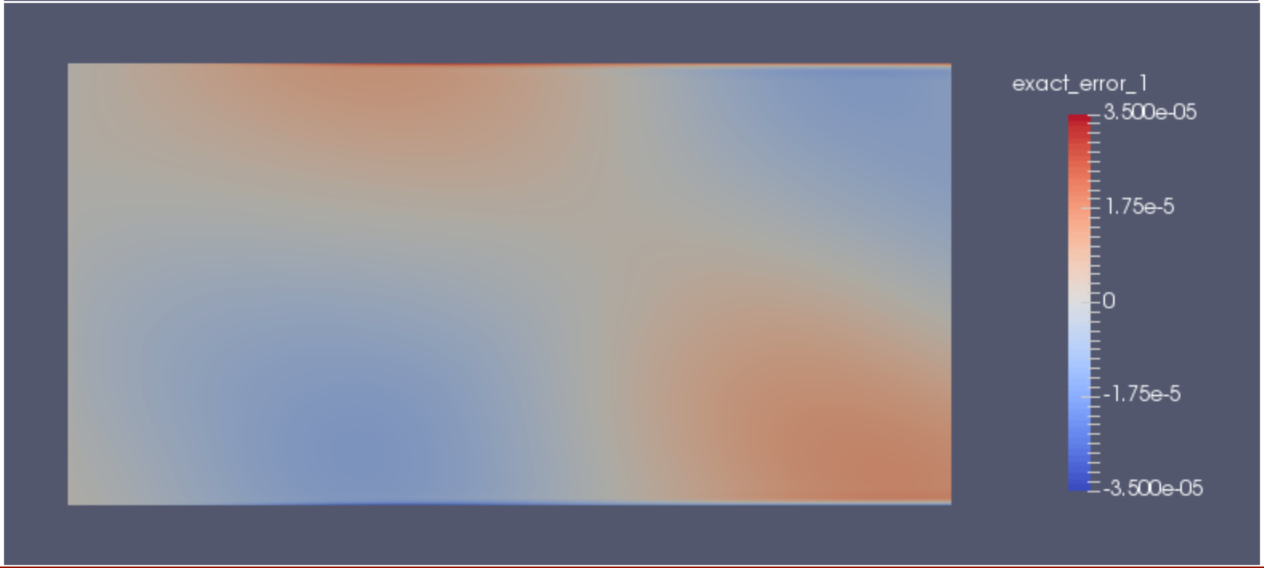
- 2D Euler equations on (0,2) x (0,1)
- $\rho = p/RT$
  $= p_0(1 - \varepsilon \sin(\pi x) \cos(\pi y) / (RT_0 (1 + \varepsilon \sin(\pi x) \sin(\pi y)))$
- $u = u_0 (1 - \varepsilon \sin(\pi x) \cos(\pi y))$
- $v = u_0 \varepsilon \sin(\pi x) \sin(\pi y)$
- $\varepsilon = 0.05$
- $P_0 = 35651.28$ Pa
- $T_0 = 236.2$ K
- $M_0 = 2.5$
- $U_0 = M_0 c_0$
  $= 770.326$ m/s



Exact density error



- Mesh 2
- 64 elements



- Mesh 6
- 16384 elements
- Highest error at upper and lower boundaries

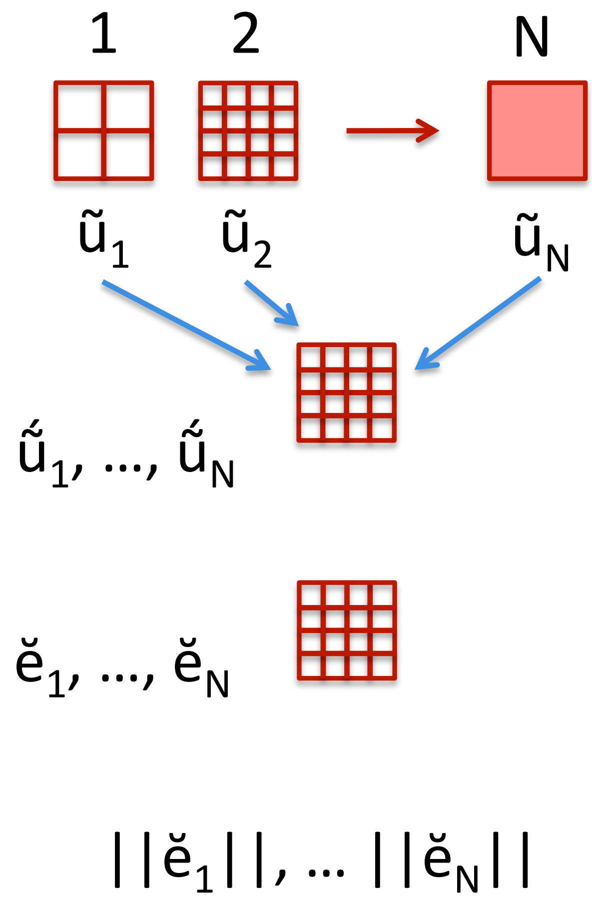
Exact density error norms

Mesh	L1 Error	Rate
2	3.26953562548e-3	-
3	5.39402430176e-4	-
4	1.08958268477e-4	2.31
5	2.53897840066e-05	2.10
6	6.22160970615e-06	2.03

- The refinement ratio is 2 across all meshes
- The expected convergence rate is 2 (limiters turned off)

Field variable solution verification

1. approximate values of field variables on suitably refined meshes
2. Spatial interpolation or “sampling” to a common mesh
3. At each node, extrapolated solution at h=0, or an error estimate on the fine mesh
4. Approximate global error norm



Case 1: Restrict to coarse mesh

- Mesh 1 is the coarsest mesh
- On mesh 1, Get $\acute{\tilde{u}}_2, \ldots, \acute{\tilde{u}}_N$ from $\tilde{u}_2, \ldots, \tilde{u}_N$ by "sampling" or restriction of values on the other (finer) meshes
- By restriction, we mean that the values on the finer mesh are injected onto the coarser mesh; not all values are used.
- If nodes on the coarse mesh are collocated with nodes on the finer meshes, this restriction operation does not involve interpolation and can be thought of as a "sample" of the finer mesh values.
- This approach leads to a coarse representation of the field variable, which may be undesirable
- Restriction and sampling are used loosely, just to indicate a fine-to-coarse injection of values

Case 2: Prolong to fine mesh

- Mesh N is the finest mesh
- On mesh N, Get $\acute{\tilde{u}}_1, \ldots, \acute{\tilde{u}}_{N-1}$ from $\tilde{u}_1, \ldots, \tilde{u}_{N-1}$ by interpolating or prolonging values on the other (coarser) meshes to the fine mesh nodes
- In contrast to case 1, values must be generated (by interpolation) for nodes not present in coarser meshes
- Interpolation introduces error
- Like restriction and sampling, prolongation is used loosely; but prolongation and interpolation indicate a coarse-to-fine injection of values

Richardson extrapolation

- Now we have approximate solution values $\acute{\breve{u}}_1, \ldots, \acute{\breve{u}}_N$ on each node of a common mesh
- Use two-mesh Richardson extrapolation to generate a sequence of error estimates:

  $u = \acute{\breve{u}}_2 + a\, h_2^p$

  $u = \acute{\breve{u}}_3 + a\, h_3^p$

  $\breve{e}_3 = (\acute{\breve{u}}_2 - \acute{\breve{u}}_3)\, h_3^p / (h_3^p - h_2^p)$
- We use the expected convergence rate of p=2
- There are other ways to do this step, which we will examine in future work

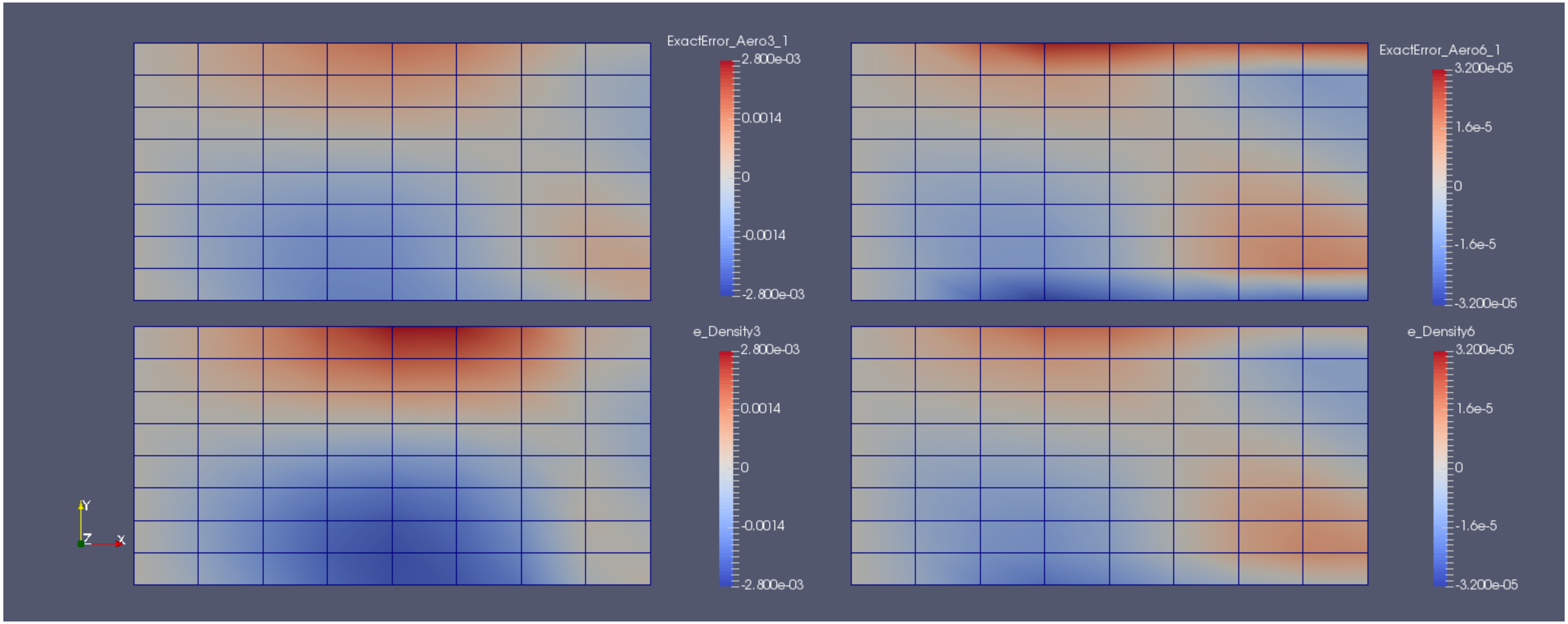
Exact and approximate error norms

Mesh	Exact L1 Error	Coarse L1 Error	Fine L1 Error
2	3.270e-3	-	-
3	5.394e-4	9.201e-4	9.384e-4
4	1.090e-4	1.446e-4	1.745e-4
5	2.539e-05	2.797e-05	3.932e-05
6	6.222e-06	6.431e-06	9.638e-06

- “Coarse” and “fine” refer to common meshes to which computed solutions were interpolated
- For the solutions computed on the finer meshes, interpolating to the finest target mesh gives a larger error norm than interpolating to the coarsest target mesh

Local errors, case 1 (coarse)

Mesh 3 solution error | Mesh 6 solution error



Exact errors (top row) are interpolated to the common mesh

Exact errors are lower than estimates for the mesh 3 solution, but higher for the mesh 6 solution

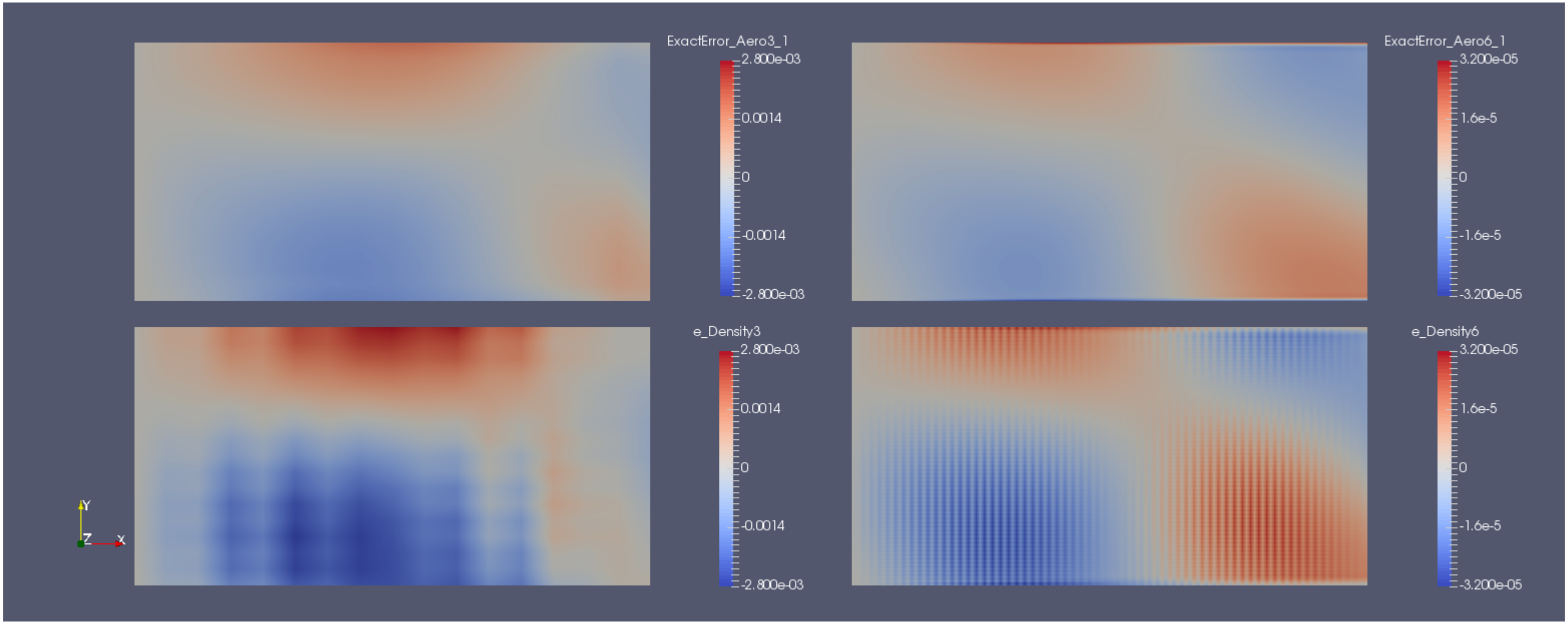
Error norms, case 1 (coarse)

Mesh	L1 Error	Rate
2	-	-
3	9.20121947074e-4	-
4	1.44585170331e-4	2.67
5	2.79745051819e-05	2.37
6	6.43078020116e-06	2.12

- The convergence rates of the approximate error norms are similar to those of the exact error norms for this problem

Local errors, case 2 (fine)

Mesh 3 solution error | Mesh 6 solution error



Exact errors (top row) are interpolated to the common mesh

Banding results from coarse mesh imprinting on error estimates (bottom row)

Error Norms, case 2 (fine)

Mesh	L1 Error	Rate
2	-	-
3	9.38383858014e-4	-
4	1.7426436576e-4	2.43
5	3.93242150122e-05	2.15
6	9.63784004988e-06	2.03

- While slightly lower than for case 1, convergence rates of the approximate error norm are similar to those of the exact error norm
- For case 2, the L1 error may be affected by interpolation to the target mesh, particularly as the meshes are refined (still need to confirm this)

Error norms, oblique shock problem

Mesh	Exact L1 Error	Coarse L1 Error	Fine L1 Error
1	3.090e-02	-	-
2	1.598e-02	1.500e-02	1.552e-02
3	8.278e-03	8.101e-03	8.966e-03
4	4.207e-03	4.360e-03	4.841e-04
5	2.142e-03	1.955e-03	2.655e-03

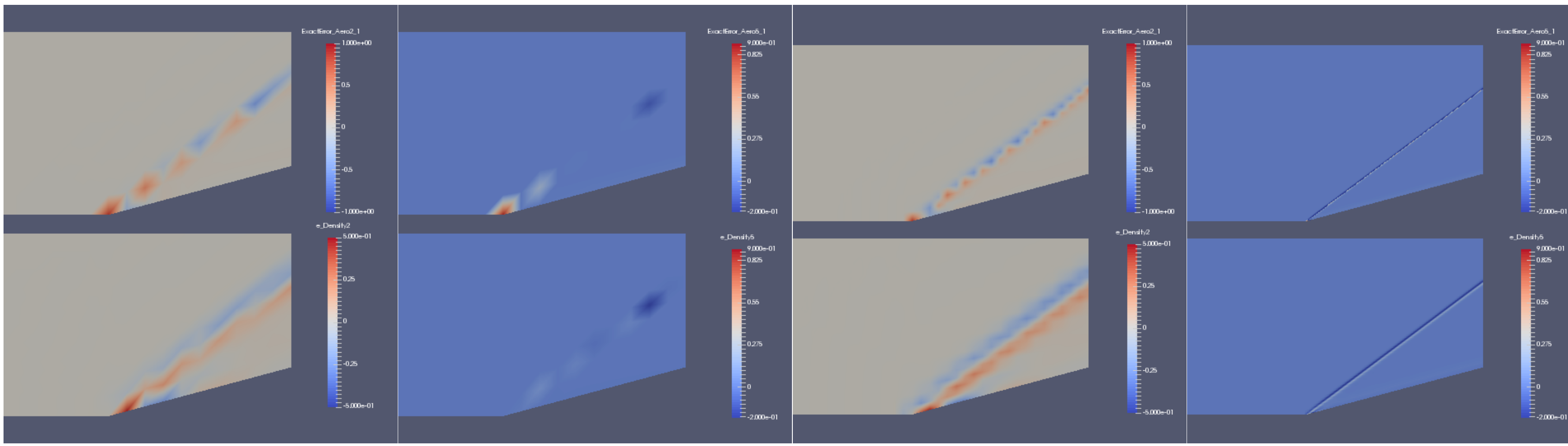
- Same trends as for smooth solution
- Convergence rate ~1, as expected. (Also, assumed p=1 in Richardson extrapolation.)

Local errors, oblique shock problem

Sandia National Laboratories

Coarse | Fine

Mesh 2 error | Mesh 5 error | Mesh 2 error | Mesh 5 error



- Exact error (top row) and approximate error (bottom)
- Note color scales are not all the same

Concluding remarks

Sandia National Laboratories

- We consider solution verification for a field variable for several reasons:
  - Provides a clearer path connecting to the numerical analysis underlying code verification
  - May expose techniques for improving error estimates
  - Can be used in concert with analyst QoIs to give additional error information
- These (easy) test problems presents a best case scenario for solution verification. More difficult tests will expose how solution verification fails.
- We have made many choices in this analysis, but will examine alternatives in future work.