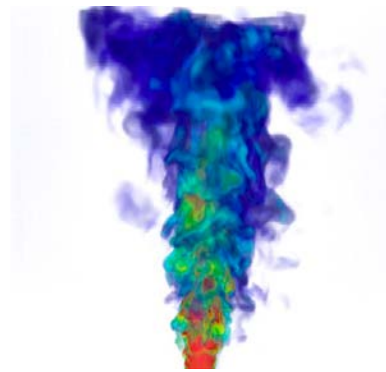
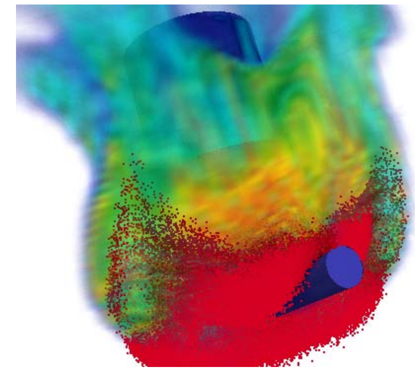
Exceptional service in the national interest

Sandia National Laboratories

A Performance-based Code Assessment for Low Mach Large Eddy Simulations

National Defense Industrial Association, NDIA

Physics-Based Modeling in Design and Development

November 7th, 2012

Stefan P. Domino

Thermal/Fluids Computational Engineering

Sandia National Laboratories, NM

Presentation Overview

- DOE's ASC Project Guiding Principle
- Typical LES Application with Physics Description
- V&V Principle
- Discretization, Algorithmic Behavior and Coupling Approaches
- Performance Results
- Summary of Accomplishments
- Conclusion

Project Guiding Principle

- The SIERRA Mechanics Integrated Code (IC) tool suite is being developed under the Department of Energy's (DOE) Advanced Scientific Computing (ASC) program to support *Science-based Stockpile Stewardship* (SBSS) at Sandia National Laboratories
- Other aspects of SBSS include:
  - Physics and engineering model development, creation of high quality validation data sets, algorithm development and Uncertainty Quantification (UQ)
- The guiding principle for this combined project is to provide a *predictive* capability for high consequence accident scenarios
- The ASC project deliverables are managed by Milestone efforts across the fully supported ASC application space

Abnormal/Thermal Environment

- Hydrocarbon JP-8 10 m fire

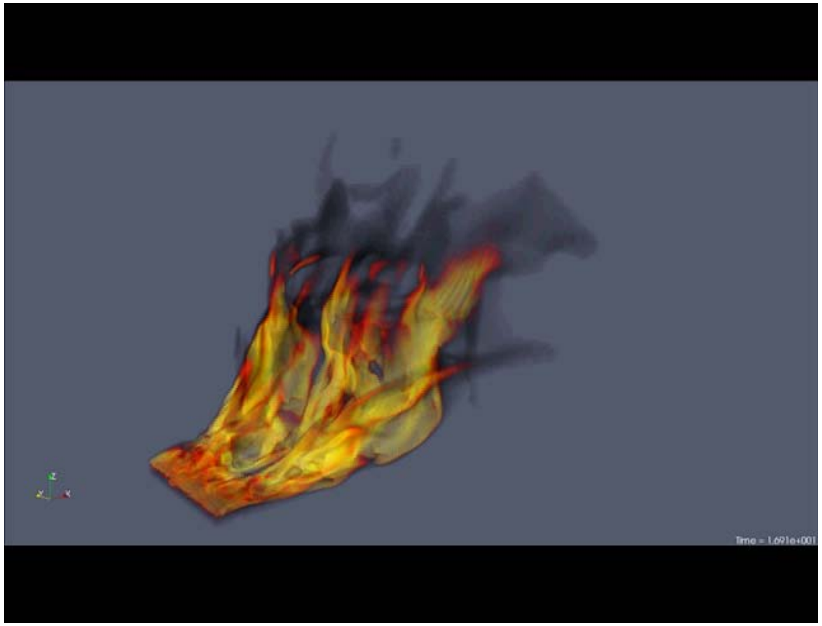
❑ Lead experimentalists: Jim Nakos

- Aluminum propellant fire

❑ Lead experimentalists: Walt Gill

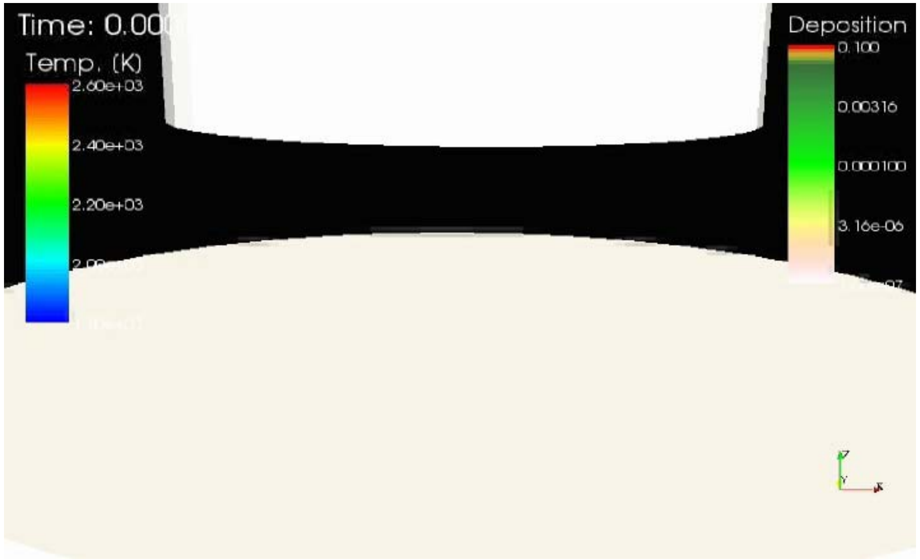
Simulation Capability

- Hydrocarbon JP-8 pool fire



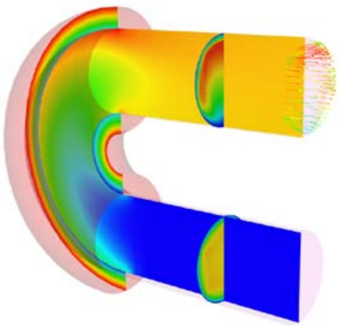
- Multi-physics pool fire simulation

- Aluminum propellant fire

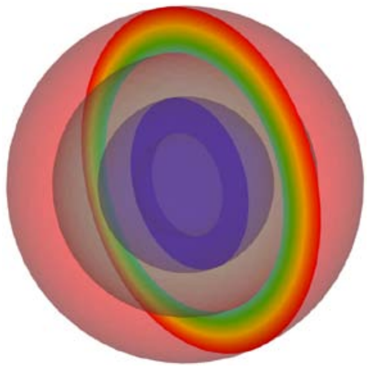


- Multi-physics propellant fire simulation

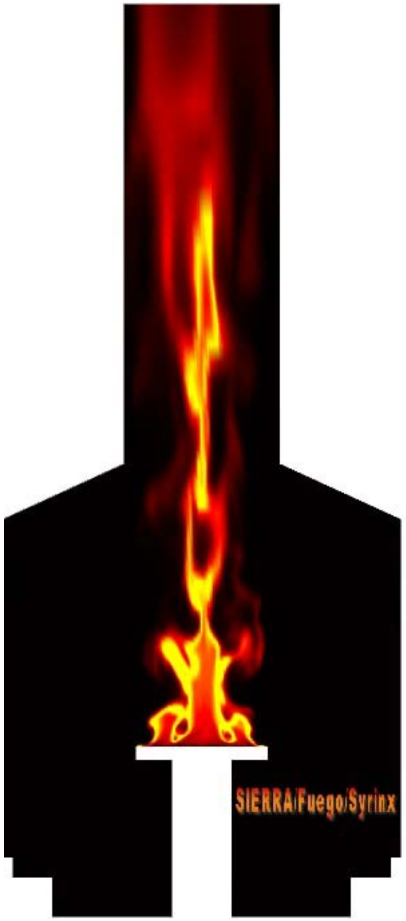
V&V, From Simple to Complex



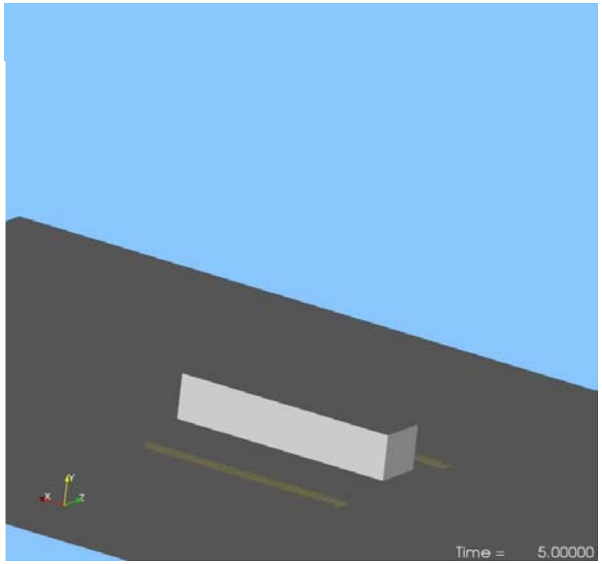
❑ CHT



❑ PMR/HC

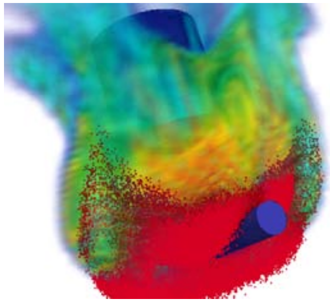


❑ 2 m methane



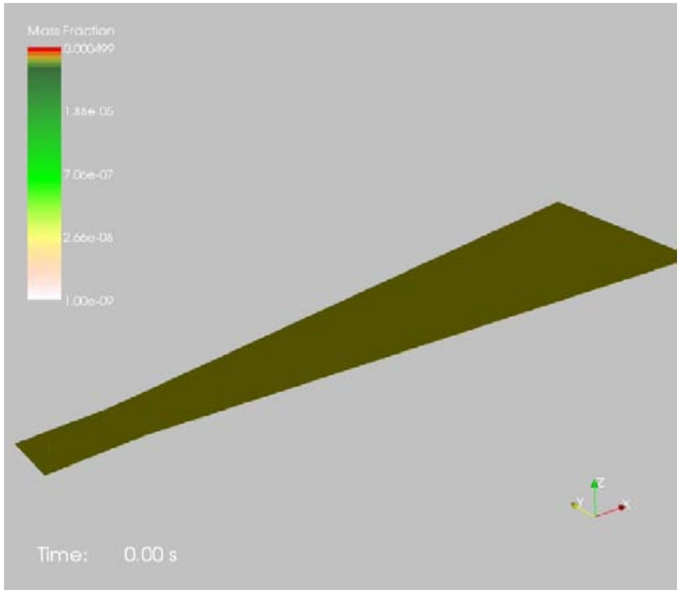
❑ Outdoor accident scenario

❑ 2 meter JP-8 pool fire validation



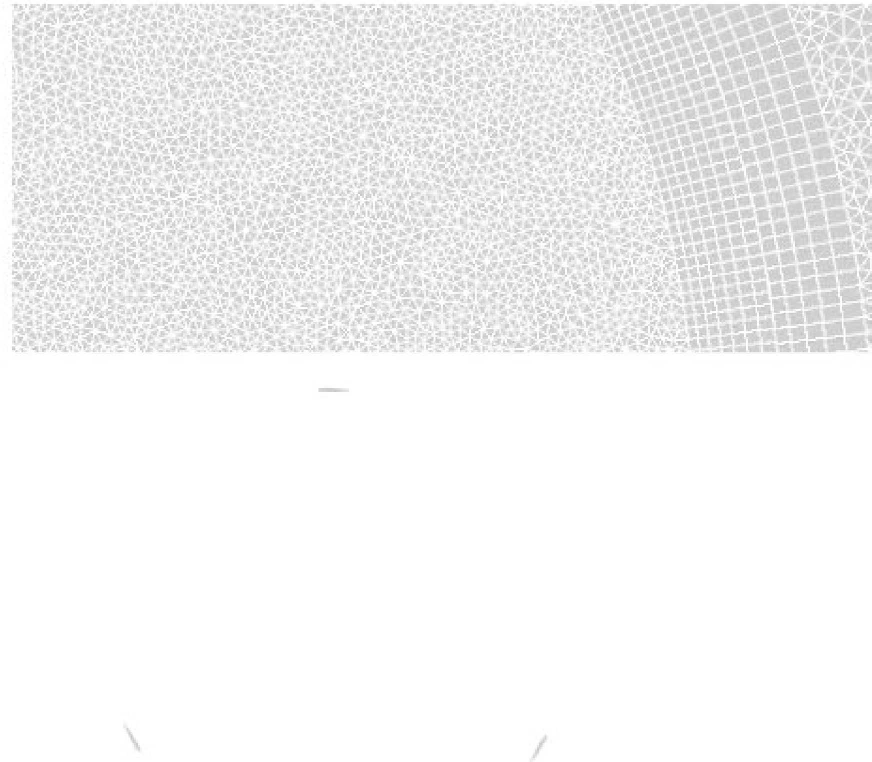
❑ Downward AL burn

Generalized Design for General Apps



- ❑ Contaminant transport

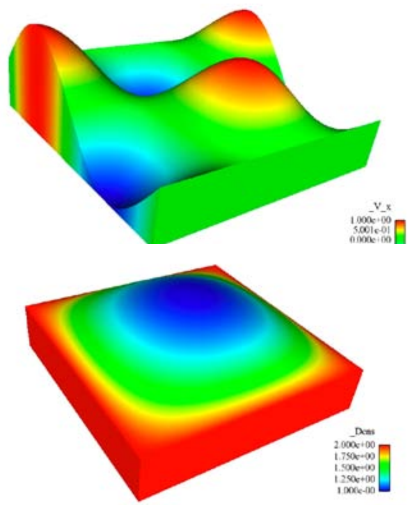
- ❑ Operator split and monolithic FSI



- ❑ Advanced sliding mesh algorithms for wind energy applications

Discretization and Coupling

- A variety of code discretizations have been implemented and verified using the Method of Manufactured Solutions
- Discretizations include:
  - Vertex centered Control Volume Methods
  - Cell centered Control Volume Methods
  - Finite Element Methods
- Couplings range from
  - explicit pressure projection
  - operator split pressure projection
  - monolithic (fully coupled)



❑ Variable density MMS; Ux (T) density (B)

- Exascale promises to be disruptive, expensive and extremely challenging
- Algorithms? Fully explicit, operator split, monolithic?

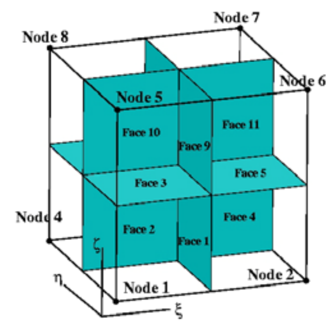
CVFEM Discretization

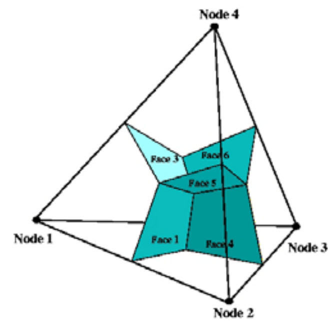
- The core discretization used in the low Mach code base has been the Control Volume Finite Element Method, CVFEM
- An elemental basis is defined from which interpolation and gradients within the element are determined
- The test function is defined to be piece-wise constant
- This method can best be described as a Petrov-Galerkin method
- The canonical 27-point stencil is recovered

$$\int w \frac{\partial \bar{\rho} \tilde{u}_j \tilde{\phi}}{\partial x_j} d\Omega = -\int \bar{\rho} \tilde{u}_j \tilde{\phi} \frac{\partial w}{\partial x_j} d\Omega + \int w \bar{\rho} \tilde{u}_j \phi n_j \, d\Gamma$$

$$w = w_I ; \frac{\partial w_I}{\partial x_j} = -\delta(x - x_{scs})$$

$$\int w \frac{\partial \bar{\rho} \tilde{u}_j \tilde{\phi}}{\partial x_j} d\Omega = \sum_{ip} \left(\bar{\rho} \tilde{u}_j \right)_{ip} \tilde{\phi}_{ip} n_j dS = \sum_{ip} \dot{m}_{ip} \tilde{\phi}_{ip}$$





$$\phi(\vec{x}) = \sum_J N(\vec{x}) \phi_J$$

Finite Element Discretization

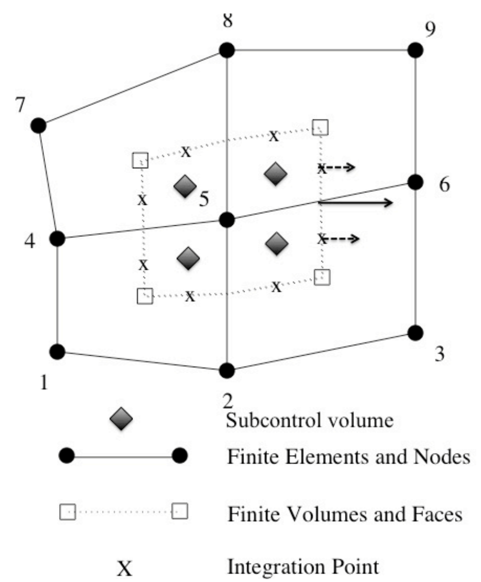
- Classic Equal Order Interpolation with explicit pressure stabilization
- Monolithic or approximate pressure projection couplings exist
- Pressure stabilization can be similar to segregated approach (2nd or 4th order) or PSPG
- Advection stabilization obtained via SUPG

$$\tilde{w} = w + \tau u_j \frac{\partial}{\partial x_j} w$$

- Ramifications for the FEM method:
  - Canonical 27-point stencil for structured hex
  - Full elemental diffusion operator (issues with diffusion operator monotonicity exists for aspect ratios greater than sqrt(2))
  - Galerkin method not regularly used due to the need for residual-based stabilization thus making most implementations a Petrov-Galerkin method
  - VMS foundation replaces classic SUPG and PSPG approach

Edge-Based Discretization

- In this method, the dual mesh is defined to establish geometric values at the edge midpoint (area vector) and node (volume)

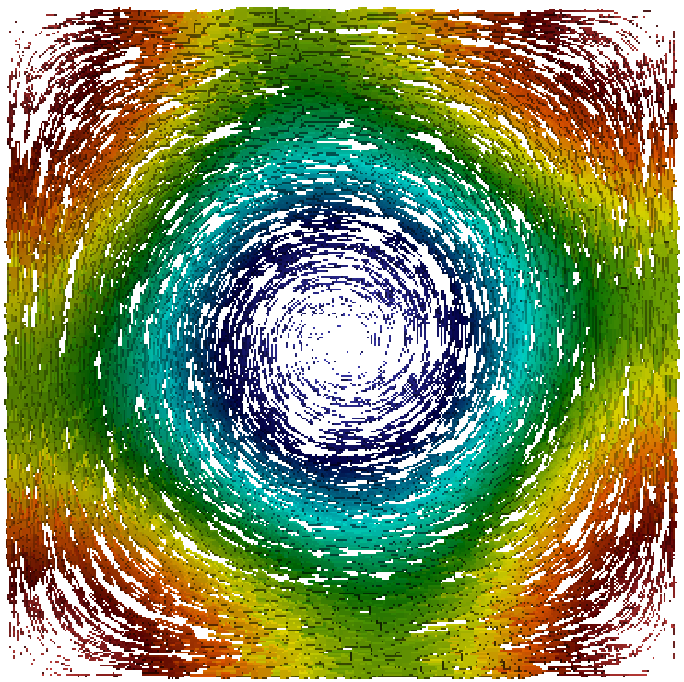


❑ Quadrature points for edge-based scheme

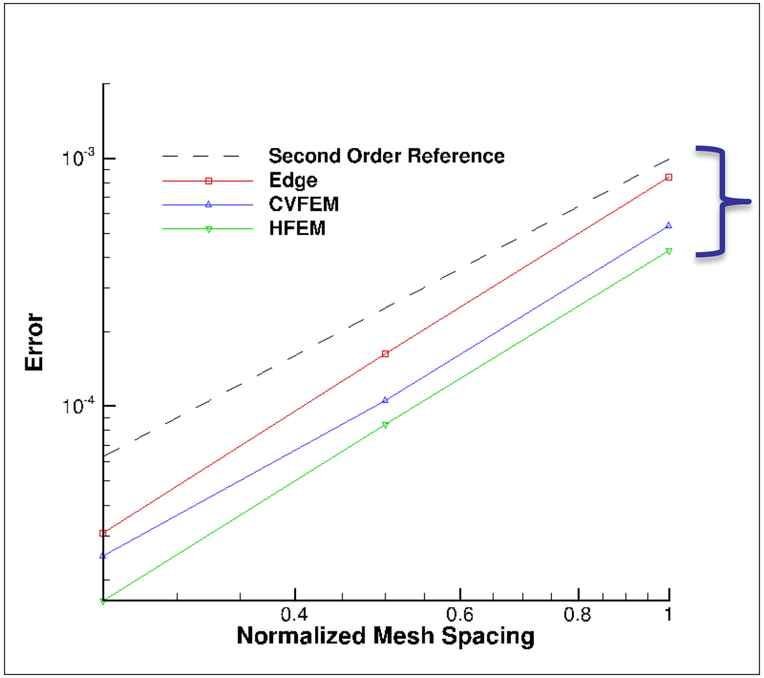
- Ramifications for the edge-based finite volume (EBFV)structure are as follows:
  - Reduced stencil (27-point to 7-point for structured hex)
  - Simple L/R data structure allows for simple interpolation and orthogonal gradient contributions
  - Lack of elemental basis requires a diffusion operator in terms of orthogonal to the edge and non-orthogonal correction that requires projected nodal gradients

Error Tradeoff

- Error disparity on “nice” mesh for a Steady Taylor Vortex MMS for each schemes are comparable
- Other attributes of the scheme, i.e., speed, robustness, time to solution, etc. are far more significant



❑ Steady TV; u_vel colored by pressure

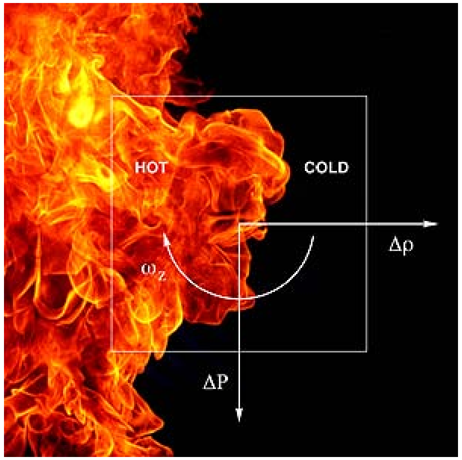


> 1 Order of magnitude speed disparity

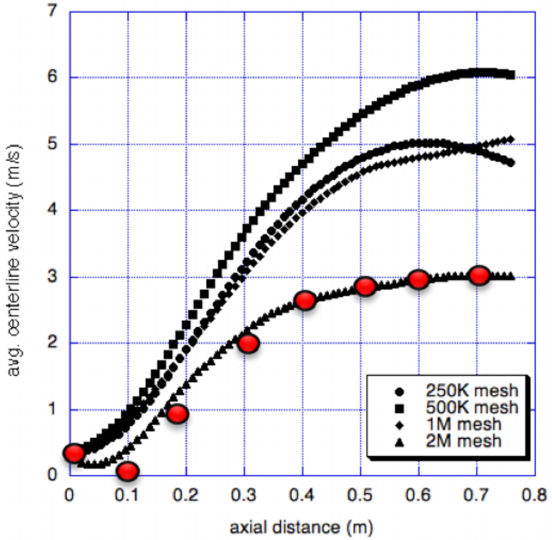
❑ Loo norms for three discretizations

Discretization Error vs Resolution

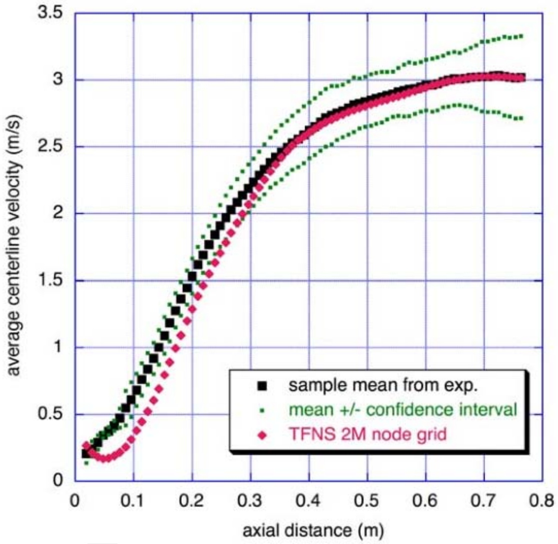
- Common Value System: The best numerical scheme is the one in which errors for a canonical code verification suite are smallest
- However, oftentimes the ability to resolve a physics scale is of prime importance



❑ Fire instability 101



❑ Core collapse as a Function of mesh resolution



❑ Data comparison

Coupling

- The traditional low Mach algorithm is an approximate projection algorithm in which splitting and pressure stabilization terms exist

$$\begin{bmatrix} A & G \\ D & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f \\ b \end{bmatrix} + \begin{bmatrix} (I - A\tau) G \left(p^{n+1/2} - \alpha p^{n+1/2} \right) \\ \tau (L - \beta D G) p^{n-1/2} \end{bmatrix}$$

- α and β define incremental pressure/pressure-free and 2nd and 4th pressure stab

- Other approaches are possible including monolithic and flavors of operator split
- In general, there exists a trade space between time scale of interest and coupling approach

Algorithm	Speed factor
uvwp; Imp/Imp	3.4x
uvw_p; Imp;/Imp	1.2x
uvw_p; Imp/Exp	0.6x
u_v_w_p; Imp/Imp	1.0x
uvw_p; Exp/Exp	0.7x

Open Literature Works

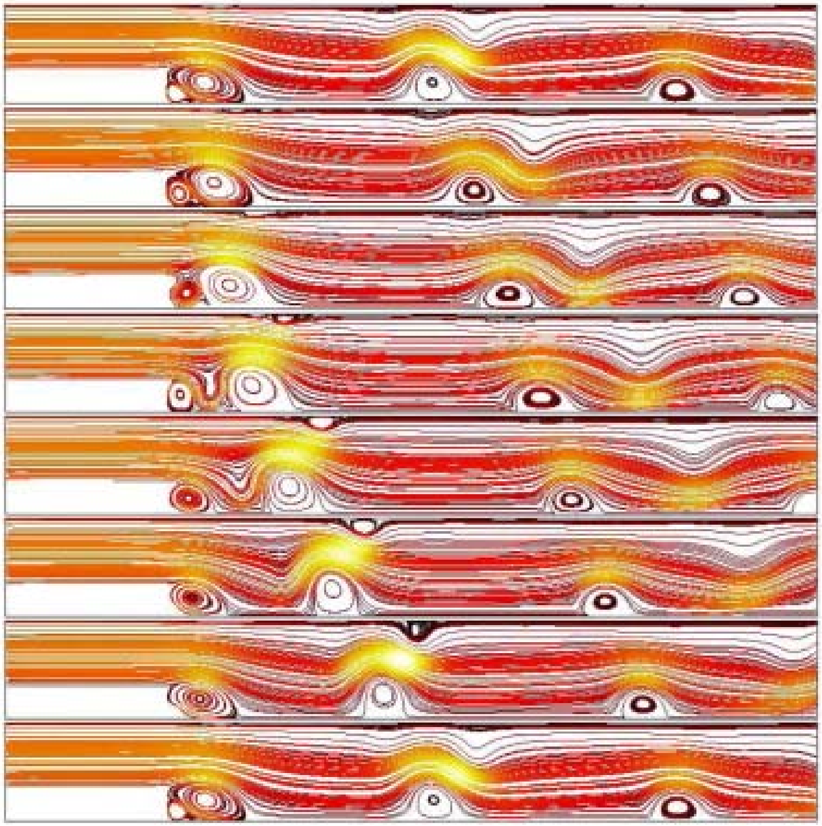
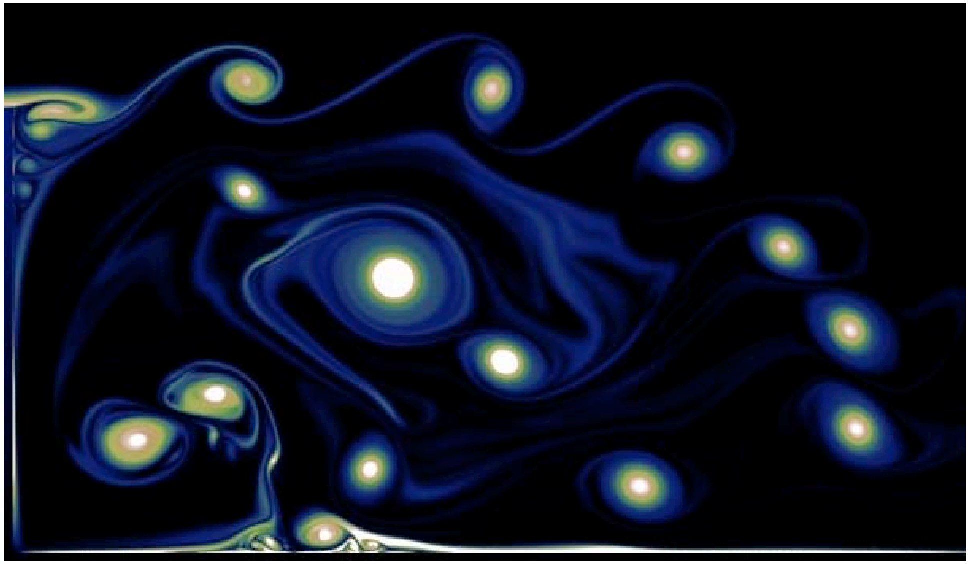


Fig. 18. Periodic evolution of streamlines from $t = 3.04$ s to $t = 3.11$ s.

- Hachem et al. JCP 229:23, 2010; monolithic stabilized FEM (40k tri)

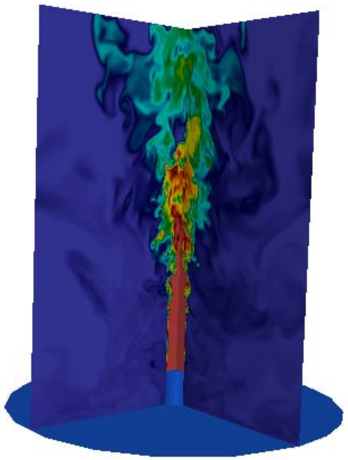
- Re 45k turbulent back step



- Domino; approximate pressure projection with KE preserving operators (8000k tri elements)

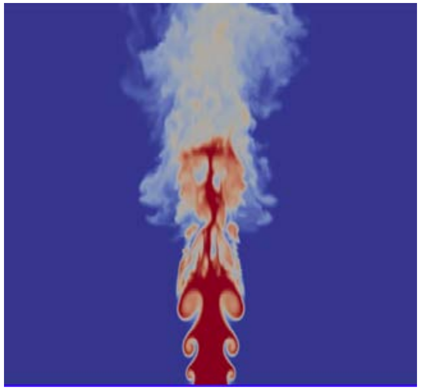
Performance Problem of Interest

- The three dimensional test problem of interest that has been used for this scaling study effort is a turbulent open jet (Re = 6,600) of Abdel et al. (1997)



❑ Re = 6,600 3D mesh unstructured hex mesh

❑ Re = 6,600 turbulent jet (volume rendered mixture fraction field)



❑ 2D plane (mixture fraction)

Physics of Interest

Sandia National Laboratories

- The variable density, low Mach set of equations are solved in which the acoustics have been filtered, thereby, allowing density to be a function of the spatially constant, possible variable in time thermodynamic pressure

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_j}{\partial x_j} = 0$$

$$DOFs = \tilde{u}_x, \tilde{u}_y, \tilde{u}_z, p, \tilde{z}$$

$$\frac{\partial \bar{\rho}\tilde{z}}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_j\tilde{z}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\bar{\rho} D \frac{\partial \tilde{z}}{\partial x_j} - \tau_{zu_j}\right)$$

$$\frac{\partial \bar{\rho}\tilde{u}_i}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_j\tilde{u}_i}{\partial x_j} = -\frac{\partial}{\partial x_i} p + \frac{\partial}{\partial x_j}\left(\overline{\tau_{ij}} - \tau_{u_i u_j}\right) + (\bar{\rho} - \rho^r) g_i$$

Turbulence closure models required for turbulent diffusive flux vector and subgrid stress tensor

$$\bar{\rho} = \frac{1}{\dfrac{\tilde{z}}{\rho(\tilde{z}=0)} + \dfrac{(1-\tilde{z})}{\rho(\tilde{z}=1)}}$$

- ❑ Regardless of coupling techniques (monolithic or pressure-projection) an elliptic pressure system is created

ASC

Evaluation of Current Code Timings

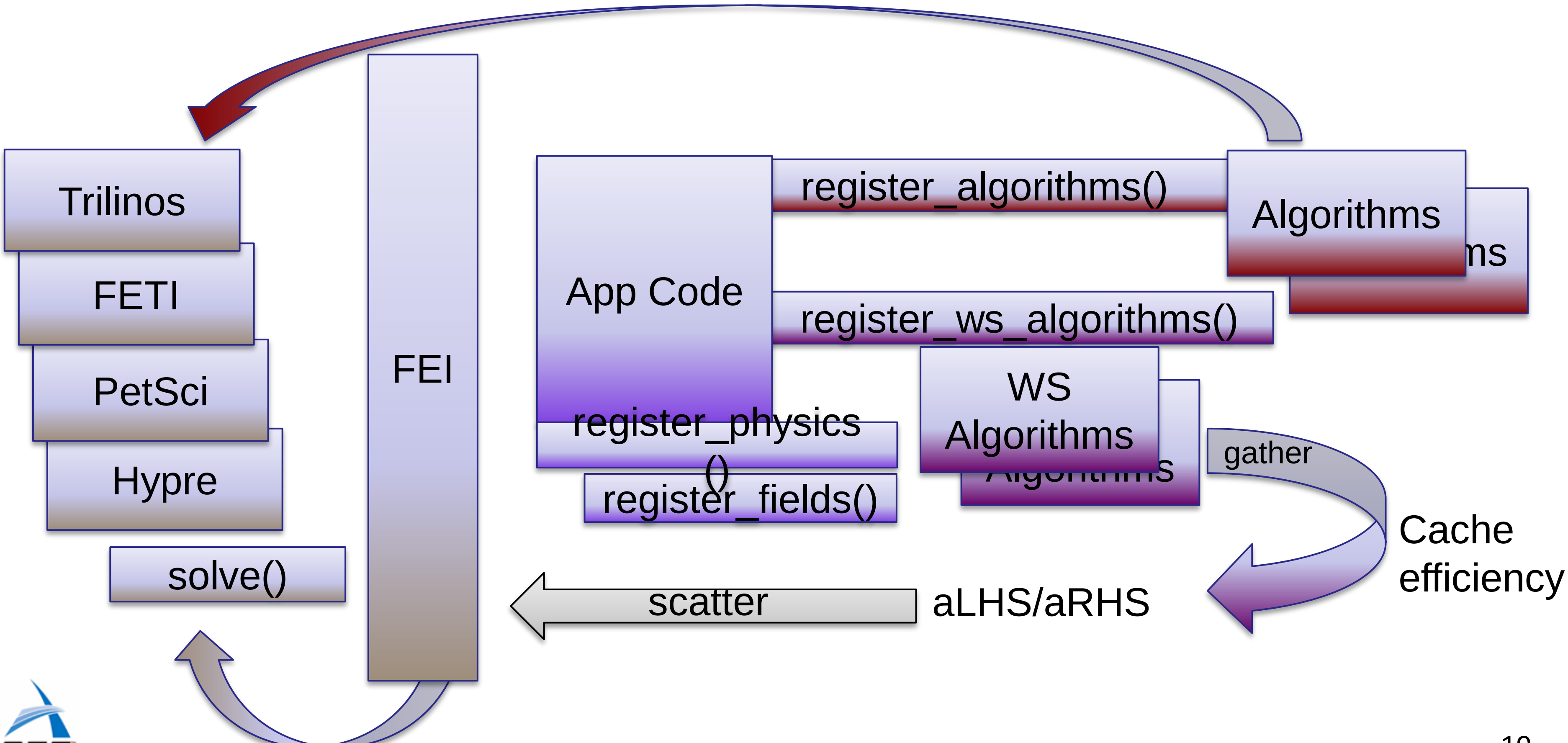
- Consider a typical mixture fraction-based LES for a transient simulation

Sum of Time	Column Labels					
Row Labels	Continuity	Mixture_Fraction	X_Momentum	Y_Momentum	Z_Momentum	Grand Total
Alloc LinSys	0.97	0.82	0.85			**2.63**
Initial Guess	5.01	4.08	4.60	4.59	4.60	**22.88**
Initialize	0.85	0.66	0.71	0.66	0.66	**3.53**
Load BC	0.00	0.04	0.10	0.09	0.09	**0.32**
Load Complete	8.94	9.47	9.62	9.39	9.49	**46.89**
Load Constraint	0.00	0.04	0.10	0.09	0.08	**0.31**
Load Contrib.	100.60	97.40	101.00	100.40	101.70	**501.10**
Reset	0.64	1.11	1.13	1.13	0.83	**4.84**
Scatter	4.18	3.44	3.69	3.78	3.84	**18.93**
Set RHS			2.92	2.92	2.89	**8.73**
Solve	306.10	22.22	18.24	18.17	18.78	**383.51**
Grand Total	**427.28**	**139.28**	**142.94**	**141.20**	**142.95**	**993.66**

- ❑ Solve and assembly time dominates

Possible Bottlenecks to Evaluate

Code abstractions for the purpose of code generality is good, right?



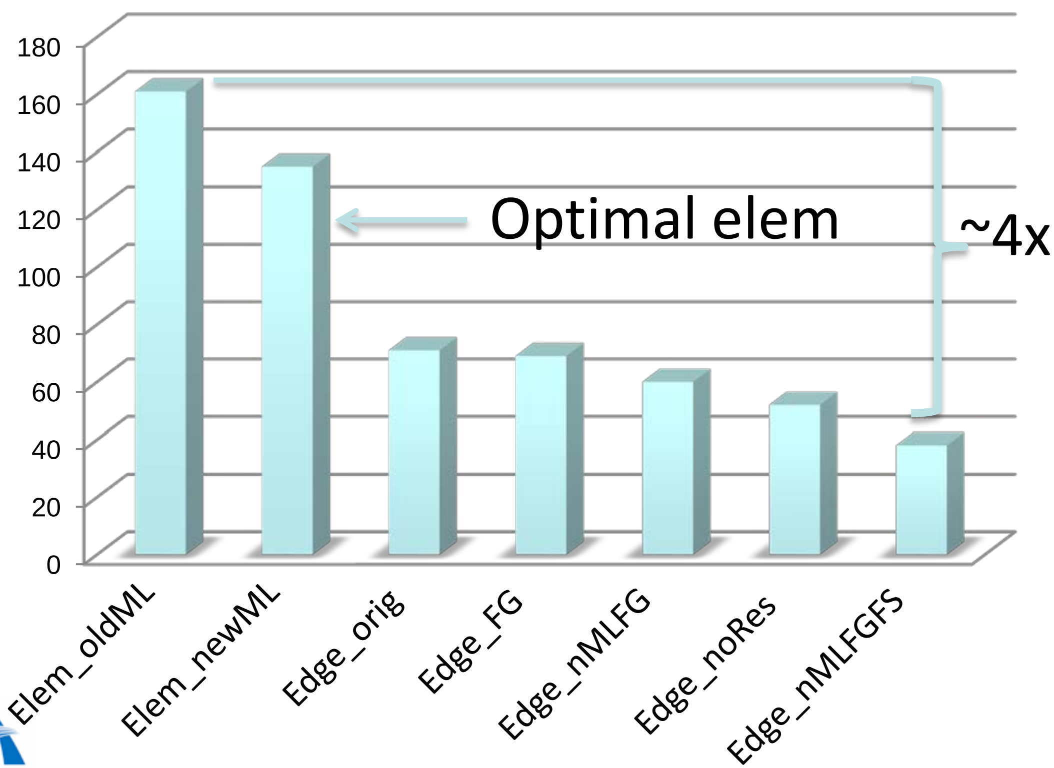
New Fast Gathers/Scatters

- New gathers/scatters have fewer instructions, fewer memory hops and fewer cache misses. Gathers dropped from 10s to <1 s

Call Stack	CPU Time▾	CPU Time:Total
▽sierra::Acon::UnitMech::solve	0s	44.507s
▽sierra::Acon::UnitMech::assemble	0s	24.370s
▷stk::diag::Timer::Timer	0s	0.010s
▽sierra::Eqns::LinearSystem::load_contributions	0s	24.070s
▷sierra::Fmwk::WorksetAlgorithm::execute	0s	4.343s
▽sierra::Acon::ScalarEdgeSolverWS::execute	0s	16.012s
▽sierra::Fmwk::WorksetAlgorithm::execute	0s	16.012s
▽sierra::Fmwk::WorksetAlgorithm::drive_workset	0.060s	16.012s
▽sierra::Acon::ScalarEdgeSolverWS::apply	1.283s	15.953s
▷sierra::Eqns::LinearSystem::apply_coefficients	1.808s	12.084s
q_edge_	1.498s	1.498s
sierra::Acon::GatheredData<double>::gather_edge_averaged_data	0.937s	0.937s
sierra::Acon::Acon_EqnsLinearSystem::apply_coefficients	0.080s	0.080s
nse3d_	0.050s	0.050s
sierra::Diag::Trace::Trace	0.010s	0.010s

Edge-based Timing History

- History of Edge-based timing compared to Element-based scheme for the mixture fraction-based open jet simulation (17 million element; 128 core)



- Elem
  - CVFEM code base
- Edge
  - EBVC code base
- oldML
  - Default ML
- nML
  - ML settings from 2008
- FG
  - Fast gather
- noRes
  - No reset of cont equation
- FS
  - Fast scatter

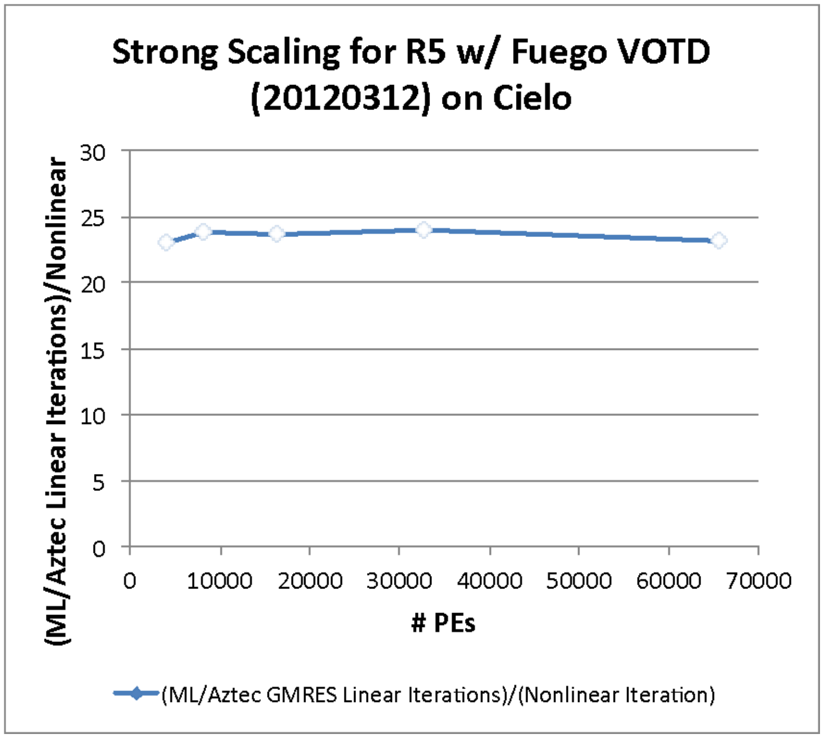
Scaling Studies

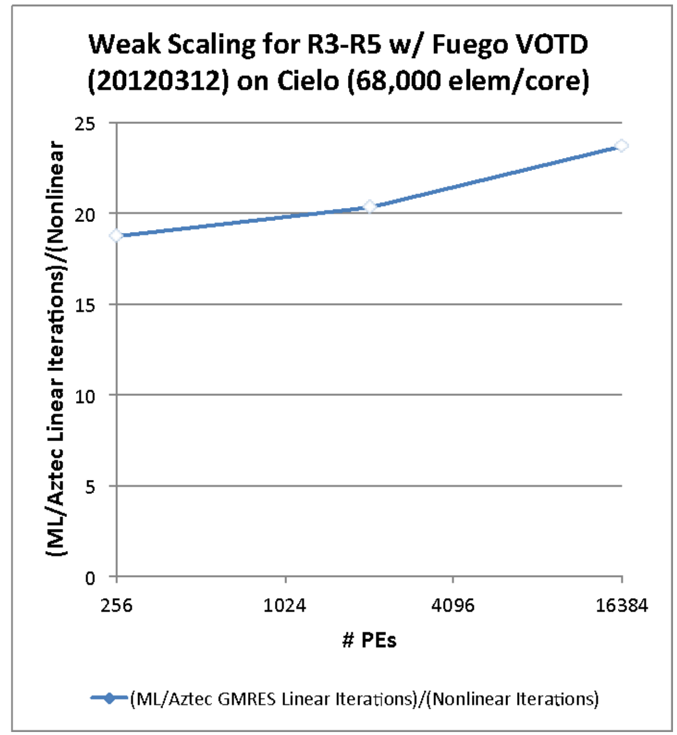
- Cielo scaling studies for mixture fraction-based turbulent open jet problem (Re=6,600)
- Sequence of meshes:
  - R3 (17.5 million elements; 64 – 4,096 cores)
  - R4 (140 million elements; 512 – 16,384 cores)
  - R5 (1.12 billion elements; 4,096 – 65,536 cores)
- Linear Solve options
  - Continuity: GMRES/ML
  - Scalars: GMRES/SGS
- Element-based algorithmic studies: R3 – R5
  - Internal code name "Fuego"
- Edge-based algorithmic studies: R4
  - Global ID size impediment due to signed int limitation
  - Internal code name "Conchas"

Cielo Details

- Cielo; a NNSA DOE resource ~1.37 petaflop
- Cray-based machine (XE6) built in Spring of 2010
  - 2 GB per core
  - Cray Gemini high-speed interconnect
- PGI, Cray, Intel and GNU compiler suites
- Design, procurement and deployment were accomplished by the NNSA's New Mexico Alliance for Computing at Extreme Scale (ACES)
  - Joint partnership between Los Alamos National Laboratory and Sandia National Laboratories

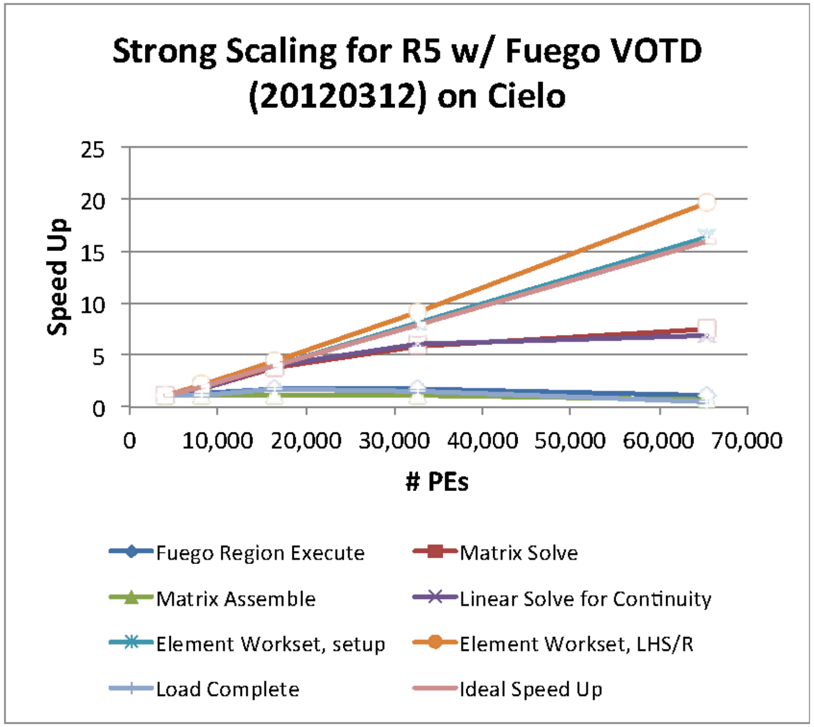
ML Algorithmic Scaling Performance

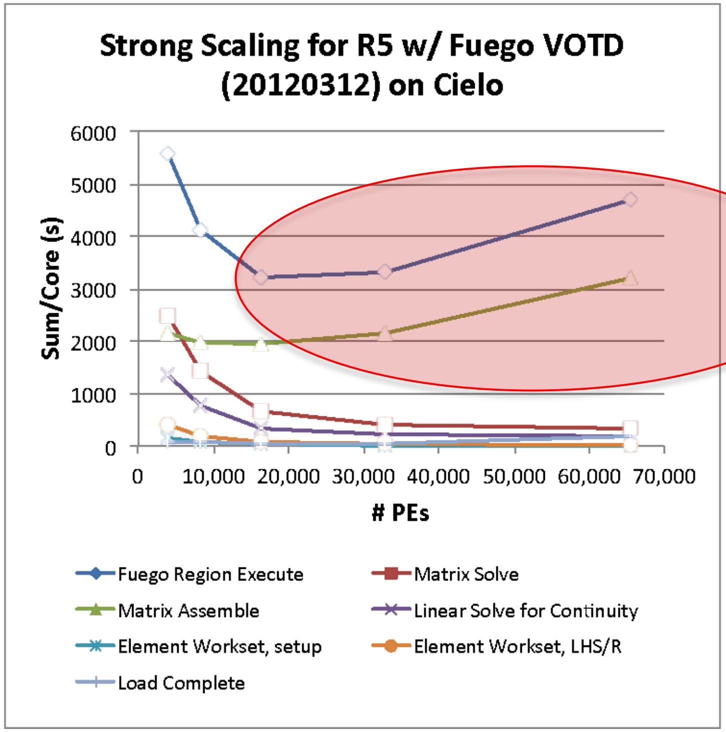




- Strong scaling for R5 mesh
- Weak scaling

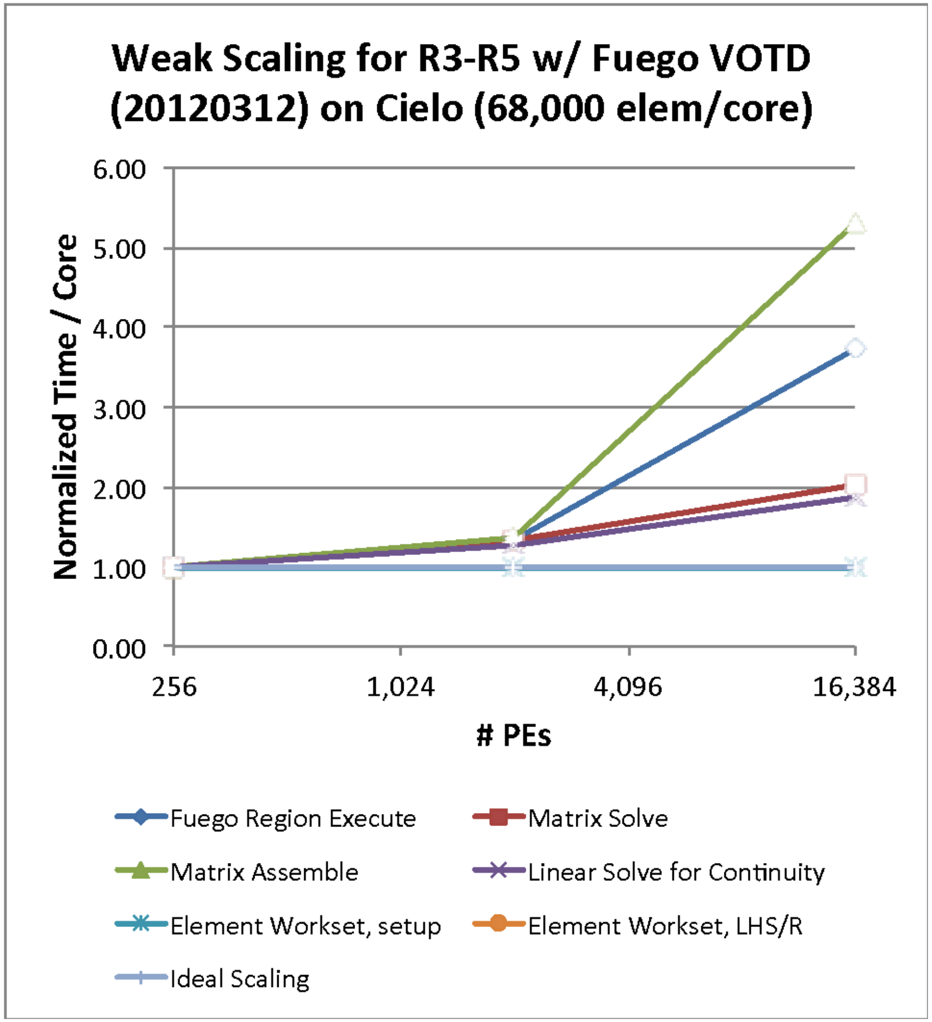
R5 Element-based Strong Scaling



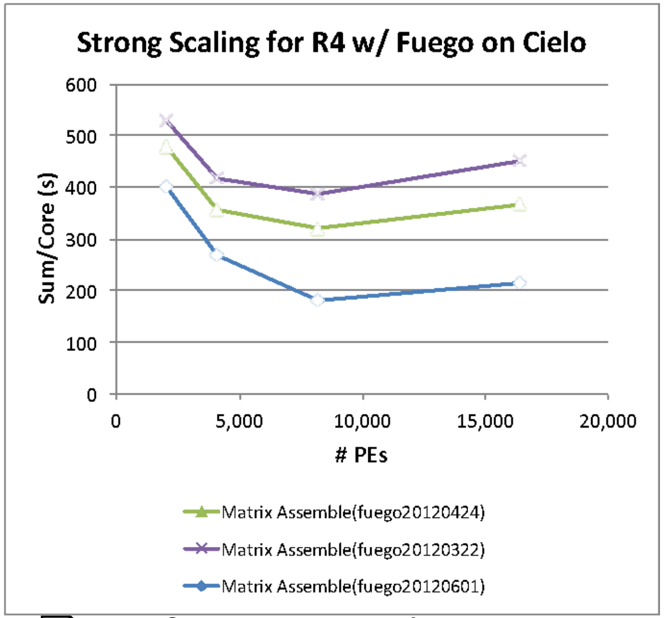


- ❑ Base for speed up is 4096 cores
- ❑ Time per core

R3-R5 Element-based Weak Scaling



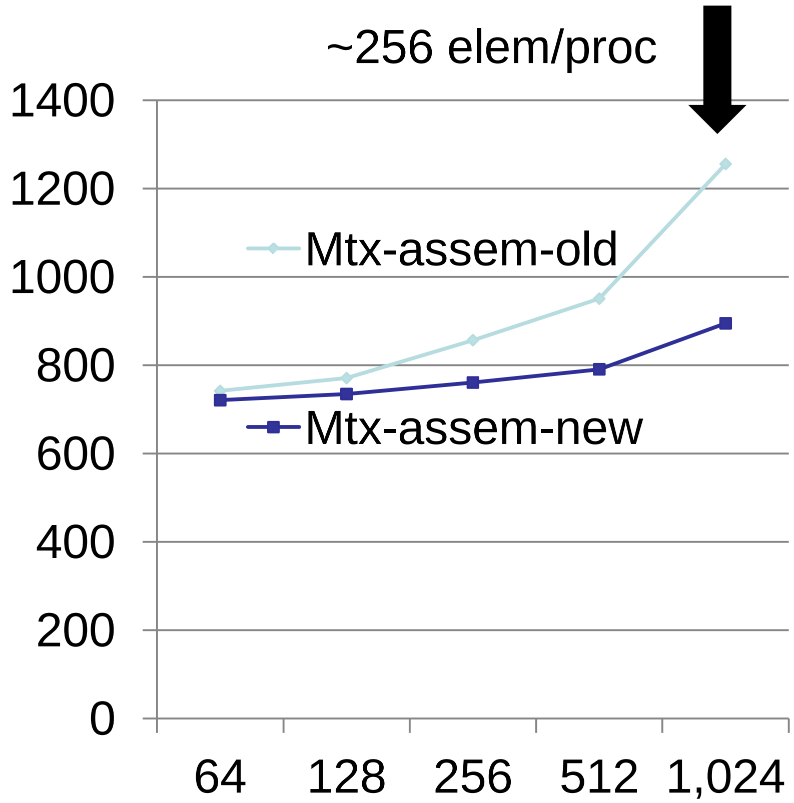
- Time per code normalized by 256-core simulation time
- Scaling of overall matrix assembly is in need of improvement as ideal scaling is expected
- Matrix solves also are non-optimal

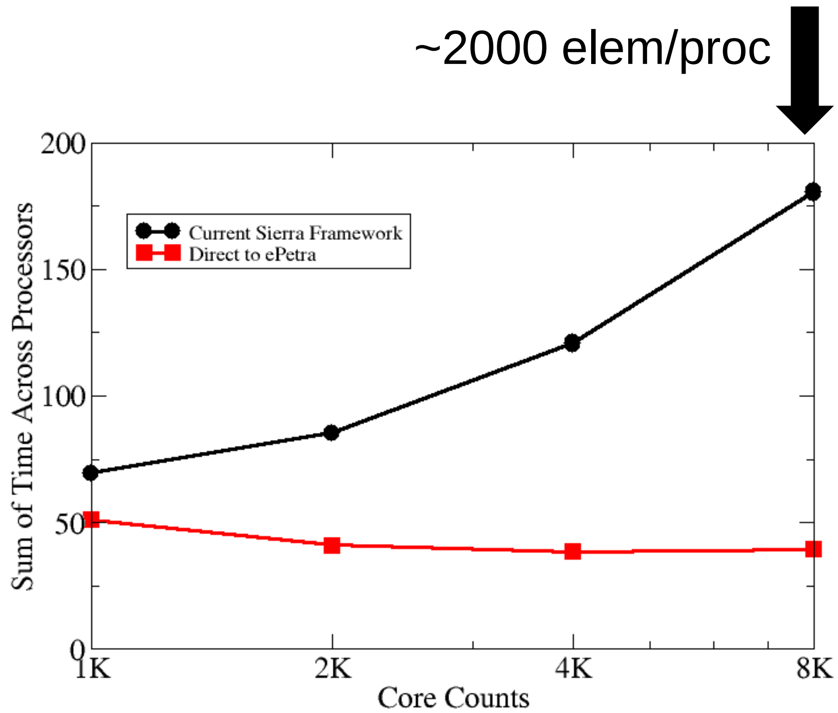


- Performance enhancement

Resolving Matrix Assembly Scaling

- Matrix assembly is expected to be optimal (close)





- ❑ Strong scaling for R2 mesh; FEI
- ❑ Strong scaling for R5 mesh; non-FEI

Conclusions

- Strong and weak scaling studies have been performed on meshes ranging from 17 million to 1.12 billion elements on core counts up to 65,536
- Various code design principles have been evaluated including software abstractions designed for the purposes of code generality
- Evaluated three discretizations with a variety of coupling paradigms to define optimal scheme for a typical LES application space
- Edge-based low Mach discretization has been shown to be second order accurate and almost ~4x faster than the current element-based approximate projection method