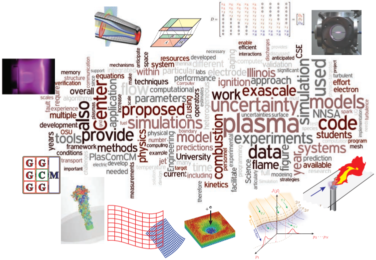
XPACC:
The Center for Exascale Simulation of Plasma-coupled Combustion

Bill Gropp
Jon Freund

Parallel Computing Institute (PCI)

Computer Science
Mechanical Science & Engineering
Aerospace Engineering

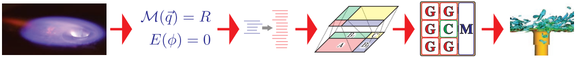


Overarching Problem

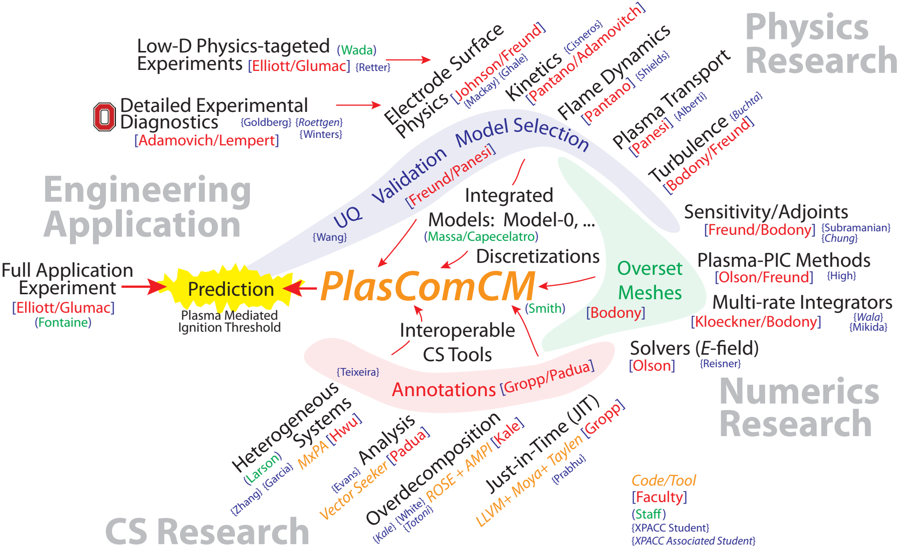
- Combustion is our energy workhorse and will remain so for the foreseeable future

- Truly predictive simulations will accelerate fundamental advances in the use of plasmas to advance combustion technology

- The computer science advances that enable such massive-scale predictive simulations will have impact across engineering and science

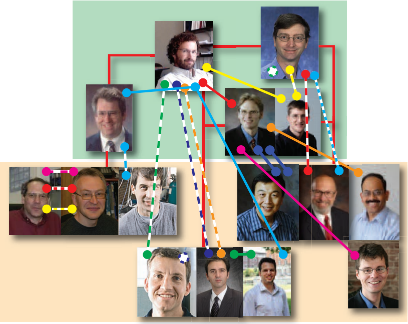


Overall Integrated Effort



Integrated Student Projects

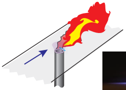
Coordination & co-advising fosters an integrated effort

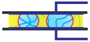


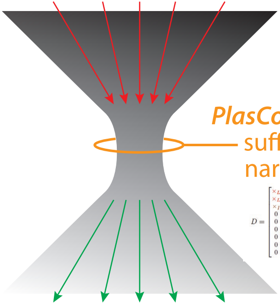
- Discrete Adjoints (Vishnampet)
- PIC in *PlasComCM* and at Exascale (High)
- Plasma Trasport Modeling (Alberti)
- Reduced Chemistry with UQ (Wang)
- Multi-hardware Programming (Garcia/Zhang)
- Multi-timestepping (Mikida)
- Runtime Over-decomposition (White)
- Autotuning/HTA for *PlasComCM* (Evans)
- Elliptic Solves at Scale (Reisner)
- Multiscale MD of Electrode Aging (MacKay)
- Plasma Actuator Design and Diagnostics (Retter)
- Chemical Reduction with UQ (Cisneros)
- *E*-field Measurement (Goldberg)
- Sub-grid-scale Plasma Comb. Models (Shields)
- Trace Species Measurements (Winters)
- Electron Density and T (Roettgen)
- JIT Code for Adaptive Runtime (Prabhu)
- Annotations to Support Autotuning (Teixeira)
- Damaged Electrode Workfunctions (Ghale)

Performance at Scale: Overall Picture

PHYSICS MODELS
plasma
flow/turbulence
combustion





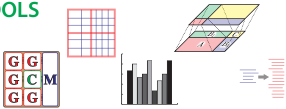


***PlasComCM* OPERATORS**
sufficiently flexible for multi-physics objectives
narrowly defined to facilitate performance tools

$$D = \begin{bmatrix} \times_L & \times_L & \times_L & \times_L & \times_L & 0 & 0 & 0 & 0 \\ \times_L & \times_L & \times_L & \times_L & \times_L & 0 & 0 & 0 & 0 \\ \times_I & \times_I & \times_I & \times_I & \times_I & 0 & 0 & 0 & 0 \\ 0 & \times_I & \times_I & \times_I & \times_I & \times_I & 0 & 0 & 0 \\ 0 & 0 & \times_I & \times_I & \times_I & \times_I & \times_I & 0 & 0 \\ 0 & 0 & 0 & \times_I & \times_I & \times_I & \times_I & \times_I & 0 \\ 0 & 0 & 0 & 0 & \times_I & \times_I & \times_I & \times_I & \times_I \\ 0 & 0 & 0 & 0 & \times_R & \times_R & \times_R & \times_R & \times_R \\ 0 & 0 & 0 & 0 & \times_R & \times_R & \times_R & \times_R & \times_R \end{bmatrix} = \begin{bmatrix} D_L \\ D_I \\ D_R \end{bmatrix}$$

```
for(i = 0; i < Ns; i++) {
     rhs(:,i) = 0;
     rhs(:,i) += ApplyOperator(D_x,-cv(:,1)*(u(:,1) + V(:,i,1))*Y(:,i),ASYNC);
     rhs(:,i) += ApplyOperator(D_y,-cv(:,1)*(u(:,2) + V(:,i,2))*Y(:,i),ASYNC);
     rhs(:,i) += ApplyOperator(S_i,[rho T Y],ASYNC);
     WaitOperators();
}
```

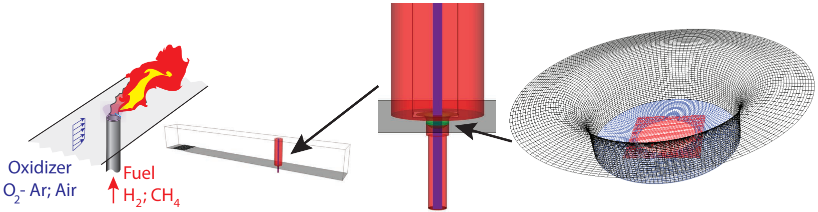
INTEROPERABLE (PRE-)EXASCALE TOOLS
scalable algorithms
overdecomposition
runtime adaptation
source-to-source transform



PlasComCM seen from the application side

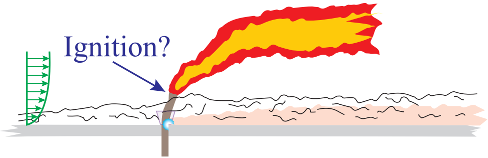
Application Code: *PlasComCM*

- **Locally structured—globally unstructured. . .**

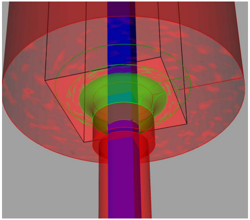


- Current:
  - Representing geometries
  - Flexible stencils (high/low-resolution, conservation)
- Planned:
  - Adaptivity (multiresolution, variable stencils)
  - Multi-rate (multi-Δt's)
  - Preconditioning (variable stencils)

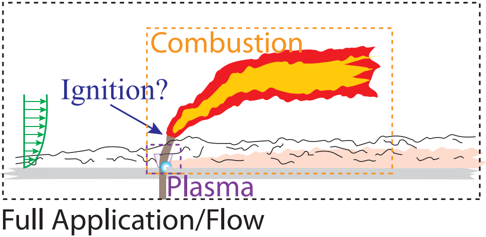
PDE-Governed Fields



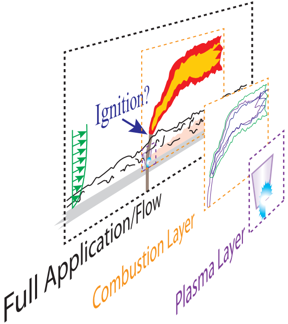
- Current:
  - DNS & Dynamic SGS turbulence
  - Hydrodynamic plasma
  - Chemistry/transport-coupled (*Cantera*)
  - Electrostatic *E*-field solver



PDE-Governed Fields

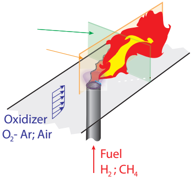


- Toward: overset layers for physics-dictated space–time resolution needs



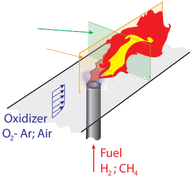
PlasComCM seen from CS side

Code Built On Discrete Operators



- Physical models: $\partial/\partial t$, $\partial/\partial x_i$, $\partial^2/\partial x_i \partial x_j$, ∇^{-1}, $\iint f_{\mathrm{col}}(\theta, \phi)\, \mathrm{d}\theta \mathrm{d}\phi$
- Exascale access: speed, memory locality, scalability, power usage, cache blocking, . . .

Code Built On Discrete Operators



- Physical models: $\partial/\partial t$, $\partial/\partial x_i$, $\partial^2/\partial x_i \partial x_j$, ∇^{-1}, $\iint f_{\mathrm{col}}(\theta, \phi)\, d\theta d\phi$
- Exascale access: speed, memory locality, scalability, power usage, cache blocking, ...

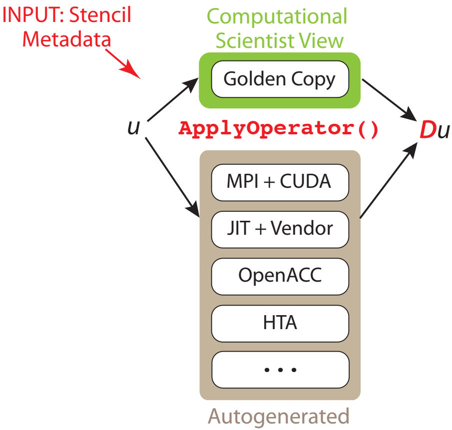
Operators *unify* these two sets of needs:

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial x^2} + f(u) \quad \underset{\{x_i\}_{i=1}^N}{\overset{\text{disc.}}{\Longrightarrow}} \quad \frac{d\vec{u}}{dt} = -a\mathbf{D_x}\vec{u} + \nu\mathbf{D_{xx}}\vec{u} + F[\vec{u}]$$

- The models care about $\vec{v} = \mathbf{D}_{[\mathbf{x}|\mathbf{xx}]}\vec{u}$
- CS exascale tools care about $\mathbf{D}_{[\mathbf{x}|\mathbf{xx}]}\vec{u}$

Operators

- Code separates operator *use* from *declaration*
- Opportunity to express additional/alternative parallelism
- Different possible routes hidden from computational scientist



Computer Science Approach

Overarching Approach

- **Scalable numerical algorithms** Exposing the amounts and kind of parallelism is key to enabling CS techniques
- **Overdecomposition** Divide data structures into more than one chunk per core to provide load balancing (redistribute), performance (match memory hierarchy), latency hiding (schedule), fault tolerance (unit of repair), heterogeneity (schedule)
- **Adaptive Computation** Have the code at runtime adapt runtime execution to respond to compile *and runtime* information about progress, efficiency
- **Source-to-source transformations** Allow the computational scientist to focus on expressing the algorithm and use tools to provide customized versions of code for different situations and platforms. *Always preserving the golden copy.*

Composition of Solutions

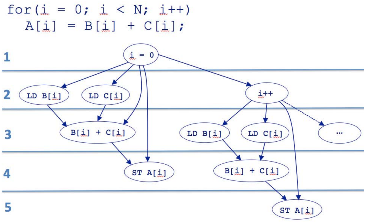
Rather than one do-everything solution (*e.g.*, a new parallel language), build *interoperable tools* that address one *or more* issues

- Enables exploitation of tools developed for commodity and smaller-scale systems (can plug into framework)
- Mitigate risk by reducing single points of failure
- Increases potential for adoption by enabling incremental adoption
- Addresses complexity by factoring methods and services (complexity becomes primarily additive rather than multiplicative)
- Time frame: must have usable tools within life of project

2 Sample Tools & Results

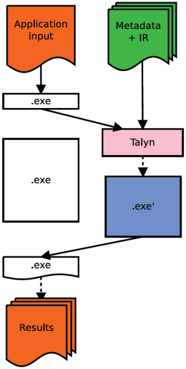
Vector Seeker

- Using *tracing* to find maximum parallelism on vector-ideal machine
- Identify missed opportunities
- Based (for now) on *Intel Pin*
- Cost: $\sim 100\times$ std. runtime
- Applied to full *PlasComCM*:
  - 2499 Lines considered
  - 748 Fully vector across whole program
  - 219 Partially vector across whole program
- Observations:
  - Most lines with significant execution counts are vectorizable
  - Many lines that are not vectorizable come from utility functions that may be vectorizable when not aggregated



JIT for *PlasComCM*

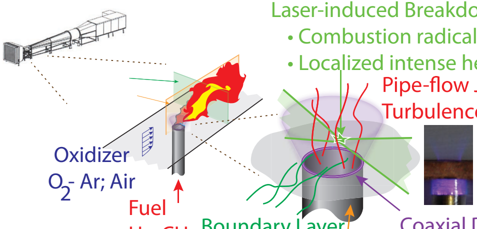
- **Just-in-time (JIT)** compilation:
  - *Static* component *Moya*: identifies potential “hot” code for input-dependent optimization (with programmer assistance)
  - *Dynamic* component *Talyn*: re-compiles with optimization for static metadata and executes



- LLVM used for both components
- Static component: currently identifies finalized variables
- Dynamic component: currently run-time constant folding
- Single-core performance: +15% with only run-time constant folding
- Polyhedral optimizations in development

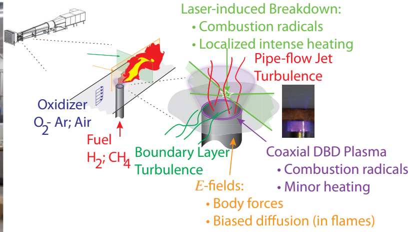
Application: Year-1 Prediction Target

Target Application



- Predict the ignition threshold of a jet in crossflow via thermal discharge and dielectric-barrier-discharge (DBD) plasmas
- Canonical combustion flow with new 'knobs' for mediating ignition

Full Application



Tunnel test section	38 cm × 38 cm × 121 cm
Cross-flow	5–15 m/s
Fuel jet	1–10 m/s
Jet diameter	4.7 mm
Fuel	H_2 (later CH_4)
Plasma #1	non-thermal: co-annular DBD
Plasma #2	thermal: laser breakdown

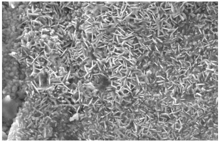
Coupled Physics

- Turbulent mixing: fuel–oxidizer
- Plasmas
  - Joule heating: initiates reactions, alters flow via expansion
  - generated radicals ‘short-circuit’ reaction pathways
  - advected by flow
- *E*-fields
  - alter diffusive radical transport: affect thin flame structure
  - force radicals: can alter flame even for standard pathways
- Electrodes: aging/wear alters plasma (via e^- workfunctions)
- Huge range of important length (and time) scales:

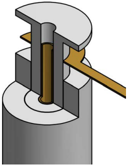
Wind Tunnel ($\sim$ m)

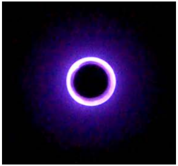
Fuel Jet ($\sim$ cm)

Electrode Wear ($\sim \mu$m)



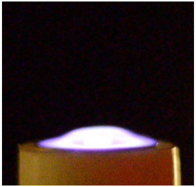
Physics-Targeted: DBD Effect on Flames





Plasma Off

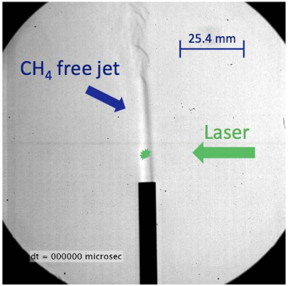
Plasma On



- H_2 stand burner
- $O(1)$ effect on flame
- Incorporated into wind tunnel

Physics-Targeted: Laser Breakdown Ignition

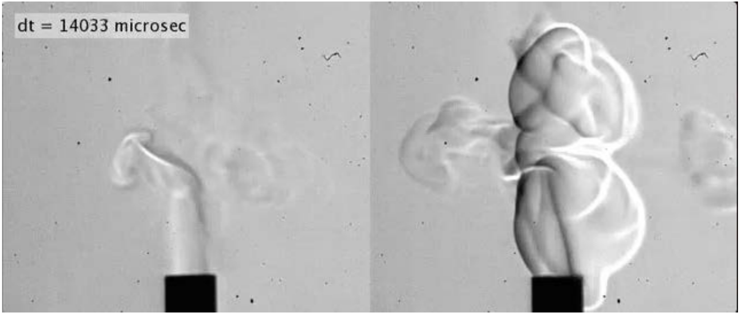
- Focused laser ⇒ breakdown ⇒ ignition seed
- Surrogate for pin-to-pin spark; proposed use in large engines



- Measurements of thermal deposition ⇒ thermal source model
- Calibration of aleatoric character...

Physics-Targeted: Laser Breakdown Ignition

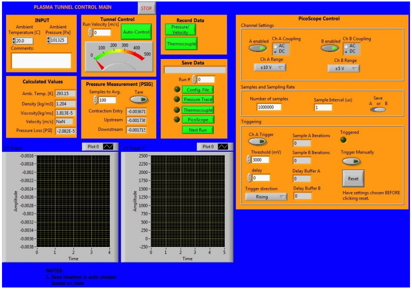
- Nominally identical conditions:



Unified Measurement Protocol

- Automated documentation of each test
- CAD drawing of wind tunnel and actuator configuration
- Experimental parameters:
  - plasma parameters
  - in-flow conditions
  - synchronized to measurements (PIV, LIF)
- Upload data to the dedicated XPACC cluster

Custom XPACC LabView VI



Ongoing Simulations

Simulation Status

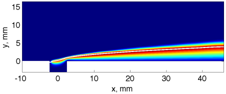
- Physics based Model-0 set for year-1 prediction ✓
  - Resolved (DNS) compressible-fluid turbulence
  - Hydrodynamic plasma model
  - Minimal 4-reaction calibrated global chemistry
  - Electrostatic *E*-field (Poisson solve)
- 2-D fuel jet in cross flow ✓
  - platform for first integration of minimal physics
- 3-D turbulent boundary layer ✓
  - establish realistic turbulence simulation 'baseline'
- Running: nearing statistically stationary "igniteable" state

Preliminary Simulation (2D)

- 2-D burning H_2 jet in cross flow
- Qualitative agreement with wind-tunnel observation:

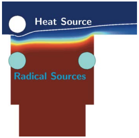
EXPERIMENT – Light Emission

SIMULATION — Temp.

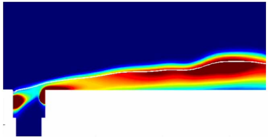


- Goals: integration of basic mechanisms
  - Spark/Breakdown: heat source
  - DBD Plasma: radical production, body force
- Shows expected power, radical, *E*-field, *etc.* sensitivities

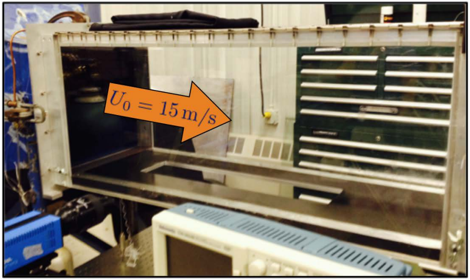
Model Sources

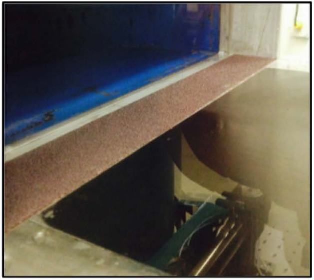


H concentration



Wind Tunnel Turbulence Conditions

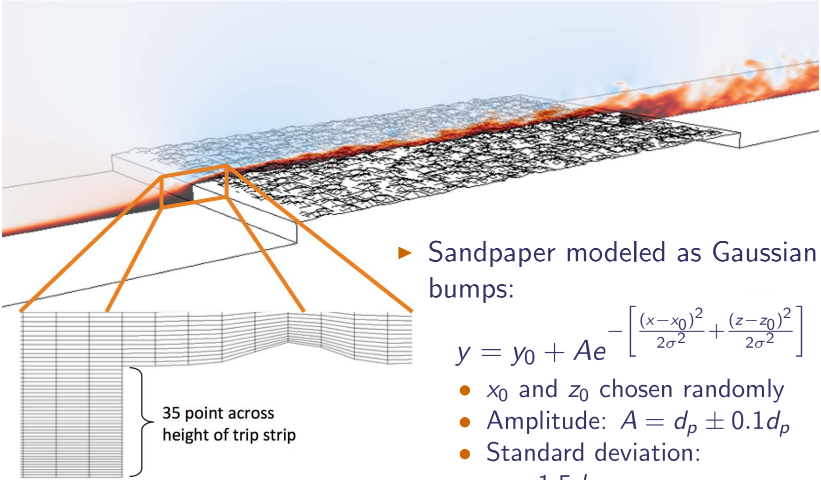




- Fuel jet diameter 0.5 cm
- Boundary layer thickness
  - $\delta = 1.2\,\mathrm{cm}$
  - $\theta = 0.13\,\mathrm{cm}$
- Reynolds numbers
  - $\mathrm{Re}_\delta = 13065.2$
  - $\mathrm{Re}_\theta = 1455.4$

- Inflow trip strip
  - 40–grit sandpaper
  - Width: 5.1 cm
  - Height: $\Delta = 1.64\,\mathrm{mm}$ $(\Delta^+ = 80)$
  - Grains: $d_p = 420\,\mu\mathrm{m}$ $(d_p^+ = 20)$

Boundary Layer Transition Trip

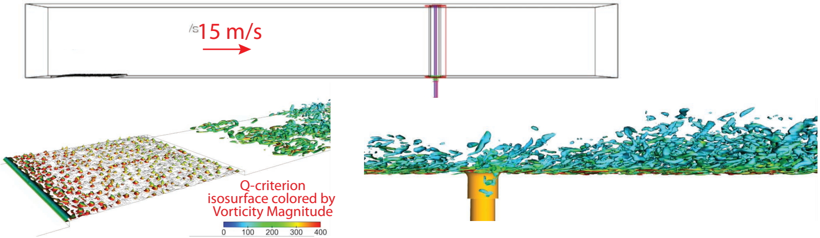


- Sandpaper modeled as Gaussian bumps:

$$y = y_0 + Ae^{-\left[\frac{(x-x_0)^2}{2\sigma^2} + \frac{(z-z_0)^2}{2\sigma^2}\right]}$$

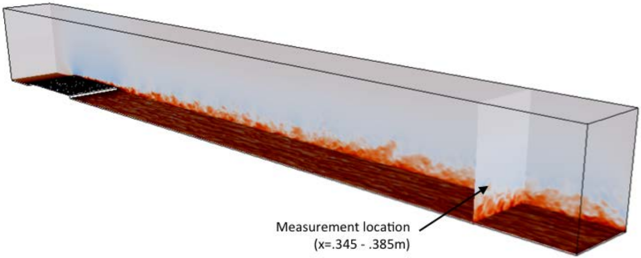
  - x_0 and z_0 chosen randomly
  - Amplitude: $A = d_p \pm 0.1 d_p$
  - Standard deviation: $\sigma = 1.5 d_p$

Jet in Turbulent Crossflow

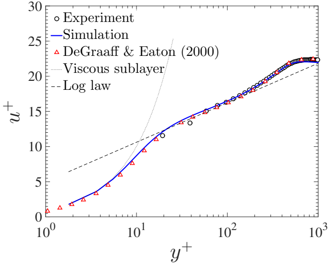


- 4 overset meshes
- 27 million mesh points
- modest size: 2 days, 1216 cores
- runs on Cab, Chama, Mapache, Vulcan, Wolf, ...

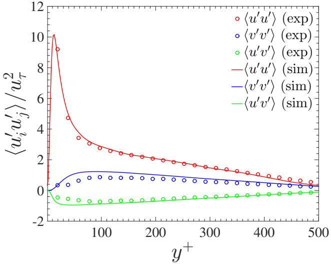
Velocity Statistics



Mean Velocity



Reynolds Stresses



Summary

- Year-1 prediction target:
  - plasmas showing $O(1)$ effect on ignition and flame
  - mechanism(s) unclear
- Year-1 simulations:
  - physics integration in 2-D phenomenological demonstration
  - inert neutral 3-D turbulence established
  - combined and running toward "ignitable" statistically stationary state
- CS tools applied to evolving *PlasComCM* 'Golden Copy':
  - extensive analysis: *Vector Seeker*
  - overdecomposition prototyping via *ROSE–AMPI*
  - JIT tools *Moya* + *Talyn* showing performance boost
  - Codelet for *MxPA* cross-platform (CPU–GPU) compiler

This material is based in part upon work supported by the Department of Energy, National Nuclear Security Administration, under Award Number DE-NA0002374.