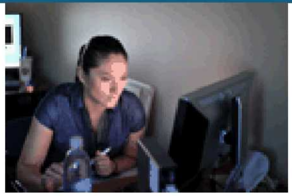
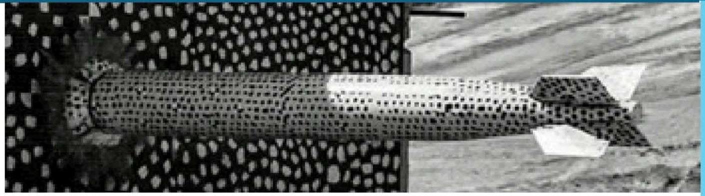
SAND2019-6226PE

Machine-Learned Models for Near-Wall Turbulence





Matthew Barone (1515), Nathan Miller (1515), Warren Davis (1461), Jeffrey Fike (1515)

Engineering Sciences LDRD, FY16-18

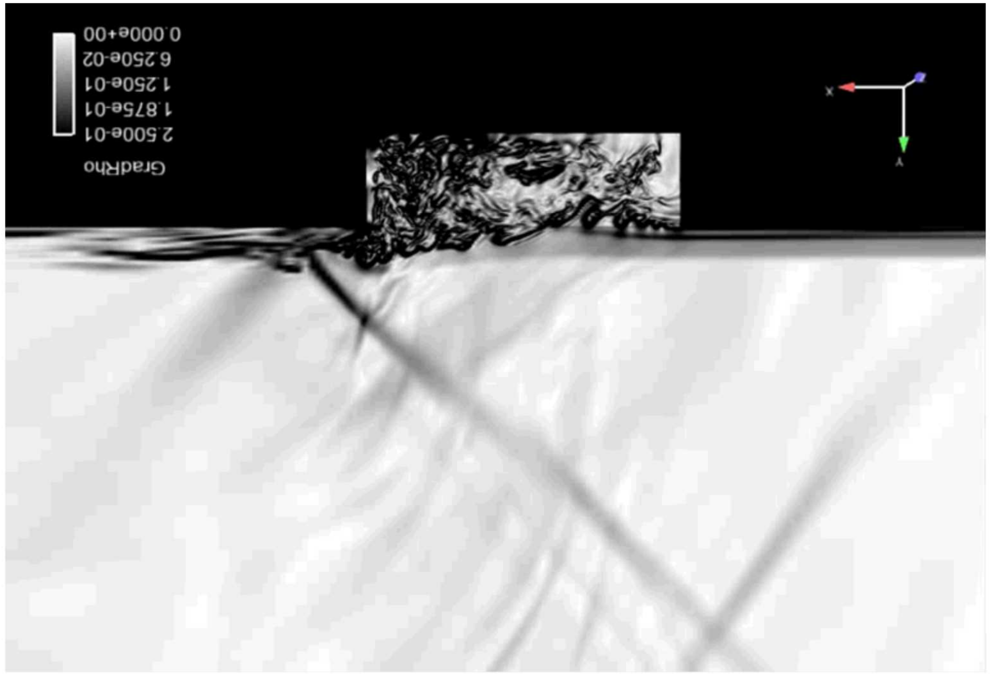
ASC V&V Project (Delivery), FY19

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

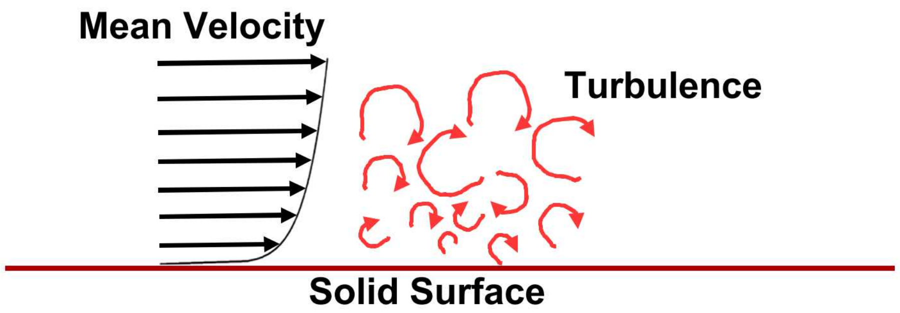
Motivation: Application Space

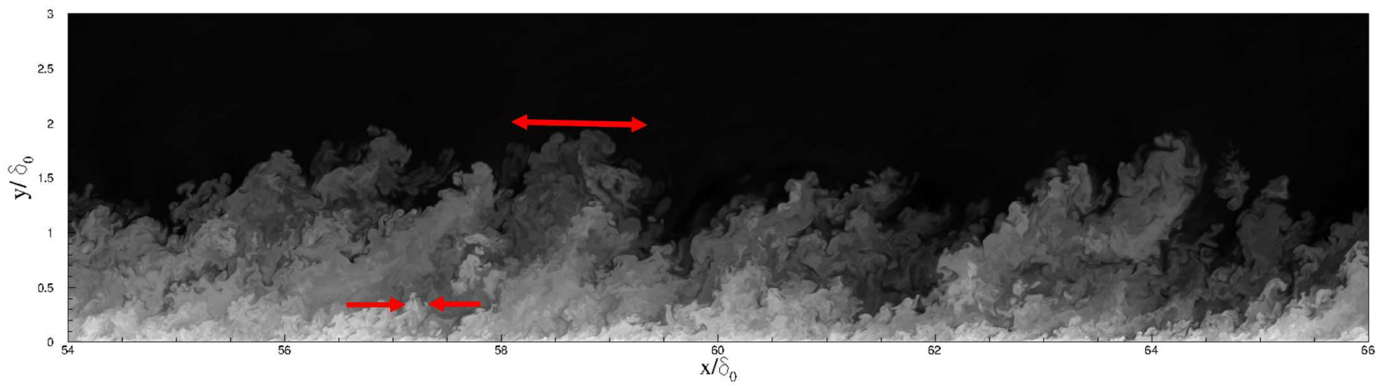
- Design environments for engineered systems of interest include **flow-induced vibrations**.
- Fluctuating pressure loads often:
  - result from turbulence
  - involve complex spatial fields
  - are non-stationary in time
  - are nonlinear
- **High-fidelity models** are often required for sufficiently accurate prediction.

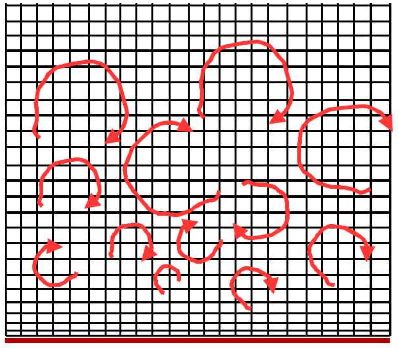
Photo from: SAND2014-17466O (UUR)

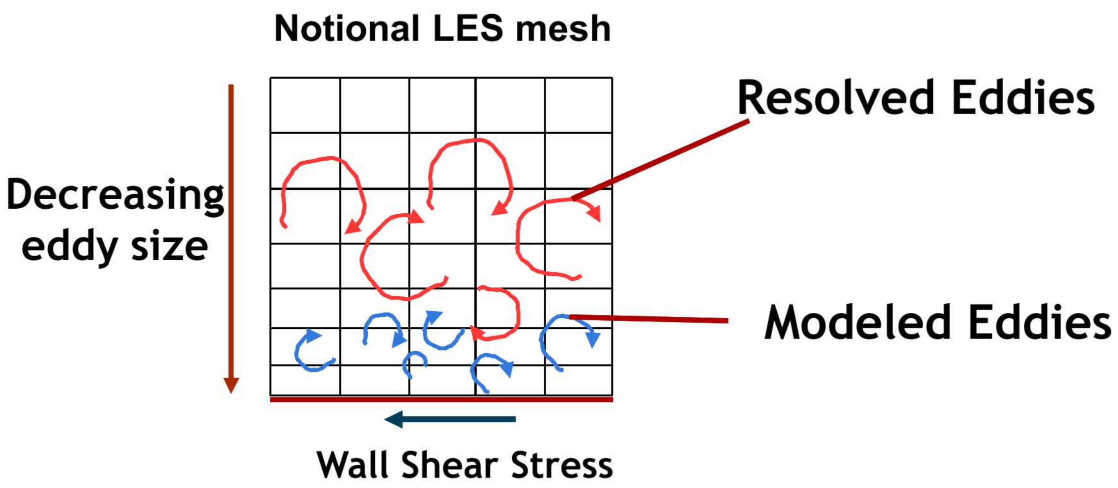


Near-wall Turbulence Modeling in Large Eddy Simulation







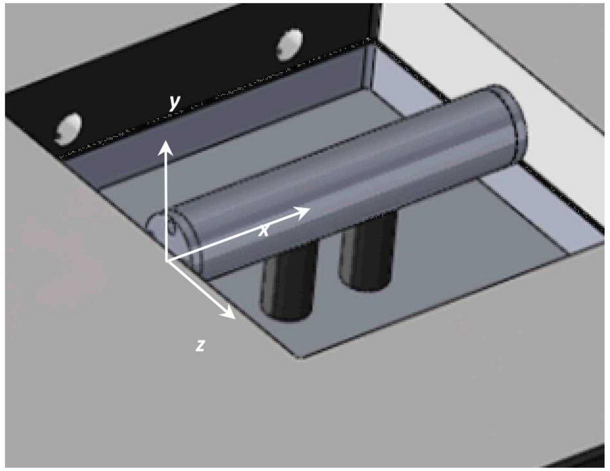


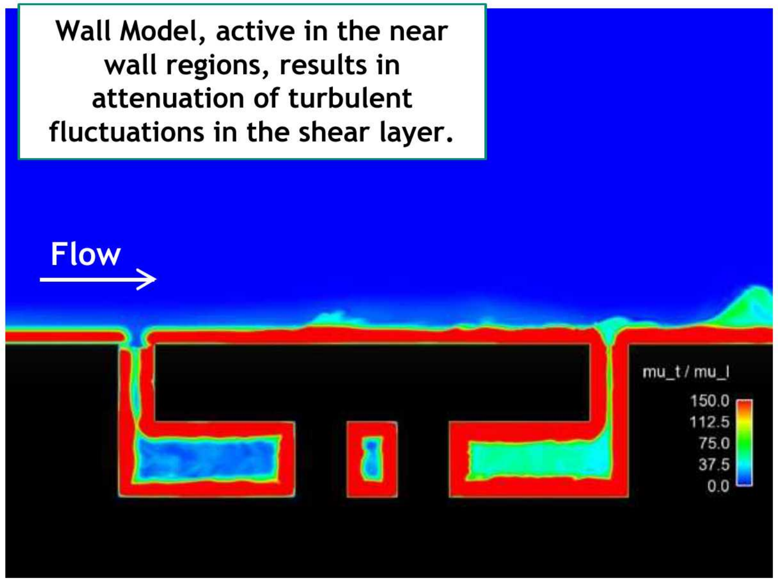
$$\boxed{\tau(\boldsymbol{x}, t) = F(\boldsymbol{U}, d, ?, ?, \dots)}$$

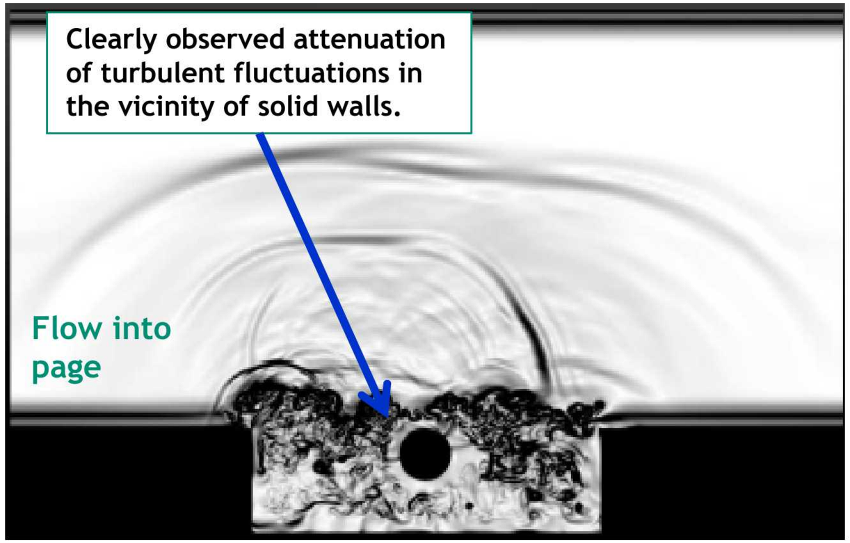
Wall-modeled LES offers computational savings decrease of *at least* several orders of magnitude over DNS for engineering systems of interest.

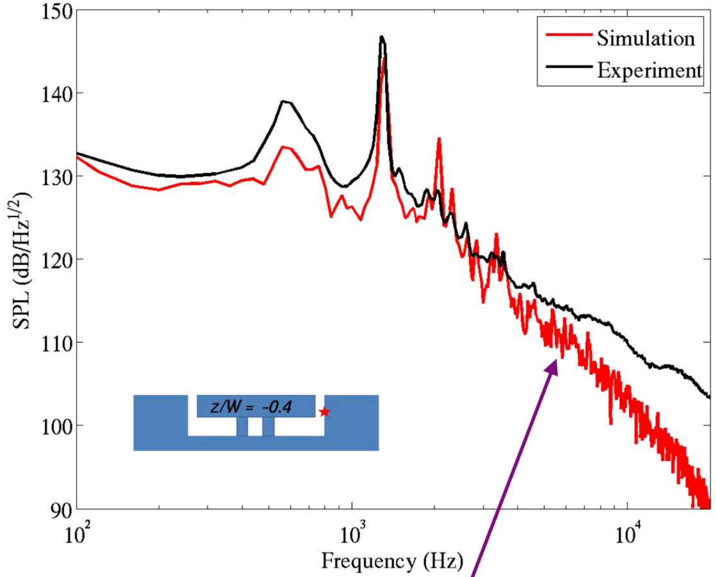
Motivation: Near-Wall Turbulence Model Deficiencies

Validation experiment: Model store within a cavity









Under-prediction of wall pressure fluctuations

The Research Question

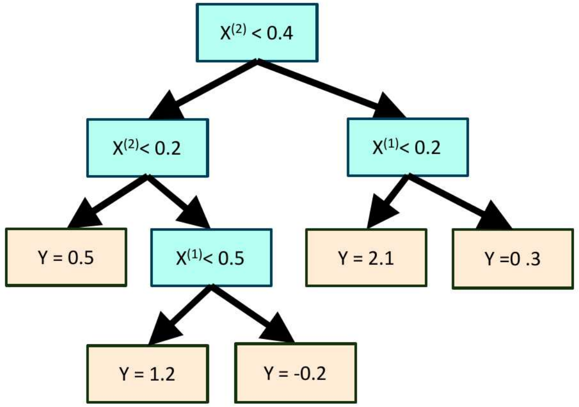
Key Premise

New near-wall turbulence models based on traditional approaches - theory, phenomenology, and limited calibration to data - will not result in significant improvements in predictive accuracy for surface loading simulations.

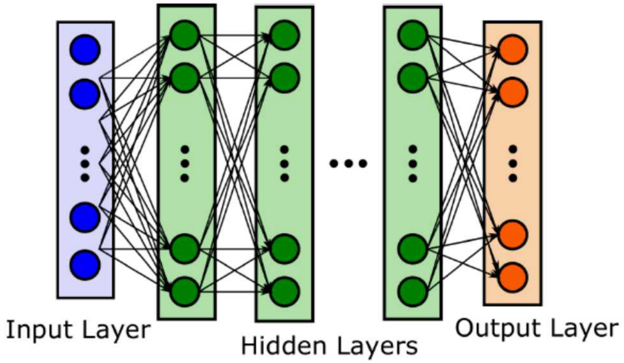
Research Question

Can ***data-driven models***, constructed using ***machine learning techniques***, provide a novel path forward for near-wall turbulence models with improved accuracy for surface loading predictions?

Decision Tree/Random Forest



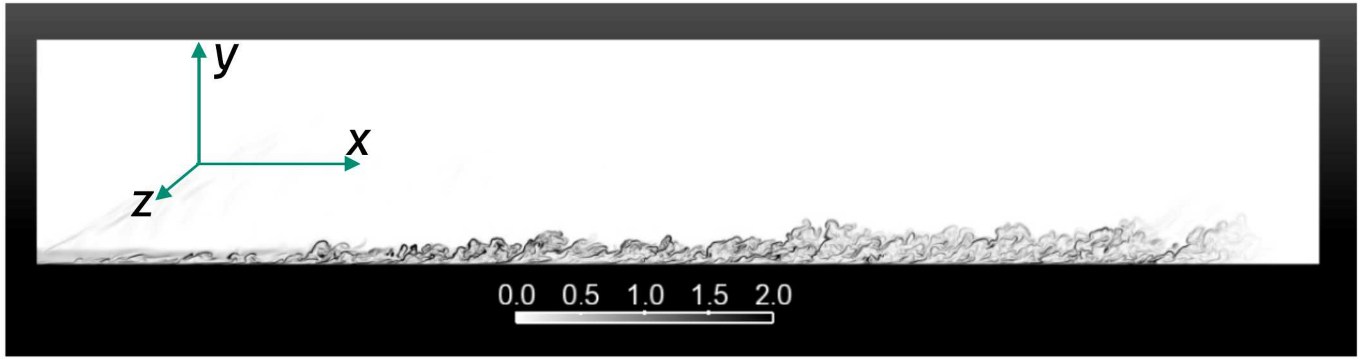
Neural Network, or Multi-Layer Perceptron (MLP)



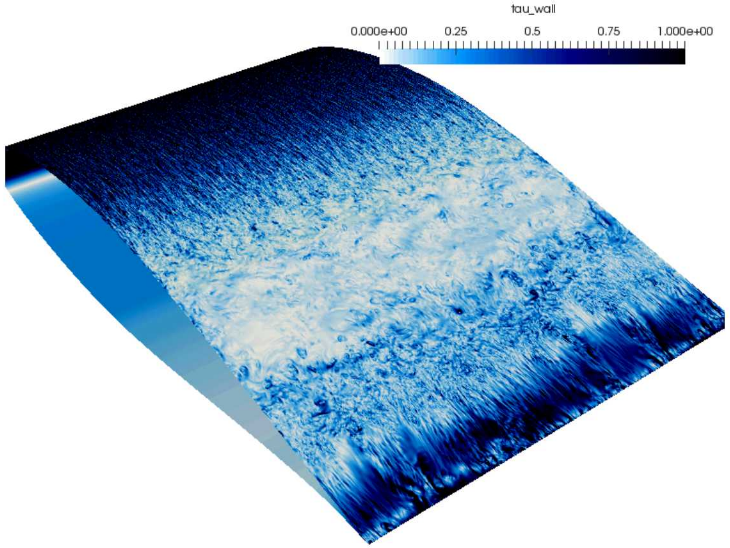
A Physics-Based Constraint

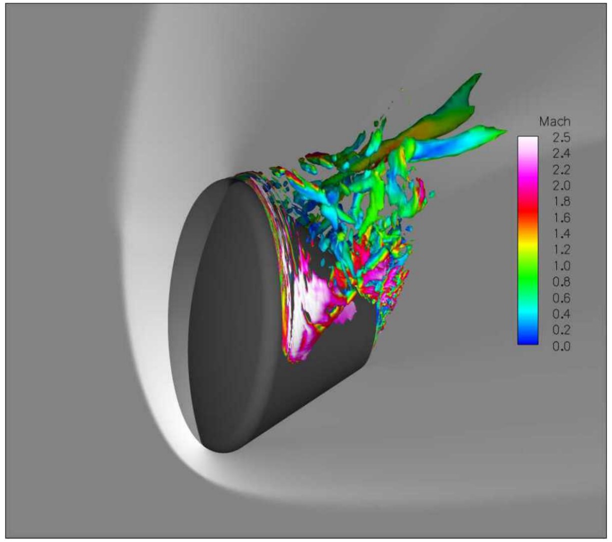
- Constraint: Our model must not depend on the coordinate system in which it is trained.
- This is called "coordinate frame invariance"

Training Data (DNS)



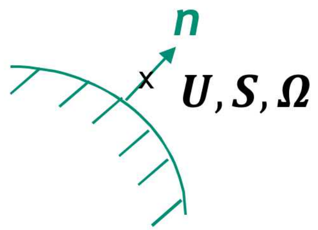
Potential Applications



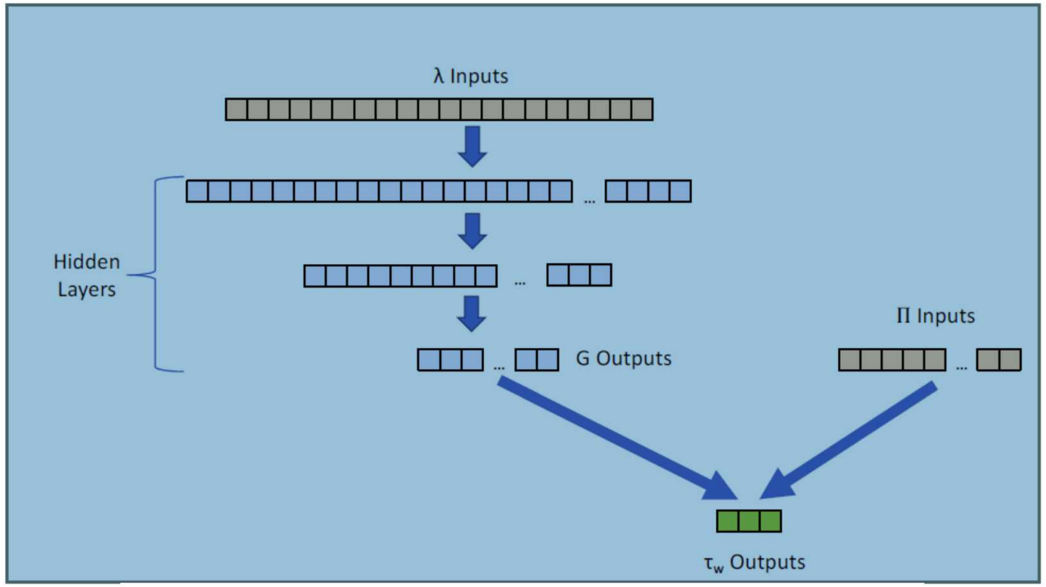


Constructing a Coordinate-Frame Invariant Model

Inspired by Ling, Kurzawski, and Templeton, "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance." *J. Fluid Mech.* 807:155-166, 2016.



- The wall shear stress model must be applicable for:
  - Arbitrary Cartesian coordinate system
  - Arbitrary orientation of the wall
- This fundamental property is ensured by using tensor invariant theory to identify:
  - The appropriate invariant features (inputs) to the model
  - A representation of the wall shear stress vector invariant to rotations about the wall-normal vector



Scalar invariant inputs

$$
\begin{aligned}
\lambda_1 &= \{\overline{\mathbf{S}}\}, & \lambda_2 &= \{\overline{\mathbf{S}}^2\}, \\
\lambda_3 &= \{\overline{\boldsymbol{\Omega}}^2\}, & \lambda_4 &= \{\overline{\mathbf{S}}^3\}, \\
\lambda_5 &= \{\overline{\mathbf{S}}\,\overline{\boldsymbol{\Omega}}^2\}, & \lambda_6 &= \overline{\mathbf{U}}\cdot\overline{\mathbf{U}}, \\
\lambda_7 &= \overline{\mathbf{U}}\cdot\overline{\mathbf{S}}\,\overline{\mathbf{U}}, & \lambda_8 &= \overline{\mathbf{U}}\cdot\boldsymbol{\epsilon}\overline{\boldsymbol{\Omega}}, \\
\lambda_9 &= \mathbf{n}\cdot\overline{\mathbf{S}}\mathbf{n}, & \lambda_{10} &= \mathbf{n}\cdot\overline{\mathbf{U}}, \\
\lambda_{11} &= \mathbf{n}\cdot\boldsymbol{\epsilon}\overline{\boldsymbol{\Omega}}, & \lambda_{12} &= \mathbf{n}\cdot\overline{\mathbf{S}}^2\mathbf{n}, \\
\lambda_{13} &= \mathbf{n}\cdot\boldsymbol{\epsilon}\overline{\mathbf{S}}\,\overline{\boldsymbol{\Omega}}, & \lambda_{14} &= \mathbf{n}\cdot(\overline{\mathbf{S}}\mathbf{n}\times\overline{\mathbf{S}}^2\mathbf{n}), \\
\lambda_{15} &= \mathbf{n}\cdot\boldsymbol{\epsilon}\overline{\mathbf{S}}\,\overline{\boldsymbol{\Omega}}^2, & \lambda_{16} &= \mathbf{n}\cdot\overline{\mathbf{S}}\,\overline{\boldsymbol{\Omega}}\mathbf{n}, \\
\lambda_{17} &= \mathbf{n}\cdot\overline{\mathbf{S}}\,\overline{\mathbf{U}}, & \lambda_{18} &= \mathbf{n}\cdot(\overline{\mathbf{U}}\times\overline{\mathbf{S}}\,\overline{\mathbf{U}}), \\
\lambda_{19} &= \mathbf{n}\cdot(\overline{\mathbf{U}}\times\overline{\mathbf{S}}\mathbf{n}), & \lambda_{20} &= \mathbf{n}\cdot\overline{\boldsymbol{\Omega}}\,\overline{\mathbf{U}},
\end{aligned}
$$

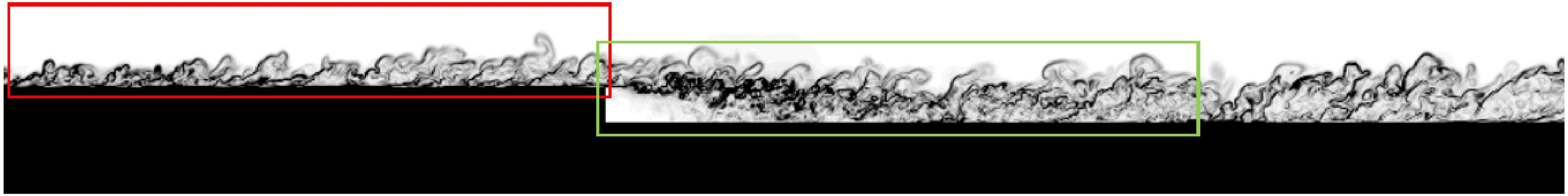
Form invariant inputs (vectors)

$$
\begin{aligned}
\Pi^{(1)} &= \overline{\mathbf{U}}, & \Pi^{(2)} &= \boldsymbol{\epsilon}\overline{\boldsymbol{\Omega}}, \\
\Pi^{(3)} &= \mathbf{n}, & \Pi^{(4)} &= \overline{\mathbf{S}}\mathbf{n}, \\
\Pi^{(5)} &= \mathbf{n}\times\overline{\mathbf{S}}\mathbf{n}, & \Pi^{(6)} &= \overline{\boldsymbol{\Omega}}\mathbf{n}, \\
\Pi^{(7)} &= \mathbf{n}\times\overline{\mathbf{U}}, & \Pi^{(8)} &= \mathbf{n}\times\overline{\boldsymbol{\Omega}}\mathbf{n}.
\end{aligned}
$$

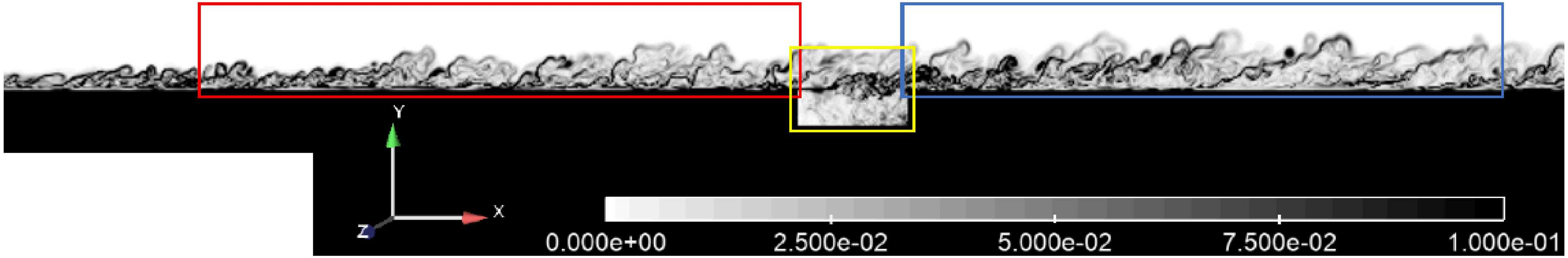
$$
\begin{bmatrix} \tau_{w,1} \\ \tau_{w,2} \\ \tau_{w,3} \end{bmatrix} = \begin{bmatrix} \Pi_1^{(1)} & \Pi_1^{(2)} & \Pi_1^{(3)} & \cdots & \Pi_1^{(8)} \\ \Pi_2^{(1)} & \Pi_2^{(2)} & \Pi_2^{(3)} & \cdots & \Pi_2^{(8)} \\ \Pi_3^{(1)} & \Pi_3^{(2)} & \Pi_3^{(3)} & \cdots & \Pi_3^{(8)} \end{bmatrix} \begin{bmatrix} G^{(1)} \\ G^{(2)} \\ G^{(3)} \\ \vdots \\ G^{(8)} \end{bmatrix}
$$

Direct Numerical Simulation Database

Backward Facing Step

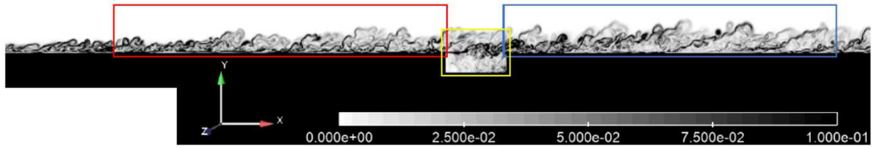


Cavity Flow

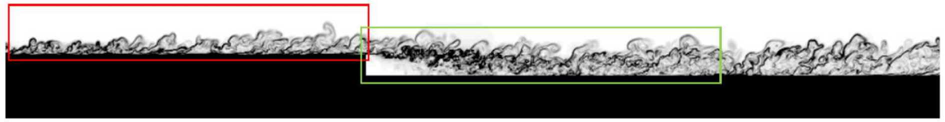


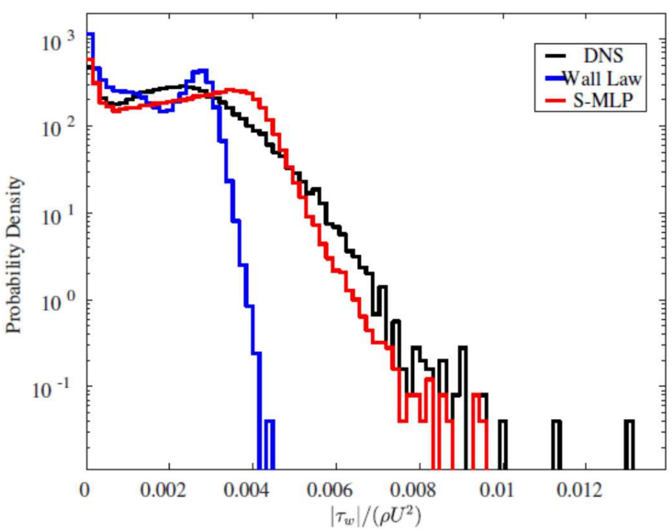
A Priori Tests of the ML Wall Shear Stress Model

Training Data

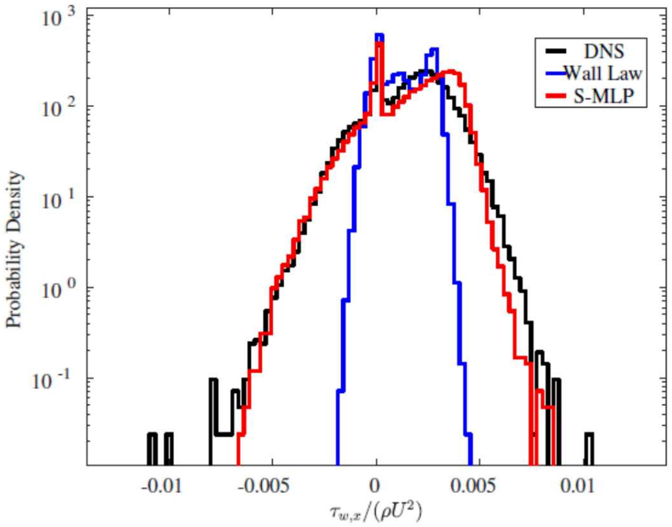


Test Case

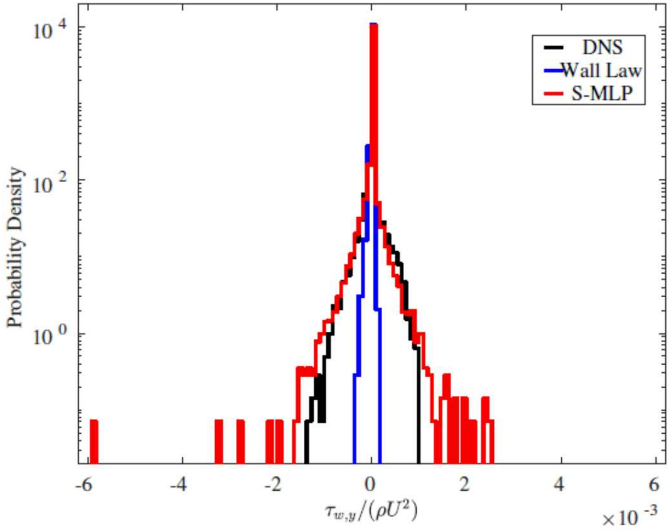




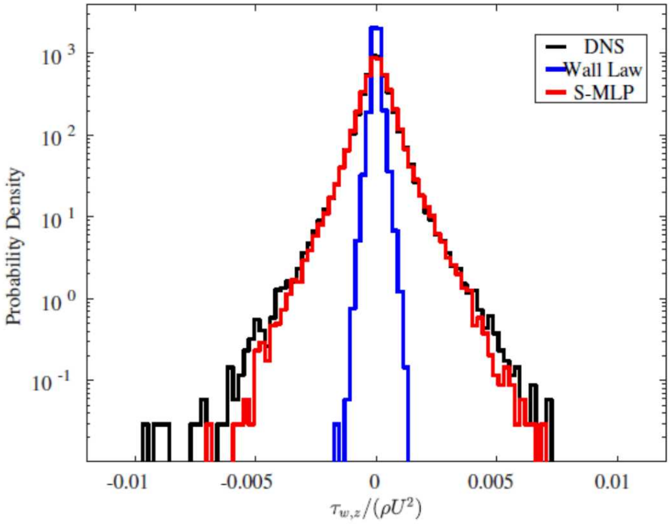
a) $|\tau_w|$



b) $\tau_{w,x}$



c) $\tau_{w,y}$

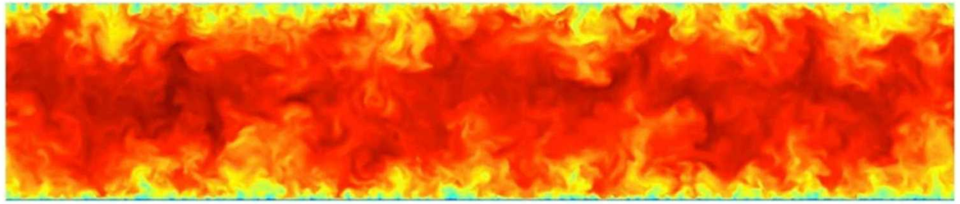


d) $\tau_{w,z}$

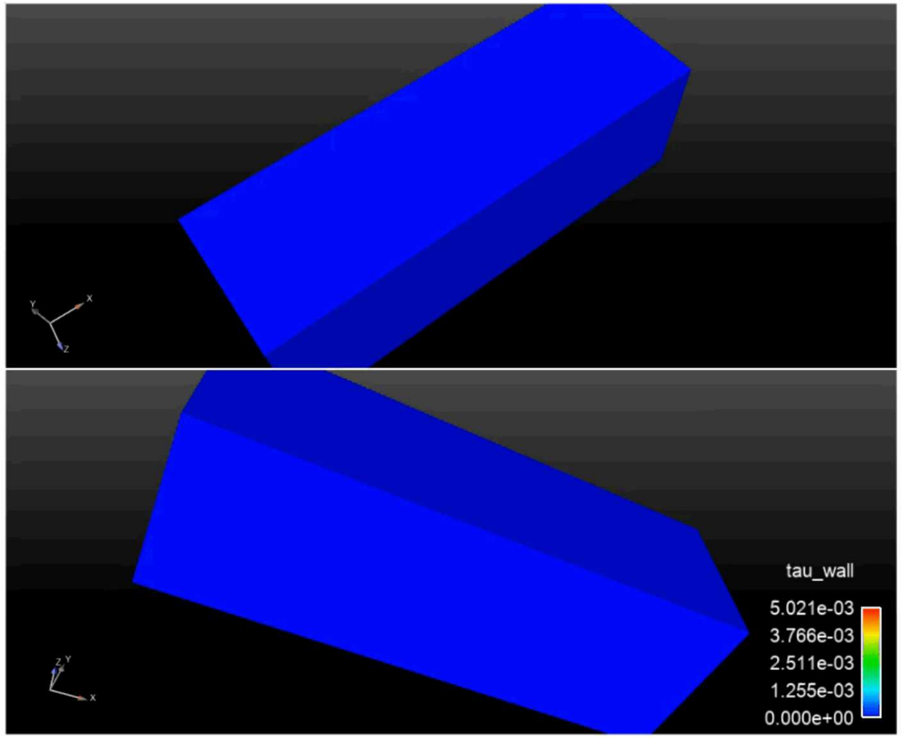
Towards V&V: Testing of ML Wall Shear Stress Model in a LES Code

- Numerical stability with ML near-wall model?
- Can we use an implicit time advancement without left-hand-side sensitivities of the near wall model?
- Does the wall model give improved wall shear stress statistics?
- Does the near wall model result in good mean flow predictions?

Training Data: JHU Turbulent Channel Flow Database

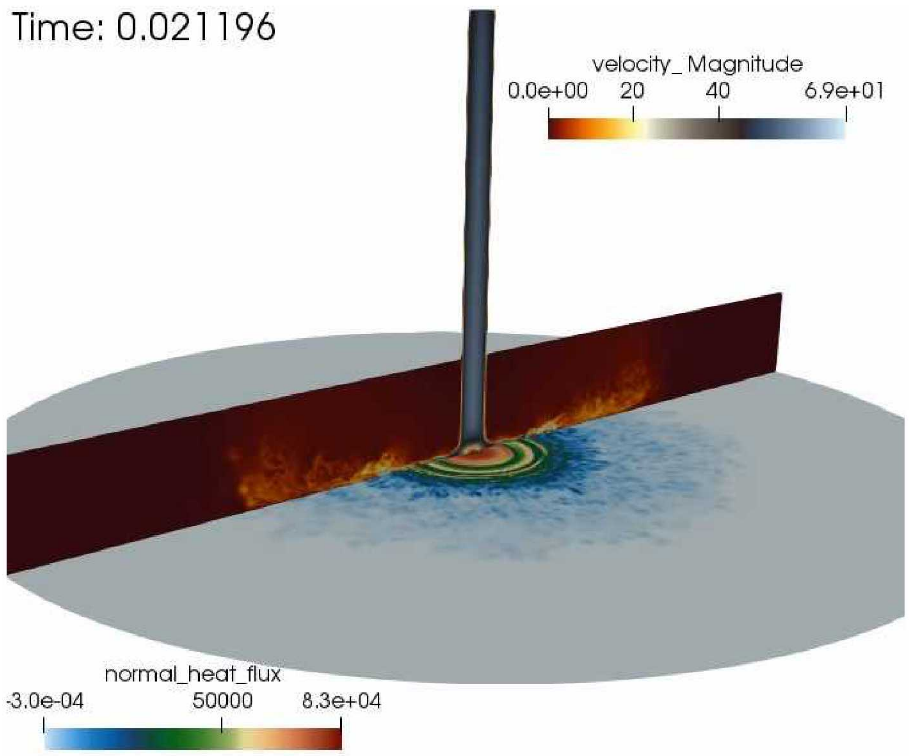


Nalu simulation of a turbulent channel flow at Re_tau = 2000. The native Nalu wall function boundary condition was applied on the top wall, with the machine-learned wall shear stress model applied on the bottom wall.



Further Research Directions

- Analyze numerical stability properties of neural-network-based near-wall turbulence models
- Incorporate proper physical scaling to ensure validity of the model for high Reynolds number flow
- Generate production-ready models using data from multiple training sets, test on complex flows
  - Leverage current ASC efforts surrounding an impinging jet flow
- Expand near-wall modeling physics to include heat transfer, develop ML model for wall heat flux
- Examine computational efficiency/accuracy tradeoffs associated with ML models within LES codes



ML Models for Turbulent Flow: Challenges

Challenge: Credibility & Incompleteness of Data-Driven Approaches

Bob MacCormack

Peter Lax

Lax Equivalence Theorem for the finite difference solution of linear PDE's.

Consistency + stability $\Longrightarrow$ convergence

Credibility and "Credibility"

How does one verify a machine-learned turbulence model?

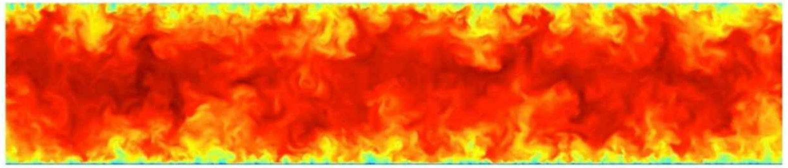
How does one select training data?

Challenge: Numerical Solutions to PDE's using Data-driven Models

Example: "Model conditioning" – the sensitivity of the solved quantities to the modeled terms.

- Studies have shown that small errors in Reynolds stresses can be amplified and result in large errors in predictions of mean velocities (Poroseva (2016), Thompson et al (2016).

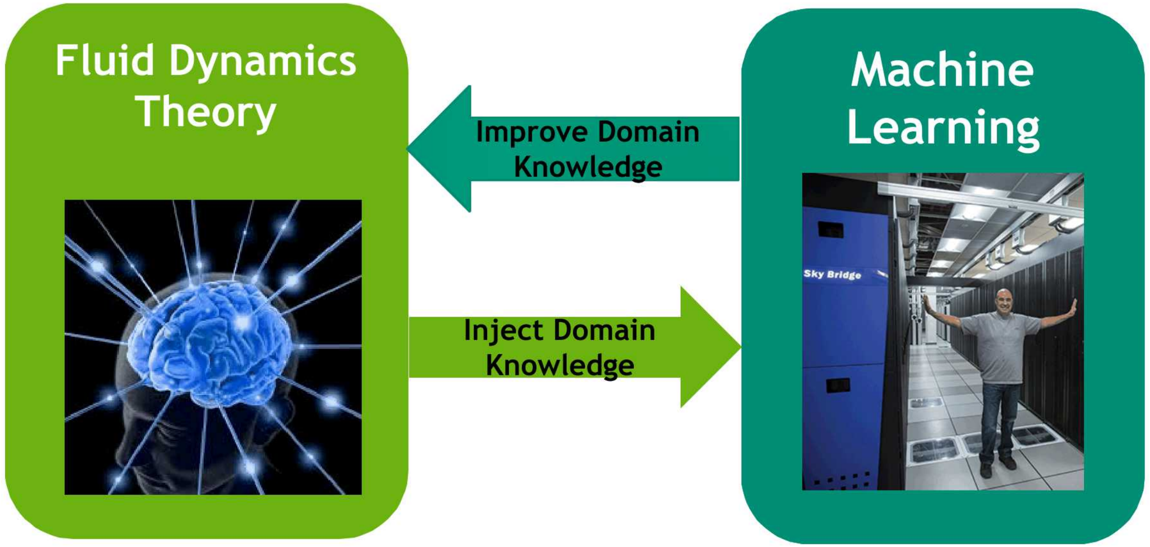
Frictional Reynolds number (Re_τ)	180	550	1000	2000	5200
Error in turbulent shear stresses					
volume averaged	0.17%	0.21%	0.03%	0.15%	0.31%
maximum	0.43%	0.38%	0.07%	0.23%	0.41%
Errors in mean velocities					
volume averaged	0.25%	1.61%	0.17%	2.85%	**21.6%**
maximum	0.36%	2.70%	0.25%	5.48%	**35.1%**



$$\frac{\partial U_i}{\partial t}+\frac{\partial}{\partial x_j}\left(U_j U_i\right)=-\frac{1}{\rho}\frac{\partial P}{\partial x_i}+\frac{\partial}{\partial x_j}\left(\nu\frac{\partial U_i}{\partial x_j}-\overline{u_i' u_j'}\right)$$

Use DNS data for the Reynolds stress

Challenge: Learning from Machine Learned Models



"…Regrettably, these [machine learning] studies have not led to insights into improving closure models."
P. Durbin, "Some recent developments in turbulence closure modeling", Annual Review of Fluid Mechanics, 2018.