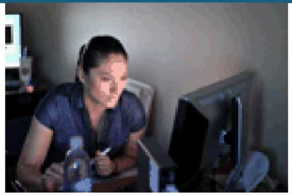
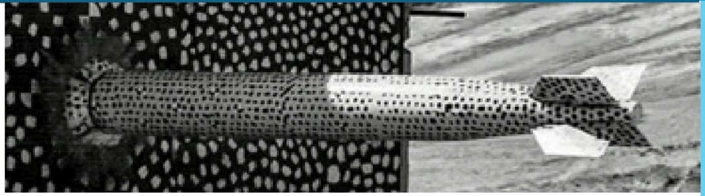
SAND2018-12914PE

Data-Driven Turbulence Modeling: A New Era in Turbulence Research?





New Research Ideas Forum
Sandia National Laboratories (NM)
6 November, 2018

Matthew Barone
Aerosciences Department

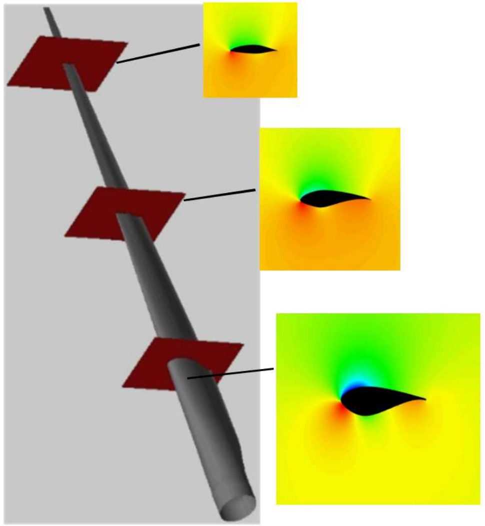
Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

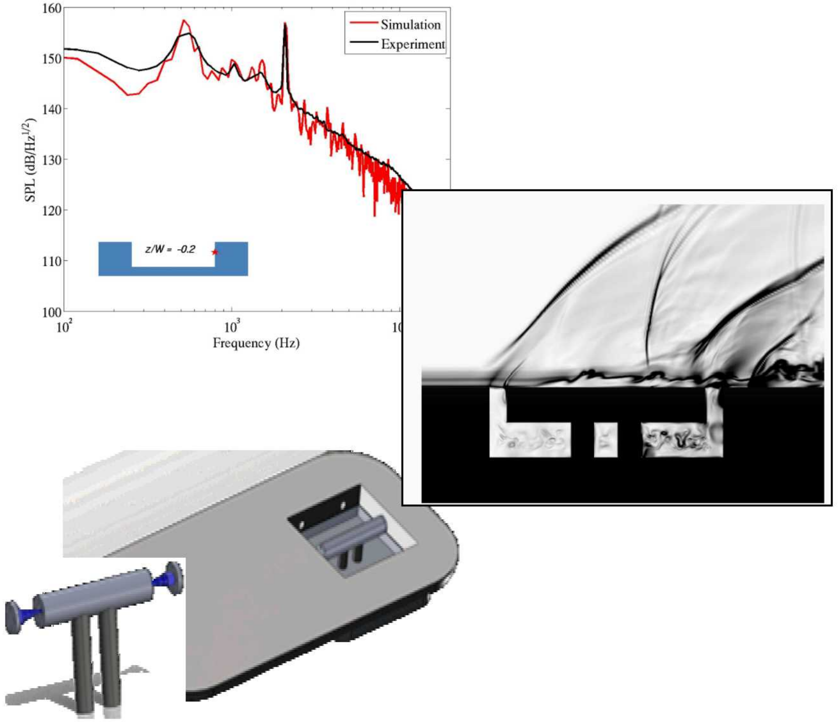
Collaborators, Colleagues, Managers:

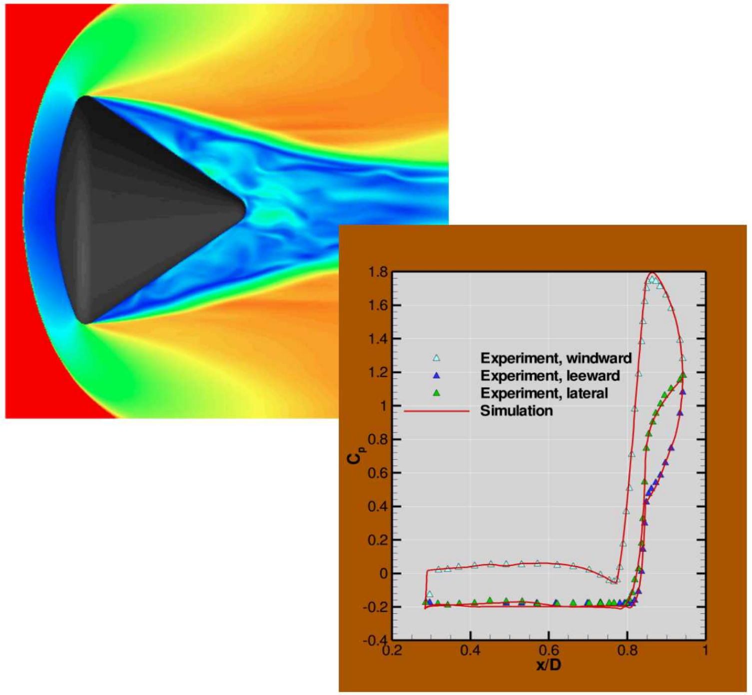
Warren Davis
Nathan Miller
Jeff Fike
Kenny Chowdhary
Julia Ling
Srini Arunajatesan
Stefan Domino
Bob Moser
Steve Beresh
Jaideep Ray
Larry DeChant
Reese Jones
Shawn Martin
Kevin Carlberg
Irina Tezaur
Basil Hassan

My Perspective

- Background in compressible fluid mechanics and computational fluid dynamics (CFD)
- Conducted research in model reduction for turbulent flows and data-driven turbulence modeling
- I am an engineer, I sometimes feel like a scientist, but alas, I am not a mathematician.
- This talk is biased towards aerospace applications.







Introduction

Backdrop:

- The intersection of computational modeling of physical systems with data science has become a very active research area.
- Progress in developing models for fluid turbulence has been slow for ~two decades.
- Naturally, turbulence researchers are turning to the new (old) field of data science.

Purpose of this talk:

To give commentary on the developing field of data-driven turbulence modeling, with the intent of motivating others, who may have something to contribute, to do so.

Disclaimer: *This talk is not in any way comprehensive or authoritative. It consists of the musings of one (part-time) researcher in the field.*

Turbulence

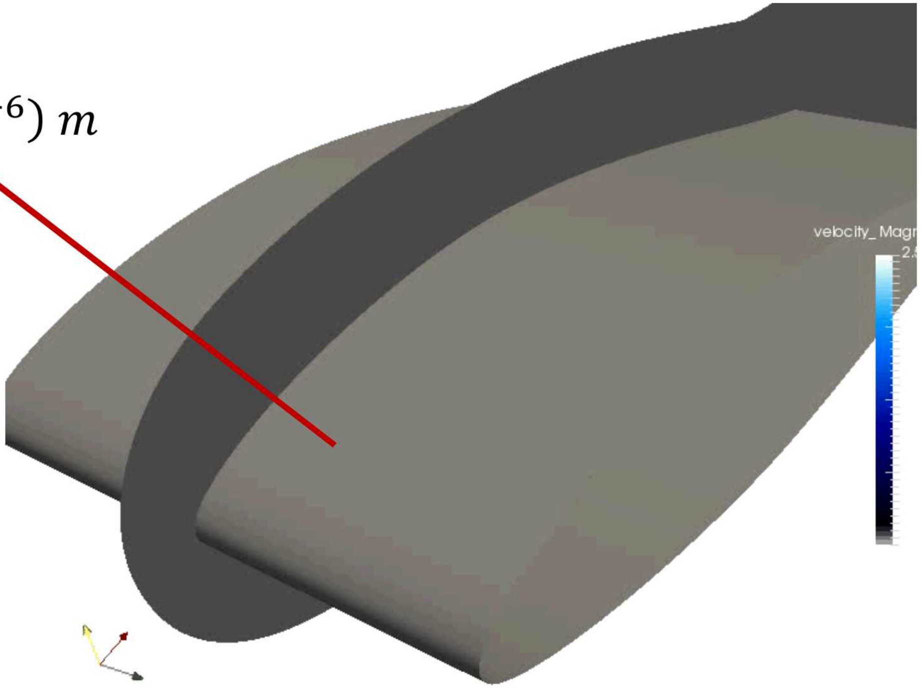
Navier-Stokes equation (Conservation of Momentum)

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho}\nabla \bar{p} + \nu \nabla^2 \mathbf{u}$$

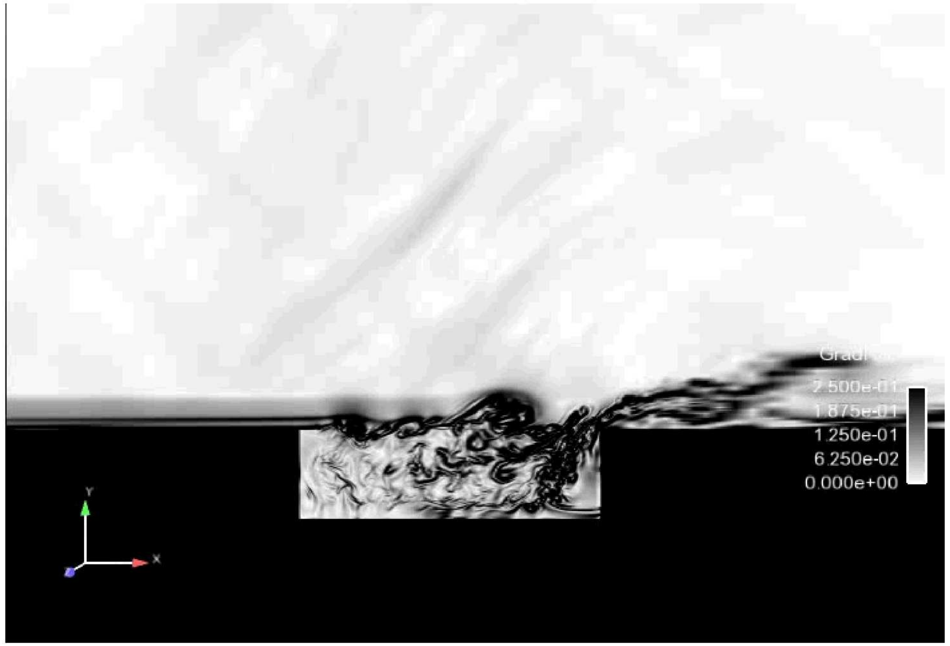
$L \sim O(10)\ m$

Photo from: SAND2014-17466O (UUR)

$l \sim O(10^{-6})\ m$



Trinity Open Science simulation, with Stefan Domino.



Reynolds Decomposition of the velocity field

$$u_i = U_i(x_k) + u_i'(x_k, t)$$

time average — fluctuation

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

RANS models provide a closure model for this stress term, in terms of things we know, such as $\frac{\partial U_i}{\partial x_j}$. How bad could that be?

Menter Shear Stress Transport RANS Model

2 PDE's, one derived from 1st principles, one "postulated" (made up)

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho u_j k)}{\partial x_j} = P - \beta^* \rho \omega k + \frac{\partial}{\partial x_j}\left[(\mu + \sigma_k \mu_t)\frac{\partial k}{\partial x_j}\right]$$

$$\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial(\rho u_j \omega)}{\partial x_j} = \frac{\gamma}{\nu_t} P - \beta \rho \omega^2 + \frac{\partial}{\partial x_j}\left[(\mu + \sigma_\omega \mu_t)\frac{\partial \omega}{\partial x_j}\right] + 2(1-F_1)\frac{\rho \sigma_{\omega 2}}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}$$

Magic blending of model constants:

$$\phi = F_1 \phi_1 + (1 - F_1)\phi_2$$

$$F_1 = tanh\left(arg_1^4\right)$$

$$arg_1 = min\left[max\left(\frac{\sqrt{k}}{\beta^* \omega d}, \frac{500\nu}{d^2 \omega}\right), \frac{4\rho\sigma_{\omega 2} k}{CD_{k\omega} d^2}\right]$$

$$CD_{k\omega} = max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-20}\right)$$

$$F_2 = tanh\left(arg_2^2\right)$$

$$arg_2 = max\left(2\frac{\sqrt{k}}{\beta^* \omega d}, \frac{500\nu}{d^2 \omega}\right)$$

$$P = \tau_{ij}\frac{\partial u_i}{\partial x_j}$$

Constitutive Equation for Turbulent Stress

$$-\overline{u_i' u_j'} \rightarrow \tau_{ij} = \mu_t\left(2S_{ij} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij}\right) - \frac{2}{3}\rho k \delta_{ij}$$

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

$$\mu_t = \frac{\rho a_1 k}{max(a_1 \omega, \Omega F_2)}$$

Constants:

$$\gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^*}} \qquad \gamma_2 = \frac{\beta_2}{\beta^*} - \frac{\sigma_{\omega 2}\kappa^2}{\sqrt{\beta^*}}$$

$\sigma_{k1} = 0.85$ $\sigma_{\omega 1} = 0.5$ $\beta_1 = 0.075$

$\sigma_{k2} = 1.0$ $\sigma_{\omega 2} = 0.856$ $\beta_2 = 0.0828$

$\beta^* = 0.09$ $\kappa = 0.41$ $a_1 = 0.31$

RANS Models?

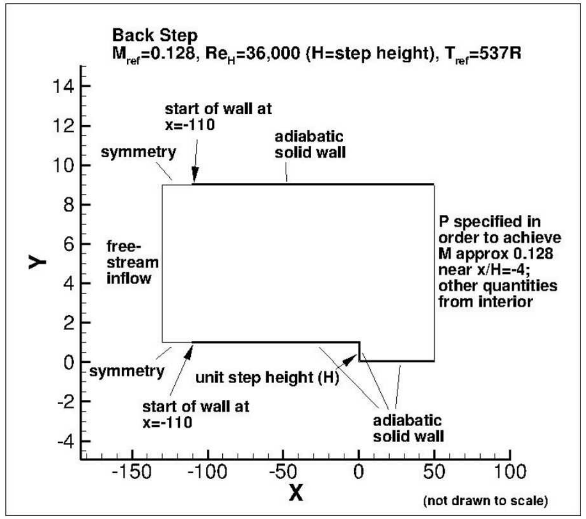
Image from www.digitaltrends.com

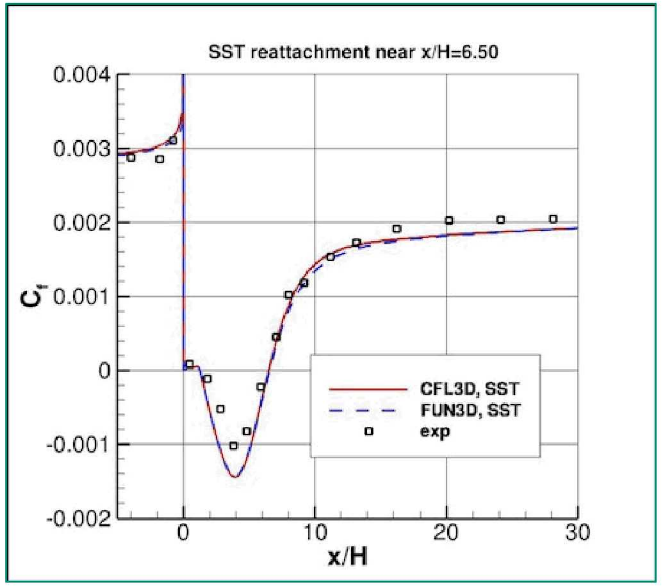
RANS Models: Commentary

- When I hired into Sandia in 2003, the Menter SST model on the previous slide was one of the best choices for use in external aerodynamics applications.
- This model is, in concept if not in execution, very similar to the first two equation RANS models developed in the late 1960's/early 1970's (I was born in 1974, the year of publication of the Launder-Sharma k-epsilon model).
- Today, in 2018, if I were asked for a RANS analysis of an external aerodynamic problem, I would probably use the Menter SST model.

RANS Model Predictive Accuracy

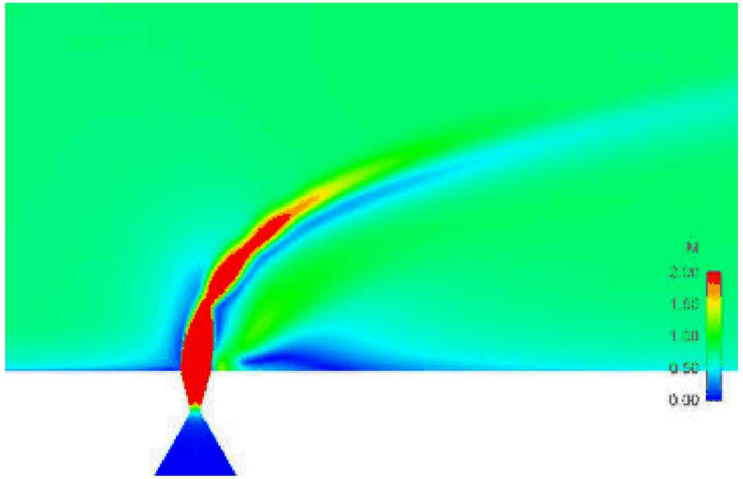
"Good"

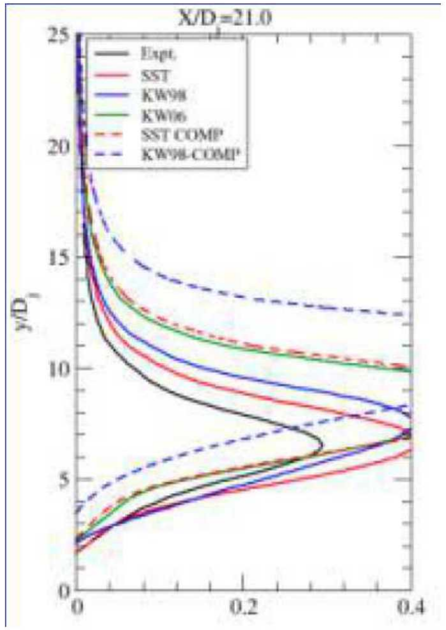




https://turbmodels.larc.nasa.gov/backstep_val.html

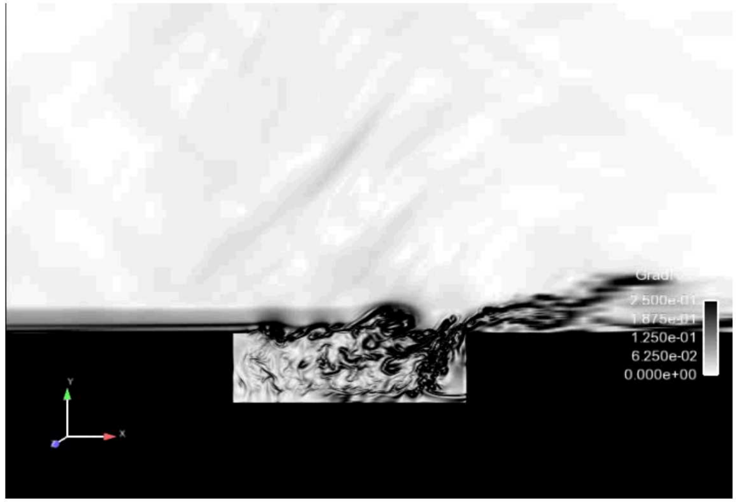
"Bad"

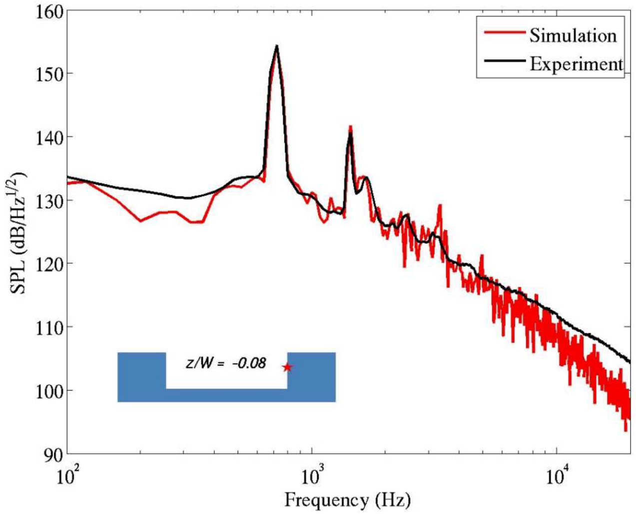




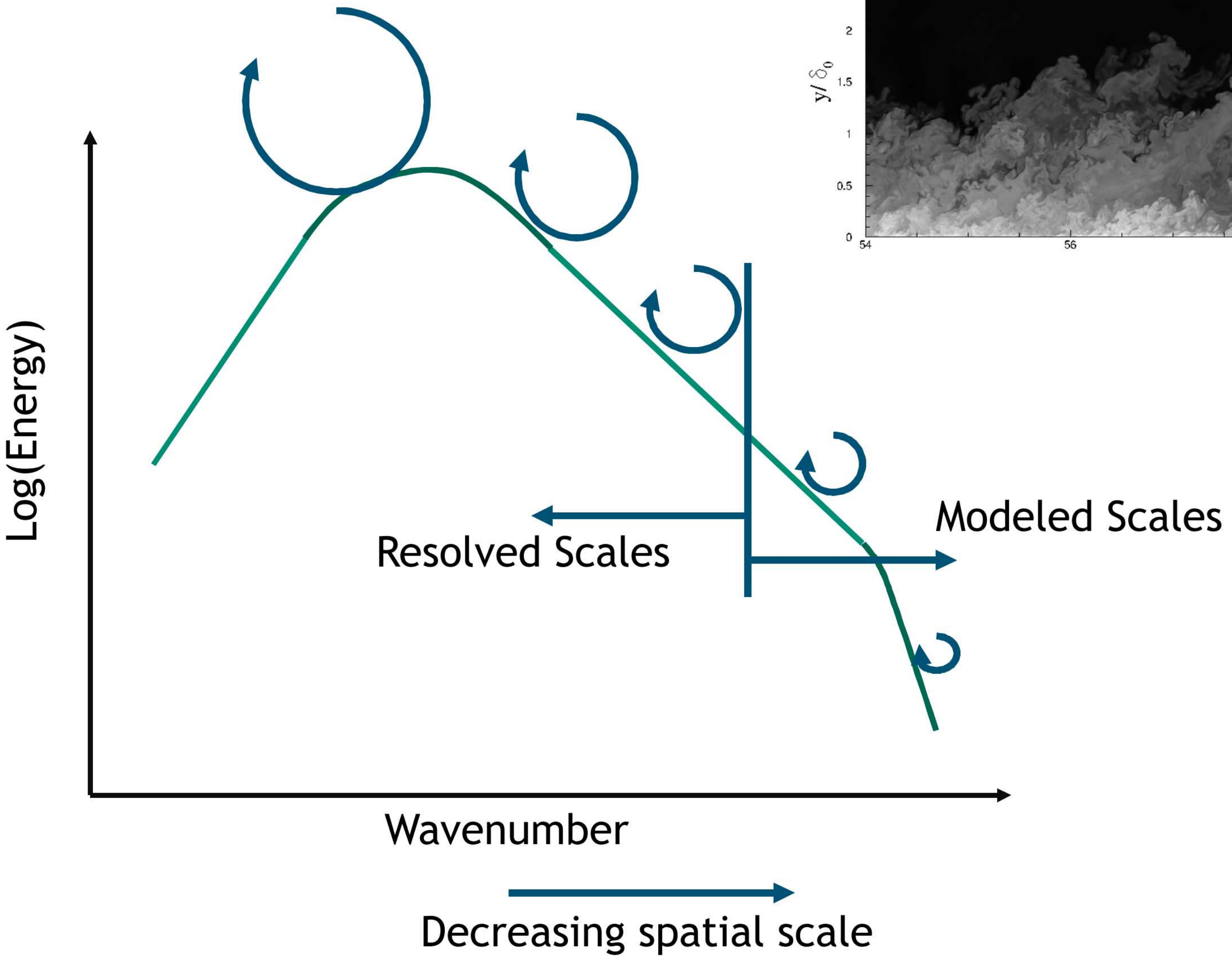
Arunajatesan, AIAA 2012-1199, 2012.

N/A

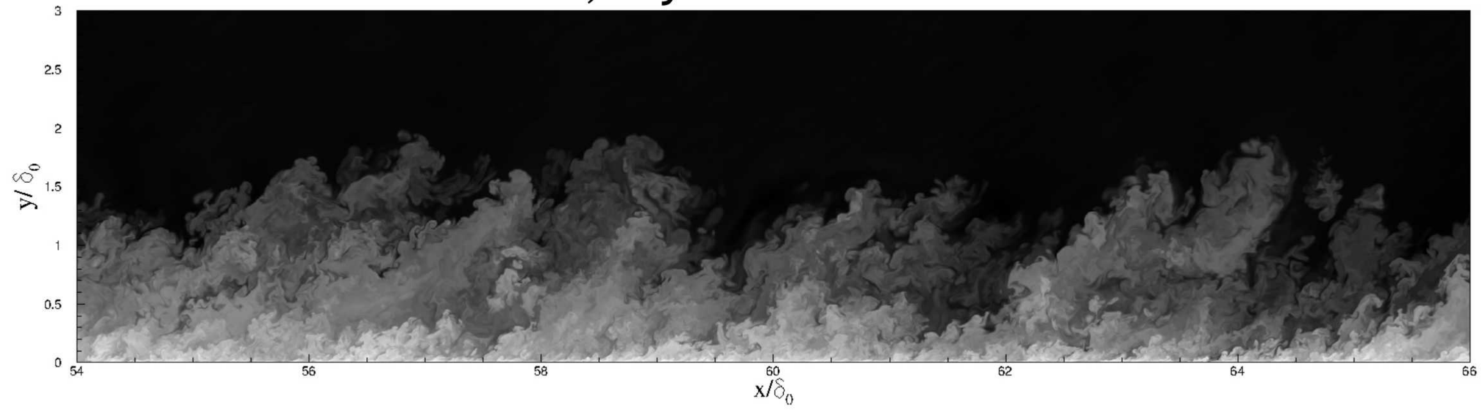




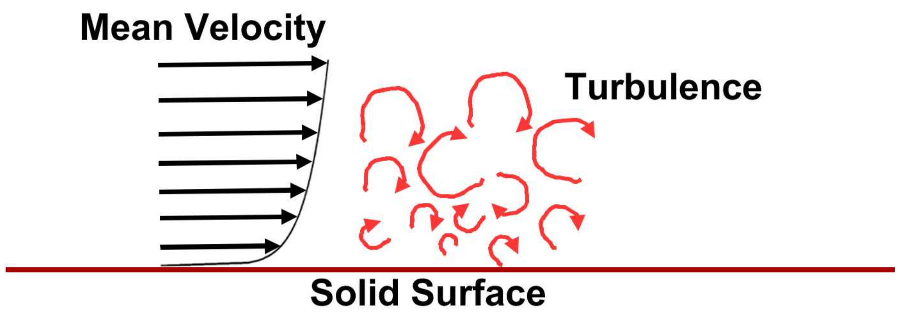
Large Eddy Simulation

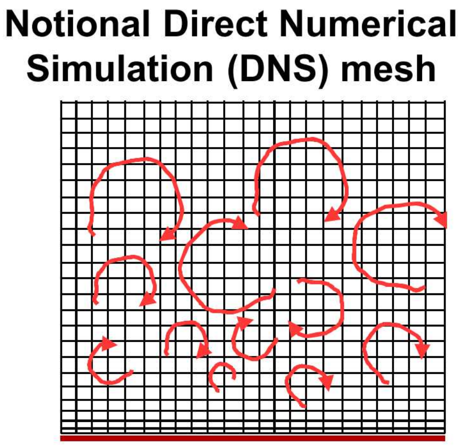


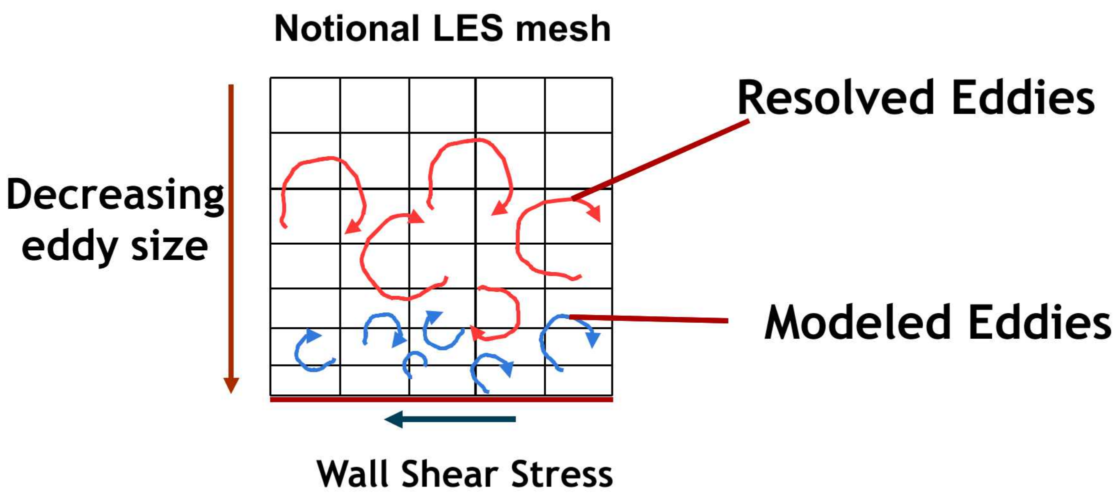
Source: Pirozzoli, Reynolds.dma.uniroma1.it/dnsm2/



Near-wall Turbulence Modeling in Large Eddy Simulation



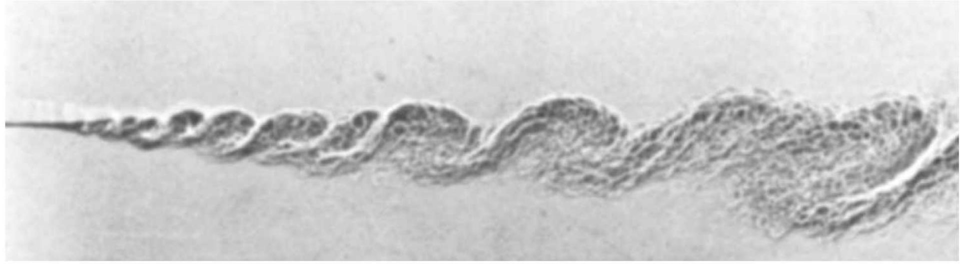




Wall-modeled LES offers computational savings decrease of *at least* several orders of magnitude over DNS for engineering systems of interest.

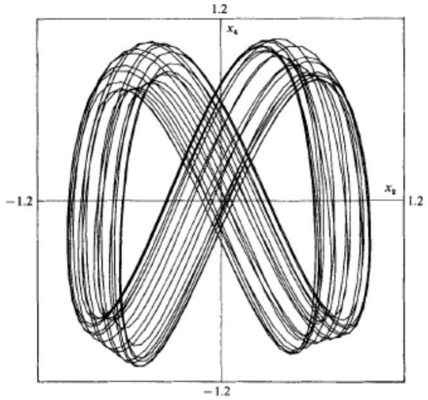
Topics in Turbulence Research over the Years

Coherent Structures



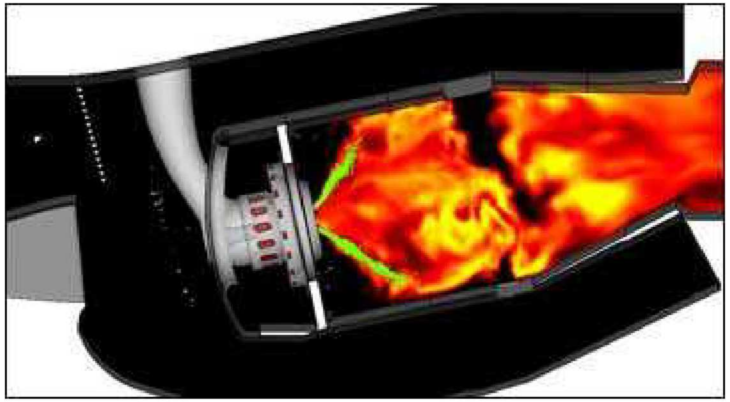
Brown and Roshko, *JFM* (1974).

Nonlinear Dynamical Systems



Aubry et al, *JFM* (1988).

Large Eddy Simulation



Ham etal, CTR (2003)

Enter Machine Learning: Signs of a New Research Frenzy

Excerpt from a recent call for white papers (2018).

7. Develop new turbulence models (or improve modeling capabilities of existing models) for RANS and LES for fully-developed turbulent boundary layers for Mach 6 and above. The models should be conducive to application within typical RANS/LES solution algorithms. These improvements should be applied to geometries such as circular cones, elliptical cones, BOLT, double wedges, or double cones. One possibility would be to use machine learning (or other similar approaches) to train hypersonic turbulence models using DNS (or perhaps using experimental data).

A Recent Call for Concepts

of innovative basic research concepts exploring radically new architectures and approaches in Artificial Intelligence (AI) that incorporate prior knowledge, such as known physical laws, to augment sparse data and to ensure robust operation. The Physics of AI (PAI) basic research Disruption Opportunity aims to develop novel AI architectures, algorithms and approaches that "bake in" the physics, mathematics and prior knowledge relevant to an application domain in order to address the technical challenges in application of AI in scientific discovery, human-AI collaboration, and a variety of defense applications.

Reactions from the Turbulence/Fluid Mechanics Community

The leaders in the field of turbulence modeling and simulation: a mixture of skepticism, enthusiasm, bewilderment, confusion, and "getting on board."

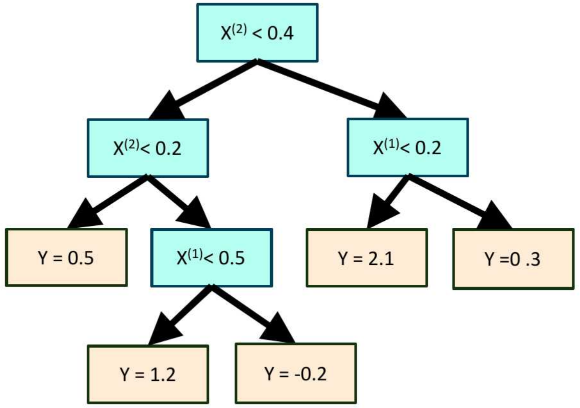
P. Spalart, NASA Ames Advanced Modeling & Simulation Seminar, July 18, 2018:

"The prospects for this new breed of research in RANS are still unclear"
"Most studies still have corrections of the type beta(x,y,z) in each flow…not a model"

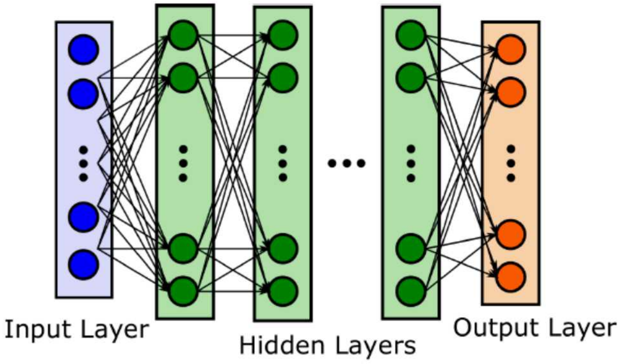
Machine Learning in Turbulence

- Supervised vs. Unsupervised Learning: examples of both, but most work is in supervised learning.
- Classification vs Regression:
  - Classification example : is my turbulence model valid at this location, or not?
  - Regression example : provide a prediction of a turbulent stress, given a local flow state vector
- Machine Learning algorithms: neural networks, random forests, Gaussian processes, …
- Training data: Direct Numerical Simulation, “well-resolved” Large Eddy Simulation, experimental data

Decision Tree/Random Forest



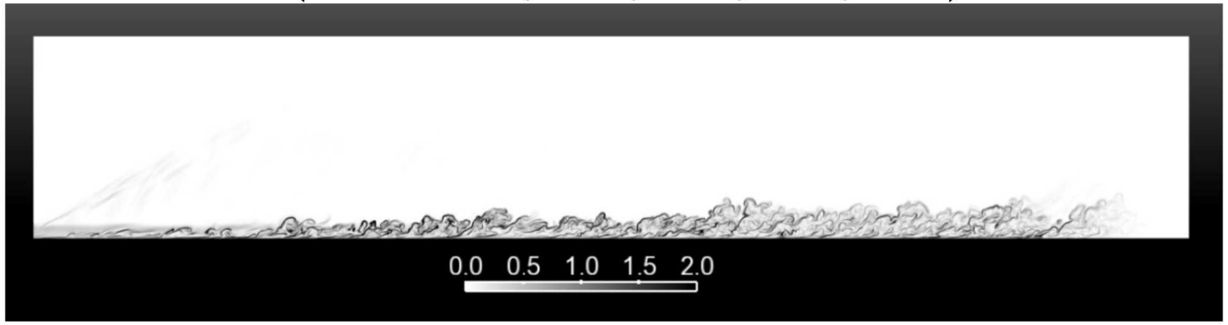
Neural Network, or Multi-Layer Perceptron (MLP)



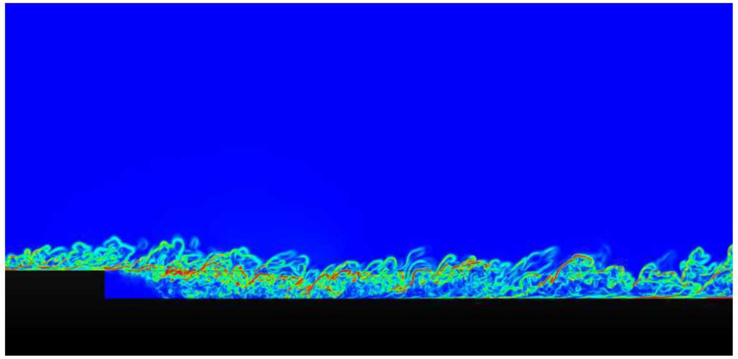
Sources of Training Data

- Direct Numerical Simulation (DNS)
- “Well-resolved” Large Eddy Simulation (LES)
- Experimental Data (for example, PIV)

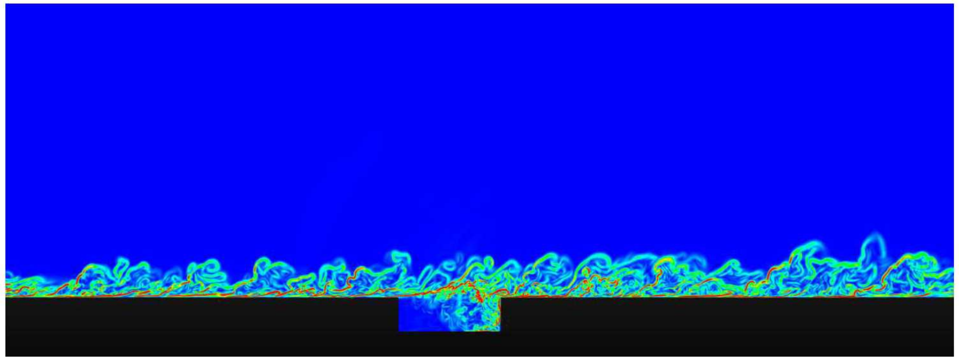
Compressible Boundary Layer (Mach=0.6, 2.0, 2.5, 3.0, 3.5)



Mach = 0.6 Backward-Facing Step



Mach = 0.6 Compressible Cavity Flow



DNS studies performed by J. Fike

Example: Tensor-basis Neural Networks for RANS
Ling, Kurzawski, & Templeton, J. Fluid Mech. 807:155-166, 2016.

Term requiring closure modeling:
Normalized Reynolds stress anisotropy tensor

$$b_{ij} = \overline{\boldsymbol{u}_i'\boldsymbol{u}_j'}/2k - 1/3\delta_{ij}$$

Linear eddy viscosity model (Boussinesq)

$$b_{ij} = -\frac{\nu_t S_{ij}}{k}$$

Mean Strain Rate (symmetric)

$$S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)$$

Mean Rotation Rate (anti-symmetric)

$$R_{ij} = \left(\frac{\partial U_i}{\partial x_j} - \frac{\partial U_j}{\partial x_i}\right)$$

Pope (1975) developed a general eddy viscosity model based on tensor invariance analysis

$$\boldsymbol{b} = \sum_{n=1}^{10} g^{(n)}(\lambda_1, \ldots, \lambda_5)\boldsymbol{T}^{(n)}$$

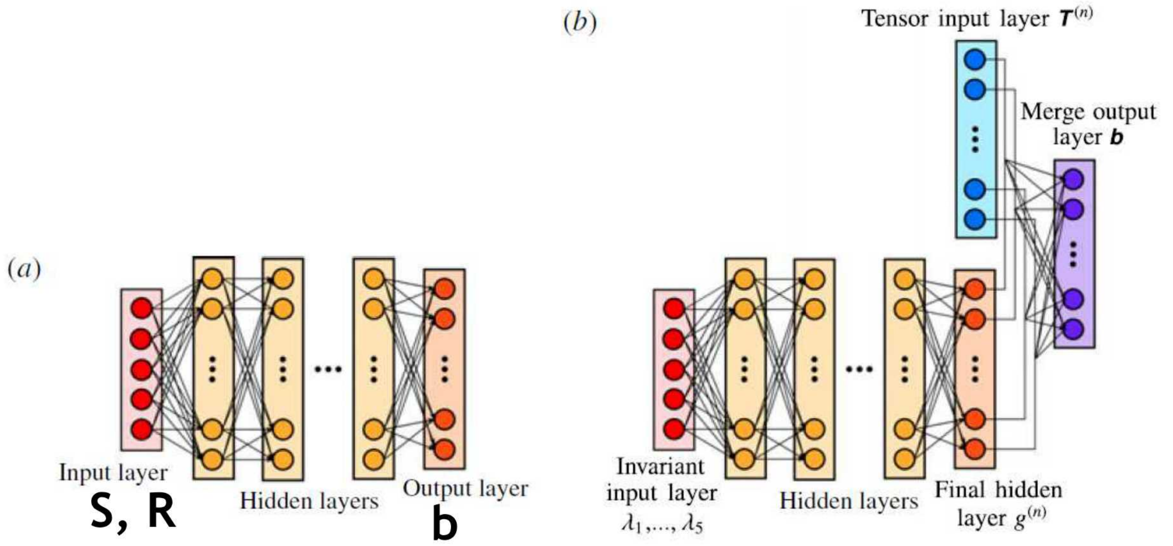
$$\left.\begin{array}{ll}
\boldsymbol{T}^{(1)} = \boldsymbol{S} & \boldsymbol{T}^{(6)} = \boldsymbol{R}^2\boldsymbol{S} + \boldsymbol{S}\boldsymbol{R}^2 - \frac{2}{3}\boldsymbol{I}\cdot Tr(\boldsymbol{S}\boldsymbol{R}^2) \\
\boldsymbol{T}^{(2)} = \boldsymbol{S}\boldsymbol{R} - \boldsymbol{R}\boldsymbol{S} & \boldsymbol{T}^{(7)} = \boldsymbol{R}\boldsymbol{S}\boldsymbol{R}^2 - \boldsymbol{R}^2\boldsymbol{S}\boldsymbol{R} \\
\boldsymbol{T}^{(3)} = \boldsymbol{S}^2 - \frac{1}{3}\boldsymbol{I}\cdot Tr(\boldsymbol{S}^2) & \boldsymbol{T}^{(8)} = \boldsymbol{S}\boldsymbol{R}\boldsymbol{S}^2 - \boldsymbol{S}^2\boldsymbol{R}\boldsymbol{S} \\
\boldsymbol{T}^{(4)} = \boldsymbol{R}^2 - \frac{1}{3}\boldsymbol{I}\cdot Tr(\boldsymbol{R}^2) & \boldsymbol{T}^{(9)} = \boldsymbol{R}^2\boldsymbol{S}^2 + \boldsymbol{S}^2\boldsymbol{R}^2 - \frac{2}{3}\boldsymbol{I}\cdot Tr(\boldsymbol{S}^2\boldsymbol{R}^2) \\
\boldsymbol{T}^{(5)} = \boldsymbol{R}\boldsymbol{S}^2 - \boldsymbol{S}^2\boldsymbol{R} & \boldsymbol{T}^{(10)} = \boldsymbol{R}\boldsymbol{S}^2\boldsymbol{R}^2 - \boldsymbol{R}^2\boldsymbol{S}^2\boldsymbol{R}
\end{array}\right\}$$

$$\lambda_1 = Tr(\boldsymbol{S}^2), \quad \lambda_2 = Tr(\boldsymbol{R}^2), \quad \lambda_3 = Tr(\boldsymbol{S}^3), \quad \lambda_4 = Tr(\boldsymbol{R}^2\boldsymbol{S}), \quad \lambda_5 = Tr(\boldsymbol{R}^2\boldsymbol{S}^2)$$

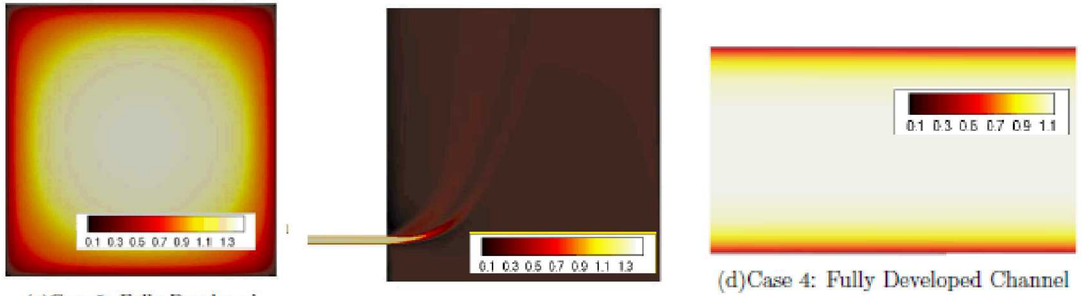
Example: Tensor-basis Neural Networks for RANS
Ling, Kurzawski, & Templeton, J. Fluid Mech. 807:155-166, 2016.

Inputs: strain rate and rotation rate tensors: **S, R**
Outputs: anisotropy tensor **b**

$$\boldsymbol{b} = \sum_{n=1}^{10} g^{(n)}(\lambda_1, \ldots, \lambda_5)\boldsymbol{T}^{(n)}$$



Training: DNS and well-resolved LES of various canonical flowfields (3 of 6 shown).



(e)Case 5: Fully Developed Square Duct Flow

(f)Case 6: Perpendicular Jet in Crossflow

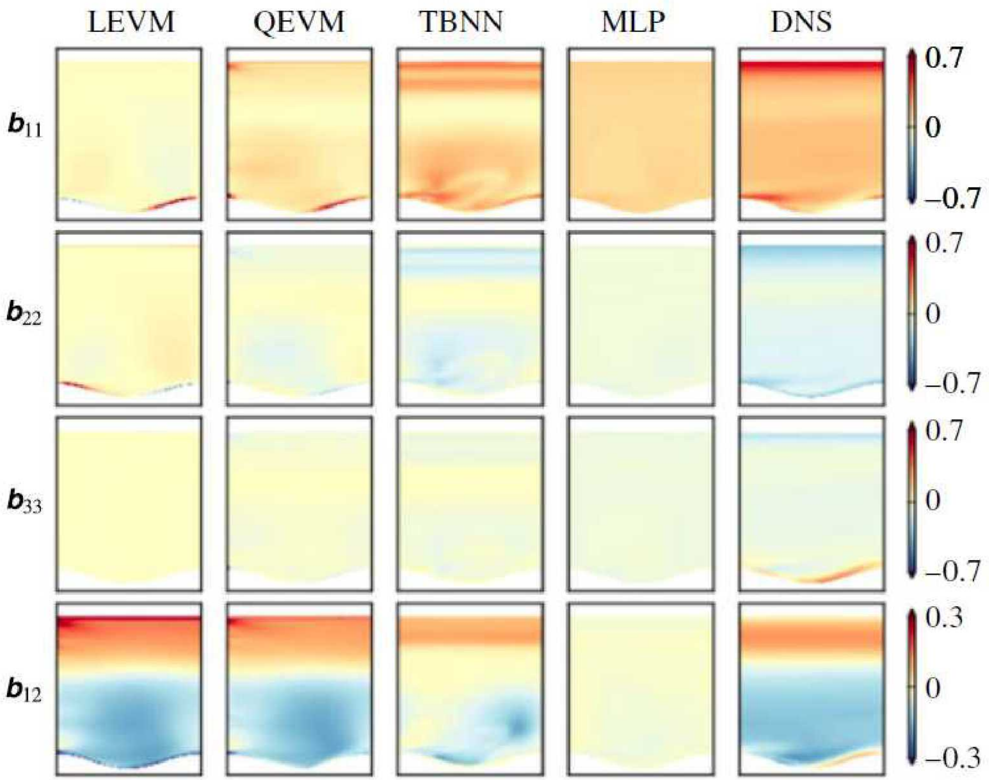
(d)Case 4: Fully Developed Channel Flow

Physical/Mathematical Constraints

- Generalized eddy viscosity form for the Reynolds stress anisotropy tensor **b**
- Rotational Invariance : the model does not depend on the orientation of the coordinate system.

Example: Tensor-basis Neural Networks for RANS Ling, Kurzawski, & Templeton, J. Fluid Mech. 807:155-166, 2016.

A priori Test: Modeled Stress Components for a wavy wall flow.



Test Cases: Predicted Velocity Fields

1. Square Duct

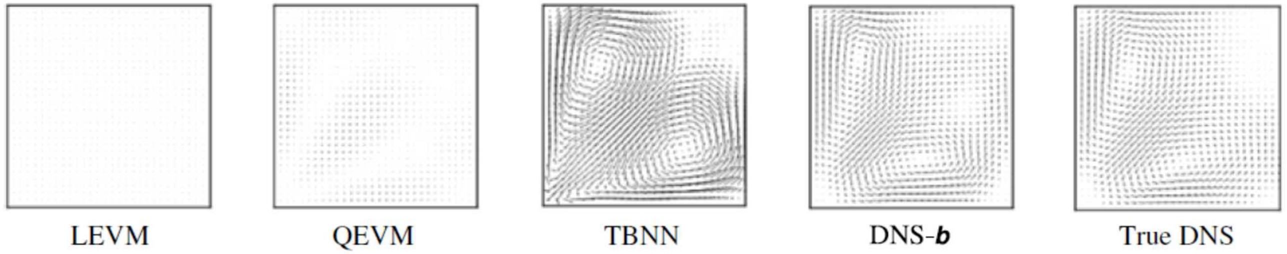


FIGURE 4. Plot of secondary flows in duct flow case. Reference arrows of length $U_b/10$ shown at the top of each plot.

2. Wavy Wall

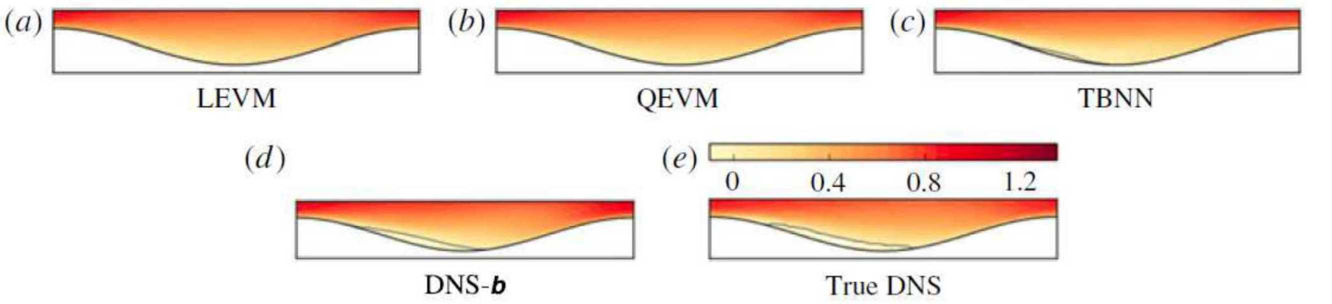
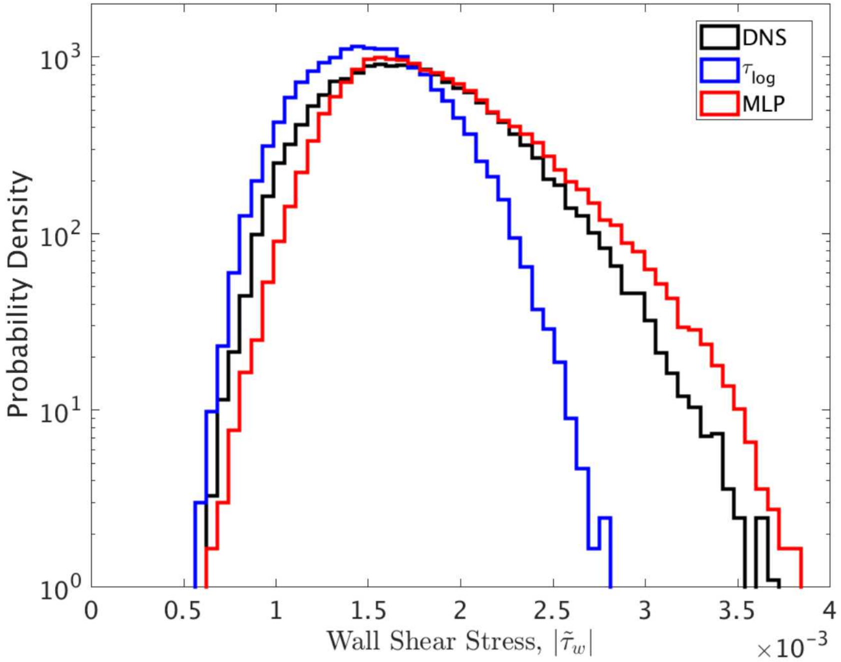


FIGURE 5. Contours of streamwise velocity normalized by bulk velocity in the wavy wall test case, zoomed into the near-wall region. Separated regions outlined in grey.

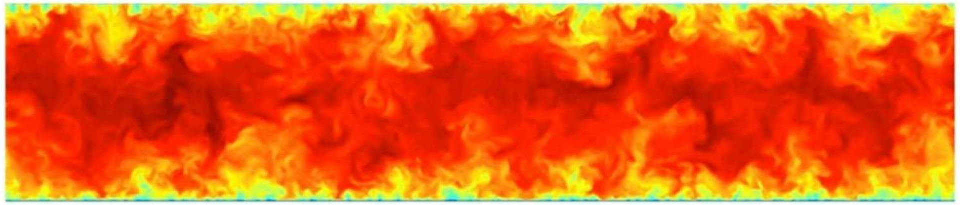
Example: Wall Shear Stress Models for LES using Neural Networks
(with Warren Davis, Nathan Miller, Jeffrey Fike, Kenny Chowdhary)

Highlighted Result: Instantaneous Wall Shear Stress Model for Turbulent Channel Flow

- Development of a machine-learned near wall turbulence model for channel flow that was predictive across Reynolds number parameter (No. features = 38).
- Utilizing near-wall scaling of variables, this model was trained for Reynolds No = 1000 and successfully tested on a different data set for Reynolds No. = 5000.



Turbulent Channel Flow DNS Database



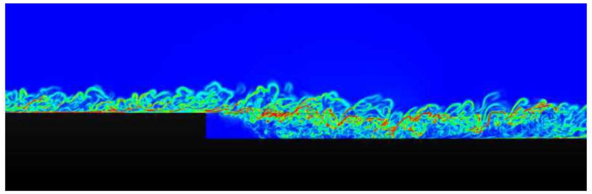
Measures of Agreement: Traditional Near Wall Model vs. Neural Network

	Mean-square Error	KL Divergence (ideal = 0)	Correlation Coefficient (ideal = 1)	Mean shear stress error	RMS shear stress error
Neural Network Model	1.18e-7	0.039	0.73	5.3%	5.2%
Log Law Model	1.51e-7	0.36	0.69	-12.7%	-13.3%

Example: Wall Shear Stress Models for LES using Neural Networks

Highlighted Result: Generalized Near Wall Model and Application to Backward Facing Step

- Created a method for generating a near-wall turbulence model that obeys the required coordinate frame invariance property by construction.



Frame-invariant approach for separated flow

- Shear stress as a function of near-wall flow state.

$$\tau_i(\mathbf{x}|_{|n|=0}) = f(\mathbf{U}(\mathbf{x}+\vec{n}), \mathbf{S}(\mathbf{x}+\vec{n}), \mathbf{\Omega}(\mathbf{x}+\vec{n})),$$

- Vector polynomial of form-invariants (qty. 8).

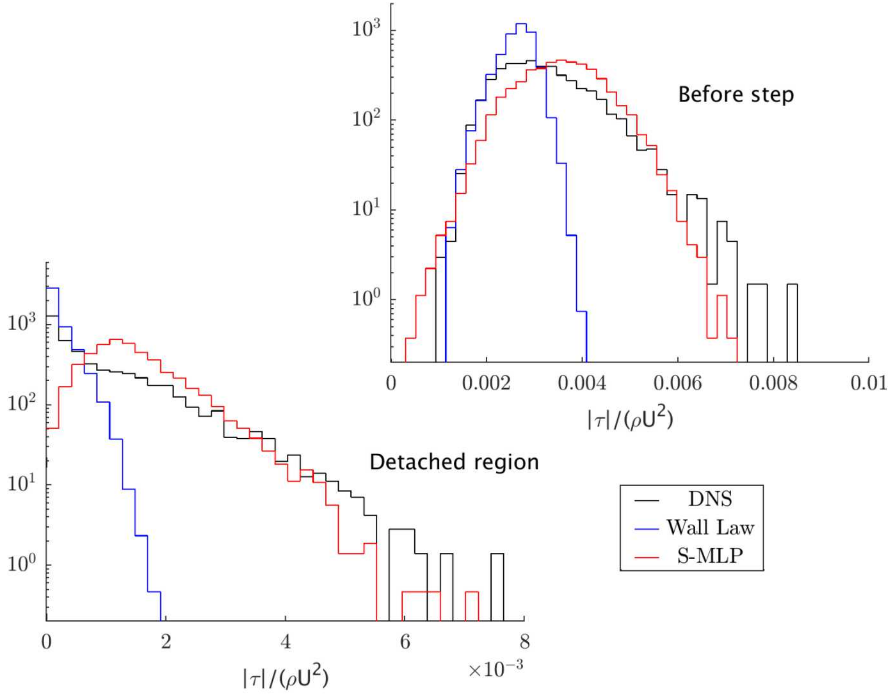
$$\tau_i = \sum_{n=1}^{8} G^{(n)} \Pi_i^{(n)}$$

- Scalar coefficients defined by unknown functions of scalar-invariants based on transverse-isotropy (qty. 20).

$$G^{(n)} = \mathcal{F}(\lambda_1, \lambda_2, \lambda_3, ..., \lambda_{20})$$

$$\lambda_1 = \{\breve{\mathbf{S}}\}, \quad \lambda_3 = \{\breve{\mathbf{\Omega}}^2\}, \quad \lambda_5 = \{\breve{\mathbf{S}}\breve{\mathbf{\Omega}}^2\}, \quad \lambda_{10} = \mathbf{n}\cdot\mathbf{U}, \quad \lambda_{18} = \mathbf{n}\cdot(\mathbf{U}\times\breve{\mathbf{S}}\mathbf{U})$$

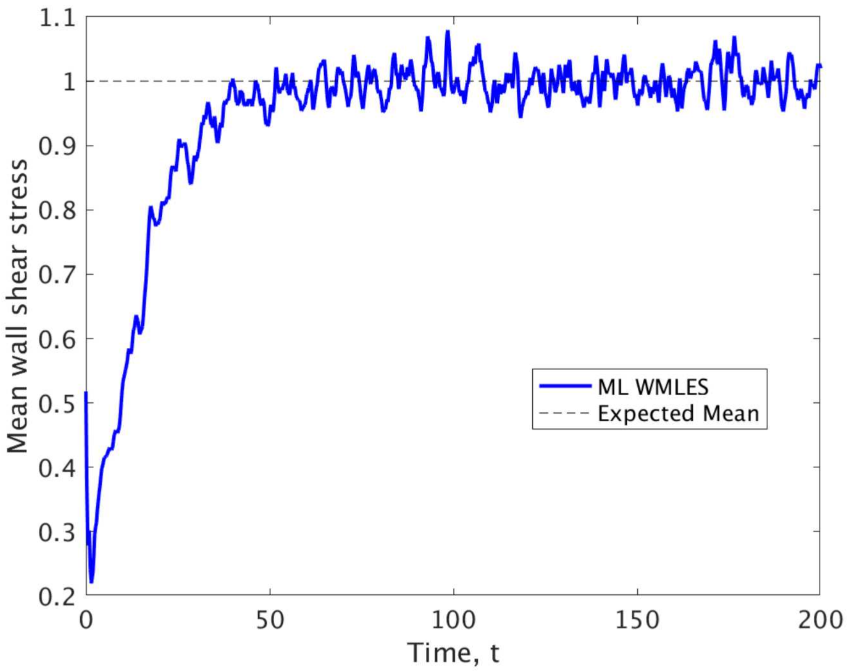
- Novel Siamese-MLP used to determine coefficients for vector polynomial for stresses on all walls

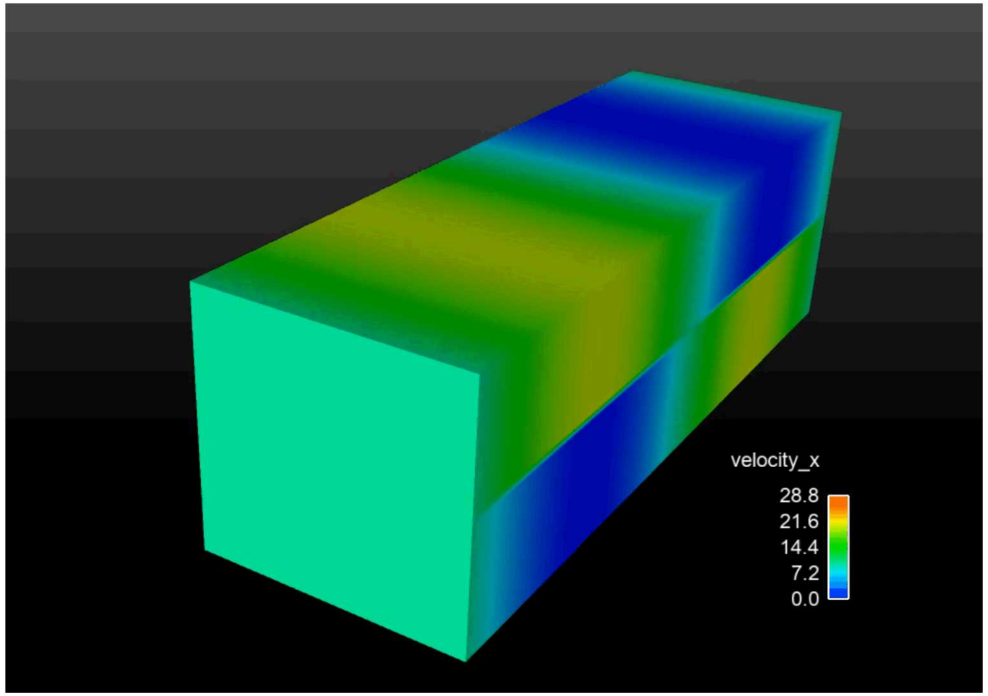


	Before Step MSE	Before Step KLD	Detached Region MSE	Detached Region KLD
Invariant Siamese-MLP	1.07e-6	0.16	1.02e-6	0.86
Law of the Wall	1.14e-6	4.69	1.49e-6	4.47

Incorporation of NN near-wall models into an LES Code (with Stefan Domino)

- Run Nalu low-Mach code on turbulent channel flow, Re_tau = 2000
- Output wall shear stress and local velocity vector as training data
- Train a 3-layer neural network to reproduce the Nalu near-wall model
- Implement neural network model in Nalu using matrix-vector products, **neglect wall-model Jacobian in implicit solve*



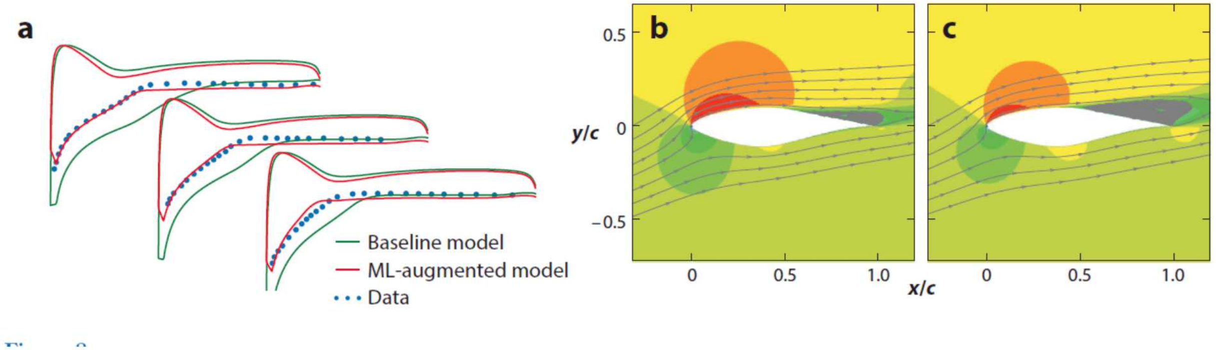


Other Contributions in Machine-Learned Turbulence Models

Karthik Durasaimy (U. of Michigan), and inverse modeling:

- Insert discrepancy terms into a RANS model
- Solve an inverse problem for the discrepancy field that corrects the solution relative to the qty's of interest.
- Use ML to map the local flow state to the local discrepancy.

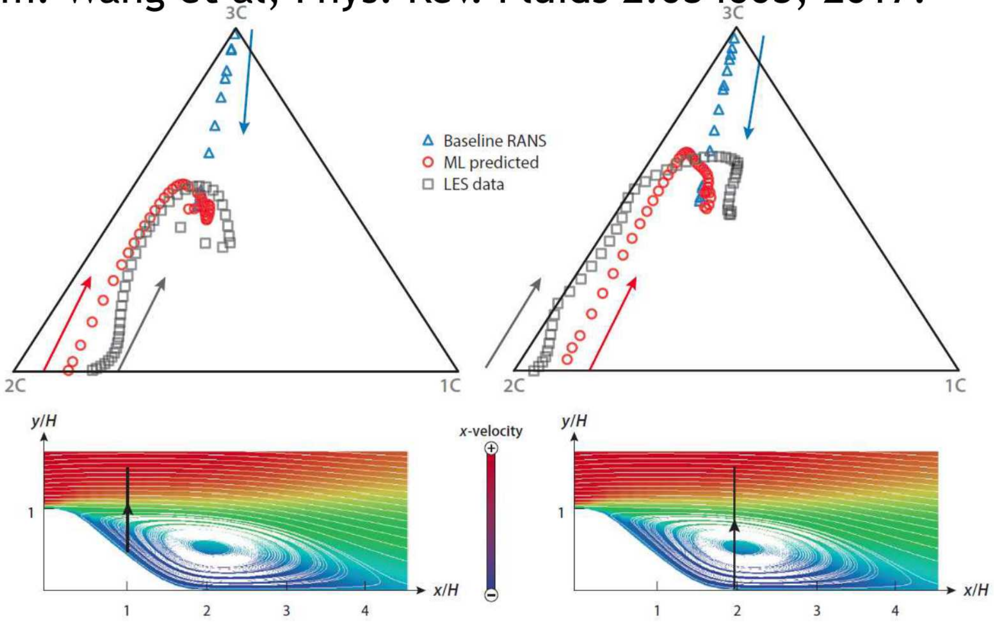
From: Singh et al, AIAA J. 55:2215-27, 2017.



Heng Xiao (Virginia Tech) Reynolds stress eigenvalue approach:

- Use ML to represent discrepancies between DNS/LES and RANS Reynolds stress in terms of stress tensor eigenvalues and eigenfunctions
- Incorporates frame-independence and smoothness requirements on the ML-model

From: Wang et al, Phys. Rev. Fluids 2:034603, 2017.



Opportunity: Turbulence Modeling Renaissance?

Traditional Method

1. Development of a (sometimes elaborate and beautiful) mathematical/theoretical basis.
2. Simplifications and approximations due to lack of knowledge.
   - Model form uncertainty
   - Model parameter uncertainty

A model with a sensible overall structure but with deficiencies due to step 2.

Possible New Approach

1. Re-visit and dust off some of the existing theoretical work on turbulence.
2. Make fewer approximations and simplifications – ***insert data-driven modeling here***.

A model that is more true to the foundational theory.

- New life for second-moment closures, structure-based turbulence models, PDF models, … ?
- Can the additional model complexity be absorbed by machine learning approaches while maintaining robustness and generality?

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 → convergence

Credibility and “Credibility”

How does one verify a machine-learned turbulence model?

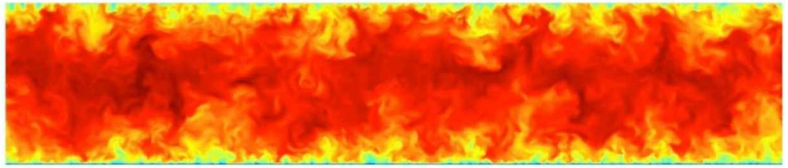
How does one select training data?

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%**



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

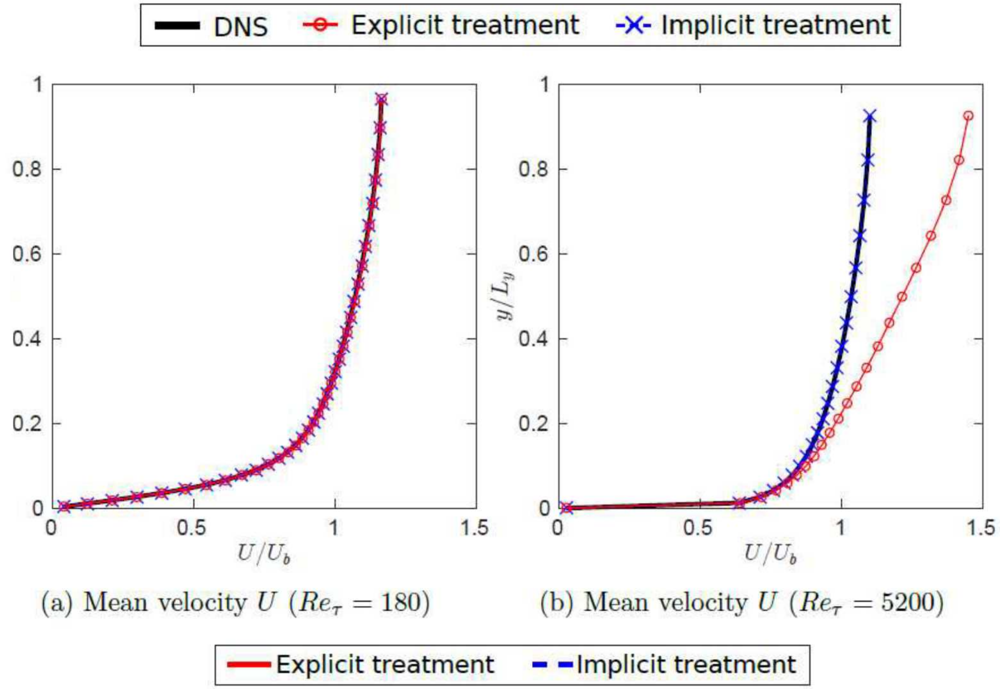
Use DNS data for the Reynolds stress

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

- Wu *et al.*, *"RANS equations with Reynolds stress closure can be ill-conditioned"*, *arXiv:1803.05581* (2018), provide a local model conditioning metric and give an example of how an implicit treatment of a data-driven model can improve accuracy.

$$\nu_t^m = \arg\min_{\nu_t} ||\boldsymbol{\tau}^{DNS} - 2\nu_t \mathbf{S}^{DNS}|| \qquad (22)$$

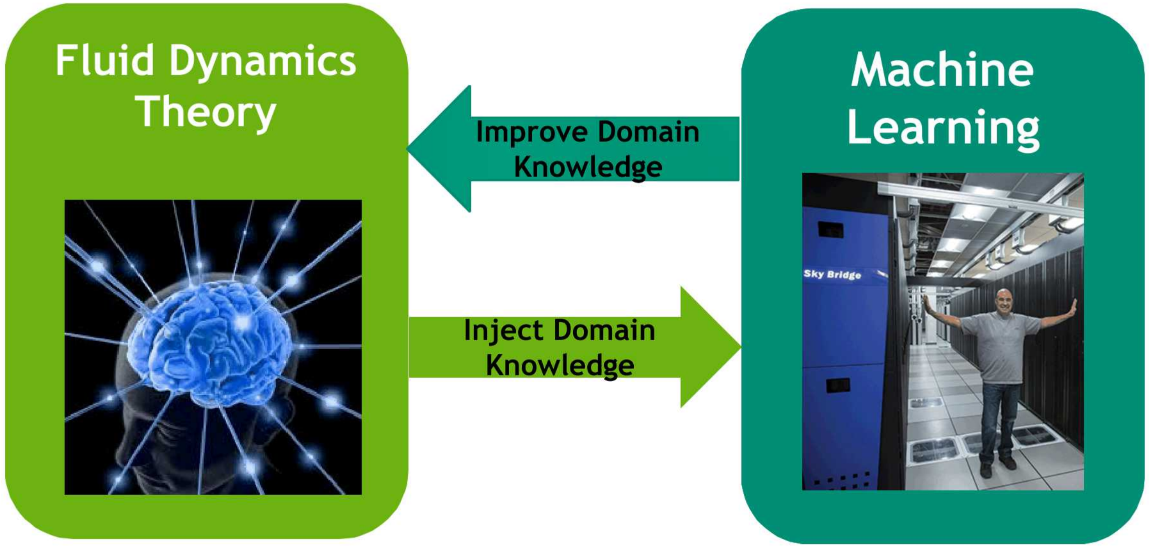
Compute optimal eddy viscosity ν_t^m from DNS Reynolds stresses based on Eq. (22)
for *each iteration step* $i = 1, 2, ..., N$ **do**
  Compute Reynolds stress: $\boldsymbol{\tau}^{(i)} = \nu_t^m \left(\nabla \overline{u}^{(i)} + (\nabla \overline{u}^{(i)})^T\right) + \boldsymbol{\tau}_{\text{DNS}}^{\perp}$
  Solve the RANS equations: $\mathcal{N}\left(\overline{u}^{(i)}\right) = \nabla \cdot \boldsymbol{\tau}^{(i)}$ to obtain $\overline{u}^{(i)}$
end



(a) Mean velocity U ($Re_\tau = 180$)
(b) Mean velocity U ($Re_\tau = 5200$)

Other issues: differentiability of ML models; linear and non-linear stability
Do we need "Machine-Learning-Resilient Numerics"? (Domino)
or
Do we need "Numerically Robust Machine Learning Models"?

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.

Takeaways

- There are signs that, whether you like it or not ☺, we are at the beginning of a "data-driven era" in turbulence modeling.
- It is worthwhile to view machine learning methods in different lights: both as useful tools with which to investigate turbulence, as well as offering possible modeling solutions.
- The following three data-driven turbulence modeling research challenges have been identified, that should be germaine across different physics applications.
  - Credibility of Data-driven Modeling Approaches
  - Numerical Solution of PDE's with Data-driven Models
  - Learning from Machine-Learned Models