VirginiaTech
Invent the Future

Towards A Physics-Informed Machine Learning Framework for Predictive Turbulence Modeling

Heng Xiao and Jinlong Wu

Department of Aerospace & Ocean Engineering
Virginia Tech

NASA/UMich Symposium, July 12, 2017

AEROSPACE & OCEAN ENGINEERING
AT VIRGINIA TECH

Acknowledgment of Contributors

Dr. Jianxun Wang

Dr. Rui Sun

Also:

Dr. Julia Ling

Dr. Eric Paterson

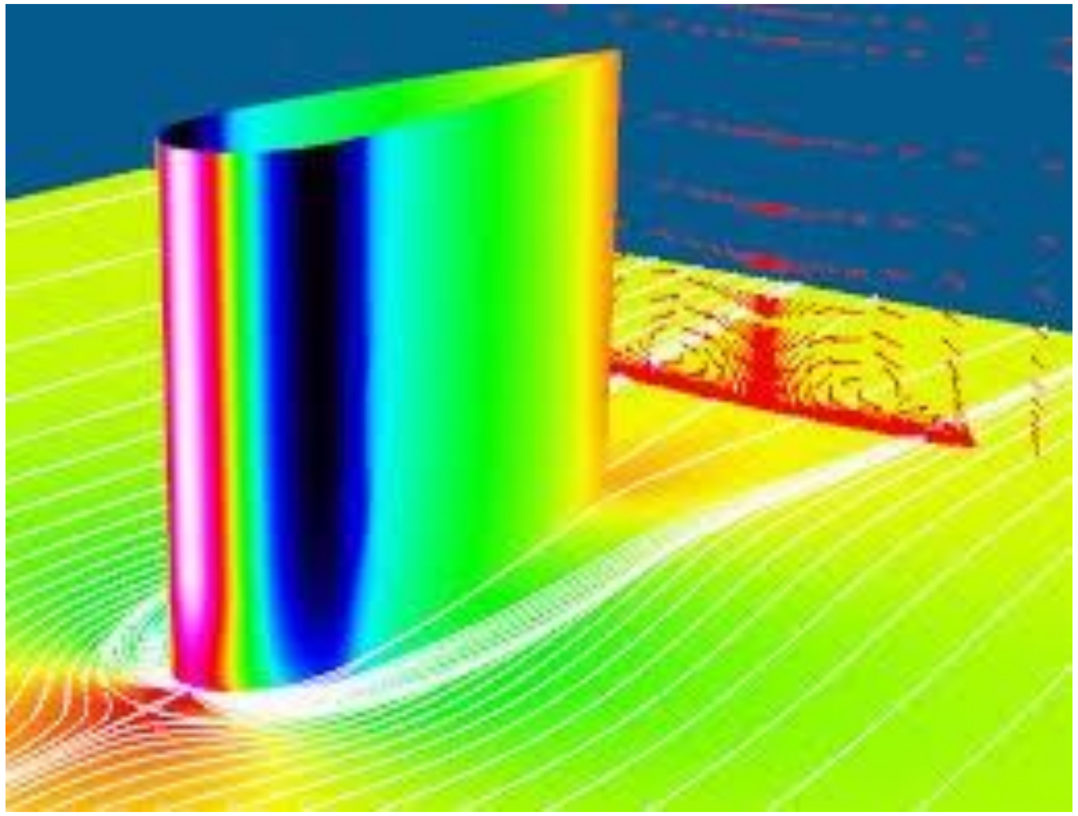
Dr. Qiqi Wang

Virginia Tech
Invent the Future

AOE
Aerospace & Ocean Engineering at Virginia Tech

A Scenario for RANS-Based Design/Optimization

Calibration Cases (offline data)	Prediction Cases (no data)
A few configuration with data (DNS or experimental measurements)	Similar configuration with different: • Twist • Sweep angles • Airfoil shape



How to leverage data to complement RANS models?

machine learning?

Machine Learning is an Umbrella Term

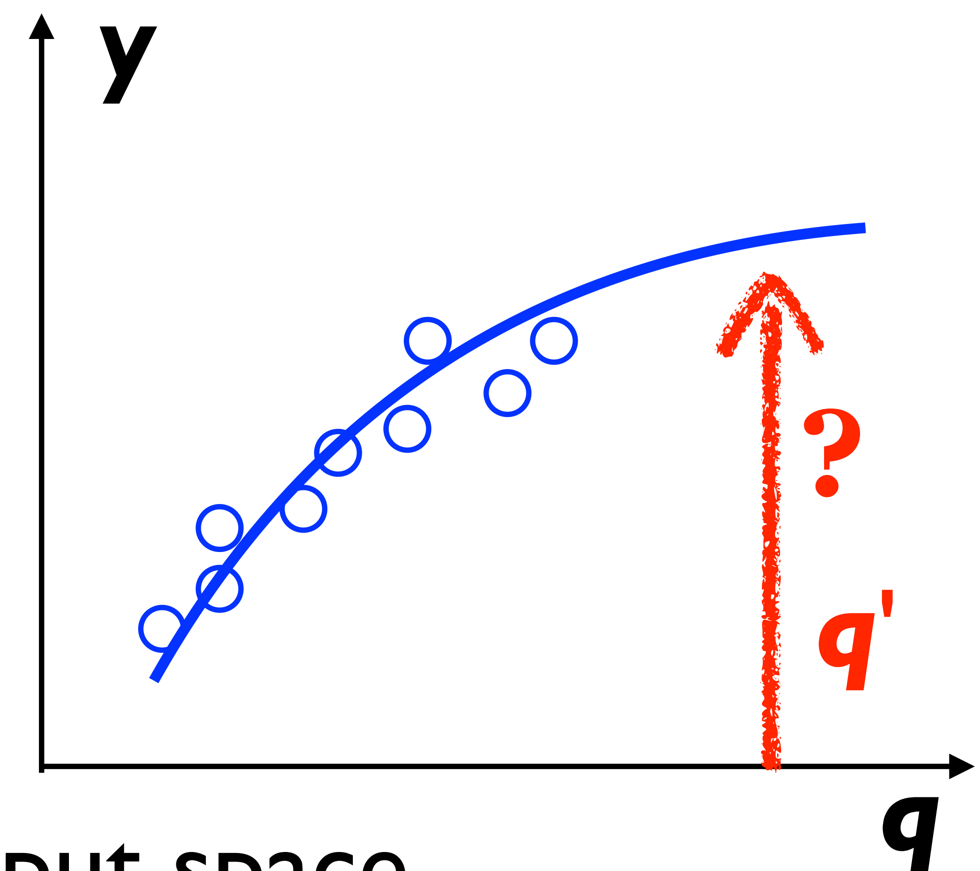
(Supervised) machine learning in a nutshell:

1. pose a function mapping from input ***q*** to output ***y***, controlled by parameters ***W***

2. fit (learn) the mapping to training data by optimizing the parameters ***W***

3. predict ***y*** for unseen inputs

$$\mathbf{q} \overset{\mathbf{W}}{\mapsto} \mathbf{y}$$



ML can handle high-dimensional input space.

Why do you expect a functional mapping between ***q*** and ***y***?

What can be corrected in RANS Simulations?

$$\frac{\partial U_i}{\partial t} + \underbrace{\frac{\partial (U_i U_j)}{\partial x_j}}_{\text{convection}} + \underbrace{\frac{1}{\rho}\frac{\partial p}{\partial x_i}}_{\text{pressure grad.}} - \nu \underbrace{\frac{\partial^2 U_i}{\partial x_j \partial x_j}}_{\text{diffusion}} = \nabla \cdot \boldsymbol{\tau}$$

$$\frac{D\tilde{\nu}_t}{Dt} = P(\tilde{\nu}_t, \mathbf{U}) - D(\tilde{\nu}_t, \mathbf{U}) + T(\tilde{\nu}_t, \mathbf{U}) \rightarrow \nu_t$$

$$\boldsymbol{\tau} = \frac{\nu_t}{2}\left(\nabla \mathbf{U} + \nabla^t \mathbf{U}\right)$$

What output **y** we want to learn from data?

$$\mathbf{q} \overset{\mathbf{w}}{\mapsto} \mathbf{y}$$

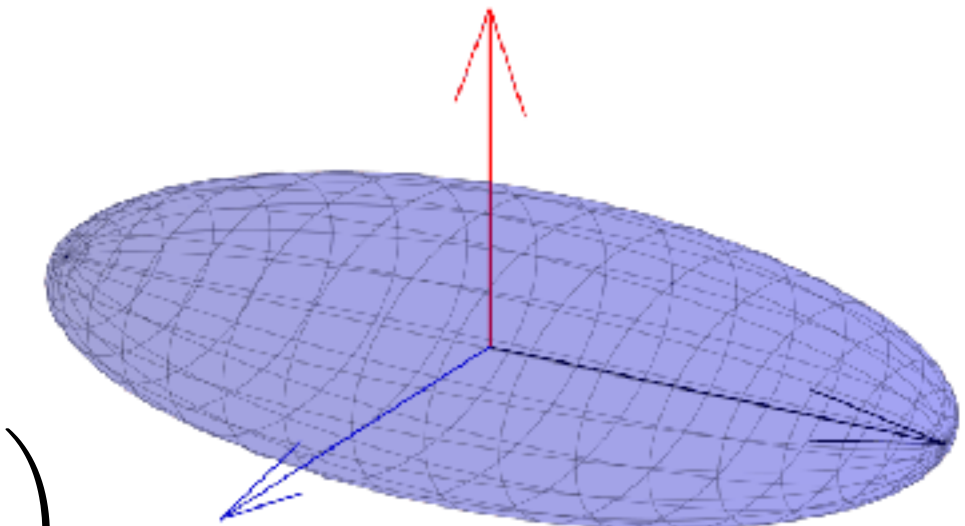
Discrepancies in:

- Model coefficients
- Unclosed terms in the transport equations (Duraisamy)
- RANS-predicted eddy viscosity
- RANS predicted Reynolds stress

$$\Delta \boldsymbol{\tau}$$

Representation of Reynolds Stress Discrepancies

- Barycentric triangle (realizability map) provides a bound of all realizable Reynolds stresses.



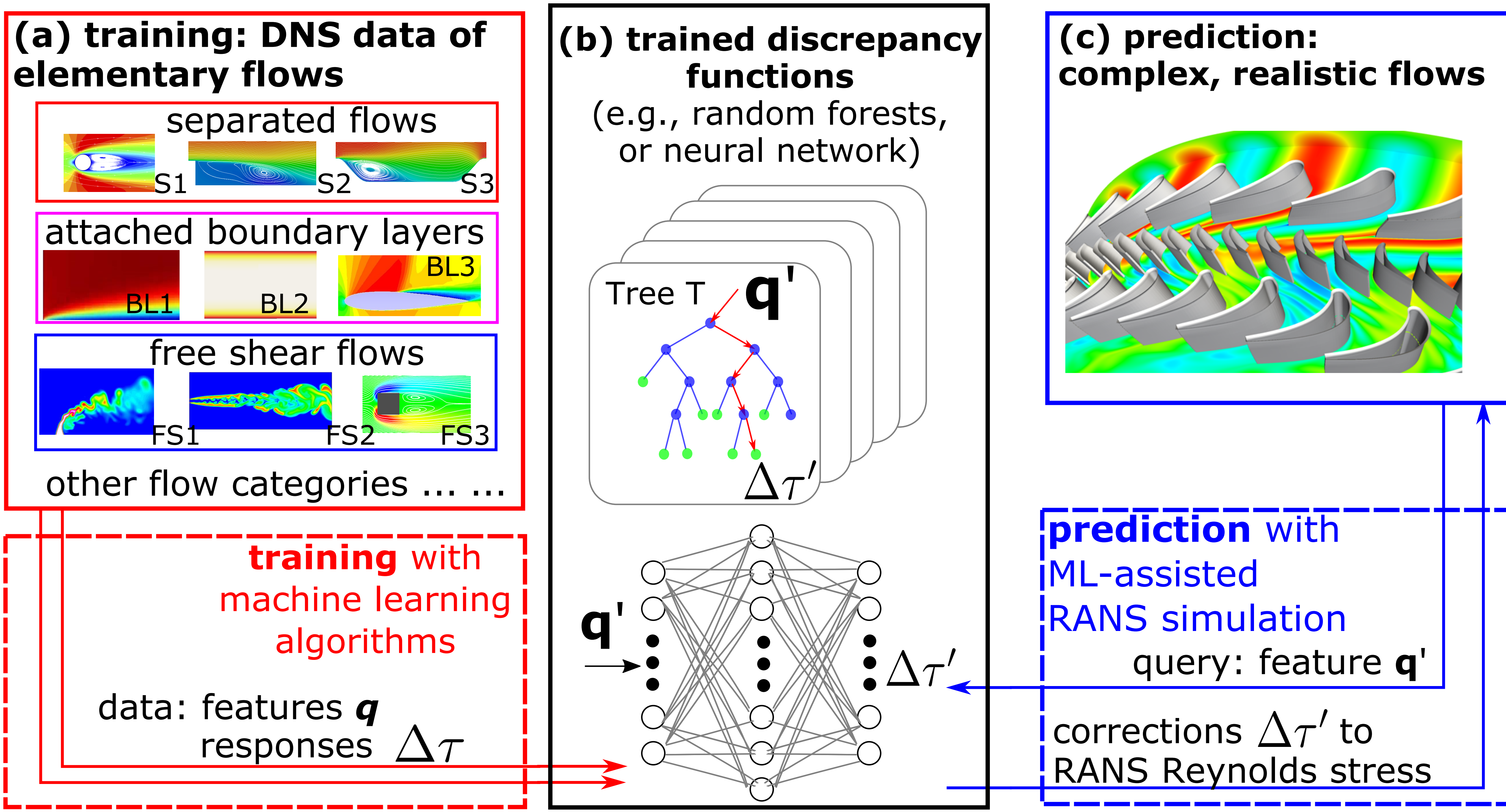
$$\boldsymbol{\tau} = 2k\left(\frac{1}{3}\mathbf{I} + \mathbf{a}\right) = 2k\left(\frac{1}{3}\mathbf{I} + \mathbf{V}\Lambda\mathbf{V}^T\right)$$

- You can perturb Reynolds stress $\boldsymbol{\tau}^{\mathrm{rans}}$ to stress $\boldsymbol{\tau}^*$ by changing the size, aspect ratio, and orientation: preserve realizability & orthogonality
- Use ML to learn the **perturbations** needed to transform $\boldsymbol{\tau}^{\mathrm{rans}}$ to $\boldsymbol{\tau}^*$!

"physics-informed"

[Iaccarino et al.]

Physics-Informed Machine Learning for Predictive Turbulence Modeling



Construction of Feature Space

$$\{S, \Omega, \nabla p, \nabla k, Re_d, \mathcal{P}/\varepsilon, k/\varepsilon, \kappa\}$$

4 tensors/vectors; 47 invariants (integrity bases)

- Invariants of 4 tensors/vectors: strain rate (S), rotation rate (Ω), pressure (p) gradient, TKE (k) gradient: draw 4 scalars: streamline curvature (κ), wall-distance based Reynolds number (Re_d), turbulent time scale
- (Normalized) feature vector **q** has a length of ~50.
- Choice of features inspired by **advanced turbulence models.**

Objective: train discrepancy functions $\Delta\tau(\mathrm{q})$

(Ling et al. JCP 2017; Wang et al. CTR Proceeding 2017)

Non-Dimensionalization of Inputs

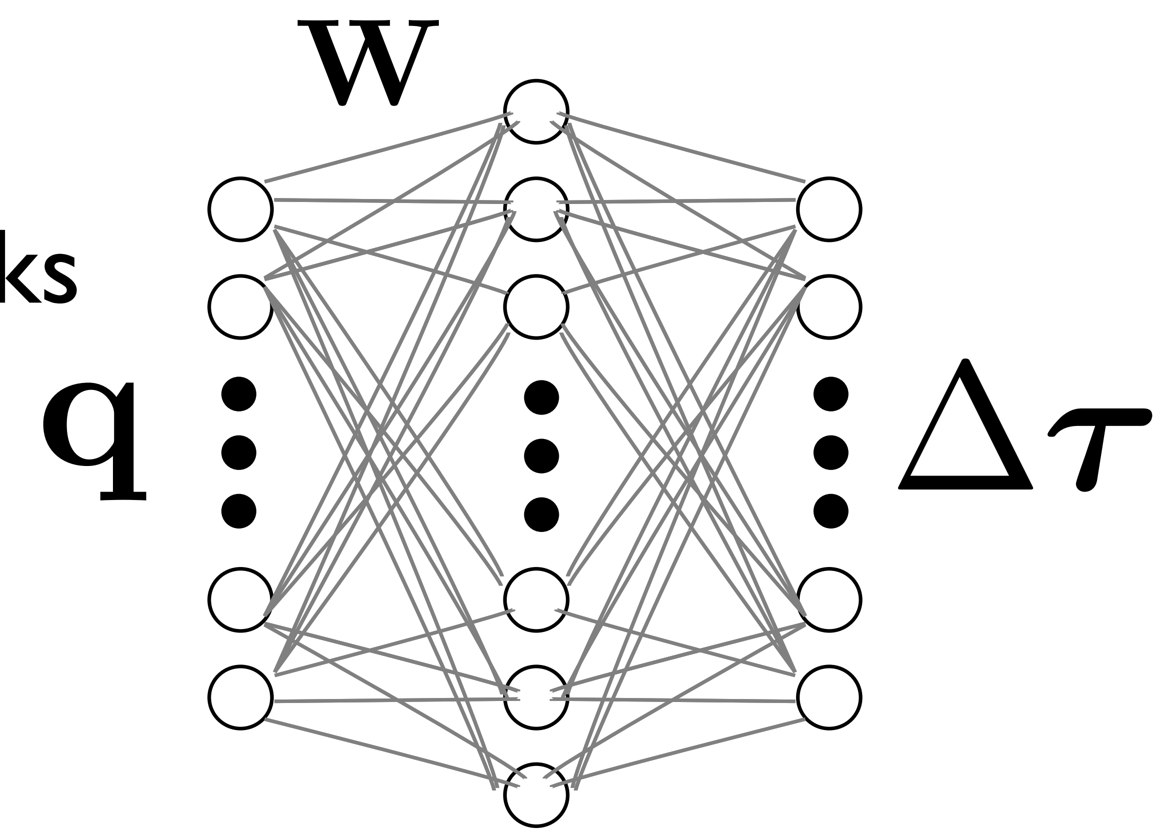
Normalized raw input $\hat{\alpha}$	description	raw input α	normalization factor β
$\hat{\mathbf{S}}$	strain rate tensor	$\mathbf{S}$	$\frac{\varepsilon}{k}$
$\hat{\mathbf{\Omega}}$	rotation rate tensor	$\mathbf{\Omega}$	$\|\mathbf{\Omega}\|$
$\widehat{\nabla p}$	Pressure gradient	∇p	$\rho\|\mathbf{U} \cdot \nabla \mathbf{U}\|$
$\widehat{\nabla k}$	Gradient of TKE	∇k	$\frac{\varepsilon}{\sqrt{k}}$

(Wang et al, CTR Proceeding 2017; Wu and Xiao, In preparation)

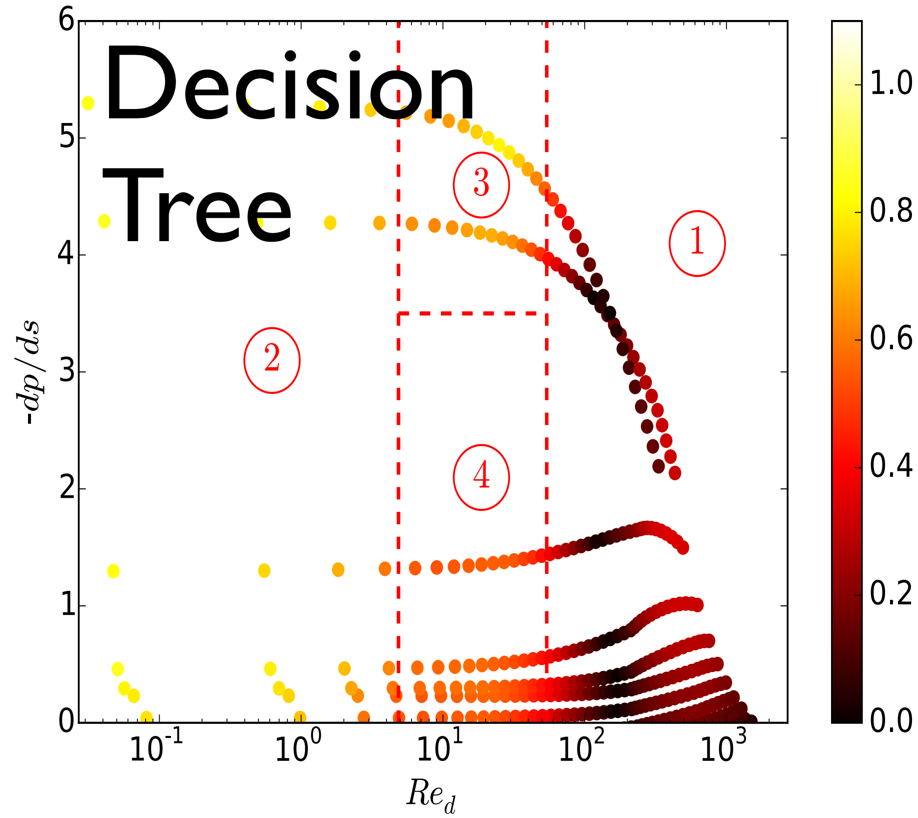
Machine Learning Techniques

$$\mathbf{q} \overset{\mathbf{w}}{\mapsto} \mathbf{y}$$

Neural Networks

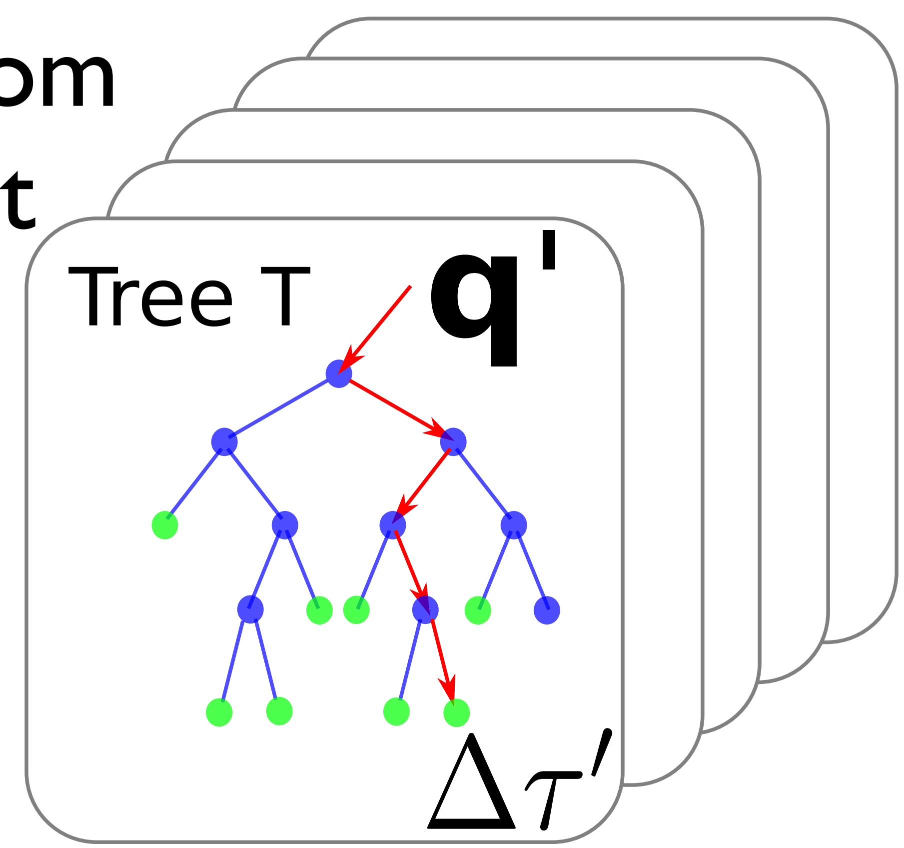


Decision Tree



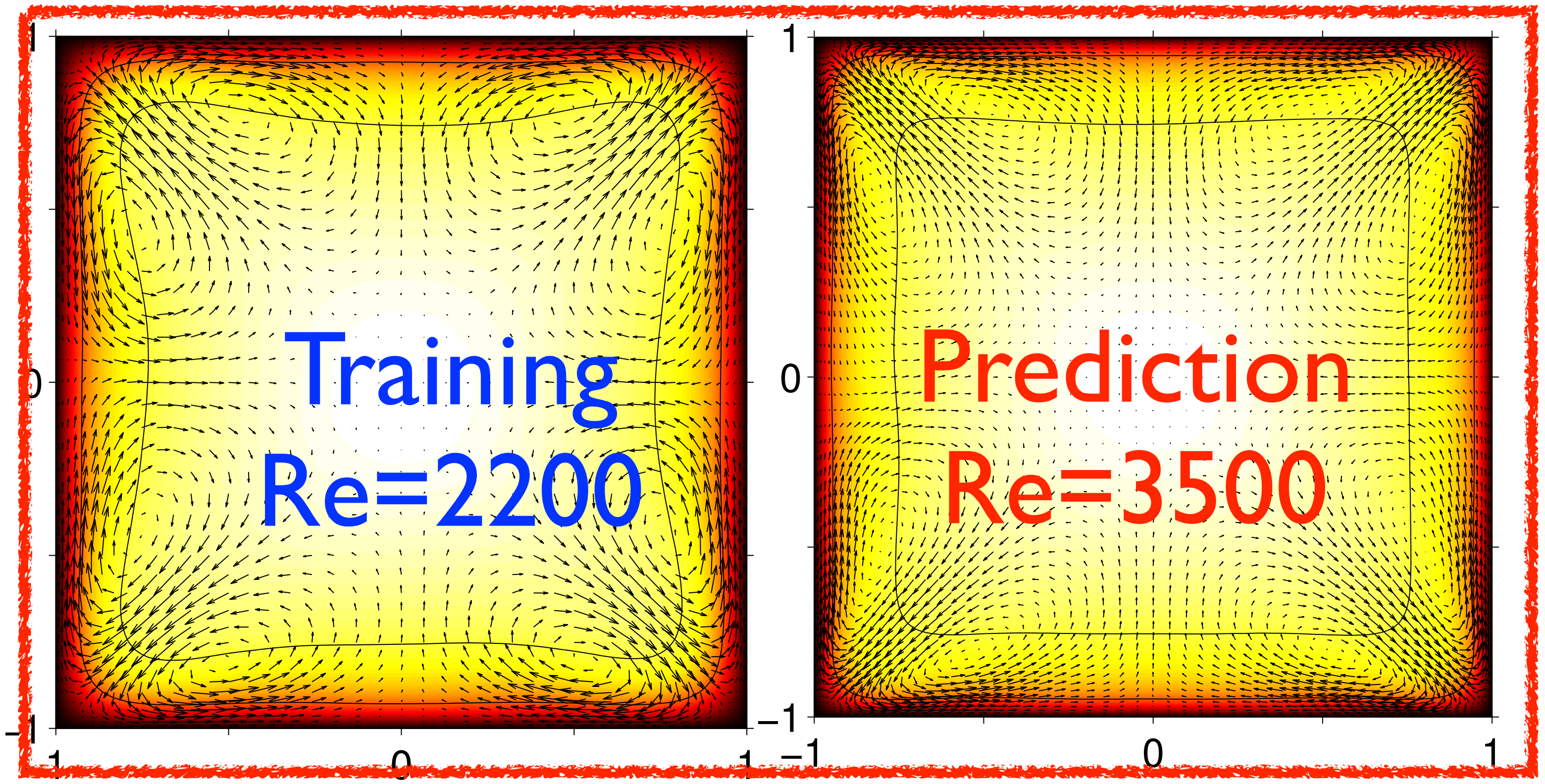
(a) feature space stratification

Random Forest

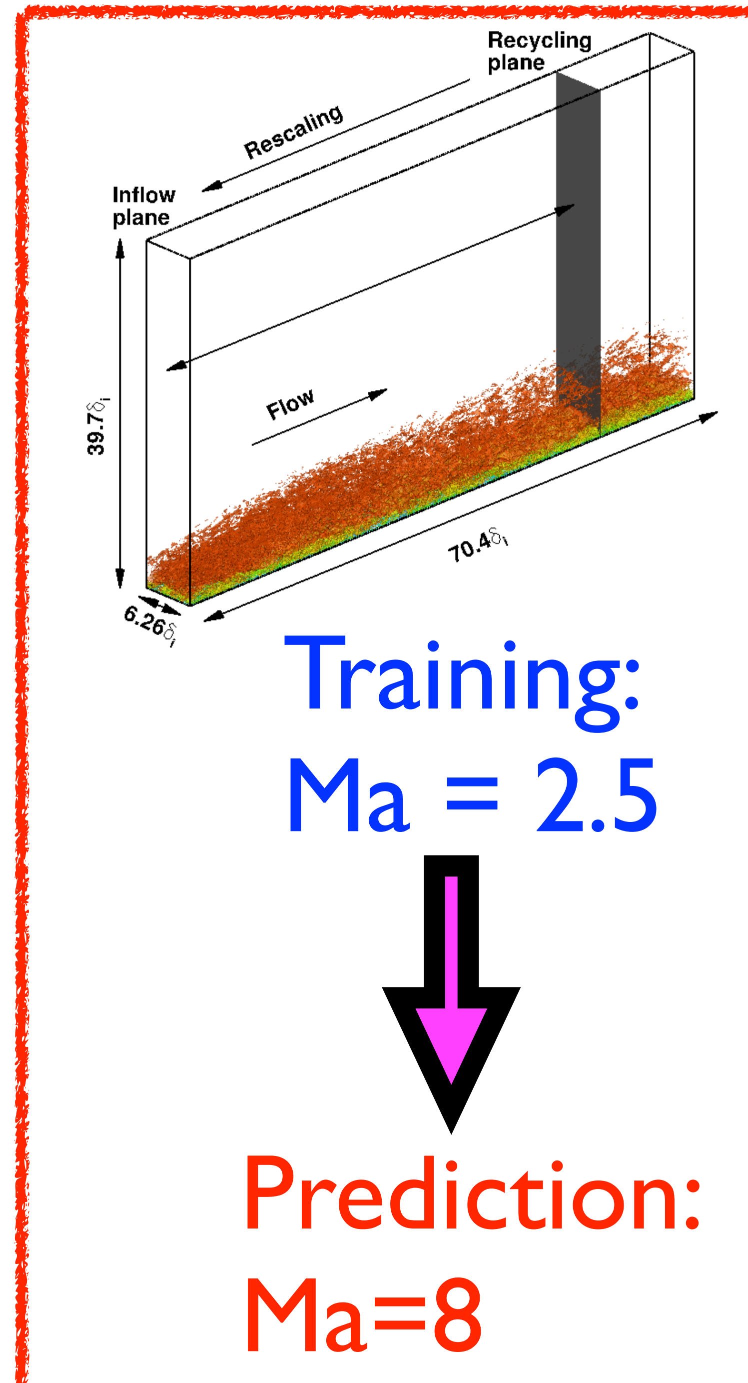


Summary of Current Progress

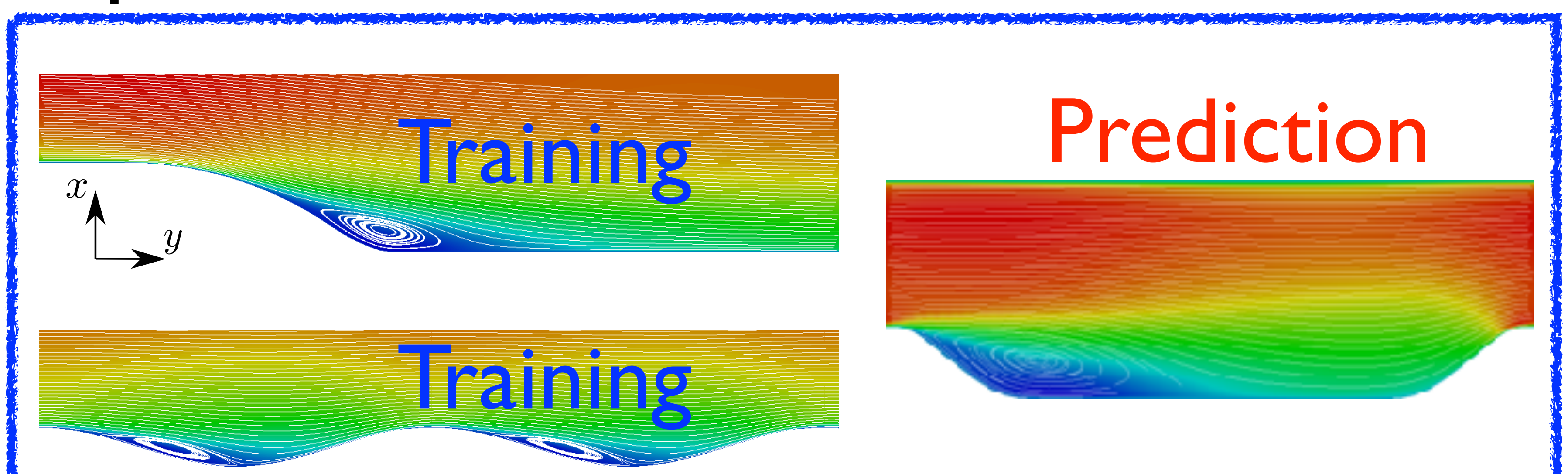
Square Duct Flows:



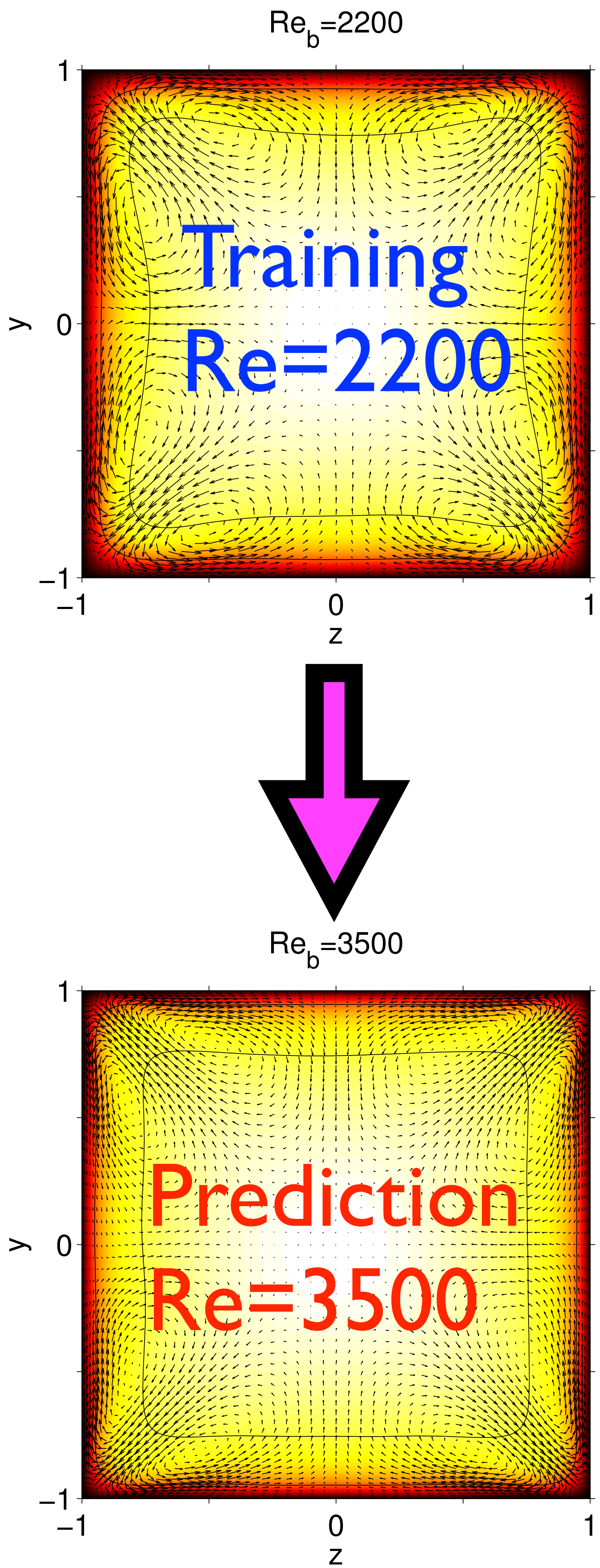
High-Mach Flat Plate BL:

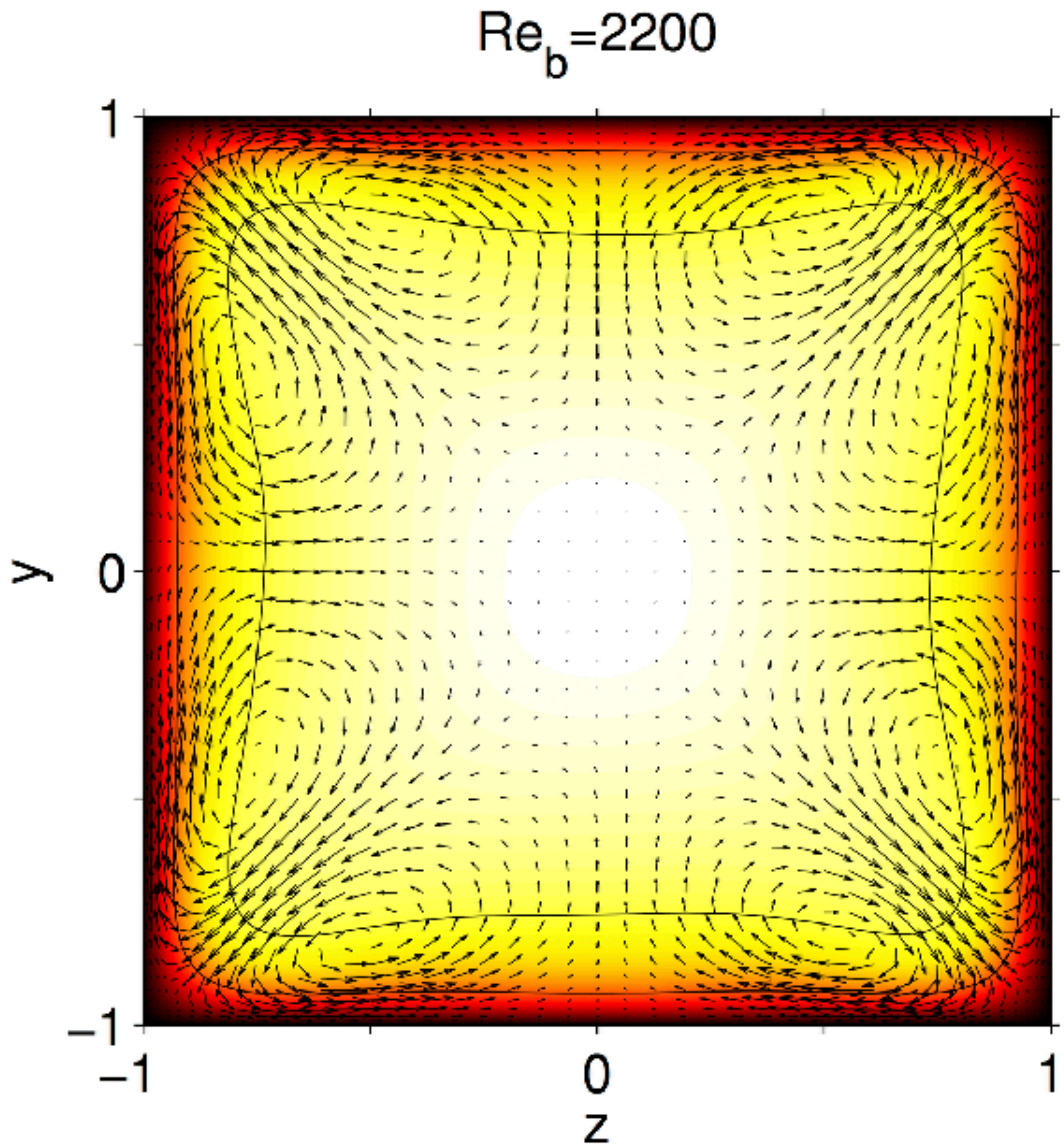


Separated Flows:



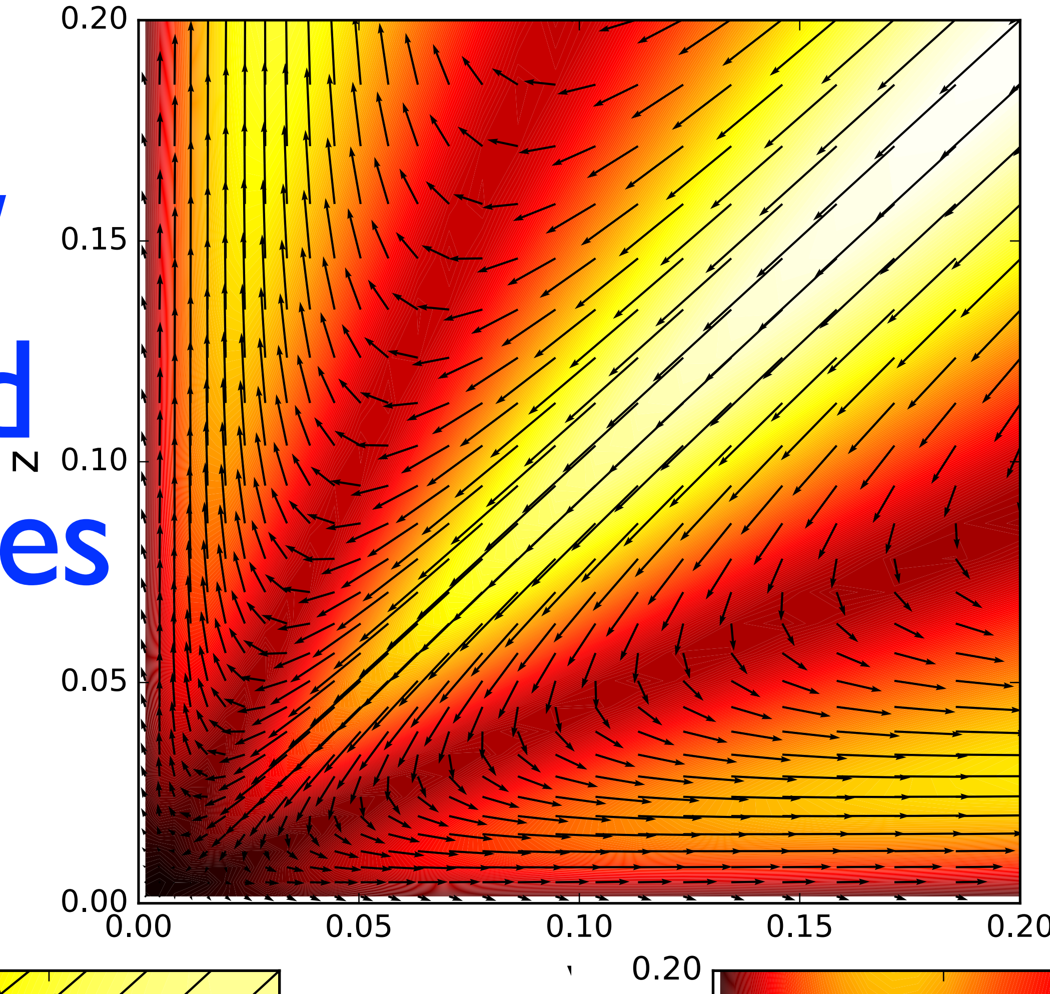
Test Case I: Turbulent Flows in Square Duct





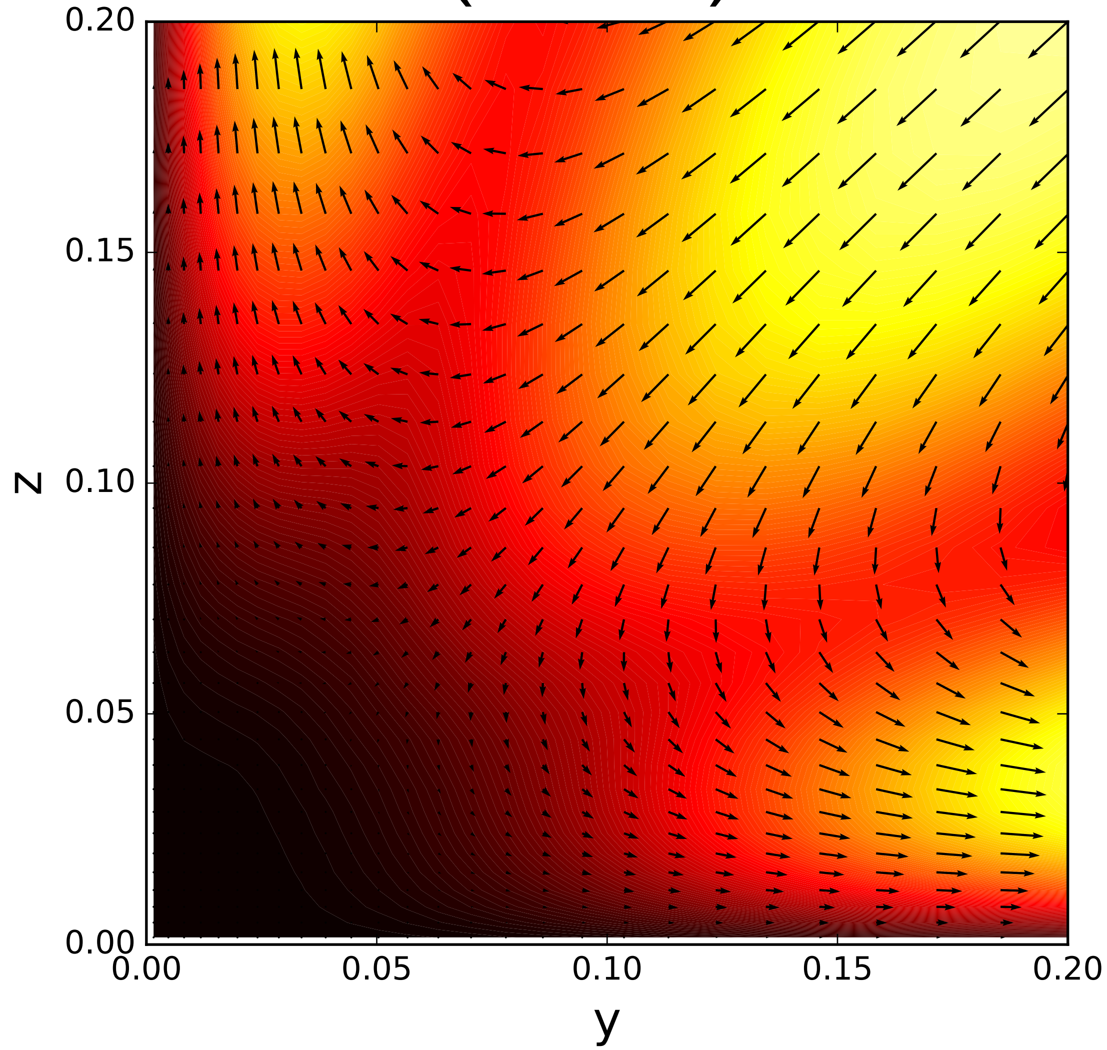
DNS data from (Uhlman et al. 2010 JFM)

Velocity Prediction w/ ML Corrected Reynolds Stresses

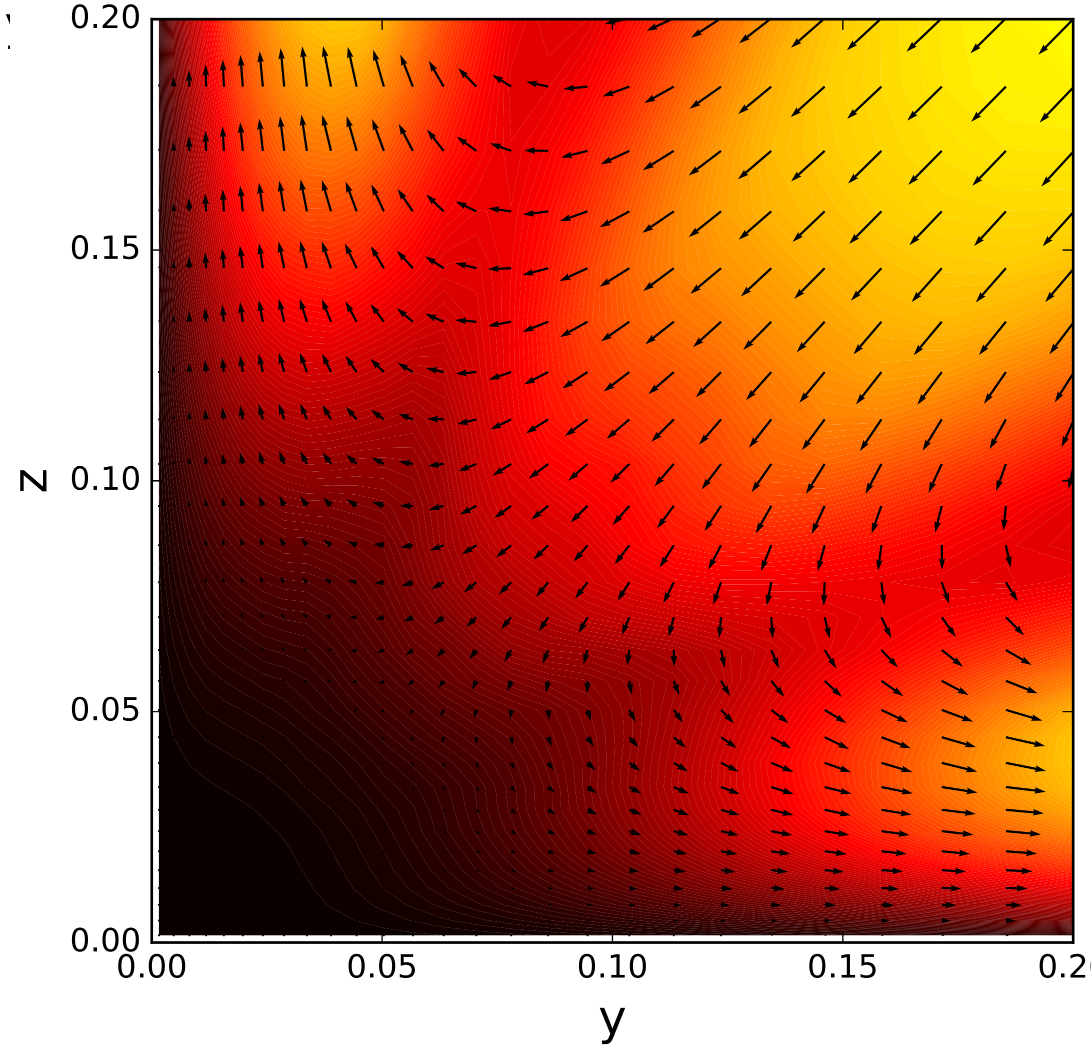


Baseline
(Launder-Gibson RSTM)

DNS (Truth)

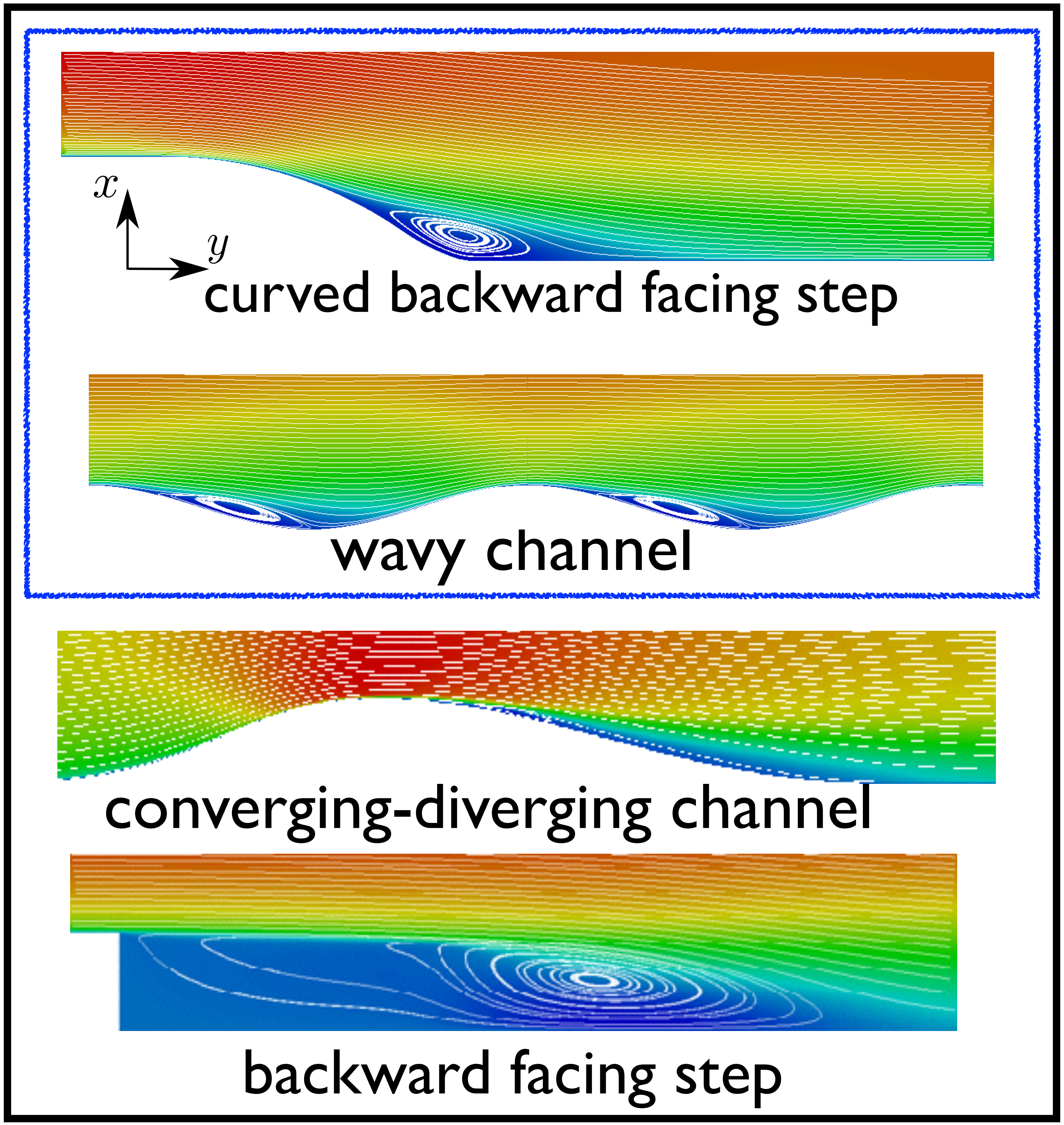


ML-Assisted Prediction



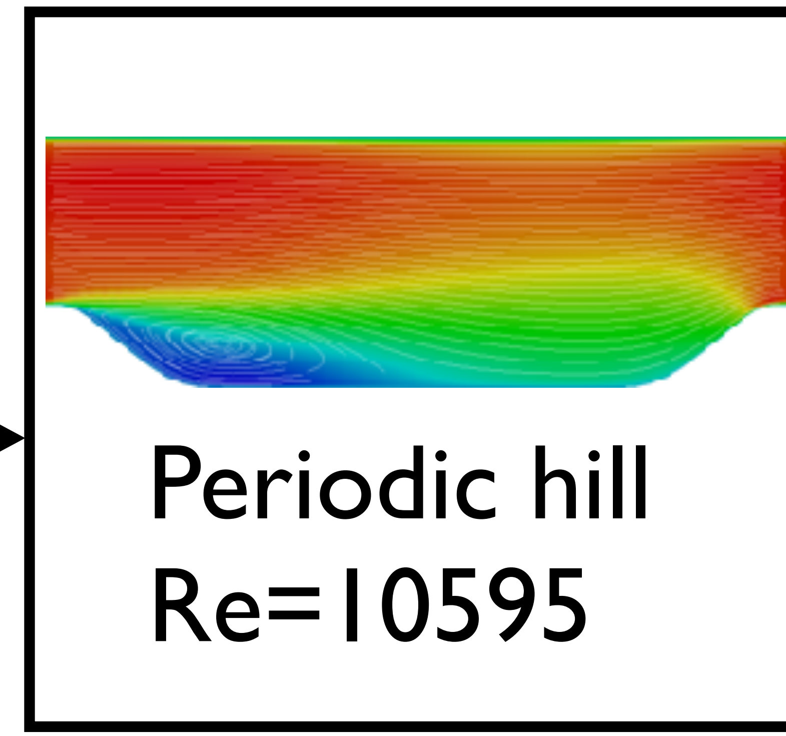
Test Case 2: Flows with Massive Separations

Training flows



Train discrepancy function $\Delta\tau(\mathbf{q})$

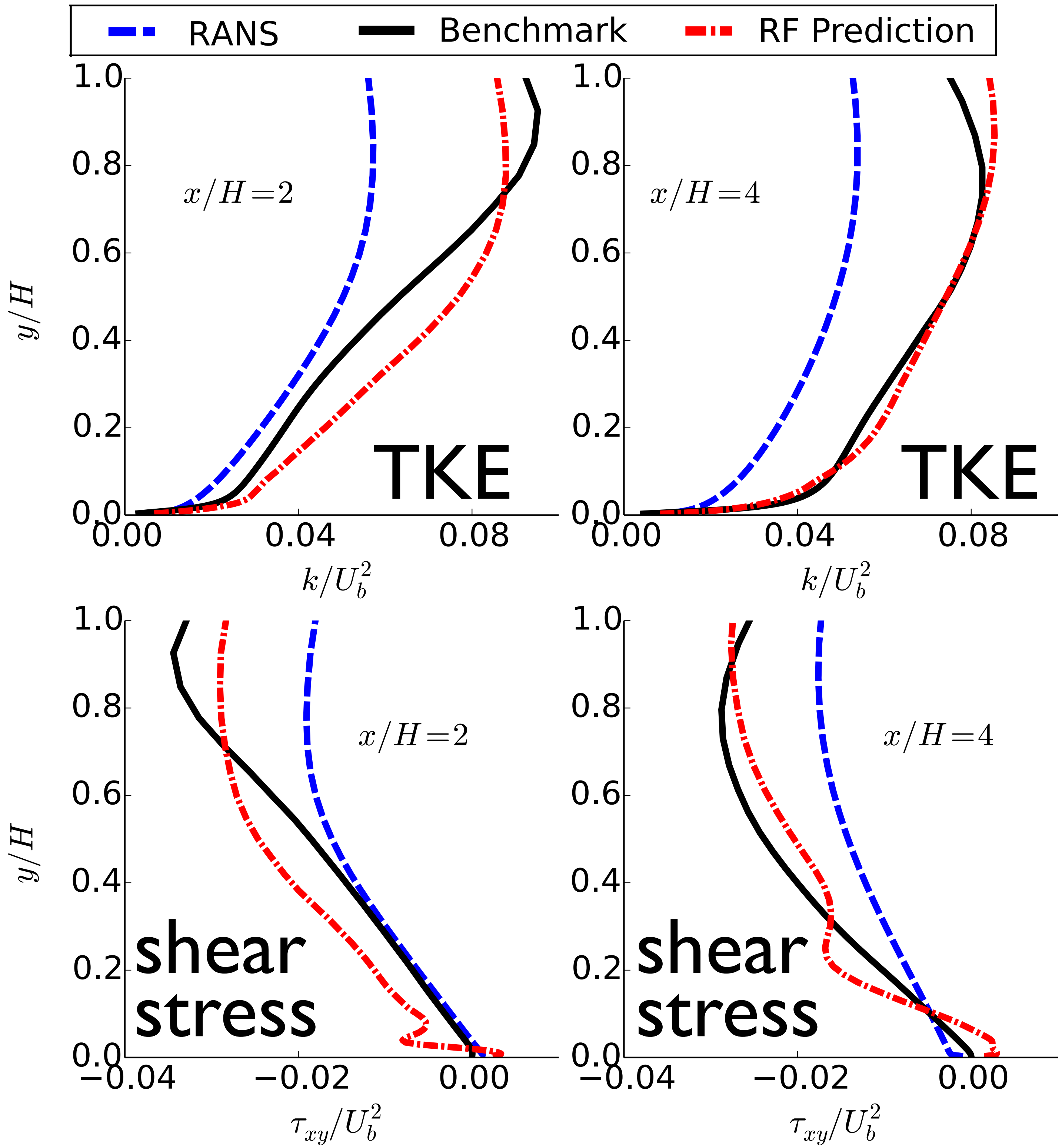
Prediction

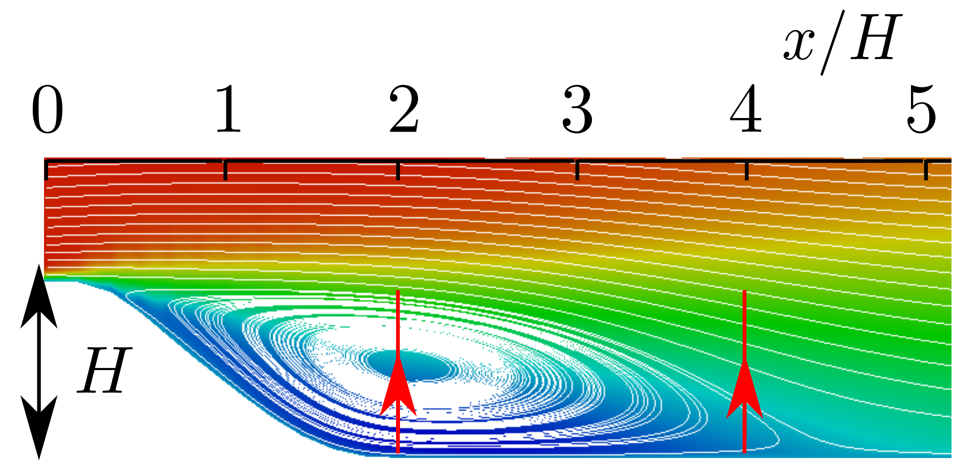


Source: http://turbmodels.larc.nasa.gov/other_dns.html

Blue: RANS | Black: LES (truth) | Red: Prediction

Predicted TKE and turbulent shear stress for Periodic Hill Re=10595





Reynolds stress improved; but what about the mean velocities?

(Wang, Wu & Xiao, PRF 2017)

Is Reynolds stress the right choice as the output of machine learning?

- Reynolds stress models are unstable
- No implicit treatment possible here in data-driven modeling.

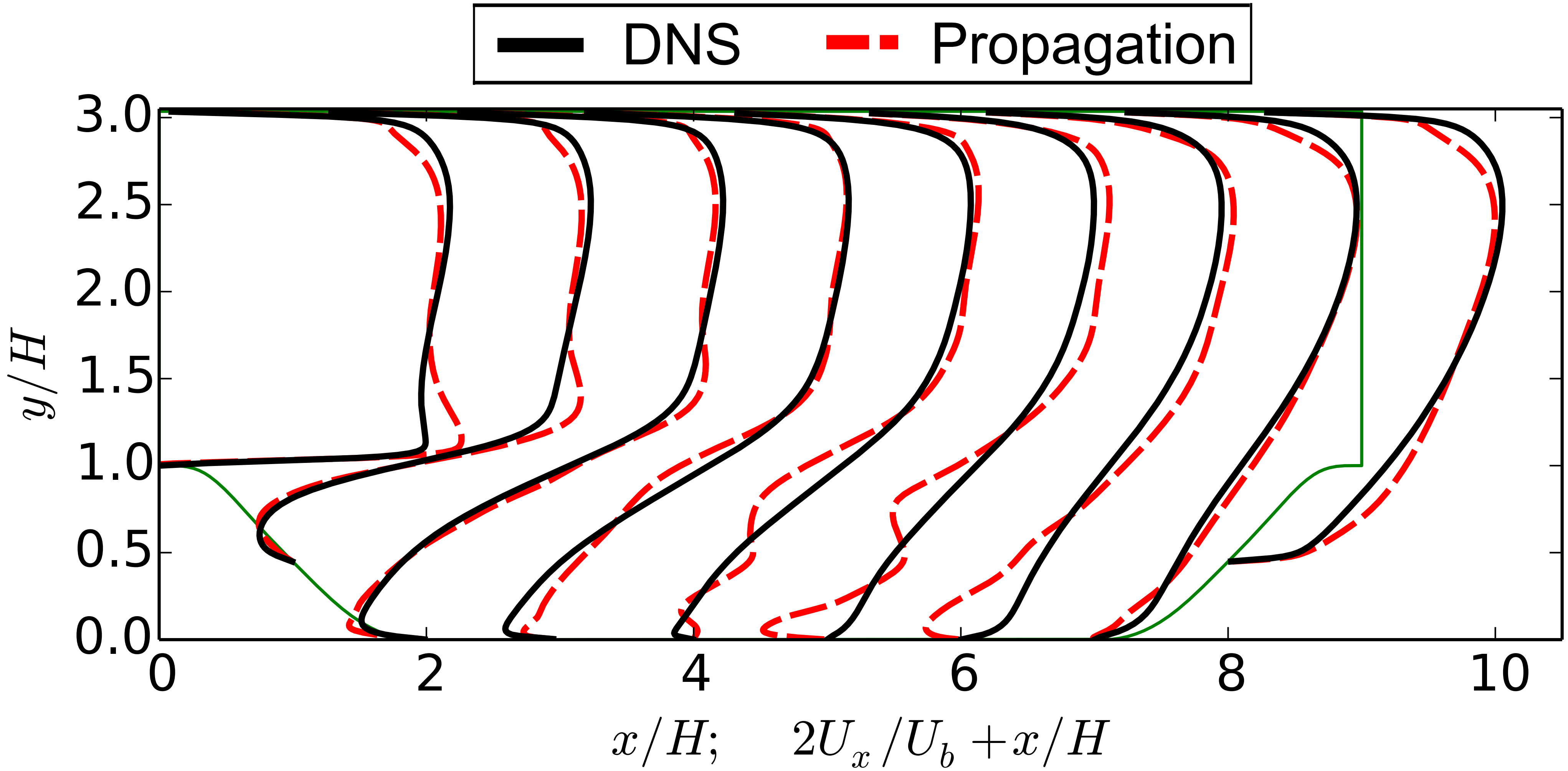
A Priori Studies: Propagating DNS Reynolds Stresses to Mean Velocities

$$\overline{\mathbf{u}} \cdot \nabla \overline{\mathbf{u}} - \nu \nabla^2 \overline{\mathbf{u}} + \nabla p - \nabla \cdot \boldsymbol{\tau} \quad = 0$$

Use Reynolds stresses from DNS

No model can give a better Reynolds stress than DNS data (EVM, algebraic/differential RSM or data-driven model).

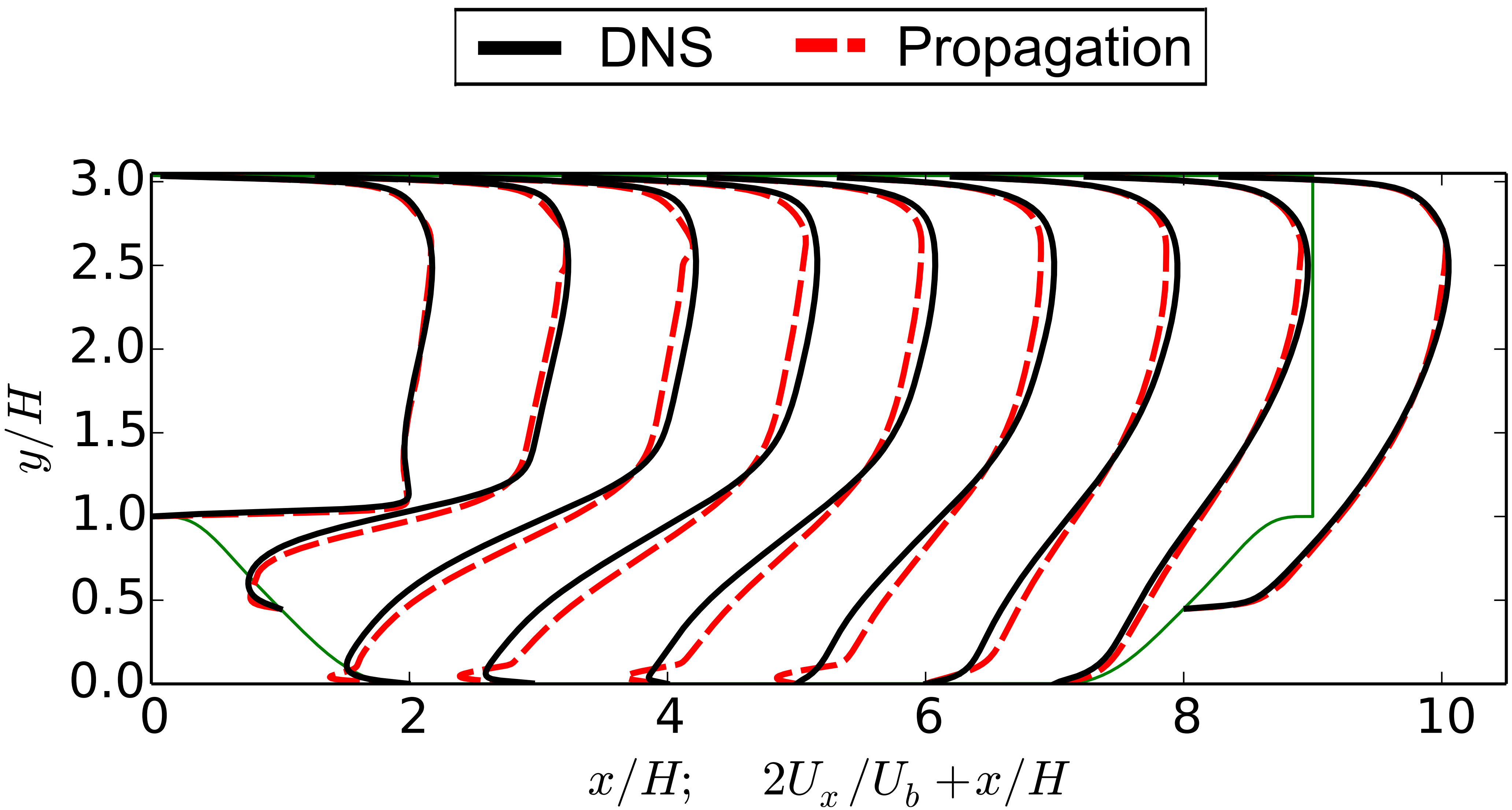
Velocity Propagated from DNS Reynolds Stress



DNS velocities vs. propagated velocities

- DNS data from (Breuer et al., 2009). Validated with new simulations by Laizet et al. (Imperial college)
- We proposed a concept of "**condition number for turbulence models**". Manuscript in preparation (Wu et al. 2017).

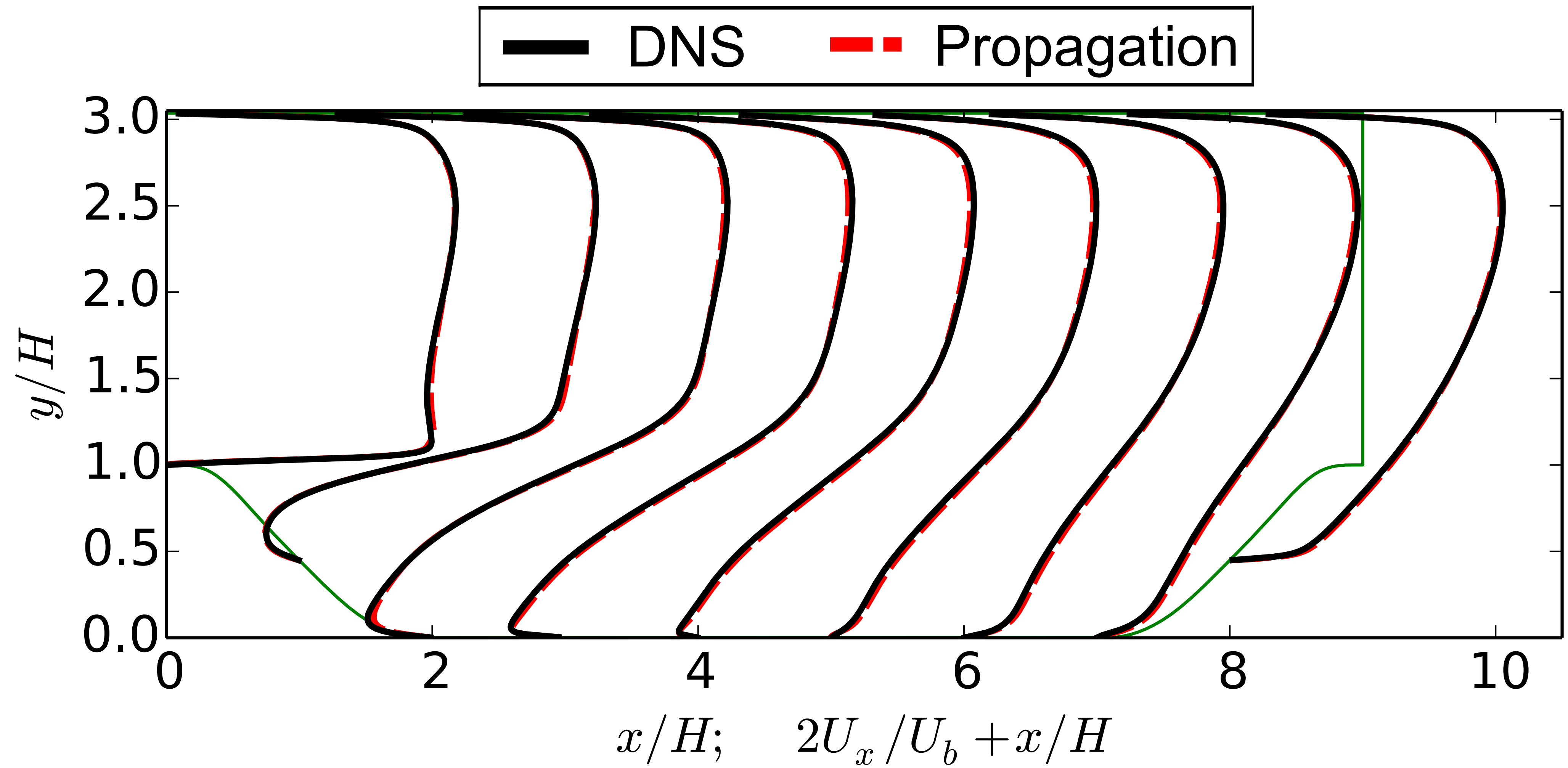
Propagated Velocity from DNS Eddy Viscosity



An "optimal" eddy viscosity field (in a least square sense) is obtained from DNS Reynolds stresses:

$$\nu_t^{LS} = \frac{\boldsymbol{\tau} : \mathbf{S} - \frac{2}{3} k \mathbf{I} : \mathbf{S}}{\mathbf{S} : \mathbf{S}}$$

Combining Reynolds Stress and Eddy Viscosity



Obtaining the eddy viscosity and non-linear component separately:

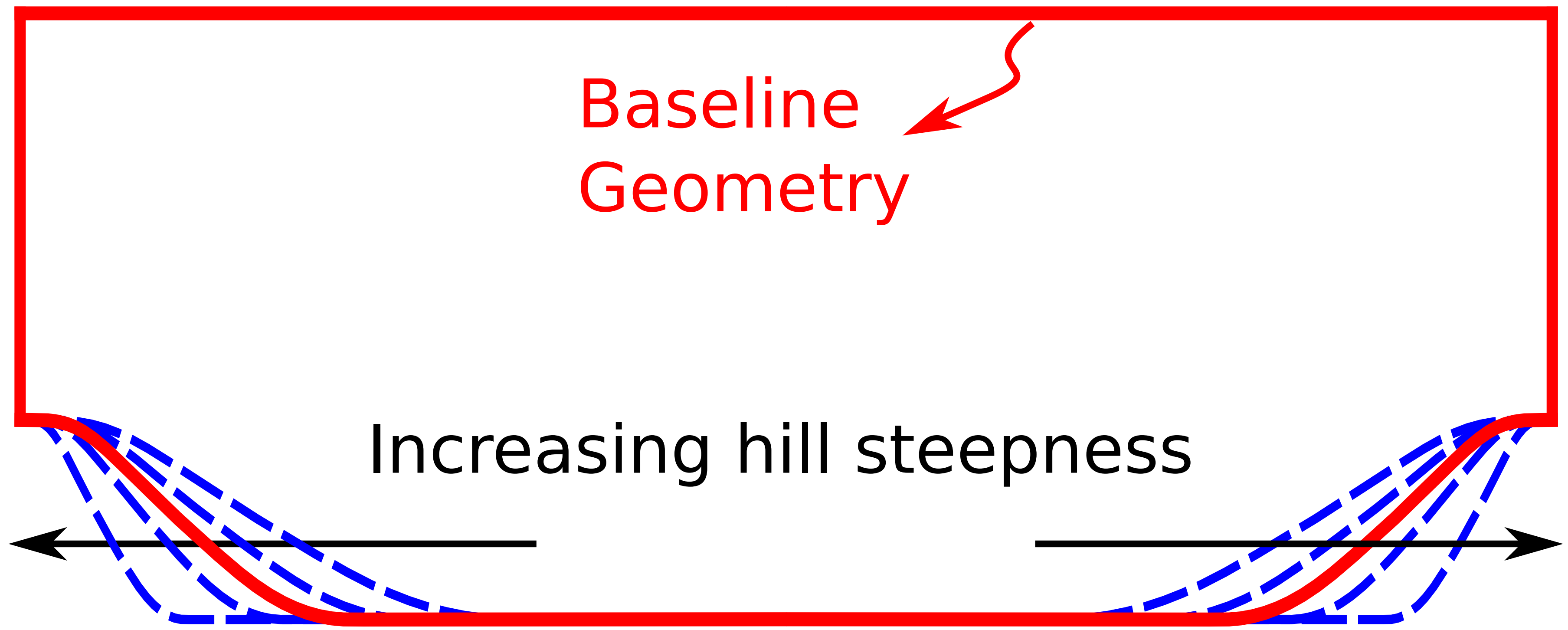
$$\boldsymbol{\tau} = \nu_t^{LS}\mathbf{S} + \boldsymbol{\tau}^{\perp}$$

Implicit Explicit

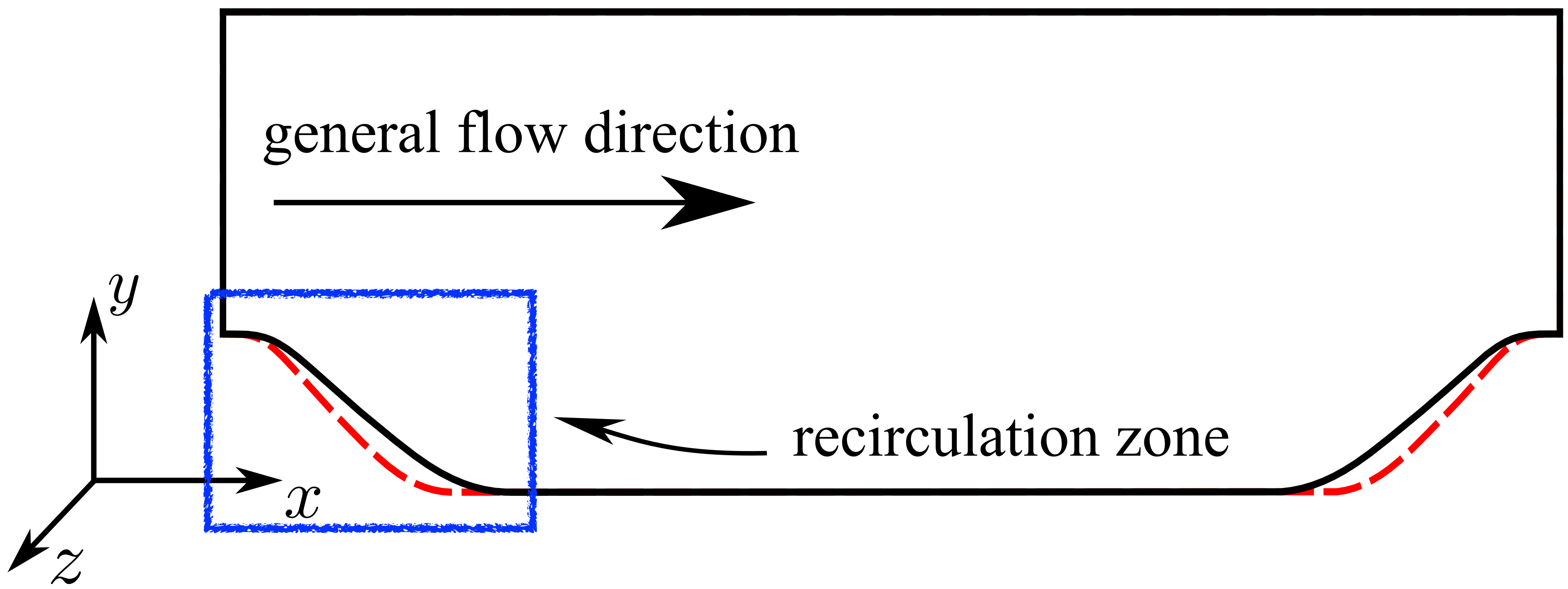
Take the Lessons Learned in *A Priori* Studies to Machine-Learning-Assisted Turbulence Modeling

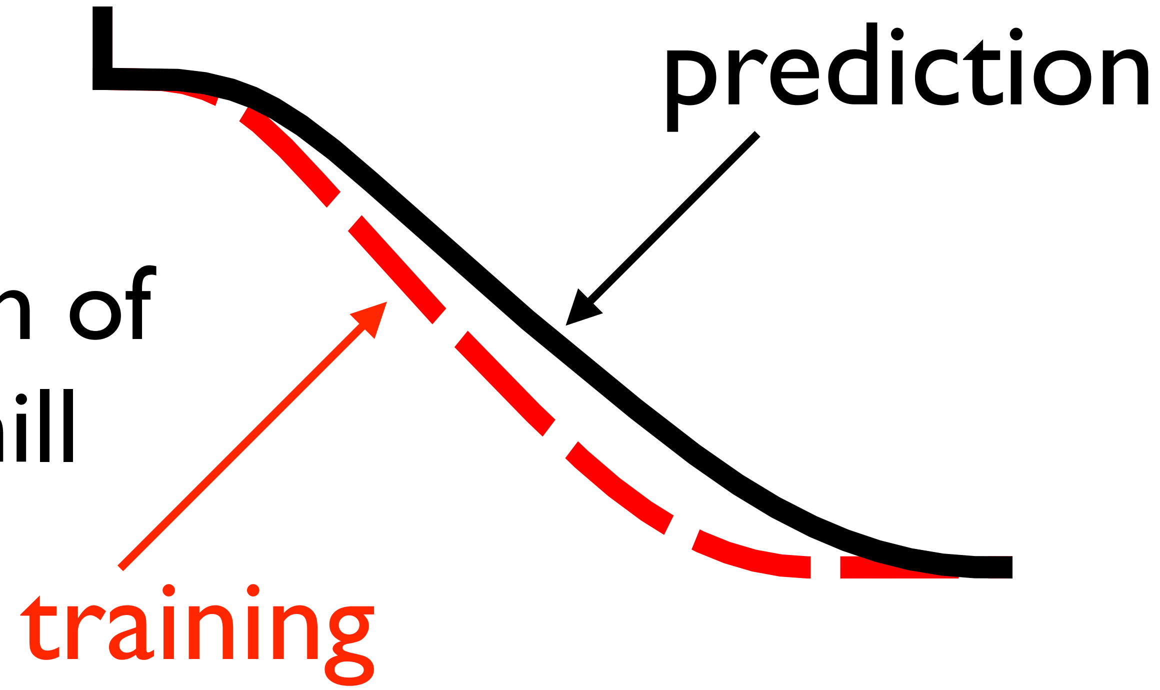
Turbulence Database In the Age of Data-Driven Modeling

- Wanted: DNS, LES, or experimental data on flows with parameterized configurations (geometry, Re, Ma, AoA).
- We need mean velocities & Reynolds stress fields, possibly at sparse yet representative locations



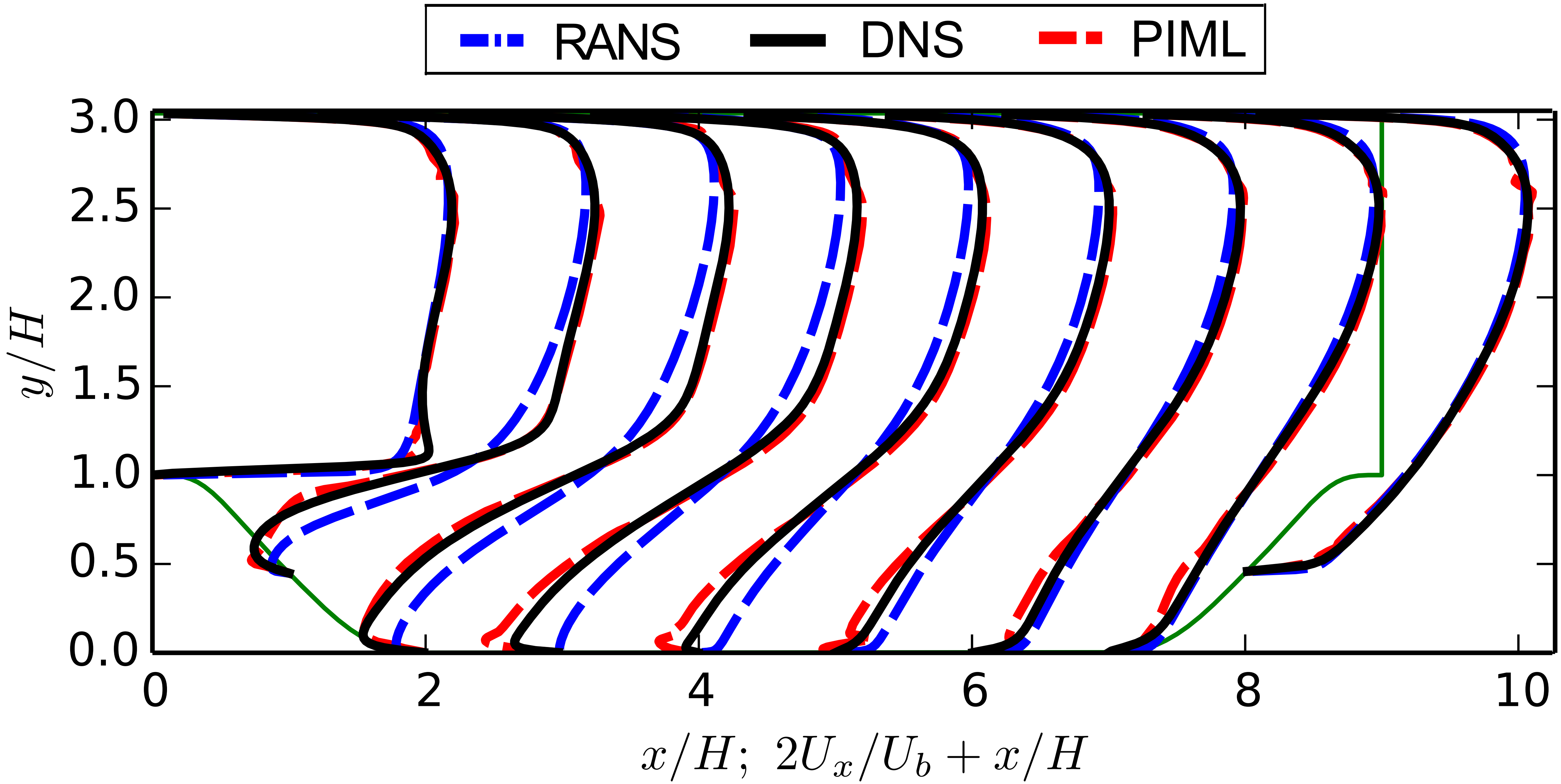
A Less Ambitious Endeavor: Training & Prediction Flows Are Very Similar





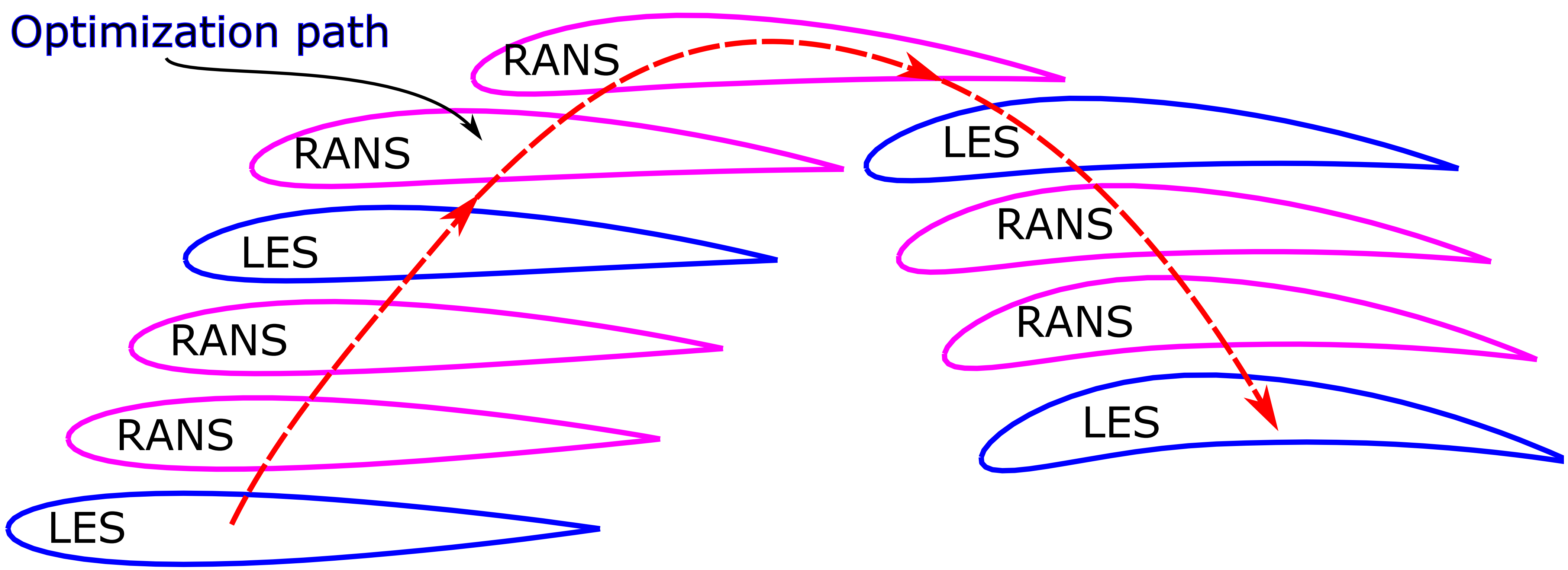
80% the width of the baseline hill

Learning Both Reynolds Stress & Eddy Viscosity



Mean Velocity Prediction

(Realistic) Vision in RANS-based Geometry Optimization

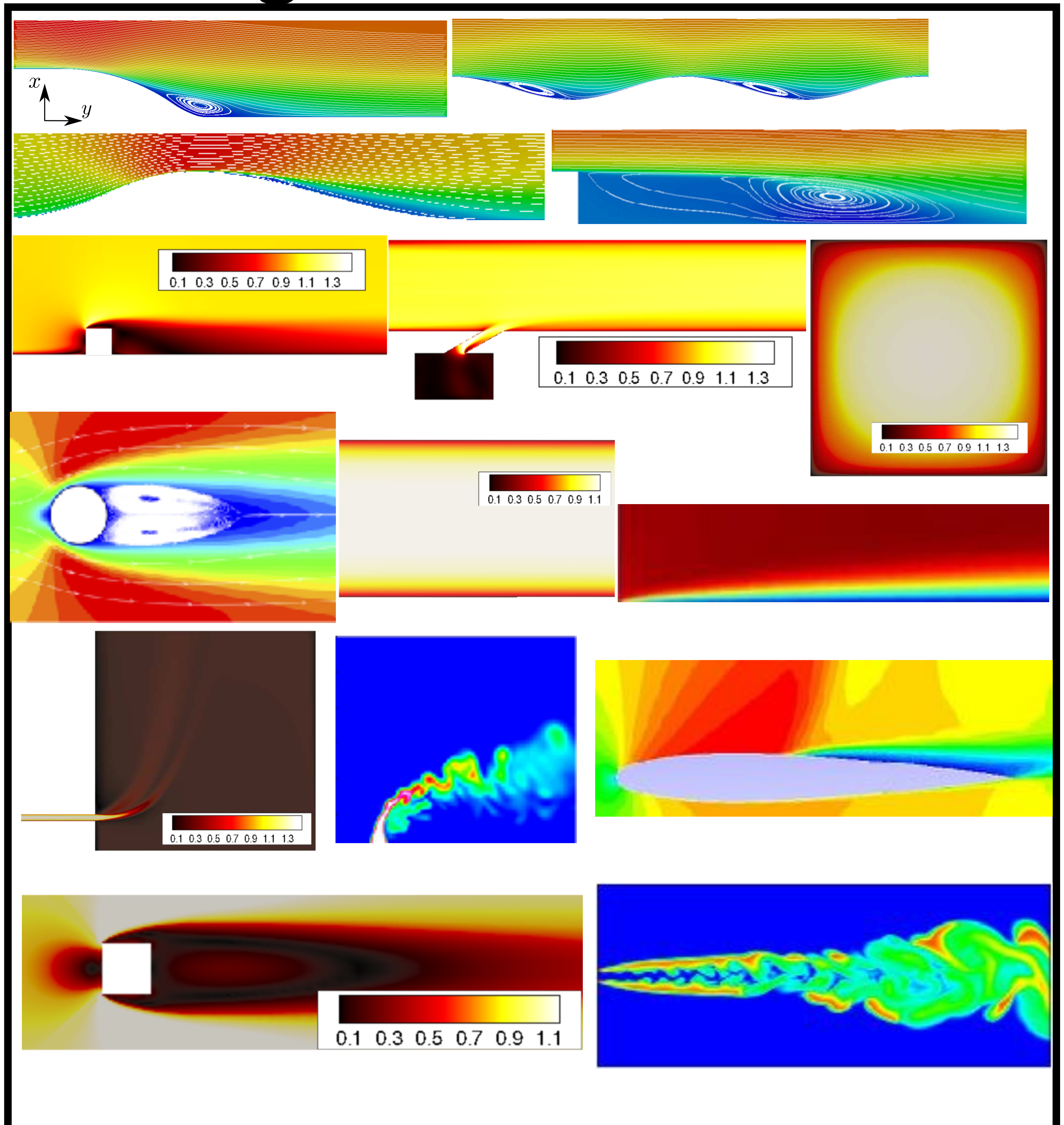


Path of geometry evolution

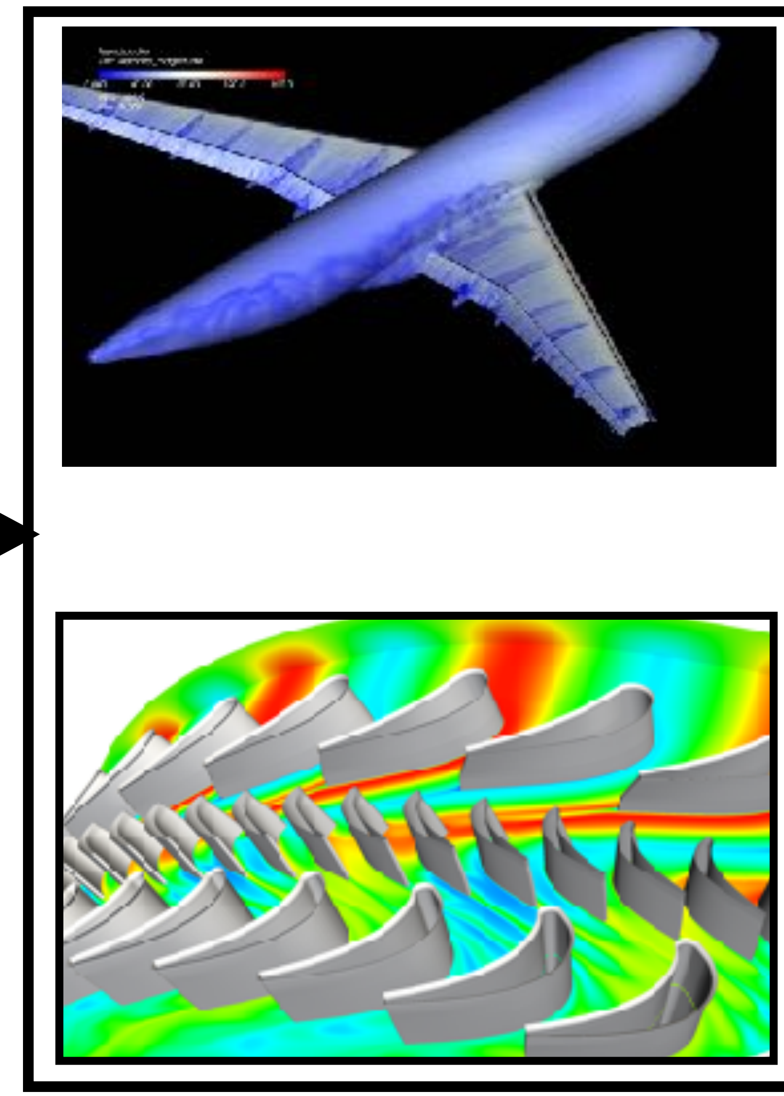
- Proposed a distance metric. (Wu et al. FTaC 2017)
- Typical configurations: flow over bumps, airfoils, wing-body junctures, blade tip clearance.

("Fantasy") Vision: Leverage Data from Elementary Flows to Predict Complex Flows

Training: data from elementary flows

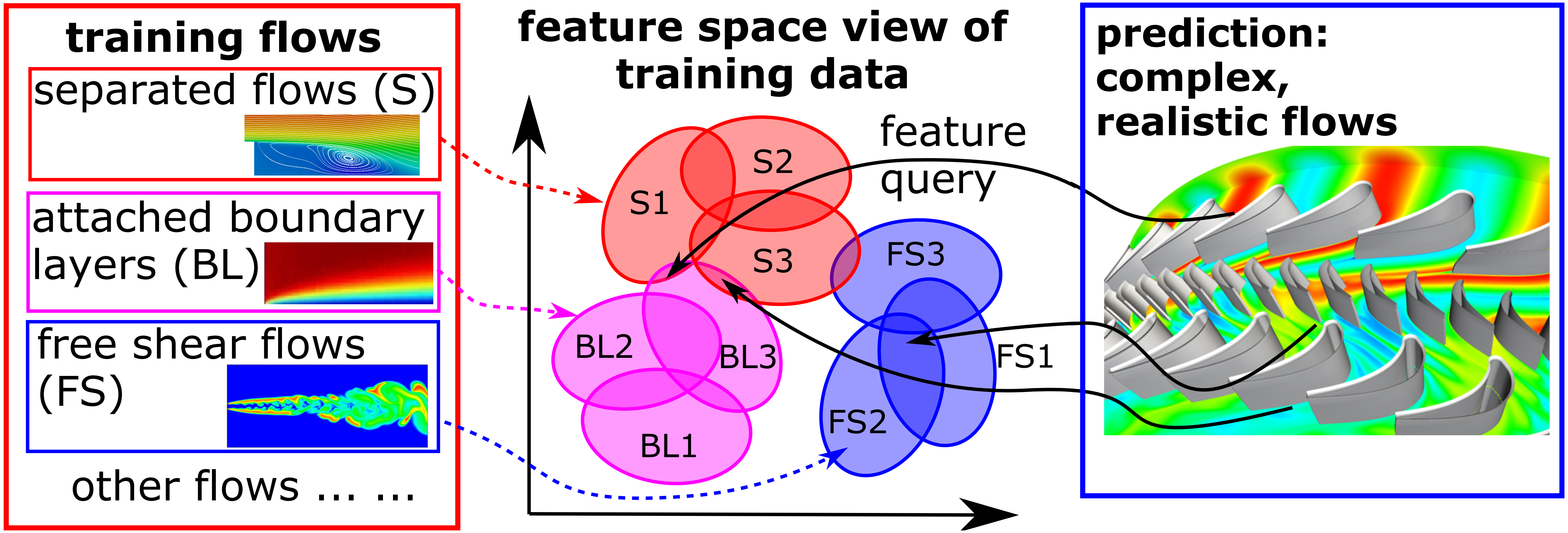


Prediction: Industrial flows



Some figures adopted from Ling et al. POF 2015; www.turbostream-cfd.com; youtube.com

Feature Space View



Addressing Dr. Menter's concerns on ML:

- Data-driven models are constructed as "add-on" (patch) for traditional models, by developers.
- The database and the machine learning are built into the model; not constructed by the users.

Traditional vs. Data-Driven Turbulence Modeling: A Unified Perspective

- Not just buzzword-chasing.
- Machine-learning-assisted turbulence modeling, as we are pursuing, is serious turbulence modeling.
- All constraints in conventional turbulence modeling must be equally respected (see *Spalart 2015: Philosophies and fallacies in turbulence modeling*. Progress in Aerospace Sciences) :
  - Objectivity and frame independence (e.g., can't use velocity or pressure as input)
  - Realizability of Reynolds stress
  - Non-dimensionization and invariance set

Summary and Open Questions

- Proposed a Physics-Informed Machine Learning (PIML) to **correct/improve** existing turbulence models.
- Learn **discrepancies** of RANS modeled Reynolds stresses (with stabilization)!
- Preliminary success in scenarios where training and prediction flows are similar.

Open Questions:

What is the limit of data-driven modeling? How different can the training/predictions flows be?

Is a *(weakly)* universal data-driven turbulence modeling possible or valuable?

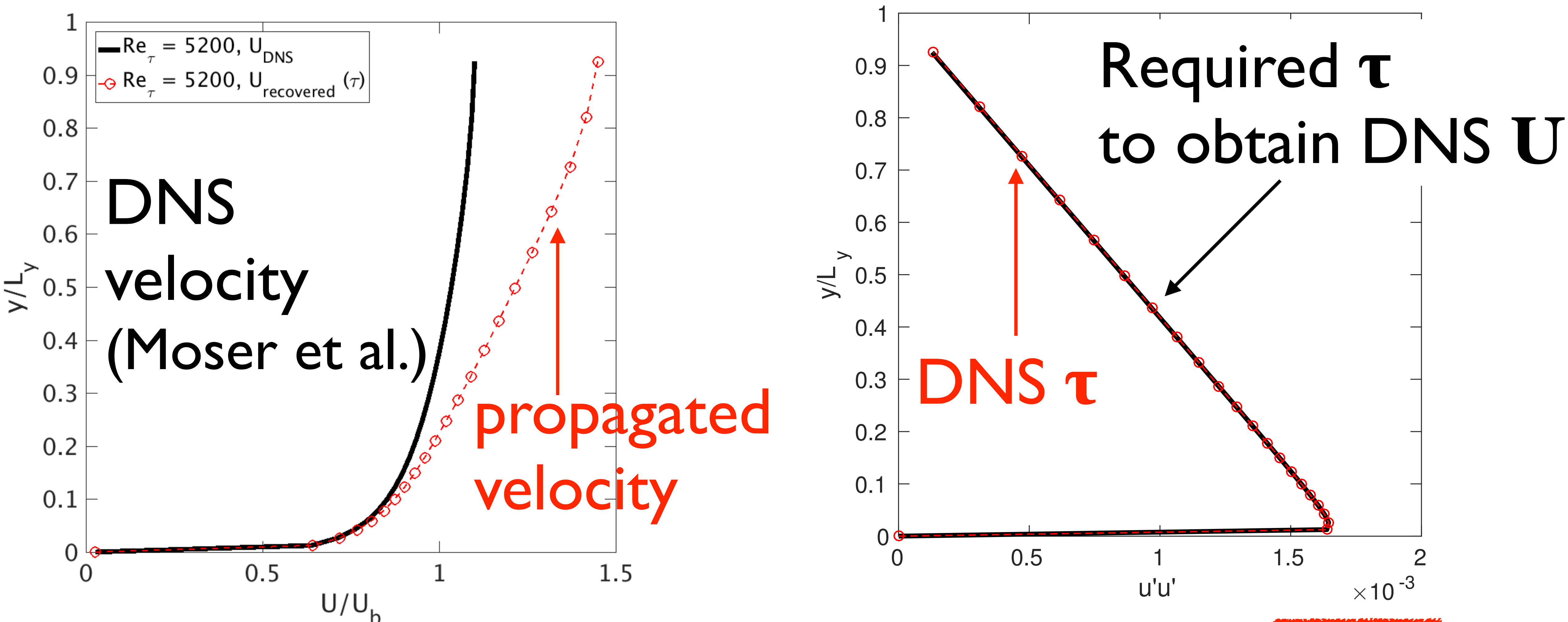
Related Papers

- J.-X. Wang, J.-L. Wu, and H. Xiao. A Physics Informed Machine Learning Approach for Reconstructing Reynolds Stress Modeling Discrepancies Based on DNS Data. Physical Review Fluids, 2(3), 034603, 1-22,2017.
- J.-L. Wu, J.-X. Wang, H. Xiao, J. Ling. A Priori Assessment of Prediction Confidence in Data-Driven Turbulence Modeling. Flow, Turbulence and Combustion, 99(1), 25-46, 2017.
- J.-L. Wu, R. Sun, H. Xiao, Q. Wang. On the conditioning of turbulence models. In preparation.

https://sites.google.com/a/vt.edu/hengxiao/

Thank you!

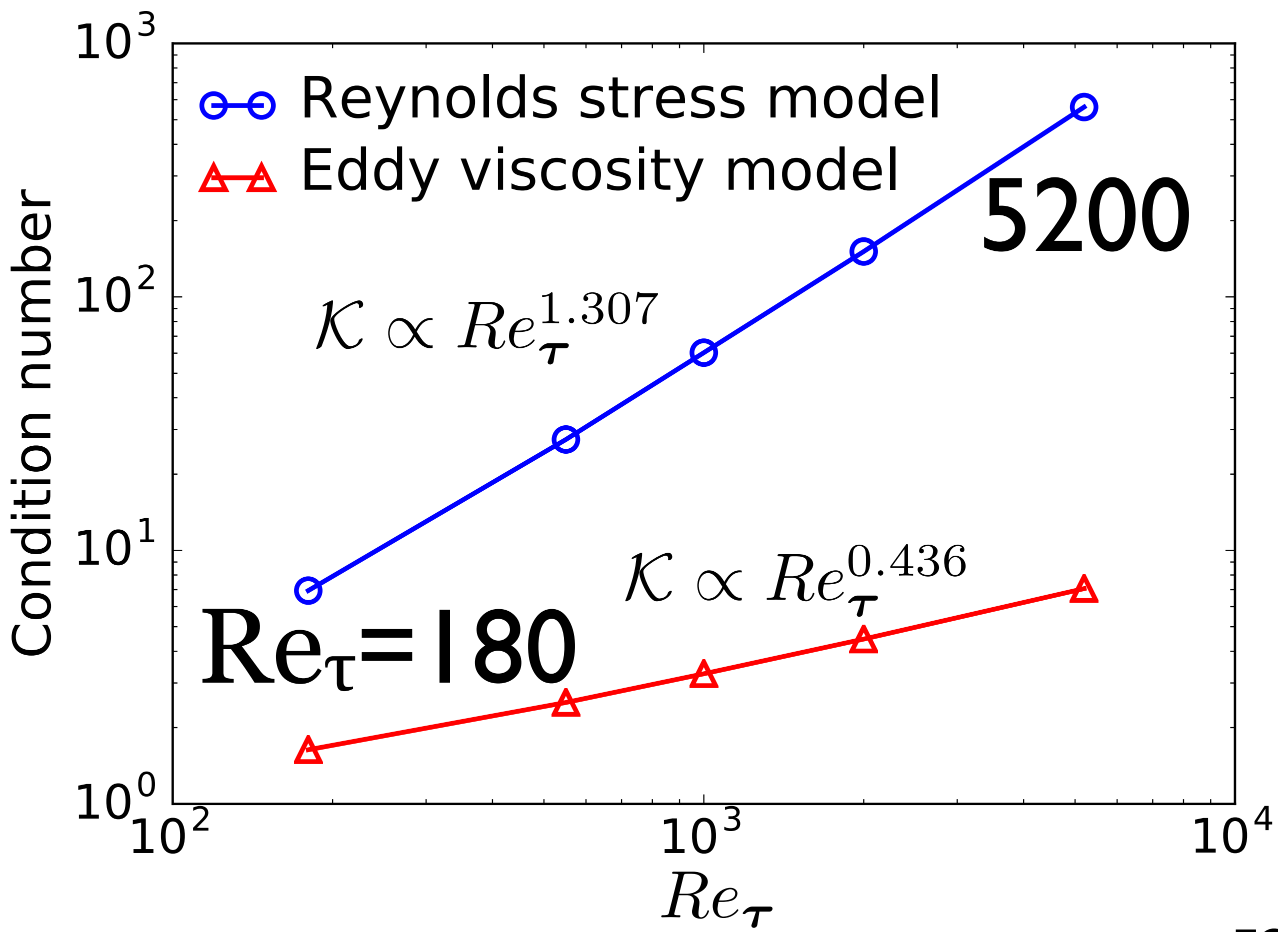
Propagating DNS Reynolds Stresses to Velocities



Frictional Reynolds number (Re_τ)	180	550	1000	2000	5200
Errors in Reynolds stresses, *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, *averaged*	0.25%	1.61%	0.17%	2.85%	**21.6%**
maximum	0.36%	2.70%	0.25%	5.48%	**35.1%**

(Thompson et al. 2016 C&F; Poroseva et. al. POF 2017; Wu et al. Under preparation.)

Condition Numbers for Channel Flows at Re_τ = 180 to 5200



$$\frac{|\delta \mathbf{U}_j|}{U_\infty} \leq \underbrace{\frac{\|\mathbf{r}_j\| \, \|\nabla \cdot \boldsymbol{\tau}\|}{U_\infty}}_{[\mathcal{K}]_j} \frac{\|\nabla \cdot \delta\boldsymbol{\tau}\|}{\|\nabla \cdot \boldsymbol{\tau}\|}$$

Error in mean velocity — Condition number — Error in Reynolds Stresses

$\mathbf{r}_j$ is the j[th] row of matrix inv(A) in discretized RANS equation:

$$\mathrm{A}\,\mathbf{U} = [\mathbf{b}]$$

(Wu et al. Under preparation)

Derivation of Local Condition Number

$$\frac{|\delta \mathbf{U}_j|}{U_\infty} \leq \frac{\|\mathbf{r}_j\| \, \|\delta \mathbf{b}\|}{U_\infty} = \frac{\|\mathbf{r}_j\| \, \|\nabla \cdot \delta \boldsymbol{\tau}\|}{U_\infty} = \underbrace{\frac{\|\mathbf{r}_j\| \, \|\nabla \cdot \boldsymbol{\tau}\|}{U_\infty}}_{[\mathcal{K}]_j} \frac{\|\nabla \cdot \delta \boldsymbol{\tau}\|}{\|\nabla \cdot \boldsymbol{\tau}\|}.$$

Non-dimensionalization of features

Normalized raw input $\hat{\alpha}$	description	raw input α	normalization factor β
$\hat{\mathbf{S}}$	strain rate tensor	$\mathbf{S}$	$\frac{\varepsilon}{k}$
$\hat{\mathbf{\Omega}}$	rotation rate tensor	$\mathbf{\Omega}$	$\|\mathbf{\Omega}\|$
$\widehat{\nabla p}$	Pressure gradient	∇p	$\rho\|\mathbf{U} \cdot \nabla\mathbf{U}\|$
$\widehat{\nabla k}$	Gradient of TKE	∇k	$\frac{\varepsilon}{\sqrt{k}}$

(Wu and Xiao, In preparation)

Feature (q_β)	Description	Raw feature ($\hat{q}_\beta$)	Normalization factor (q^*_β)
q_1	Ratio of excess rotation rate to strain rate (Q criterion)	$\frac{1}{2}(\|\boldsymbol{\Omega}\|^2 - \|\mathbf{S}\|^2)$	$\|\mathbf{S}\|^2$
q_2	Turbulence intensity	k	$\frac{1}{2}U_i U_i$
q_3	Wall-distance based Reynolds number	$\min\left(\frac{\sqrt{k}d}{50\nu}, 2\right)$	not applicable[a]
q_4	Pressure gradient along streamline	$U_k \frac{\partial P}{\partial x_k}$	$\sqrt{\frac{\partial P}{\partial x_j}\frac{\partial P}{\partial x_j} U_i U_i}$
q_5	Ratio of turbulent time scale to mean strain time scale	$\frac{k}{\varepsilon}$	$\frac{1}{\|\mathbf{S}\|}$
q_6	Cratio of pressure normal stresses to shear stresses	$\sqrt{\frac{\partial P}{\partial x_i}\frac{\partial P}{\partial x_i}}$	$\frac{1}{2}\rho \frac{\partial U_k^2}{\partial x_k}$
q_7	Nonorthogonality between velocity and its gradient [28]	$\left\lvert U_i U_j \frac{\partial U_i}{\partial x_j}\right\rvert$	$\sqrt{U_l U_l \, U_i \frac{\partial U_i}{\partial x_j} U_k \frac{\partial U_k}{\partial x_j}}$
q_8	Ratio of convection to production of TKE	$U_i \frac{dk}{dx_i}$	$\lvert \overline{u'_j u'_k} S_{jk} \rvert$
q_9	Ratio of total to normal Reynolds stresses	$\|\overline{u'_i u'_j}\|$	k
q_{10}	Streamline curvature	$\left\lvert \frac{D\boldsymbol{\Gamma}}{Ds} \right\rvert$ where $\boldsymbol{\Gamma} \equiv \mathbf{U}/\lvert\mathbf{U}\rvert$, $Ds = \lvert\mathbf{U}\rvert Dt$	$\frac{1}{L_c}$

(Wang, Wu, Xiao, PRF 2017)

Test Case 3: Flat Plate Boundary Layer

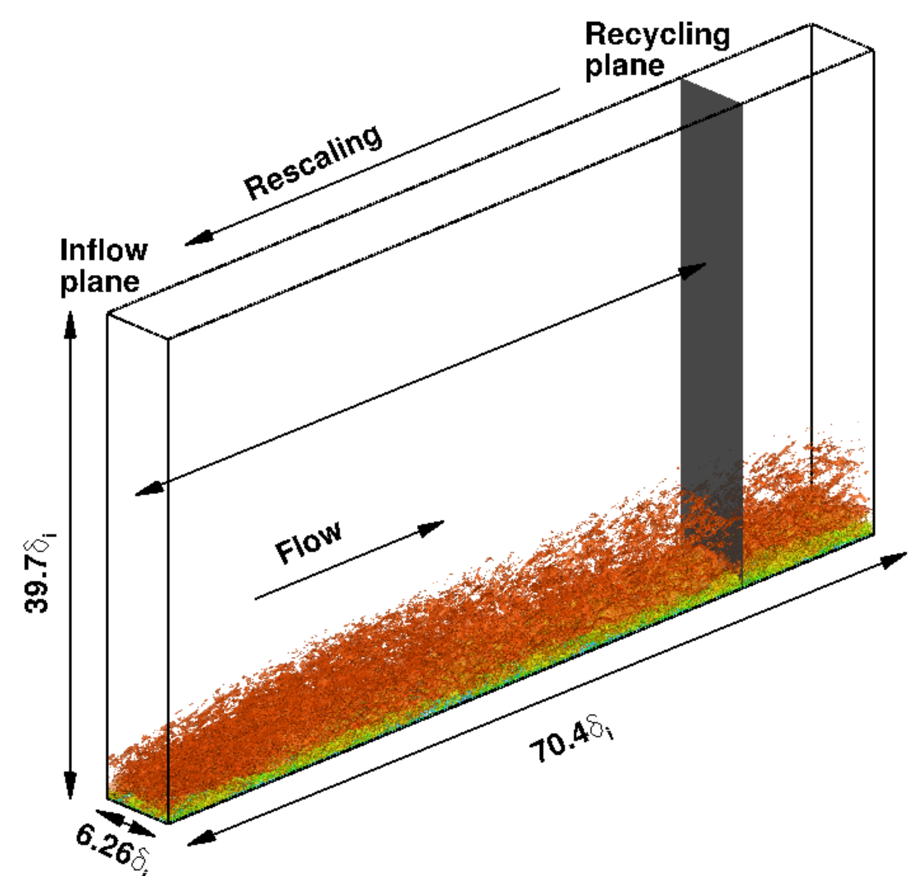
- Flow to be predicted: Ma=8, Tw=0.53
- Flows in the database:

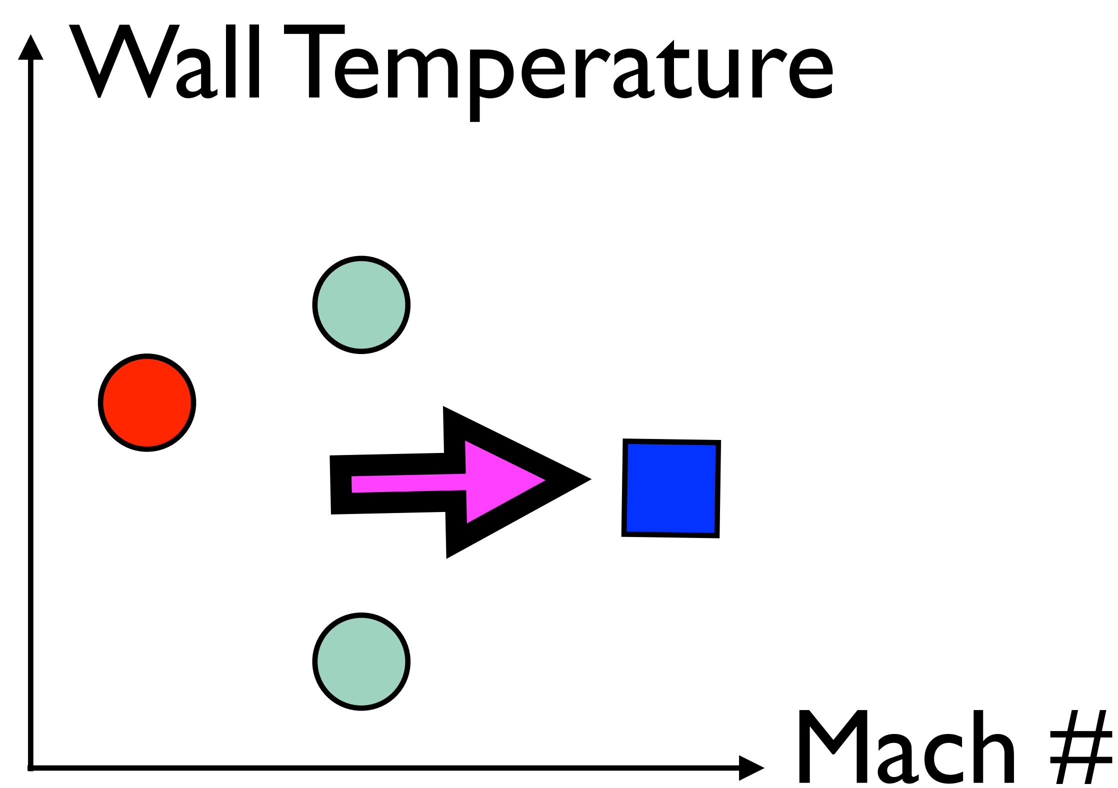
  Ma=6.0, Tw=0.25 [cold wall]
  Ma=2.5, Tw=1.0
  Ma=6.0, Tw=0.76

Only the Ma=2.5 case is used for training

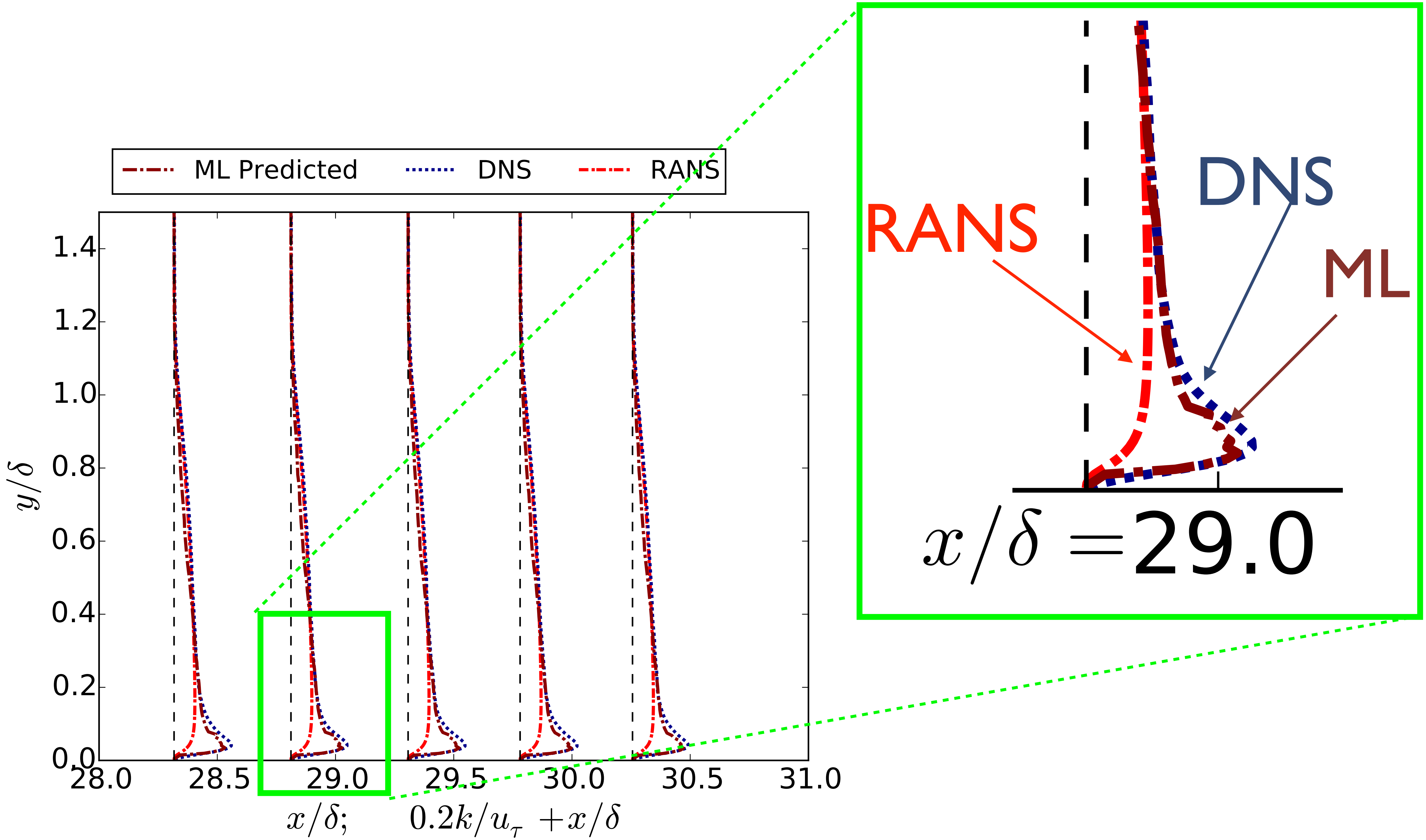
Wall temperature Tw normalized by recovery temperature Tr:

$$T_r = T_\infty (1 + 0.9 * \frac{\gamma - 1}{2} M_\infty^2)$$

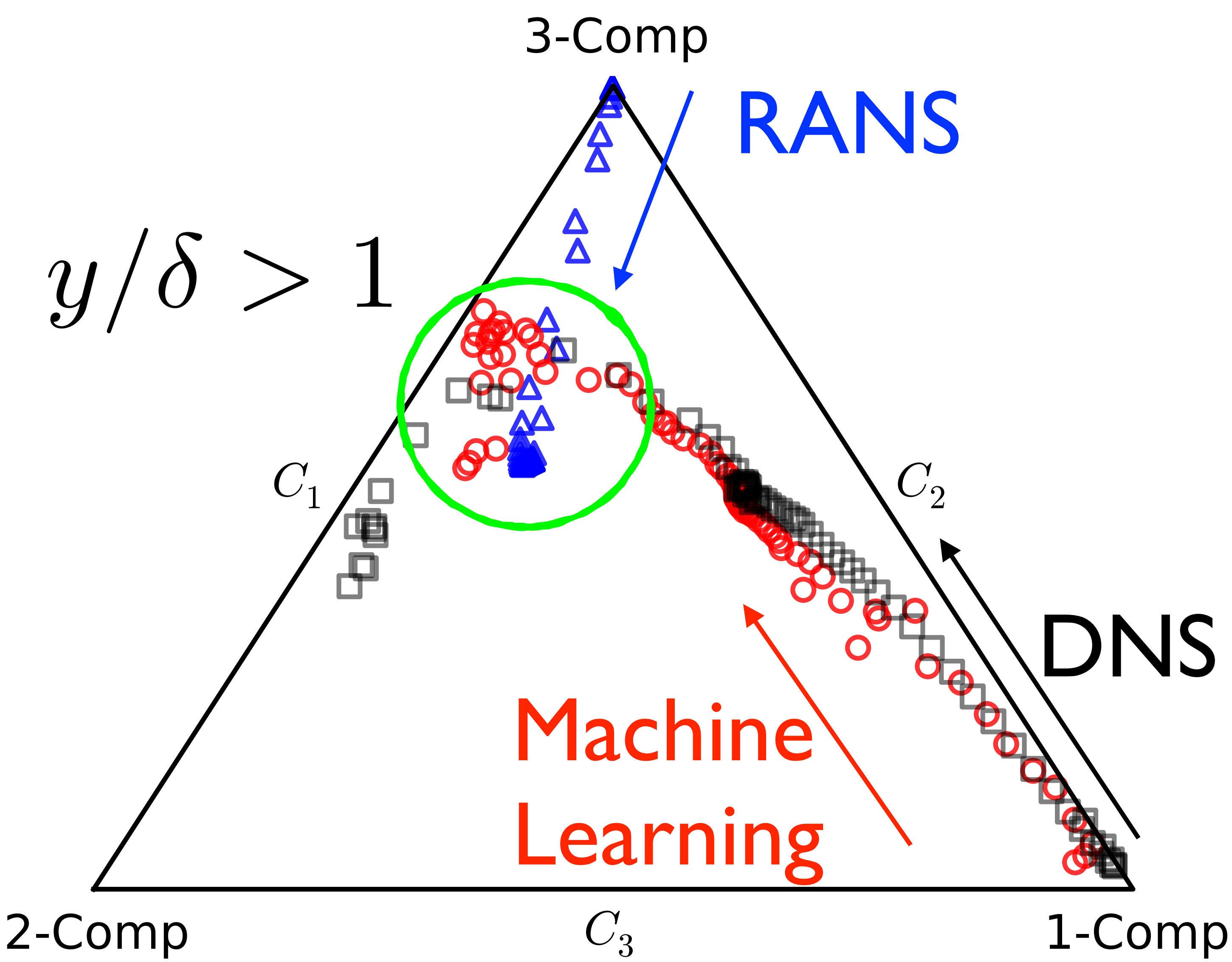


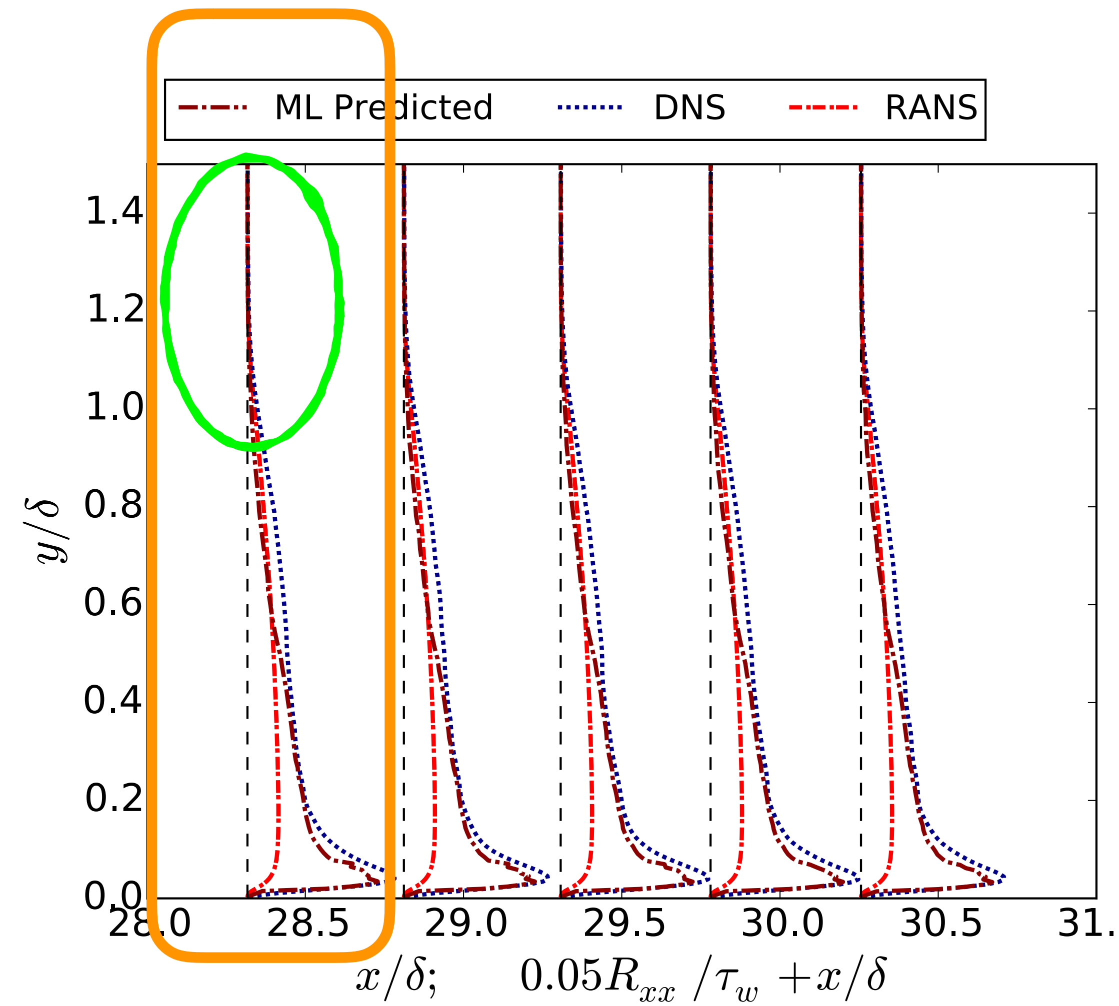


Turbulent Kinetic Energy



Realizability Map

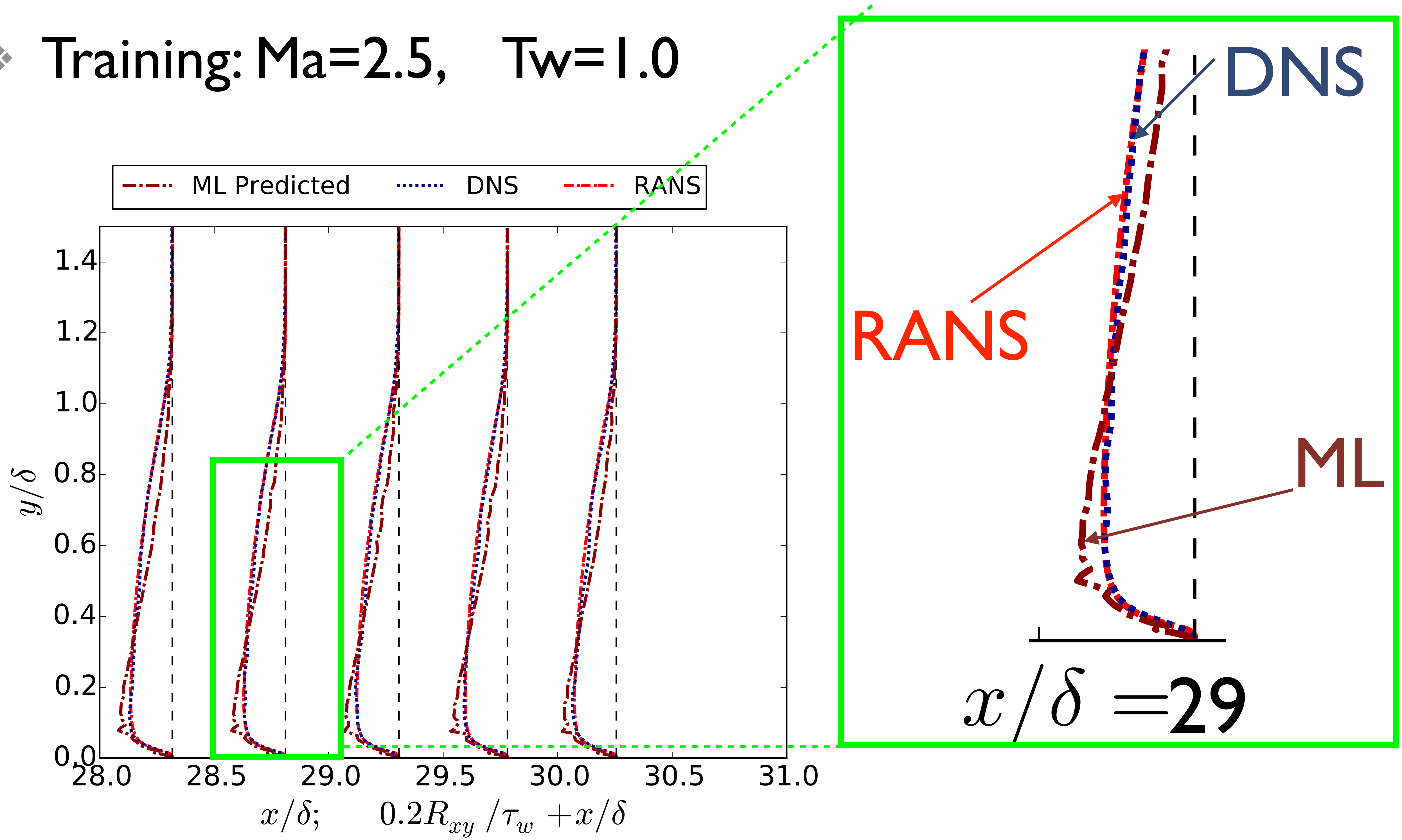




- Outside the boundary layer, the Reynolds stress anisotropy does not have physical significance.

Turbulent Shear Stress

- Training: Ma=2.5, Tw=1.0



Prediction: Ma=8, Tw=0.53