Development of A Deep Learning Turbulence Model

Feiyu Yang, Tyler Strickland and Prof. Sage Kokjohn

Funding Sponsor – China Scholarship Council

Objectives

Couple OpenFOAM and machine learning to improve RANS and LES turbulence modeling.

Governing equations

The averaged N-S equation:

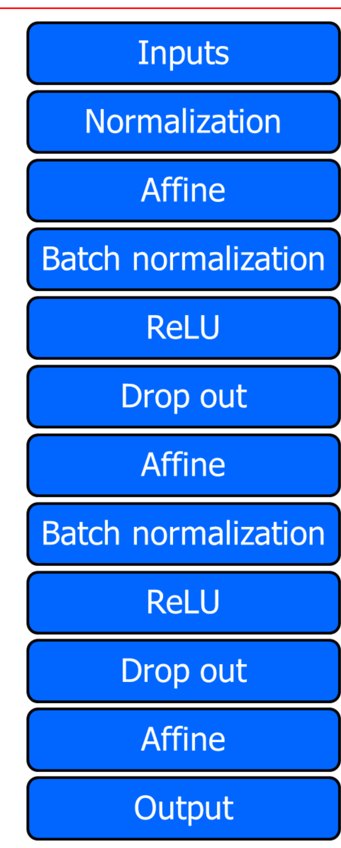
$$\frac{\partial \overline{u_i}}{\partial t}+\frac{\partial \overline{u_i}\,\overline{u_j}}{\partial x_j}=-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i}+\nu_{eff}\frac{\partial^2 \overline{u_i}}{\partial x_i \partial x_j}-\frac{\partial \tau^R}{\partial x_j}$$

Using machine learning method to determine turbulence viscosity:

$$\nu_{t,LES}=\frac{-\overline{u_i'u_j'}S_{ij}+\frac{2}{3}k\delta_{ij}S_{ij}}{2S_{kl}S_{kl}}$$

Deep learning model architecture

- Inputs: features extracted from high-fidelity simulation
- Affine layer: application of weight and bias matrix
- Batch normalization layer: improves gradient flow and allows higher learning rate.
- ReLU layer: filters non-physical values
- Drop out layer: avoids over-fitting.



Feature extracting

Features are extracted from high-fidelity LES using:

#	Description	Formula	#	Description	Formula
F1	Distance to the nearest wall	$\frac{d}{d_{characteristic}}$	F8	Ratio of convection to production of k	$\frac{U_i\frac{\partial k}{\partial x_i}}{\left\lvert\overline{u_j'u_l'}S_{jl}\right\rvert+U_l\frac{\partial k}{\partial x_l}}$
F2	Nondimensionalized Q criterion	$\frac{\lVert R\rVert^2-\lVert S\rVert^2}{\lVert R\rVert^2+\lVert S\rVert^2}$	F9	Ratio of total Reynolds stresses to normal Reynolds stresses	$\frac{\lVert\overline{u_i'u_j'}\rVert}{k+\lVert\overline{u_i'u_j'}\rVert}$
F3	Turbulence intensity	$\frac{k}{0.5U_iU_i+k}$	F10-12	Strain rate tensor	$\frac{Sk}{\lVert S\rVert k+\varepsilon}$
F4	Turbulence Reynolds number	$\min(\frac{\sqrt{k}d}{50\nu},2)$	F13	Rotation rate tensor	$\frac{R}{2\lVert R\rVert}$
F5	Pressure gradient along streamline	$\frac{U_k\frac{\partial P}{\partial x_k}}{\sqrt{\frac{\partial P}{\partial x_j}\frac{\partial P}{\partial x_j}U_iU_i}+\left\lvert U_l\frac{\partial P}{\partial x_l}\right\rvert}$	F14-15	Pressure gradient vector	$\frac{\nabla p}{\lvert\nabla p\rvert+\rho\lvert DU/Dt\rvert}$
F6	Ratio of turbulent time scale to mean strain time scale	$\frac{\lVert S\rVert k}{\lVert S\rVert k+\varepsilon}$	F16-17	Pressure gradient vector	$\frac{\nabla k\sqrt{k}}{\lvert\nabla k\rvert\sqrt{k}+\varepsilon}$
F7	Ratio of pressure normal stresses to normal shear stresses	$\frac{\sqrt{\frac{\partial P}{\partial x_i}\frac{\partial P}{\partial x_i}}}{\sqrt{\frac{\partial P}{\partial x_j}\frac{\partial P}{\partial x_j}}+0.5\rho\frac{\partial U_k^2}{\partial x_k}}$	F18	Turbulence intensity	$\frac{k}{k+\nu\lVert S\rVert}$

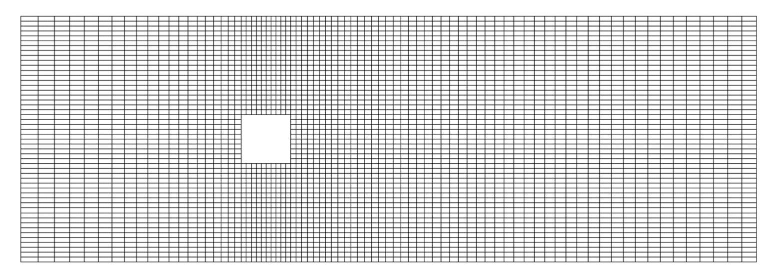
Simulation setup

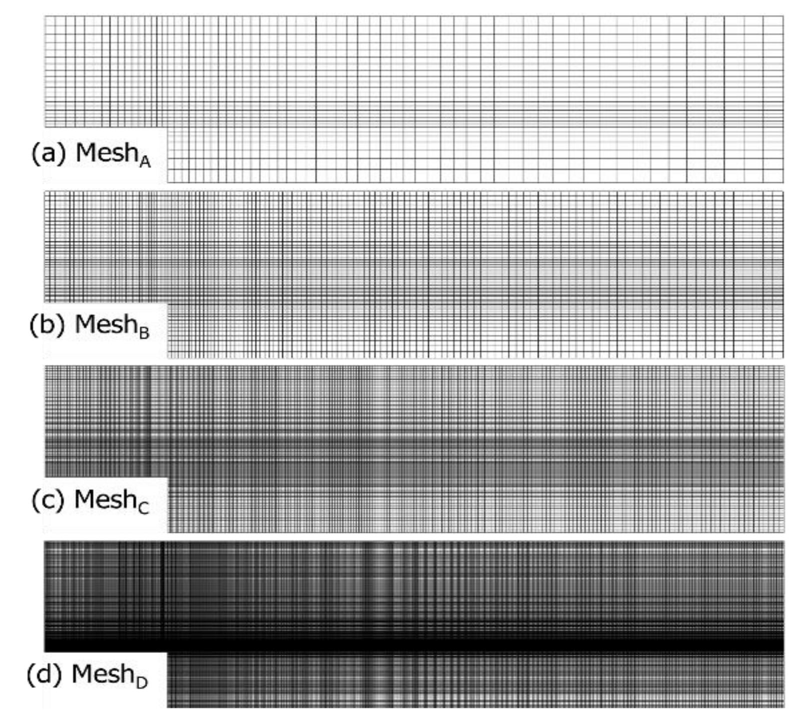
Backward facing flow

The $Mesh_B$ with 6820 cells is regard as base mesh and $Mesh_A$, $Mesh_C$ and $Mesh_D$ can be obtained by refine $Mesh_B$ by a factor of 0.25, 4 and 16, respectively. The $Mesh_A$ with 1670 cells is highly coarse on which the LES simulation is invalid due to the violation of isotropy. This mesh will be applied to test the performance of the machine learning turbulence model where the LES simulation fails. The intermediate $Mesh_B$ and $Mesh_C$ will be applied to test the machine learning turbulence model performance on relative coarse mesh where LES simulation is valid. And last, the $Mesh_D$ with 109120 cells is highly fine, its high-fidelity simulation result is regard as the golden criterion and used to train the machine learning model.

Square cylinder flow

The square cylinder flow is applied to test the model performance on "unfamiliar" flow





Flow parameters

Parameters for backward facing step flow and square cylinder flow:

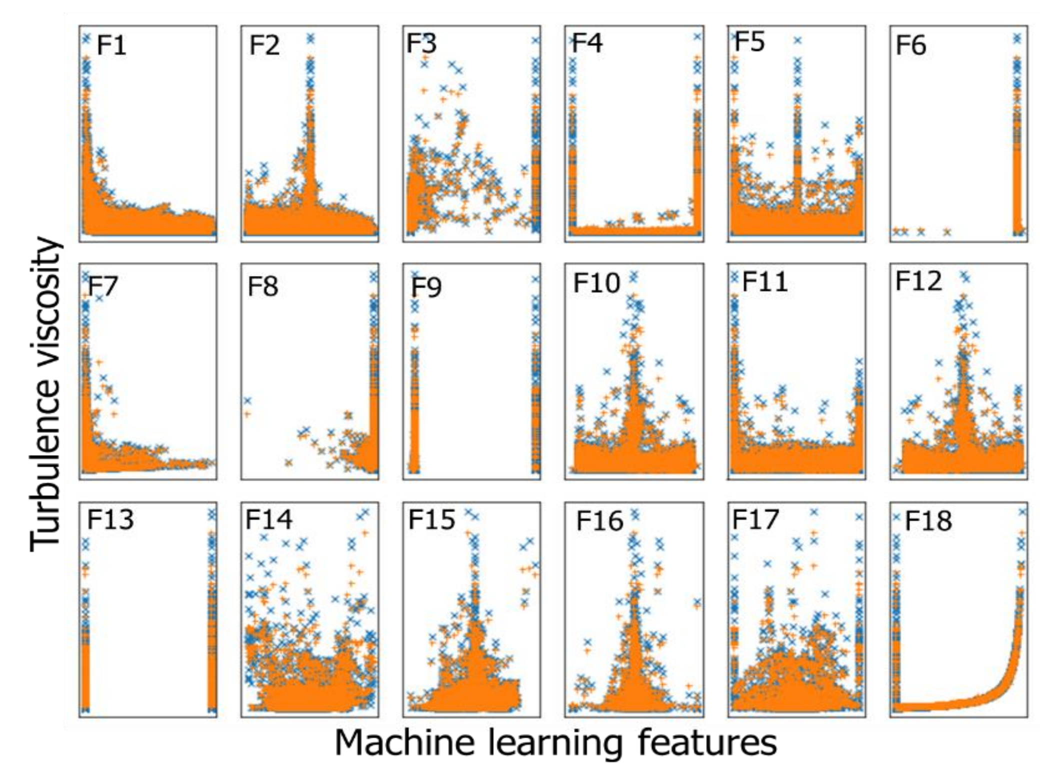
Parameters	Backward facing flow	Square cylinder flow
Reynolds number	4.4e+4	1e+6
Characteristic length	0.06 m	0.07 m
Turbulence intensity	5%	5%
Molecular viscosity	1e-5	1e-5

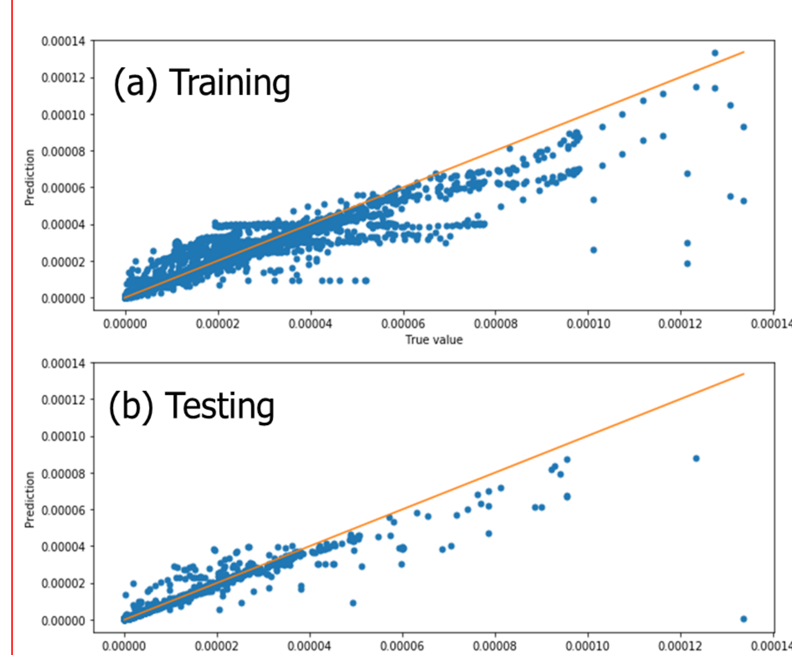
Model training

The machine learning model was trained on a single snapshot in time using the fine grid LES (Mesh D)

Feature analysis

The machine learning model reproduces key features of the simulations. The area of the symbols can be interpreted as the contribution of the feature → the larger the area, the more information it contributes.

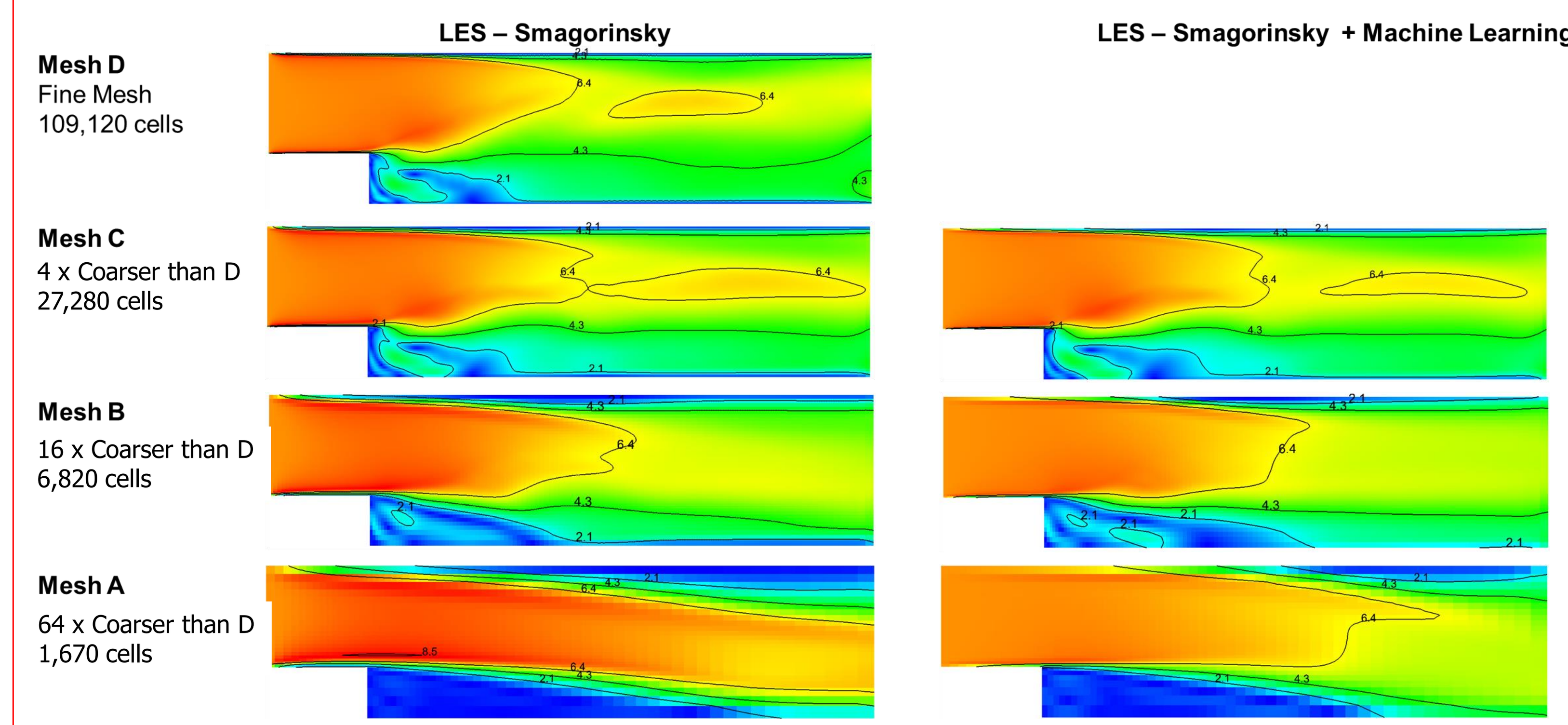




Model errors

Despite some outliers, most points are near the criterion line. Note that there are tens of thousands of points in the dense region. The test and training datasets shows similar trends and accuracy.

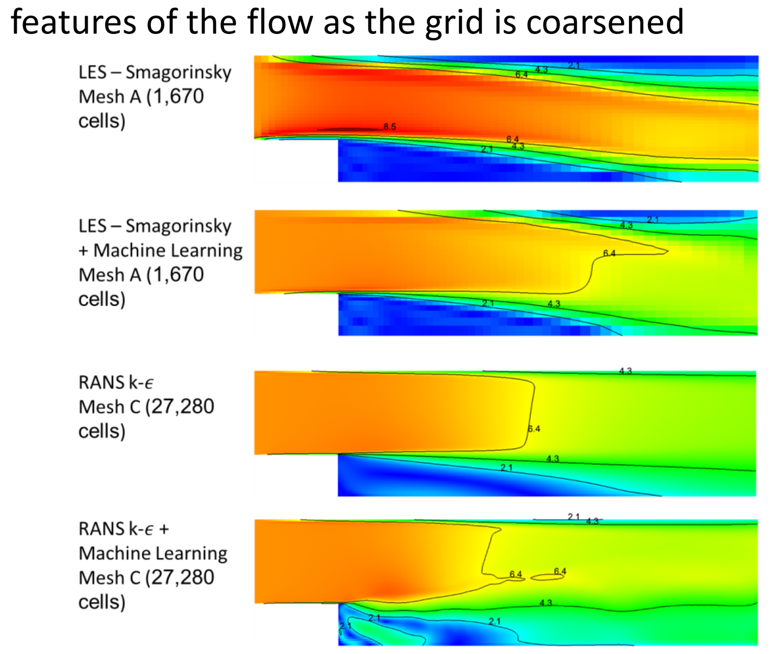
Results: Backward Facing Step



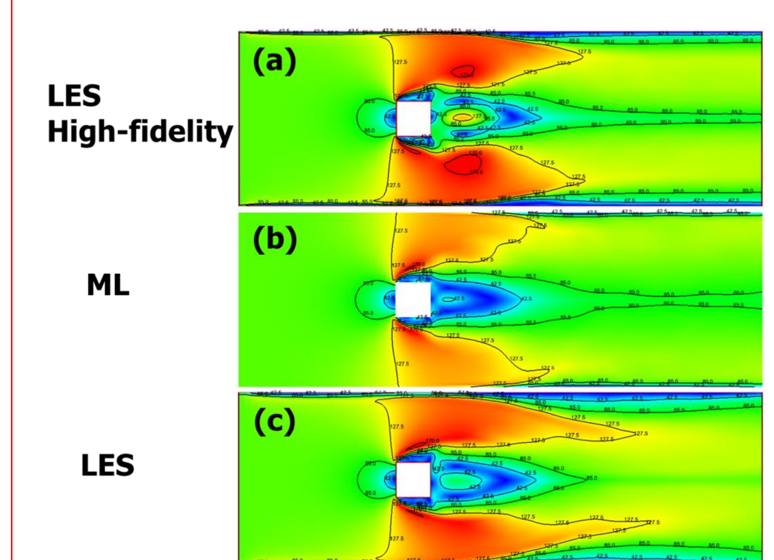
In the geometry used for training, the LES and LES + Machine Learning models produce similar results; however, the machine learning model retains more features of the flow as the grid is coarsened

On the very course grid (Mesh A), the LES solution deviates substantially from both the fine grid LES and RANS solutions. The course grid LES with Machine Learning produces similar results to the RANS case.

The finer grid RANS solution with Machine Learning captures features present in the fine grid LES solution



Results: Square Cylinder



The same Machine Learning model used above (i.e., trained on a single snapshot of the data from a backward facing step) is applied to a different flow.

The ML model is able to capture the "tail" and the area of the "wings" more accurately than the base LES model when the grid is coarsened.

Enough scales are available in a single LES snapshot to provide a general fit for different geometries

Conclusions

- The machine learning model reduces the mesh sensitivity of the LES predictions → results revert to a RANS-like solution on a course grid
- Training the machine learning model on a single snapshot of the backward facing step enables prediction in different geometries