Modeling subgrid-scale dynamics using machine learning

Melody meeting

Hugo Frezat with Guillaume Balarac[1] Julien Le Sommer[2] Ronan Fablet[3] Redouane Lguensat[4]

October 14, 2020

[1]LEGI
[2]IGE
[3]IMT Atlantique
[4]Laboratoire des Sciences du Climat et de l'Environnement

Subgrid-scale modeling

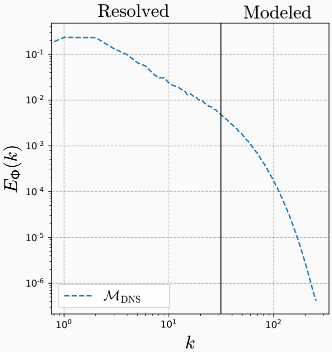

Figure 1: Resolved and modeled scales in spectral space in LES.

- DNS is still impossible for realistic applications.
- LES models being used in climate science, oceanography.
- Challenges remaining with algebraic models (not stable, large error over time, etc).

NS - SGS scalar transport

In an incompressible Navier-Stokes framework,

$$\frac{\partial \mathrm{u}}{\partial t} + (\mathrm{u} \cdot \nabla)\mathrm{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \mathrm{u}, \qquad \nabla \cdot \mathrm{u} = 0, \tag{1}$$

transport of a scalar quantity Φ

$$\frac{\partial \Phi}{\partial t} + (\mathrm{u} \cdot \nabla)\Phi = \nabla \cdot (\kappa \nabla \Phi) \tag{2}$$

$$\Downarrow \text{ filtered s.t. } \overline{\Phi(x)} = \int_V \Phi(x_f)\, G(x_f - x)\, dx_f \tag{3}$$

$$\frac{\partial \overline{\Phi}}{\partial t} + (\overline{\mathrm{u}} \cdot \nabla)\overline{\Phi} = \nabla \cdot (\kappa \nabla \overline{\Phi}) + \nabla \cdot \underbrace{(\overline{\mathrm{u}\,\Phi} - \overline{\mathrm{u}}\,\overline{\Phi})}_{\text{SGS residual flux s}} \tag{4}$$

Dataset generation

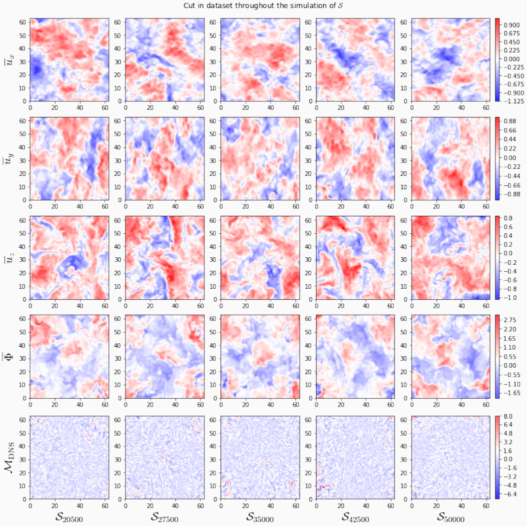

Figure 2: Data from DNS. (available on Zenodo)

- Ω: 512^3
- T: $[0, 5e^4], \Delta\mathcal{S} = 500$
- $Re_\lambda = 160$
- $Sc = 0.7$

State of the art: machine learning

Topic	Model	Input	Output
Specific app.			
Reaction rates (Lapeyre et al. 2019 [2])	Compressible	-	-
Ocean dynamics (Bolton et al. 2019 [1])	QG	-	-
Spectral			
3D HIT (Vollant et al. 2017 [6])	Incompressible	Noll's	$\nabla \cdot s$
3D HIT (Portwood et al. 2020 [4])	Incompressible	Gradients	s

Table 1: Some applications of SGS modeling in specific topics where inputs and outputs are application dependent and idealized scenarios such as HIT with spectral solvers.

Transformation-invariant NN framework

Physical constraints

We define our model such that

$$\mathcal{M}_{\mathrm{NN}}(\mathsf{i}) \approx \nabla \cdot \mathsf{s}, \qquad \mathsf{i} = \left\{\overline{\mathsf{u}}, \overline{\Phi}\right\} \tag{5}$$

Following classical physical invariances [3, 5] and mathematical equalities, we chose to embed the following:

- Translation invariance
- Scalar linearity
- Galilean invariance
- Permutation (rotation) invariance

Architecture

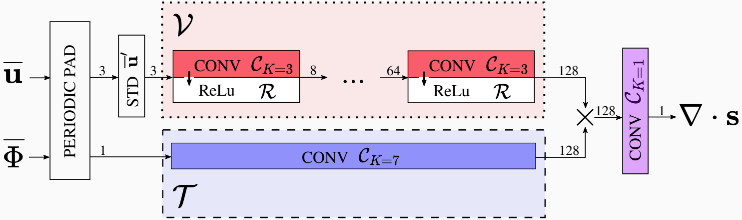

Figure 3: Illustration of the sub-grid transport neural network architecture (SGTNN).

Translation invariance

$$\forall \delta \in \mathbb{R}, \qquad \mathcal{M}_{\mathrm{NN}}(T_\delta \overline{\mathrm{u}}, T_\delta \overline{\Phi}) = T_\delta \mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}}, \overline{\Phi}) \tag{6}$$

Architecture

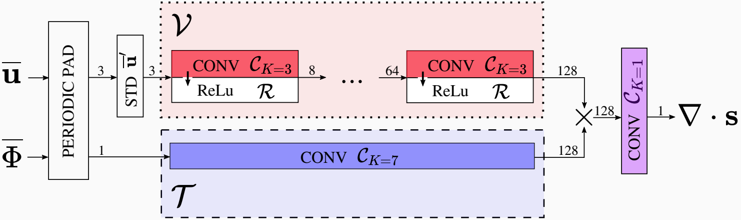

Figure 4: Illustration of the sub-grid transport neural network architecture (SGTNN).

Scalar linearity

$$\forall \lambda \in \mathbb{R}, \qquad \mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}}, \lambda\overline{\Phi}) = \lambda \mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}}, \overline{\Phi}) \tag{7}$$

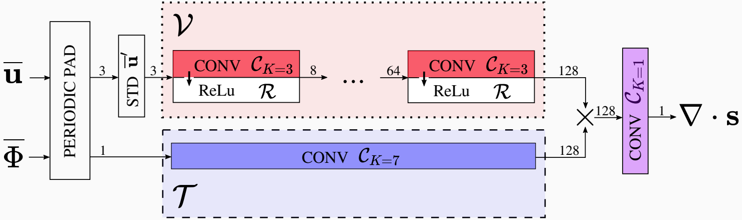

Figure 5: Illustration of the sub-grid transport neural network architecture (SGTNN).

Galilean invariance

$$\forall \beta \in \mathbb{R}, \qquad \mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}} + \beta, \overline{\Phi}) = \mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}}, \overline{\Phi}) \tag{8}$$

Architecture

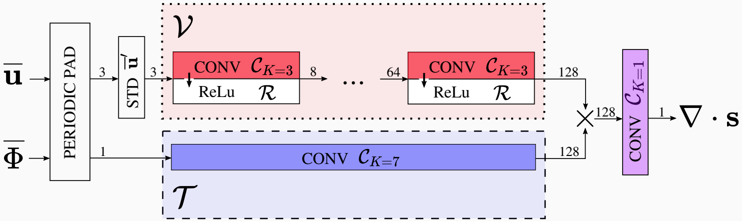

Figure 6: Illustration of the sub-grid transport neural network architecture (SGTNN).

Permutation (rotation) invariance

$$\mathcal{M}_{\mathrm{NN}}(A_{ij}\overline{\mathrm{u}}, \overline{\Phi}) = \mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}}, \overline{\Phi}) \tag{9}$$

Verifying invariances

- Random realizations with λ and β
- 6 permutations

	$\mathcal{M}_{\mathrm{CNN}}$		$\mathcal{M}_{\mathrm{SGTNN}}$	
	E[MSE]	Var[MSE]	E[MSE]	Var[MSE]
$\mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}}, \lambda\overline{\Phi})$	1.7251	**0.0629**	1.4643	**0.0**
$\mathcal{M}_{\mathrm{NN}}(\overline{\mathrm{u}} + \beta, \overline{\Phi})$	1.8307	**0.0373**	1.4643	**0.0**
$\mathcal{M}_{\mathrm{NN}}(A_{ij}\overline{\mathrm{u}}, \overline{\Phi})$	1.6022	$1.9539 \cdot 10^{-6}$	1.4638	$2.5107 \cdot 10^{-7}$

Table 2: Evaluation of the three additional physical constraints provided by $\mathcal{M}_{\mathrm{SGTNN}}$.

Results: testing dataset a priori

	$\downarrow \mathcal{L}_2(X,Y)$	$\uparrow \mathcal{P}(X,Y)$	$\downarrow \mathcal{J}(P_X \| P_Y)$	$\downarrow \mathcal{I}(X,Y)$
$\mathcal{M}_{\mathrm{DynSmag}}$	1.8531	0.3602	0.2840	0.1767
$\mathcal{M}_{\mathrm{DynRG}}$	1.6048	0.4969	0.3471	**0.0624**
$\mathcal{M}_{\mathrm{MLP}}$	3.1645	0.1365	**0.0449**	0.0896
$\mathcal{M}_{\mathrm{CNN}}$	1.5996	0.5597	0.0868	0.1021
$\mathcal{M}_{\mathrm{SGTNN}}$	**1.4643**	**0.6118**	0.1070	0.1190

Table 3: A priori evaluation of the SGS term in developed turbulence regime. Mean-squared-error $\mathcal{L}_2$, Pearson's coefficient $\mathcal{P}$, Jensen-Shannon distance $\mathcal{J}(P_X \| P_Y)$ and integral dissipation error $\mathcal{I}(X,Y)$.

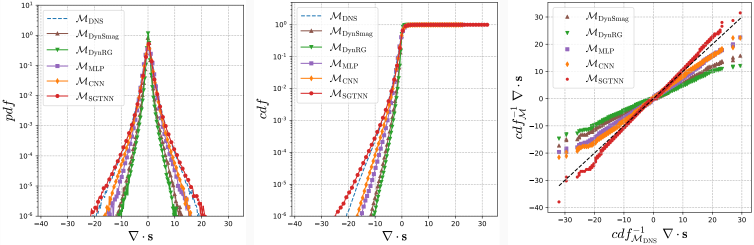

Figure 7: Statistical evaluation with testing data.

Results: decaying scalar a priori

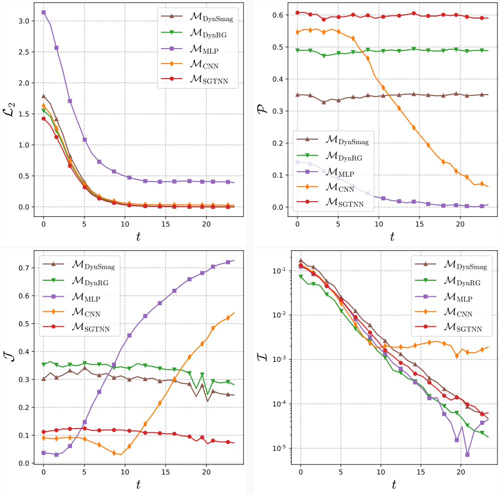

Figure 8: Time evolution of the different metrics in decaying scalar regime.

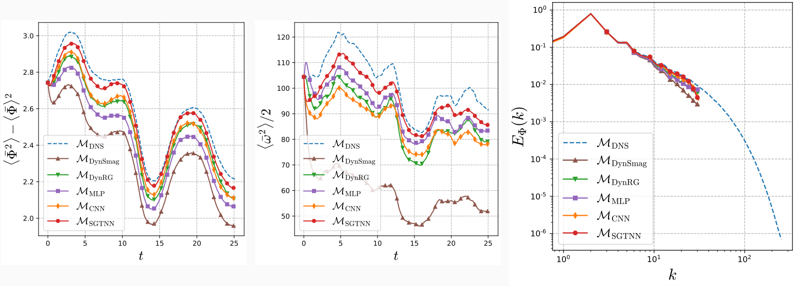

Figure 9: Statistics of simulation in developed turbulence regime. Scalar variance (left), resolved scalar enstrophy (middle), spectrum (right).

Results: decaying scalar a posteriori

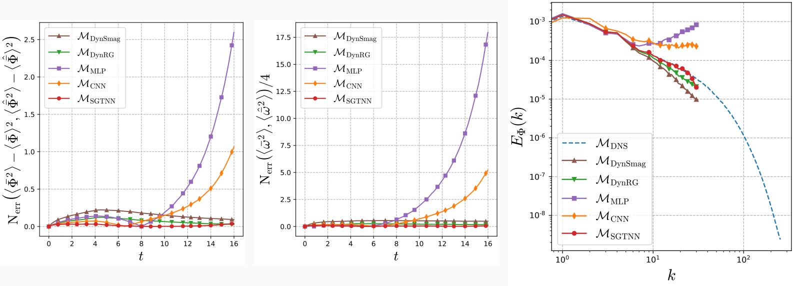

Figure 10: Statistics of simulation in decaying scalar regime. Scalar variance (left), resolved scalar enstrophy (middle), spectrum (right).

Flow learning: ideas

1. Preprint: https://arxiv.org/abs/2010.04663
2. End-to-end learning: access to many more diagnostic variables
3. Effect of training on multiple time-steps (as compared to single one with a dataset)
4. Large filter size, computationally less consuming system: e.g. QG
5. Investigate asymptotics quantities: e.g. stability

Questions?

References i

T. Bolton and L. Zanna.
Applications of deep learning to ocean data inference and subgrid parameterization.
Journal of Advances in Modeling Earth Systems, 11(1):376–399, 2019.

C. J. Lapeyre, A. Misdariis, N. Cazard, D. Veynante, and T. Poinsot.
Training convolutional neural networks to estimate turbulent sub-grid scale reaction rates.
Combustion and Flame, 203:255–264, 2019.

M. Oberlack.
Invariant modeling in large-eddy simulation of turbulence.
Annual Research Briefs, pages 3–22, 1997.

G. Portwood, B. Nadiga, J. Saenz, and D. Livescu.
Interpreting neural network models of residual scalar flux.
arXiv preprint arXiv:2004.07207, 2020.

C. G. Speziale.
Galilean invariance of subgrid-scale stress models in the large-eddy simulation of turbulence.
Journal of fluid mechanics, 156:55–62, 1985.

A. Vollant, G. Balarac, and C. Corre.
Subgrid-scale scalar flux modelling based on optimal estimation theory and machine-learning procedures.
Journal of Turbulence, 18(9):854–878, 2017.