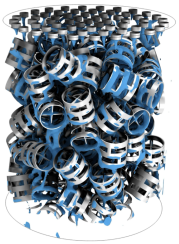
Computational Fluid Dynamics Surrogates for Carbon Capture Design Optimization

Phan Nguyen, Brian Bartoldson, Yeping Hu, Jose Cadena, Amar Saini, Brenda Ng
Lawrence Livermore National Laboratory

Under the Carbon Capture Simulation for Industry Impact ($CCSI^2$) initiative, our project goal is to develop surrogates of computational fluid dynamics (CFD) models of carbon capture systems (CCSs). In this work, the physical phenomena under study are solvent-based post-combustion CCSs, in which CO_2 is captured through an absorption process caused by the interaction between a particular liquid solvent and the CO_2-laden gas inside a reactor column filled with packings. The figure on the left shows a representative section of the bench-scale column of our CCS, in which solvent is being injected into the column from the top "inlets" and the CO_2-laden gas (known as flue gas) is injected below.



CFD models are crucial for the fundamental understanding of interactions between the solvent and packing geometry. The simulation data from these CFD models capture local effects from the thermodynamics, kinetics, heat, and mass transfer of the absorption reaction, and are used to compute the interfacial area (IA, defined as the total area of contact between the solvent and the flue gas) – a key determinant of CO_2 capture efficiency. Using CFD to derive the IAs associated with numerous CCS designs is prohibitively costly. Thus, the goal of our work is to develop surrogates that can predict IAs from the CFD inputs, *without* having to actually run the CFD simulations.

Our data was developed by our subject matter experts from PNNL, who implemented the CFD model of the CCS column described above in StarCCM+ [4]. Taking a vertical slice of the 3D column, the CFD model was run to generate 2D data. Our data consists of two sets of 50 CFD runs – one set for each different packing configuration. Within each set, a run consisted of 500 timesteps (each consisting of velocity, pressure, and volume fraction data at 150K irregularly spaced points) and differed only in the inlet velocity simulation parameter. We used a 80/20 train-validation split, in which the training data was used to train a ML surrogate model, and the validation set was used to evaluate the trained model (i.e., by comparing its predictions of volume-fraction, velocity, and IA against the ground-truth values).

We started with DeepFluids (DF) (Kim et al., 2019), which has two components: a neural network (convolutional autoencoder) that compresses each time frame into a low-dimensional *latent* feature, and a neural network (multilayer perceptron) that propagates this feature forward in time. We improved upon DF by enhancing its constituent components and performing end-to-end training, where both components were optimized together. Our final model [2] achieved 4,000x speedup compared to the original 2D CFD model with only 4% relative error in its IA predictions. While DF delivers massive speedup, it does so at the expense of accuracy because it uses a lossy representation (a grid) of the data. Moreover, it has limited transferability to new packings. Thus, we pivoted towards using graph neural networks (GNNs), where mesh points are represented as graph nodes, point-to-point dependencies are represented as edges connecting the nodes, and interactions between nodes are propagated using a message-passing procedure. Focusing on MeshGraphNets (MGN, a GNN that efficiently learns and produces mesh-based simulations) [5], we open-sourced the first PyTorch implementation of MGN [3] and customized it for distributed training on multiple GPUs, which was required for us to enable training on our 3D column data (3.1 million points compared to 5K points used in the largest model from the original MGN paper).

[1] Bartoldson, B., Hu, Y., Saini, A., Cadena, J., Fu, Y., Bao, J., Xu, Z., Ng, B., and Nguyen, P. (2023). Scientific Computing Algorithms to Learn Enhanced Scalable Surrogates for Mesh Physics. *ICLR 2023 Workshop on Physics for Machine Learning.* https://openreview.net/forum?id=r7n2TPI-4A

[2] Bartoldson, B., Wang, R., Fu, Y., Widemann, D., Nguyen, S., Bao, J., Xu, Z. and Ng, B. (2022). Latent Space Simulation for Carbon Capture Design Optimization. *Proceedings of the AAAI Conference on Artificial Intelligence*, *36*(11), 12447-12453. https://doi.org/10.1609/aaai.v36i11.21511

[3] Nguyen, P., Bartoldson, B., Nguyen, S., Cadena, J., Wang, R., Widemann, D. and Ng, B. (2021). MeshGraphNets [Computer software]. https://github.com/CCSI-Toolset/MGN

[4] Fu, Y., Bao, J., Singh, R., Wang, C., and Xu, Z (2020). Investigation of countercurrent flow profile and liquid holdup in random packed column with local CFD data. *Chemical Engineering Science, 221,* 115693. https://doi.org/10.1016/j.ces.2020.115693

[5] Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., and Battalgia, P. W. (2021). Learning Mesh-Based simulation with Graph Networks. *International Conference on Learning Representations. https://arxiv.org/abs/2010.03409*

This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 using NERSC award BES-ERCAP0024437. This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-ABS-844633.