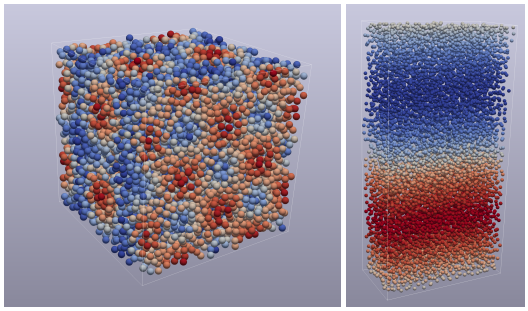
E(3) EQUIVARIANT GRAPH NEURAL NETWORKS FOR PARTICLE-BASED FLUID MECHANICS

Anonymous authors
Paper under double-blind review

ABSTRACT

We contribute to the vastly growing field of machine learning for engineering systems by demonstrating that equivariant graph neural networks have the potential to learn more accurate dynamic-interaction models than their non-equivariant counterparts. We benchmark two well-studied fluid flow systems, namely the 3D decaying Taylor-Green vortex and the 3D reverse Poiseuille flow, and compare equivariant graph neural networks to their non-equivariant counterparts on different performance measures, such as kinetic energy or Sinkhorn distance. Such measures are typically used in engineering to validate numerical solvers. Our main findings are that while being rather slow to train and evaluate, equivariant models learn more physically accurate interactions. This indicates opportunities for future work towards coarse-grained models for turbulent flows, and generalization across system dynamics and parameters.



(a) (b)

Figure 1: Velocity magnitude of Taylor-Green vortex (a) and x-velocity of Reverse Poiseuille (b).

1 PARTICLE-BASED FLUID MECHANICS

Navier-Stokes equations (NSE) are omnipresent in fluid mechanics, hydrodynamics or weather modeling. However, for the majority of problems, solutions are analytically intractable, and obtaining accurate predictions necessitates falling back to numerical solution schemes. Those can be split into two categories: grid/mesh-based (Eulerian description) and particle-based (Lagrangian description).

Smoothed Particle Hydrodynamics. In this work, we investigate Lagrangian methods and more precisely the Smoothed Particle Hydrodynamics (SPH) approach, which was independently developed by Gingold & Monaghan (1977) and Lucy (1977) to simulate astrophysical systems. Since then, SPH has established as the preferred approach in various applications ranging from free surfaces such as ocean waves (Violeau & Rogers, 2016), through fluid-structure interaction systems (Zhang et al., 2021), to selective laser melting in additive manufacturing (Weirather et al., 2019). The main idea behind SPH is to represent the fluid properties at discrete points in space and to use truncated radial interpolation kernel functions to approximate them at any arbitrary location. The kernel functions are used to estimate state statistics which define continuum-scale interactions between particles. The justification for truncating kernel support is the assumption of local interactions

between particles. The resulting discretized equations are then integrated in time using numerical integration techniques like symplectic Euler by which the particle positions are updated.

To generate training data, we implemented our own SPH solver based on the transport velocity formulation by Adami et al. (2013), which promises a homogeneous particle distribution over the domain. We then selected two flow cases, both of which have analytical solutions: the 3D laminar Taylor-Green Vortex and the 3D Reverse Poiseuille Flow. Having the analytical solution is useful both to guarantee that the datasets are physically valid, and secondly to directly evaluate the learned solver on the analytical solutions. We are planning to open-source the datasets in the near future.

Taylor-Green Vortex. The Taylor-Green vortex system (TGV, see Figure 1 (a)) with Reynolds number of $\mathrm{Re} = 100$ is *laminar*, i.e., the initial vortex system simply decays without nonlinear generation of small scales. The TGV has been extensively studied starting with Taylor & Green (1937) and continuing all the way to Sharma & Sengupta (2019). The TGV system is typically initialized with a velocity field given by

$$u = -\cos(kx)\cos(ky)\cos(kz)\ , \qquad v = \sin(kx)\cos(ky)\cos(kz)\ , \qquad w = 0\ , \tag{1}$$

where k is an integer multiple of 2π. For this set of initial conditions a time-dependent analytical solution for the kinetic energy exists, which exhibits exponential decay as $\propto \exp(-24\pi^2/\mathrm{Re}\cdot t)$. The TGV datasets used in this work consist of 8/2/2 trajectories for training/validation/testing, where each trajectory comprises 8000 particles. Each trajectory spans 1s physical time and was simulated with $dt = 0.001$ resulting in 1000 time steps per trajectory. The ultimate goal would be to learn the dynamics over much larger time steps than those taken by the numerical solver, but here we just want to demonstrate the applicability of learned approaches to reproducing numerical solver results.

Reverse Poiseuille Flow. The Poiseuille Flow, i.e. laminar channel flow, is another well-studied flow case in fluid mechanics. However, channel flow requires the treatment of wall-boundary conditions, which is beyond the current focus of this work. In this work, we therefore consider data obtained by Reverse Poiseuille Flow (RPF, see Figure 1 (b)) (Fedosov et al., 2008), which essentially consists of two opposing streams in a fully periodic domain. Those flows are exposed to opposite force fields, i.e., the upper and lower half are accelerated in negative x direction and positive x direction, respectively. Due to the stationarity of the flow, the RPF dataset consists of one long trajectory spanning 120s. The flow field is discretized by 8000 particles and simulated with $dt = 0.001$, followed by subsampling at every 10th step. Thus, we aim to train models to perform temporal coarse-graining. The resulting number of training/validation/testing instances is the same as for TGV, namely 8000/2000/2000.

2 (EQUIVARIANT) GRAPH NETWORK-BASED SIMULATORS

We first formalize the task of autoregressively predicting the next state of a Lagrangian fluid mechanics simulation based on the notation from Sanchez-Gonzalez et al. (2020). Let X^t denote the state of a particle system at time t. One full trajectory of $K+1$ steps can be written as $\mathbf{X}^{t_{0:K}} = (X^{t_0}, \ldots, X^{t_K})$. Each state $\mathbf{X}^t$ is made up of N particles, namely $\mathbf{X}^t = (\mathbf{x}_1^t, \mathbf{x}_2^t, \ldots \mathbf{x}_N^t)$, where each $\mathbf{x}_i$ is the state vector of the i-th particle. However, the inputs to the learned simulator can span multiple time instances. Each node $\mathbf{x}_i^t$ can contain node-level information like the current position $\mathbf{p}_i^t$ and a time sequence of H previous velocity vectors $\dot{\mathbf{p}}^{t_{1+k-H:k}}$, as well as global features like the external force field $\mathbf{f}_i$ in the Reverse Poiseuille Flow. To build the connectivity graph, we use an interaction radius of ~ 1.5 times the average interparticle distance. This results in around 10-20 one-hop neighbors.

Graph Network-based Simulator. The Graph Network-based Simulator (GNS) framework (Sanchez-Gonzalez et al., 2020) is one of the most popular learned surrogates for engineering particle-based simulations. The main idea of the GNS model is to use the established encoder-processor-decoder architecture (Battaglia et al., 2018) with a processor that stacks several message passing layers (Gilmer et al., 2017). One major strength of the GNS model lies in its simplicity given that all its building blocks are simple MLPs. However, the performance of GNS when predicting long trajectories strongly depends on choosing the right amount of Gaussian noise to perturb input data. Additionally, GNS and other non-equivariant models are less data-efficient (Batzner et al.,

Table 1: Performance measures on the Taylor-Green vortex and Reverse Poiseuille flow. The Sinkhorn distance is averaged over test rollouts, the inference time is obtained for one rollout step of 8000 particles. The $\text{MSE}_{E_{kin}}$ measure is relevant on the Taylor-Green vortex, the Sinkhorn distance on the Reverse Poiseuille flow.

	Taylor-Green vortex			Reverse Poiseuille flow		
	SEGNN	GNS	SPH	SEGNN	GNS	SPH
$\text{MSE}_{\mathbf{p}}$	7.7e-5	1.3e-4	-	7.7e-3	8.0e-3	-
$\text{MSE}_{E_{kin}}$	5.3e-5	1.3e-4	-	2.8e-1	3.0e-1	-
Sinkhorn	1.3e-7	1.1e-7	-	7.8e-8	1.9e-6	-
Time [ms]	290	32	9.7	180	33	110
# params	720k	630k	-	180k	630k	-

2022). For these reasons, we implement and tune GNS as a comparison baseline, and use it as an inspiration for which setup, features, and hyperparameters to use for equivariant models.

Steerable E(3)-equivariant Graph Neural Network. Steerable E(3)-equivariant Graph Neural Networks (SEGNNs) (Brandstetter et al., 2022a) are an instance of E(3)-equivariant GNNs, i.e., GNNs that are equivariant with respect to isometries of the Euclidean space (rotations, translations, and reflections). Most E(3)-equivariant GNNs that are tailored towards molecular property prediction tasks, (Batzner et al., 2022; Batatia et al., 2022) restrict the parametrization of the Clebsch-Gordan tensor products to an MLP-parameterized embedding of pairwise distances. In contrast, SEGNNs use general steerable node and edge attributes which can incorporate any kind of physical quantity, and directly learn the weights of the Clebsch-Gordan tensor product. This makes SEGNN well suited for a wider scope of problems, as for example fluid mechanics problems.

Steerable attributes strongly impact the Clebsch-Gordan tensor products, and thus finding physically meaningful edge and node attributes is crucial for good performance. In particular, we chose edge attributes $\hat{a}_{ij} = V(\mathbf{p}_{ij})$, where $V(\cdot)$ is the spherical harmonic embedding and $\mathbf{p}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ are the pairwise distances. We further choose node attributes $\hat{a}_i = V(\bar{\dot{\mathbf{p}}}_i) + \sum_{k \in \mathcal{N}(i)} \hat{a}_{ik}$, where $\bar{\dot{\mathbf{p}}}_i$ are averaged historical velocities and $\mathcal{N}(i)$ is the i-neighborhood. As for node and edge features, we found that concatenated historical velocities for the nodes and pairwise displacements for the edges capture best the Navier-Stokes dynamics.

For training SEGNNs, we verified that adding Gaussian noise to the inputs (Sanchez-Gonzalez et al., 2020) indeed significantly improves performance. We further found that explicitly concatenating the external force vector $\mathbf{f}_i$ to the node features boosts performance in the RPF case. However, adding $\mathbf{f}_i$ to the node attributes $\hat{a}_i\prime = V(\mathbf{f}_i) + V(\bar{\dot{\mathbf{p}}}_i) + \sum_{k \in \mathcal{N}(i)} \hat{a}_{ik}$ does not improve performance.

3 Results

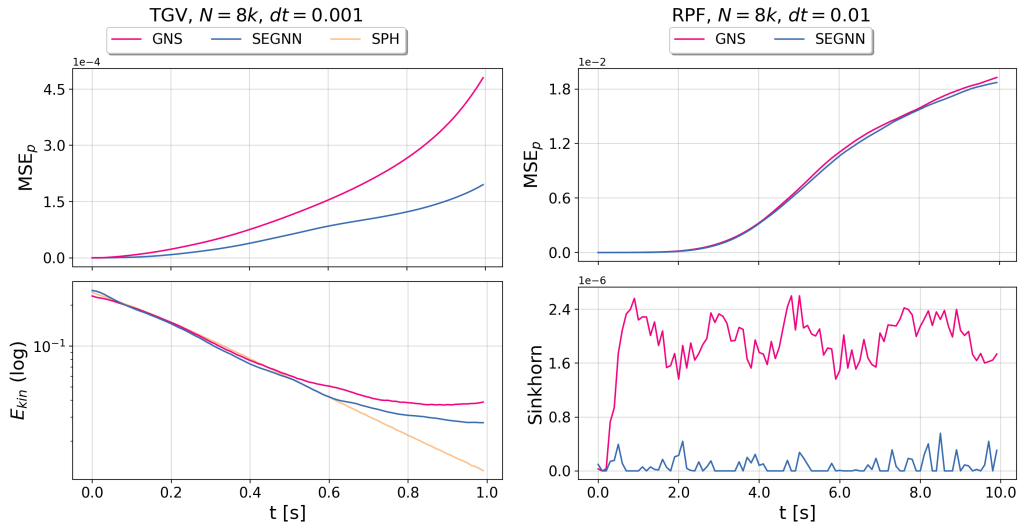
The task we train on is the autoregressive prediction of accelerations $\ddot{\mathbf{p}}$ given the current position $\mathbf{p}_i$ and $H = 5$ past velocities of the particles. We measured the performance of the GNS and the SEGNN models in four aspects when evaluating on the test dataset:

- We use the mean-squared error (MSE) of particle positions when rolling out a trajectory over 1000 time steps, and average over those. This is also the validation loss during training.
- We use Sinkhorn distance, as an optimal transport distance measure between distributions. Lower values indicate that the particle distribution is closer to the reference data.
- We track the kinetic energy ($= 0.5mv^2$) as a global measure of physical behavior.
- We measure the runtime per iteration as a measure of efficiency.

Performance comparisons are summarized in Table 1. GNS and SEGNN models have roughly the same number of parameters for Taylor-Green (both have 5 layers and 128-dim features), whereas for

reverse Poiseuille SEGNN has three times less parameters than GNS (SEGNN has 64-dim features). Looking at the timing in Table 1, equivariant models of similar size are one order of magnitude slower than non-equivariant ones. This is a known result and is related to the constraint of how the Clebsch-Gordan tensor product can be implemented on accelerators like GPUs.

Taylor-Green Vortex. One of the major challenges of the Taylor-Green dataset are the varying input and output scales throughout a trajectory, by up to one order of magnitude. Consequently, this results in the larger importance of first-time steps in the loss even after data normalization. Figure 2 (a) summarizes the most important performance properties of the Taylor-Green vortex experiment. In general, both models match the ground truth kinetic energy well, but GNS drifts away from the reference SPH curve earlier. Both learned solvers, seem to preserve larger system velocities resulting in higher E_{kin}. The rollout MSE for this case matches the behavior seen in E_{kin}.



(a) (b)

Figure 2: Taylor-Green vortex (a) and Reverse Poiseuille (b) performance evolution.

Reverse Poiseuille Flow. The challenge of the reverse Poiseuille case lies in the different velocity scales between the main flow direction (x-axis) and the y and z components of the velocity. Although such unbalanced velocities are used as inputs, target accelerations in x-, y-, and z-direction all underlie similar distributions. This, combined with temporal coarsening makes the problem sensitive to input deviations. Figure 2 (b) shows that SEGNN reproduces the particle distribution almost perfectly, whereas GNS shows signs of particle clustering, resulting in a larger Sinkhorn distance. Interestingly, the shear layers in-between the inverted flows (around planes $y = \{0, 1, 2\}$) seem to have the largest deviation from the ground truth, which could be source of clusters, see Figure 3.

4 Future Work

In this work, we demonstrate that equivariant models are well suited to capture underlying physics properties of particle-based fluid mechanics systems. Natural future steps are enforcing physical behaviors such as homogeneous particle distributions, and including recent developments for neural PDE training into the training procedure of Sanchez-Gonzalez et al. (2020). The latter include e.g., the push-forward trick and temporal bundling (Brandstetter et al., 2022b). One major weakness of recursively applied solvers, which these strategies aim to mitigate, is error accumulation, which in most cases leads to out-of-distribution states, and consequently unphysical behavior after several rollout steps. We conjecture that together with such extensions equivariant models offer a promising direction to tackle some of the long-standing problems in fluid mechanics, such as the learning of coarse-grained representations of turbulent flow problems, e.g. Taylor-Green (Brachet et al., 1984), or learning the multi-resolution dynamics of NSE problems (Hu et al., 2017).

REFERENCES

Stefan Adami, XY Hu, and Nikolaus A Adams. A transport-velocity formulation for smoothed particle hydrodynamics. *Journal of Computational Physics*, 241:292–307, 2013.

Ilyes Batatia, Dávid Péter Kovács, Gregor N. C. Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields, 2022.

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. 2018.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1): 2453, 2022.

Marc E Brachet, D Meiron, S Orszag, B Nickel, R Morf, and Uriel Frisch. The taylor-green vortex and fully developed turbulence. *Journal of Statistical Physics*, 34(5-6):1049–1063, 1984.

Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling. Geometric and physical quantities improve e(3) equivariant message passing. In *International Conference on Learning Representations*, 2022a.

Johannes Brandstetter, Daniel E. Worrall, and Max Welling. Message passing neural PDE solvers. In *International Conference on Learning Representations*, 2022b.

Dmitry A Fedosov, Bruce Caswell, and George Em Karniadakis. Reverse poiseuille flow: the numerical viscometer. In *Aip conference proceedings*, volume 1027, pp. 1432–1434. American Institute of Physics, 2008.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pp. 1263–1272. PMLR, 2017.

Robert A Gingold and Joseph J Monaghan. Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*, 181(3):375–389, 1977.

Wei Hu, Wenxiao Pan, Milad Rakhsha, Qiang Tian, Haiyan Hu, and Dan Negrut. A consistent multi-resolution smoothed particle hydrodynamics method. *Computer Methods in Applied Mechanics and Engineering*, 324:278–299, 2017.

Leon B Lucy. A numerical approach to the testing of the fission hypothesis. *Astronomical Journal, vol. 82, Dec. 1977, p. 1013-1024.*, 82:1013–1024, 1977.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter Battaglia. Learning to simulate complex physics with graph networks. In *International conference on machine learning*, pp. 8459–8468. PMLR, 2020.

Nidhi Sharma and Tapan K Sengupta. Vorticity dynamics of the three-dimensional taylor-green vortex problem. *Physics of Fluids*, 31(3):035106, 2019.

Geoffrey Ingram Taylor and Albert Edward Green. Mechanism of the production of small eddies from large ones. *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*, 158(895):499–521, 1937.

Damien Violeau and Benedict D Rogers. Smoothed particle hydrodynamics (sph) for free-surface flows: past, present and future. *Journal of Hydraulic Research*, 54(1):1–26, 2016.

Johannes Weirather, Vladyslav Rozov, Mario Wille, Paul Schuler, Christian Seidel, Nikolaus A Adams, and Michael F Zaeh. A smoothed particle hydrodynamics model for laser beam melting of ni-based alloy 718. *Computers & Mathematics with Applications*, 78(7):2377–2394, 2019.

Chi Zhang, Massoud Rezavand, and Xiangyu Hu. A multi-resolution sph method for fluid-structure interactions. *Journal of Computational Physics*, 429:110028, 2021.

A APPENDIX

Reverse Poiseuille plots.

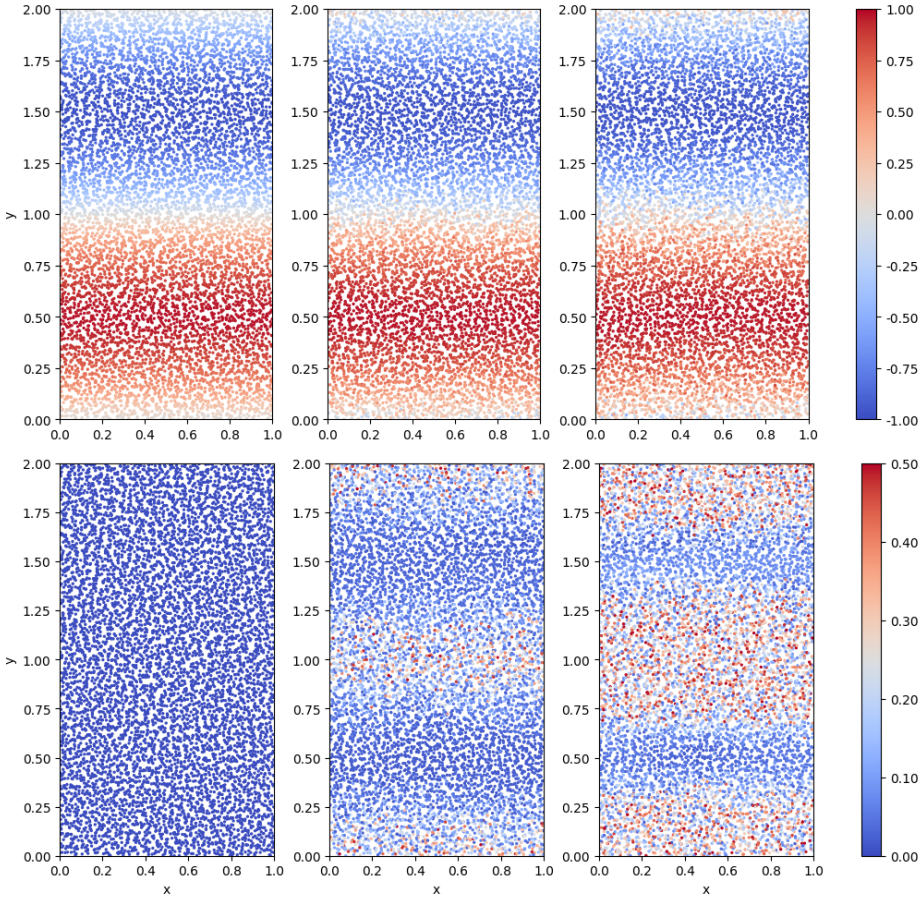


Figure 3: Reverse Poiseuille x-y view of velocity fields (top) and position error (bottom) at time steps [10, 500, 990] (left to right) of the test rollout.