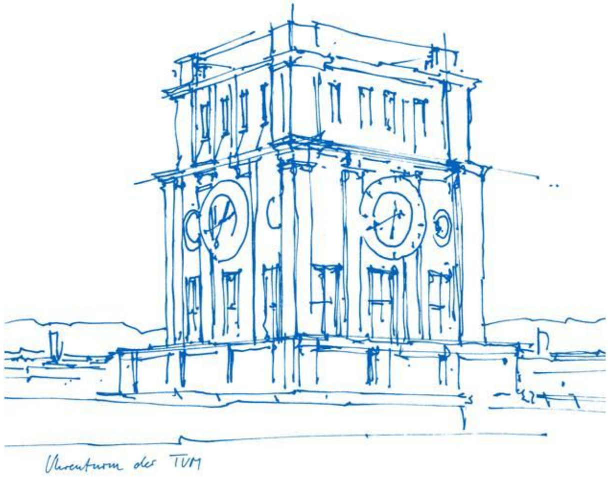
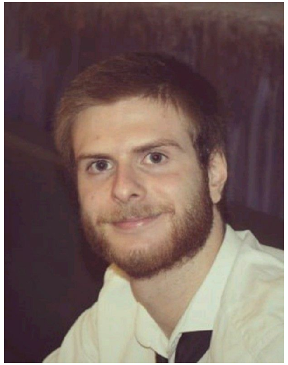
Learning Lagrangian Fluid Mechanics with E(3)-Equivariant GNNs

Artur Toshev
Technical University of Munich

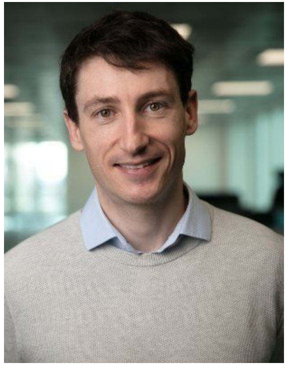


Geometric Science of Information, 2023

Main Collaborators



Gianluca Galletti
TUM, Informatics
@ggalletti_



Johannes Brandstetter
MSR AI4Science
@jo_brandstetter

Stefan Adami
TUM, Mech. Eng.

Nikolaus Adams
TUM, Mech. Eng.

Outline

- Datasets
- Smoothed Particle Hydrodynamics and JAX-SPH
- Historic Attribute Embeddings + SEGNN
- Experiments

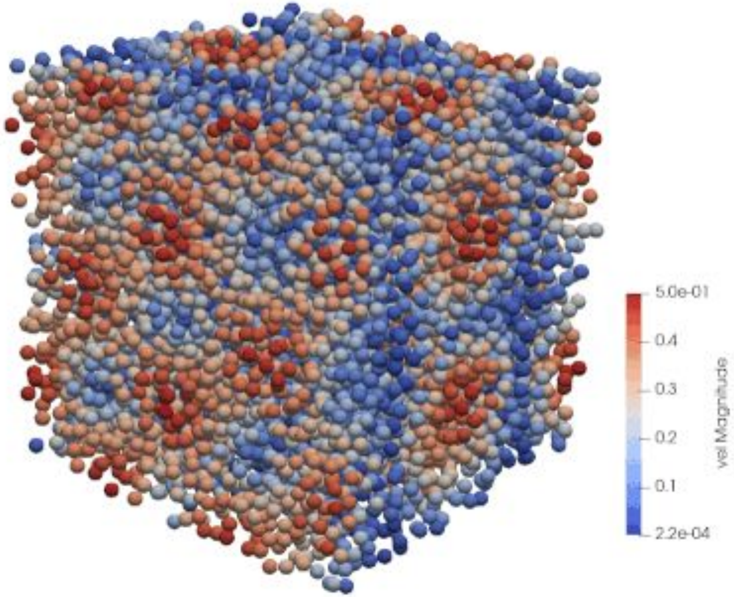
Datasets

Decaying Taylor-Green Vortex - 3D

<u>Training data</u>

- 20x20x20 particles
- periodic boundaries
- 80 runs
- 1000 steps each

-> 8000 samples (every 10th)

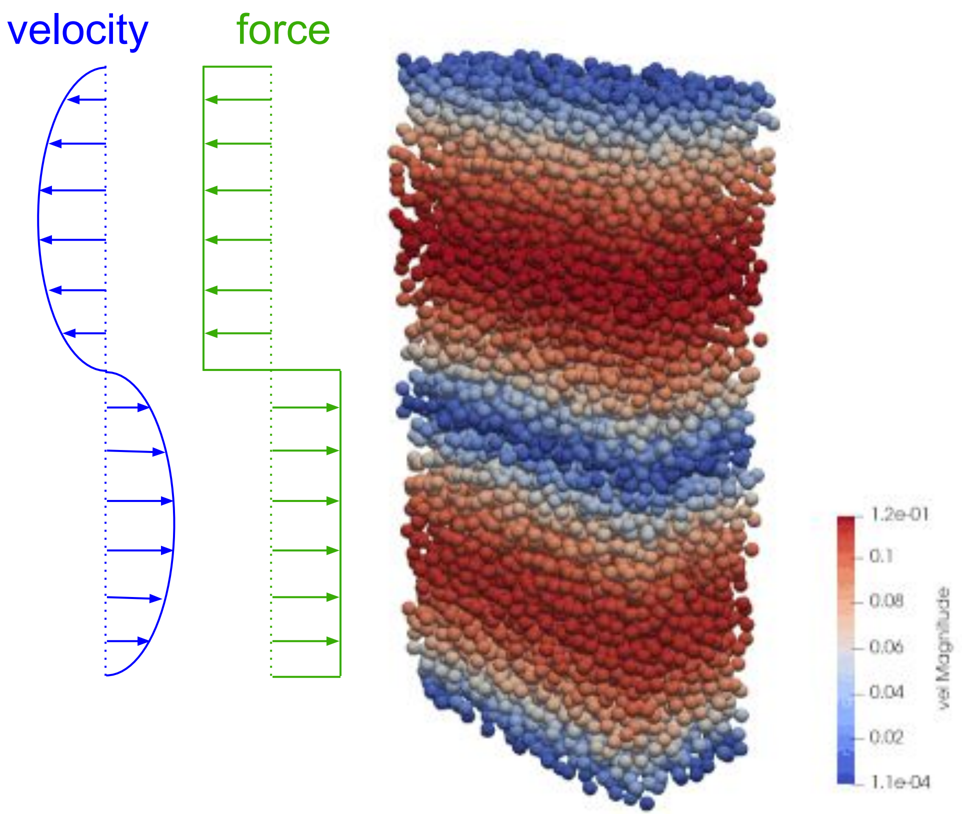


Reverse Poiseuille Flow - 3D

<u>Training data</u>

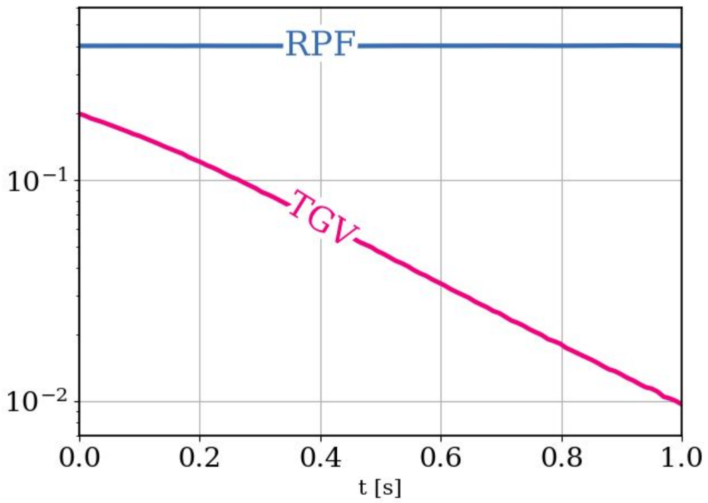
- 20x40x10 particles
- periodic boundaries
- 1 run
- 80k steps

-> 8k samples (every 10th)



Datasets

Total kinetic energy evolution



Smoothed Particle Hydrodynamics

Smoothed Particle Hydrodynamics

1. Kernel interpolation

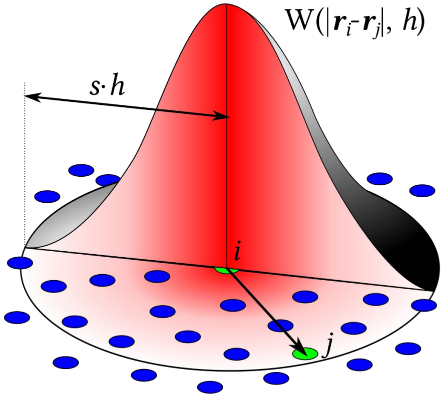
$$A(\mathbf{r}) = \int A(\mathbf{r}')W(\mathbf{r}-\mathbf{r}', h)\mathbf{dr}' \qquad 1 = \int W(\mathbf{r}-\mathbf{r}', h)\mathbf{dr}'$$

continuous

discrete

$$A_i = \sum_{j=1}^{N} A_j w_{ij} = \sum_{j=1}^{N} A_j W_{ij} dV_j \qquad 1 = \sum_{j=1}^{N} w_{ij}$$

$$W(\mathbf{r}_i - \mathbf{r}_j, h) = w_{ij} dV_j$$



https://en.wikipedia.org/wiki/Smoothed-particle_hydrodynamics

2. Navier Stokes Equations (incompressible)

$$\nabla \cdot \mathbf{u} = 0$$

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \mathbf{u} + g$$

“Smoothed particle hydrodynamics: theory and application to non-spherical star” (Gingold & Monaghan, 1977)
"A numerical approach to the testing of the fission hypothesis” (Lucy, 1977)

Smoothed Particle Hydrodynamics

Algorithm 3: One step of SPH

input : $\mathbf{x}, \mathbf{u}, m, \nu$
output: $\mathbf{x}, \mathbf{u}$
for $i \in particles$ **do**
    $L_i = \text{FindNeighbors}(i)$
end
for $i \in particles$ **do**
    $\rho_i = \sum_{L_i} m_j W_{ij}$
    $p_i = \text{EquationOfState}(\rho_i)$
end
for $i \in particles$ **do**
    $\mathbf{a}_i^{pressure} = -\sum_{L_i} m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2}\right) \nabla W_{ij}$ // $-\frac{1}{\rho_i}\nabla p_i$
    $\mathbf{a}_i^{viscosity} = 2\nu \sum_{L_i} \frac{m_j}{\rho_j} \mathbf{u}_{ij} \frac{\mathbf{x}_{ij} \cdot \nabla W_{ij}}{\mathbf{x}_{ij} \cdot \mathbf{x}_{ij} + 0.01h^2}$ // $\nu \nabla^2 \mathbf{u}_i$
    $\mathbf{a}_i^{force} = \mathbf{g}$
    $\mathbf{a}_i(t) = \mathbf{a}_i^{pressure} + \mathbf{a}_i^{viscosity} + \mathbf{a}_i^{force}$
end
for $i \in particles$ **do**
    $\mathbf{u}_i(t + \Delta t) = \mathbf{u}_i(t) + \Delta t \mathbf{a}_i(t)$
    $\mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \Delta t \mathbf{u}_i(t + \Delta t)$ // *symplectic Euler*
end

Smoothed Particle Hydrodynamics

Algorithm 3: One step of SPH

input : $\mathbf{x}, \mathbf{u}, m, \nu$
output: $\mathbf{x}, \mathbf{u}$
for $i \in particles$ **do**
  $L_i = \text{FindNeighbors}(i)$
end

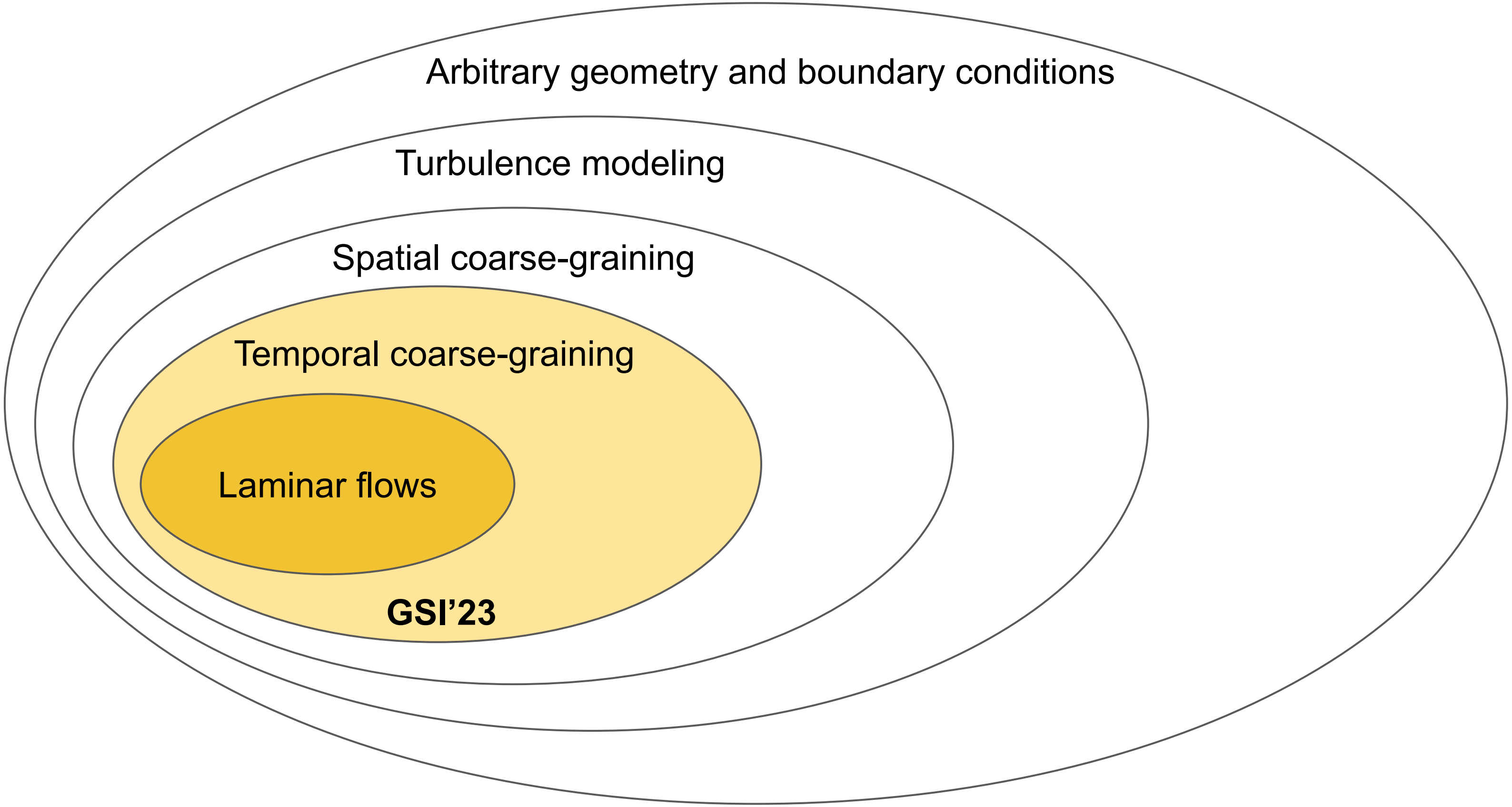
Learned

for $i \in particles$ **do**
  $\rho_i = \sum_{L_i} m_j W_{ij}$
  $p_i = \text{EquationOfState}(\rho_i)$
end
for $i \in particles$ **do**
  $\mathbf{a}_i^{pressure} = -\sum_{L_i} m_j \left(\frac{p_i}{\rho_i^2} + \frac{p_j}{\rho_j^2}\right) \nabla W_{ij}$ // $-\frac{1}{\rho_i}\nabla p_i$
  $\mathbf{a}_i^{viscosity} = 2\nu \sum_{L_i} \frac{m_j}{\rho_j} \mathbf{u}_{ij} \frac{\mathbf{x}_{ij} \cdot \nabla W_{ij}}{\mathbf{x}_{ij} \cdot \mathbf{x}_{ij} + 0.01h^2}$ // $\nu \nabla^2 \mathbf{u}_i$
  $\mathbf{a}_i^{force} = \mathbf{g}$
  $\mathbf{a}_i(t) = \mathbf{a}_i^{pressure} + \mathbf{a}_i^{viscosity} + \mathbf{a}_i^{force}$
end
for $i \in particles$ **do**
  $\mathbf{u}_i(t + \Delta t) = \mathbf{u}_i(t) + \Delta t \mathbf{a}_i(t)$
  $\mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \Delta t \mathbf{u}_i(t + \Delta t)$ // *symplectic Euler*
end

Ultimate Goal

?

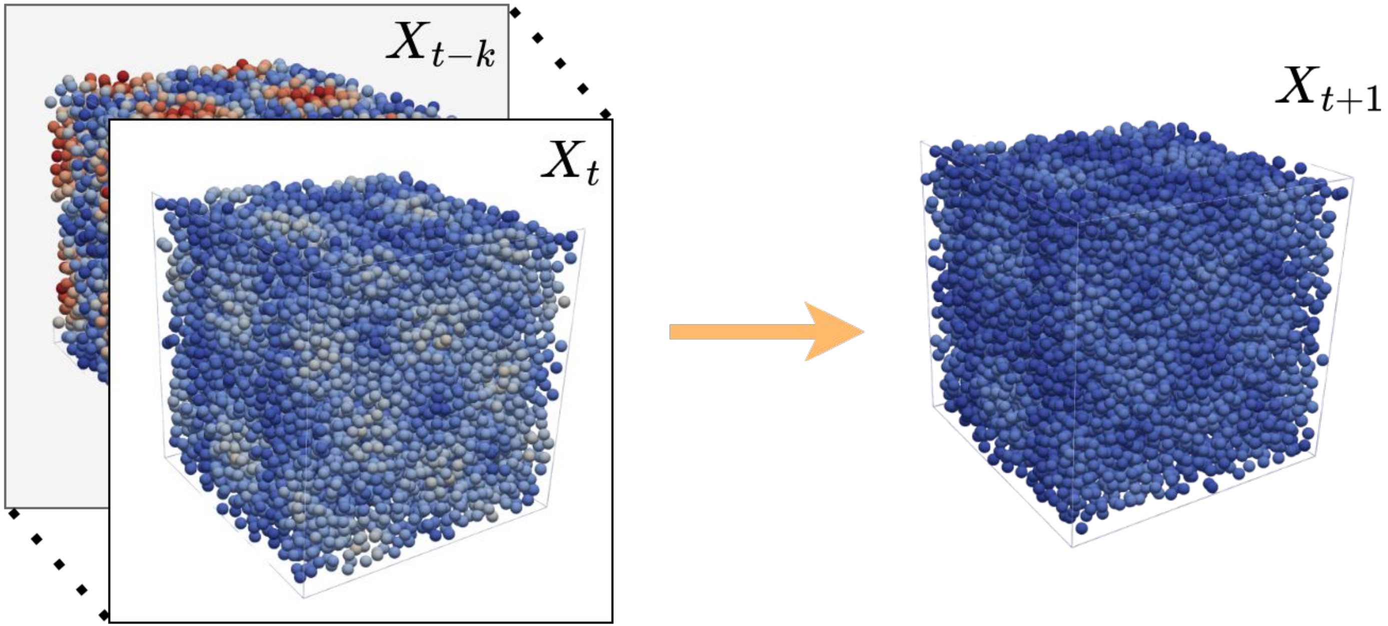
Ultimate Goal - by problem complexity



Historic Attribute Embeddings + SEGNN

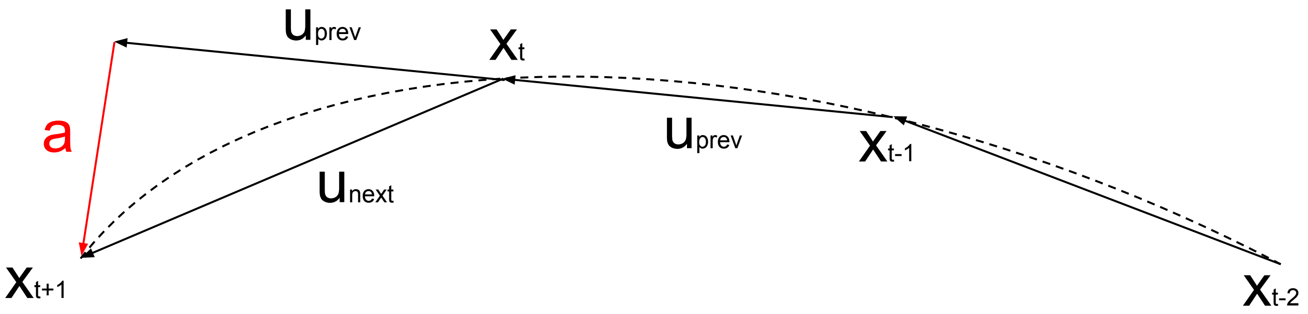
Learning problem

Task: predict acceleration from previous velocities

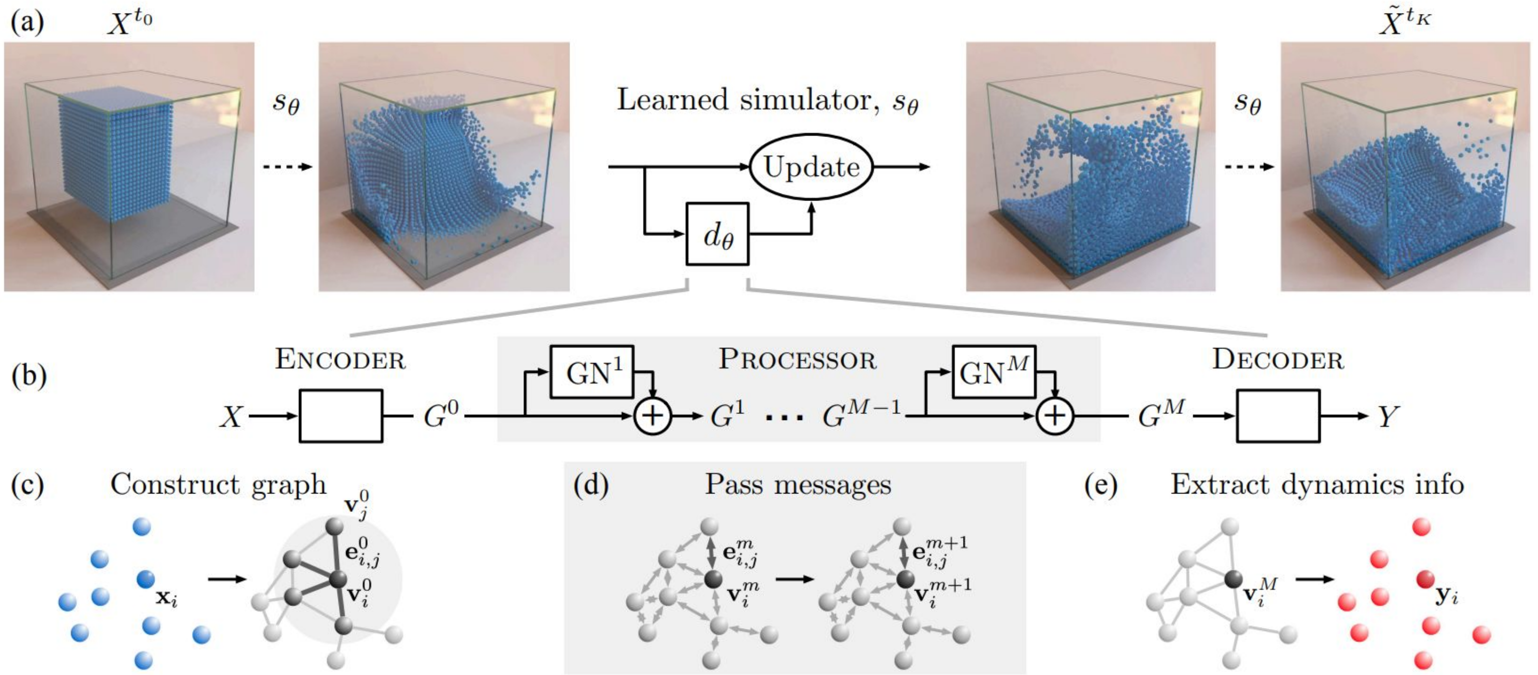


Learning problem

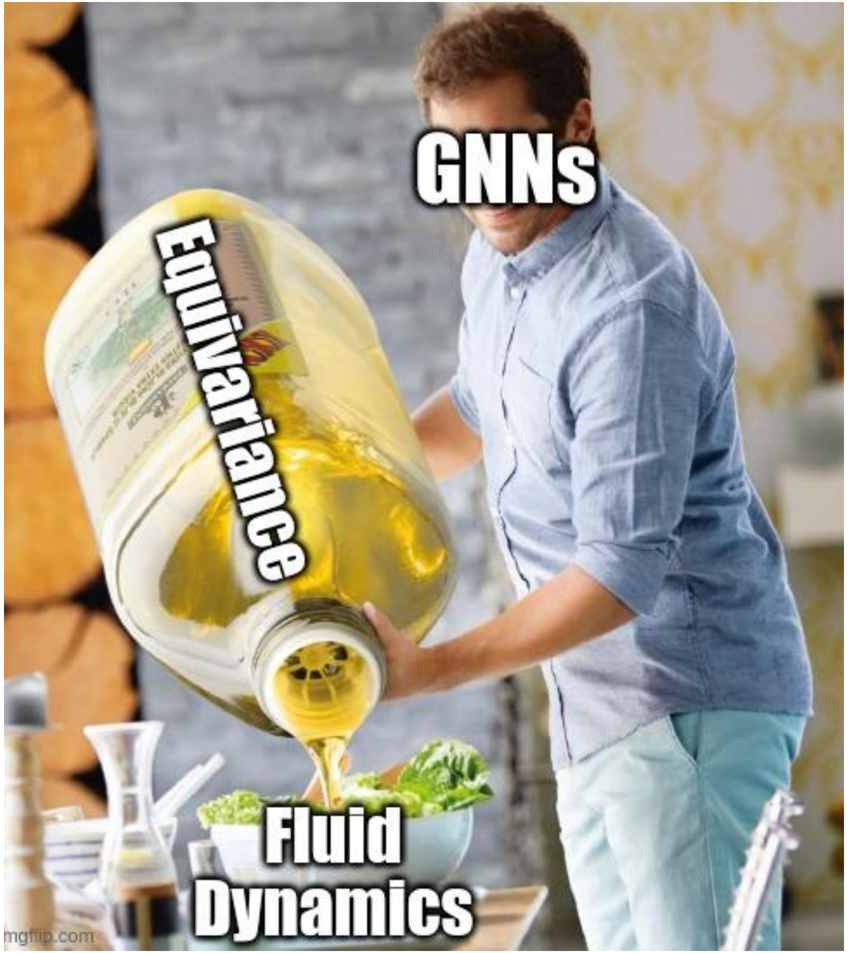
Task: predict acceleration from previous velocities



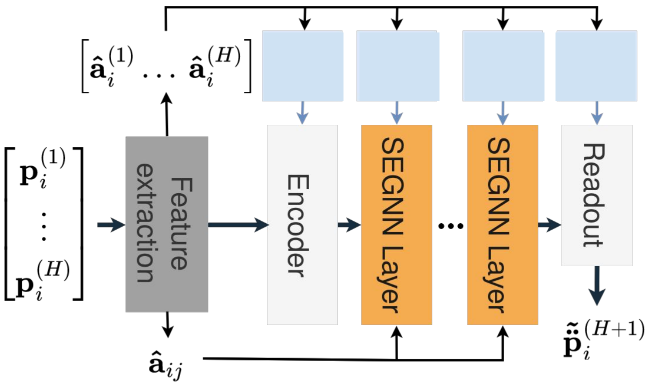
Architecture - GNS



"Learning to Simulate Complex Physics with Graph Networks" (Sanchez-Gonzalez et al., 2020)

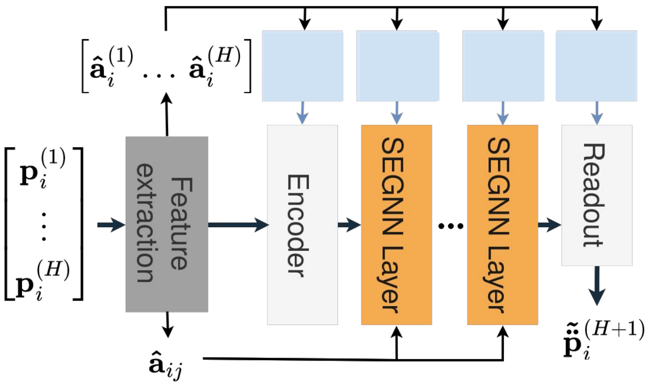

GNNs
Equivariance
Fluid
Dynamics
imgflip.com

Architecture - SEGNN



"Geometric and Physical Quantities Improve E(3) Equivariant Message Passing" (Brandstetter et al., 2022)

Architecture - SEGNN



$$\hat{\mathbf{m}}_{ij} = M_{\hat{\mathbf{a}}_{ij}}\left(\hat{\mathbf{f}}_i, \hat{\mathbf{f}}_j, \|x_i - x_j\|^2\right)$$

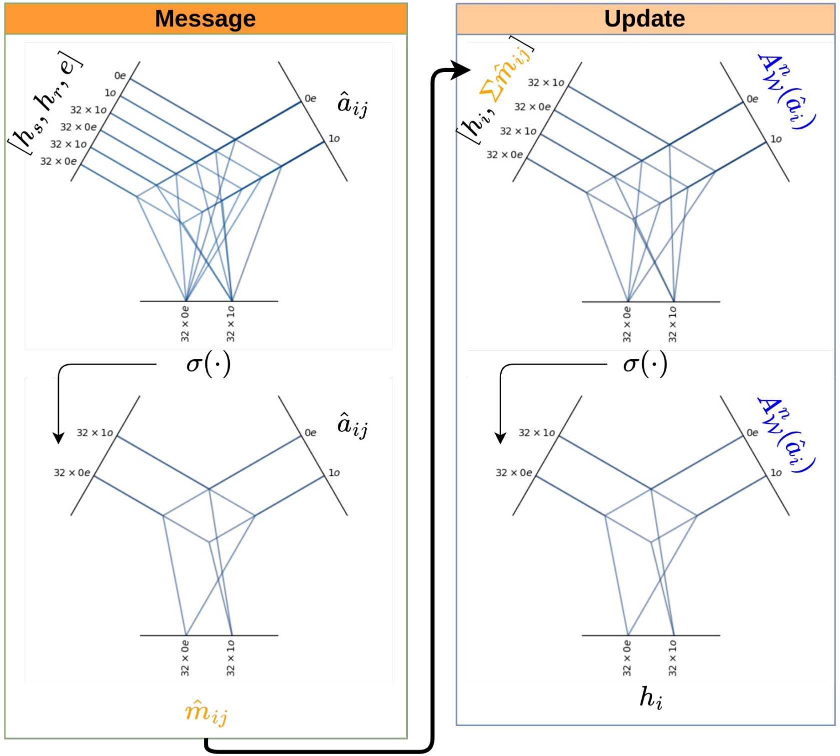
$$\hat{\mathbf{f}}_i' = U_{\hat{\mathbf{a}}_i}\left(\hat{\mathbf{f}}_i, \sum_{j \in \mathcal{N}(i)} \hat{\mathbf{m}}_{ij}\right)$$

“Geometric and Physical Quantities Improve E(3) Equivariant Message Passing” (Brandstetter et al., 2022)

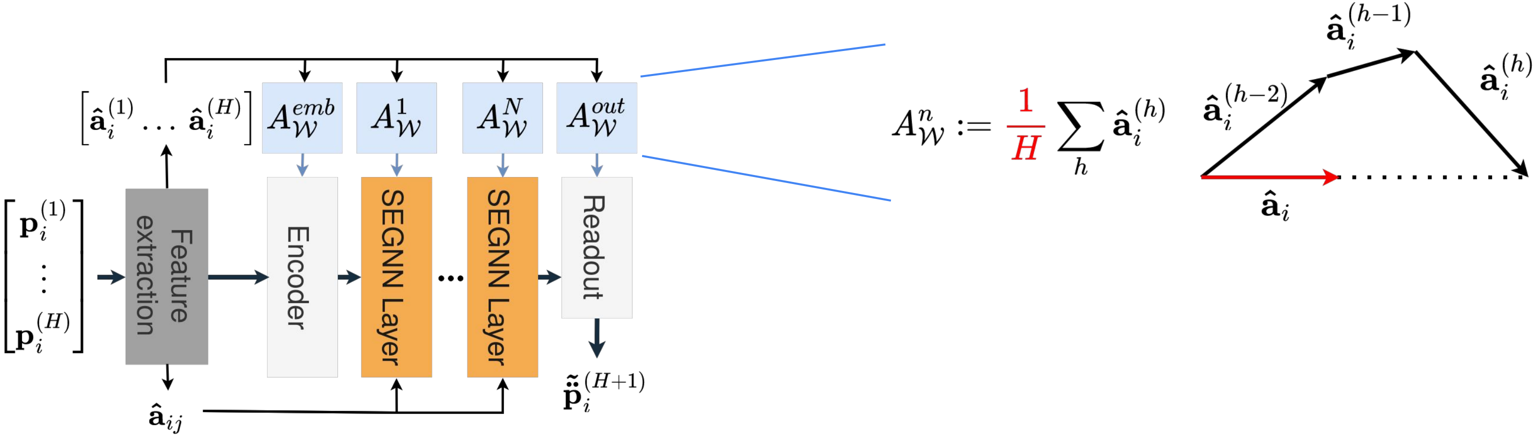
Architecture - HAE-SEGNN layer

$$\hat{\mathbf{m}}_{ij} = M_{\hat{\mathbf{a}}_{ij}}\left(\hat{\mathbf{f}}_i, \hat{\mathbf{f}}_j, \|x_i - x_j\|^2\right)$$

$$\hat{\mathbf{f}}_i' = U_{\hat{\mathbf{a}}_i}\left(\hat{\mathbf{f}}_i, \sum_{j \in \mathcal{N}(i)} \hat{\mathbf{m}}_{ij}\right)$$

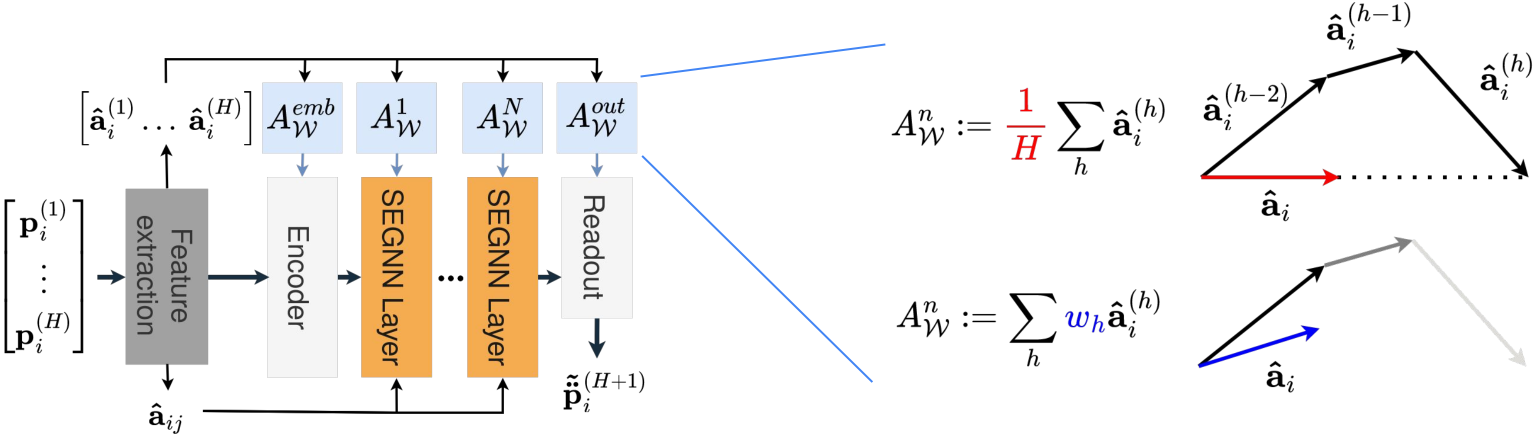


HAE-SPH



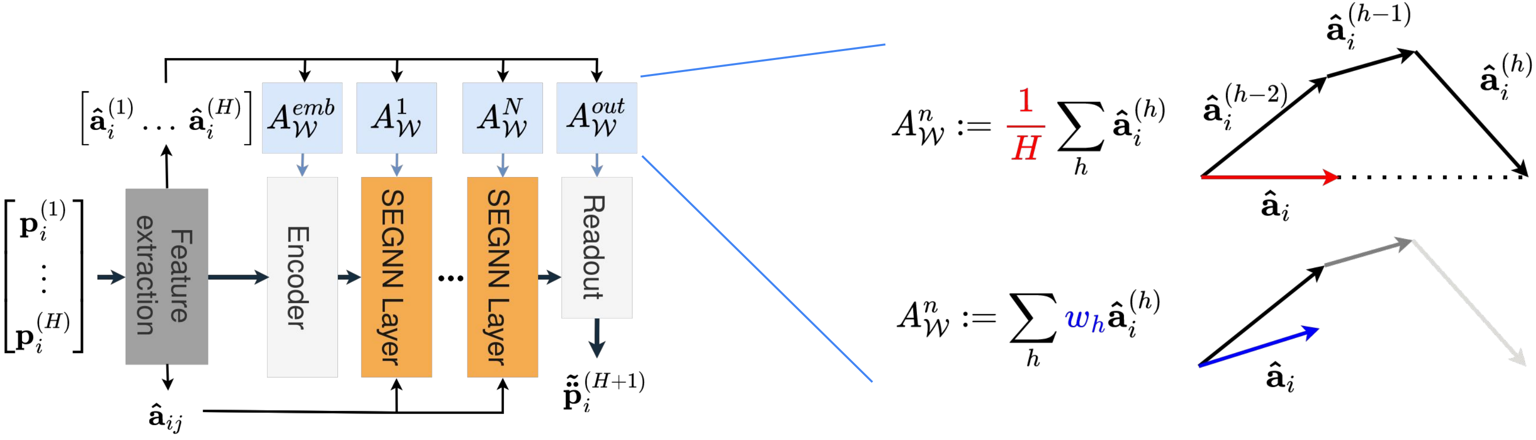
$\left[\hat{\mathbf{a}}_i^{(1)} \ \dots \ \hat{\mathbf{a}}_i^{(H)}\right]$
$A_{\mathcal{W}}^{emb}$
$A_{\mathcal{W}}^{1}$
$A_{\mathcal{W}}^{N}$
$A_{\mathcal{W}}^{out}$
$\begin{bmatrix} \mathbf{p}_i^{(1)} \\ \vdots \\ \mathbf{p}_i^{(H)} \end{bmatrix}$
Feature extraction
Encoder
SEGNN Layer
...
SEGNN Layer
Readout
$\hat{\mathbf{a}}_{ij}$
$\tilde{\ddot{\mathbf{p}}}_i^{(H+1)}$
$A_{\mathcal{W}}^{n} := \frac{1}{H} \sum_h \hat{\mathbf{a}}_i^{(h)}$
$\hat{\mathbf{a}}_i^{(h-1)}$
$\hat{\mathbf{a}}_i^{(h-2)}$
$\hat{\mathbf{a}}_i^{(h)}$
$\hat{\mathbf{a}}_i$

HAE-SPH



$\left[\hat{\mathbf{a}}_i^{(1)} \ldots \hat{\mathbf{a}}_i^{(H)}\right]$
$A_{\mathcal{W}}^{emb}$
$A_{\mathcal{W}}^{1}$
$A_{\mathcal{W}}^{N}$
$A_{\mathcal{W}}^{out}$
$\begin{bmatrix} \mathbf{p}_i^{(1)} \\ \vdots \\ \mathbf{p}_i^{(H)} \end{bmatrix}$
Feature extraction
Encoder
SEGNN Layer
SEGNN Layer
Readout
$\hat{\mathbf{a}}_{ij}$
$\tilde{\ddot{\mathbf{p}}}_i^{(H+1)}$
$A_{\mathcal{W}}^{n} := \frac{1}{H} \sum_h \hat{\mathbf{a}}_i^{(h)}$
$\hat{\mathbf{a}}_i^{(h-2)}$
$\hat{\mathbf{a}}_i^{(h-1)}$
$\hat{\mathbf{a}}_i^{(h)}$
$\hat{\mathbf{a}}_i$
$A_{\mathcal{W}}^{n} := \sum_h w_h \hat{\mathbf{a}}_i^{(h)}$
$\hat{\mathbf{a}}_i$

HAE-SPH

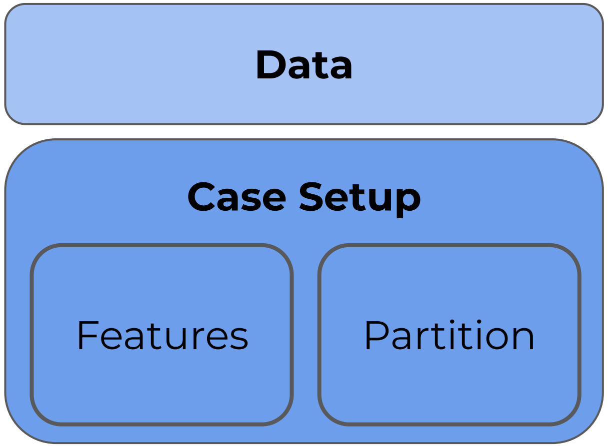


Three flavors of historic attribute embedding:

$$A_{\mathcal{W},avg} = \frac{1}{H}\sum_h \hat{\mathbf{a}}_i^{(h)}\,, \qquad A_{\mathcal{W},lin} = \sum_h w_h \hat{\mathbf{a}}_i^{(h)}\,, \qquad A_{\mathcal{W},\otimes} = \sigma\left(\hat{\mathbf{a}}_i^{(1:H)} \otimes_{CG}^{\mathcal{W}} \hat{\mathbf{a}}_i^{H}\right)$$

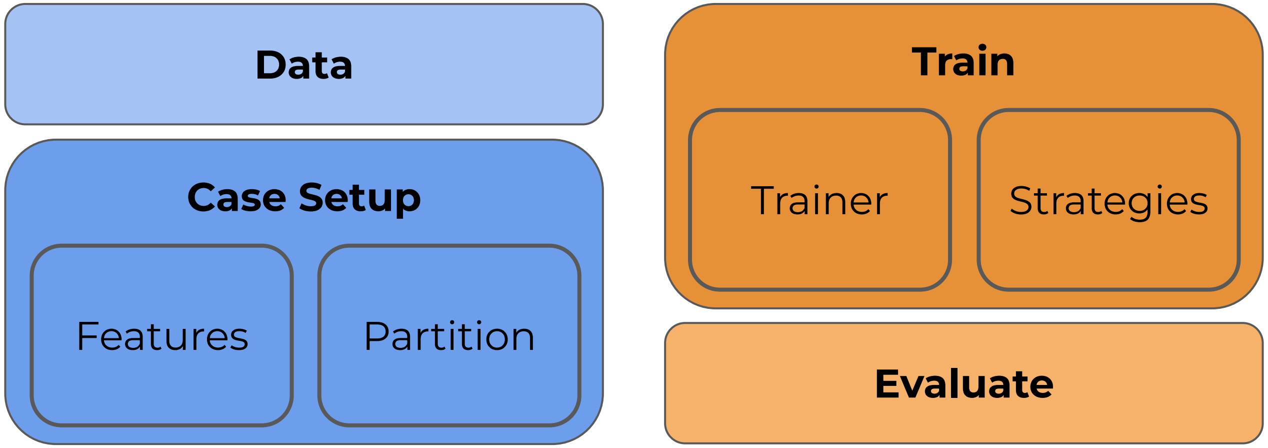
Experiments

LagrangeBench



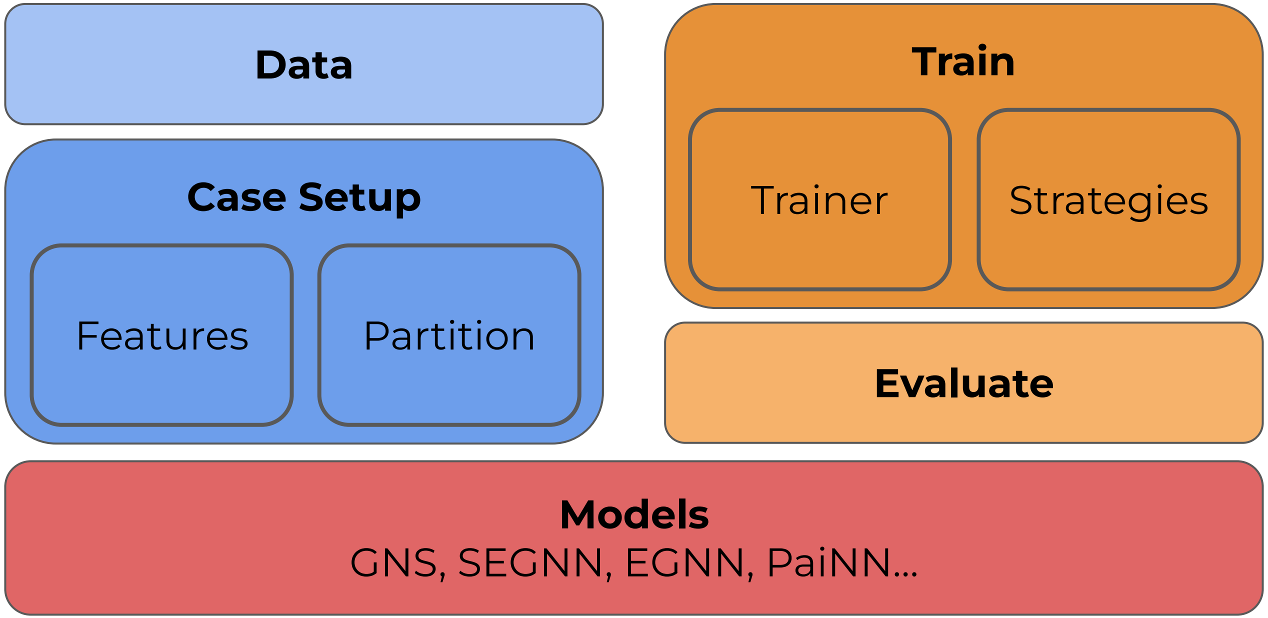
https://github.com/tumaer/lagrangebench

LagrangeBench



https://github.com/tumaer/lagrangebench

LagrangeBench



https://github.com/tumaer/lagrangebench

Lagrangel

Fea

tegies

https://github.com/tumaer/lagrar

Experiments

MSE$_p$ - position error

E$_{kin}$ - physical behavior

Sinkhorn - distribution

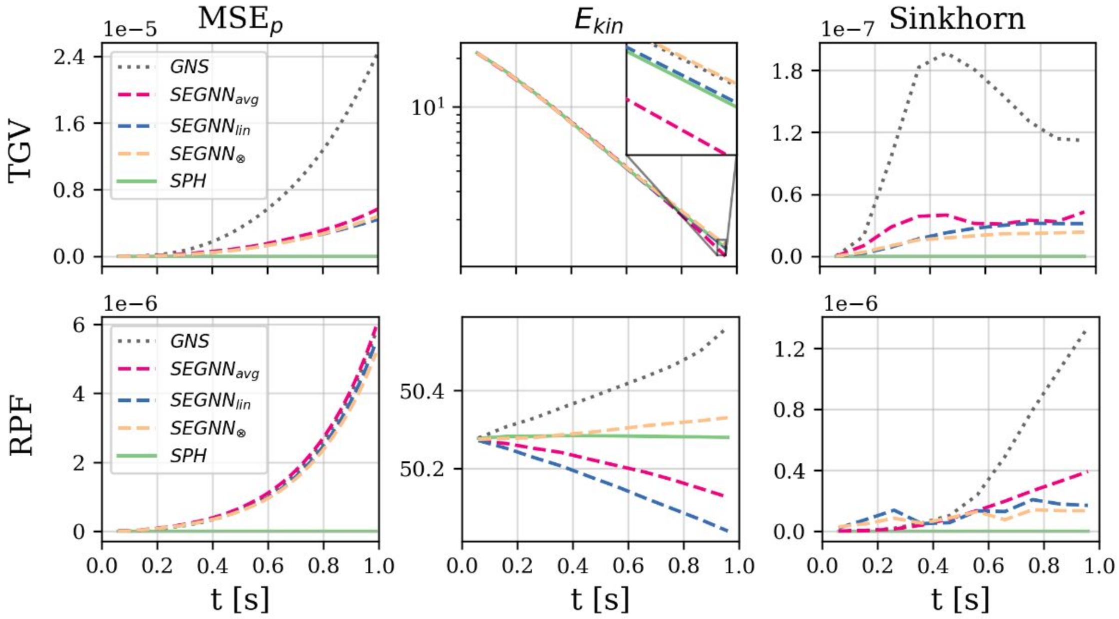


Fig. 3: Evolution of performance measures over time on the Taylor-Green vortex (top) and reverse Poiseuille flow (bottom).

Code:
https://github.com/tumaer/lagrangebench
https://github.com/tumaer/sph-hae

Experiments

Model	Time [ms]*	Taylor-Green vortex			reverse Poiseuille flow		
		MSE_p	$MSE_{E_{kin}}$	$\overline{\text{Sinkhorn}}$	MSE_p	$MSE_{E_{kin}}$	$\overline{\text{Sinkhorn}}$
GNS	35	6.7e-6	7.1e-3	1.2e-7	1.4e-6	2.2e-2	4.1e-7
SEGNN_{avg}	150	1.6e-6	8.4e-3	2.9e-8	1.4e-6	8.2e-3	1.4e-7
SEGNN_{lin}	150	**1.4e-6**	**3.1e-4**	2.0e-8	**1.3e-6**	2.0e-2	1.2e-7
$\text{SEGNN}_{\otimes}$	152	**1.4e-6**	1.9e-3	**1.6e-8**	**1.3e-6**	**9.4e-4**	**9.1e-8**

* Reference SPH solver takes 100ms to iterate 10 steps.

Code:
https://github.com/tumaer/lagrangebench
https://github.com/tumaer/sph-hae

Questions?

@ArturToshev
Email: artur.toshev@tum.de