Towards learning a Lattice Boltzmann collisional operator

SMFT 2022, Bari

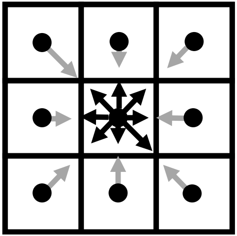
A. Gabbana[1], A. Corbetta [1], V. Gyrya[2], D. Livescu[2],
J. Prins[1] , F. Toschi[1]

[1] Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
[2] Los Alamos National Laboratory, NM 87545 Los Alamos, USA

Bari - December 19, 2022

Collisional operator for the Lattice Boltzmann Method

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t) - f_i(\boldsymbol{x}, t) = \Omega(f_i)$$

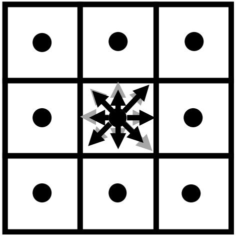


- Lattice Boltzmann Method: Computational fluid dynamics method for solving complex fluid flows.
- Second order approximation of the **Navier-Stokes** equations.
- A set of virtual particles called **populations** arranged at the edges of a discrete regular mesh.
- Particles only have a finite number of velocity directions:

$$\boldsymbol{v} \rightarrow \{\boldsymbol{c}_i, i = 1, \ldots, q\}$$

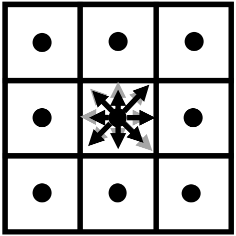
Learn novel collisional operators from data?

$$f_i^{\rm post} = \color{blue}{\Omega_{\rm NN}}(f_i^{\rm pre})$$



Learn novel collisional operators from data?

$$f_i^{\mathrm{post}} = \Omega_{\mathrm{NN}}(f_i^{\mathrm{pre}})$$



Can we learn the BGK model from data?

Learning problem:

- Map 9 pre-collisional into 9 post-collisional distribution
- What accuracy can we get?
- Can we preserve physical properties? (symmetries, conservation laws)

Learning the BGK collisional operator from data

1. Define training set $[\boldsymbol{f}^{\mathrm{pre}}, \boldsymbol{f}^{\mathrm{post}}]$
2. Define structure and hyper-parameters of the neural network
3. Train the network by minimizing a chosen error metric
4. Evaluate the accuracy

1. Define the training set

We consider N pairs $[\boldsymbol{f}^{\text{pre}}, \boldsymbol{f}^{\text{post}}]$, where

$$f_i^{\text{post}} = f_i^{\text{pre}} + \frac{1}{\tau}\left(f_i^{\text{eq}}(\rho, \boldsymbol{u}) - f_i^{\text{pre}}\right)$$

and the pre-collisional distributions are obtained as follows:

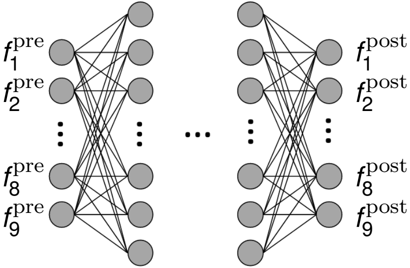
- Define a set $\{(\rho, u_x, u_y)_j,\ j = 1, 2, \dots N\}$, with ρ, u_x, u_y randomly sampled (e.g. from a uniform distribution)
- Randomly sample { $\boldsymbol{f}_j^{\text{neq}}$, j = 1, 2, . . . N } such that

$$\sum_{i=1}^{9}(f_i^{\text{post}} - f_i^{\text{pre}}) = 0$$

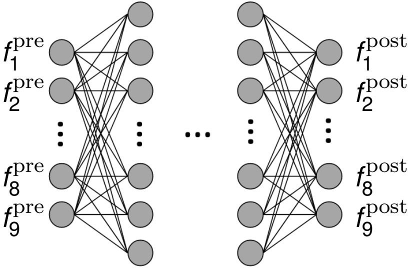
$$\sum_{i=1}^{9}(f_i^{\text{post}} - f_i^{\text{pre}})\xi_i = \boldsymbol{0}.$$

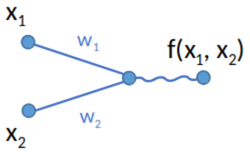
- $\boldsymbol{f}^{\text{pre}} = \boldsymbol{f}^{\text{eq}}(\rho, \boldsymbol{u}) + \boldsymbol{f}^{\text{neq}}$

2. Define the neural network structure

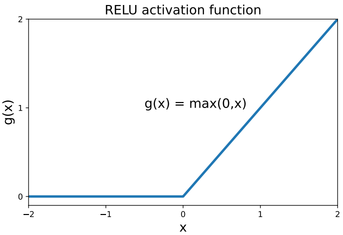


2. Define the neural network structure

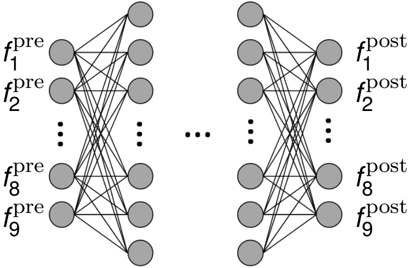




$$f(x_1, x_2) = g(w_1 x_1 + w_2 x_2)$$

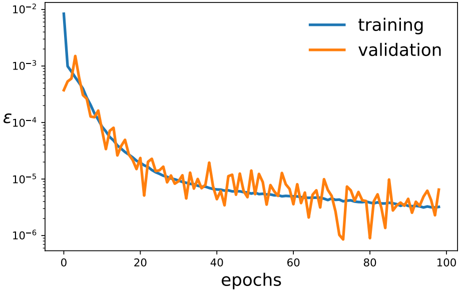


2. Define the neural network structure



- 2 Hidden Dense layers, 50 neurons each
- RELU activation function
- Last layer linear activation

3. Training the neural network

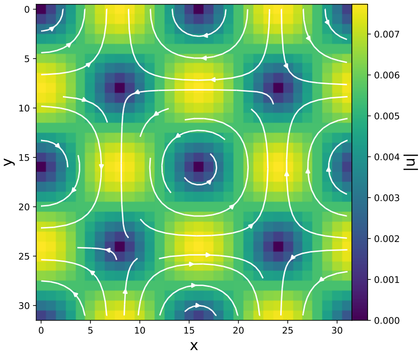


- ADAM optimizer
- 100 training epochs
- Batch size: 32
- Error Metric:

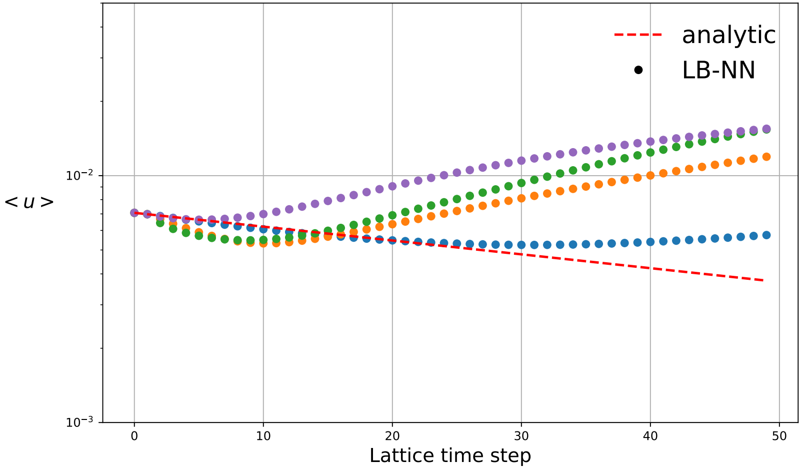
$$\epsilon = \frac{1}{N} \sum_{s=1}^{N} \sum_{i=1}^{9} \left(\frac{\Omega_{\mathrm{NN}}(f_{i,s}^{\mathrm{pre}}) - \Omega_{\mathrm{BGK}}(f_{i,s}^{\mathrm{pre}})}{\Omega_{\mathrm{BGK}}(f_{i,s}^{\mathrm{pre}})} \right)^2$$

4. Accuracy Evaluation

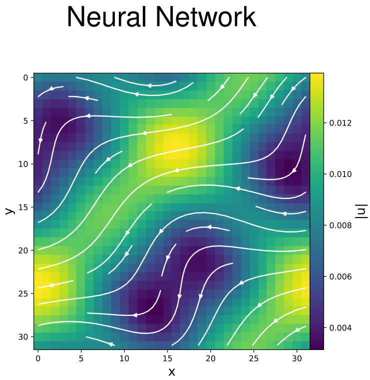
- We perform a simulation where we replace the BGK operator with the Neural Network
- Simulation of a decaying Taylor-Green vortex flow
- Grid size: 32x32
- $\tau = 1$

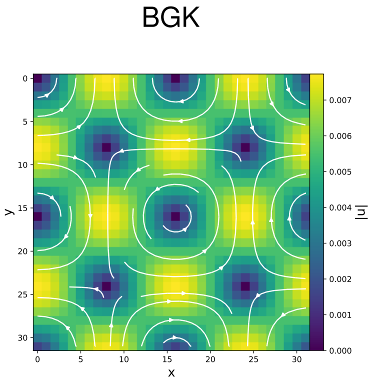


4. Accuracy Evaluation



4. Accuracy Evaluation





Snapshot of the velocity field after 10 iterations

Physics-Informed Machine Learnin

How can we improve the results? In what follows we provide recipes to embed the following physical properties in the Neural Network:

- Scale invariance
- Positivity
- Symmetries
- Conservation laws

Scale Invariance and Positivity

- We train the network over populations scaled by their density:

$$\phi^{\text{post}} = \boldsymbol{\Omega}_{\text{NN}}(\phi^{\text{pre}}).$$

with $\phi^{\text{pre}} \equiv \boldsymbol{f}^{\text{pre}}/\rho$ and $\phi^{\text{post}} \equiv \boldsymbol{f}^{\text{post}}/\rho$.

- We can then define a scale invariant collision operator as

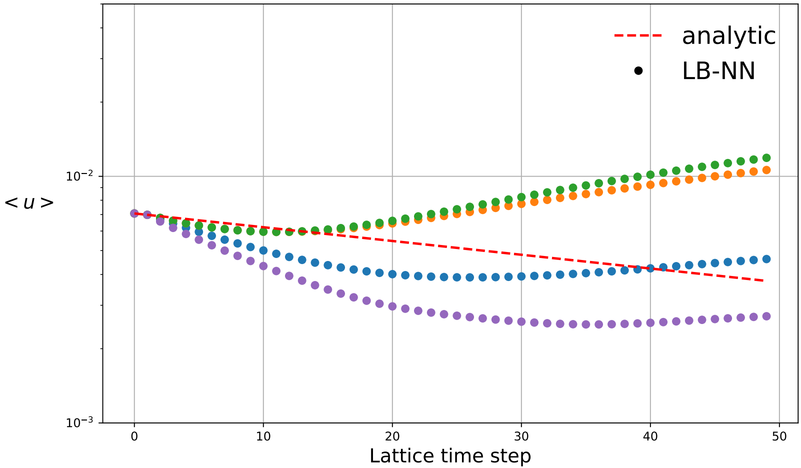
$$\boldsymbol{f}^{\text{post}} = \boldsymbol{\Omega}(\boldsymbol{f}^{\text{pre}}) \equiv \rho \boldsymbol{\Omega}^{\text{NN}}(\boldsymbol{f}^{\text{pre}}/\rho).$$

- To ensure positivity, we make use of the softmax output activation function:

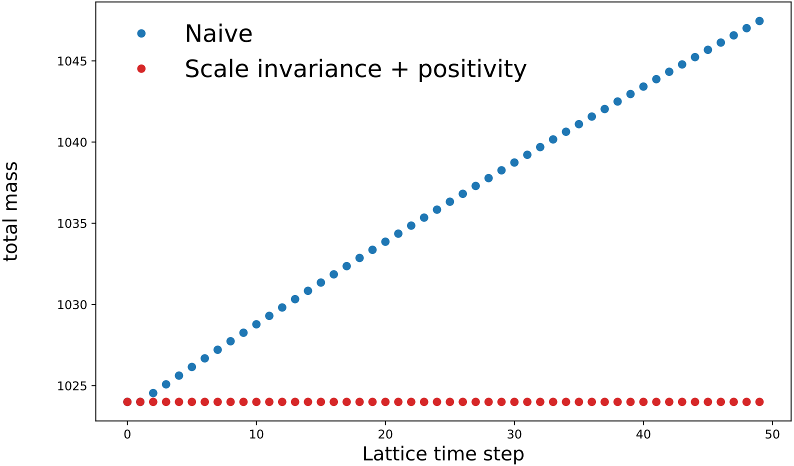
$$\phi_i^{\text{post}} = \text{softmax}(\hat{\phi}_{\text{i}}^{\text{post}}) \equiv \frac{e^{\hat{\phi}_{\text{i}}^{\text{post}}}}{\sum_{\text{i}} e^{\hat{\phi}_{\text{i}}^{\text{post}}}}.$$

Note: $\sum_i \phi_i^{\text{post}} = 1$.

Scale Invariance: Results



Scale Invariance: Results

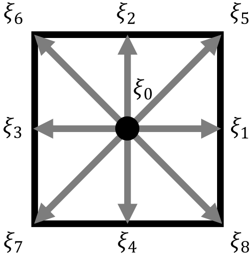


Symmetries

The symmetry group of the D2Q9 stencil is given by the 4th order dihedral symmetry group D_8:

$$D_8 = \{\boldsymbol{I}, \boldsymbol{R}, \boldsymbol{R}^2, \boldsymbol{R}^3, \boldsymbol{M}, \boldsymbol{RM}, \boldsymbol{R}^2\boldsymbol{M}, \boldsymbol{R}^3\boldsymbol{M}\}.$$

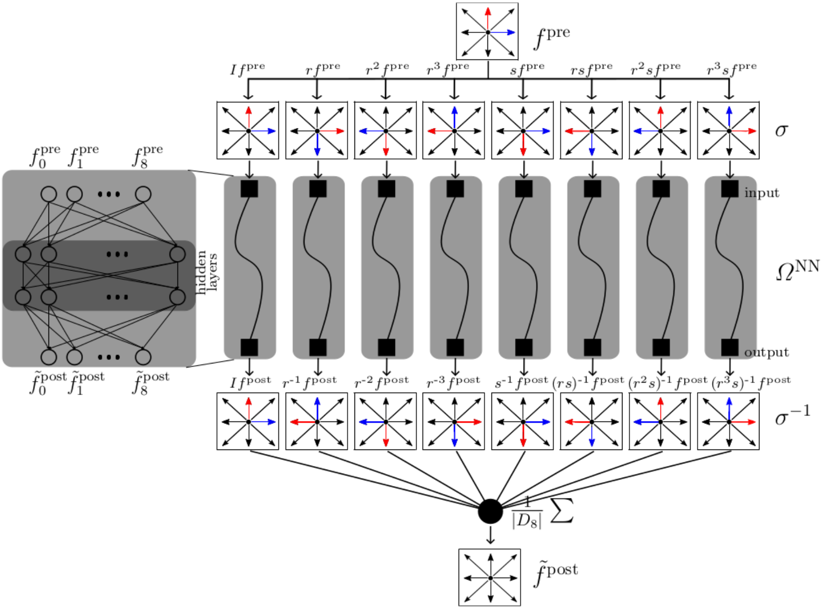
- $\boldsymbol{I}$ is the identity
- $\boldsymbol{R}$ is a 90 degrees rotation matrix
- $\boldsymbol{M}$ is a reflection matrix (e.g. along the x-axis)



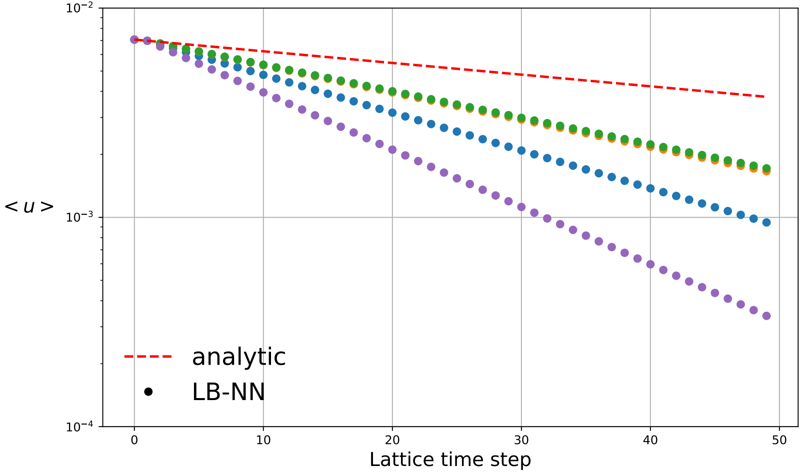
It is possible to define a collision operator Ω that preserves the symmetries of D_8 as

$$\boldsymbol{f}^{\mathrm{post}} = \mathcal{G}\Omega_{\mathrm{NN}}(\boldsymbol{f}^{\mathrm{pre}}) = \frac{1}{|D_8|}\sum_{\forall\sigma\in D_8} \sigma^{-1}\Omega_{\mathrm{NN}}(\sigma\boldsymbol{f}^{\mathrm{pre}}),$$

Symmetries: Architecture

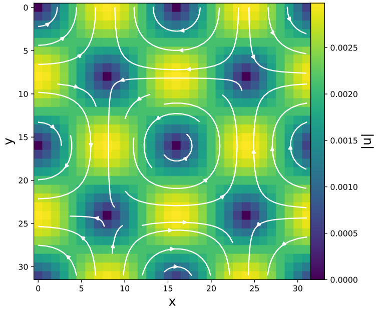


Symmetries: Results

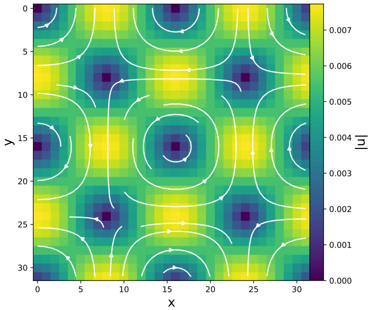


Symmetries: Results

Neural Network



BGK



Snapshot of the velocity field after 50 iterations

Conservation Laws

It is possible to define a collision operator Ω that preserves the conservation of mass and momentum:

$$\boldsymbol{f}^{\mathrm{post}} = \Omega(\boldsymbol{f}^{\mathrm{pre}}) \equiv \mathcal{A}\Omega_{\mathrm{NN}}(\boldsymbol{f}^{\mathrm{pre}})$$

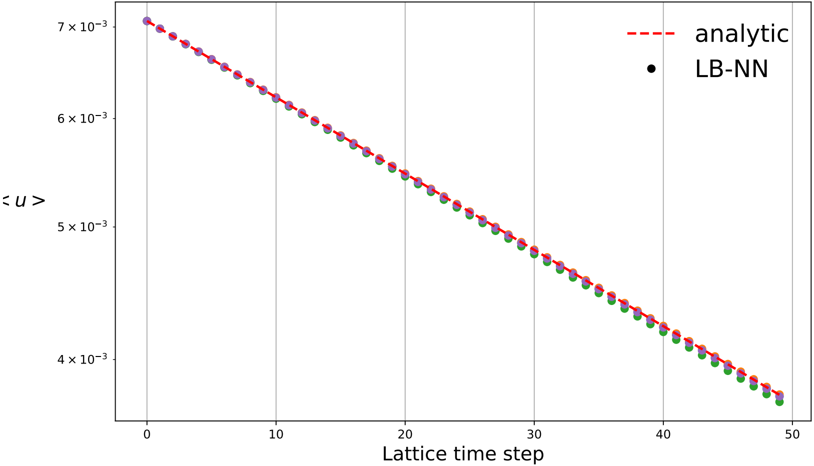
s.t.

$$\sum_{i=1}^{9}(f_i^{\mathrm{post}} - f_i^{\mathrm{pre}}) = 0$$

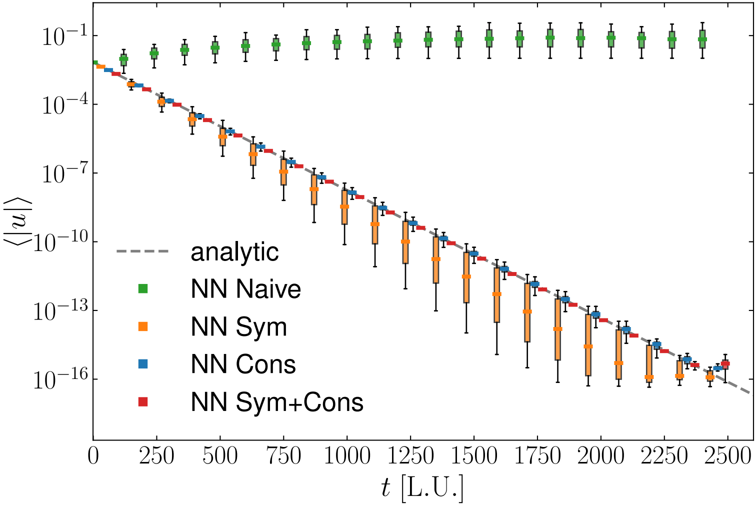
$$\sum_{i=1}^{9}(f_i^{\mathrm{post}} - f_i^{\mathrm{pre}})\boldsymbol{\xi}_i = \mathbf{0}.$$

One possible approach: algebraic reconstruction of 3 populations in order to enforce conservation laws.

Conservation Laws: Results



Summary



10^{-1}
10^{-4}
10^{-7}
10^{-10}
10^{-13}
10^{-16}
$\langle|u|\rangle$
analytic
NN Naive
NN Sym
NN Cons
NN Sym+Cons
0
250
500
750
1000
1250
1500
1750
2000
2250
2500
t [L.U.]

Outlook

- Design of a Neural Network capable of learning the BGK collisional operator from data
- Naive approach does not allow to follow the dynamics if not for just few iterations
- Embedding physics in the Neural Network, in particular symmetries and conservation laws, crucial in order to achieve results comparable to that of LBGK.
- For the future we plan to attempt learning new collisional operators / corrections to the BGK in order to extend the applicability of LB.

arXiv:2212.06124

Towards learning a Lattice Boltzmann collisional operator

SMFT 2022, Bari

A. Gabbana[1], A. Corbetta [1], V. Gyrya[2], D. Livescu[2], J. Prins[1] , F. Toschi[1]

[1] Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands
[2] Los Alamos National Laboratory, NM 87545 Los Alamos, USA

Bari - December 19, 2022

Soft Constraints

$$\mathcal{L} = \mathrm{MSRE} + \alpha \frac{\|\tilde{\boldsymbol{u}} - \boldsymbol{u}\|}{\|1 + \boldsymbol{u}\|}$$