New Closure for the Vlasov-Poisson Equations using Machine Learning

L. Bois[1,2] E. Franck[1,2] L. Navoret[1,2] V. Vigon[1]

[1] Institut de Recherche Mathématique Avancée, UMR 7501,
Université de Strasbourg et CNRS, 7 rue René Descartes,
67000 Strasbourg, France

[2] INRIA Nancy-Grand Est, TONUS Project, Strasbourg, France

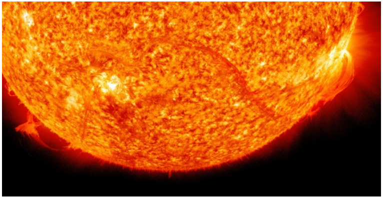
November 2020

Outline

Table of Contents

Plasmas

- State of the matter, similar to gas, except that the particles are electrically charged.
- Can be found in very hot environments like the sun



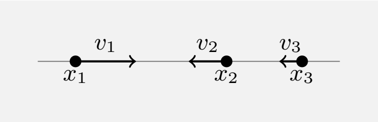
- Studied for nuclear fusion

- In this work: 1D plasma with periodic boundary conditions.

Table of Contents

N-Body Model

Follows the position and velocity of all $N \simeq 10^{10}$ particles as functions of the time t.



For all i in $\{1, ..., N\}$,

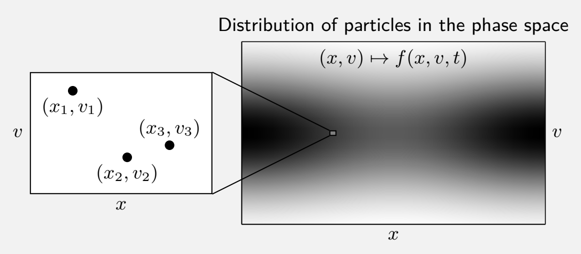
$$x_i'(t) = v_i(t)$$ transport at velocity v_i

$$v_i'(t) = \sum_{j=1}^{N} F_{j\to i}(t)$$ electrical interactions and collisions

Not suited to numerical simulation since it results in billions of coupled ODEs.

Kinetic Model

Describes the statistical distribution of the particles in the phase space.
x and v become variables of the unknown $f(x, v, t)$.



Vlasov equation:

$$(V) \qquad \partial_t f + \underbrace{v\,\partial_x f}_{\text{Transport}} - \underbrace{E\,\partial_v f}_{\text{Electrical}} = \underbrace{\frac{1}{\varepsilon}(M(f) - f)}_{\text{Collisions}}$$

A single PDE (+ Poisson equation on E), but still expensive to solve numerically in 3D because it requires the discretization of the 6D phase space.

Fluid Model

Gets rid of the variable v by considering 3 statistical indicators instead of the whole distribution of velocities $f(x, ., t)$ for all x:

- The total number of particles (density)

$$\rho(x,t) = \int f(x,v,t)\,dv$$

- The mean velocity

$$u(x,t) = \frac{1}{\rho(x,t)} \int v f(x,v,t)\,dv$$

- The variance (temperature)

$$T(x,t) = \frac{1}{\rho(x,t)} \int \big(v - u(x,t)\big)^2 f(x,v,t)\,dv$$

Fluid Model

From the Vlasov equation can be derived the following system of equations on ρ, u and T:

$$\partial_t \rho + \partial_x(\rho u) = 0$$
$$\partial_t(\rho u) + \partial_x(\rho u^2 + \rho T) = -E\rho$$
$$\partial_t w + \partial_x(wu + \rho u T + q) = -E\rho u,$$

where $w = \frac{1}{2}(\rho u^2 + \rho T)$ is the kinetic energy, and q the heat flux.

Reducing the whole distribution to its first three moments comes with a price: the above system is not closed, and requires the moment of order 3

$$q(x,t) = \frac{1}{2}\int \big(v - u(x,t)\big)^3 f(x,v,t)\, dv.$$

Table of Contents

Classical Approach

In general, $q(x,t)$ cannot be expressed as a function of ρ, u and T.

Classical approach: find a relevant approximation on the underlying distribution f, that allows to compute q from ρ, u and T.

Collisions

$$(V) \qquad \partial_t f + v\,\partial_x f - E\,\partial_v f = \underbrace{\frac{1}{\varepsilon}(M(f) - f)}_{\text{BGK operator}}$$

BGK collision operator (Bhatnagar, Gross and Krook):

- Knudsen number ε: mean free path between two collisions (fixed)
- Maxwellian $M(f)$: state of equilibrium
- When $\frac{1}{\varepsilon}$ goes up, f goes towards $M(f)$.

Maxwellian $M(f)$:

$$M(f)(x,v,t) = \frac{\rho(x,t)}{\sqrt{2\pi T(x,t)}} e^{-\frac{(v-u(x,t))^2}{2T(x,t)}}$$

- Same density ρ, mean velocity u and temperature T as f
- Gaussian profile in velocity (characterized by ρ, u and T)

Euler Closure

$$(V) \qquad \partial_t f + \ v\, \partial_x f - \ E\, \partial_v f = \frac{1}{\varepsilon}(M(f) - f)$$

When the Knudsen number ε tends to 0, then the solution f satisfies

$$f(x, v, t) = M(f)(x, v, t) + O(\varepsilon)$$

Thus, when $\varepsilon << 1$, the heat flux q of f can be approximated by the heat flux $\hat{q}$ of $M(f)$:

$$\hat{q}(x, t) = 0.$$

Other Closures

In a similar way, the second order approximation

$$f(x,v,t) = M(f)(x,v,t) + g\varepsilon + O(\varepsilon^2)$$

leads to the Navier-Stokes closure:

$$\hat{q}(x,t) = \frac{3}{2}\varepsilon\rho(x,t)T(x,t)\partial_x T(x,t).$$

NB: Other approximations can lead to non-local closures (e.g Hammett-Perkins closure).

Learned Closure

What if there is no known approximation both relevant and allowing to compute q from the ρ, u and T?

Example: Knudsen number $\varepsilon \simeq 0.1$ or above

- Navier-Stokes closure lacks accuracy
- Yet a good approximation of the heat flux might exist.

Idea: Training a neural network to be used as closure.

$$\varepsilon, \rho, u, T \longrightarrow \boxed{\text{N.N}} \longrightarrow \hat{q}$$

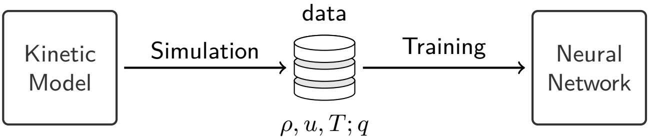
Points of interest:

- Data generation
- Network architecture
- Data processing

Table of Contents

Training Scheme

Off-line phase



On-line phase

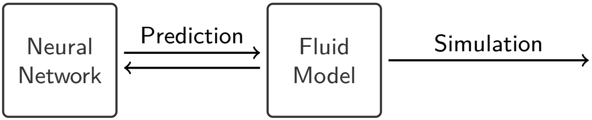
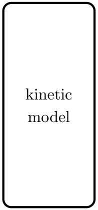


Table of Contents

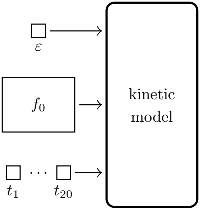
Data generation

The data is generated from the kinetic model.



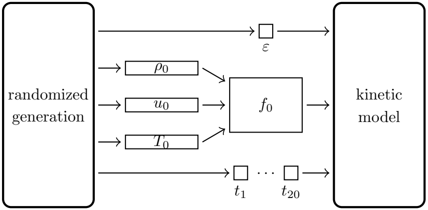
Data generation

It takes a Knudsen number ε, an initial condition f_0, and 20 times t_1, ..., t_{20}.



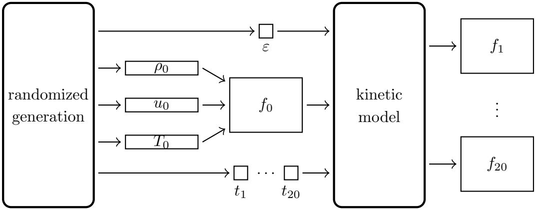
Data generation

These quantities are generated randomly. In particular, $\varepsilon \in [0.01, 1]$.



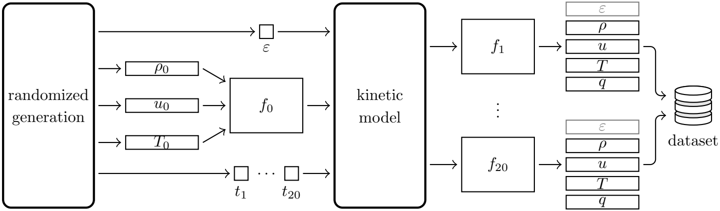
Data generation

The model outputs the distributions f_1, ..., f_{20} at times t_1, ..., t_{20} respectively.

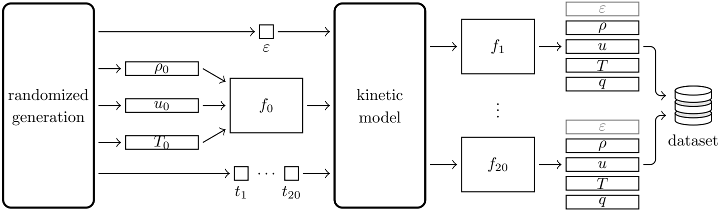


Data generation

Finally ρ, u, T and q are computed and stored in the dataset, along with ε.



Data generation



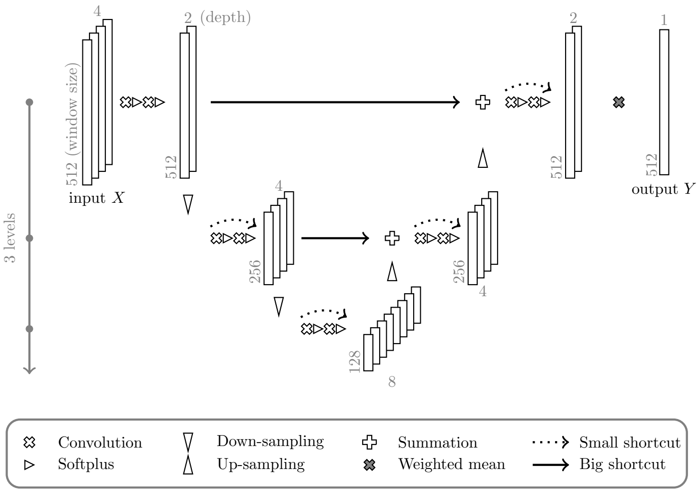
Two datasets with $500 \times 20 = 10\,000$ *entries are generated this way: one for training, and one for testing.*

Table of Contents

Architecture

- Non-local closure: whole functions in, whole function out
- Functions as signals or 1D images: convolutional network
- Multiscale analysis: V-Net like architecture

V-Net architecture



Hyper-parameters

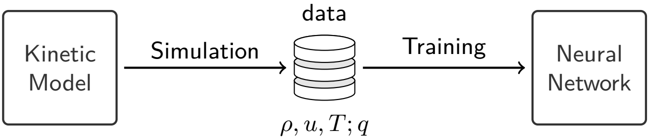
Hyper-parameter	Value
length of the input	512
number of levels (ℓ)	5
depth (d)	4
size of the kernels (p)	11
activation function	softplus

Total parameters: $161\,937$

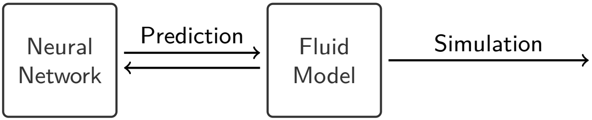
Table of Contents

Training Scheme

Off-line phase

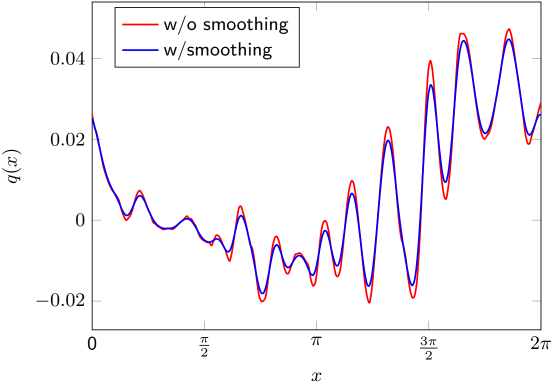


On-line phase



Smoothing

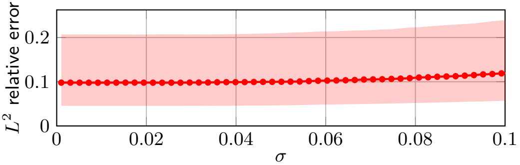
- Smoothing of the output is used to ensure stability.
- It uses a convolution with a gaussian kernel of std σ.
- The higher the standard deviation, the less accurate the prediction, but the more stable the numerical scheme.



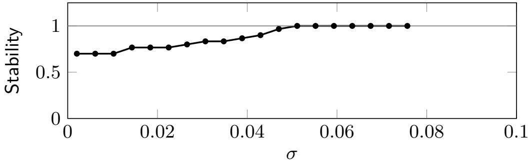
Smoothing

Based on the following results, we chose $\sigma \simeq 0.06$.

- Accuracy of the predictions on the test dataset depending on σ:



- Proportion of simulations reaching final time depending on σ:



Resampling

Resampling is used to adapt to different resolutions is space.

Relative error at final time depending on the number of cells

- Option 1: without resampling
- Option 2: with resampling

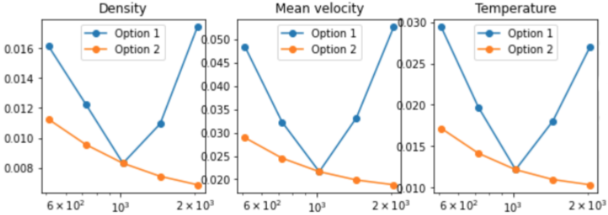
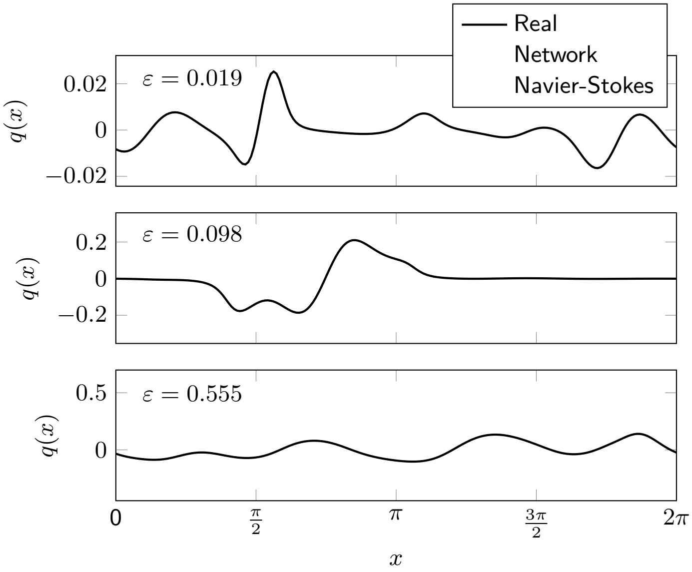


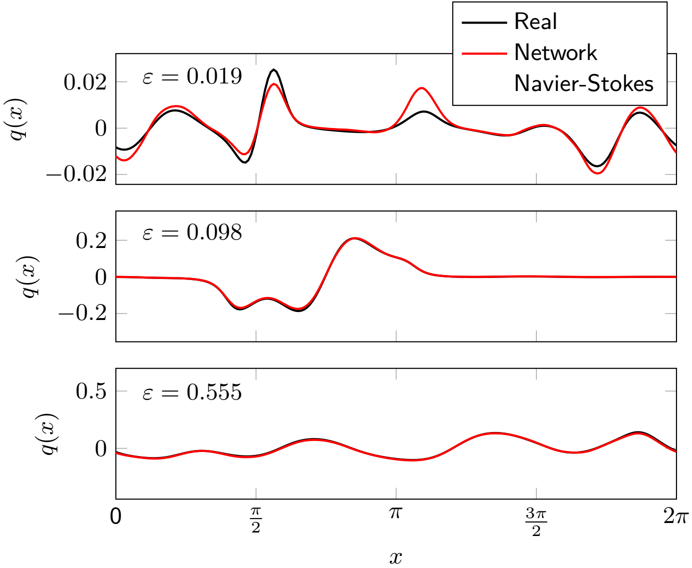
Table of Contents

Table of Contents

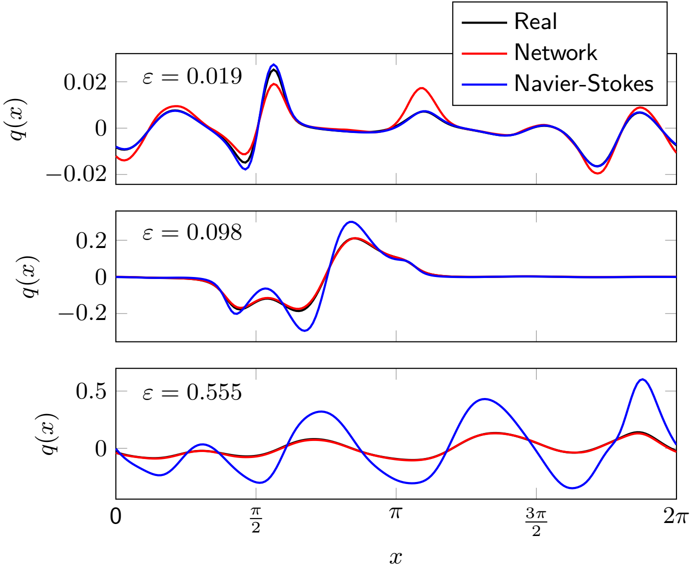
Examples from the test dataset



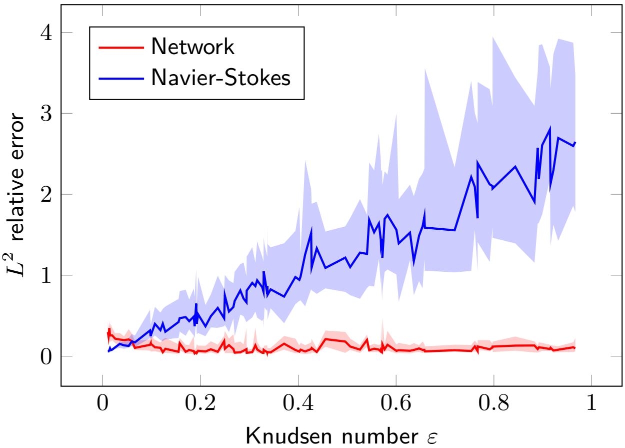
Examples from the test dataset



Examples from the test dataset



Influence of ε



As expected the accuracy of the Navier-Stokes approximations greatly depends on ε. But it does not seem to be the case for the neural network predictions.

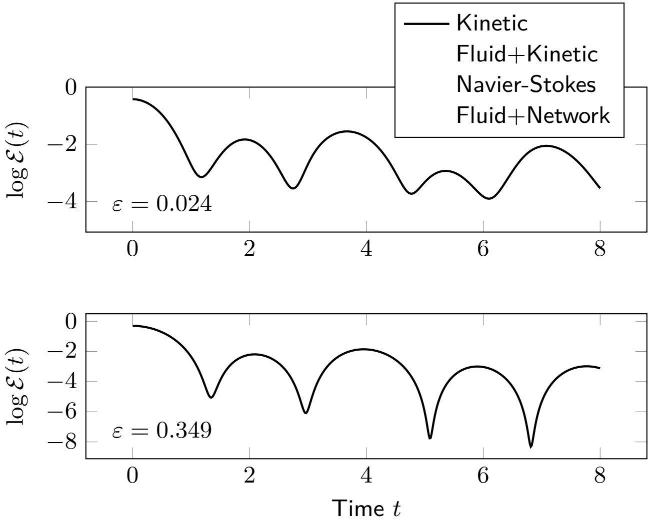
Table of Contents

The models

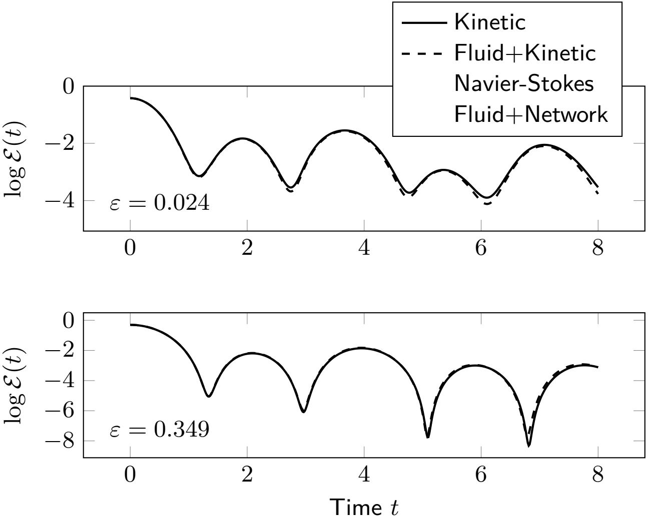
We compare the *electric energy* $\mathcal{E}(t) = \int E(x,t)^2 \, dx$ of the following models:

solid black line	Kinetic
dashed black line	Fluid+Kinetic: $\hat{q} = q$
dashed red line	Fluid+Network: $\hat{q} = C_\theta(\varepsilon, \rho, u, T)$
dashed blue line	Navier-Stokes: $\hat{q} = -\frac{3}{2}\varepsilon\rho T \partial_x T$

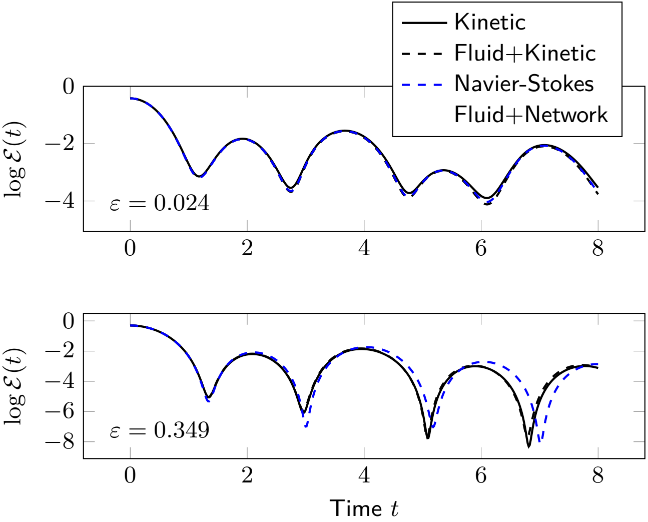
Examples (1/2)


Kinetic
Fluid+Kinetic
Navier-Stokes
Fluid+Network
$\log \mathcal{E}(t)$
$\varepsilon = 0.024$
0
−2
−4
0
2
4
6
8
$\log \mathcal{E}(t)$
$\varepsilon = 0.349$
0
−2
−4
−6
−8
0
2
4
6
8
Time t

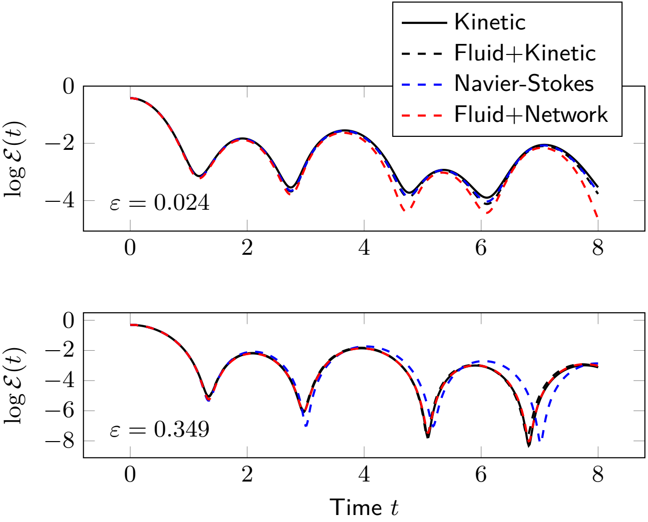
Examples (1/2)


Kinetic
Fluid+Kinetic
Navier-Stokes
Fluid+Network
$\log \mathcal{E}(t)$
$\varepsilon = 0.024$
$\varepsilon = 0.349$
0
−2
−4
−6
−8
0
2
4
6
8
Time t

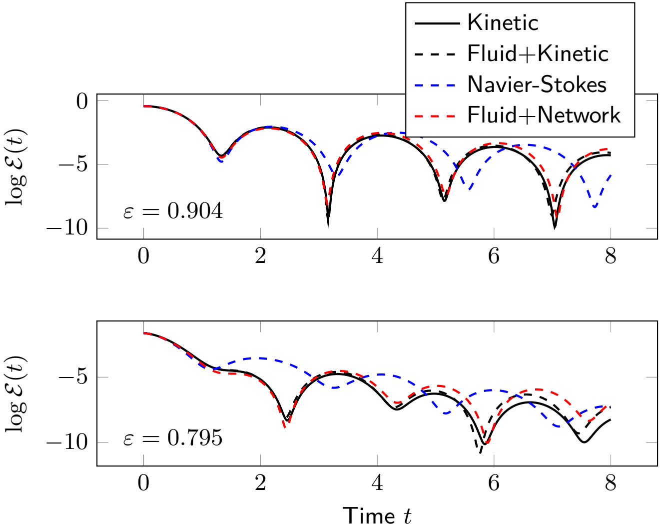
Examples (1/2)


Kinetic
Fluid+Kinetic
Navier-Stokes
Fluid+Network
$\log \mathcal{E}(t)$
$\varepsilon = 0.024$
0
−2
−4
0
2
4
6
8
$\log \mathcal{E}(t)$
$\varepsilon = 0.349$
0
−2
−4
−6
−8
0
2
4
6
8
Time t

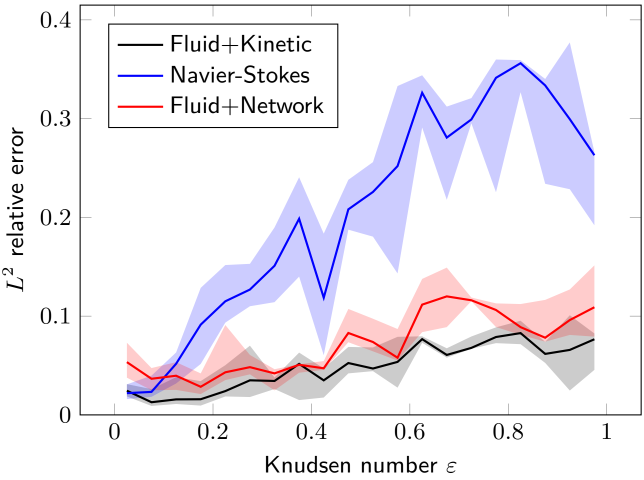
Examples (1/2)


Kinetic
Fluid+Kinetic
Navier-Stokes
Fluid+Network
$\log \mathcal{E}(t)$
$\varepsilon = 0.024$
$\varepsilon = 0.349$
0
−2
−4
−6
−8
0
2
4
6
8
Time t

Examples (2/2)

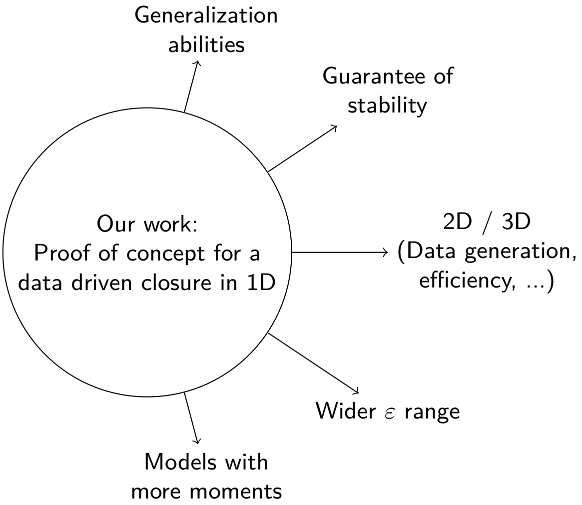

Kinetic
Fluid+Kinetic
Navier-Stokes
Fluid+Network
$\log \mathcal{E}(t)$
0
−5
−10
$\varepsilon = 0.904$
0
2
4
6
8
$\log \mathcal{E}(t)$
−5
−10
$\varepsilon = 0.795$
0
2
4
6
8
Time t

Influence of ε



The error of the "Fluid+Network" model seems to increase in a similar way than that of the "Fluid+Kinetic" model.

Conclusion


Generalization abilities
Guarantee of stability
Our work:
Proof of concept for a data driven closure in 1D
2D / 3D (Data generation, efficiency, ...)
Wider ε range
Models with more moments

Extra

Numerical methods

Vlasov-Poisson

1. Electric field:

$$E = -\partial_x \phi, \quad \partial_{xx}\phi = \rho - \int_0^L \rho \, dx$$

Finite difference method.

2. Transport:

$$\partial_t f + v\partial_x f - E\partial_v f = 0$$

Discretization in velocity:

$$\frac{\mathbf{f}^{n+1} - \mathbf{f}^n}{\Delta t} + \Lambda \partial_x \mathbf{f}^n + EB(\mathbf{f}^n) = 0$$

Finite volume method with upwind flux.

3. Collision operator:

$$\partial_t f = \frac{1}{\varepsilon}(M(f) - f)$$

Implicit scheme.

Numerical methods

Euler

1. Electric field:

$$E = -\partial_x \phi, \quad \partial_{xx}\phi = \rho - \int_0^L \rho \, dx$$

   Finite difference method.

2. Fluid equations:

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) = -E\mathbf{H}(\mathbf{U}),$$

   $\mathbf{U} = (\rho, \rho u, w)$, $\mathbf{F}(\mathbf{U}) = (\rho u, \rho u^2 + p, wu + pu + q)$, $\mathbf{H}(\mathbf{U}) = (0, \rho, \rho u)$
   Finite volume method with local Lax-Friedrichs numerical flux and explicit scheme in time.

$$\frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \frac{\mathbf{F}(\mathbf{U})_{i+\frac{1}{2}}^n - \mathbf{F}(\mathbf{U})_{i-\frac{1}{2}}^n}{\Delta x} = -E_i^n \mathbf{H}(\mathbf{U})_i^n$$

Numerical methods

Navier-Stokes

1. Electric field:

$$E = -\partial_x \phi, \quad \partial_{xx}\phi = \rho - \int_0^L \rho \, dx$$

   Finite difference method.

2. First two fluid equations: same as Euler

3. Third fluid equation:

$$\partial_t w + \partial_x(wu + pu) - \frac{3}{2}\varepsilon \partial_x(p\partial_x T) = -E\rho u$$

   with $w = \frac{1}{2}\rho u^2 + \frac{1}{2}\rho T$
   Finite difference approximation for $\partial_x(p\partial_x T)$ and implicit scheme in time.

Time efficiency

Mean time for simulations up to $t = 8$ with $N_x = 512$ and $N_v = 101$:

Kinetic	70 sec
Fluid+Kinetic	78 sec
Fluid+Network	74 sec
Navier-Stokes	3 sec

Complexity in different dimensions:

V-Net 1D	$O(2^\ell d^2 p N_x)$
V-Net 2D	$O(\ell d^2 p^2 N_x^2)$
V-Net 3D	$O(d^2 p^3 N_x^3)$
Kinetic mD	$O(N_v^m N_x^m)$

Hyper-parameters