Hybrid ML-PDE reduced modeling for Vlasov Poisson in plasma physics

Emmanuel Franck[*], C. Courtès[*], Leo Bois[+], L. Navoret[*], V. Vigon[*], Y. Nasseri[*]

October 1, 2024
THALES JPAL workshop on "Analytical Model-Informed Neural Network", Paris

[*] MACARON project-team, Université de Strasbourg, CNRS, Inria, IRMA, France

[+] MIMESIS project-team, Université de Strasbourg, CNRS, Inria, France

Outline

Vlasov Poisson equation and plasma modeling

Model hierarchy for particles/body dynamic

- We want to simulate a very large number of interacting particles or bodies.
- We consider N particles of position $\mathbf{x}(t)$ and impulsion $\mathbf{p}(t)$ given by the Fundamental Principles of Dynamics:

$$\begin{cases} \frac{d}{dt}\mathbf{x}_i(t) = \mathbf{p}_i(t) \\ \frac{d}{dt}\mathbf{p}_i(t) = -\frac{1}{N}\sum_{1\leqslant i,j\leqslant N} \nabla_{\mathbf{x}} V(\mathbf{x}_i,\mathbf{x}_j) \end{cases}$$

  with V a potential
- In practice the simulation for realistic N are intractable.

Kinetic equation

We define the empirical measure $f_N(t,\mathbf{x},\mathbf{p}) = \sum_{i=1}^{N} \delta(\mathbf{x}-\mathbf{x}_i(t))\delta(\mathbf{p}-\mathbf{p}_i(t))$. We can prove that $f_N(t,\mathbf{x},\mathbf{p}) \rightarrow_{N\rightarrow\infty} f(t,\mathbf{x},\mathbf{p})$, which satisfies

$$\partial_t f + \mathbf{p}\cdot\nabla_x f + F(t,\mathbf{x})\cdot\nabla_{\mathbf{p}} f = 0,$$

with

$$F(t,\mathbf{x}) = -\nabla_{\mathbf{x}}\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} V(\mathbf{x},\mathbf{y})f(\mathbf{x},\mathbf{y},\mathbf{p})d\mathbf{y}d\mathbf{p}.$$

Vlasov-Poisson equation

- Electrostatic potential for charged particle and gravitational potential for a body:

$$V(\mathbf{x},\mathbf{y})=\frac{q}{4\pi}\frac{1}{|\mathbf{x}-\mathbf{y}|},\qquad V(\mathbf{x},\mathbf{y})=\frac{-Gm}{|\mathbf{x}-\mathbf{y}|}$$

- We remark that

$$\phi(t,\mathbf{x})=\pm\int_{\mathbb{R}^d}\int_{\mathbb{R}^d}\frac{1}{|\mathbf{x}-\mathbf{y}|}f(t,\mathbf{y},\mathbf{p})d\mathbf{y}d\mathbf{p}\longleftrightarrow\pm\Delta\phi=\int_{\mathbb{R}^d}f(t,\mathbf{y},\mathbf{p})d\mathbf{p}$$

- Adding a binary collision process leads to the Vlasov-Poisson equation.

Final Vlasov-Poison equation

The Vlasov equation in the phase-space for $f(t,\mathbf{x},\mathbf{v})$ is given by

$$\partial_t f+\mathbf{v}\cdot\nabla_x f\pm\nabla\phi\cdot\nabla_{\mathbf{v}}f=\frac{1}{\epsilon}Q(f,f)$$

with ϵ free mean path and

$$-\Delta\phi=\int_{\mathbb{R}^d}f(t,\mathbf{x},\mathbf{v})d\mathbf{v}$$

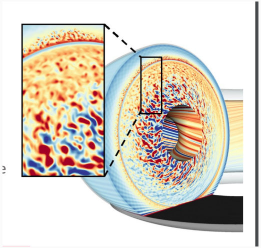
- This equation is central to simulate plasma turbulence in fusion reactor or galaxy dynamics in astrophysics.

Objective

Difficulties

The Vlasov Poisson is a multiscale PDE in 7-dimensional space (3 space + 3 velocity + 1 time). It generates huge simulations.

- In plasma fusion we simulate 5D approximations: Gyro-kinetic models.
- Example: Gysela code (V. Grandgirard)



1 simulation:
- **100 billion points** (5D mesh: 3D space + 2D velocity)
- **> 8 million hours** (20 days / 18 432 cores),
- **10 PBytes of data** (3 Tbytes saved)

Aim

Proposed reduced models in the weakly collisional regime ($\tau \in [0.01, 1]$) using deep learning.

Micro-Macro decomposition I

First objective

Understand the behavior of the solution in the different collision regimes.

- The collision term $Q(f,f)$ generates a dissipation toward an equilibrium called Maxwellian:

$$\mathcal{M}(\rho, \mathbf{u}, T, \mathbf{v}) = \frac{\rho}{(2\pi T)^{\frac{d}{2}}} e^{-\frac{|\mathbf{v}-\mathbf{u}|^2}{2T}}$$

with

$$\rho(t,\mathbf{x}) = \int_{\mathbb{R}^d} f d\mathbf{v}, \quad (\rho\mathbf{u})(t,\mathbf{x}) = \int_{\mathbb{R}^d} \mathbf{v} f d\mathbf{v}, \quad (\rho T)(t,\mathbf{x}) = \int_{\mathbb{R}^d} |\mathbf{v}-\mathbf{u}|^2 f d\mathbf{v}$$

Micro-Macro

The relaxation toward the Maxwellian is central to understand the dynamics. The Micro-Macro decomposition propose to write

$$f(t,\mathbf{x},\mathbf{v}) = \mathcal{M}(\rho, \mathbf{u}, T, \mathbf{v}) + g(t,\mathbf{x},\mathbf{v})$$

with the first three moments vanish. The second step consists in writing the dynamics on both terms $\mathcal{M}$ and g.

Micro-Macro decomposition II

- The decomposition consists in projecting $f(t,\mathbf{x},\mathbf{v})$ on the space

$$\mathcal{N}(L_Q) = \text{Span}\left\{\mathcal{M}, \mathbf{v}\mathcal{M}, |\mathbf{v}|^2\mathcal{M}\right\}$$

  and on its orthogonal complement.
- Defining $\Pi_{\mathcal{M}}$ the projector on the space $\mathcal{N}$ we can write a new system:

$$\begin{cases} \partial_t\rho + \nabla\cdot(\rho\mathbf{u}) = 0 \\ \partial_t\rho\mathbf{u} + \nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u} + \rho T I_d) + \nabla\cdot\hat{\Pi} = \rho\nabla\phi \\ \partial_t E + \nabla\cdot(E\mathbf{u} + \rho T\mathbf{u}) + \nabla\cdot\mathbf{Q} = \rho(\mathbf{u},\nabla\phi) \\ \partial_t g + (I-\Pi_M)((\mathbf{v}\cdot\nabla_{\mathbf{x}}g + (\nabla\phi,\nabla_{\mathbf{v}}g)) = \frac{1}{\epsilon}[-g - \tau(I-\Pi_M)((\mathbf{v}\cdot\nabla_{\mathbf{x}}\mathcal{M} + (\nabla\phi,\nabla_{\mathbf{v}}\mathcal{M}))] \\ -\Delta\phi = \rho \end{cases}$$

  with the total energy $E = \frac{1}{2}\rho|\mathbf{u}|^2 + \rho T$ and

$$\hat{\Pi} = \int_{\mathbb{R}^d}\mathbf{v}\otimes\mathbf{v}g d\mathbf{v}, \quad \mathbf{Q} = \int_{\mathbb{R}^d}\frac{1}{2}|\mathbf{v}|^2\mathbf{v}g d\mathbf{v}$$

- The equation on g is the only high-dimensional one.

Reduced models and closures

We can a close the system by forgetting the dynamics of g, and propose an approximation of the form:

$$\hat{\Pi} \approx \mathcal{C}_{\pi}(\rho, \mathbf{u}, T, \tau), \quad \mathbf{Q} \approx \mathcal{C}_{Q}(\rho, \mathbf{u}, T, \epsilon)$$

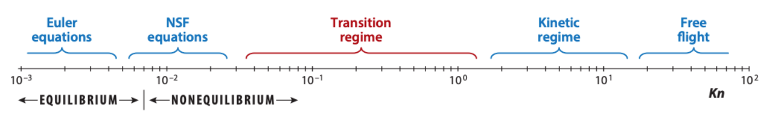
- Taking the **collision limit** ($\tau \to 0$) gives

$$g(t, \mathbf{x}, \mathbf{v}) = 0 + \epsilon g_1(t, \mathbf{x}, \mathbf{v}) + \epsilon^2 g_2(t, \mathbf{x}, \mathbf{v}) + O(\epsilon^3)$$

- The Euler closure choose the zero term in the expansion: $\hat{\Pi} = 0, \quad \mathbf{Q} = 0$
- The Navier-Stokes closure chooses the first term in the expansion:

$$\hat{\Pi} = \epsilon\left(\mu(T)[\nabla \mathbf{u} + (\nabla \mathbf{u})^T] + \mu_2(T)\nabla \cdot \mathbf{u} I_d\right), \quad \mathbf{Q} = -\epsilon\kappa(T)\nabla T$$

- Specific kinetic effects are obtain with nonlocal closure. Validity:



Learning closure

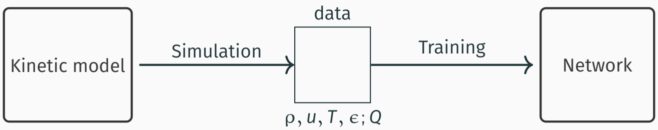
Principle

- We propose to learn a nonlocal closure for one dimensional space-velocity Vlasov equation with a simple collisional operator

$$Q(f,f) = \mathcal{M}(\rho, u, T, v) - f(t,x,v)$$

- In this case $\Pi = 0$ and the closure is only for $Q = \mathcal{C}(\rho(.), u(.), T(.), \epsilon)$
- Two part in the process:

Off line phase

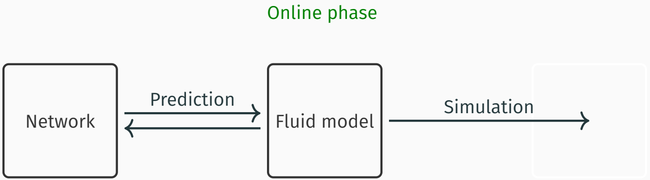


Principle

- We propose to learn a nonlocal closure for one dimensional space-velocity Vlasov equation with a simple collisional operator

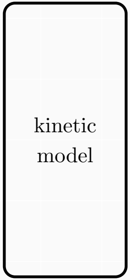
$$Q(f,f) = \mathcal{M}(\rho, u, T, v) - f(t, x, v)$$

- In this case $\Pi = 0$ and the closure is only for $Q = \mathcal{C}(\rho(.), u(.), T(.), \epsilon)$
- Two part in the process:



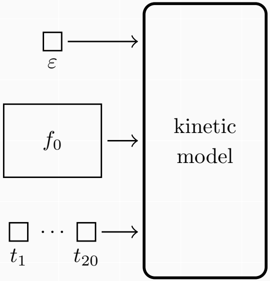
Data generation

- Process to generate data
- The data are obtained using the kinetic model.

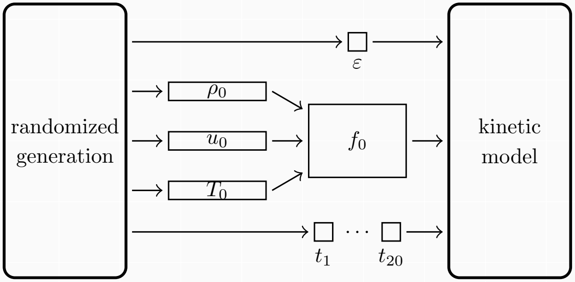


Data generation

- Process to generate data
- We take a Knudsen number ϵ, initial data f_0, and twenty time steps t_1, ..., t_{20}

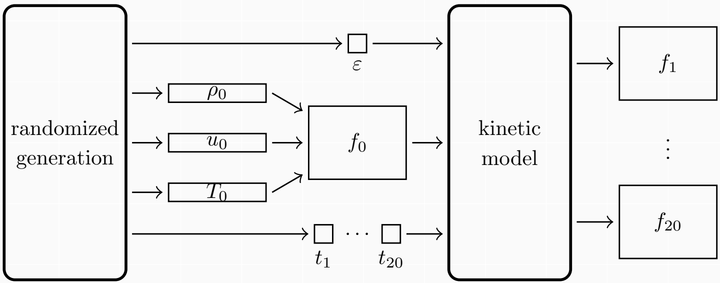


- Process to generate data
- Initial data are generated randomly with Fourier modes and ϵ uniformly sampled in $[0.01, 1]$.



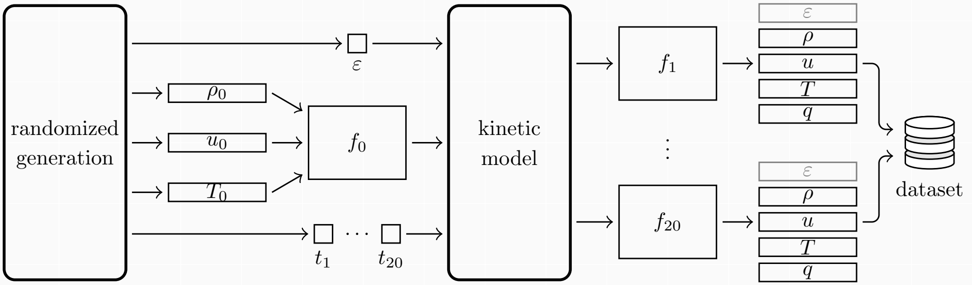
Data generation

- Process to generate data
- The numerical scheme gives $f_1, ..., f_{20}$ at time $t_1, ..., t_{20}$.



Data generation

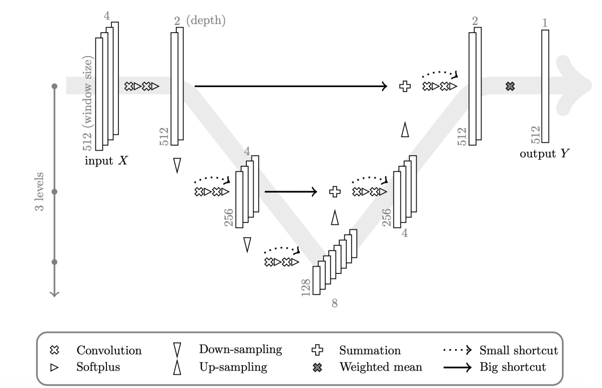
- Process to generate data
- The moments ρ, u, T and Q are computed and store ϵ.



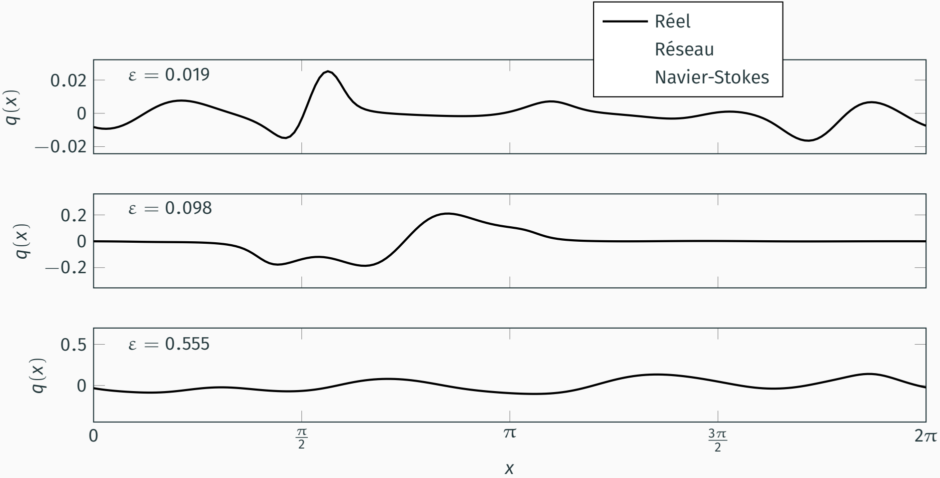
- Two data set ($500 \times 20 = 10\,000$) are generated: one for training, one for validation

Neural Network architecture

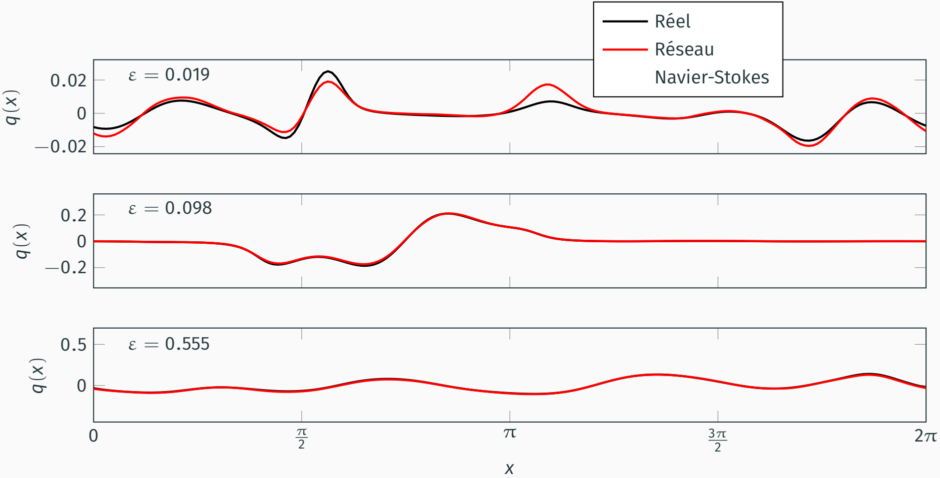
- This data work is a little bit old. At the time, convolutional architectures were the most relevant for this.
- Now neural operator and transformer offer a good alternative.
- We choose an architecture called V-net.



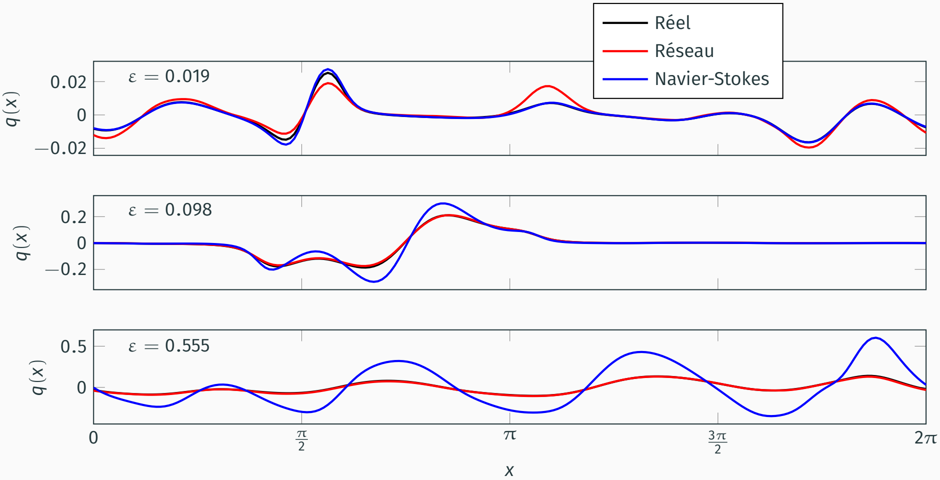
Results on test data



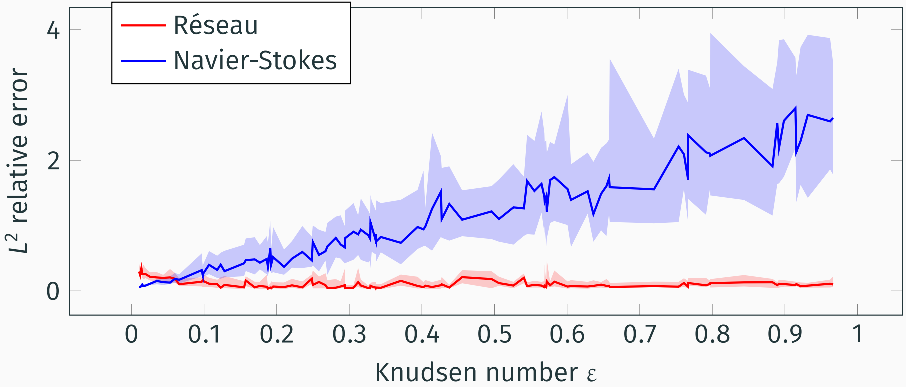
Results on test data



Results on test data



Influence of ϵ



- As expected, the Navier-Stokes closure error increase when ϵ increase. For the Network closure the error seems uniform. The error of NN is smaller than the error of NS excepted for small ϵ.

Full simulations

- We compare the evolution of electrical energy $\mathcal{E}(t) = \int E(x,t)^2\, dx$ for the different models :

Kinetic

- **In the following** : The difference between the kinetic model, the fluid model with kinetic closure, and the “Fluid + kinetic” model come from the errors associated to the scheme.
- **The best that we can do** : “Fluid + kinetic”.

Full simulations

- We compare the evolution of electrical energy $\mathcal{E}(t) = \int E(x,t)^2\,dx$ for the different models :

——— Kinetic

- - - - - Fluid + kinetic (perfect closure) : $\hat{q} = q$

- **In the following** : The difference between the kinetic model, the fluid model with kinetic closure, and the "Fluid + kinetic" model come from the errors associated to the scheme.
- **The best that we can do** : "Fluid + kinetic".

Full simulations

- We compare the evolution of electrical energy $\mathcal{E}(t) = \int E(x,t)^2\, dx$ for the different models :

———— Kinetic

- - - - - Fluid + kinetic (perfect closure) : $\hat{q} = q$

- - - - - Fluid + network : $\hat{q} = \mathcal{C}_\theta(\epsilon, \rho, u, T)$

- **In the following** : The difference between the kinetic model, the fluid model with kinetic closure, and the "Fluid + kinetic" model come from the errors associated to the scheme.
- **The best that we can do** : "Fluid + kinetic".

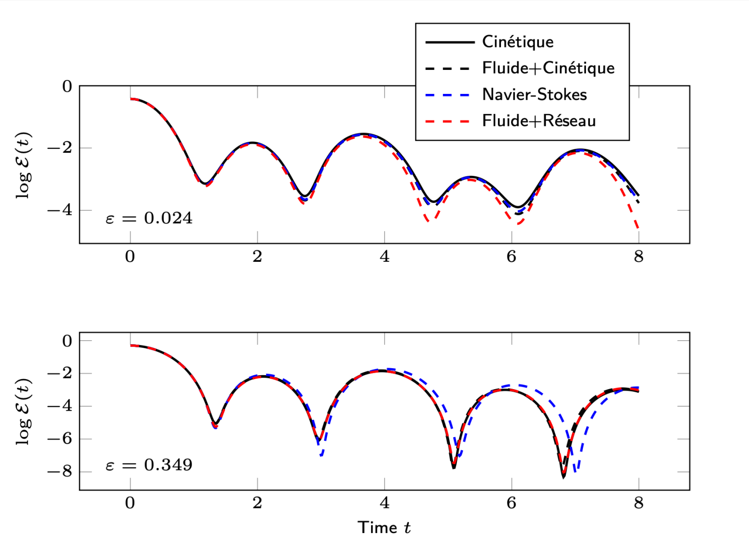
Full simulations

- We compare the evolution of electrical energy $\mathcal{E}(t) = \int E(x,t)^2 \, dx$ for the different models :

———	Kinetic
- - - - -	Fluid + kinetic (perfect closure) : $\hat{q} = q$
- - - - - (red)	Fluid + network : $\hat{q} = \mathcal{C}_\theta(\epsilon, \rho, u, T)$
- - - - - (blue)	Navier-Stokes : $\hat{q} = -\frac{3}{2}\epsilon\rho T \partial_x T$

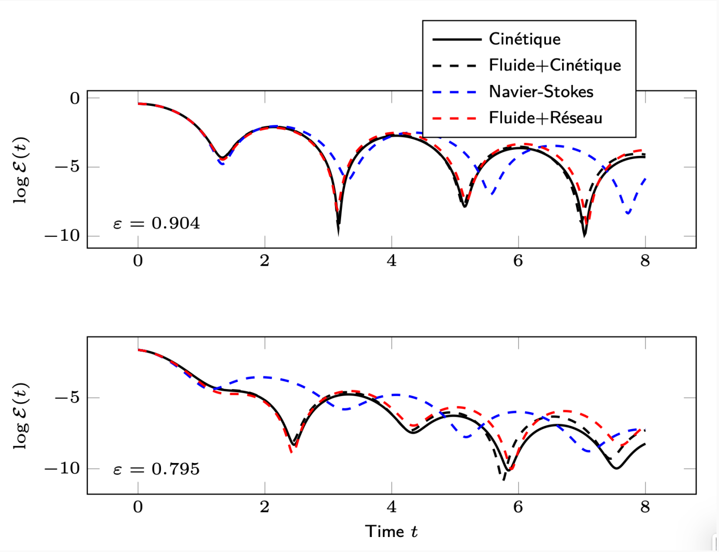
- **In the following** : The difference between the kinetic model, the fluid model with kinetic closure, and the “Fluid + kinetic” model come from the errors associated to the scheme.
- **The best that we can do** : “Fluid + kinetic”.

Examples (1/2)

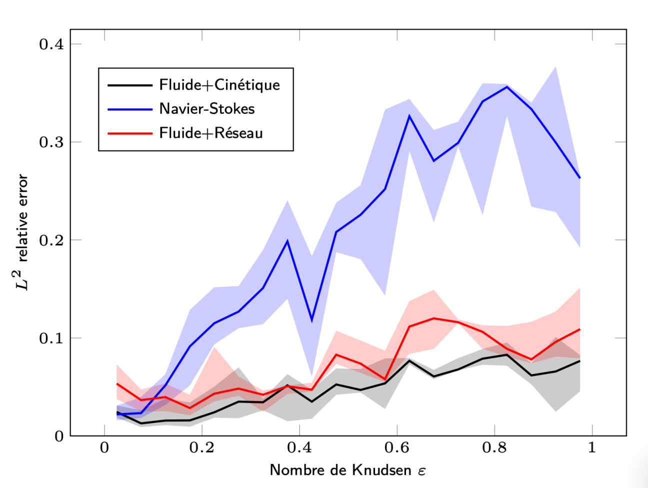


Cinétique
Fluide+Cinétique
Navier-Stokes
Fluide+Réseau
$\log \mathcal{E}(t)$
$\varepsilon = 0.024$
$\varepsilon = 0.349$
Time t

Examples (2/2)



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



- The error of the model “Fluid+network” increase like the error of the “fluid+kinetic” model. It can be explain by the numerical error which are different for the two models.

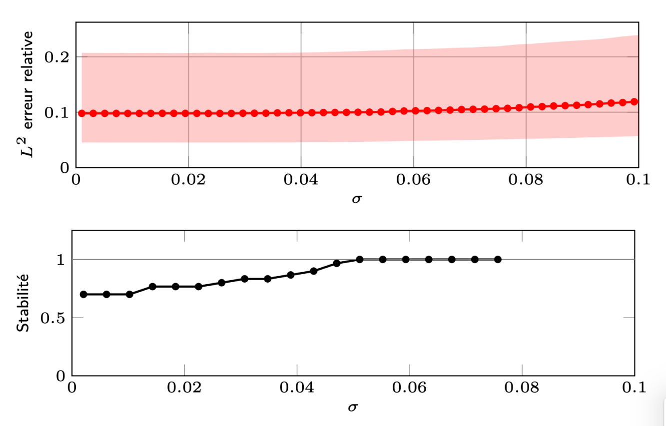
Difficulty: stability

- Network hyper-parameters :

Hyper-parameter	Value
input size	512
number of level (ℓ)	5
initial witdh (d)	4
kernel size (p)	11
activation function	softplus

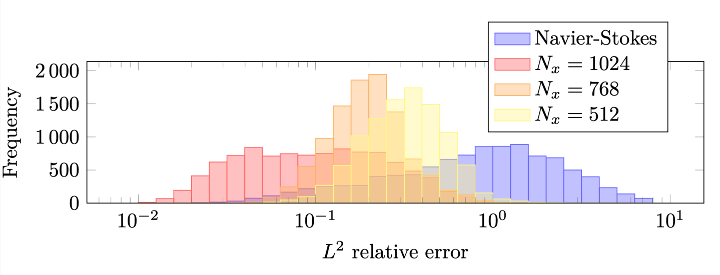
- total number of parameters : 161 937
- **Scheme** :
  - We compute Q_θ^n avec ρ^n, u^n and T^n
  - We compute the discrete derivative of $\partial_x Q_\theta^n$
  - We compute the fluid model with a explicit scheme and $\partial_x Q_\theta^n$ is used as a source term.
- We cannot assure that Q_θ^n and $\partial_x Q_\theta^n$ doest not generate spurious oscillations.
- The numerical oscillations can destroy the stability of the full scheme.
- **To limit this problem** : Gaussian regularization.

- **Top**: relative error on Q depending on regularization parameter σ
- **Bottom**: percentage of simulations that are stable



Resolution changing

- For real applications, we wish to change the mesh resolution.
- We plot the error at different resolution (training resolution is 1024):



- Use the network on other resolutions increase the error.
- A solution is to use interpolation/restriction operator to always evaluate the network on the same resolution. This works but is probably limited to smooth flows.

Reduced model for perturbation dynamics

Principle

Alternative method

We can construct a reduced model on the g dynamics and compute **Q** and Π using this approximation

- **Current work**: we consider a 1D2V problem with a complex collision operator:

$$\mathcal{Q}(f,f)=\nabla_v\cdot\int_{\mathbb{R}^2}A(\mathbf{v}-\mathbf{v}_*)(f(\mathbf{v}_*)\nabla_v f(\mathbf{v})-f(\mathbf{v})\nabla_{v_*}f(\mathbf{v}_*))d\mathbf{v}_*$$

- where $A(\mathbf{v})$ is given by:

$$A(\mathbf{v})=\frac{1}{|\mathbf{v}|}\left(\mathcal{I}-\frac{\mathbf{v}\otimes\mathbf{v}}{|\mathbf{v}|^2}\right)$$

Idea

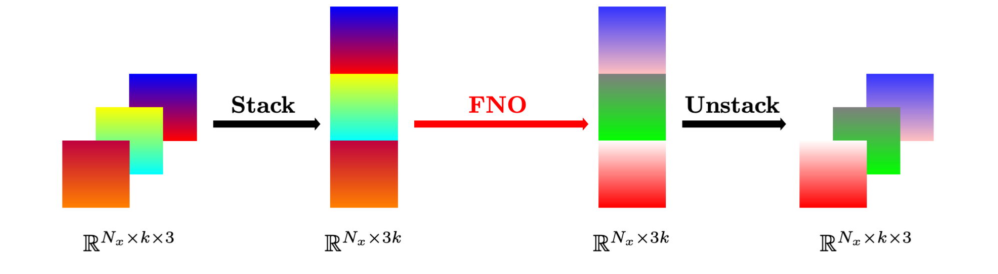
We will approximate the dynamic of moment of $g(t,x,\mathbf{v})$. At the very least, we need moments:

$$\int_{\mathbb{R}}\mathbf{v}v_1 g dv_1,\qquad \int_{\mathbb{R}}\frac{1}{2}|\mathbf{v}|^2 v_1 g dv_1$$

to reconstruct Π_{11}, Π_{21} and Q

Choice

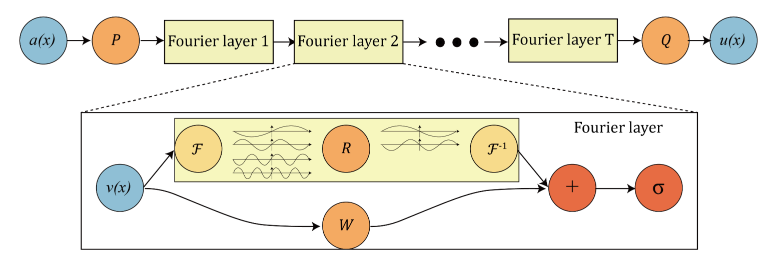
- To obtain the resolution invariance we use a Fourier neural operator.
- Between two time steps, g slowly evolves, and so:
  - We apply the FNO to the m time steps
  - The FNO take k time steps and predict the next k steps.



Neural Network architecture I

Choice

- To obtain the resolution invariance we use a Fourier neural operator.
- Between two time steps, g slowly evolves, and so:
  - We apply the FNO to the m time steps
  - The FNO take k time steps and predict the next k steps.

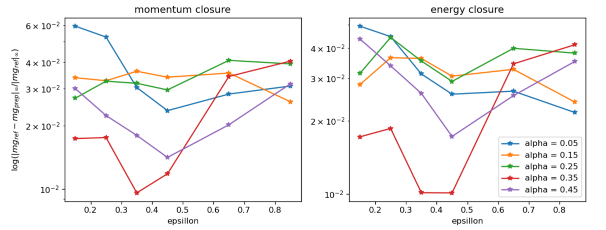


- For now, we consider small, parametric data sets.
- **Example 1**: Landau damping We consider the Landau damping test posed on

$$f(0,x,\mathbf{v}) = \frac{\rho(0,x)}{2\pi T(0,x)} \exp\left(-\frac{|\mathbf{v}-\mathbf{u}(0,x)|^2}{2T(0,x)}\right)$$

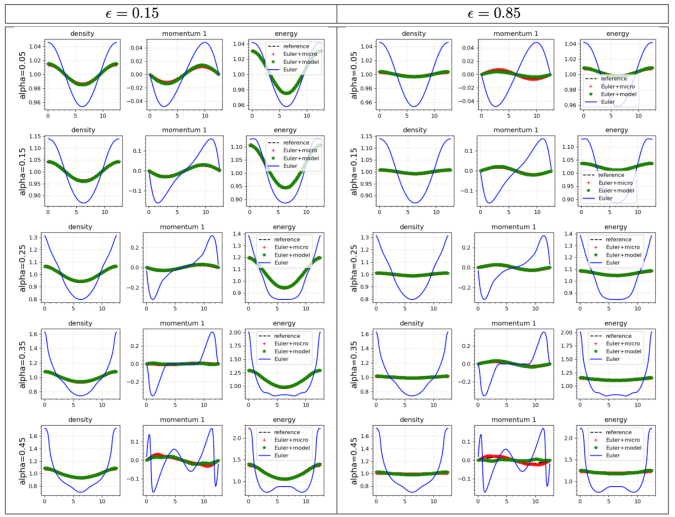
with $\mathbf{u}(0,x) = [0,0], T(0,x) = 1$ and $\rho(0,x) = 1 + \alpha\cos(0.5x)$, where $\alpha \in [0.05, 0.5]$, $\epsilon \in [0.1, 1.0]$ and we set periodic BC on the domain $[0,4\pi] \times [-6,6]^2$.

- Training data : 72 simulations and 2500 epochs.
- Train and test simulations: resolution grid 128 × 32 × 32, dt = 0.0025, T = 15.



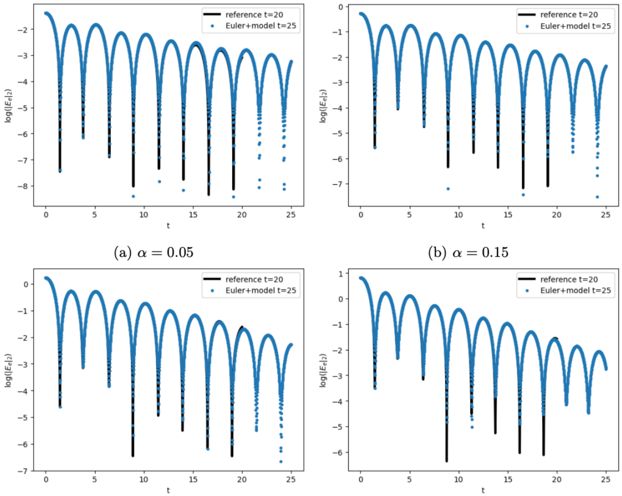
Results I

- Comparison solution at final time



Results II

- Electrical energy decay during the time simulation



(a) $\alpha = 0.05$ (b) $\alpha = 0.15$

(c) $\alpha = 0.25$ (d) $\alpha = 0.45$

Conclusion

Conclusion

Conclusion

The results show that learning methods are relevant for building hybrid reduced models for plasmas. The closure approach has been validated on larger data set. The second approach allows to approximate more physical phenomena.

Difficulties

- Generating data can be costly...
- Theoretical and numerical stability remains an open question!

Stability

- The Vlasov equation admit a metriplectic structure:

$$\frac{df}{dt} = [f, H] + (f, S)$$

  with H the conserved Hamiltonian and dissipated entropy S.
- **Aim**: preserve this structure in the reduced models.