Fermeture pour les équations de Vlasov-Poisson par apprentissage profond

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

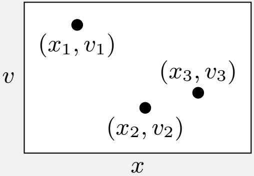
[2] INRIA Nancy-Grand Est, TONUS Project, Strasbourg, France

CEA-Gamni, 26/01/21

Sommaire

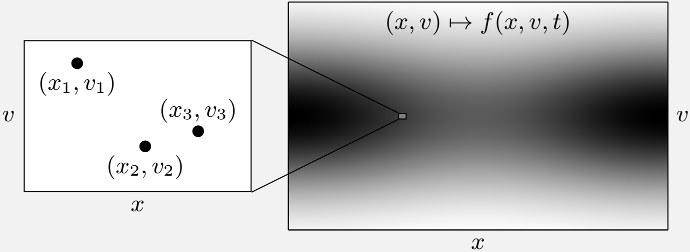
Sommaire

Modèle cinétique



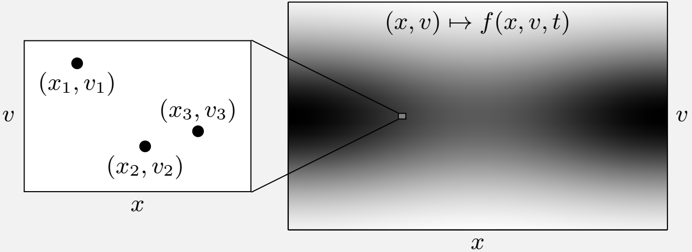
Modèle cinétique

Distribution de particules dans l'espace des phases



Modèle cinétique

Distribution de particules dans l'espace des phases



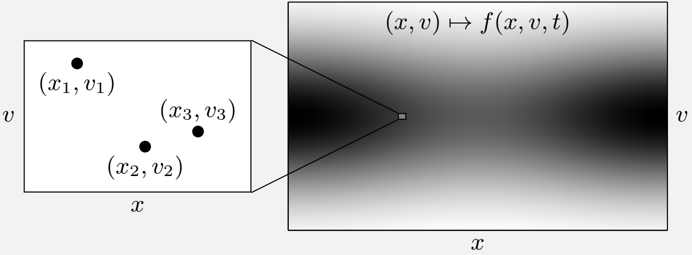
Equation de Vlasov-Poisson:

$$
\begin{array}{ll}
(V) & \partial_t f + v\partial_x f - E\partial_v f = \frac{1}{\varepsilon}(M(f) - f) \\
(P) & E(x,t) \text{ donné (slide suivant)}
\end{array}
$$

$E(x,t)$ champ électrique, $M(f)$ Maxwellienne de f, ε fréquence de collision.

Modèle cinétique

Distribution de particules dans l'espace des phases



Equation de Vlasov-Poisson:

$$\begin{aligned} (V) \qquad & \partial_t f + v\partial_x f - E\partial_v f = \frac{1}{\varepsilon}(M(f) - f) \\ (P) \qquad & E(x,t) \text{ donné (slide suivant)} \end{aligned}$$

$E(x,t)$ champ électrique, $M(f)$ Maxwellienne de f, ε fréquence de collision.

Côuteux en 3D, puisqu'on résout une EDP en dimension 6.

Passage du cinétique au fluide

Modèle fluide: modèle sur des quantités macroscopiques.
Les quantités fluides ρ, u, T, q peuvent être calculés à partir des moments en vitesse de la distribution f:

Passage du cinétique au fluide

Modèle fluide: modèle sur des quantités macroscopiques.
Les quantités fluides ρ, u, T, q peuvent être calculés à partir des moments en vitesse de la distribution f:

$$\int f(x,v,t)\,dv = \rho(x,t) \qquad\qquad \rho \text{ densité}$$

Passage du cinétique au fluide

Modèle fluide: modèle sur des quantités macroscopiques.
Les quantités fluides ρ, u, T, q peuvent être calculés à partir des moments en vitesse de la distribution f:

$$\int f(x,v,t)\,dv = \rho(x,t) \qquad \rho \text{ densité}$$

$$\int f(x,v,t)v\,dv = \rho(x,t)u(x,t) \qquad u \text{ vitesse moyenne}$$

Passage du cinétique au fluide

Modèle fluide: modèle sur des quantités macroscopiques.
Les quantités fluides ρ, u, T, q peuvent être calculés à partir des moments en vitesse de la distribution f:

$$\int f(x,v,t)\,dv = \rho(x,t) \qquad \rho \text{ densité}$$

$$\int f(x,v,t)v\,dv = \rho(x,t)u(x,t) \qquad u \text{ vitesse moyenne}$$

$$\int f(x,v,t)(v-u(x,t))^2\,dv = \rho(x,t)T(x,t) \qquad T \text{ température}$$

Passage du cinétique au fluide

Modèle fluide: modèle sur des quantités macroscopiques. Les quantités fluides ρ, u, T, q peuvent être calculés à partir des moments en vitesse de la distribution f:

$$\int f(x,v,t)\,dv = \rho(x,t) \qquad \rho \text{ densité}$$

$$\int f(x,v,t)v\,dv = \rho(x,t)u(x,t) \qquad u \text{ vitesse moyenne}$$

$$\int f(x,v,t)(v-u(x,t))^2\,dv = \rho(x,t)T(x,t) \qquad T \text{ température}$$

$$\frac{1}{2}\int f(x,v,t)(v-u(x,t))^3\,dv = q(x,t) \qquad q \text{ flux de chaleur}$$

Passage du cinétique au fluide

Modèle fluide: modèle sur des quantités macroscopiques.
Les quantités fluides ρ, u, T, q peuvent être calculés à partir des moments en vitesse de la distribution f:

$$\int f(x,v,t)\,dv = \rho(x,t) \qquad \rho \text{ densité}$$

$$\int f(x,v,t)v\,dv = \rho(x,t)u(x,t) \qquad u \text{ vitesse moyenne}$$

$$\int f(x,v,t)(v-u(x,t))^2\,dv = \rho(x,t)T(x,t) \qquad T \text{ température}$$

$$\frac{1}{2}\int f(x,v,t)(v-u(x,t))^3\,dv = q(x,t) \qquad q \text{ flux de chaleur}$$

L'équation de Poisson donne une relation entre E et ρ:

$$(P) \qquad E = -\partial_x\phi \quad , \quad \partial_{xx}\phi = \rho - \int \rho\,dx$$

Passage du cinétique au fluide

En calculant les moments en vitesse de Vlasov on obtient les équations d'Euler:

$$(V) \quad \xrightarrow{\quad \int_v \quad} \quad \begin{aligned} \partial_t \rho + \partial_x(\rho u) &= 0 \\ \partial_t(\rho u) + \partial_x(\rho u^2 + \rho T) &= -E\rho \\ \partial_t w + \partial_x(wu + \rho u T + q) &= -E\rho u \end{aligned}$$

Passage du cinétique au fluide

En calculant les moments en vitesse de Vlasov on obtient les équations d'Euler:

$$(V) \quad \xrightarrow{\quad \int_v \quad} \quad \begin{aligned} \partial_t \rho + \partial_x(\rho u) &= 0 \\ \partial_t(\rho u) + \partial_x(\rho u^2 + \rho T) &= -E\rho \\ \partial_t w + \partial_x(wu + \rho u T + q) &= -E\rho u \end{aligned}$$

Problème: trois équations, quatre variables. Comment on règle cela ? En ajoutant des moments ? La processus fait apparaitre toujours des moments supplémentaires de f et cela ne suffit pas pour fermer le système.

Femerture

Le flux de chaleur (pas d'équation temporelle dessus) q est donné par:

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

Principe de la fermeture: on cherche une approximation $\hat{q}$ de q basée sur les quantités ρ, u, T etc:

$$\hat{q}(\rho,u,T,\epsilon,..)$$

Femerture

Le flux de chaleur (pas d'équation temporelle dessus) q est donné par:

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

Principe de la fermeture: on cherche une approximation $\hat{q}$ de q basée sur les quantités ρ, u, T etc:

$$\hat{q}(\rho, u, T, \epsilon, ..)$$

Deux approches:

Fermeture analytique : faire des hypothèses f pour calculer analytiquement une expression de $\hat{q}$ basée sur ρ, u et T.

Fermeture basée sur les données : on simule f pour générer plusieurs (ρ, u, T, ϵ; q), et trouver une régression $(\rho, u, T, \epsilon) \mapsto \hat{q}$.

Fermetures analytiques

La fermeture la plus simple est celle d'*Euler*

$$\hat{q} = 0,$$

obtenue à partir de l'hypothèse $f = M(f) + O(\varepsilon)$ en tronquant les termes en ϵ.

Fermetures analytiques

La fermeture la plus simple est celle d'*Euler*

$$\hat{q} = 0,$$

obtenue à partir de l'hypothèse $f = M(f) + O(\varepsilon)$ en tronquant les termes en ϵ.

La seconde fermeture est celle de *Navier-Stokes*

$$\hat{q} = -\frac{3}{2}\varepsilon\rho T\partial_x T$$

obtenue avec l'approximation $f = M(f) + \varepsilon g + O(\varepsilon^2)$.
Elle est valide pour des petits ϵ.
Il s'agit d'une fermeture locale: on obtient $\hat{q}(x)$ à partir de $\rho(x)$, $u(x)$ et $T(x)$.

Fermetures analytiques

La fermeture la plus simple est celle d'*Euler*

$$\hat{q} = 0,$$

obtenue à partir de l'hypothèse $f = M(f) + O(\varepsilon)$ en tronquant les termes en ϵ.

La seconde fermeture est celle de *Navier-Stokes*

$$\hat{q} = -\frac{3}{2}\varepsilon\rho T\partial_x T$$

obtenue avec l'approximation $f = M(f) + \varepsilon g + O(\varepsilon^2)$.
Elle est valide pour des petits ϵ.
Il s'agit d'une fermeture locale: on obtient $\hat{q}(x)$ à partir de $\rho(x)$, $u(x)$ et $T(x)$.

Autre fermeture: *Hammett-Perkins* qui permet de récupérer l'amortissement Landau.
Il s'agit d'une fermeture non-locale: on obtient $\hat{q}(x)$ en fonction de $\rho(.)$, $u(.)$ et $T(.)$.

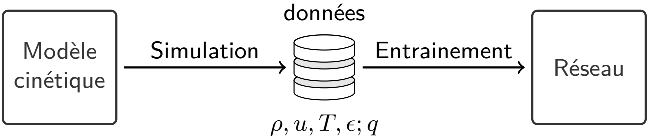
Fermeture basée sur les données

Deux phases de calcul:

Fermeture basée sur les données

Deux phases de calcul:

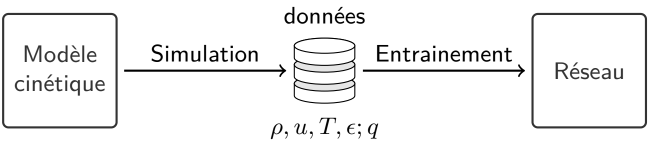
Phase hors-ligne



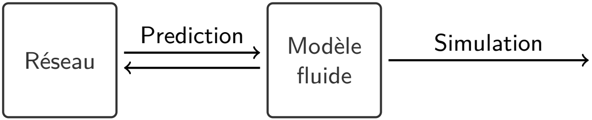
Fermeture basée sur les données

Deux phases de calcul:

Phase hors-ligne



Phase en ligne



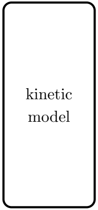
Sommaire

Sommaire

Génération des données

On considère une fermeture globale :$\hat{q}(\rho(.), u(.), T(.), \epsilon)$

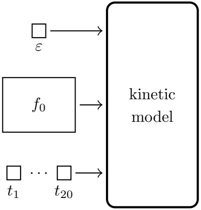
Les données sont calculées avec cinétique



Génération des données

On considère une fermeture globale :$\hat{q}(\rho(.), u(.), T(.), \epsilon)$

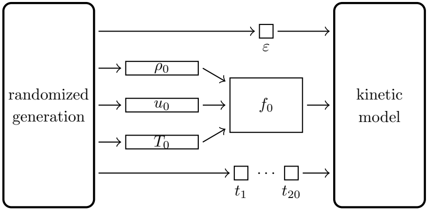
On prend un nombre de Knudsen ε, une condition initiale f_0, et 20 temps t_1, ..., t_{20}.



Génération des données

On considère une fermeture globale :$\hat{q}(\rho(.), u(.), T(.), \epsilon)$

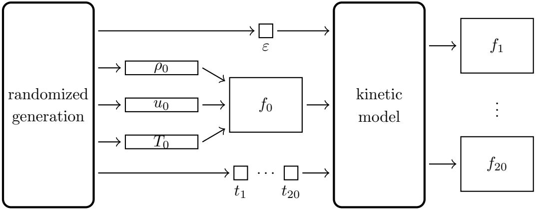
Les quantités sont générés aléatoirement. En particulier, $\varepsilon \in [0.01, 1]$.



Génération des données

On considère une fermeture globale :$\hat{q}(\rho(.), u(.), T(.), \epsilon)$

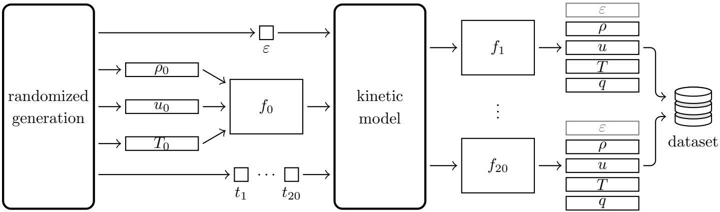
Le modèle donne f_1, ..., f_{20}
aux temps t_1, ..., t_{20} respectivement.



Génération des données

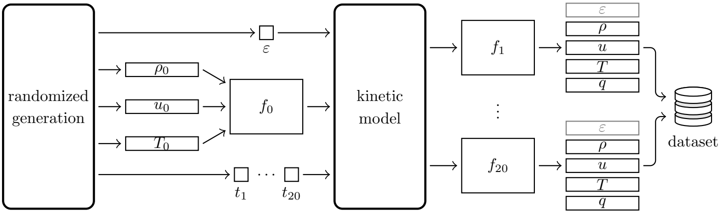
On considère une fermeture globale :$\hat{q}(\rho(.), u(.), T(.), \epsilon)$

Finalement les fonctions ρ, u, T et q sont calculées et stockées avec ε.



Génération des données

On considère une fermeture globale :$\hat{q}(\rho(.), u(.), T(.), \epsilon)$

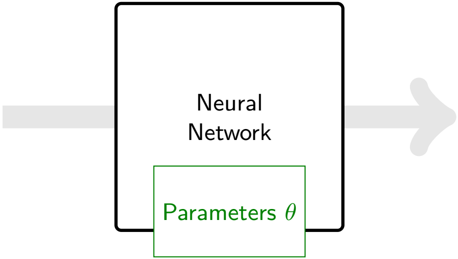


Deux jeux de données ($500 \times 20 = 10\,000$) sont générés: un pour l'entrainement, et un pour la validation.

Sommaire

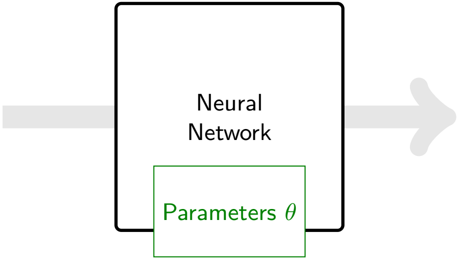
Entrainement

Un *réseau de neurones* une fonction particulière non-linéaire dépendante d'un grand nombre de paramètres.



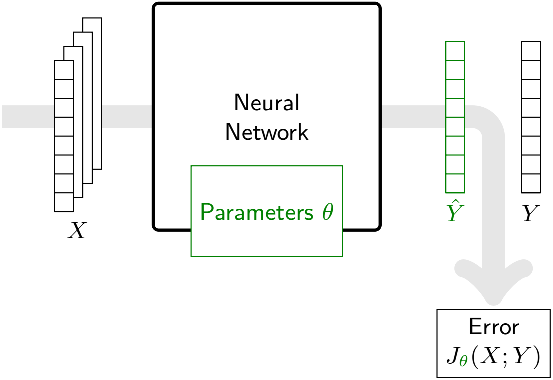
Entrainement

L'*entrainement* d'un réseau est un processus d'optimisation qui permet d'ajuster les paramètres.



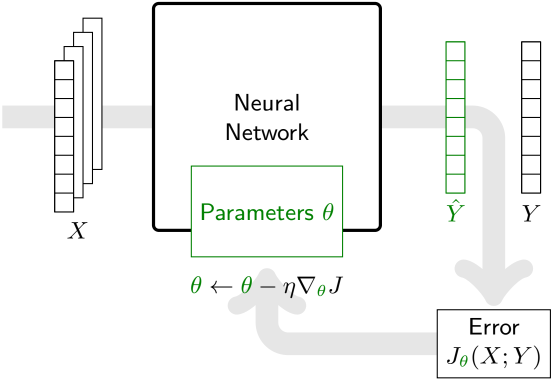
Entrainement

pour chaque couple (X, Y) dans l'ensemble de données,
une erreur est calculée entre la prédiction $\hat{Y}$ et la solution Y.



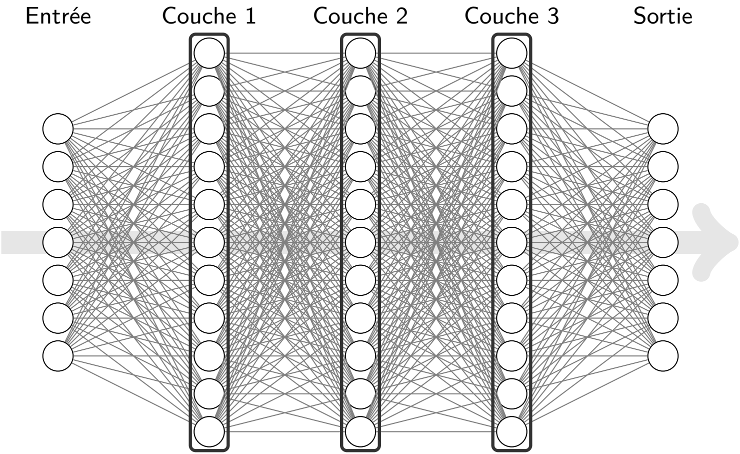
Entrainement

L'erreur permet d'ajuster les paramètres avec un algorithme de gradient.



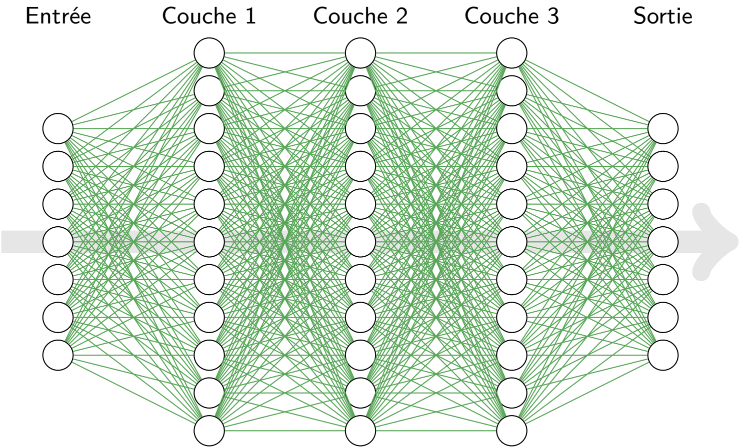
Structure générale

L'entrée passe sucessivement dans des *couches* cachées.



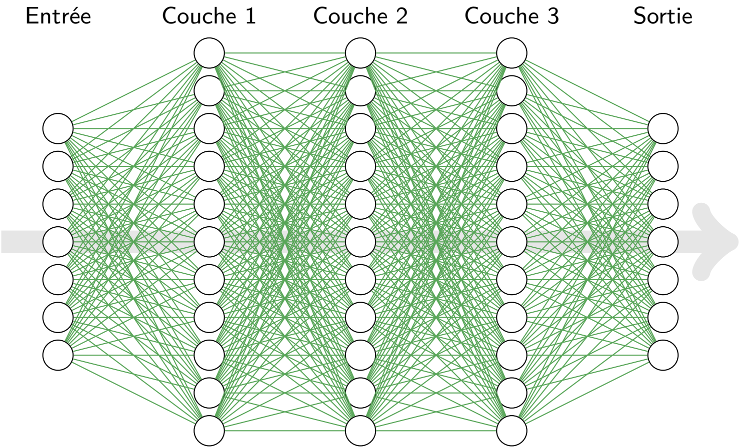
Structure générale

Chaque arête contient un paramètre qui peut être ajusté dans l'entrainement.



Structure générale

Chaque neurone applique une fonction d'activation nonlinéaire a son résultat.



Convolution

Réseaux convolutifs: étape linéaire basée sur des convolutions contre des petits noyaux.

Avantages: Réseau de neurones creux. Très efficace pour les données structurées (données spatiale sur grilles cartésiennes, images, signaux temporelles etc).

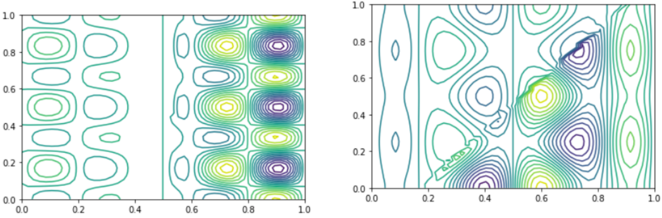
Exemple : détection de discontinuités dans des fonctions. On génère 1000 fonctions avec des modes de Fourier choisis aléatoirement. On ajoute une discontinuité aléatoire. 890 (entrainement), 110 (pour monitorer le sur-apprentissage).

But: classifier avec/sans discontinuité:

$$f(x,y) = \left(\sum_{i=1}^{5} \alpha_i cos(2\pi k_i x) sin(2\pi l_i y) \right) Disc(x,y)$$

avec k_i, l_i aléatoire entre 0 et 4, $\alpha_i \in [-1,1]$ et $Disc$ une discontinuité aléatoire. Maillage: 48×48.

Convolution II



Test: 200 fonctions supplémentaires.

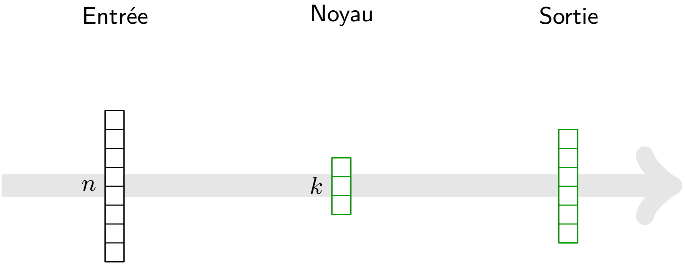
Réseau totalement connecté (4 couches) 1: 566 000 poids. Efficacité $\approx 0.5 - 0.6$.

Réseau totalement connecté (4 couches) 2: 230 000 poids. Efficacité $\approx 0.5 - 0.6$.

CNN (3 couches Conv, 3 couches denses) : 41 300 poids. Efficacité $\approx 0.8 - 0.92$.

Convolution
Base

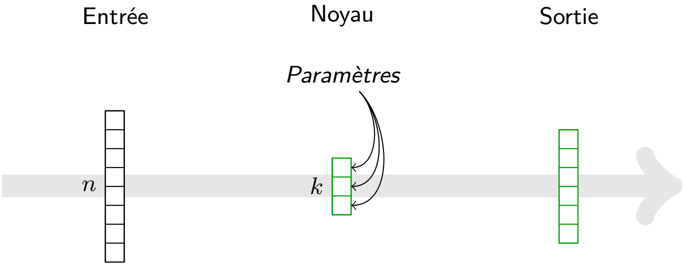
Couche convolutive:



Convolution

Base

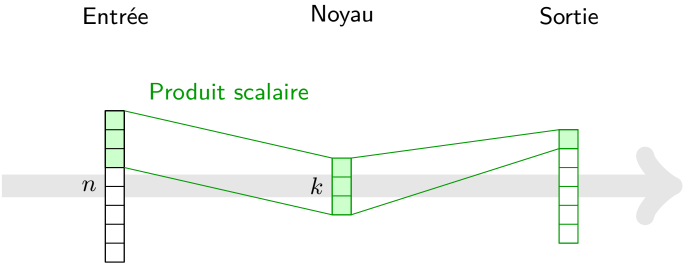
Les paramètres sont localisés dans le noyau.



Convolution

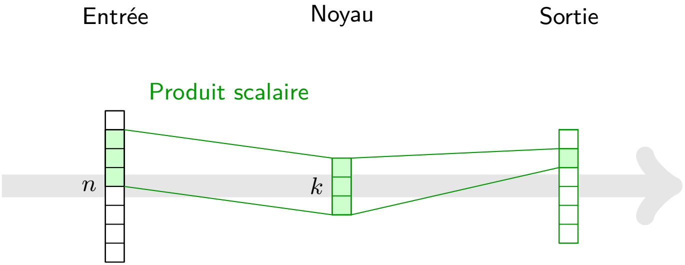
Base

Le produit avec le noyau applique un filtre.



Convolution
Base

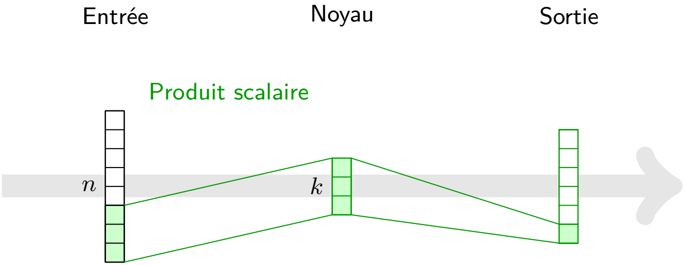
Le produit avec le noyau applique un filtre.



Convolution

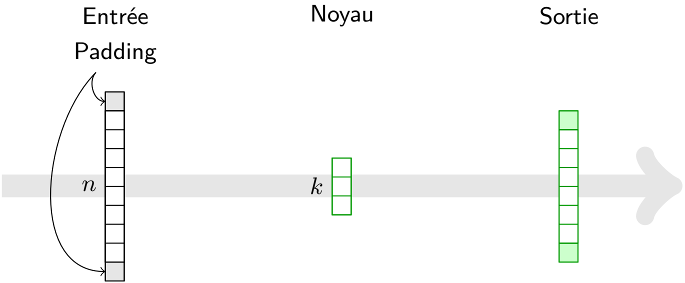
Base

Le produit avec le noyau applique un filtre.



Convolution
Padding

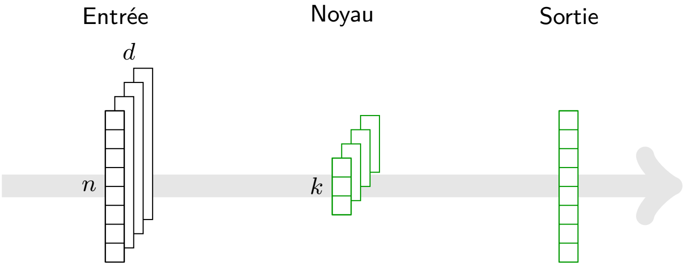
Pour obtenir une sortie de même taille: "padding".



Convolution

Entrée à plusieurs vecteurs

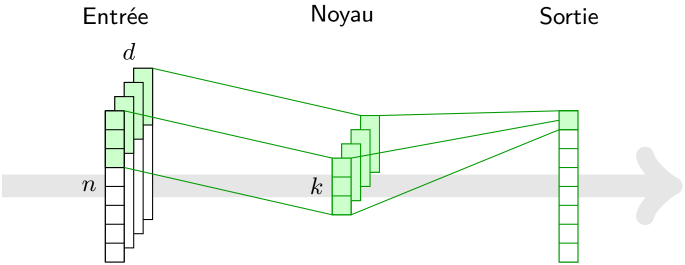
Entrées multiples: couleurs dans une image...



Convolution

Entrée à plusieurs vecteurs

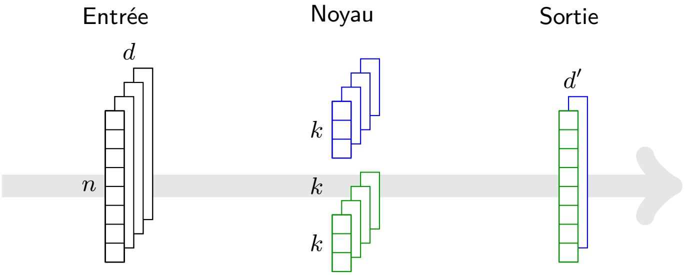
La sortie est donnée par la somme des produits scalaires.



Convolution

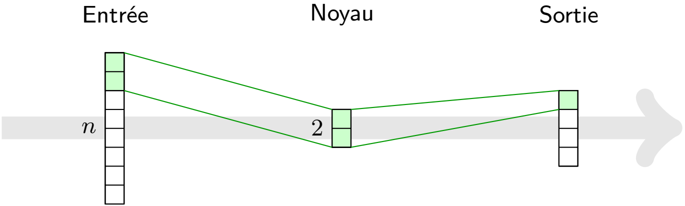
Sortie à plusieurs vecteurs

On peut appliquer plusieurs filtres (noyau).



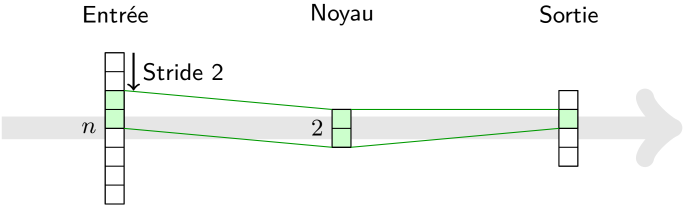
"Sous et sur echantillonnage"

On utilise une convolution différente pour diminuer la taille du signal.



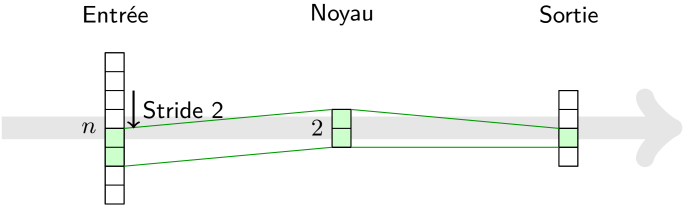
"Sous et sur echantillonnage"

Noyau de taille 2 avec un *pas* de 2



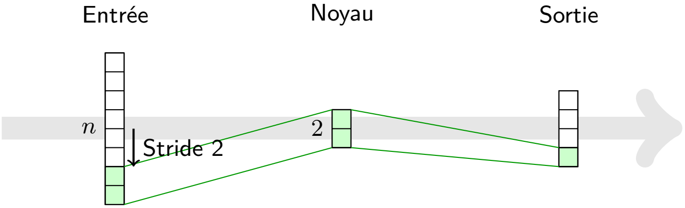
"Sous et sur echantillonnage"

En négligeant les bords, sortie= entrée/pas



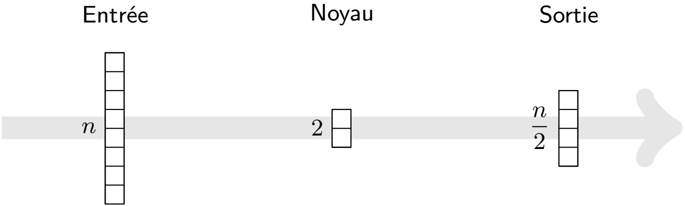
"Sous et sur echantillonnage"

En négligeant les bords, sortie= entrée/pas



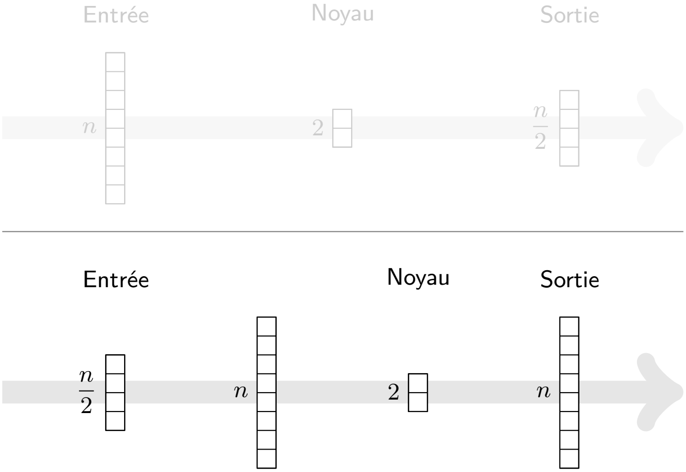
"Sous et sur echantillonnage"

Ici la sortie est deux fois echantillonné que l'entrée.



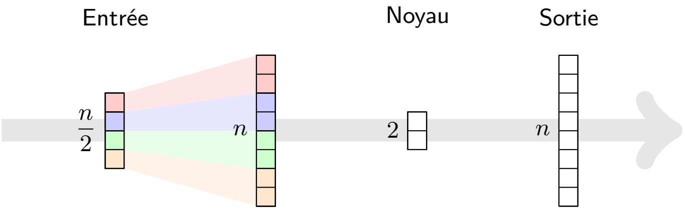
"Sous et sur echantillonnage"

L'opération inverse:



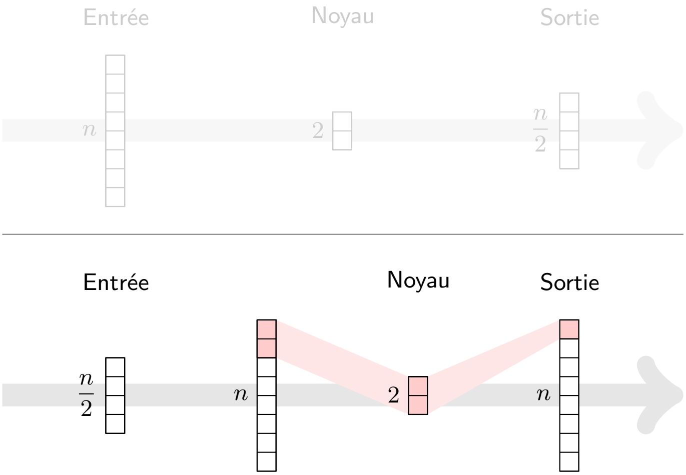
"Sous et sur echantillonnage"

on duplique les coefficients.



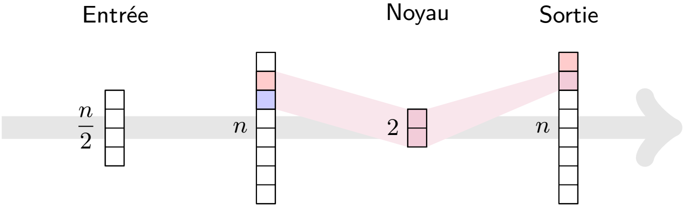
"Sous et sur echantillonnage"

On applique une convolution de taille 2 pour régulariser.



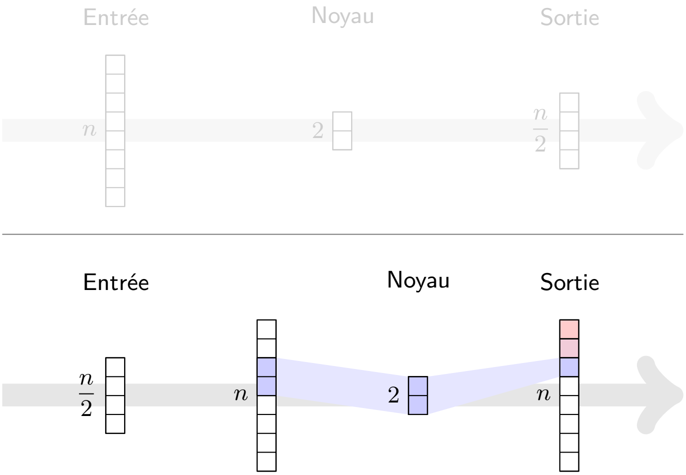
"Sous et sur echantillonnage"

On applique une convolution de taille 2 pour régulariser.



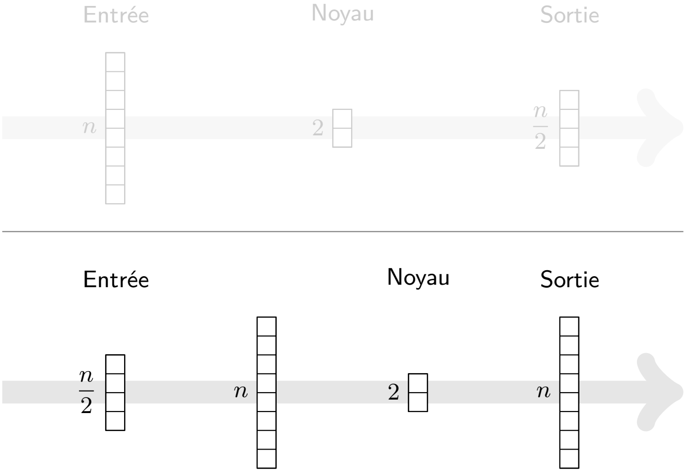
"Sous et sur echantillonnage"

On applique une convolution de taille 2 pour régulariser.



"Sous et sur echantillonnage"

On a, au final, doublé la taille.

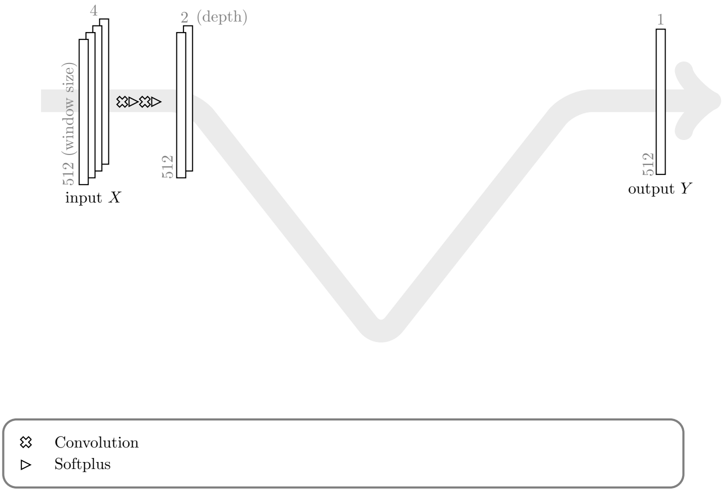


Réseau V-Net

On utilise une version 1D de l'architecture *V-Net*, pour la fermeture non-locale.

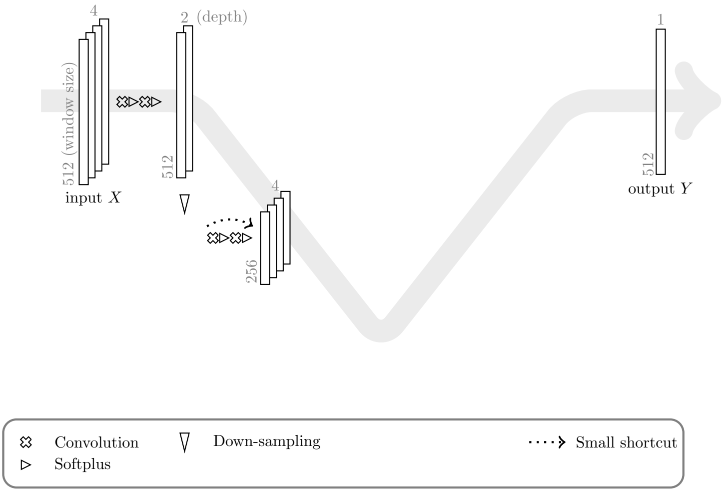
Réseau V-Net

On applique deux couches de convolutions
pour obtenir d (profondeur) signaux de sortie. Activation: softplus.



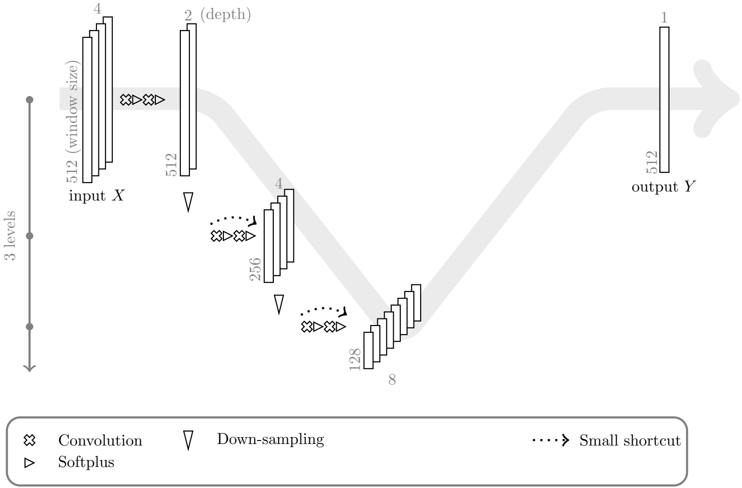
Réseau V-Net

On sous échantillonne,
on applique deux convolutions en multipliant par deux la profondeur.



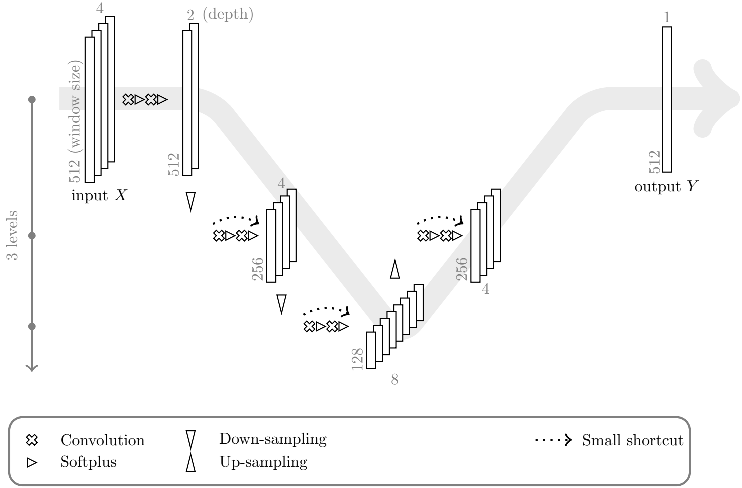
Réseau V-Net

L'opération est repétée sur ℓ niveaux.



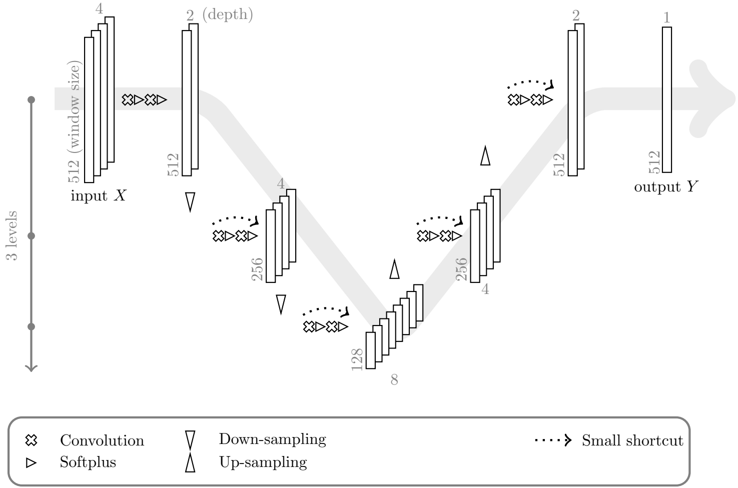
Réseau V-Net

On applique les opérations inverses pour obtenir la résolution d'origine.



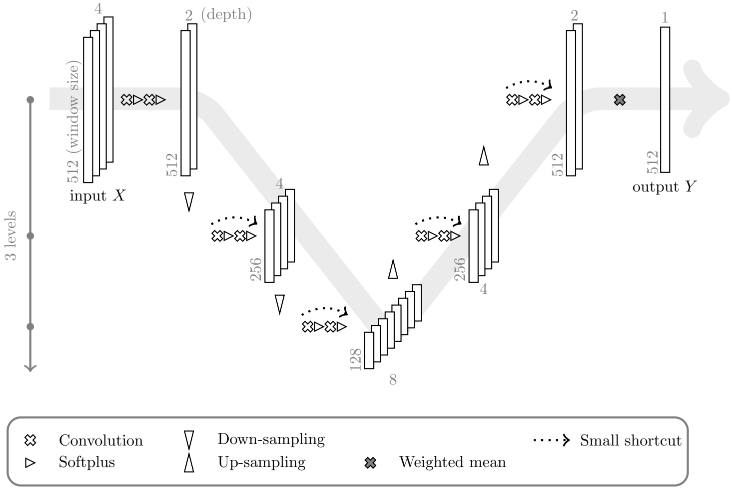
Réseau V-Net

On applique les opérations inverses pour obtenir la résolution d'origine.



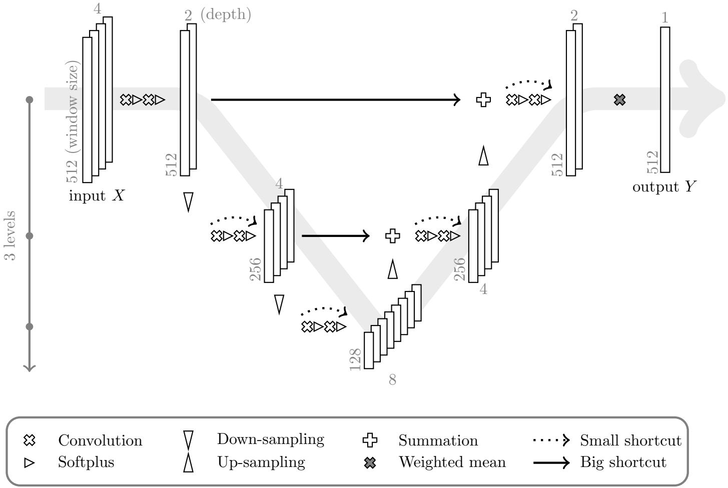
Réseau V-Net

A l'aide d'une moyenne pondérée pour obtenir la sortie.



Réseau V-Net

On moyenne les signaux avant et après reconstruction



Hyper-paramètres

Hyper-paramètres	Valeur
taille de l'entrée	512
nombre de niveau (ℓ)	5
profondeur initiale (d)	4
taille des noyaux (p)	11
fonction d'activation	softplus

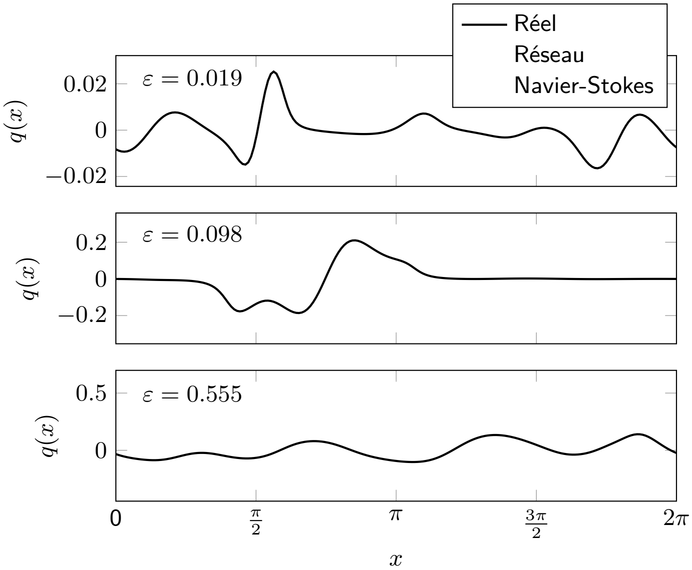
Nombre total de paramètres: $161\,937$

Préparation des données (standardisation, découpage, normalisation) en annexe.

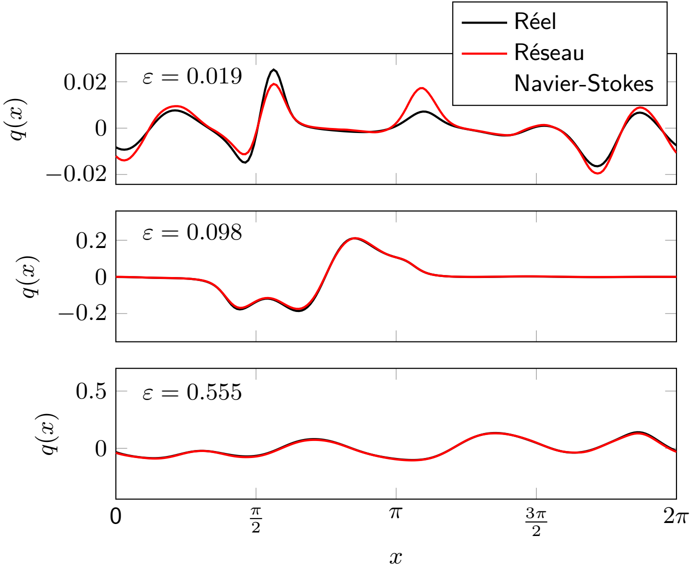
Sommaire

Sommaire

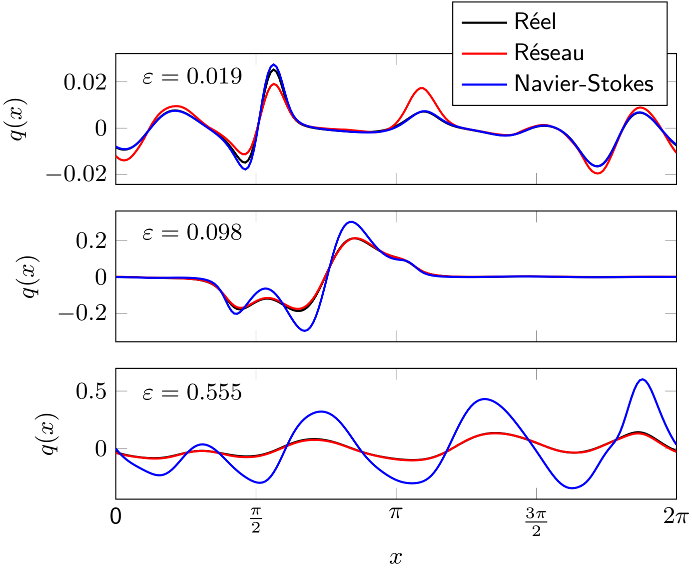
Exemples issus du jeu de données test



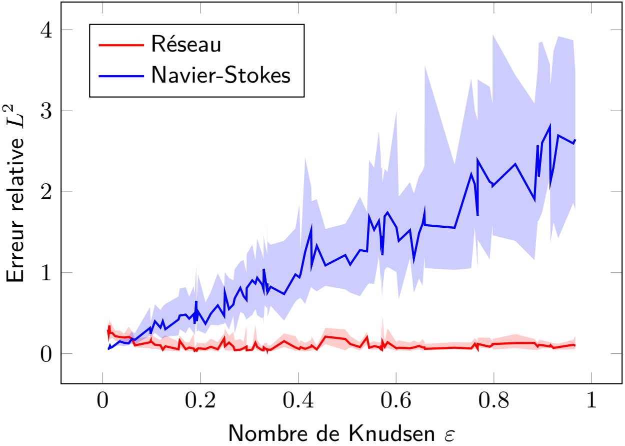
Exemples issus du jeu de données test



Exemples issus du jeu de données test



Influence de ε



Comme attendu l'erreur explose pour Navier-Stokes quand ε augmente. L'erreur semble quasiment uniforme pour le réseau (moins bon pour les petits ϵ).

Sommaire

Modèle fluide

On compare l'*énergie électrique* $\mathcal{E}(t) = \int E(x,t)^2\,dx$ dans les modèles suivants:

Cinétique

Dans la suite: Il y a des différences entre "le modèle cinétique" and "le modèle fluide+cinétique". Cela semble provenir des méthodes numériques. **On peut pas faire mieux que**:"fluide + cinétique".

Modèle fluide

On compare l'*énergie électrique* $\mathcal{E}(t) = \int E(x,t)^2 \, dx$ dans les modèles suivants:

——— Cinétique

- - - - - Fluide+Cinétique: $\hat{q} = q$

Dans la suite: Il y a des différences entre "le modèle cinétique" and "le modèle fluide+cinétique". Cela semble provenir des méthodes numériques. **On peut pas faire mieux que**:**"fluide + cinétique"**.

Modèle fluide

On compare l'*énergie électrique* $\mathcal{E}(t) = \int E(x,t)^2\, dx$ dans les modèles suivants:

———	Cinétique
- - - - -	Fluide+Cinétique: $\hat{q} = q$
- - - - - (rouge)	Fluide+Réseau: $\hat{q} = C_\theta(\varepsilon, \rho, u, T)$

Dans la suite: Il y a des différences entre "le modèle cinétique" and "le modèle fluide+cinétique". Cela semble provenir des méthodes numériques. **On peut pas faire mieux que**:"fluide + cinétique".

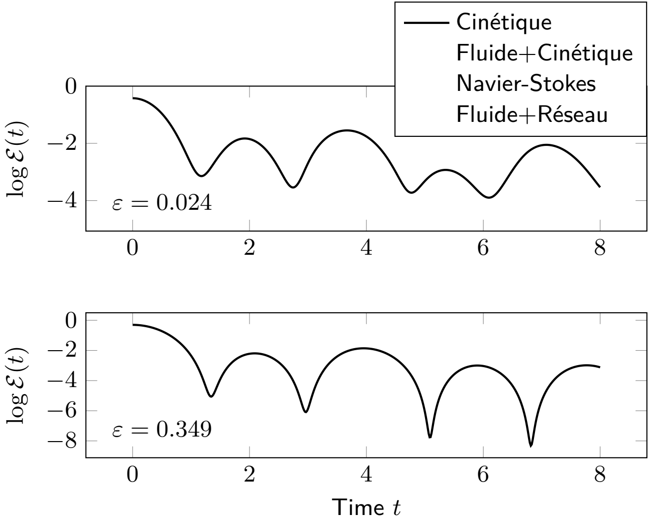
Modèle fluide

On compare l'*énergie électrique* $\mathcal{E}(t) = \int E(x,t)^2\, dx$ dans les modèles suivants:

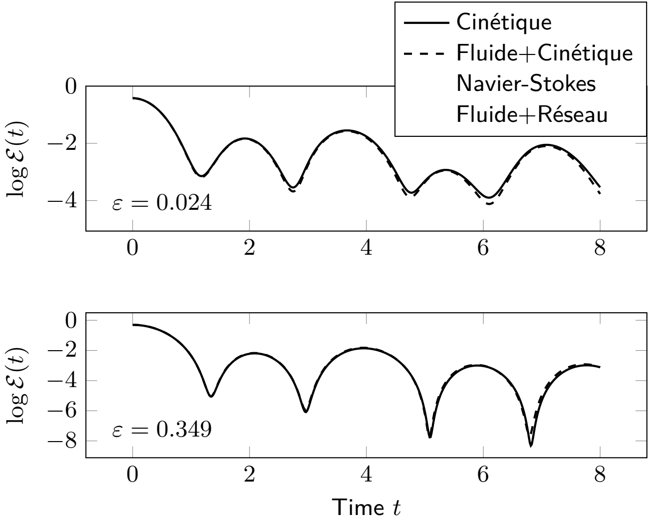
——— (noir, trait plein)	Cinétique
- - - - - (noir)	Fluide+Cinétique: $\hat{q} = q$
- - - - - (rouge)	Fluide+Réseau: $\hat{q} = C_\theta(\varepsilon, \rho, u, T)$
- - - - - (bleu)	Navier-Stokes: $\hat{q} = -\frac{3}{2}\varepsilon\rho T\partial_x T$

Dans la suite: Il y a des différences entre "le modèle cinétique" and "le modèle fluide+cinétique". Cela semble provenir des méthodes numériques. **On peut pas faire mieux que**:"fluide + cinétique".

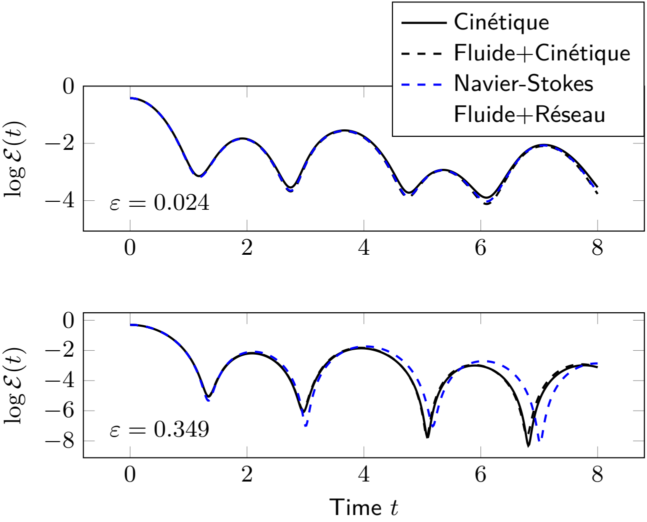
Exemples (1/2)


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

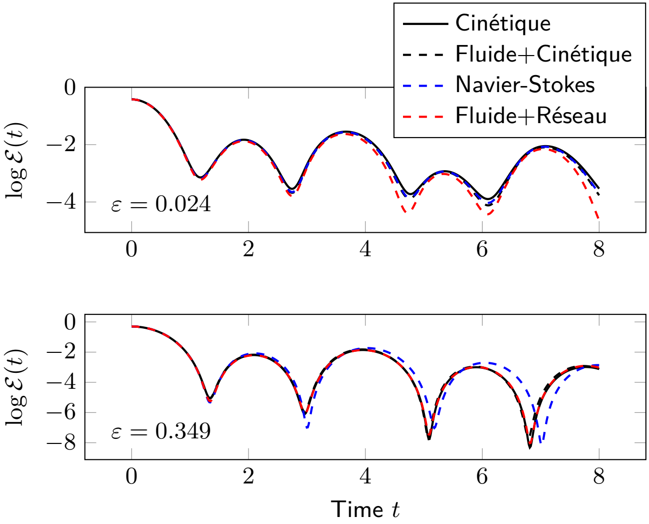
Exemples (1/2)


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

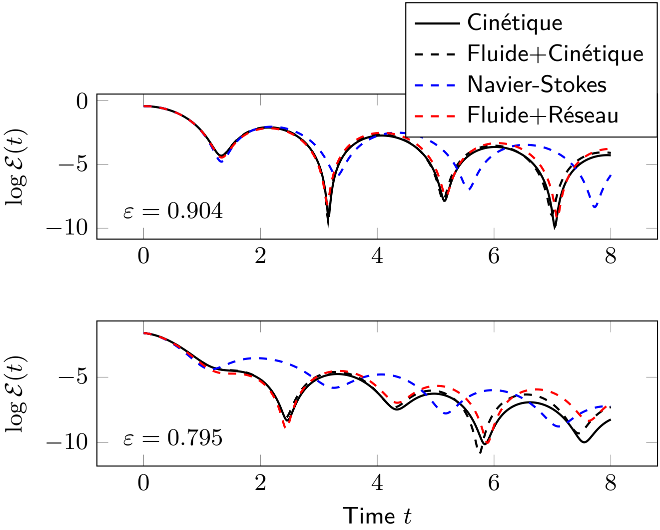
Exemples (1/2)


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

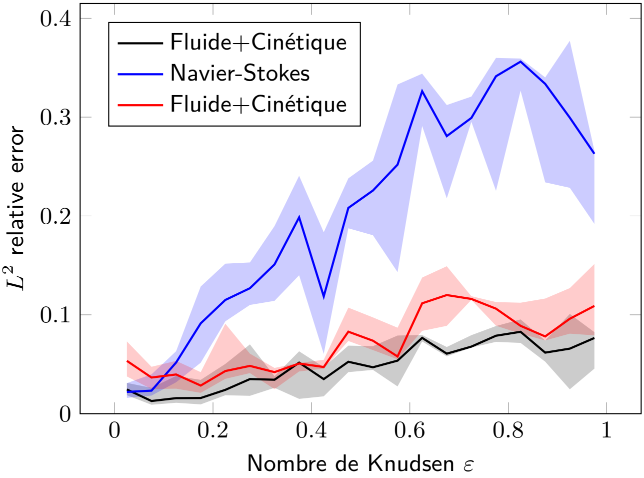
Exemples (1/2)


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

Exemples (2/2)


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

Influence de ε



L'erreur du modèle "Fluide+Réseau" augmente de façon similaire à celle du modèle "Fluide+Cinétique". Dérive de l'erreur expliquée par des problèmes numériques ?

Sommaire

Stabilité

Schéma :

- On calcul Q_θ^n avec ρ^n, u^n et T^n
- On calcul l'approximation de $\partial_x Q_\theta^n$
- On calcul le modèle fluide avec un schéma explicite $\partial_x Q_\theta^n$ traité comme un terme source.

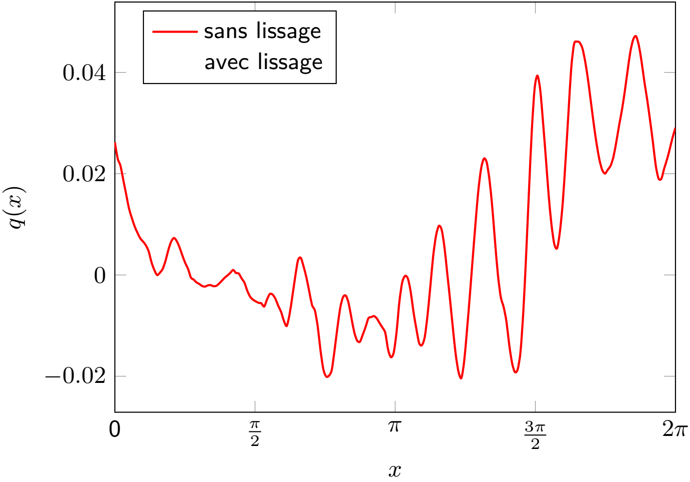
On peut pas assurer que Q_θ^n et $\partial_x Q_\theta^n$ ne génèrent pas des oscillations numériques.

Les oscillations peuvent détruire la stabilité du schéma complet. C'est le cas en pratique.

1ère approche (simple): Régularisation Gaussienne.

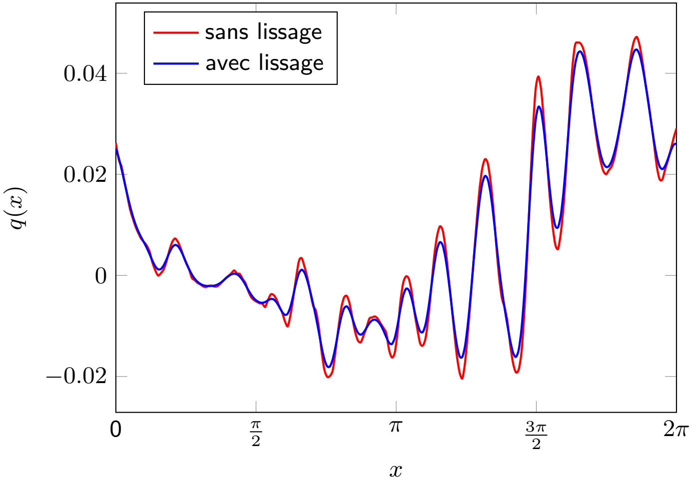
Exemple

Exemple de flux de chaleur particulièrement oscillant.



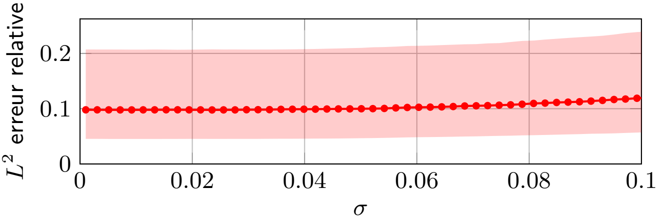
Exemple

On régularise la prédiction avec une convolution et un noyau Gaussien de paramètre σ.



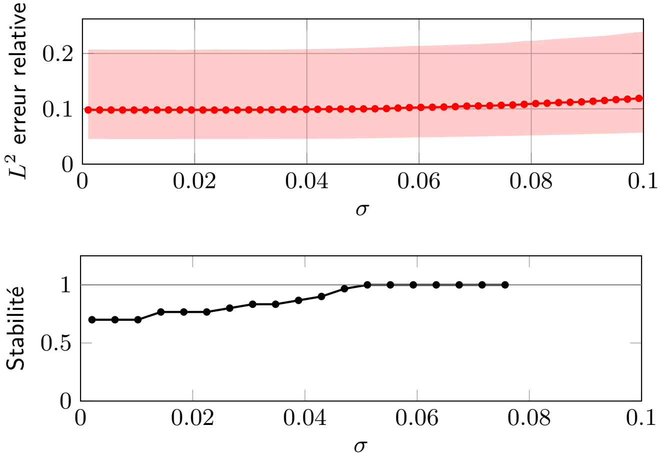
Analyse de stabilité

Première figure: erreur de prédiction en fonction du lissage.



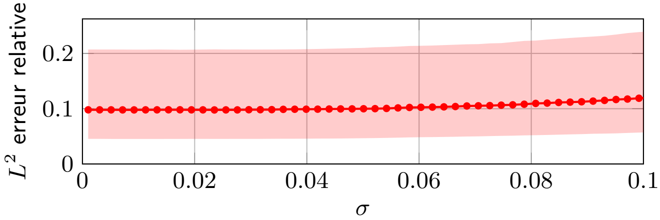
Analyse de stabilité

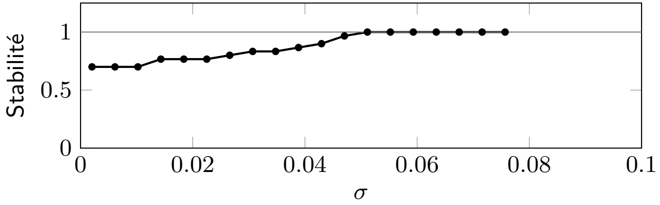
Cette deuxième figure montre
la proportion croissante de simulations réussies
à mesure que le lissage augmente.



Analyse de stabilité

Nous avons choisi $\sigma \simeq 0.06$,
car cela semble être suffisant pour assurer la stabilité sans pour autant diminuer la précision.





Coût CPU

Temps moyen pour des simulations avec $T_f = 8$ avec $N_x = 512$ et $N_v = 101$:

Cinétique	70 sec
Fluide+Cinetique	78 sec
Fluide+Réseau	74 sec
Navier-Stokes	3 sec

Une partie du temps de calcul peut venir de problème d'implémentation (communications CPU/GPU)

Complexité selon la dimension:

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	$O(d^2 p^3 N_x^3)$
Cinétique mD	$O(N_v^m N_x^m)$

Conclusion

Conclusion:
Preuve de concept en 1D

Conclusion

Ajout du champ magnétique

Assurer la stabilité.
Apprentissage par renforcement ?

Conclusion:
Preuve de concept en 1D

Conclusion

Ajout du champ magnétique

Assurer la stabilité.
Apprentissage par renforcement ?

Conclusion:
Preuve de concept en 1D

plus grand variété de ϵ

Modèles avec plus de moments

Conclusion

Ajout du champ magnétique

Assurer la stabilité.
Apprentissage par renforcement ?

Conclusion:
Preuve de concept en 1D

2D / 3D

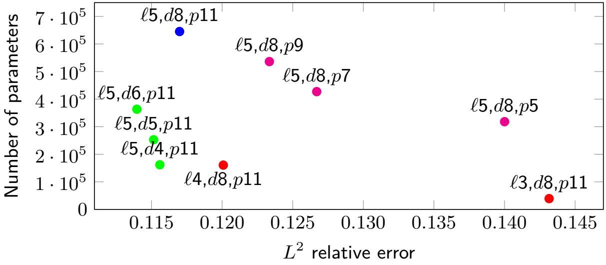
plus grand variété de ϵ

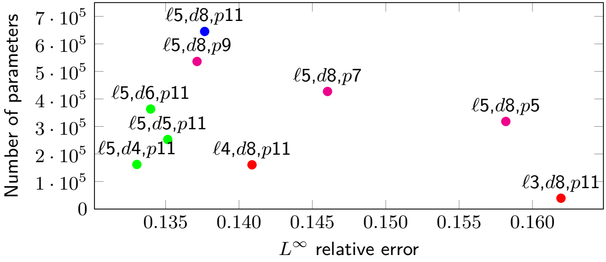
Modèles avec plus de moments

Extra

Hyper-paramètres

Résultats pour des jeux d'hyper-paramètres

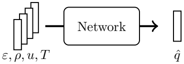




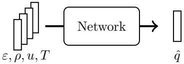
Préparation des données

Préparation des données.

Training



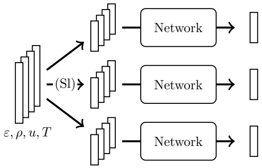
Prediction



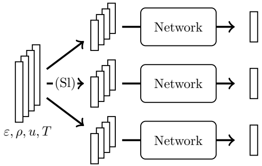
Préparation des données

Les données sont découpées (Sl) en plusieurs morceaux.

Training

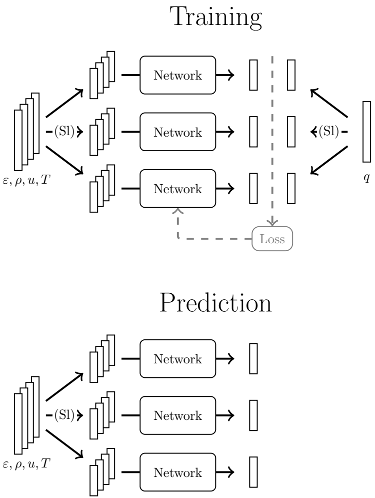


Prediction



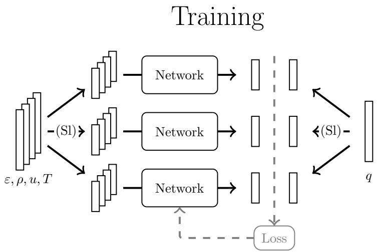
Préparation des données

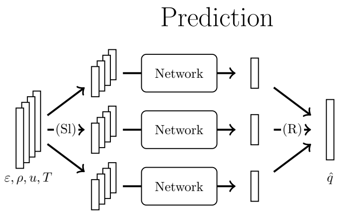
Pour l'entrainement, la sortie de référence (Sl) est aussi découpée.



Préparation des données

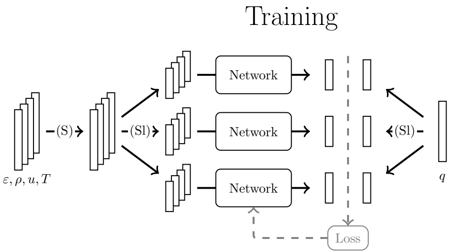
Pour la prédiction, la sortie est réconstruite (R) à partir des morceaux.

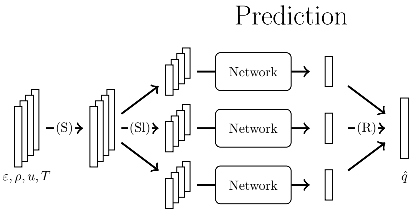




Préparation des données

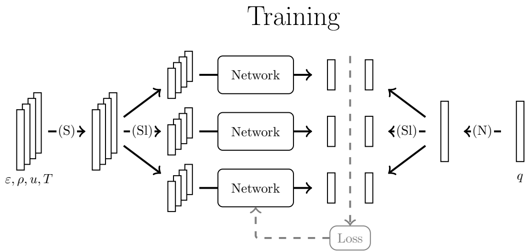
Les données son standardisées (S).

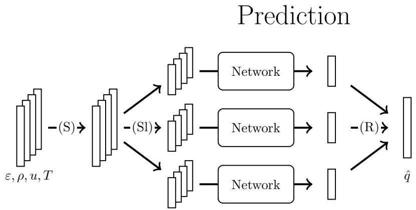




Préparation des données

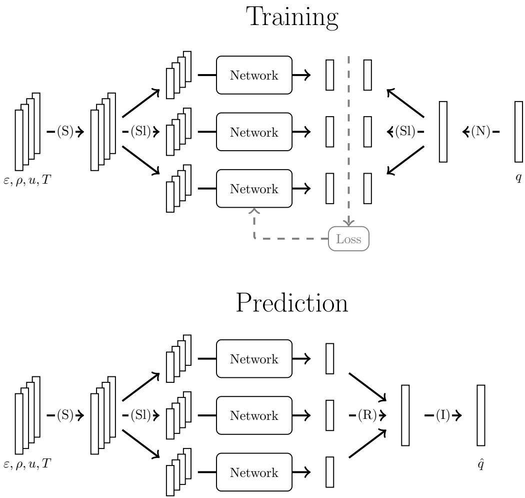
Pour l'entrainement, la sortie de référence est normalisée (N).





Préparation des données

Pour la prédiction, on applique la normalisation inverse (I).



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.

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

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.