

---

## UE APPRENTISSAGE PROFOND AVANCÉ

---

### - RAPPORT DE PROJET -

CELIK Baran



## Table des matières

<b>1.</b>	<b><i>Introduction</i></b>	<b>3</b>
<b>2.</b>	<b><i>Description du dataset physique et formulation du problème</i></b>	<b>4</b>
2.1.	Présentation du dataset	4
2.2.	Problème de prédiction	4
2.3.	Références physiques	5
<b>3.</b>	<b><i>Méthodologie</i></b>	<b>5</b>
3.1.	Modèles testés	5
3.2.	« Loss function » et contraintes physiques	6
3.3.	Protocole de training	7
3.4.	Analyse de la convergence des modèles	7
<b>4.</b>	<b><i>Résultats</i></b>	<b>8</b>
4.1.	Analyse quantitative des modèles sans PINN	8
4.2.	PINN : Impact des contraintes physiques	9
4.2.1.	<i>Ajout de la conservation de la masse</i>	9
4.2.2.	<i>Ajout de la conservation de l'énergie totale avec refroidissement</i>	10
4.3.	Analyse qualitative	11
4.3.1.	<i>Profils moyens de pression et de température</i>	12
4.3.2.	<i>« Cold Mass Fraction » (CMF)</i>	14
4.4.	Rollouts temporels	15
4.4.1.	<i>Analyse quantitative : MSE et VRMSE</i>	15
4.4.2.	<i>Analyse qualitative : évolution du champs de densité</i>	16
<b>5.</b>	<b><i>Discussion</i></b>	<b>17</b>
5.1.	Pertinence de la métrique VRMSE	17
5.2.	Influence de la normalisation manuelle	18
5.3.	Limite de l'évaluation par rollout statique	18
<b>6.</b>	<b><i>Conclusion</i></b>	<b>18</b>
<b>7.</b>	<b><i>Références bibliographiques</i></b>	<b>19</b>
<b>8.</b>	<b><i>Environnement de travail</i></b>	<b>19</b>
<b>9.</b>	<b><i>Annexe : Description détaillée des modèles utilisés</i></b>	<b>20</b>

## 1. Introduction

La modélisation numérique des phénomènes physiques complexes, tels que les écoulements turbulents, les instabilités hydrodynamiques ou encore les processus radiatifs, repose traditionnellement sur des solveurs d'équations aux dérivées partielles. Bien que ces méthodes soient précises, leur coût computationnel élevé limite leur application dans des contextes exploratoires, de contrôle ou d'optimisation. Pour pallier cette limitation, une approche prometteuse consiste à construire des modèles de substitution (*surrogate models*) capables d'approcher la dynamique d'un système à moindre coût à l'aide de l'apprentissage automatique. L'émergence récente de benchmarks à grande échelle dédiés à la simulation physique a ouvert de nouvelles perspectives dans ce domaine. C'est le cas de *The Well* [Ohana et al., 2024], une collection de 16 datasets représentant des systèmes physiques variés (biologie, astrophysique...), incluant plus de 15 To de simulations en grille régulière. Ce benchmark propose une interface PyTorch unifiée permettant d'entraîner et d'évaluer des modèles profonds sur des dynamiques complexes, avec des métriques standardisées comme le VRMSE.

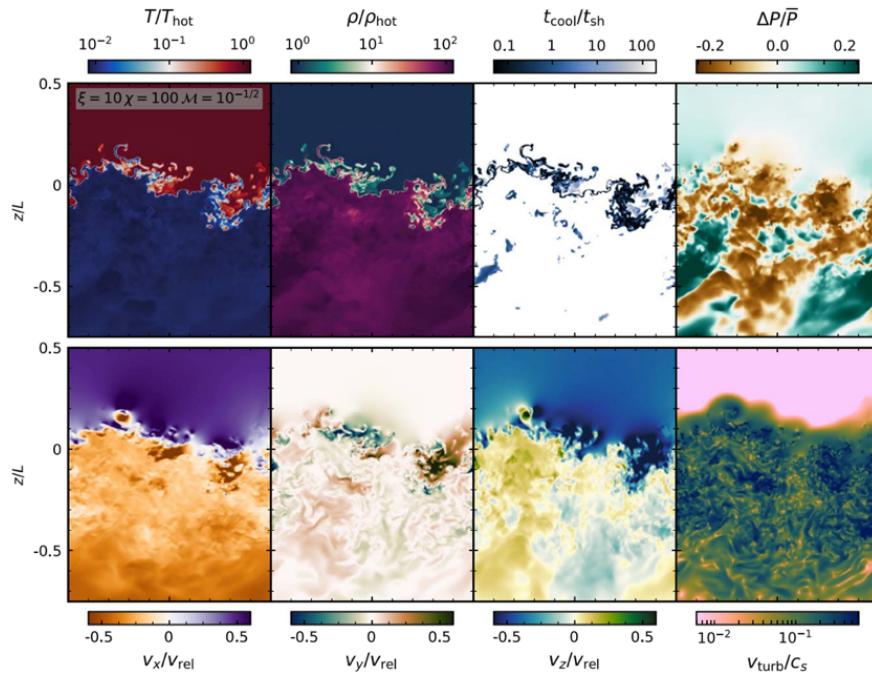


Figure 1 : Illustration du phénomène physique étudié : couches de gaz chaud et froid en mélange turbulent, issues de simulations de Fielding et al. (2020). Le refroidissement rapide est localisé dans une couche mince corrélée à la turbulence.

Dans ce projet, nous nous concentrons sur le dataset *turbulent\_radiative\_layer\_2D*, issu des travaux de Fielding et al. (2020). Ce cas représente une couche de mélange turbulente entre un gaz froid et dense et un gaz chaud et diffus. L'instabilité de Kelvin–Helmholtz entraîne un mélange turbulent, tandis que le refroidissement radiatif rapide modifie la dynamique du système. Ce phénomène, omniprésent en astrophysique (vents galactiques, milieu circumgalactique, filaments cosmiques), est modélisé ici en deux dimensions sur une grille de  $128 \times 384$ . Notre objectif est d'évaluer la capacité de plusieurs architectures de réseaux de neurones à

prédir l'évolution de champs physiques. Les performances des modèles sont évaluées à l'aide d'indicateurs quantitatifs (VRMSE, MSE) et de visualisations qualitatives des champs prédicts. Ce travail s'inscrit dans une démarche exploratoire visant à comprendre l'apport des contraintes physiques dans l'apprentissage profond pour la dynamique de systèmes complexes.

*Un notebook Jupyter accompagne ce rapport : il contient l'intégralité des visualisations et du code. Seuls les éléments les plus pertinents ont été extraits ici pour en faciliter la lecture.*

## 2. Description du dataset physique et formulation du problème

### 2.1. Présentation du dataset

Le dataset *turbulent\_radiative\_layer\_2D*, utilisé dans ce projet, est fourni dans le cadre du benchmark The Well. Il se présente sous la forme de tenseurs à quatre dimensions de taille (T,C,H,W) : T=101 est le nombre de pas de temps, C=4 correspond aux quatre champs physiques disponibles (*densité*, *pression*, *vitesse\_x*, *vitesse\_y*), et (H,W)=(128,384) désigne les dimensions spatiales de la grille régulière sur laquelle ces champs sont définis. Chaque échantillon représente une simulation indépendante couvrant une courte séquence temporelle, ce qui permet d'entraîner et d'évaluer des modèles sur des cas physiquement similaires mais initialisés différemment. Le sous-ensemble utilisé dans ce projet comprend 90 trajectoires spatio-temporelles, extraites aléatoirement du dataset complet. Les données sont stockées dans des fichiers HDF5, exploitables via l'interface WellDataset en PyTorch. Chaque échantillon contient l'ensemble des variables physiques sur toute la grille et pour tous les pas de temps. Avant l'apprentissage, les données sont normalisées manuellement : la moyenne et l'écart-type sont calculés sur le training set pour chaque champ, puis utilisés pour normaliser les valeurs. Cette étape permet une échelle comparable entre les variables et facilite la convergence des modèles.

### 2.2. Problème de prédiction

L'objectif de ce projet est de prédire l'évolution des champs physiques dans le temps à partir d'un état initial partiel. Plus précisément, il s'agit d'un problème de prédiction à un pas de temps (one-step prediction) : à partir de plusieurs états passés successifs, le modèle doit estimer l'état du système à l'instant suivant. Concrètement, pour chaque trajectoire de la base de données, on extrait une séquence de quatre pas de temps consécutifs (t-3, t-2, t-1, t) en entrée du modèle, et on lui demande de prédire les champs physiques au pas de temps suivant (t+1). Afin de maximiser l'exploitation des données, ce découpage est appliqué sous la forme de « slicing » : on fait défiler cette fenêtre temporelle le long de chaque trajectoire, en générant ainsi un grand nombre d'exemples distincts à partir d'un même échantillon. Par exemple, une trajectoire de 101 pas de temps permet de produire 97 couples entrée-sortie de ce type. La tâche est formulée comme un problème de régression supervisée, où les données d'entrée sont des tenseurs de forme (4,4,H,W) (quatre étapes temporelles, quatre champs physiques, grille spatiale), et la cible un tenseur de forme (4,H,W) (les quatre champs à prédire au temps t+1). Bien que ce projet se concentre principalement sur une prédiction à court terme, le cadre est naturellement généralisable à des prédictions multi-pas de temps (rollouts), en réinjectant les prédictions successives du modèle comme entrées pour les pas suivants. Ce type d'extension permet d'évaluer la capacité des modèles à maintenir la stabilité numérique et la cohérence

physique sur des horizons temporels étendus. Dans ce projet, quelques rollouts ont été réalisés en fin d'analyse afin d'illustrer ce comportement à plus long terme.

### 2.3. Références physiques

Le système modélisé dans ce projet correspond à une couche de mélange turbulente entre deux phases de gaz en contact : l'une froide et dense, l'autre chaude et diffuse, initialement en équilibre de pression. Ce type de configuration est sujet à l'instabilité de Kelvin–Helmholtz, qui génère un mélange progressif des deux phases à travers une interface dynamique. Dans le cas étudié, cette dynamique est renforcée par un refroidissement radiatif rapide, qui agit de manière préférentielle dans la phase intermédiaire, modifiant profondément l'évolution du système. Cette situation est inspirée des travaux de *Fielding et al. (2020)*, qui étudient ce processus dans un cadre astrophysique à l'aide de simulations tridimensionnelles haute résolution. Ils montrent que le refroidissement se produit principalement dans une couche mince et corrélée à la turbulence, dont la surface présente une structure fractale. Le transport d'énergie, de masse et de quantité de mouvement entre les phases chaudes et froides y est fortement influencé par la géométrie et la dynamique de cette interface.

Dans notre cas, le phénomène est simulé en deux dimensions et extrait dans un format discret et régulier adapté à l'apprentissage automatique. Les champs physiques étudiés (densité, pression, vitesse\_x, vitesse\_y) sont directement hérités des équations d'Euler compressibles, sans viscosité explicite ni conduction thermique. Le modèle est donc entraîné à reproduire les effets de ces mécanismes de mélange et de refroidissement, sans connaissance explicite des équations sous-jacentes. L'ajout de contraintes physiques avec les réseaux PINN (Physics-Informed Neural Networks) vise justement à réintroduire une forme de cohérence dynamique, en rappelant au modèle que certaines quantités (pression, énergie) doivent évoluer selon des relations imposées par la physique. Cette perspective permet d'aller au-delà d'une simple régression, et de rapprocher les modèles d'apprentissage profond de la modélisation physique interprétable.

## 3. Méthodologie

### 3.1. Modèles testés

Plusieurs architectures de réseaux de neurones ont été testées dans ce projet afin de comparer leurs performances pour la prédiction des champs physiques. Tous les modèles sont conçus pour fonctionner en apprentissage supervisé, en prenant comme entrée une séquence de quatre pas de temps et en produisant comme sortie l'estimation des champs physiques au pas de temps suivant. La famille principale de modèles étudiée est constituée de réseaux convolutifs CNN, auxquels différentes modifications ont été appliquées :

- **CNN** : architecture de base, peu profonde, composée de blocs convolutionnels avec ReLu et BatchNorm. Sert de point de référence minimal.
- **CNN2** : version plus profonde du modèle précédent avec un encodage renforcé (plus de couches, capacité temporaire augmentée à 256 canaux), mais sans skip connexion.
- **SkipCNN** : introduit une connexion de type U-Net entre l'encodeur et le décodeur, permettant de mieux préserver l'information spatiale fine.
- **CNN3** : architecture U-Net améliorée avec activation SiLu et GroupNorm, plus stable que BatchNorm pour de petits batchs.

- **CNN4** : version la plus avancée, encore plus profonde que CNN3, avec des blocs de raffinements supplémentaires. Ce modèle CNN4, en particulier, a été retenu comme architecture de référence pour l'intégration de contraintes physiques.

En parallèle, un modèle de type **FNO** (Fourier Neural Operator) a également été entraîné. Ce modèle est directement fourni par l'infrastructure du benchmark The Well, où il est utilisé comme baseline officielle sur plusieurs datasets. Contrairement aux CNN, le FNO agit dans l'espace spectral : il applique des transformations de Fourier aux champs d'entrée, effectue des opérations linéaires et non linéaires dans cet espace, puis revient à l'espace physique via une transformée inverse. Cette approche permet de capturer efficacement des interactions à longue portée, ce qui est particulièrement adapté aux phénomènes de type turbulent ou instationnaire. Le FNO constitue ainsi une référence solide pour la comparaison des performances des modèles convolutifs classiques. Dans notre projet, il est utilisé tel que fourni dans The Well, avec les hyperparamètres recommandés. Il permet d'évaluer si les architectures CNN spécifiques, conçues ou modifiées pour cette tâche, sont capables de rivaliser avec une approche spectrale.

*Voir la partie 8. Annexe pour le détail de l'architecture des modèles.*

### 3.2. « Loss function » et contraintes physiques

Tous les modèles sont entraînés à l'aide d'une fonction de coût de type MSE (Mean Squared Error), calculée entre les prédictions du réseau et le « ground truth » pour chacun des champs physiques. Cette perte mesure l'erreur quadratique moyenne sur l'ensemble de la grille, et constitue la base de l'apprentissage supervisé dans les modèles sans contrainte. Dans le cas du modèle CNN4, nous avons enrichi l'apprentissage par l'ajout de contraintes physiques, selon le formalisme des PINN. Deux contraintes sont ajoutées à la loss standard :

- **Conservation de la masse :**

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0$$

Cette équation exprime la conservation locale de la masse, en assurant que le flux massique est équilibré dans l'espace et dans le temps. Le terme correspondant est calculé numériquement à partir des prédictions du modèle à différents instants et dérivées spatiales (via différences finies).

- **Conservation de l'énergie totale avec refroidissement :**

$$\frac{\partial E}{\partial t} + \nabla \cdot ((E + P)\mathbf{v}) = -\frac{E}{t_{\text{cool}}}, \quad \text{où} \quad E = \frac{P}{\gamma - 1}, \quad \gamma = \frac{5}{3}$$

Ce terme modélise l'évolution de l'énergie totale (énergie interne plus énergie cinétique), avec une dissipation due au refroidissement radiatif. La forme résiduelle est également estimée à partir des sorties du réseau et intégrée dans la fonction de perte.

Ces deux résidus physiques sont ajoutés à la loss totale avec des coefficients de pondération  $\beta_1$  et  $\beta_2$ , ce qui donne :

$$\mathcal{L}_{totale} = \mathcal{L}_{MSE} + \beta_1 \cdot \mathcal{L}_{PINN,masse} + \beta_2 \cdot \mathcal{L}_{PINN,\text{énergie}}$$

Dans ce projet, ces coefficients ont été déterminés de manière empirique, en testant différentes valeurs fixes (par exemple  $\beta_1 \in \{0.1, 0.5, 1\}$ ) et en observant leur influence sur la convergence, la stabilité du modèle et la qualité des prédictions.

### 3.3. Protocole de training

Tous les modèles ont été entraînés en utilisant la bibliothèque PyTorch, sur GPU. L'ensemble des trajectoires disponibles a été divisé en trois sous-ensembles : 70 % pour le training, 15 % pour la validation et 15 % pour le test. Les séquences temporelles sont découpées par fenêtres « sliced » de longueur 5, où les quatre premiers pas servent d'entrée et le cinquième constitue la cible. Le batch size est fixé à 4 pour tous les modèles. L'optimisation est effectuée par l'algorithme Adam, avec un learning rate initial de  $10^{-3}$ . Le scheduler StepLR est utilisé pour diviser le taux d'apprentissage par 2 toutes les 5 epochs. La fonction de coût principale est la MSE, éventuellement complétée par les loss PINN dans le cas du modèle CNN4 modifié. Les modèles sont entraînés pendant 20 epochs, selon la convergence observée. Une légère régularisation est introduite dans certaines architectures via un dropout spatial. Les performances sont évaluées en fin d'apprentissage sur l'ensemble de validation, à l'aide des métriques MSE et VRMSE, calculées indépendamment pour chaque champ physique. Enfin, pour les modèles intégrant des contraintes PINN, les coefficients ont été ajustés manuellement en testant différentes valeurs, afin de trouver un compromis entre précision numérique et respect des lois physiques.

### 3.4. Analyse de la convergence des modèles

Afin de comparer la stabilité et l'efficacité de l'apprentissage des différentes architectures, les courbes de training loss ont été enregistrées au cours des 20 epochs. Les figures correspondantes sont présentées en Figure 2 : à gauche, les modèles convolutifs (CNN, CNN2, CNN3, CNN4, SkipCNN) ; à droite, la comparaison entre CNN4 et le FNO. Les modèles CNN, CNN2 et SkipCNN présentent une loss initiale d'environ 0.3, qui décroît relativement rapidement mais tend à se stabiliser à des valeurs comprises entre 0.20 et 0.23. Cette stagnation assez tôt peut s'expliquer par une capacité de représentation limitée et une profondeur modérée. CNN2 introduit une légère amélioration par rapport à CNN, avec une descente légèrement plus prolongée, mais finit lui aussi par atteindre un plateau dès la moitié du training. Le modèle SkipCNN, qui introduit une connexion de type skip entre les couches hautes et basses résolutions, améliore légèrement la régularité du signal mais reste freiné par un manque de raffinement en sortie. Dans l'ensemble, ces modèles semblent insuffisamment pour capturer les structures spatiales complexes de ces champs physiques.

À l'inverse, les architectures CNN3 et CNN4 convergent vers des training loss significativement plus basses. CNN3 atteint une perte d'environ 0.091 à la dernière epoch, tandis que CNN4 descend jusqu'à 0.0802, avec une décroissance continue sur l'ensemble des itérations. Leur progression reste active jusqu'à la fin du training, sans stagnation apparente. Cela suggère une meilleure capacité d'optimisation, probablement due à une structure plus adaptée à la complexité des données. CNN4 en particulier montre une convergence stable et régulière, ce qui en fait un choix pertinent pour l'ajout de contraintes PINN.

Le modèle FNO présente une dynamique de convergence très différente. Sa loss décroît de manière extrêmement rapide et atteint une valeur proche de 0.019 dès la vingtième epoch. Ce comportement est caractéristique des FNO, qui permettent une capture efficace des structures globales à grande échelle. Toutefois, cette rapidité ne se traduit pas nécessairement par une meilleure capacité de généralisation. Comme le montrent les résultats quantitatifs (section 4.1), le FNO présente des erreurs plus élevées que CNN4 sur plusieurs champs physiques (vitesse\_x, vitesse\_y, densité), malgré une training loss plus basse. Cela peut témoigner d'une forme d'ajustement rapide aux grandes tendances des données, mais peut être une difficulté à restituer les structures locales ou les zones de forte variabilité.

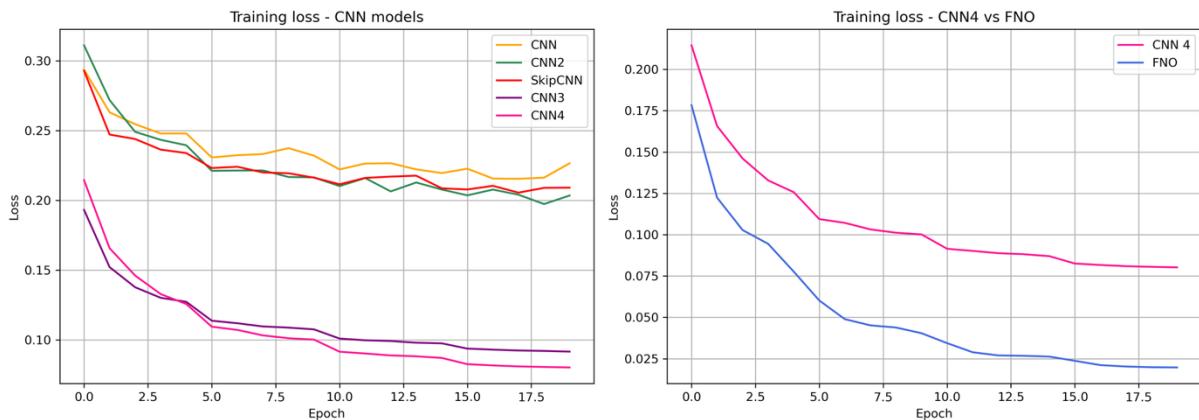


Figure 2 : Training loss des différents modèles (loss MSE sur 20 epochs) : à gauche, les modèles convolutifs (CNN, CNN2, CNN3, CNN4, SkipCNN) ; à droite, la comparaison entre CNN4 et le FNO.

## 4. Résultats

### 4.1. Analyse quantitative des modèles sans PINN

L'évaluation quantitative des modèles a été réalisée sur l'ensemble de validation, à l'aide des métriques MSE et VRMSE (Variance-Scaled Root MSE). Cette dernière est utilisée comme critère principal de comparaison, car elle permet de normaliser l'erreur par la variance du champ, ce qui la rend robuste aux différences d'échelle entre les variables physiques (par exemple, densité vs pression). Chaque modèle a été évalué indépendamment sur les quatre champs physiques du dataset. Les erreurs sont calculées en moyenne sur toutes les grilles et tous les exemples de la validation. Les résultats sont présentés sous forme de barplot comparatif (Figure 3), qui montre les VRMSE moyens obtenus par chaque modèle sur les quatre champs physiques.

Le modèle CNN4 se distingue comme le plus performant globalement, obtenant les meilleurs VRMSE sur trois des quatre champs : densité, vitesse\_x et vitesse\_y. Il affiche une bonne stabilité et des performances homogènes, ce qui en fait le modèle le plus robuste de l'étude. Sur le champ pression, le FNO obtient cependant la meilleure précision, avec un VRMSE de 0.6433 contre 0.7996 pour CNN4. Ce résultat peut être attribué à la capacité du FNO à capturer les structures globales via sa représentation spectrale. Les autres modèles (CNN, CNN2, CNN3, SkipCNN) présentent des performances plus variables. En particulier, SkipCNN affiche une

erreur très élevée sur la pression, suggérant une instabilité ou une mauvaise convergence. Le modèle CNN de base, utilisé comme référence minimale, est systématiquement en retrait.

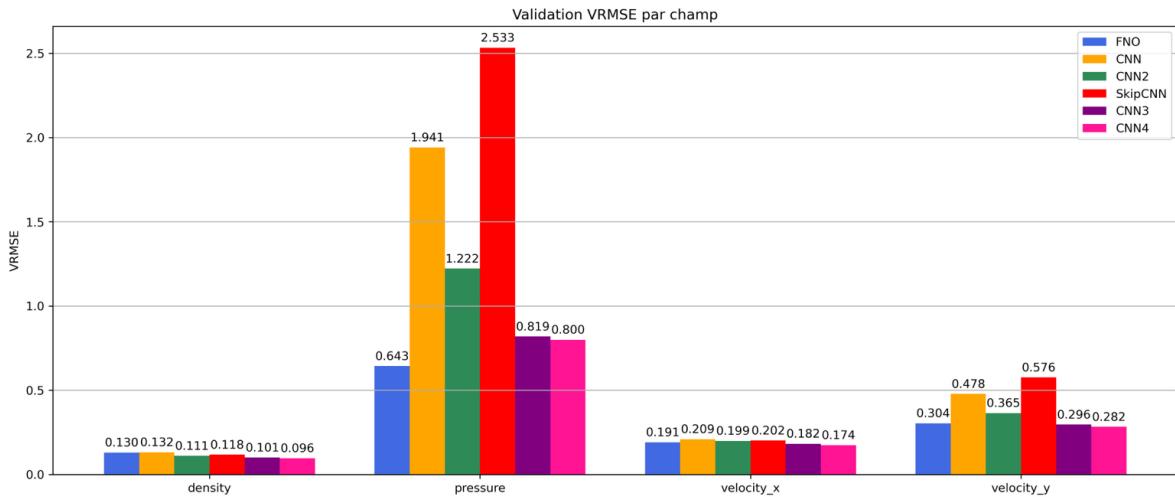


Figure 3 : Barplot des VRMSE moyens obtenus par les modèles sur les quatre champs physiques.

Ces résultats montrent que les architectures bien régularisées, telles que CNN4, peuvent surpasser des opérateurs plus sophistiqués comme le FNO lorsqu’elles sont correctement calibrées. L’analyse qualitative qui suit permettra de mieux visualiser la nature des prédictions et les effets des contraintes physiques intégrées dans CNN4.

## 4.2. PINN : Impact des contraintes physiques

### 4.2.1. Ajout de la conservation de la masse

Dans un premier temps, seule la contrainte de conservation de la masse a été intégrée, avec différents coefficients de pondération  $\beta_1$  (0.1, 0.5, 1.0). Les performances des modèles correspondants (PINN\_0.1, PINN\_0.5, PINN\_1.0) sont illustrées dans la Figure 4. Les résultats montrent que l’ajout de cette contrainte a un effet contrasté selon les champs physiques. Sur la densité, les performances se dégradent progressivement avec  $\beta_1$  (de 0.096 pour CNN4 à 0.233 pour  $\beta_1=1$ ). En revanche, le VRMSE sur la pression s’améliore légèrement avec une contrainte plus forte (de 0.866 pour  $\beta_1=0.1$  à 0.778 pour  $\beta_1=1$ ), ce qui peut témoigner d’un effet régularisant bénéfique sur ce champ. Les vitesses (en x et en y) sont peu affectées, avec une légère hausse de l’erreur mais sans changement drastique. Le modèle avec  $\beta_1=0.1$  reste globalement proche de CNN4, tandis que les valeurs un peu plus élevées de  $\beta_1$  produisent un compromis entre cohérence physique et perte de précision sur certains champs, notamment la densité.

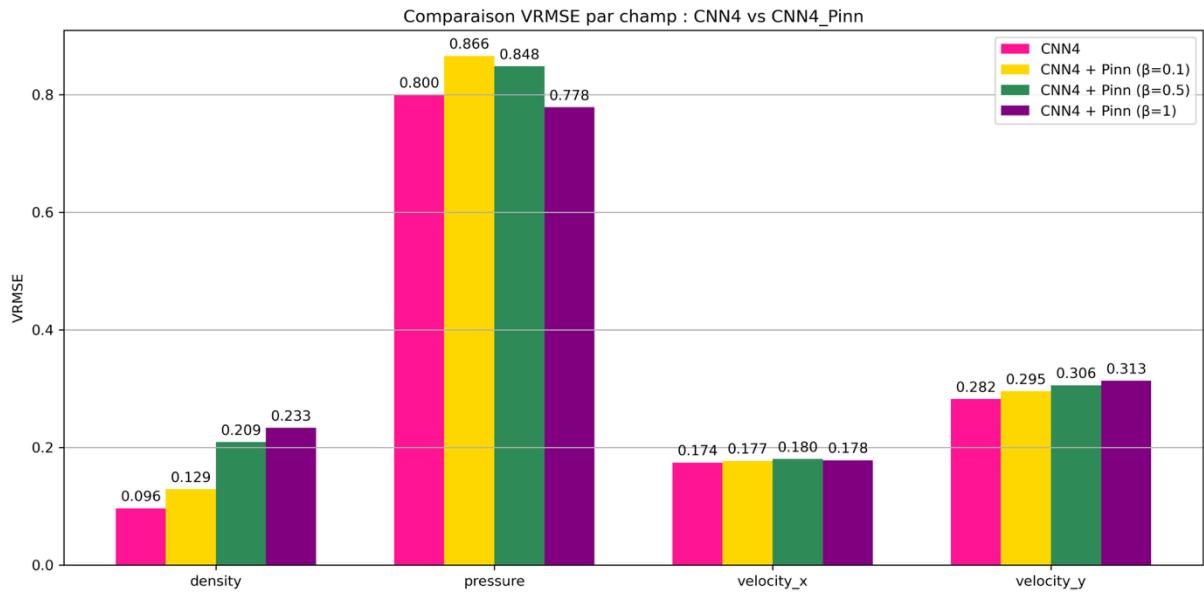


Figure 4 : Barplot des VRMSE moyens obtenus par les modèles CNN4 et CNN4+PINN ( $\beta=0.1, 0.5, 1$ ) sur les quatre champs physiques.

Pour évaluer dans quelle mesure les modèles respectent effectivement la loi de conservation de la masse, une divergence du flux de masse a été calculée numériquement sur les prédictions du réseau :  $\text{Div}_{masse} = \nabla \cdot (\rho v)$ . Dans un système conservatif cette grandeur devrait idéalement être nulle. En pratique, la valeur absolue moyenne de cette divergence sur l'ensemble de validation est utilisée comme indicateur de cohérence physique.

Modèle	Divergence moyenne
CNN4 simple	5.00e-2
CNN4+PINN 0.1	4.95e-2
CNN4+PINN 0.5	5.02e-2
CNN4+PINN 1.0	5.50e-2

Tableau 1 : Résultat des divergences moyennes pour les modèles CNN4 et CNN4+PINN ( $\beta=0.1, 0.5, 1$ ).

On observe que le modèle PINN\_0.1 réduit légèrement la divergence moyenne par rapport à CNN4 (4.95e-2 contre 5.00e-2), ce qui montre un léger gain de cohérence physique. Cependant, pour  $\beta_1=1$ , la divergence repart fortement à la hausse, atteignant 5.50e-2. Ce comportement peut être une première suggestion sur les valeurs des pondérations : lorsqu'elles sont élevées, elles peuvent déséquilibrer l'apprentissage.

#### 4.2.2. Ajout de la conservation de l'énergie totale avec refroidissement

Dans un second temps, une seconde contrainte PINN a été ajoutée au modèle PINN\_0.1 (le modèle ci-dessus avec  $\beta_1=0.1$ ), portant sur la conservation de l'énergie totale avec terme de refroidissement. Le nouveau modèle CNN4\_Pinn2 a été entraîné avec  $\beta_1=0.1$  pour la masse et  $\beta_2=0.01$  pour l'énergie. Les résultats (Figure 5) montrent une dégradation marquée sur le champ de pression, dont le VRMSE passe de 0.8659 à 1.2753. Les autres champs (densité, vitesse\_x, vitesse\_y) présentent une hausse plus modérée, de l'ordre de quelques points. Cela suggère que la contrainte d'énergie, bien que faiblement pondérée, perturbe l'équilibre du

training, affectant particulièrement la précision de la pression, probablement en raison de couplages forts dans l'équation de conservation.

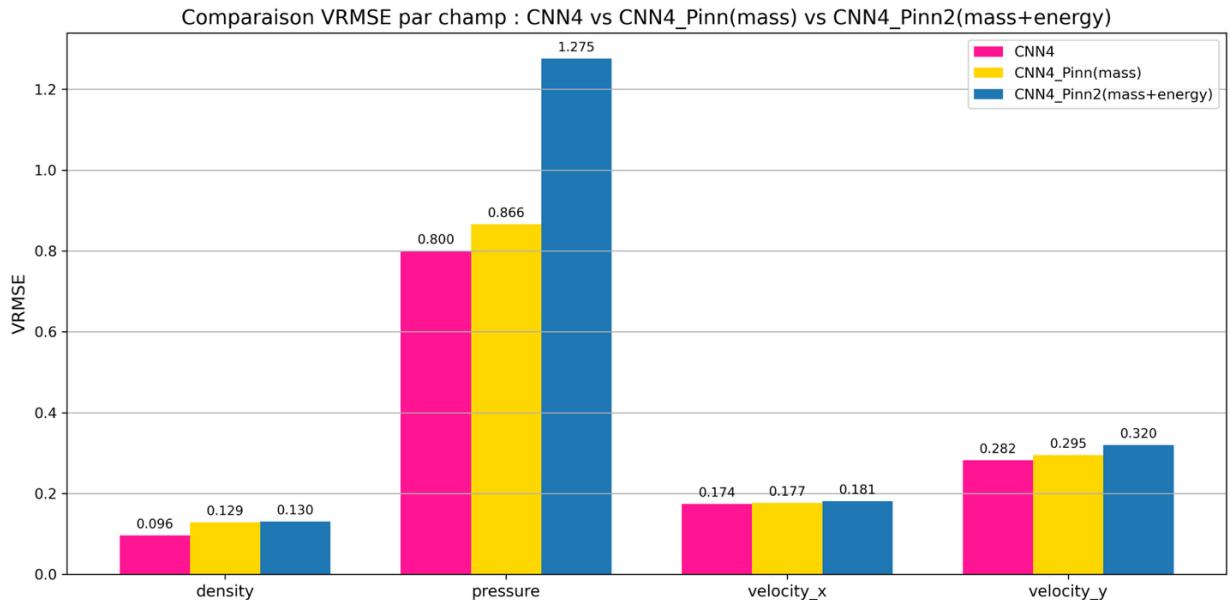


Figure 5 : Barplot des VRMSE moyens obtenus par les modèles CNN4, CNN4+PINN\_mass(beta=0.1) et CNN4+PINN\_energy(beta=0.01) sur les quatre champs physiques.

Ces expériences montrent que l'ajout de contraintes PINN permet d'améliorer légèrement la cohérence physique (réduction mesurée de la divergence de masse), mais au prix d'une dégradation quantitative des performances, surtout si les pondérations sont mal calibrées. Le compromis optimal se situe autour de  $\beta_1=0.1$ , sans contrainte énergétique. Cependant, il est important de souligner que le VRMSE ne constitue pas une métrique suffisante pour évaluer la qualité d'un modèle constraint par la physique (voir partie 5. Discussion). Un modèle comme le CNN4+PINN(masse, énergie), bien que moins performant selon cette métrique globale, peut néanmoins produire des champs plus cohérents sur le plan physique, préserver certaines structures fines, ou mieux respecter des symétries ou invariants locaux. Son intérêt ne peut donc être pleinement apprécié qu'à travers une analyse qualitative approfondie, complétée par l'observation du comportement en rollout temporel.

#### 4.3. Analyse qualitative

L'évaluation quantitative à l'aide du VRMSE permet de comparer globalement les performances des modèles, mais elle reste insuffisante pour juger de leur cohérence physique ou de leur capacité à reproduire des structures spatiales fines. Pour cette raison, une analyse qualitative a été menée afin d'examiner les prédictions des modèles sous un angle physique, visuel, et structurel, en se concentrant sur des grandeurs pertinentes pour la dynamique du système simulé. Cette analyse a porté sur quatre modèles : le modèle FNO, le modèle CNN4, le modèle CNN4\_PINN (avec contrainte PINN sur la masse seule), et le modèle CNN4\_PINN2 (avec contraintes PINN sur la masse et l'énergie). Les visualisations ont été réalisées sur l'ensemble de validation, en s'appuyant sur les profils moyens de pression et de température, et l'évaluation du Cold Mass Fraction (CMF).

#### 4.3.1. Profils moyens de pression et de température

L'analyse des profils moyens vise à évaluer la capacité des modèles à restituer la structure spatiale globale des champs physiques, indépendamment des fluctuations locales. Pour cela, les champs de pression et de température ont été moyennés selon l'axe y, puis représentés en fonction de la position normalisée  $x \in [0,1]$ . Les courbes obtenues ont été tracées pour l'ensemble du set de validation.

##### Profils moyens de pression

Le modèle CNN4 reproduit correctement le pic de pression et la décroissance en aval ( $x > 0.4$ ), mais présente une erreur notable dans les faibles valeurs de  $x$ , c'est-à-dire avant le pic. Dans cette région, la pression est largement surestimée, ce qui dégrade la qualité globale du profil malgré une bonne correspondance locale autour du minimum. Le modèle CNN4+PINN avec  $\beta=1$  suit bien la forme générale du ground truth. Toutefois, il surestime le minimum de pression, générant un creux plus prononcé que dans les données réelles. Malgré cela, le reste du profil reste bien aligné avec la courbe cible. Les modèles CNN4+PINN avec  $\beta=0.1$  et  $\beta=0.5$  présentent des variations parasites pour  $x < 0.1$ . Dans la zone centrale, ils sous-estiment le minimum de pression, tandis qu'aux grandes valeurs de  $x$ , ils produisent une surpression persistante. Cela montre une difficulté à maintenir une forme lisse et physiquement cohérente dans l'ensemble du profil. Le FNO, quant à lui, montre un profil globalement cohérent avec le ground truth, avec une pente bien reproduite. Néanmoins, il sous-estime légèrement le minimum de pression.

Comparaison des profils de pression moyenne par modèle

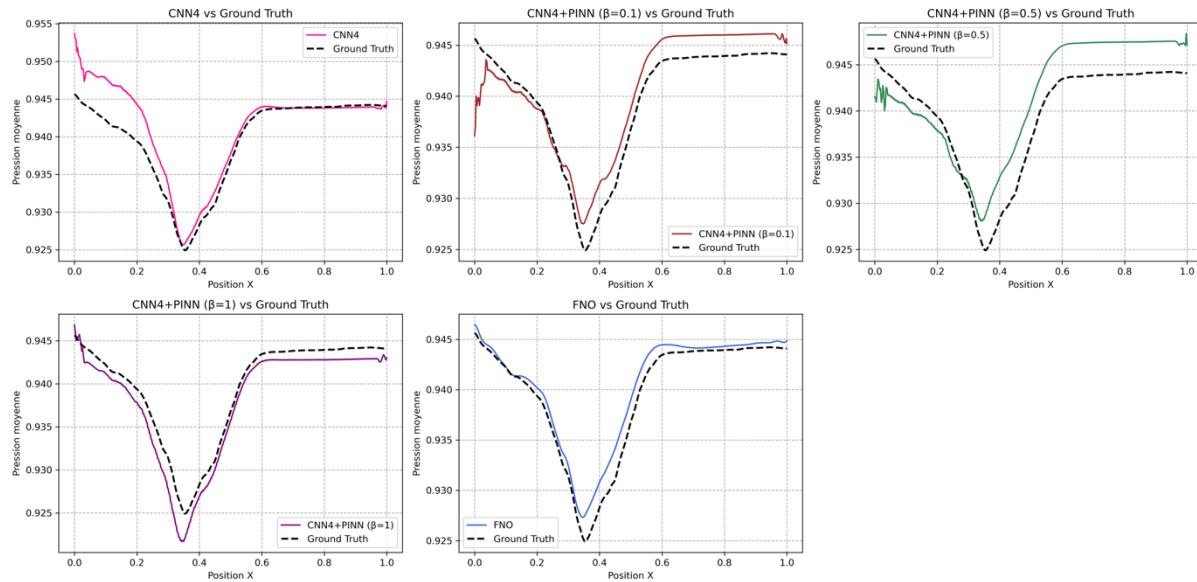


Figure 6 : Comparaison des profils moyens de pression entre modèles et ground truth - Validation

##### Profils moyens de température

La température n'étant pas directement présente dans le dataset, elle a été estimée à partir de la relation des gaz parfaits sous forme adimensionnée :  $T \propto P/\rho$  où  $P$  est la pression prédictée et

$\rho$  la densité prédictive. Les courbes sont présentées en échelle logarithmique pour mieux faire apparaître les zones de transition froid/chaud.

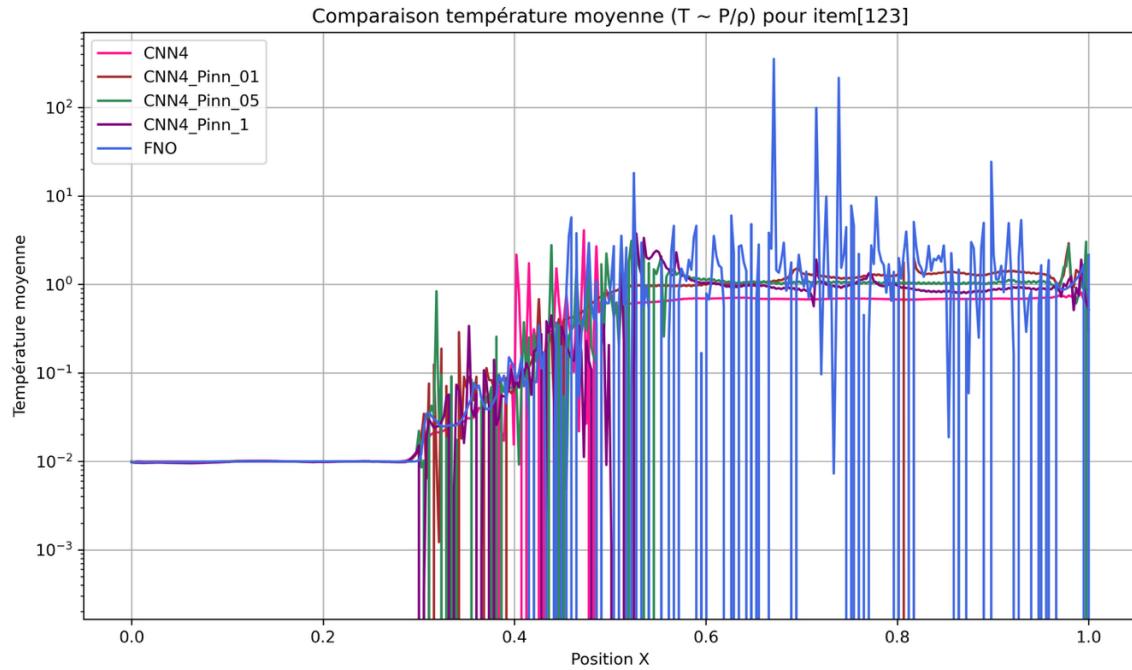


Figure 7 : Illustration des instabilités pour le profil moyen de température sur un item individuel du dataset

Cette opération, physiquement légitime, peut toutefois générer des instabilités numériques dans les zones à faible densité, notamment dans les couches chaudes et diffuses. Ce phénomène est observé sur la Figure 7. Pour éviter ces artefacts, un filtrage a été appliqué : seuls les points pour lesquels  $\rho > 0.001$  sont conservés pour le calcul de la température. De plus, afin d'obtenir des courbes plus lisibles, un lissage spatial a été réalisé sur les profils moyens de température.

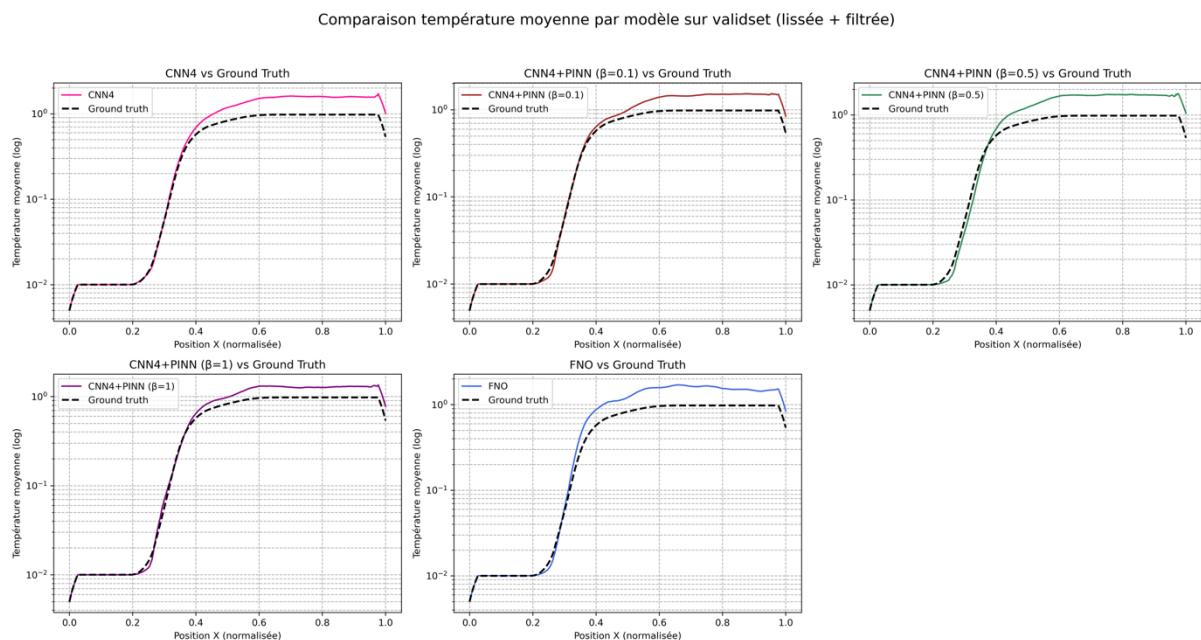


Figure 8 : Comparaison des profils moyens de température entre modèles et ground truth - Validation

Les profils ainsi filtrés et lissés permettent une comparaison stable entre les modèles. Jusqu'à  $x=0.4$ , les prédictions suivent correctement la référence, mais s'en écartent ensuite progressivement (surestimation de la température dans les zones chaudes). Parmi les modèles testés, CNN4+PINN avec  $\beta=1$  est celui qui suit le plus fidèlement la pente de la transition thermique, avec un bon ajustement dans la zone de mélange. Le FNO, de son côté, présente une pente plus abrupte, ce qui indique une transition thermique un peu plus rapide par rapport au ground truth. Tous les modèles tendent à légèrement surestimer la température dans les régions chaudes.

### Modèle final avec contraintes de masse et d'énergie

Le modèle CNN4+PINN2, qui combine simultanément des contraintes sur la conservation de la masse et de l'énergie, a également été évalué sur les profils moyens. Sur le champ de pression, il suit correctement la forme autour du minimum du ground truth. Il présente quelques fluctuations parasites pour les  $x$  proches de 0 et de 1. Et, il a tendance à sous-estimer la pression dans les zones chaudes. Concernant la température, le profil du modèle est cohérent avec celui du ground truth, aussi bien dans la pente de la transition thermique que dans les zones extrêmes. La courbe suit correctement la zone de mélange, sans surestimation marquée dans les régions chaudes.

Comparaison des profils moyens - CNN4+PINN2(mass+energy) vs Ground truth

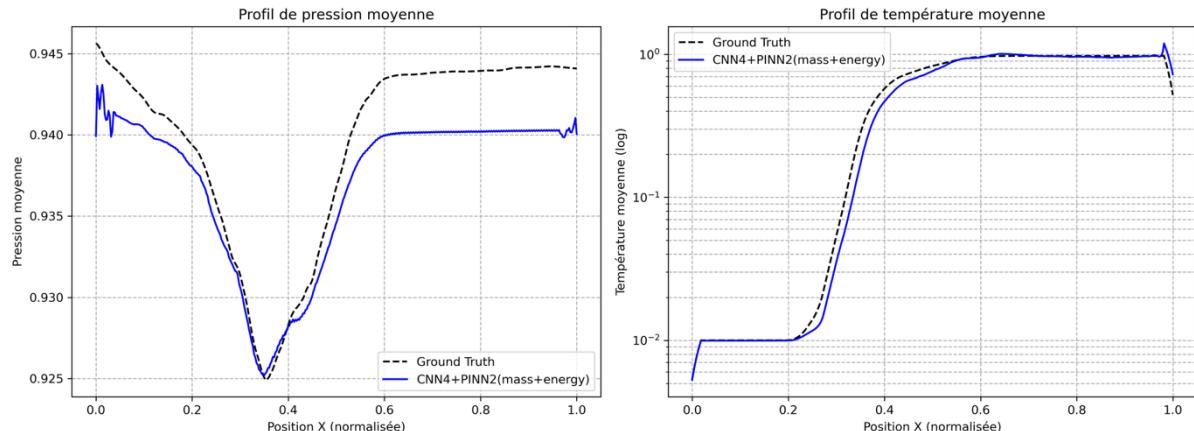


Figure 9 : Comparaison des profils moyens de pression et de température pour le modèle PINN (masse et énergie)

L'ajout de contraintes PINN, qu'elles portent sur la masse seule ou sur la masse et l'énergie, permet d'introduire davantage de cohérence physique dans les prédictions. Une pondération modérée ( $\beta=0.1$ ) montre un bon compromis entre VRMSE et respect des lois physiques. Toutefois, une pondération plus élevée comme  $\beta=1$  améliore nettement la qualité des prédictions, en particulier sur les profils moyens de pression et de température, qui suivent fidèlement la structure du ground truth. Cela montre que le VRMSE ne suffit pas à lui seul à évaluer la pertinence physique d'un modèle.

#### 4.3.2. « Cold Mass Fraction » (CMF)

Au-delà des profils locaux de pression et de température, il est pertinent d'évaluer la capacité des modèles à préserver certaines propriétés globales du système simulé. Parmi celles-ci, la CMF constitue une mesure intégrée essentielle pour caractériser le comportement

thermodynamique d'un gaz en refroidissement. La CMF est défini comme la fraction de masse dont la température est inférieure à la moitié de celle du gaz chaud. Cette dernière est estimée à partir de la moyenne de la température dans la région chaude du domaine : on considère la région  $x>0.80$ . L'indicateur ainsi obtenu mesure la proportion de matière effectivement refroidie, ce qui est particulièrement pertinent dans le contexte d'un système turbulent soumis à un refroidissement radiatif intense, tel que celui modélisé ici. Les valeurs de la CMF ont été calculées pour chaque modèle sur l'ensemble du set de validation, à partir des champs de température. Les résultats montrent que tous les modèles testés préservent une fraction froide élevée, supérieure à 98 %. Cette performance globale suggère que les réseaux de neurones convolutifs, y compris pour les PINN, parviennent à restituer de manière robuste le phénomène de refroidissement dominant.

Toutefois, le modèle CNN4+PINN avec  $\beta=0.5$  atteint un CMF de 0.9882, légèrement supérieur à celui du modèle CNN4 simple (0.9857), indiquant une meilleure restitution de la masse froide. Le modèle FNO obtient un résultat similaire (0.9883). À l'inverse, le modèle CNN4+PINN2, qui combine à la fois des contraintes sur la masse et l'énergie, affiche le CMF le plus faible (0.9833). Cette valeur, bien que toujours élevée, traduit une légère perte de matière froide.

#### 4.4. Rollouts temporels

L'évaluation des modèles sur un unique pas de temps ne permet pas, à elle seule, de juger leur stabilité lorsque ceux-ci sont utilisés de manière autorégressive. Dans un contexte plus réaliste, le modèle est sollicité pour prédire successivement plusieurs étapes de la dynamique, en reinjectant à chaque pas sa propre sortie comme entrée. Ce processus, appelé rollout temporel, est particulièrement exigeant car les erreurs peuvent s'accumuler rapidement, provoquant une dérive ou une perte de cohérence physique. Nous avons évalué six modèles : CNN4, CNN4+PINN (avec  $\beta_1=0.1, 0.5, 1$ ), CNN4+PINN2 (avec  $\beta_1=0.1$  et  $\beta_2=0.01$ ), et FNO. L'évaluation a été réalisée sur 30 pas de rollout. À chaque étape, les prédictions sont comparées à une référence fixe (l'état cible  $t=1$ ), choisie volontairement pour tester la stabilité intrinsèque des modèles, indépendamment de l'évolution naturelle du système. Cette partie montre la capacité des modèles à maintenir une solution plausible autour d'un état de référence lorsqu'ils sont confrontés à l'auto-régression.

##### 4.4.1. Analyse quantitative : MSE et VRMSE

Les courbes de la figure ci-dessous présentent l'évolution du MSE et du VRMSE au cours du rollout. Sans surprise, toutes les architectures montrent une augmentation de l'erreur au fil du temps. Le FNO présente la dérive la plus rapide, avec une erreur qui croît fortement dès les premiers pas et reste supérieur à tous les autres modèles durant tout le rollout. De plus, la courbe du FNO présente un minimum local pour  $t=16$  avant de repartir sur une croissance. Le CNN4 suit une croissance plus régulière, mais sans amortissement notable.

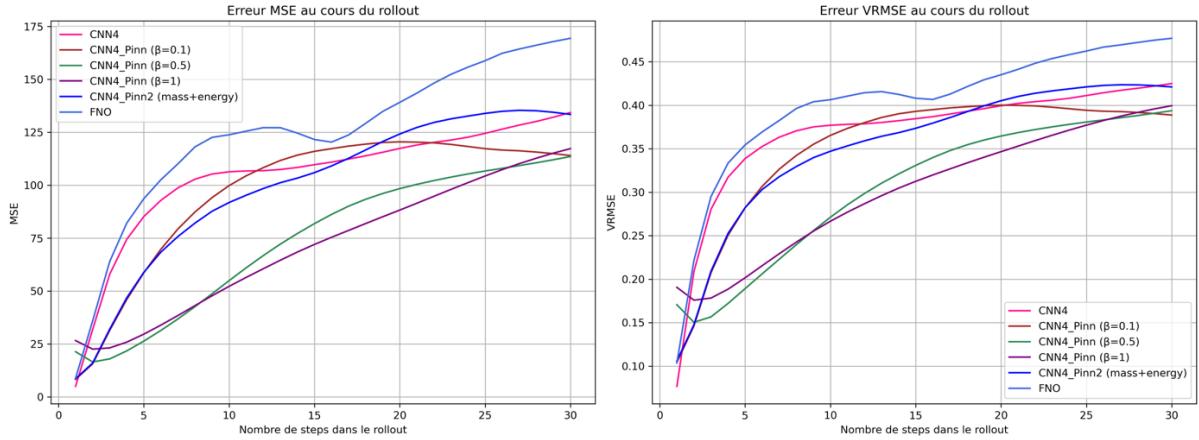


Figure 10 : Évolution des erreurs MSE et VRMSE au cours des rollouts autorégressifs sur 30 pas de temps

Par contraste, les modèles CNN4+PINN avec  $\beta=0.5$  et surtout  $\beta=1.0$  montrent une croissance plus lente de l'erreur, en particulier au-delà de  $t=10$ . Le modèle avec  $\beta=0.1$  illustre une courbe concave. Après  $t=20$ , la courbe devient décroissante. Ce comportement est difficile à interpréter étant donné que les erreurs devraient s'accumuler au fil des rollouts. Le modèle CNN4+PINN2, qui combine des contraintes de masse et d'énergie, montre également une dynamique particulière avec une stabilisation vers  $t=25$ .

Globalement, les courbes MSE/VRMSE de ces modèles n'ont pas les mêmes comportements au cours des rollouts.

#### 4.4.2. Analyse qualitative : évolution du champs de densité

Pour compléter ces résultats, nous avons visualisé les champs de densité prédits aux étapes  $t=0, 3, 5, 7, 10$ . Ces images révèlent des différences significatives entre modèles. Le CNN4, bien qu'il conserve la structure générale du front turbulent jusqu'à  $t=5$ , montre une dégradation rapide des détails à partir de  $t=7$ , avec un aplatissement progressif du champ. Le modèle CNN4+PINN avec  $\beta=0.1$  montre une meilleure préservation des structures fines : le filament devient plus net au fil du temps. Les versions avec  $\beta=0.5$  et  $\beta=1.0$  présentent des champs plus lisses et moins détaillés. Une chose qu'on peut noter est que ces trois modèles avec PINN ne montrent pas une évolution temporelle du système : pour ces modèles, les 5 images sont dynamiquement similaires.

Le comportement du modèle CNN4+PINN2 est plus nuancé. Visuellement, il conserve la forme globale du front de mélange, mais perd en expressivité spatiale.

Enfin, le modèle FNO parvient à capturer des dynamiques plus proches du ground truth. Le front turbulent conserve ses motifs tourbillonnants, et la zone de mélange verte enveloppe progressivement la région violette, ce qui correspond bien à l'instabilité de Kelvin–Helmholtz. Ce comportement dynamique est un bon indicateur de la capacité du modèle à simuler l'évolution réelle du système.

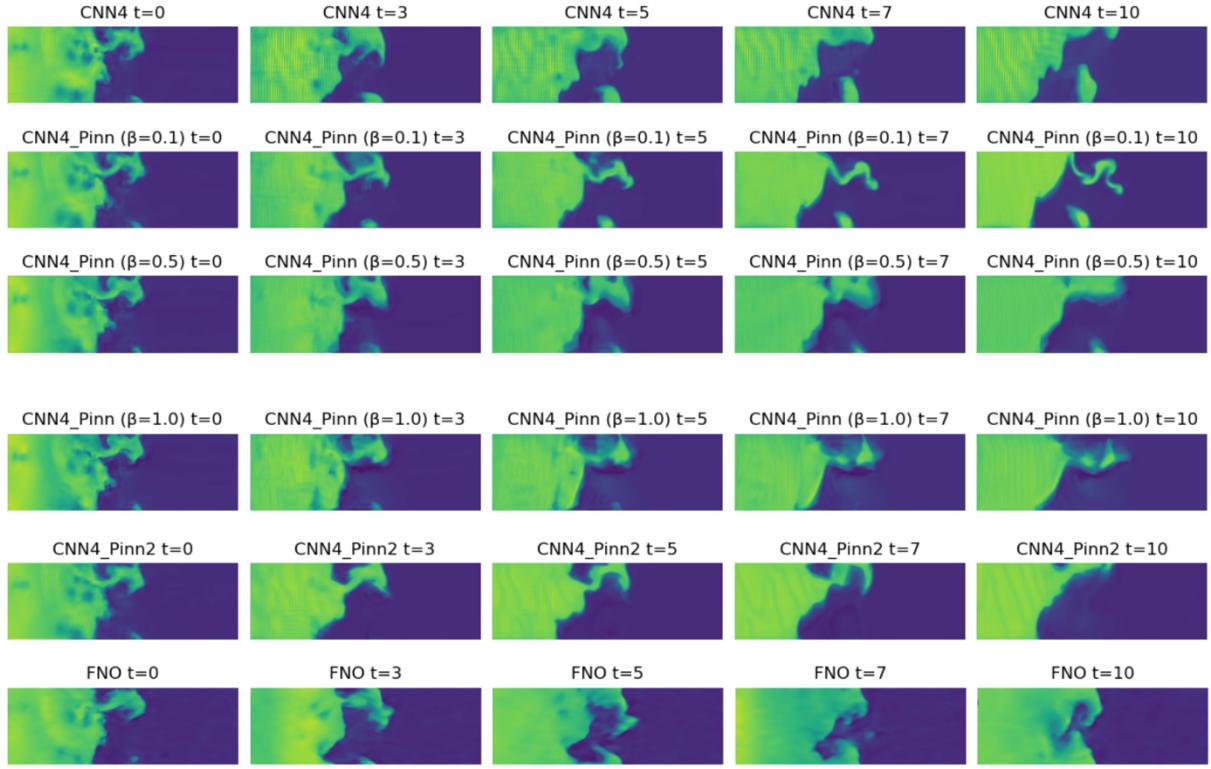


Figure 11 : Évolution des champs de densité prédis au cours du rollout autorégressif ( $t=0, 3, 5, 7, 10$ )

Afin de comparer ces modèles au cours du rollout, il aurait été pertinent d'afficher l'erreur par rapport au ground truth à chaque pas de temps. Cependant, le dataset de validation ne fournit que des sorties pour le pas de prédiction immédiat (one-step prediction). Malgré cela, nous avons représenté dans le notebook associé l'erreur entre chaque prédiction et le ground truth pour  $t=1$ . Bien que cette erreur ne reflète pas directement la dynamique du système, elle permet d'estimer si les structures instables comme le tourbillon sont correctement capturées à différents moments. Une erreur faible dans cette zone signale un comportement statique tandis qu'une erreur localisée au niveau de l'instabilité peut indiquer une dynamique correctement amorcée.

## 5. Discussion

### 5.1. Pertinence de la métrique VRMSE

Dans ce travail, l'évaluation quantitative des performances a reposé principalement sur la VRMSE, une métrique proposée et utilisée par défaut dans le benchmark The Well. Cette mesure a l'avantage d'être invariante à l'échelle, en normalisant l'erreur quadratique moyenne par l'écart-type du champ de référence. Cela permet de comparer des champs aux dynamiques très différentes (par exemple, pression vs vitesse), ce qui est pertinent dans une approche multi-physique.

Cependant, la VRMSE présente aussi des limites. En particulier, elle peut atténuer des erreurs absolues significatives si le champ considéré est fortement variable (grande variance), ou au contraire amplifier de petites erreurs sur des champs plus homogènes. Elle reste donc sensible à la distribution statistique des cibles, sans toujours refléter l'importance physique des écarts. Par exemple, une erreur localisée sur un front de choc ou une interface turbulente peut avoir un impact physique important, mais n'avoir qu'une contribution limitée à la VRMSE globale.

De plus, dans le contexte des rollouts, où les prédictions deviennent parfois lissées ou saturées, la variance des champs cibles reste constante alors que celle des prédictions décroît : cela peut fausser l'interprétation de la stabilité réelle du modèle.

### 5.2. Influence de la normalisation manuelle

Dans ce projet, les champs physiques ont été normalisés manuellement à l'aide d'un sous-échantillon du dataset. Plus précisément, 20 trajectoires ont été extraites à intervalles réguliers, et les tenseurs d'entrée `input_fields` ont été concaténés pour calculer la moyenne `mu` et l'écart-type `sigma` de chaque variable physique. Cette normalisation est ensuite appliquée globalement à toutes les données avec les fonctions `preprocess` et `postprocess`. Ce choix sert à homogénéiser les échelles entre les champs physiques pour stabiliser la descente de gradient. Il est d'autant plus important ici que les modèles doivent apprendre conjointement à prédire plusieurs variables dont les ordres de grandeur diffèrent sensiblement.

Cependant, cette méthode suppose implicitement que les statistiques des 20 trajectoires sélectionnées soient représentatives du dataset complet, ce qui peut ne pas être garanti dans un système turbulent et non stationnaire. En effet, les fronts d'instabilité ou les phases de refroidissement rapide peuvent induire des distributions locales très différentes, mal capturées par une normalisation globale. De plus, normaliser chaque champ indépendamment revient à ignorer les dépendances physiques existantes entre eux (par exemple, la relation  $T \propto P/\rho$ ).

### 5.3. Limite de l'évaluation par rollout statique

Dans notre évaluation, les rollouts ont été comparés à une cible temporelle statique obtenue par répétition du champ de sortie `y`. C'est-à-dire que chaque prédiction multi-step a été évaluée en mesurant son écart à une solution figée, supposée constante dans le temps. Ce choix simplifie l'analyse en permettant de tester la stabilité numérique des modèles (capacité à ne pas diverger lorsqu'ils sont réappliqués sur leur propre sortie), mais il présente des limites importantes dans le cadre d'une dynamique turbulente.

En effet, les champs physiques considérés dans le dataset `The Well` évoluent de manière non triviale dans le temps. Ainsi, prendre une référence statique revient à ignorer cette dynamique, et à pénaliser à tort les modèles qui la suivent correctement. À l'inverse, un modèle incapable de reproduire la physique mais qui converge vers un état moyen pourra obtenir une erreur faible simplement parce qu'il se rapproche de cette "cible moyenne". Ainsi, l'évaluation par rollout statique est plus un test de stabilité que de fidélité dynamique, et peut conduire à une interprétation biaisée de la performance réelle. Une meilleure évaluation consisterait à comparer les rollouts aux vraies séquences temporelles extraites du dataset.

## 6. Conclusion

Ce travail a exploré l'apprentissage profond pour la modélisation de dynamiques physiques complexes, en s'appuyant sur le benchmark `The Well` et le cas d'étude `turbulent_radial_layer_2D`. L'objectif était d'évaluer la capacité de différentes architectures de réseaux de neurones, avec ou sans contraintes physiques, à prédire l'évolution spatio-temporelle d'un système turbulent.

Dans un premier temps, nous avons comparé plusieurs modèles, montrant que la profondeur et les mécanismes de skip connections améliorent significativement la précision. Le modèle final, intitulé CNN4 dans ce projet, s'est révélé le plus performant en prédiction à court terme, surpassant même le modèle FNO (implémenté par The Well) sur plusieurs champs physiques. Cette performance s'explique par sa capacité à capturer finement les structures locales et à maintenir une convergence stable.

Dans un second temps, des contraintes physiques ont été intégrées au modèle CNN4. L'ajout de la conservation de la masse a permis de stabiliser partiellement les rollouts, mais au prix d'une perte de diversité dynamique. L'introduction simultanée d'une contrainte sur l'énergie a accentué cet effet de régularisation excessive, conduisant à des prédictions trop lisses.

Les résultats ont également révélé certaines limites du cadre méthodologique utilisé : la métrique VRMSE, bien que standardisée, ne permet pas d'évaluer pleinement la fidélité physique des prédictions ; la normalisation globale peut masquer des comportements locaux rares. Ces observations invitent à compléter l'évaluation des modèles par des métriques structurelles. De plus, on pourrait passer au dataset 3D, disponible dans les données The Well (*turbulent\_radiative\_layer\_3D*) afin de comparer avec les résultats de *Fielding et al. (2020)*.

Finalement, ce projet a permis de montrer les forces et les limites de l'apprentissage dans un contexte de prédictions de la physique. Il souligne l'importance d'un calibrage précis entre précision numérique et cohérence physique, et ouvre des perspectives vers des architectures hybrides. L'approche PINN nécessite encore des ajustements fins pour exploiter pleinement son potentiel dans cette étude.

## 7. Références bibliographiques

- Ohana, R., et al. (2024). *The Well: A comprehensive benchmark for surrogate modeling of complex physical systems*. arXiv:2412.00568v2
- Fielding, D. B., et al. (2020). *Radiative Cooling Layers in Turbulent Astrophysical Flows*. ApJL 894, L24
- Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A. M., & Anandkumar, A. (2021). *Fourier Neural Operator for Parametric Partial Differential Equations*. In *International Conference on Learning Representations (ICLR)*. arXiv:2010.08895.
- Raissi, M., Perdikaris, P., Karniadakis, G. E. (2019). *Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear PDEs*. Journal of Computational Physics, 378

## 8. Environnement de travail

- Jupyter Notebook
- Site web de Polymathic AI pour s'initier avec les données
- LLM pour la reformulation de texte

## 9. Annexe : Description détaillée des modèles utilisés

Cette annexe présente (en détail) les différents modèles implémentés dans ce projet.

- **CNN :**

Le modèle CNN constitue l'architecture de référence minimale. Il se compose d'un encodage séquentiel suivi d'une remontée de résolution simple :

- Encodage initial : deux blocs convolutionnels Conv2d → ReLU → BatchNorm (64 puis 128 canaux), suivis d'un MaxPool2d réduisant la résolution spatiale de moitié ( $128 \times 384 \rightarrow 64 \times 192$ ), puis d'un Dropout.
- Transformation (bottleneck) : deux convolutions supplémentaires (128→128 puis 128→64) avec normalisation et activation identiques.
- Décodage : un ConvTranspose2d (64→64) permet de revenir à la résolution initiale, suivi d'une dernière convolution (64 → 4) pour produire les champs de sortie.

- **CNN2 :**

CNN2 reprend l'architecture de base en y ajoutant des couches supplémentaires :

- Trois blocs de convolution dans l'encodeur (jusqu'à 128 canaux), avec une convolution supplémentaire 128→128 par rapport à CNN.
- Une montée à 256 canaux, puis une réduction progressive vers 64.
- Un ConvTranspose2d (64→64) .
- Une sortie identique au modèle de base.

- **SkipCNN :**

Ce modèle introduit une connexion de type U-Net entre l'encodeur initial et le décodeur final :

- Encodage : deux blocs Conv2d → ReLU → BatchNorm (64→128), suivis d'un MaxPool2d et d'un Dropout.
- Transformation : deux convolutions supplémentaires (128→128→64).
- Décodage : la sortie upsample (64 canaux) est concaténée avec le tenseur de l'encodeur initial (128 canaux), puis traitée par une convolution (192→64) et une dernière couche de sortie (64→4).

- **CNN3 :**

On a ici une structure U-Net améliorée :

- Encodage : trois convolutions (jusqu'à 128 canaux) avec activation SiLU (fonction lisse) et GroupNorm (8 groupes), plus stable que BatchNorm sur de petits batchs.
- Bottleneck : compression par MaxPool2d, montée à 256 canaux puis redescendre à 64.
- Décodage : upsampling, concaténation avec le skip, décodeur (192→64), suivi d'un bloc de raffinement supplémentaire (64→64).
- Sortie : dernière convolution (64→4).

- **CNN4 :**

C'est la version la plus avancée. CNN4 a été retenu pour l'intégration des contraintes physiques (voir section 4.2).

- Encodage profond : quatre convolutions successives (jusqu'à 128 canaux), suivies d'un bloc de compression à 256 canaux puis redescense vers 64, avec MaxPool2d et Dropout2d.
- Décodage U-Net : upsampling, concaténation avec le skip, décodeur ( $192 \rightarrow 128 \rightarrow 64$ ).
- Raffinement final : deux couches supplémentaires ( $64 \rightarrow 64$ ) affinent la prédiction.
- Activation et normalisation : SiLU + GroupNorm(8) dans toutes les couches.