Microsoft

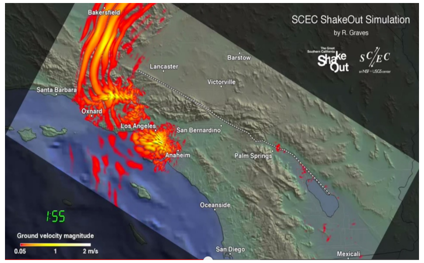
Modeling Accurate Long Rollouts with Temporal Neural PDE Solvers

Phillip Lippe[1,2], Bas S. Veeling[1], Paris Perdikaris[1], Richard E. Turner[1], Johannes Brandstetter[1]

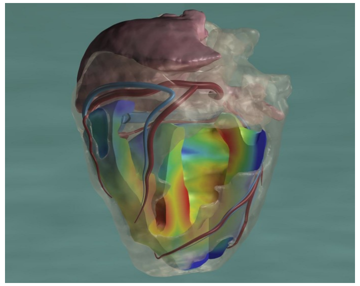
[1]Microsoft Research AI4Science, [2]University of Amsterdam

Project Website

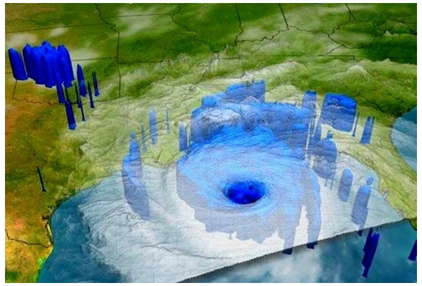
(Large-scale) PDE systems are ubiquitous



Earthquakes

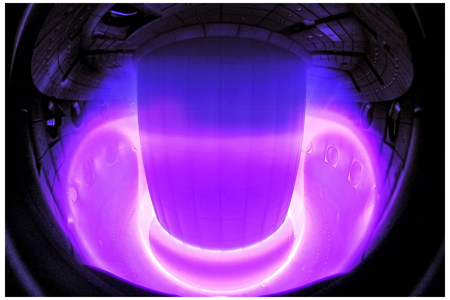


Heart dynamics

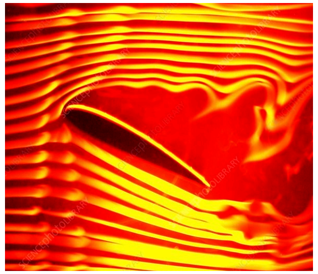


Weather prediction

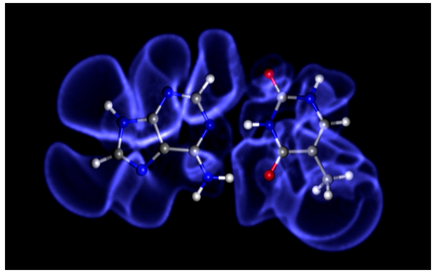
Galaxy collisions



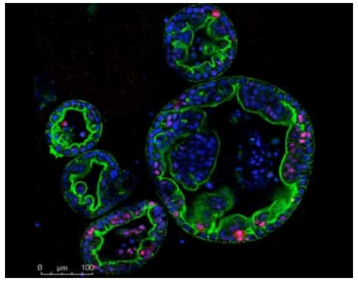
Plasma physics



Airplane design



Electronic structure

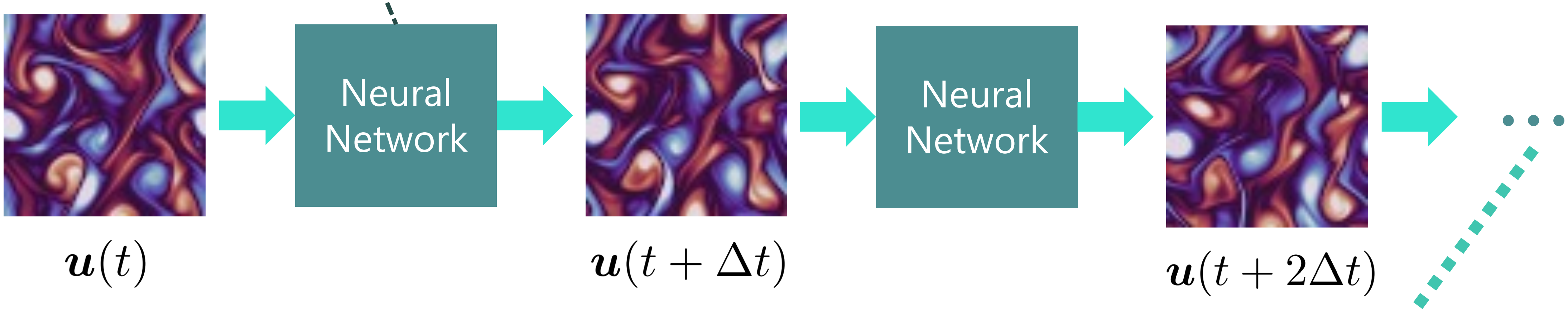


Tumor growth

Neural PDE Solvers

- Neural Operators learn to predict future solutions

$$\boldsymbol{u}(t + \Delta t) = \mathcal{G}_t(\Delta t, \boldsymbol{u}(t))$$



- Trained on one-step predictions
- Long horizon predictions via autoregressive rollout

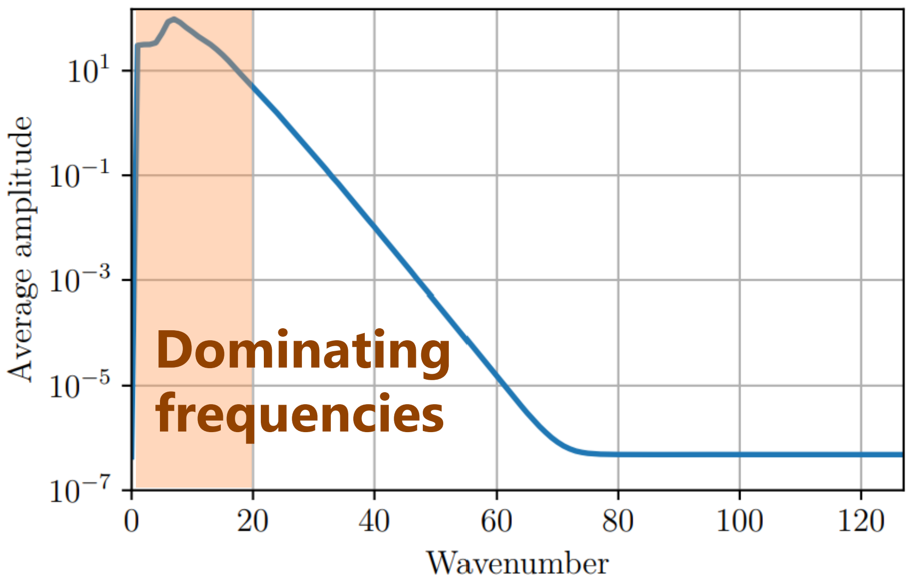
How can Neural Operators obtain long accurate rollouts?

Case Study: Kuramoto-Sivashinsky

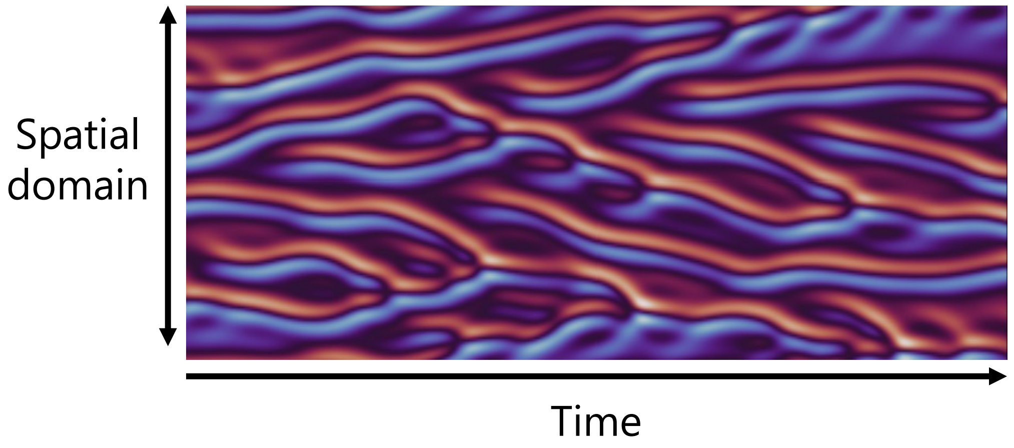
- Example: 1D Kuramoto-Sivashinsky equation (KS)

$$u_t + uu_x + u_{xx} + \nu u_{xxxx} = 0.$$

Spatial Frequency Spectrum



Example Trajectory



Case Study: Kuramoto-Sivashinsky

- Example: 1D Kuramoto-Sivashinsky equation (KS)

$$u_t + \boxed{uu_x} + \boxed{u_{xx} + \nu u_{xxxx}} = 0.$$

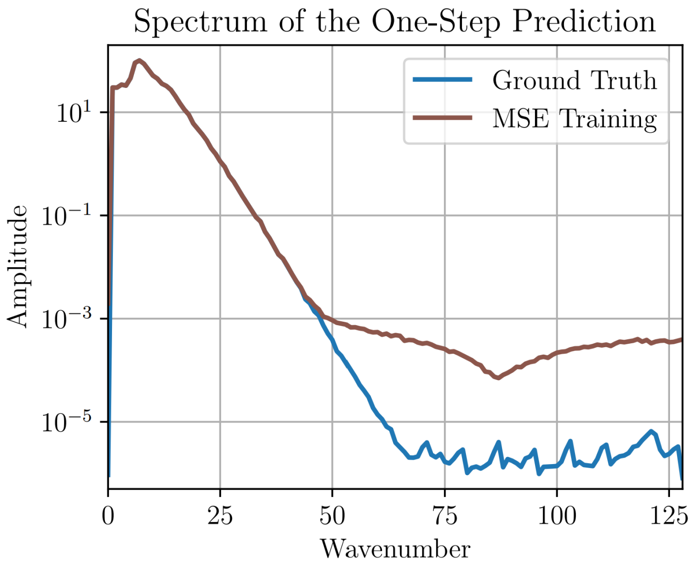
Non-linear term causes all spatial frequencies to interact long-term

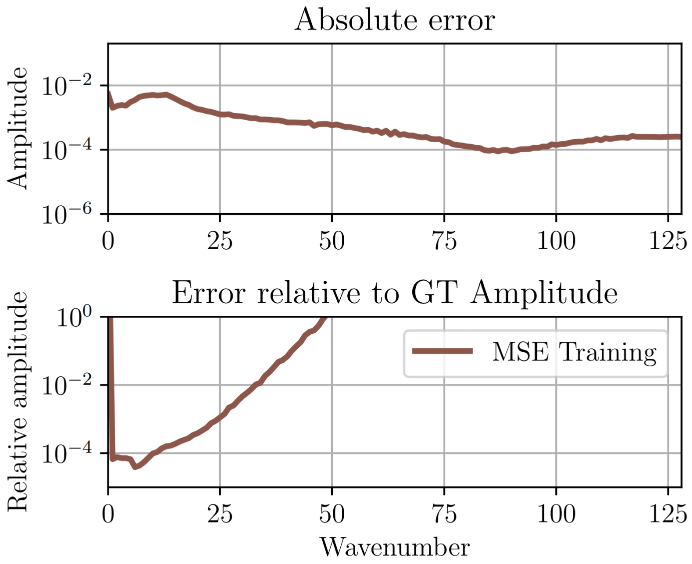
High-order derivatives increase importance of high frequencies in spatial domain

For long accurate rollouts, model **all** spatial frequencies accurately
Errors in higher frequencies have low short-term, but **high long-term impact**

Case Study: Kuramoto-Sivashinsky

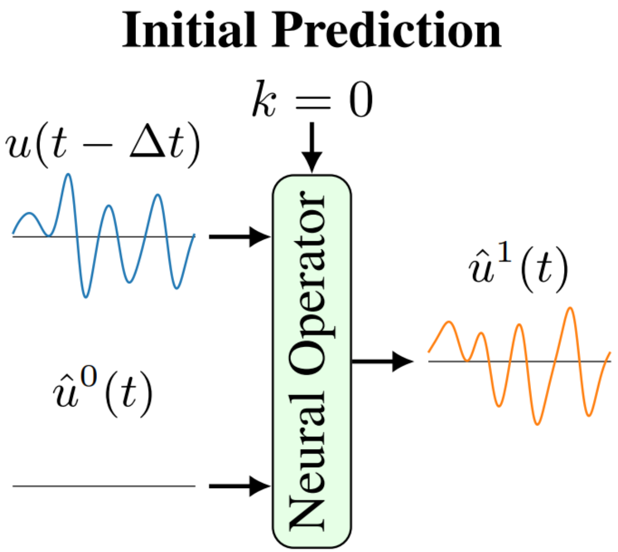
- How well do MSE-trained surrogates cover the frequency spectrum?





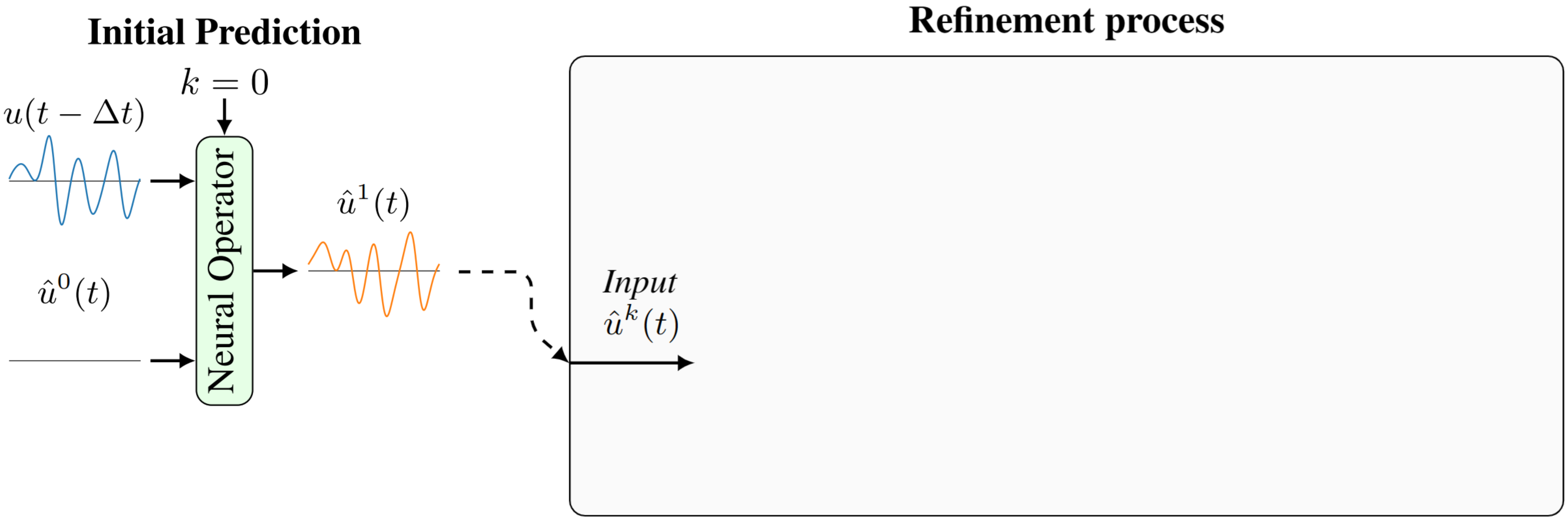
- Neural surrogates focus on **dominating** frequencies, losing high frequencies
- Inherently limits the maximum rollout time

PDE-Refiner

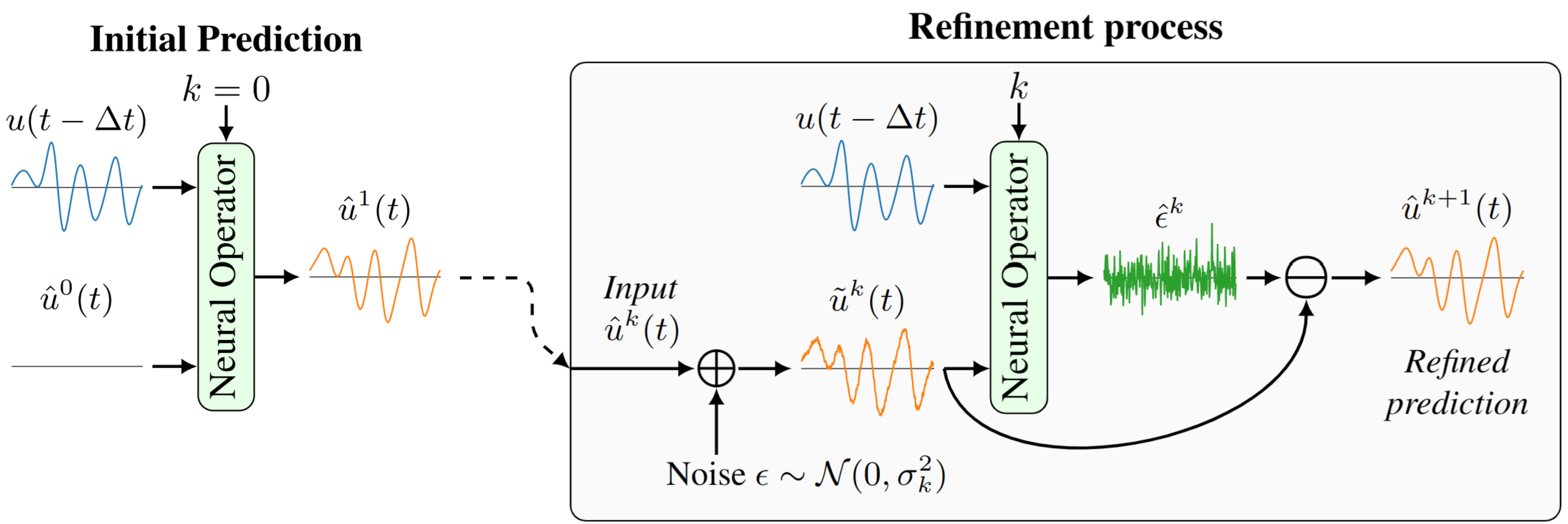


Initial Prediction
$k = 0$
$u(t - \Delta t)$
$\hat{u}^0(t)$
Neural Operator
$\hat{u}^1(t)$

PDE-Refiner

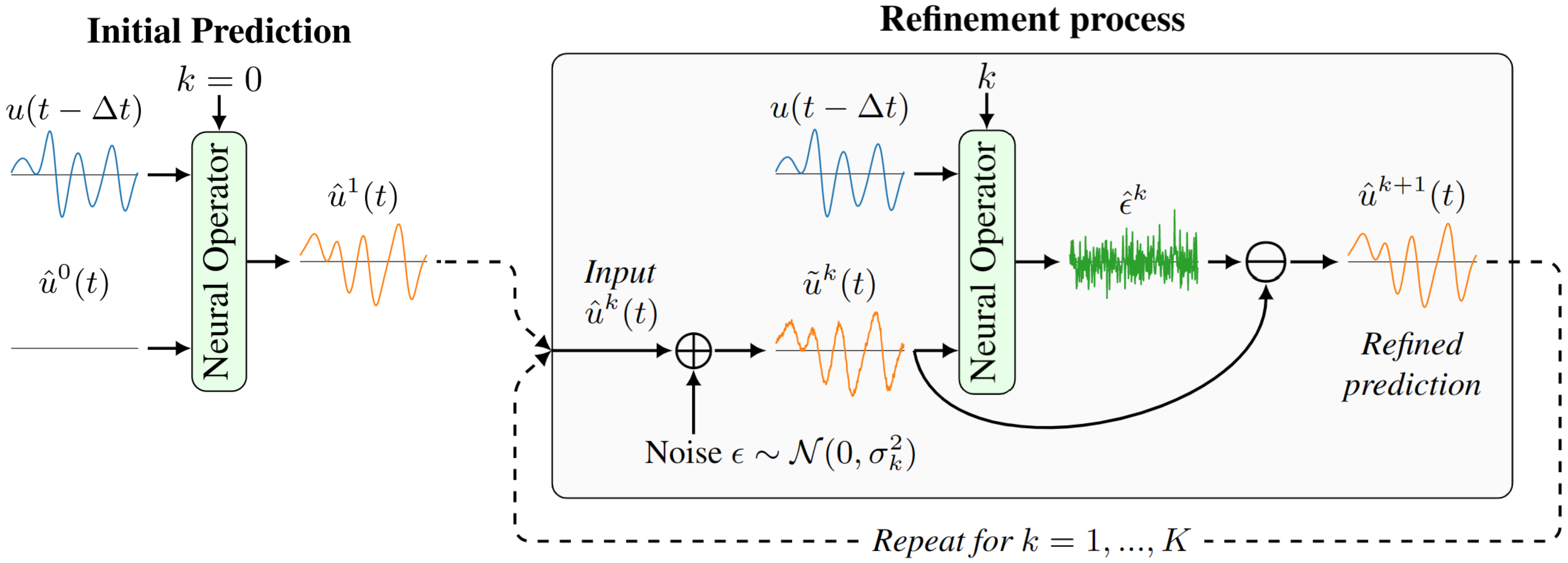

Initial Prediction
Refinement process
$k = 0$
$u(t - \Delta t)$
$\hat{u}^0(t)$
Neural Operator
$\hat{u}^1(t)$
Input
$\hat{u}^k(t)$

PDE-Refiner



Initial Prediction
$k = 0$
$u(t - \Delta t)$
Neural Operator
$\hat{u}^1(t)$
$\hat{u}^0(t)$
Refinement process
k
$u(t - \Delta t)$
Neural Operator
$\hat{\epsilon}^k$
$\hat{u}^{k+1}(t)$
Input
$\hat{u}^k(t)$
$\tilde{u}^k(t)$
Refined prediction
Noise $\epsilon \sim \mathcal{N}(0, \sigma_k^2)$

PDE-Refiner


Initial Prediction
$k = 0$
$u(t - \Delta t)$
$\hat{u}^0(t)$
Neural Operator
$\hat{u}^1(t)$
Refinement process
k
$u(t - \Delta t)$
Neural Operator
$\hat{\epsilon}^k$
$\hat{u}^{k+1}(t)$
Input
$\hat{u}^k(t)$
$\tilde{u}^k(t)$
Refined prediction
Noise $\epsilon \sim \mathcal{N}(0, \sigma_k^2)$
Repeat for $k = 1, \ldots, K$

PDE-Refiner – Relation to Diffusion Models

- Popular usage of denoising: Diffusion Models (DDPM) [Ho et al., 2020]

- Key differences of PDE-Refiner to DDPMs:
  1. GT is deterministic ⇒ exponential decreasing noise schedule with very small minimum
  2. Speed is of essence for application ⇒ very few denoising steps (usually 1-4)
  3. Different objective ⇒ predicts signal at initial step

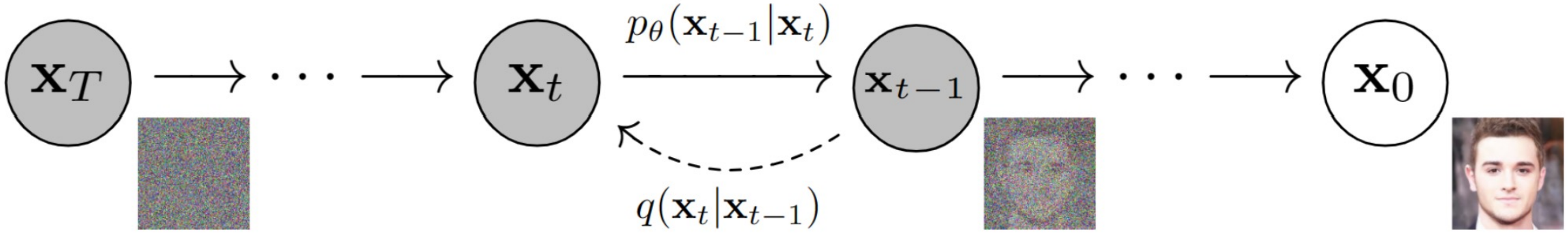
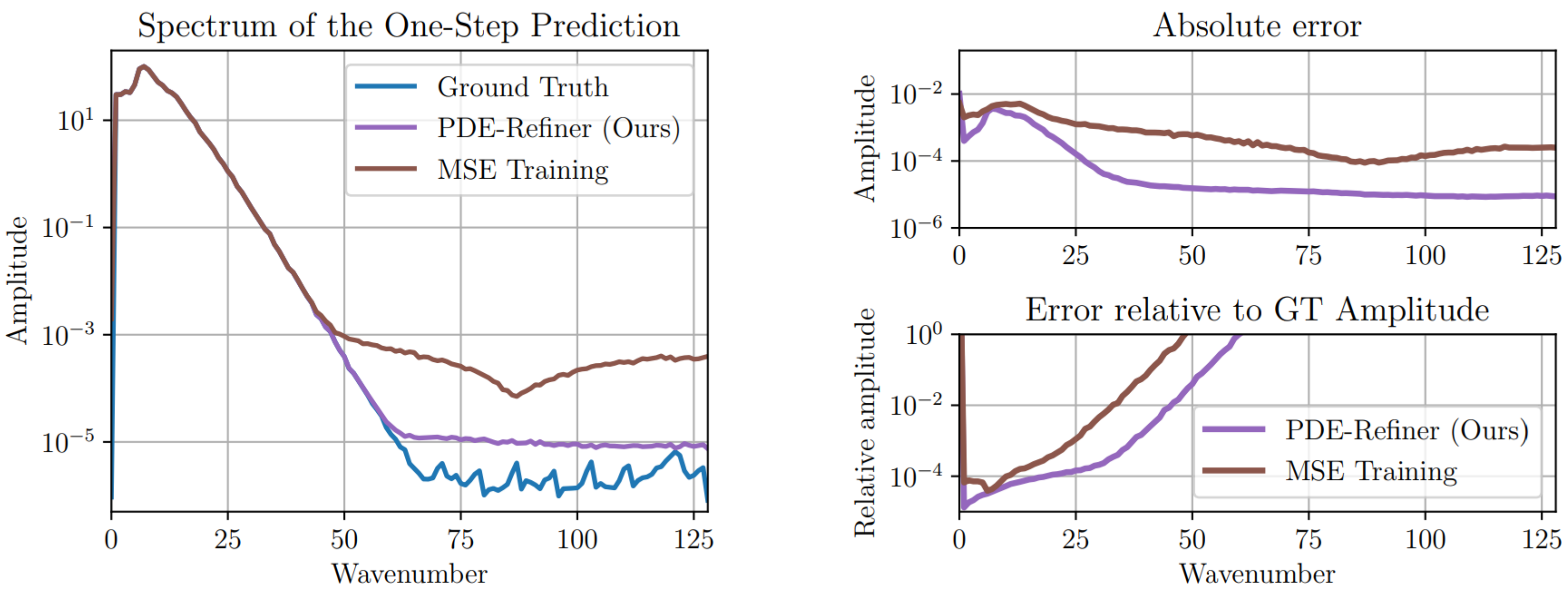


Figure credit: [Ho et al., 2020] Ho, J., Jain, A., & Abbeel, P. (2020). Denoising diffusion probabilistic models. *NeurIPS, 2020.*

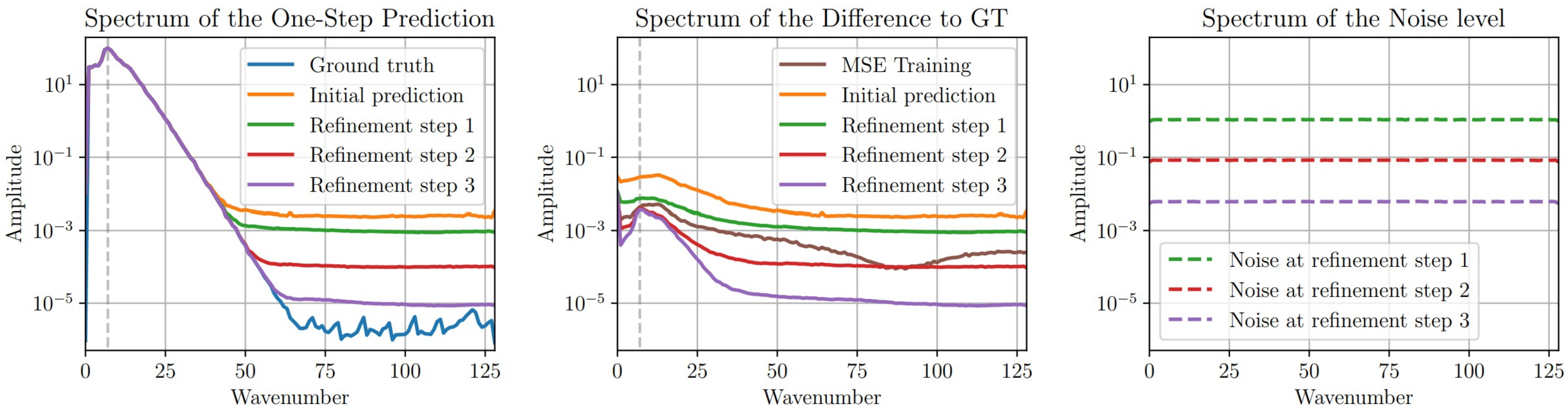
PDE-Refiner – Frequency Spectrum KS equation

- PDE-Refiner models a larger frequency band accurately

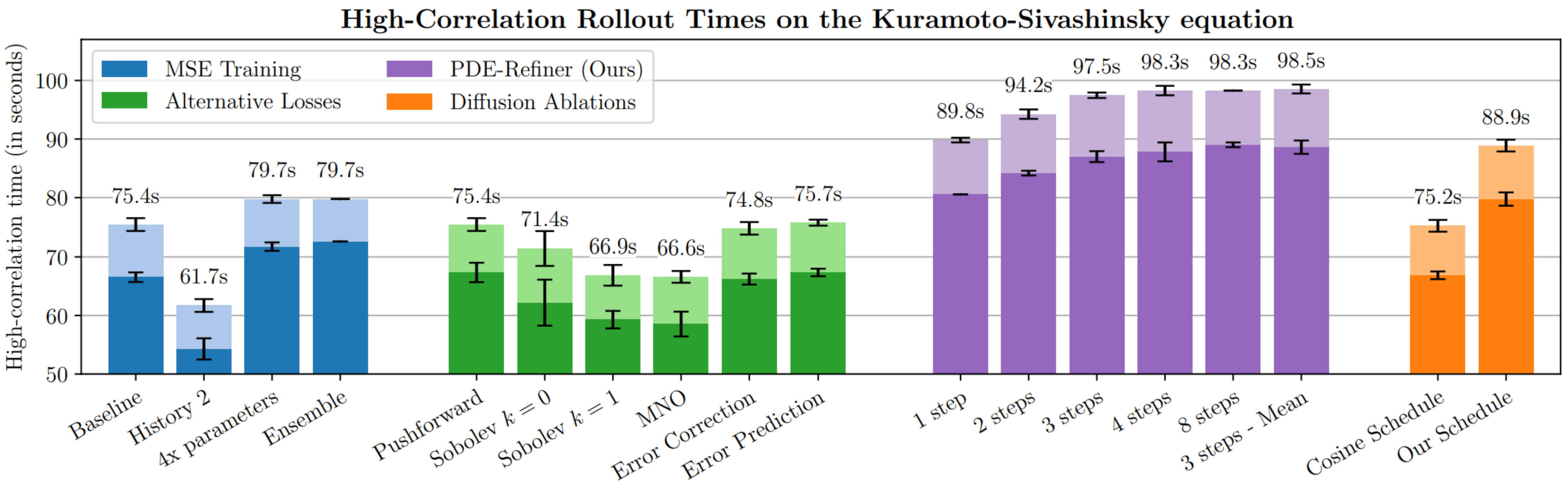


PDE-Refiner – Frequency Spectrum KS equation

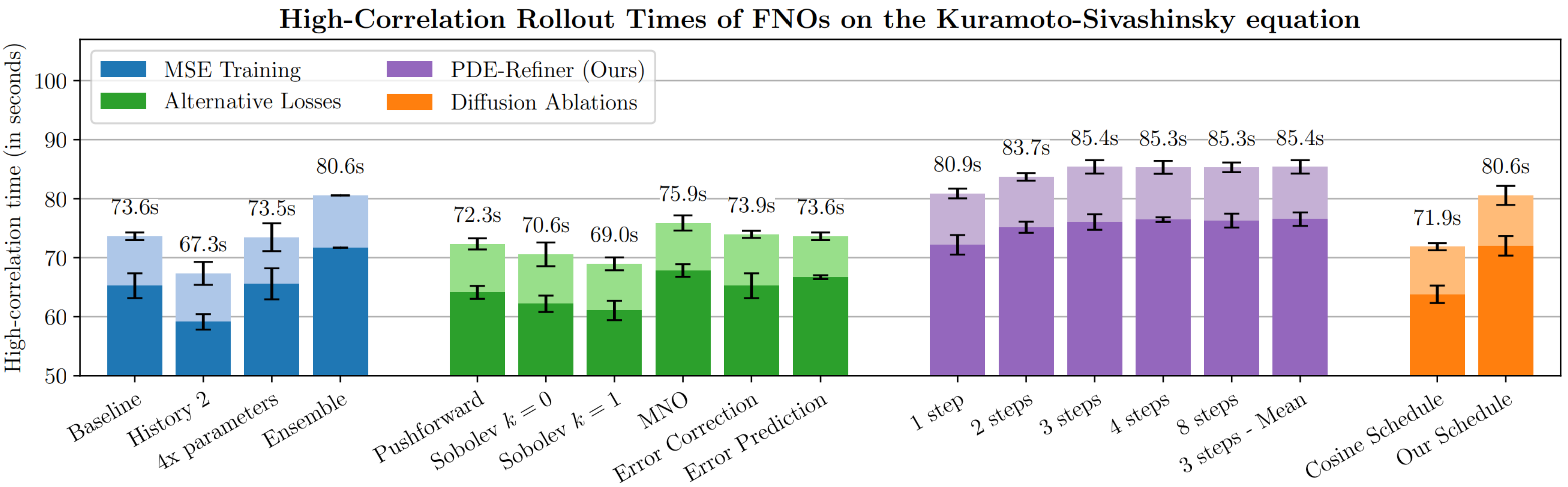
- Refinement steps focus on different amplitude levels



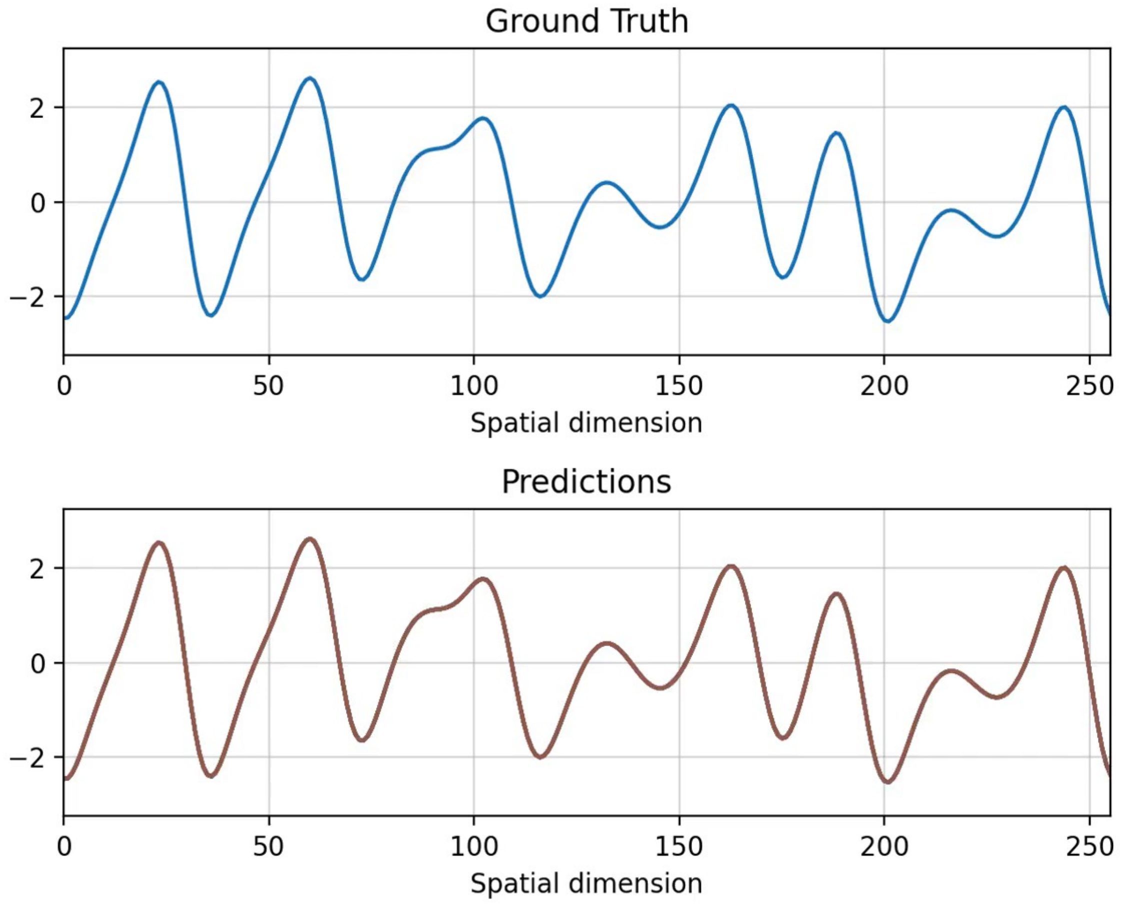
PDE-Refiner – Rollout Performance (U-Net)

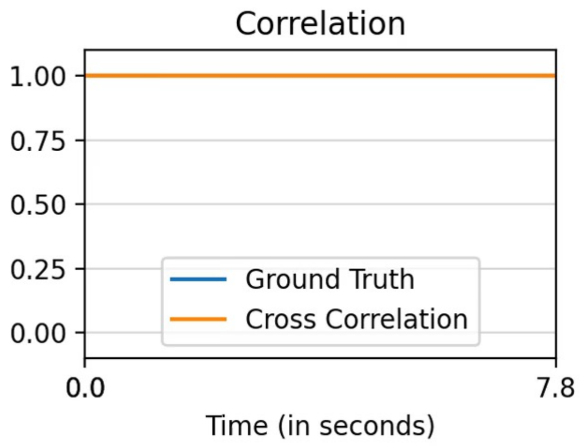


PDE-Refiner – Rollout Performance (FNO)



PDE-Refiner – Uncertainty Estimation

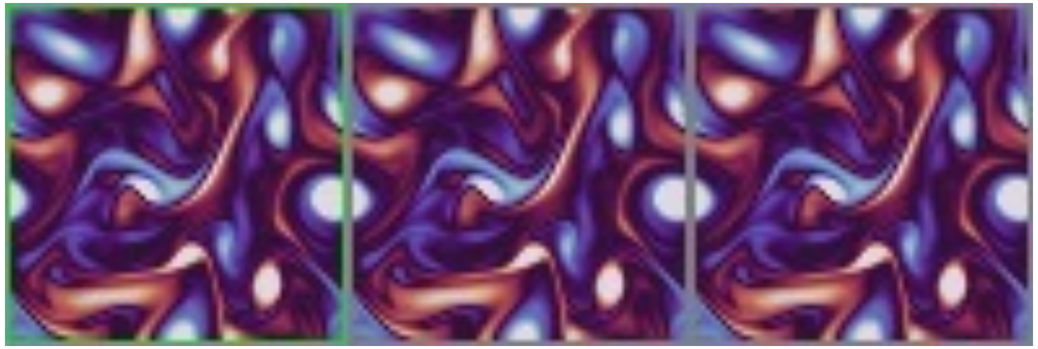




PDE-Refiner – 2D Kolmogorov Flow

- PDE-Refiner also improves on 2D equations
- Speed comparison
  - MSE model: 4 seconds
  - PDE-Refiner: (K+1) x MSE =16 seconds
  - Hybrid solver: 20 seconds
  - Classical solver: 31 minutes

Ground Truth Predictions



Method	Corr. > 0.8 time
Classical PDE Solvers	
DNS - 64×64	2.805
DNS - 128×128	3.983
DNS - 256×256	5.386
DNS - 512×512	6.788
DNS - 1024×1024	8.752
Hybrid Methods	
LC [42, 79] - CNN	6.900
LC [42, 79] - FNO	7.630
LI [42] - CNN	7.910
TSM [75] - FNO	7.798
TSM [75] - CNN	8.359
TSM [75] - HiPPO	9.481
ML Surrogates	
MSE training - FNO	6.451 ± 0.105
MSE training - U-Net	9.663 ± 0.117
PDE-Refiner - U-Net	**10.659** ± 0.092

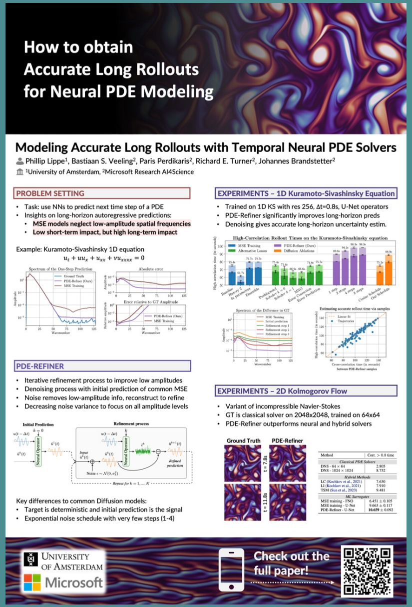
Summary

- Modeling a large spatial frequency band is key for long accurate rollouts
- PDE-Refiner achieves this by an iterative refinement process, gaining up to 30% longer rollouts
- Denoising process inherently learns accurate uncertainty estimate
- PDE-Refiner offers flexible tradeoff between accuracy and speed

Modeling Accurate Long Rollouts with Temporal Neural PDE Solvers

Phillip Lippe, Bas S. Veeling, Paris Perdikaris, Richard E. Turner, Johannes Brandstetter

Visit our poster!



Project Website