



**HAL**  
open science

# Continuous PDE Dynamics Forecasting with Implicit Neural Representations

Yuan Yin, Matthieu Kirchmeyer, Jean-Yves Franceschi, Alain Rakotomamonjy, Patrick Gallinari

► **To cite this version:**

Yuan Yin, Matthieu Kirchmeyer, Jean-Yves Franceschi, Alain Rakotomamonjy, Patrick Gallinari. Continuous PDE Dynamics Forecasting with Implicit Neural Representations. The Eleventh International Conference on Learning Representations, May 2023, Kigali, Rwanda. . hal-04081163

**HAL Id: hal-04081163**

**<https://hal.science/hal-04081163v1>**

Submitted on 25 Apr 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Continuous PDE Dynamics Forecasting with Implicit Neural Representations

Yuan Yin,<sup>\*1</sup> Matthieu Kirchmeyer,<sup>\*1,2</sup> Jean-Yves Franceschi,<sup>\*2</sup> Alain Rakotomamonjy,<sup>2</sup> Patrick Gallinari<sup>1,2</sup> <sup>\*Equal Contribution</sup> <sup>1Sorbonne Université, CNRS, ISIR</sup> <sup>2Criteo AI Lab</sup>



## MOTIVATION

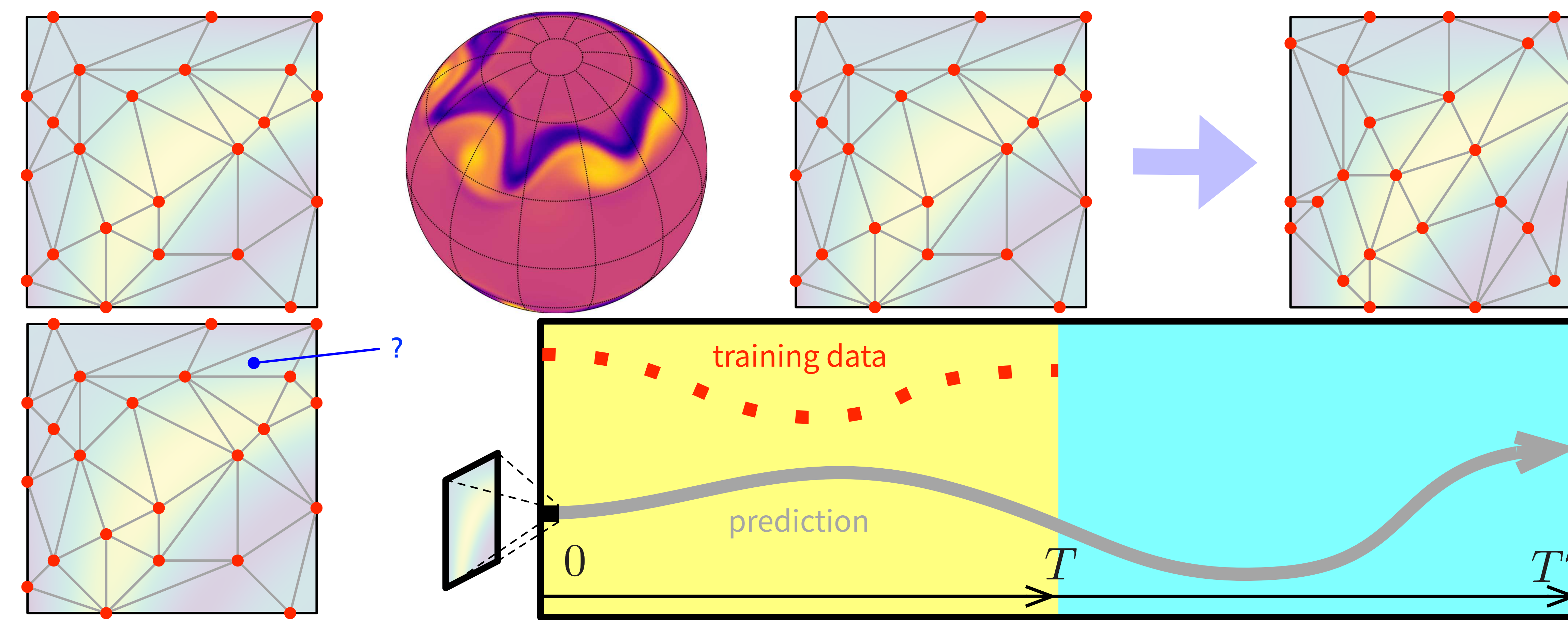
➤ Data-driven modeling of Partial Differential Equations (PDEs):

$$\frac{\partial v(x, t)}{\partial t} = F\left(x, v(x, t), \frac{\partial v(x, t)}{\partial x}, \frac{\partial^2 v(x, t)}{\partial x^2}, \dots\right)$$

➤ Variety of neural PDE forecasters... but with limitations.

➤ Spatial and temporal flexibility requirements.

- [1] Chen et al. Neural ordinary differential equations. *NeurIPS 2018*.
- [2] Brandstetter et al. Message passing neural PDE solvers. *ICLR 2022*.
- [3] Li et al. Markov neural operators for learning chaotic systems. *arXiv, 2021*.
- [4] Lu et al. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence, 2021*.
- [5] Sitzmann et al. INRs with periodic activation functions. *NeurIPS 2020*.
- [6] Fathyony et al. Multiplicative filter networks. *ICLR 2021*.



Model	Ref.	Space			Time		PDE agnostic, new initial conditions
		Train / test grid independence	Free-form grid and topology	Evaluation at un-observed locations	Time continuous	Time extrapolation	
Dis-crete	{ NODE [1] MP-PDE [2]	{ X X	{ X ✓	{ X X	{ ✓ X	{ ✓ ✓	{ ✓ ✓
NO	{ MNO [3] DeepONet [4]	{ ✓ X	{ X ✓	{ X ✓	{ X ✓	{ ✓ X	{ ✓ ✓
INR	{ SIREN, MFN [5, 6] Ours	{ ✓ ✓	{ ✓ ✓	{ ✓ ✓	{ ✓ ✓	{ X ✓	{ X ✓

➤ DINO solves prior limitations by combining INRs and ODEs.

## DINO: SPACE-TIME-CONTINUOUS FORECASTING

### INFERENCE

➤ INR-based decoder with hypernet:

$$(DEC) \forall t, \tilde{v}_t = I_{h_\phi}(\alpha_t)$$

➤ Encoding via auto-decoding, without explicit encoder NN:

$$(ENC) \alpha_0 = \text{auto-dec}(v_0)$$

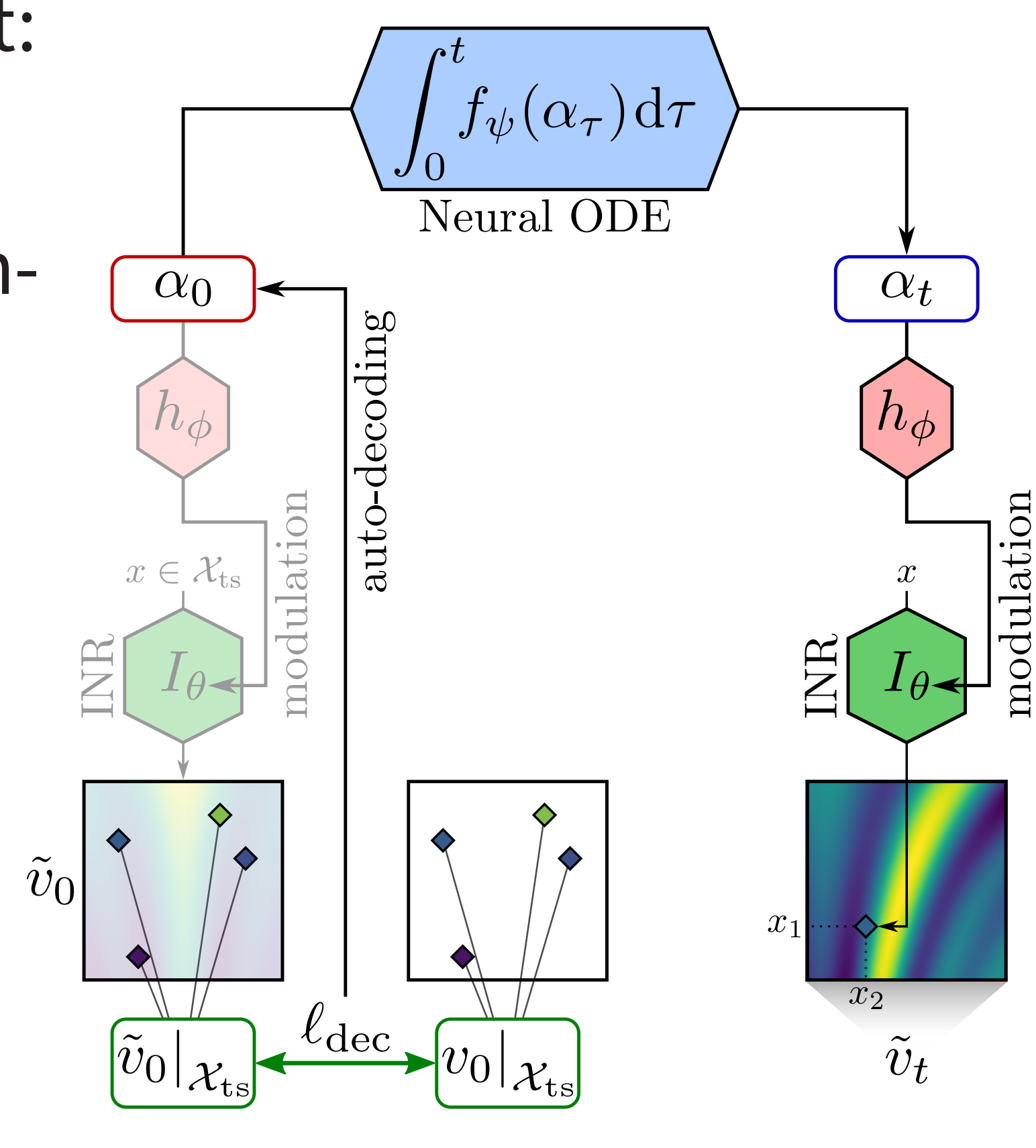
Inverse problem solved by gradient descent:

$$\alpha_0^k = \alpha_0^{k-1} - \eta \nabla_{\alpha_t} \ell_{\text{dec}}(\phi, \alpha_t^{k-1}; v_t),$$

$$\alpha_0 = \alpha_0^K$$

➤ Neural ODE as dynamics model:

$$(DYN) \frac{d\alpha_t}{dt} = f_\psi(\alpha_t)$$



### TWO-STAGE TRAINING PROCESS

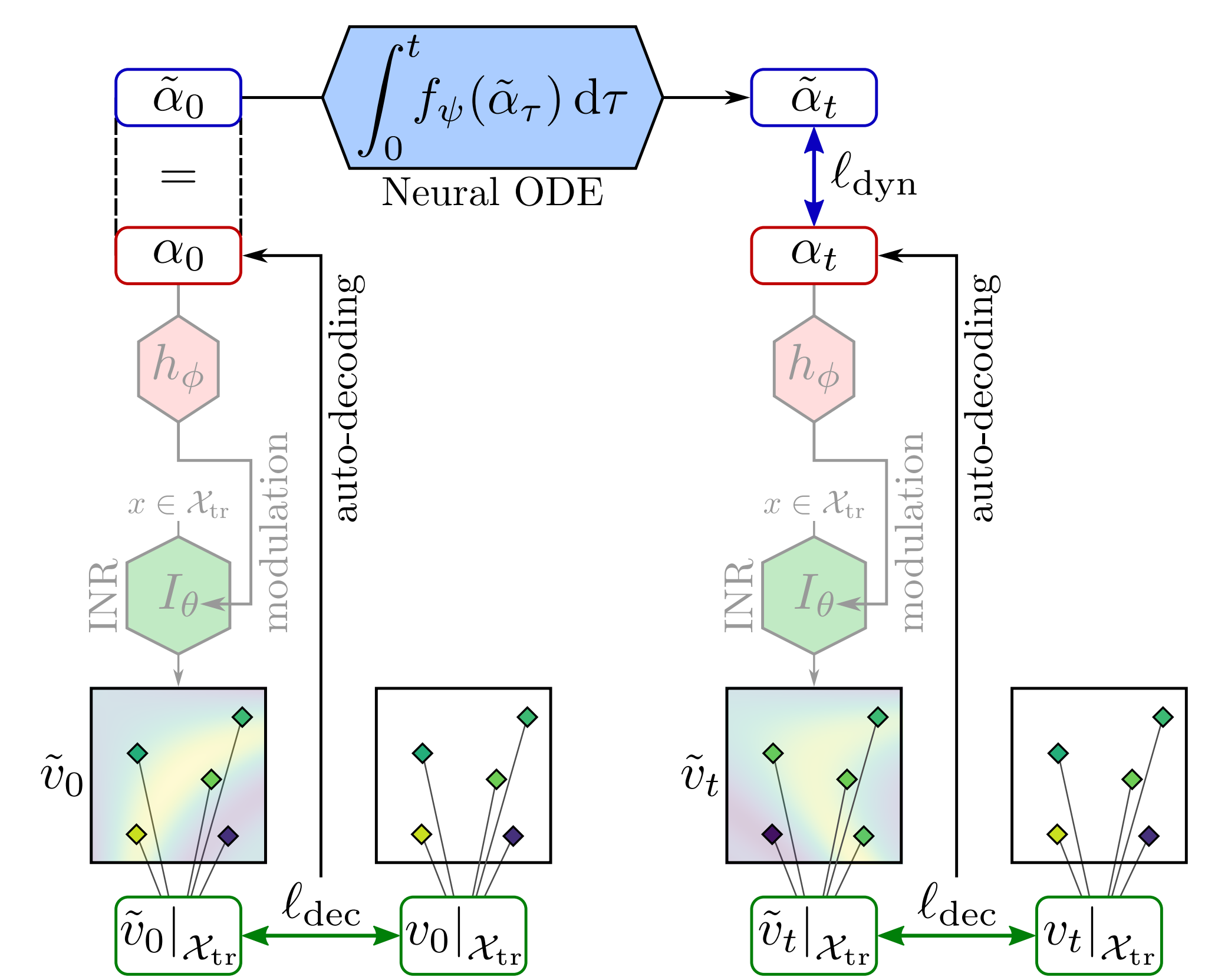
Separate training of model components:

1. Train (ENC)-(DEC):

- hypernetwork  $h_\phi$ ;
- latent state  $\{\alpha_t\}_t$ .

2. Train (DYN) on the learned  $\{\alpha_t\}_t$ :

- dynamics  $f_\psi$ .



➤ Stable training of model components.

## FOURIERNET AND AMPLITUDE MODULATION

➤ We modulated a Multiplicative Filter Network (Fathyony et al., 2021) with a linear hypernet from the latent state  $\alpha_t$ .

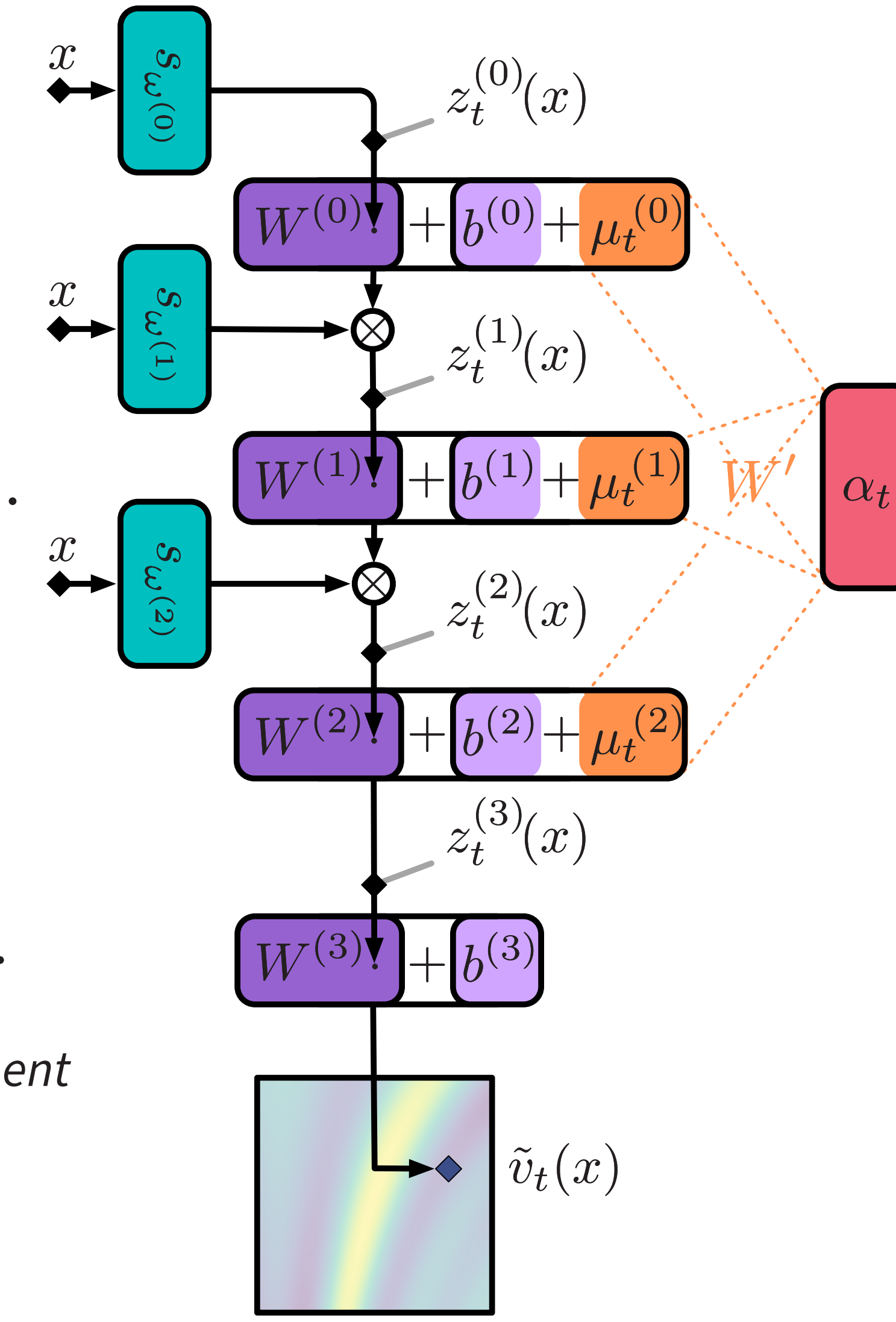
➤ We use a FourierNet:

$$s_{\omega(t)}(x) = [\cos(\omega(t)x), \sin(\omega(t)x)].$$

**Proposition 1** (Separation of time and space variables).

$$\tilde{v}_t(x) = \sum_m \underbrace{c^{(m)}(\alpha_t)}_{\text{time-dependent coefficient}} \times \underbrace{s_{\gamma^{(m)}}(x)}_{\text{spatial basis element}}$$

→ Inductive bias for better time extrapolation.



## MORE INFORMATION



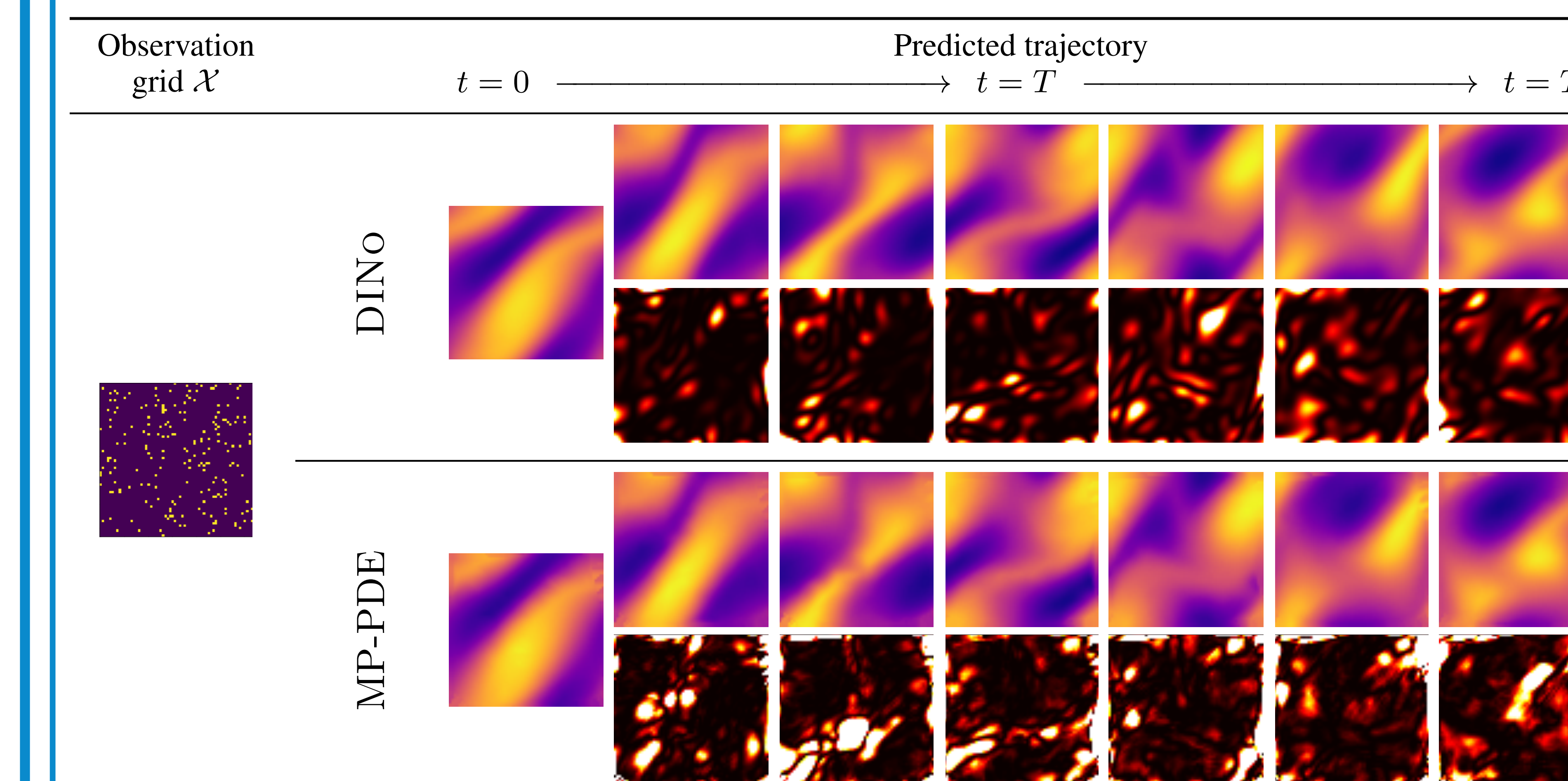
{yuan.yin, matthieu.kirchmeyer, patrick.gallinari}@sorbonne-universite.fr  
{jycja.franceschi, a.rakotomamonjy}@criteo.com



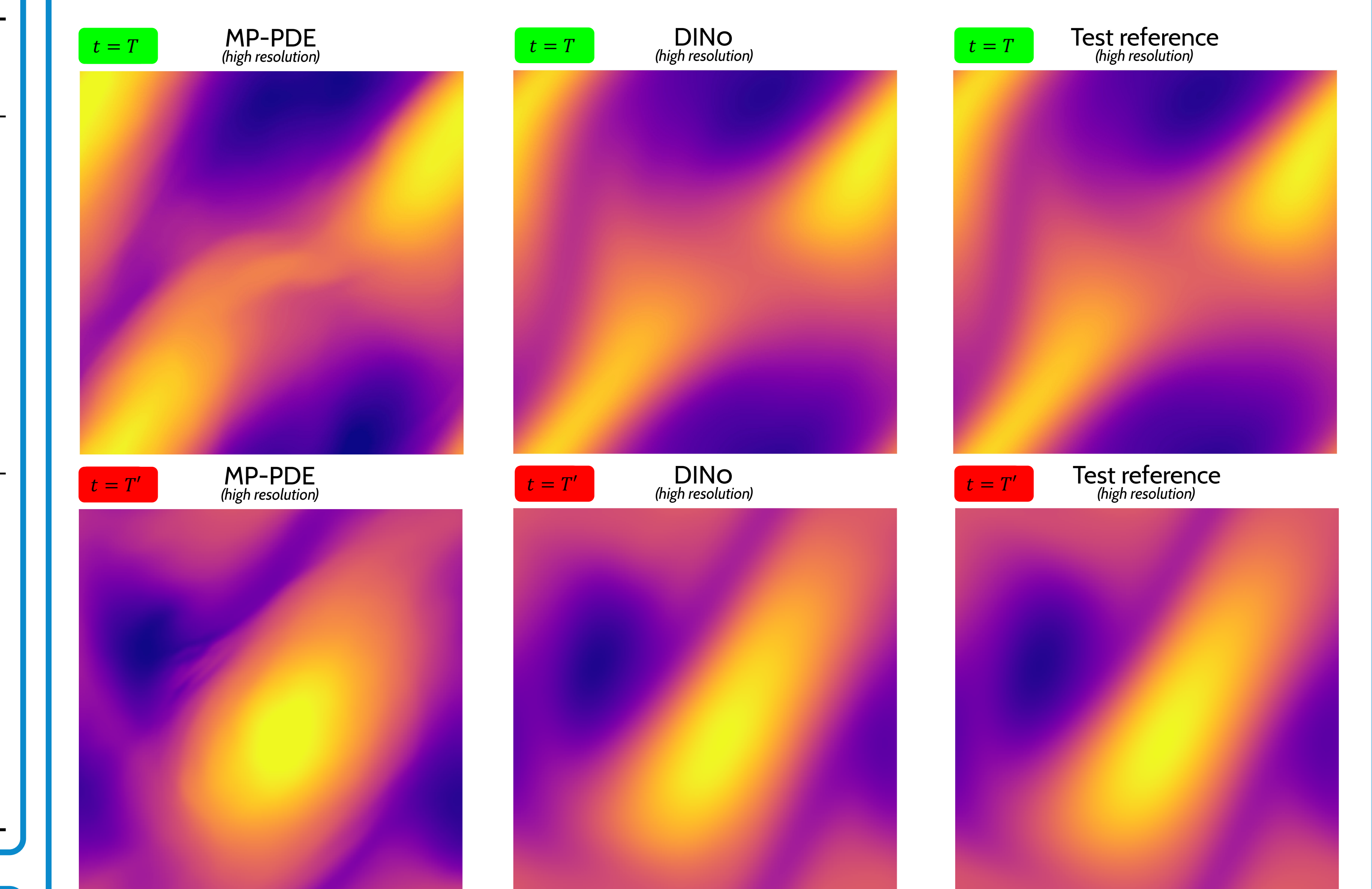
## DIVERSE EVALUATION SCENARIOS

➤ Comparison to recent neural PDE forecasters [1, 2, 3, 4, 5, 6] on planar (*Wave*, *Navier-Stokes*) and spherical (*Shallow-Water*) PDEs.

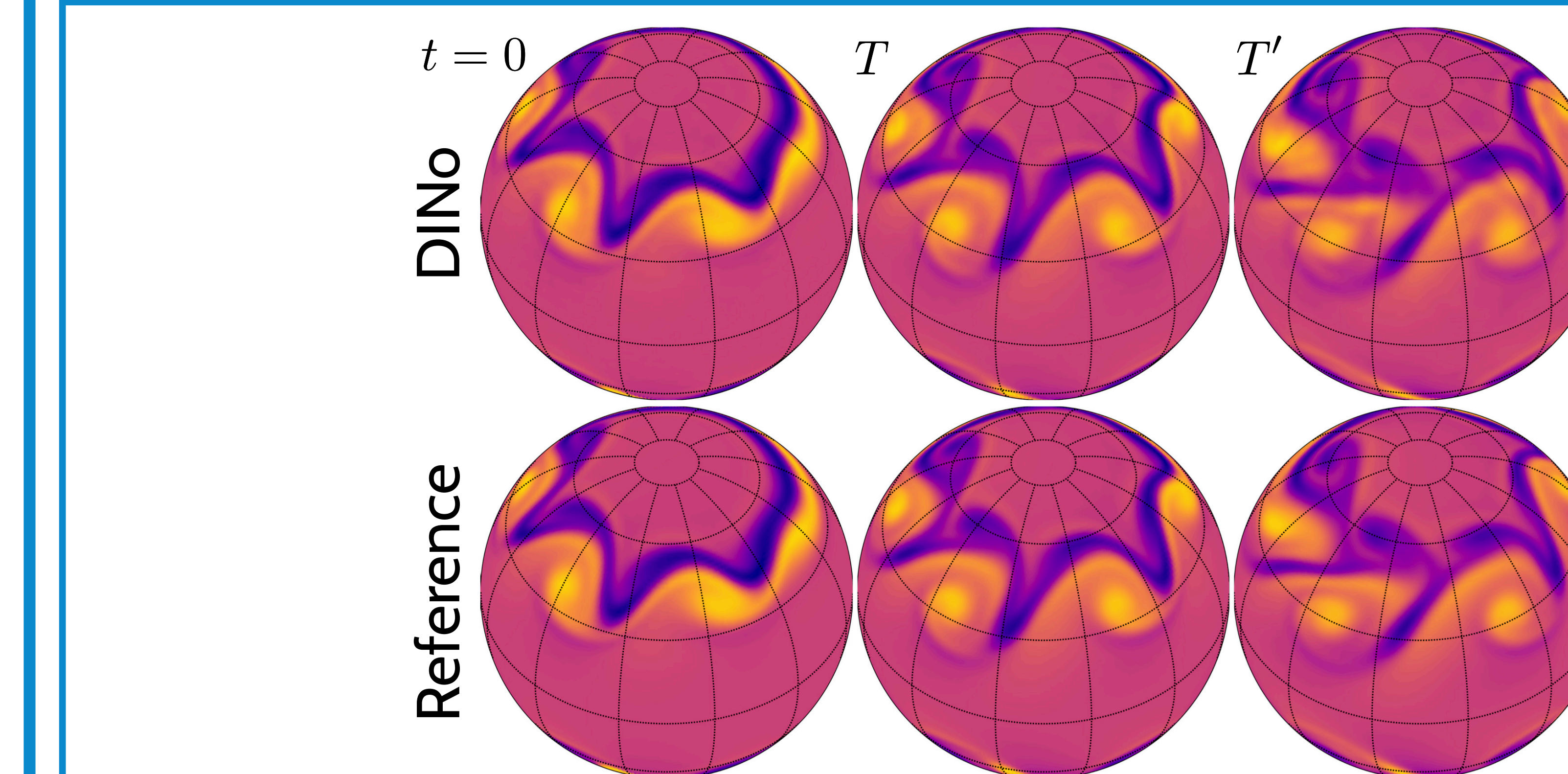
### COMPARISON TO STANDARD INTERPOLATION



### GENERALIZATION TO HIGHER RESOLUTION



### MODELING PDEs ON MANIFOLDS



### FINER TIME RESOLUTION

Model	In-t	Out-t
I-DINO (linear)	$3.459 \times 10^{-4}$	$5.598 \times 10^{-4}$
I-DINO (quadratic)	$2.165 \times 10^{-4}$	$4.473 \times 10^{-4}$
DINO (ODE solve)	<b><math>2.151 \times 10^{-4}</math></b>	<b><math>4.388 \times 10^{-4}</math></b>

