Continuous PDE Dynamics Forecasting with Implicit Neural Representations

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

Continuous PDE Dynamics Forecasting with Implicit Neural Representations

Yuan Yin,*[1] Matthieu Kirchmeyer,*[1,2] Jean-Yves Franceschi,*[2] Alain Rakotomamonjy,[2] Patrick Gallinari[1,2]

*Equal Contribution [1]Sorbonne Université, CNRS, ISIR [2]Criteo AI Lab

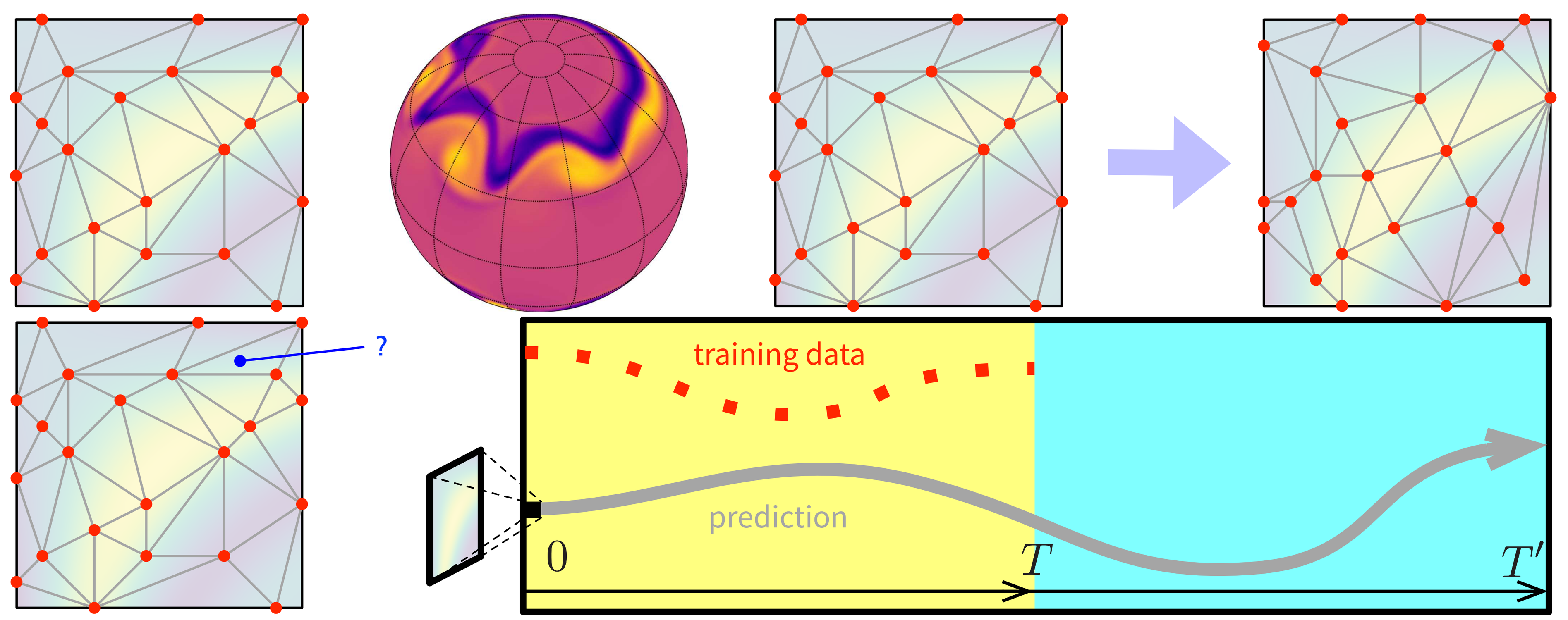
Motivation

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

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

- 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] Fathony 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 unobserved locations	Time continuous	Time extrapolation	
Discrete	NODE	[1]	✗	✗	✗	✓	✓	✓
	MP-PDE	[2]	✗	✓	✗	✗	✓	✓
NO	MNO	[3]	✓	✗	✗	✗	✓	✓
	DeepONet	[4]	✗	✓	✓	✓	✗	✓
INR	SIREN, MFN	[5, 6]	✓	✓	✓	✓	✗	✗
	DINo	Ours	✓	✓	✓	✓	✓	✓

➪ DINo solves prior limitations by combining INRs and ODEs.

DINo: Space-Time-Continuous Forecasting

Inference

- INR-based decoder with hypernet:

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

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

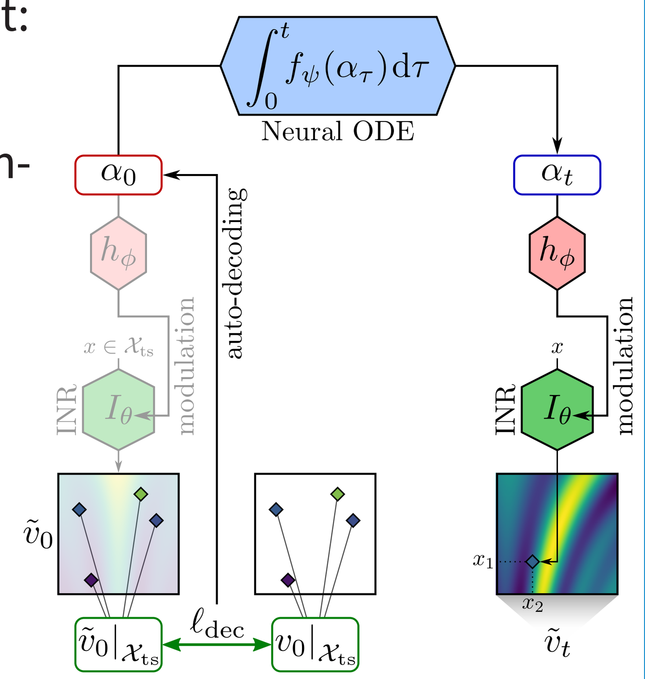
$$\text{(ENC) } \alpha_0 = \text{auto-dec}(v_0).$$

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:

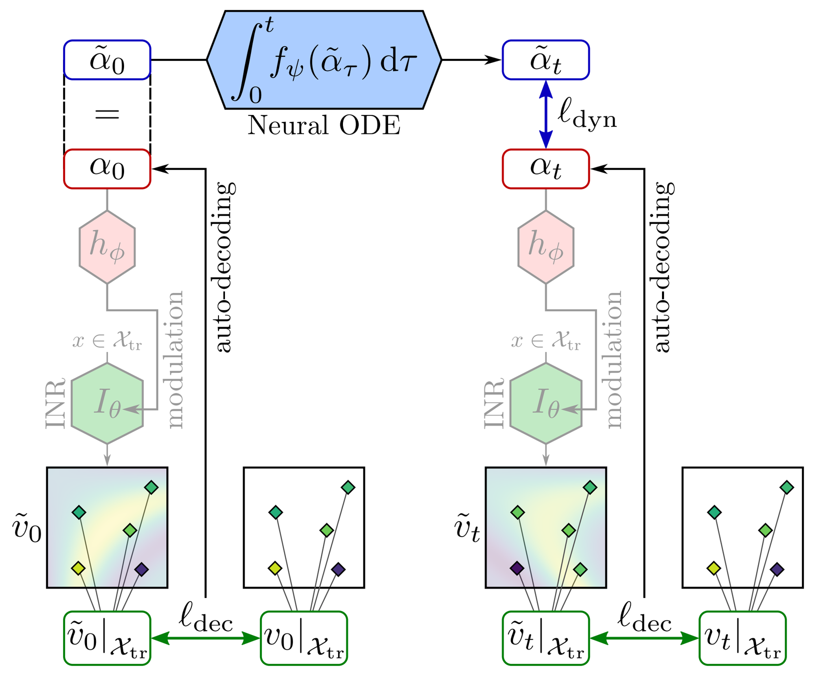
$$\text{(DYN) } \frac{\mathrm{d}\alpha_t}{\mathrm{d}t} = f_\psi(\alpha_t).$$



Two-stage Training Process

Separate training of model components:

1. Train (ENC)-(DEC):
   - hypernetwork h_ϕ;
   - latent state $\{\alpha_t\}_t$.
2. Train (DYN) on the learned $\{\alpha_t\}_t$:
   - dynamics f_ψ.



- Stable training of model components.

FourierNet and Amplitude Modulation

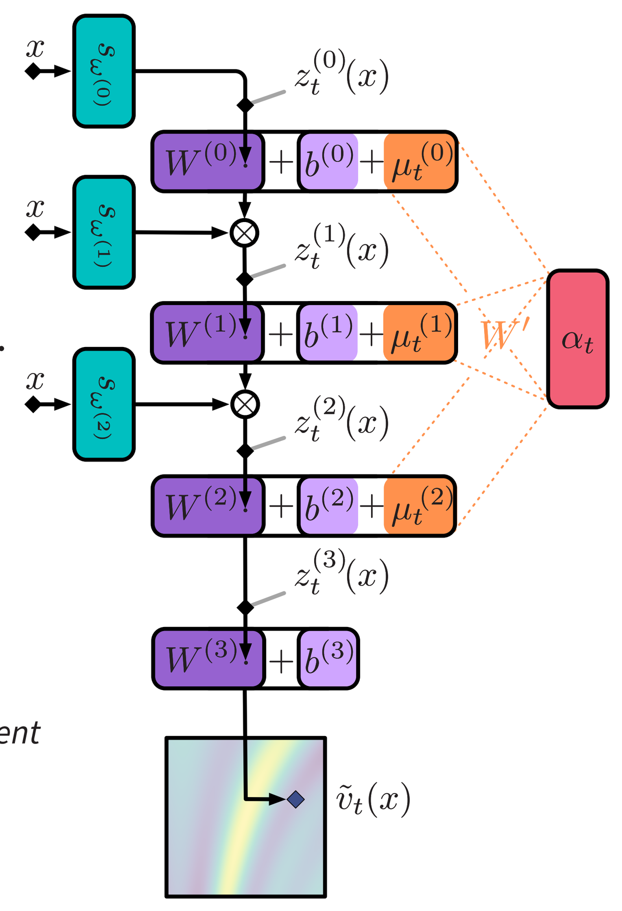
- We modulated a Multiplicative Filter Network (Fathony et al., 2021) with a linear hypernet from the latent state α_t.
- We use a FourierNet:

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

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

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

→ Inductive bias for better time extrapolation.



More Information

Paper

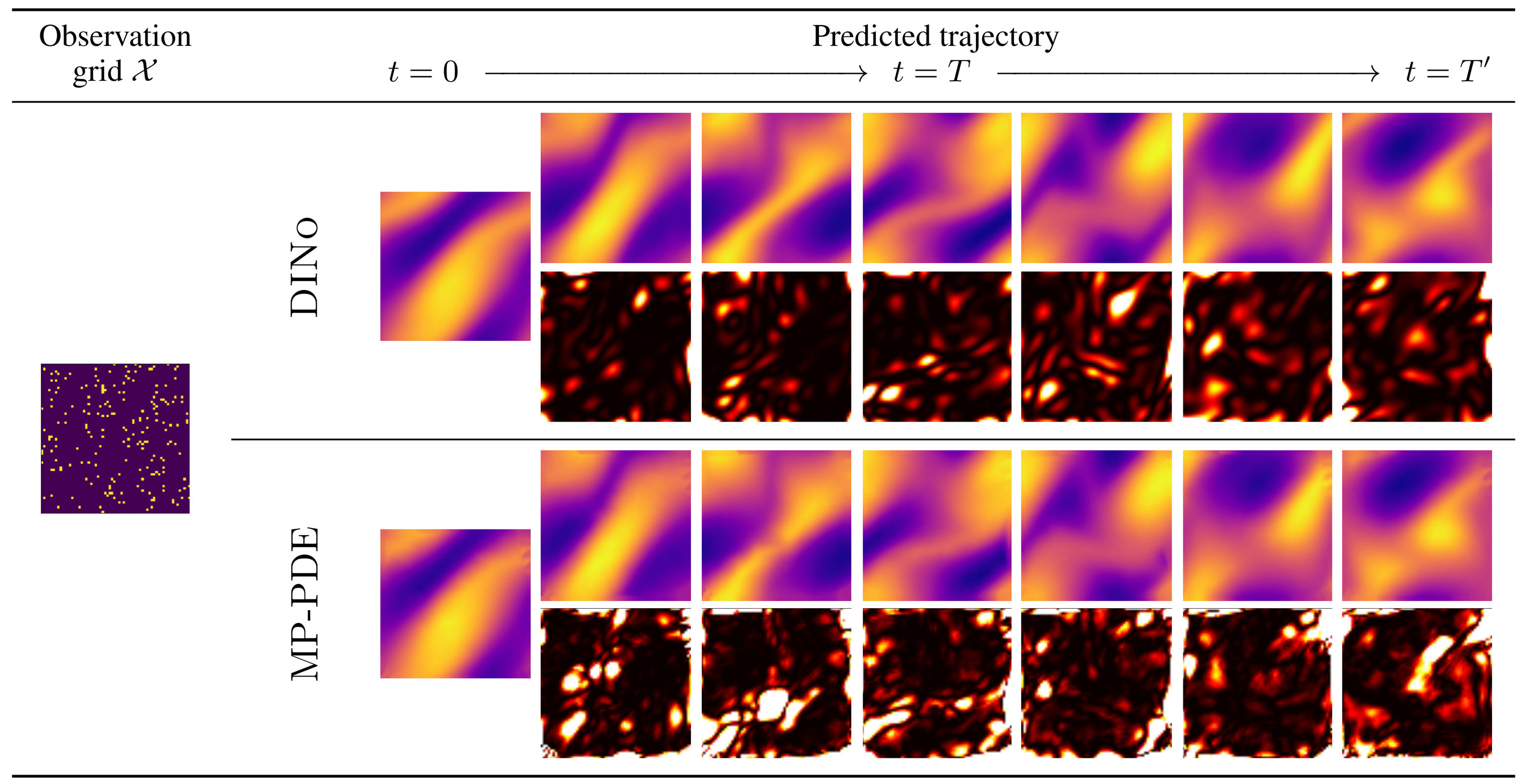
Code

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

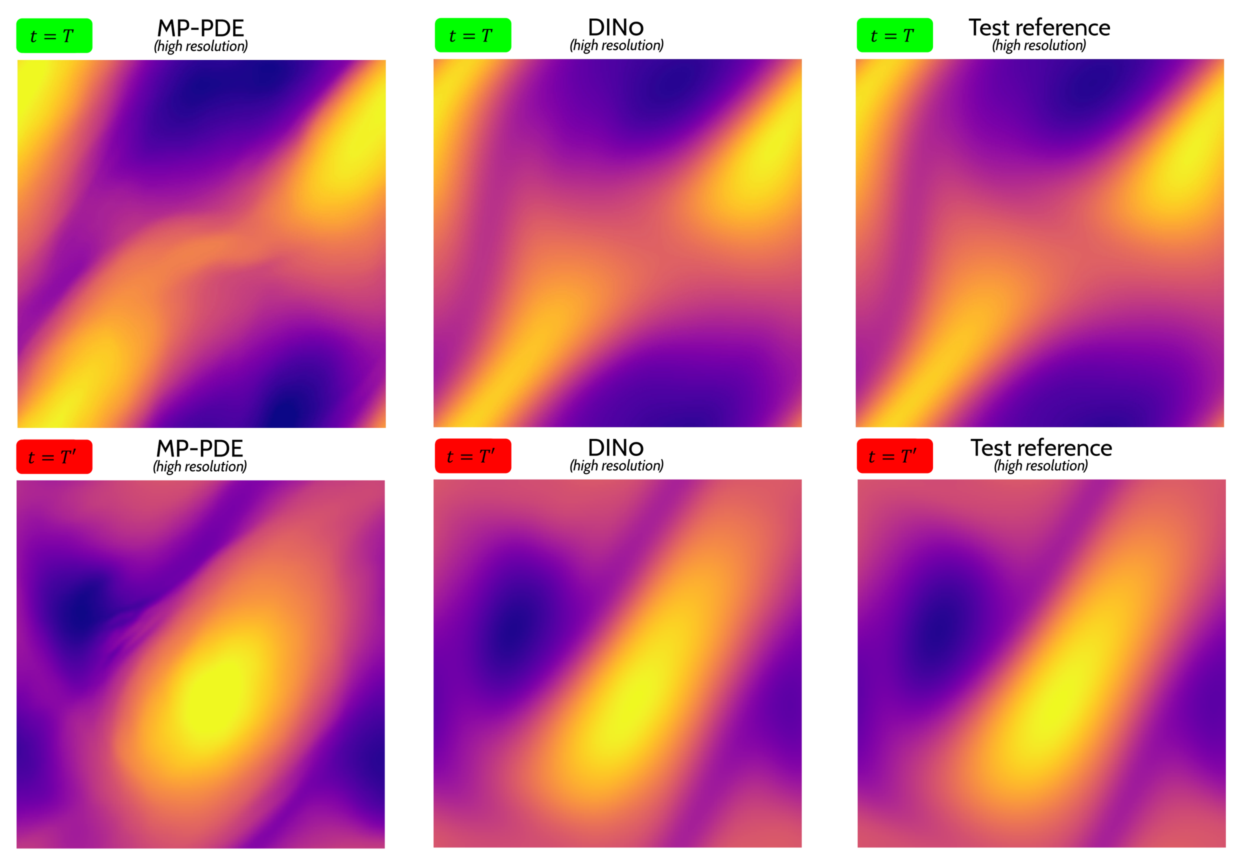
Diverse Evaluation Scenarios

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

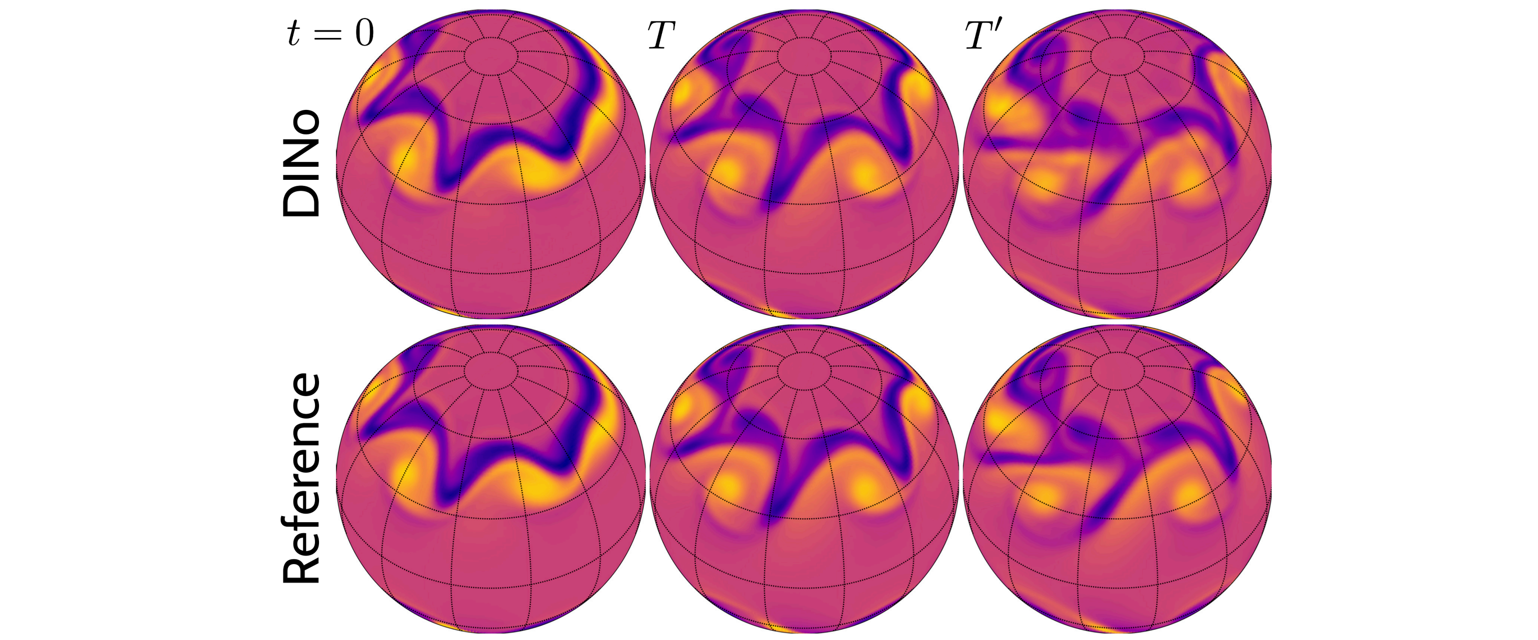
Comparison to Standard Interpolation



Generalization to Higher Resolution



Modeling PDEs on Manifolds



Finer Time Resolution

Model	In-t	Out-t
I-DINo (linear)	3.459 × 10^{-4}	5.598 × 10^{-4}
I-DINo (quadratic)	2.165 × 10^{-4}	4.473 × 10^{-4}
DINo (ODE solve)	**2.151 × 10^{-4}**	**4.388 × 10^{-4}**