Physics vs. Performance: Arrow of Time, Butterflies, and the Role of Training Data

Pedram Hassanzadeh
Geophysical Sciences
Computational and Applied Math
Data Science Institute

University of Chicago





AI for Climate Initiative

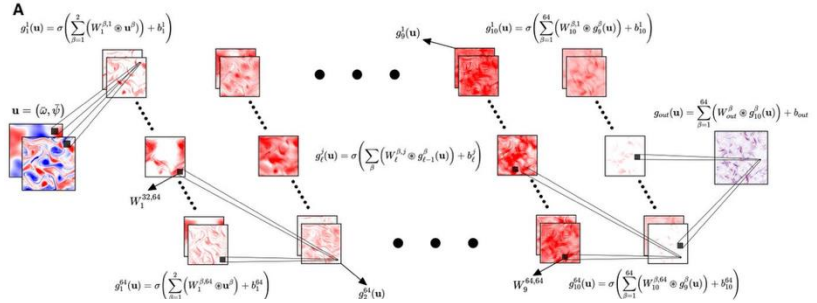
Learning in AI Weather/Climate Models: How, Why, and What?

How do AI models learn?

From Libby's talk:

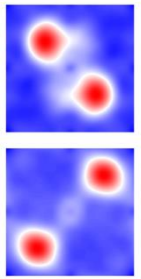
- *Predictions:* what the model **can do**
- *Post-hoc XAI:* **where they look**
- Training: **how they think**

What do individual neurons/kernels learn?



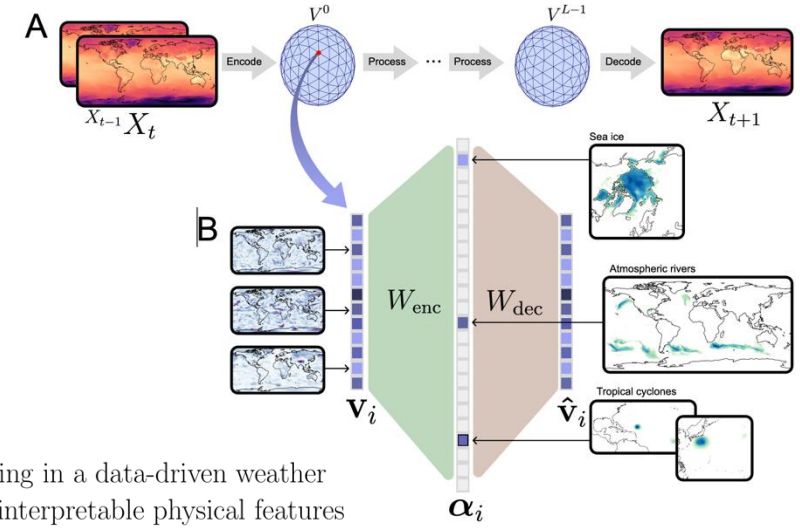
Explaining the physics of transfer learning in data-driven turbulence modeling

Adam Subel[a,1], Yifei Guan[a], Ashesh Chattopadhyay[a] and Pedram Hassanzadeh[a,b,*]



Gabor wavelets and others spectral filters

What does a collection of neurons/kernels learn?



Towards mechanistic understanding in a data-driven weather model: internal activations reveal interpretable physical features

Theodore MacMillan *1 and Nicholas T. Ouellette †1

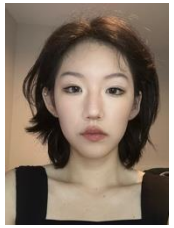
What is the system that the AI model learns?



Qiang Sun
UChicago → Nanjing U



Weidong Li
(PhD student)



Jiangdi Wang
(PhD student)



Alex Wikner
(Schmidt Fellow)



Justin Finkel
UChicago → ASU



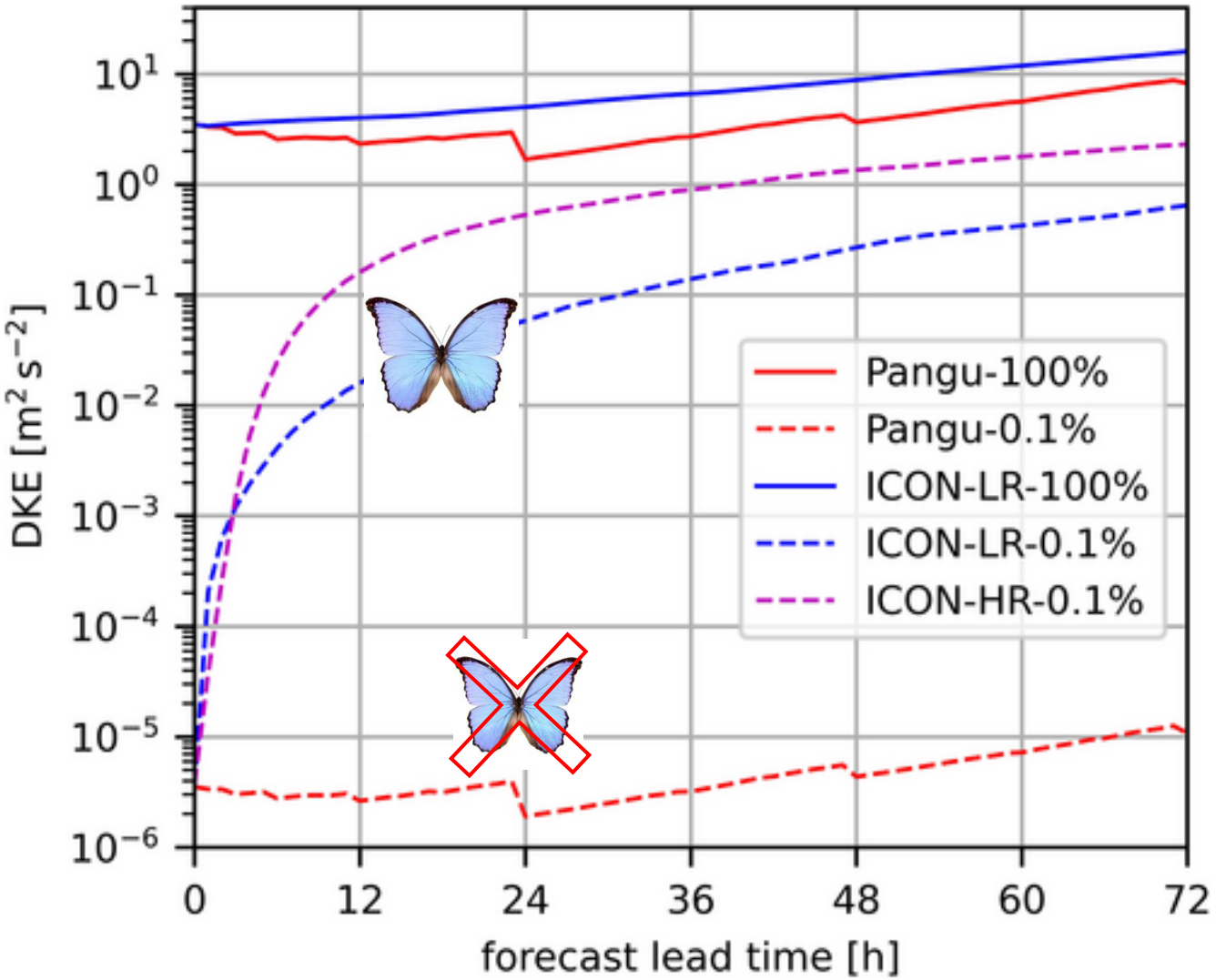
Jon Weare
NYU

Missing the Butterfly and Predicting the Past: Features or Bugs of Accurate AI Weather Models?

https://arxiv.org/abs/2608.25835

AI Weather Models do not have the "Real" Butterfly Effect!

Why? Missing chaos? Implications for practical forecasting?

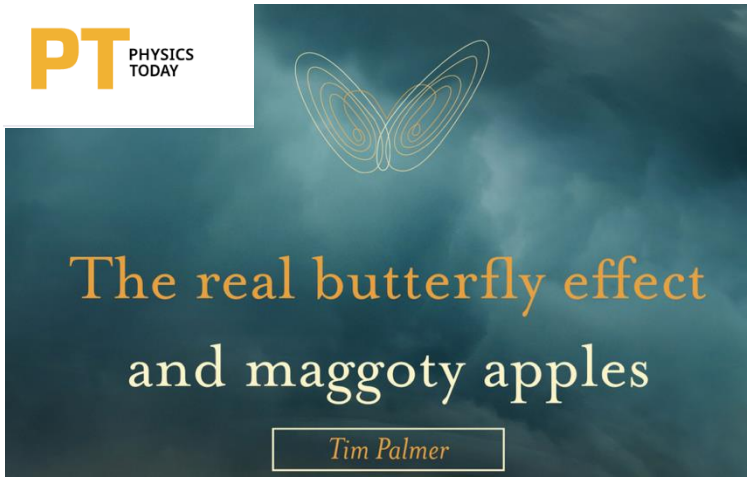


Geophysical Research Letters

Research Letter | Open Access

Can Artificial Intelligence-Based Weather Prediction Models Simulate the Butterfly Effect?

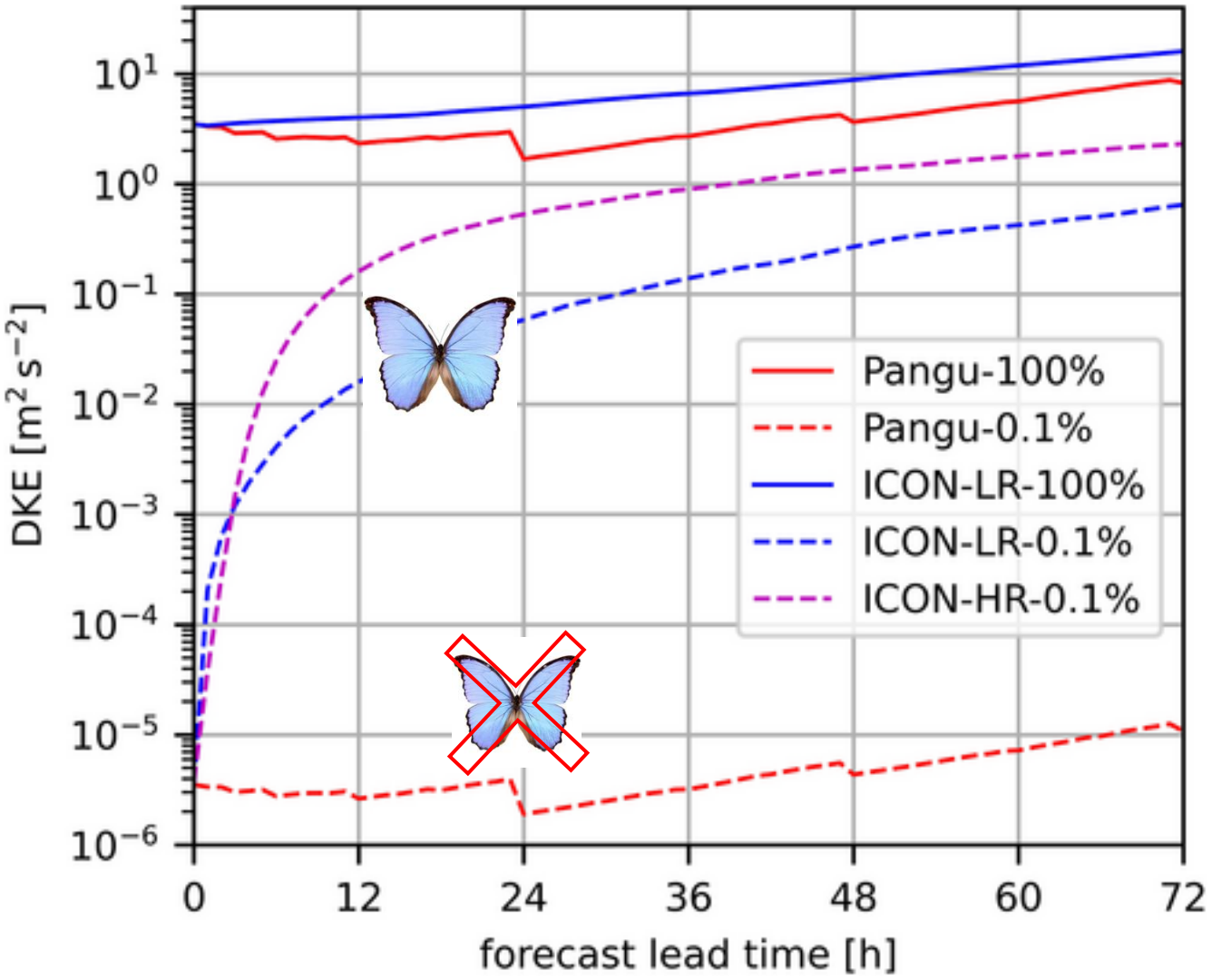
T. Selz, G. C. Craig



10.1063/pt.eike.hsbz

AI Weather Models do not have the "Real" Butterfly Effect!

Why? Missing chaos? Implications for practical forecasting?



Geophysical Research Letters

Research Letter | Open Access

Can Artificial Intelligence-Based Weather Prediction Models Simulate the Butterfly Effect?

T. Selz, G. C. Craig

Can AI-Based Weather Prediction Models Simulate the Butterfly Effect? The Role of Architecture and Implementation

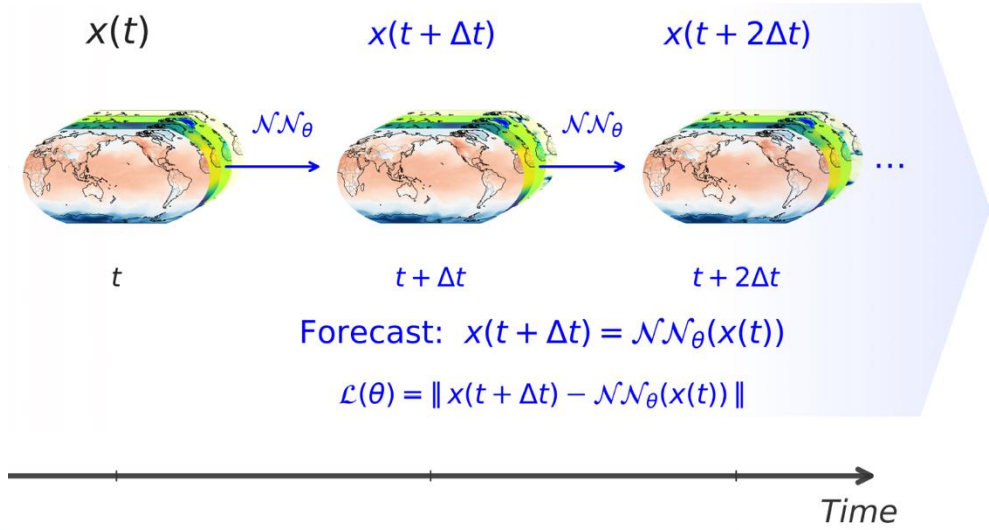
T. Selz[1] and G. C. Craig[2]

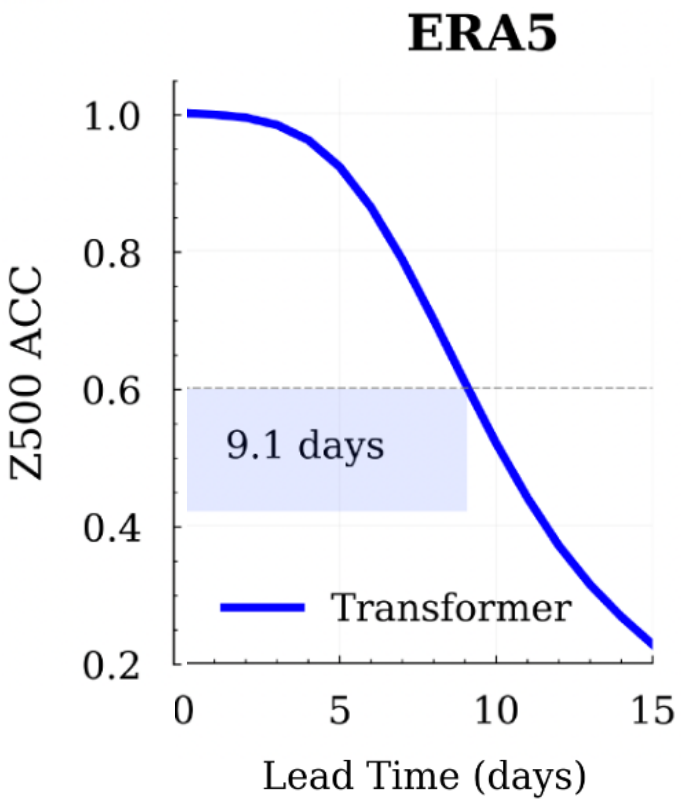
[1]Karlsruhe Institute for Technology, Karlsruhe, Germany, [2]Ludwig-Maximilians-Universität München, Munich, German

$$\boldsymbol{X}(t+\Delta t) = \mathbf{NN}(\boldsymbol{X}(t), \boldsymbol{\theta})$$

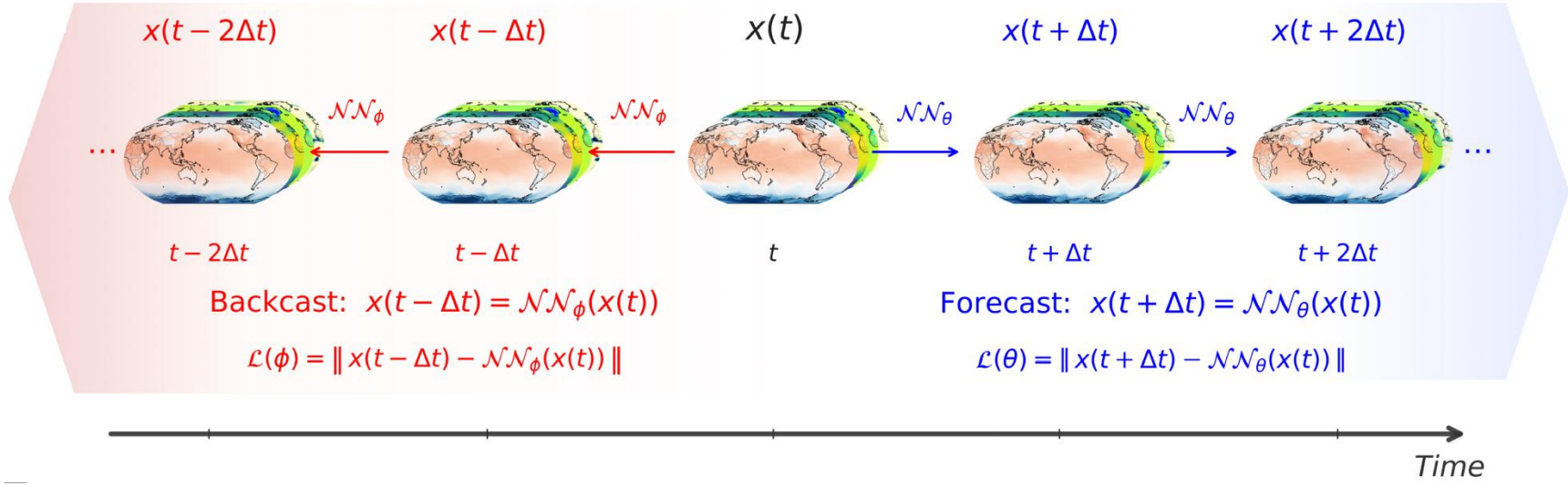
$$\mathcal{L} = \|\boldsymbol{X}(t+\Delta t) - \mathbf{NN}(\boldsymbol{X}(t), \boldsymbol{\theta})\|_2$$

Forecast


$x(t)$
$x(t+\Delta t)$
$x(t+2\Delta t)$
$\mathcal{NN}_\theta$
$\mathcal{NN}_\theta$
...
t
$t+\Delta t$
$t+2\Delta t$
Forecast: $x(t+\Delta t) = \mathcal{NN}_\theta(x(t))$
$\mathcal{L}(\theta) = \| x(t+\Delta t) - \mathcal{NN}_\theta(x(t)) \|$
Time


ERA5
Z500 ACC
1.0
0.8
0.6
0.4
0.2
9.1 days
Transformer
0
5
10
15
Lead Time (days)

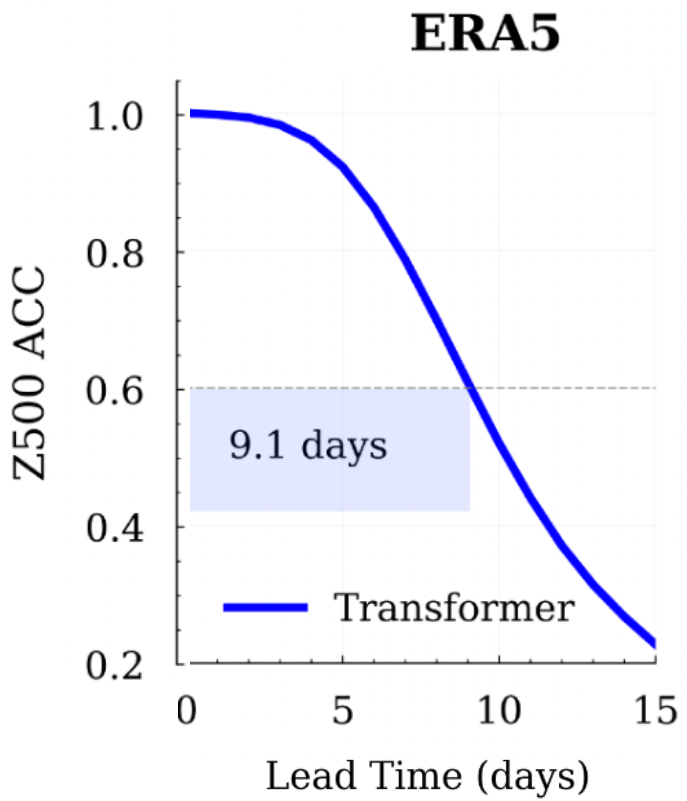
Backcast



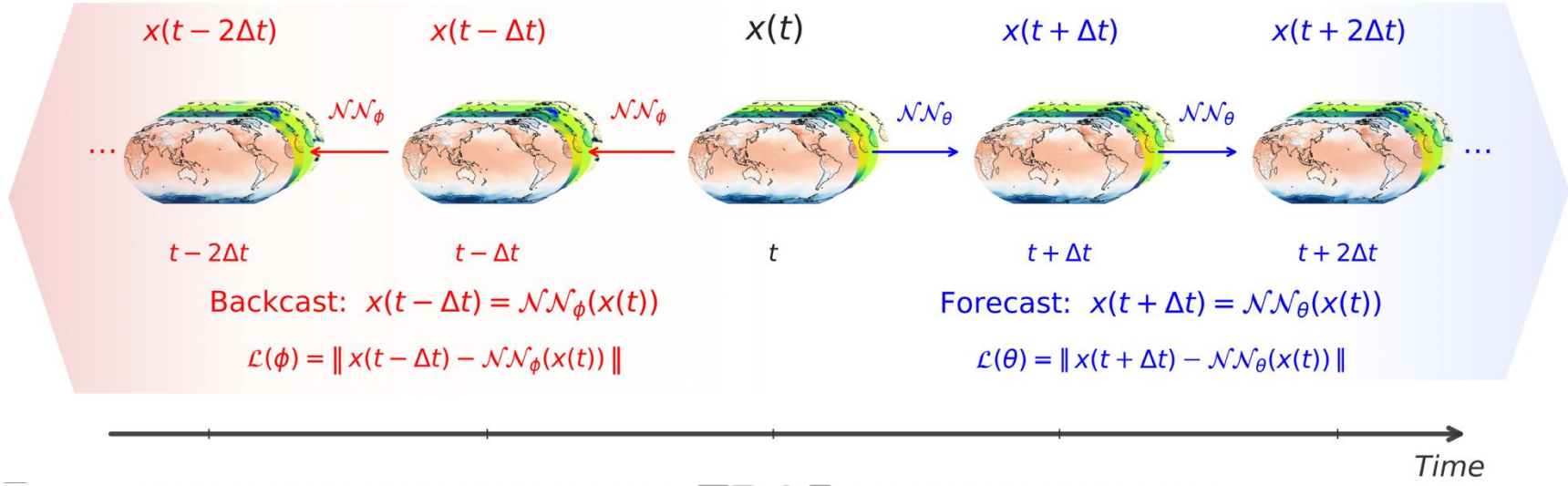
Forecast

Do AI models learn atmospheric dynamics, or merely learn the evolution of weather the way one learns a video?

A video plays backward as easily as forward.



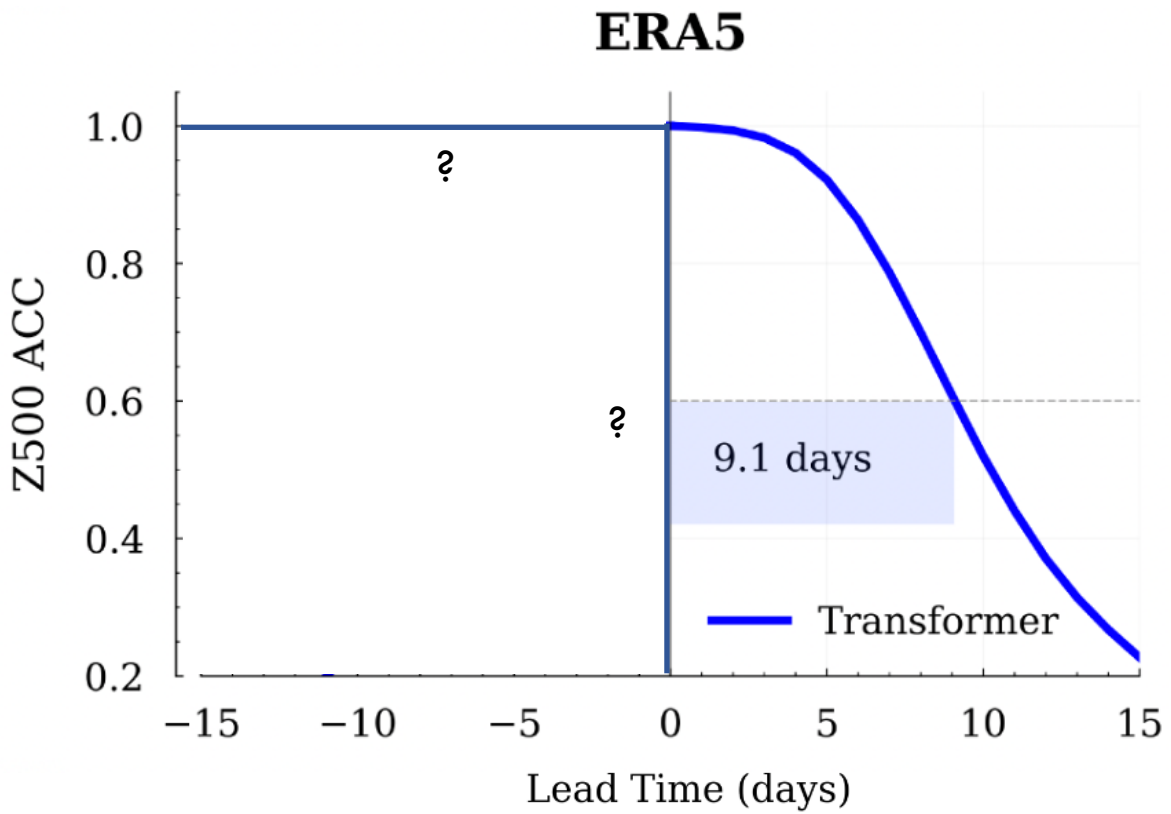
Backcast



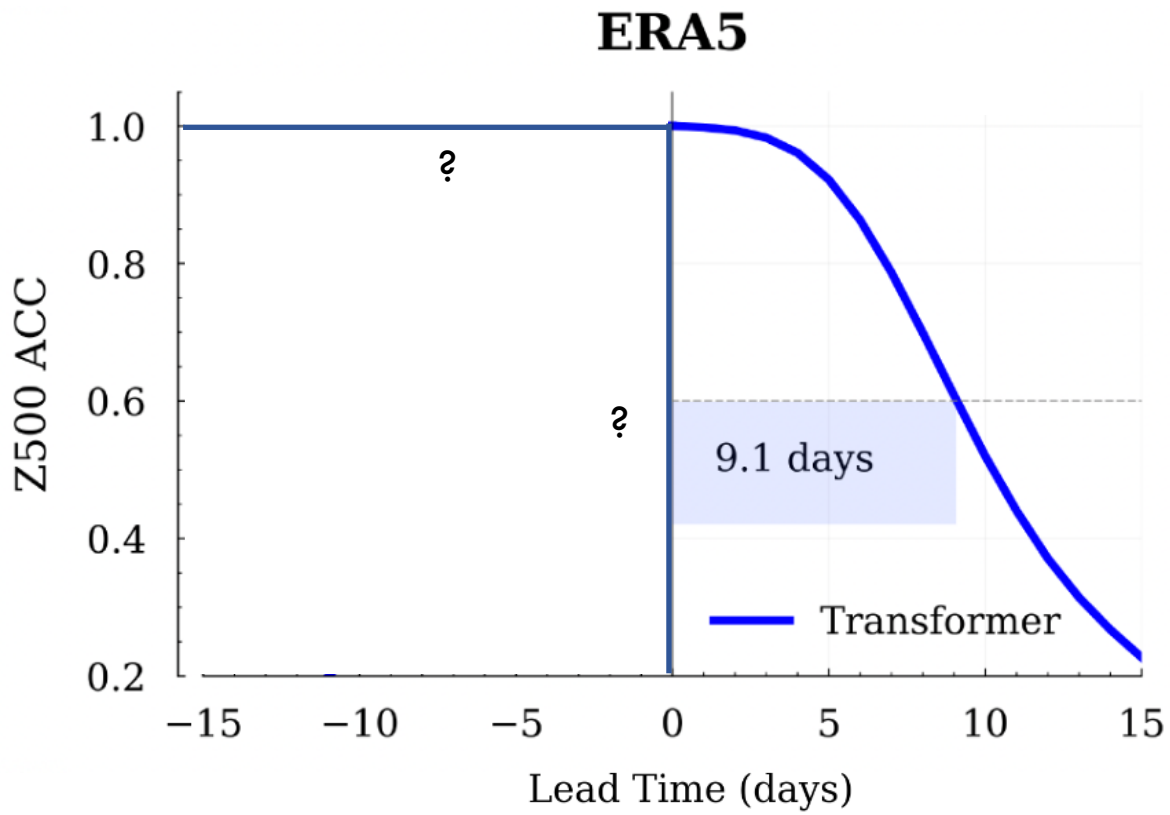
Forecast

Do AI models learn atmospheric dynamics, or merely learn the evolution of weather the way one learns a video?

A video plays backward as easily as forward.



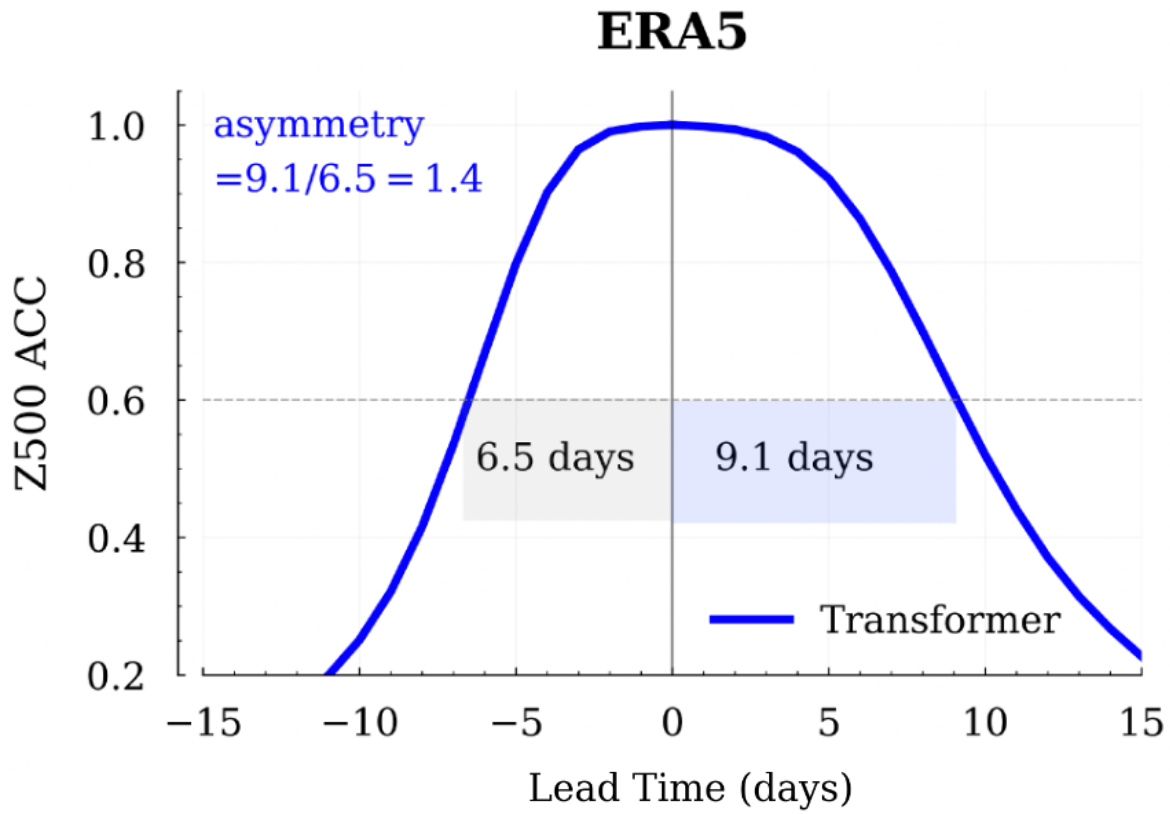
“It's tough to make predictions, especially about the future”

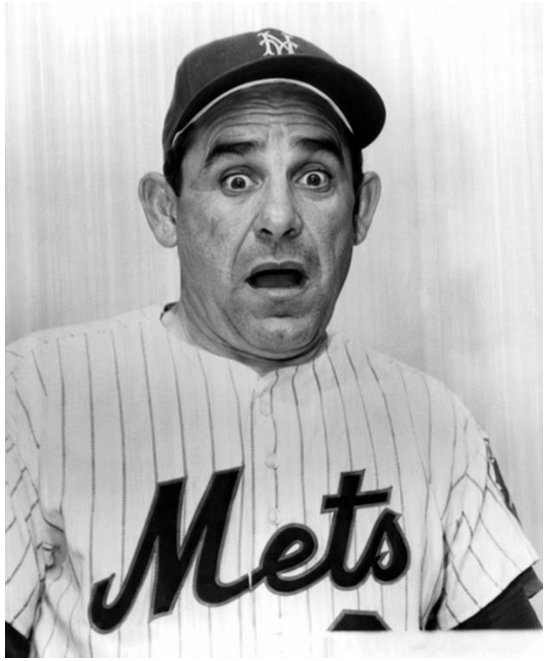




modified by ChatGPT

“It's tough to make predictions, especially about the past”



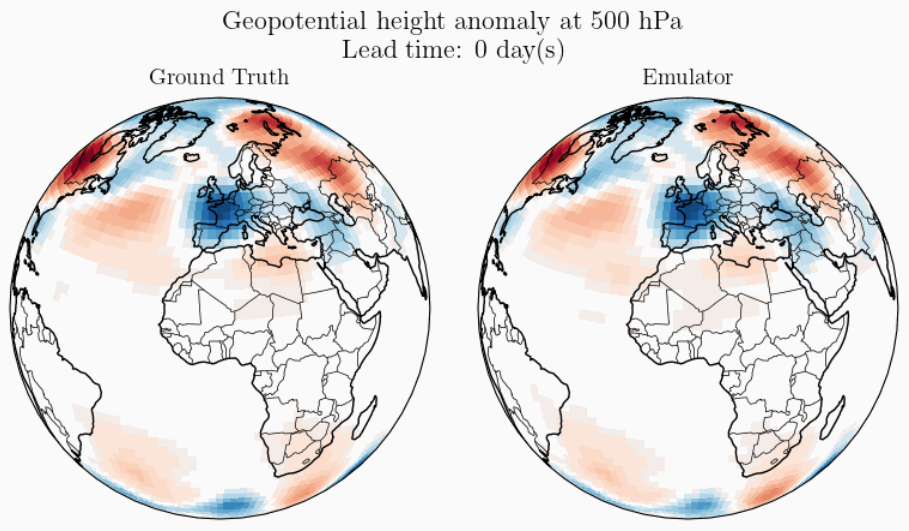


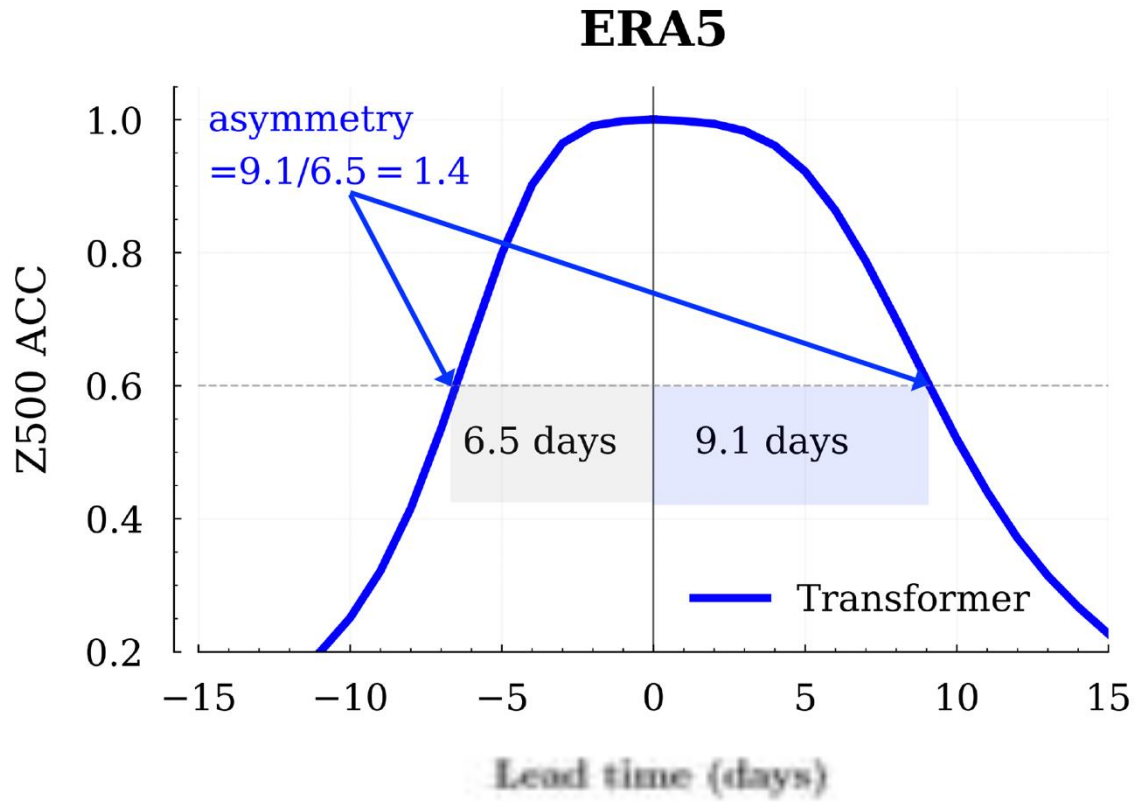
modified by ChatGPT

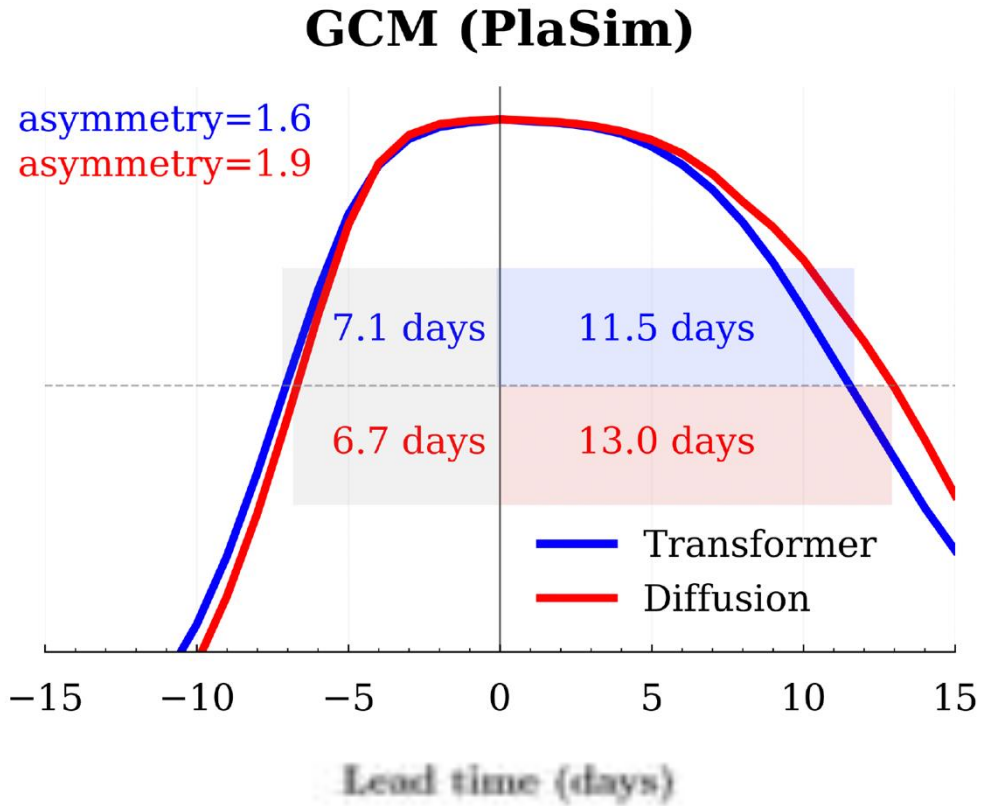
Asymmetry: Consistent Across ERA5, GCM, & Architectures (Transformer/SFNO/Diffusion)

Why is the past harder to predict?

Asymmetry>1 is largest in the tropics!

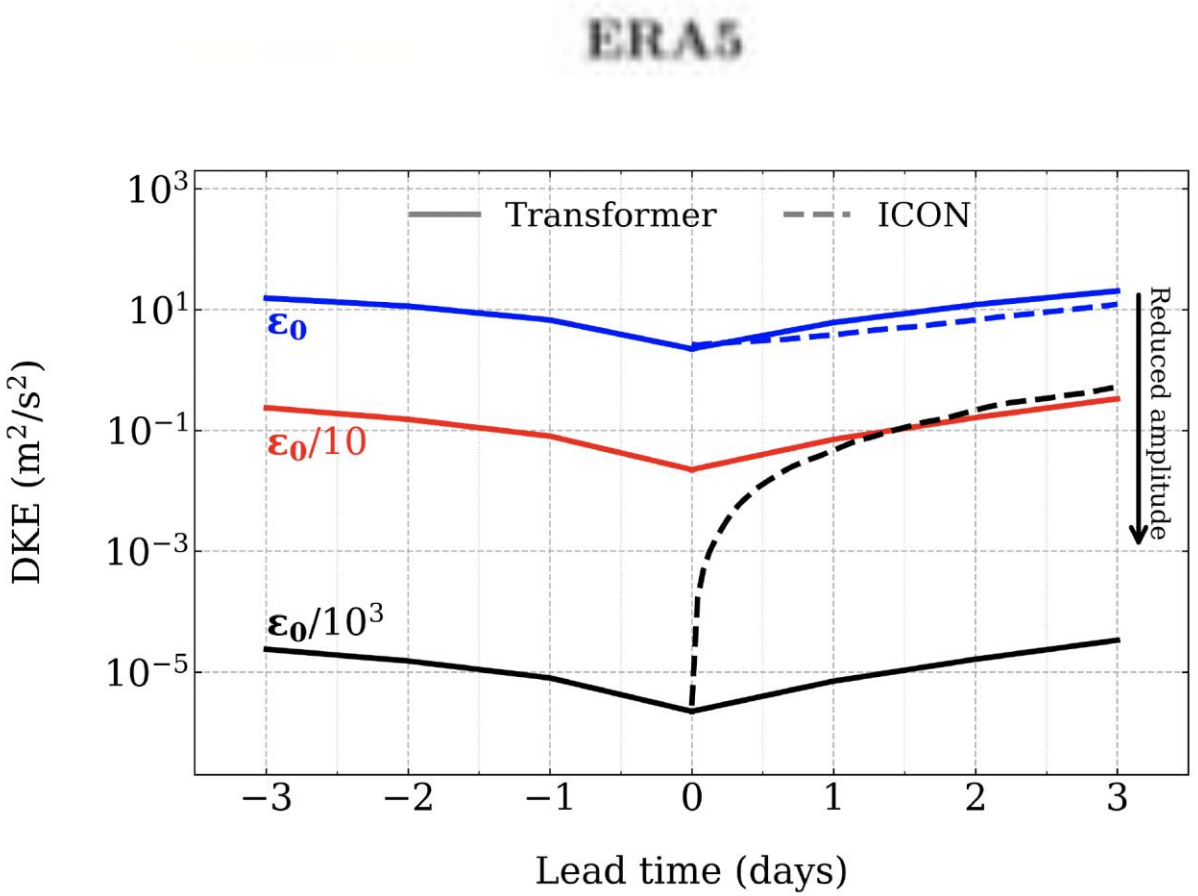


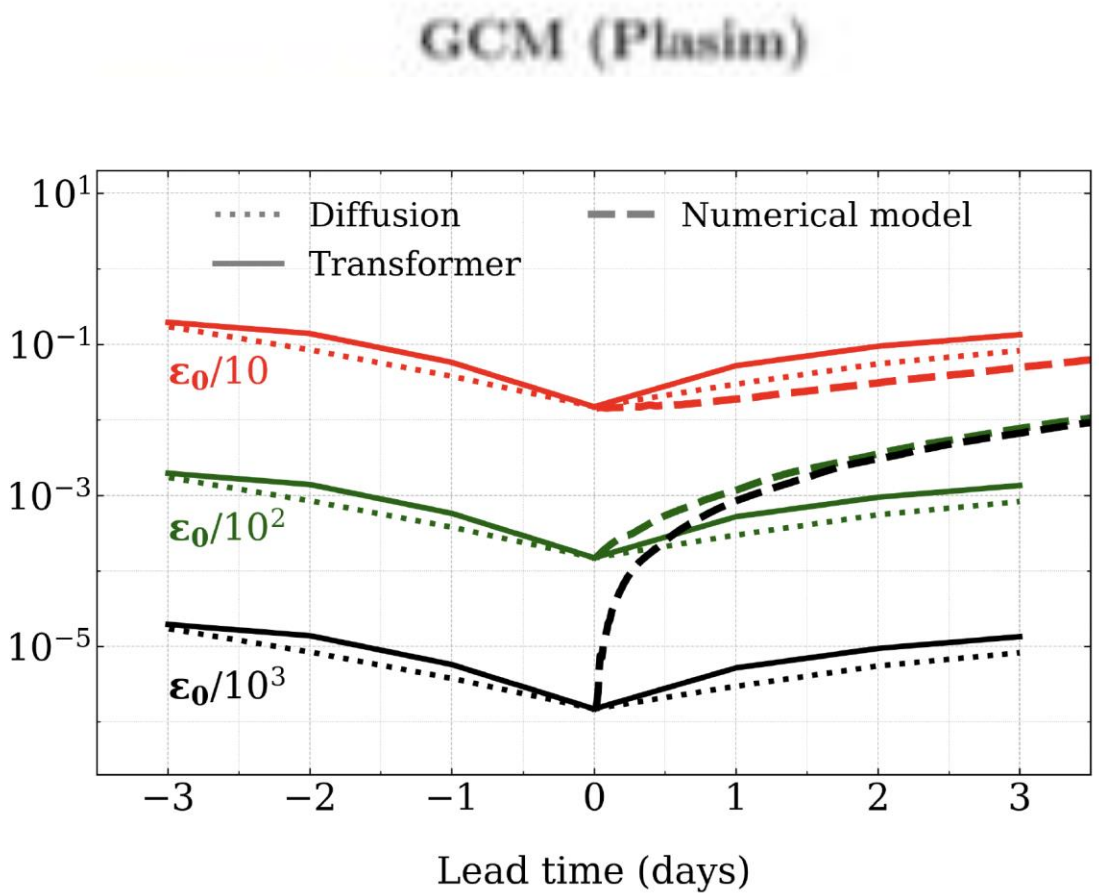




Missing Butterfly: Consistent Across ERA5, a GCM & Architectures

No butterfly in the future or past predictions!





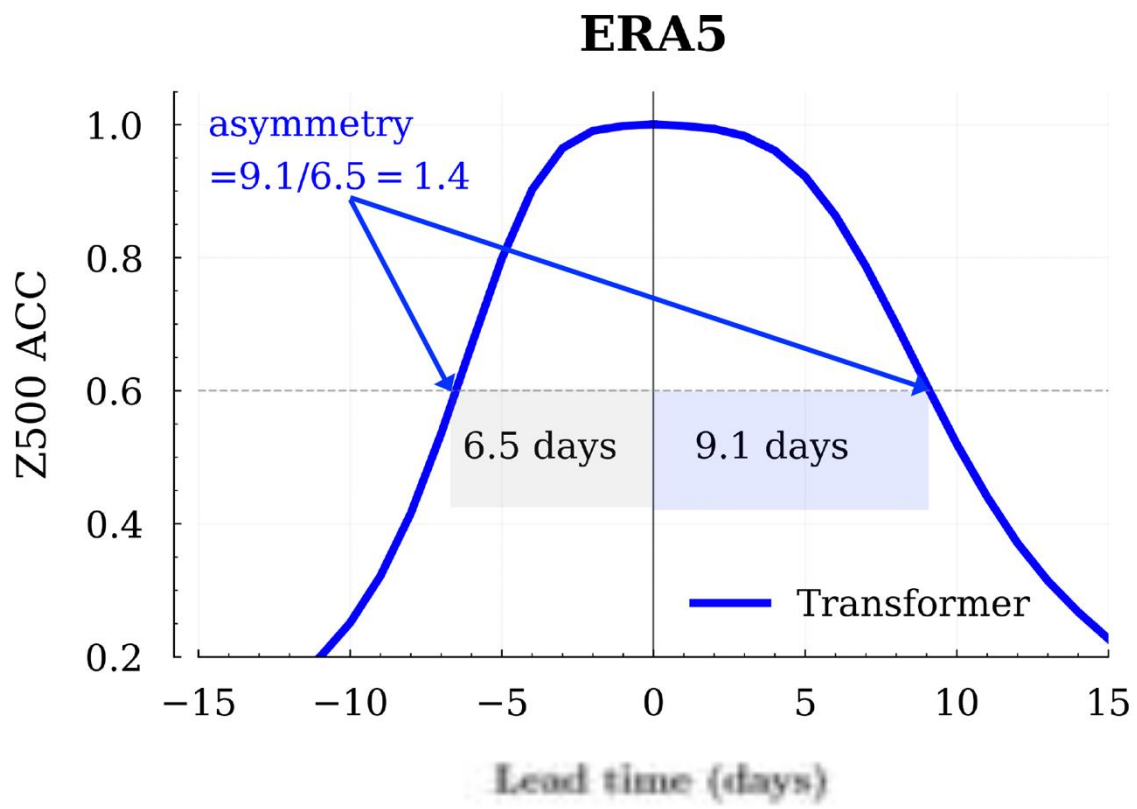
Consistent Across ERA5, a GCM & Architectures

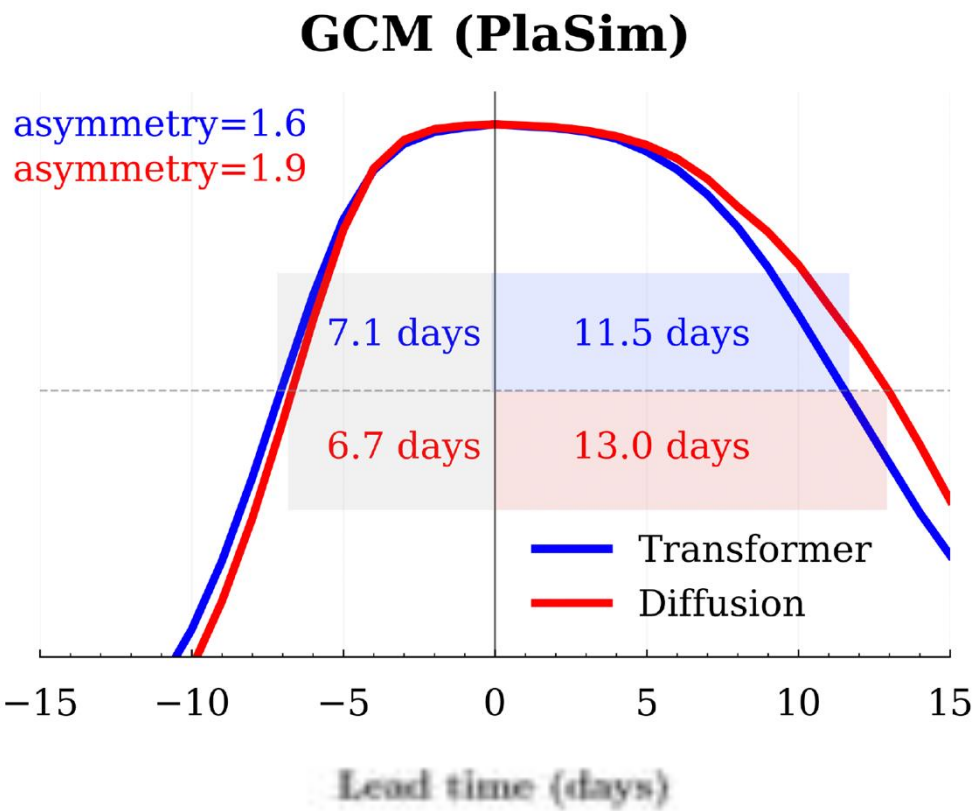
Why is the past harder to predict? It should be practically impossible to predict!

$$\frac{\partial T}{\partial t} = D\frac{\partial^2 T}{\partial x^2}$$

$$T(k,t) \sim e^{-Dk^2 t}$$

$$T(k,-t) \sim e^{+Dk^2 t}$$





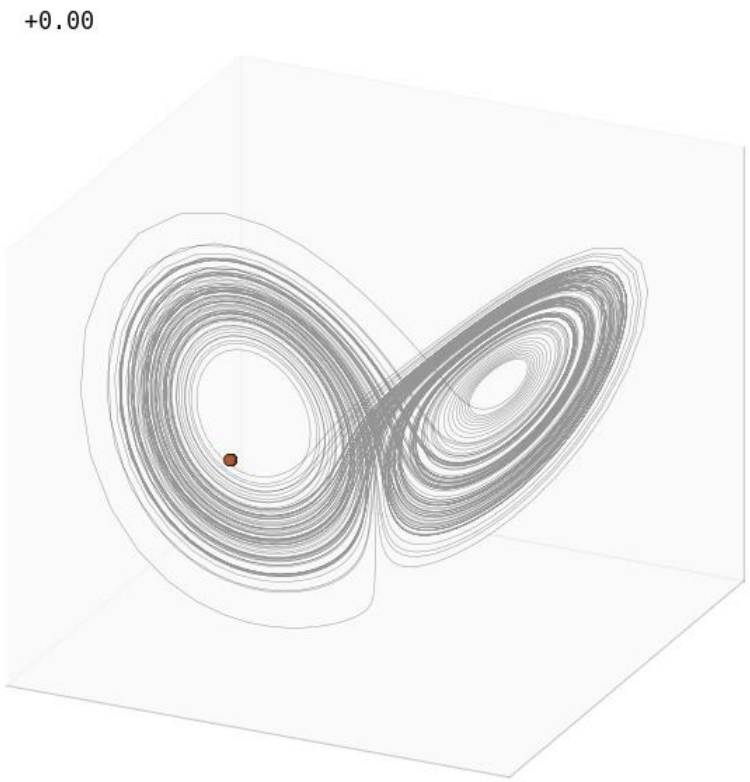
Predicting the Past of Dissipative Systems is *Practically* Impossible!

So how can AI models do this?

$$e^{\lambda_i t}$$

$\lambda_1 = +0.9, \lambda_2 = 0, \lambda_3 = -14.57$

$\sum \lambda_i < 0$: phase−space volume contracts



$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x),$$

$$\frac{\mathrm{d}y}{\mathrm{d}t} = x(\rho - z) - y,$$

$$\frac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z.$$

Generated by Claude

Predicting the Past of Dissipative Systems is *Practically* Impossible!

So how can AI models do this?

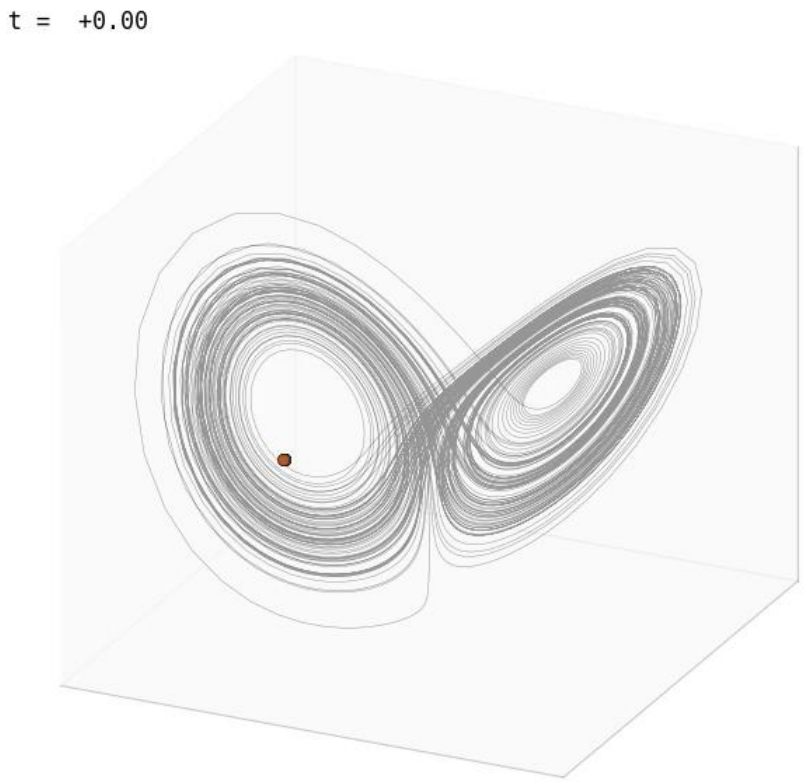
$e^{\lambda_i t}$

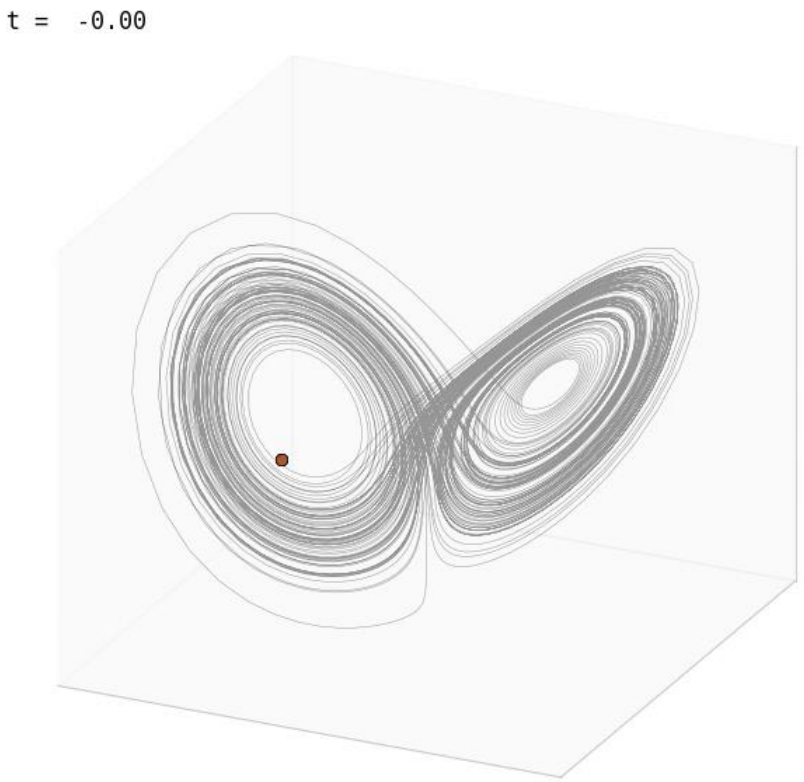
$\lambda_1 = +0.9, \lambda_2 = 0, \lambda_3 = -14.57$

$\sum \lambda_i < 0$: phase−space volume contracts

$\lambda_1 = +14.57, \lambda_2 = 0, \lambda_3 = -0.9$

$\sum \lambda_i > 0$: phase−space volume expands





$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x),$$

$$\frac{\mathrm{d}y}{\mathrm{d}t} = x(\rho - z) - y,$$

$$\frac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z.$$

Generated by Claude

Predicting the Past of Dissipative Systems is *Practically* Impossible!

So how can AI models do this?

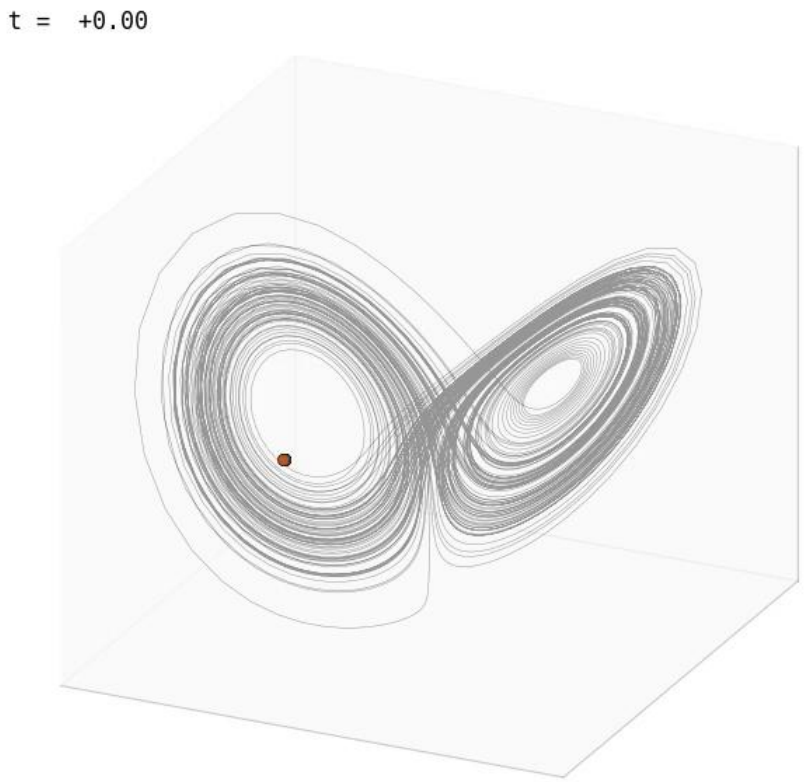
$e^{\lambda_i t}$

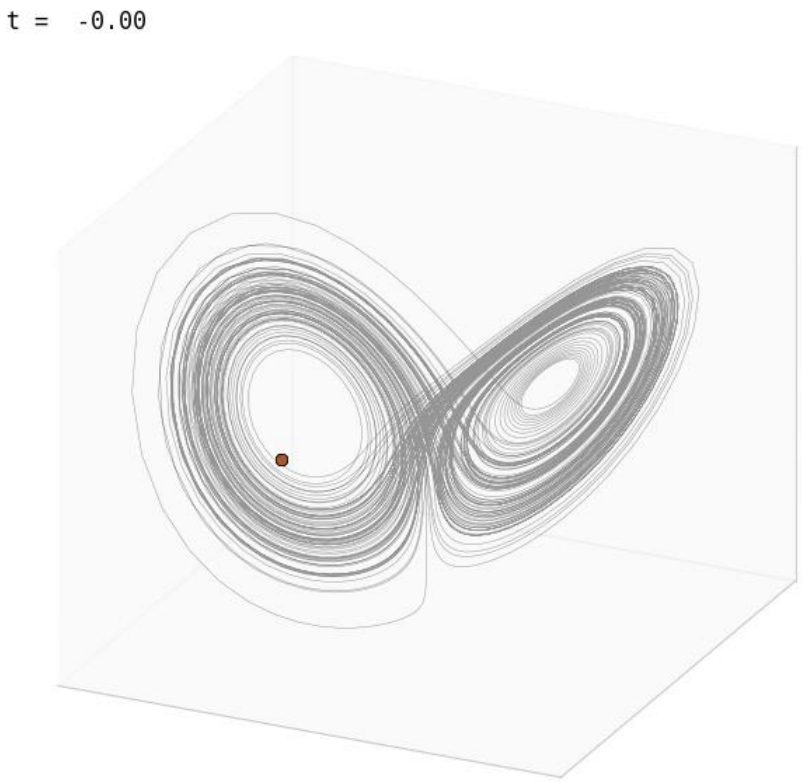
$\lambda_1 = +0.9, \lambda_2 = 0, \lambda_3 = -14.57$

$\sum \lambda_i < 0$: phase−space volume contracts

$\lambda_1 = +14.57, \lambda_2 = 0, \lambda_3 = -0.9$

$\sum \lambda_i > 0$: phase−space volume expands





$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma(y - x),$$

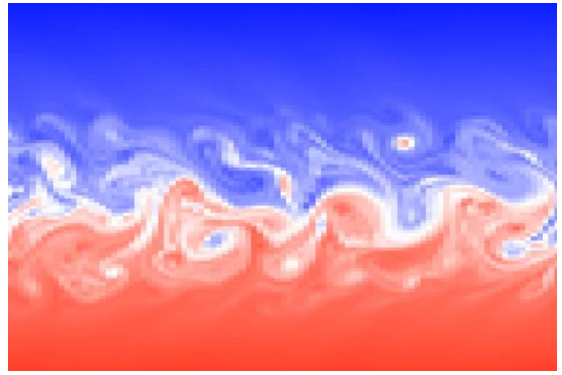
$$\frac{\mathrm{d}y}{\mathrm{d}t} = x(\rho - z) - y,$$

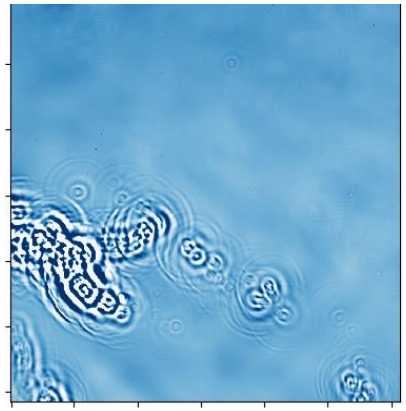
$$\frac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z.$$

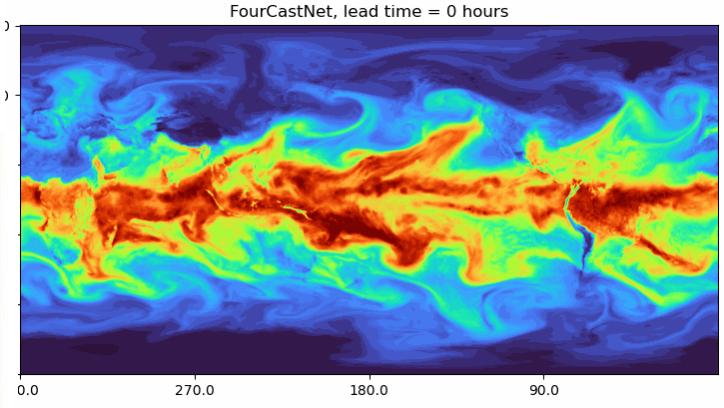
Generated by Claude

Quasi-geostrophic (QG) turbulence

ML for DA, structural error, forecasting, interpretability, transfer learning





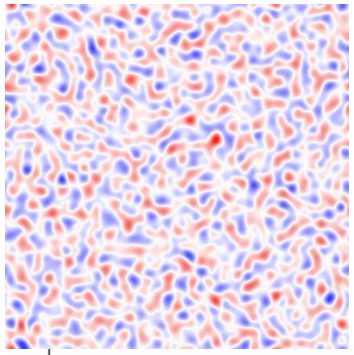


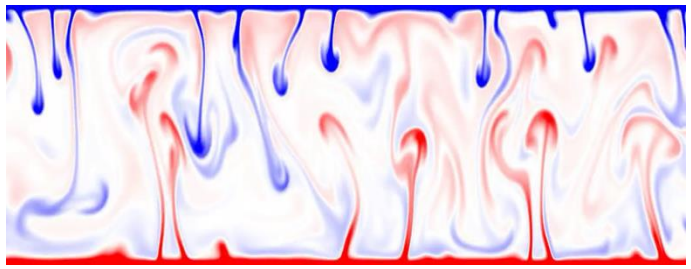
Climate data & models

Extreme weather forecasting, DA for ML, stability

2D turbulence & Rayleigh-Benard

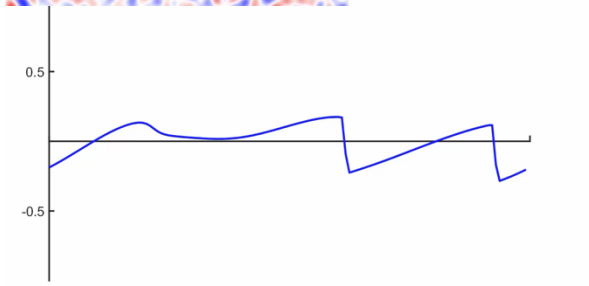
SGS modeling, transfer learning, interpretability, equation discovery





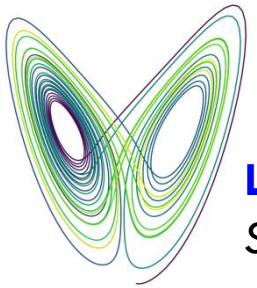
Gravity waves

SGS modeling, learning rare events UQ, transfer learning



Burgers turbulence

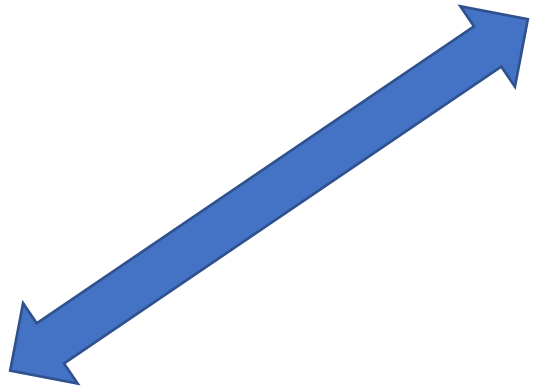
SGS modeling, Lagrangian PINN



Lorenz systems, Kuramoto-Sivashinsky equation

SGS modeling, forecasting, structural error, stability

Scalability



Understanding

Multi-scale Lorenz 96

$$\frac{dX_k}{dt} = \underbrace{-X_{k-1}(X_{k-2} - X_{k+1})}_{\text{Advection}} \underbrace{-X_k}_{\text{Diffusion}} \underbrace{+F}_{\text{Forcing}} \underbrace{-hc\bar{Y}_k}_{\text{Coupling}}$$

$$\frac{1}{c}\frac{dY_{j,k}}{dt} = \underbrace{-bY_{j+1,k}(Y_{j+2,k} - Y_{j-1,k})}_{\text{Advection}} \underbrace{-Y_{j,k}}_{\text{Diffusion}} \underbrace{+\frac{h}{J}X_k}_{\text{Coupling}}$$

$k = 1 \ldots 8$

$j = 1 \ldots 32$

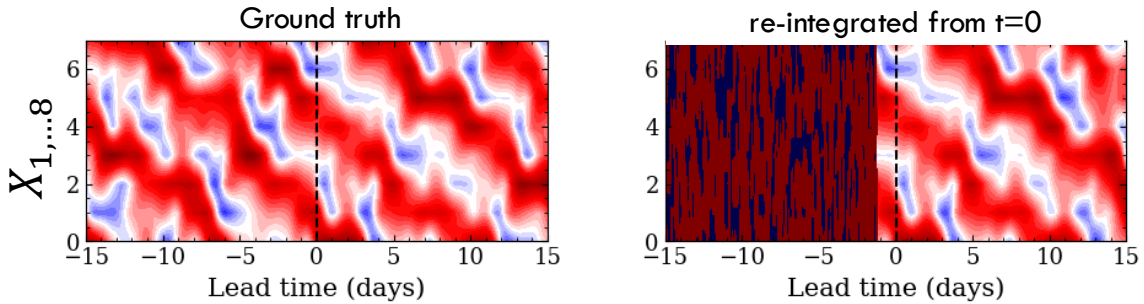


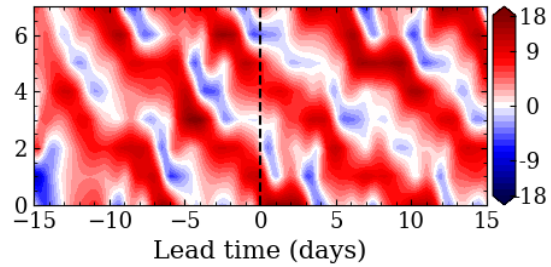
$$\boldsymbol{X} = \{X_k\}$$
$$k = 1 \ldots 8$$

$$\boldsymbol{X}(t + 10\delta t) = \mathbf{NN}(\boldsymbol{X}(t), \boldsymbol{\theta})$$

$$\mathcal{L} = \|\boldsymbol{X}(t + 10\delta t) - \mathbf{NN}(\boldsymbol{X}(t), \boldsymbol{\theta})\|_2$$

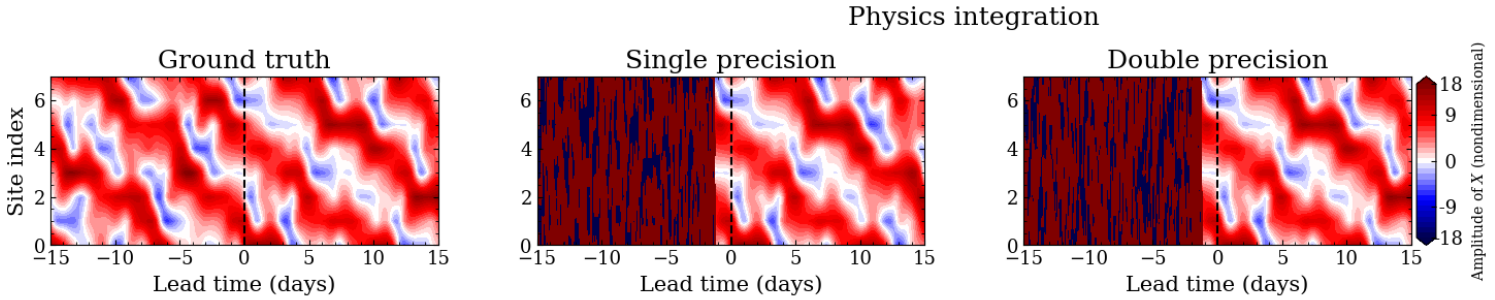
From numerical integrations



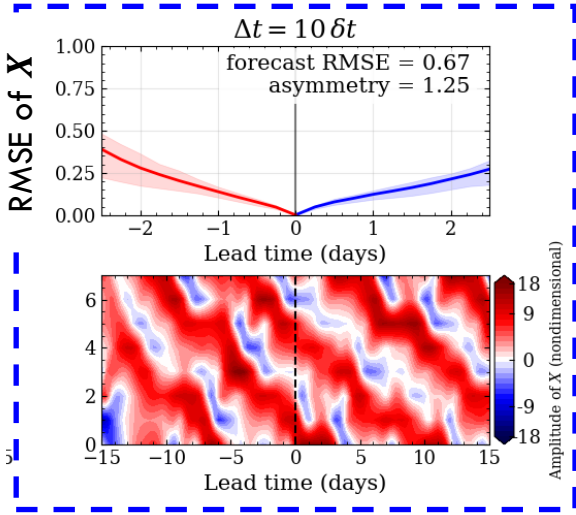


Effects of Coarse-grained Training Data?

Filtered fast/small scales, missing variables



Common AI models



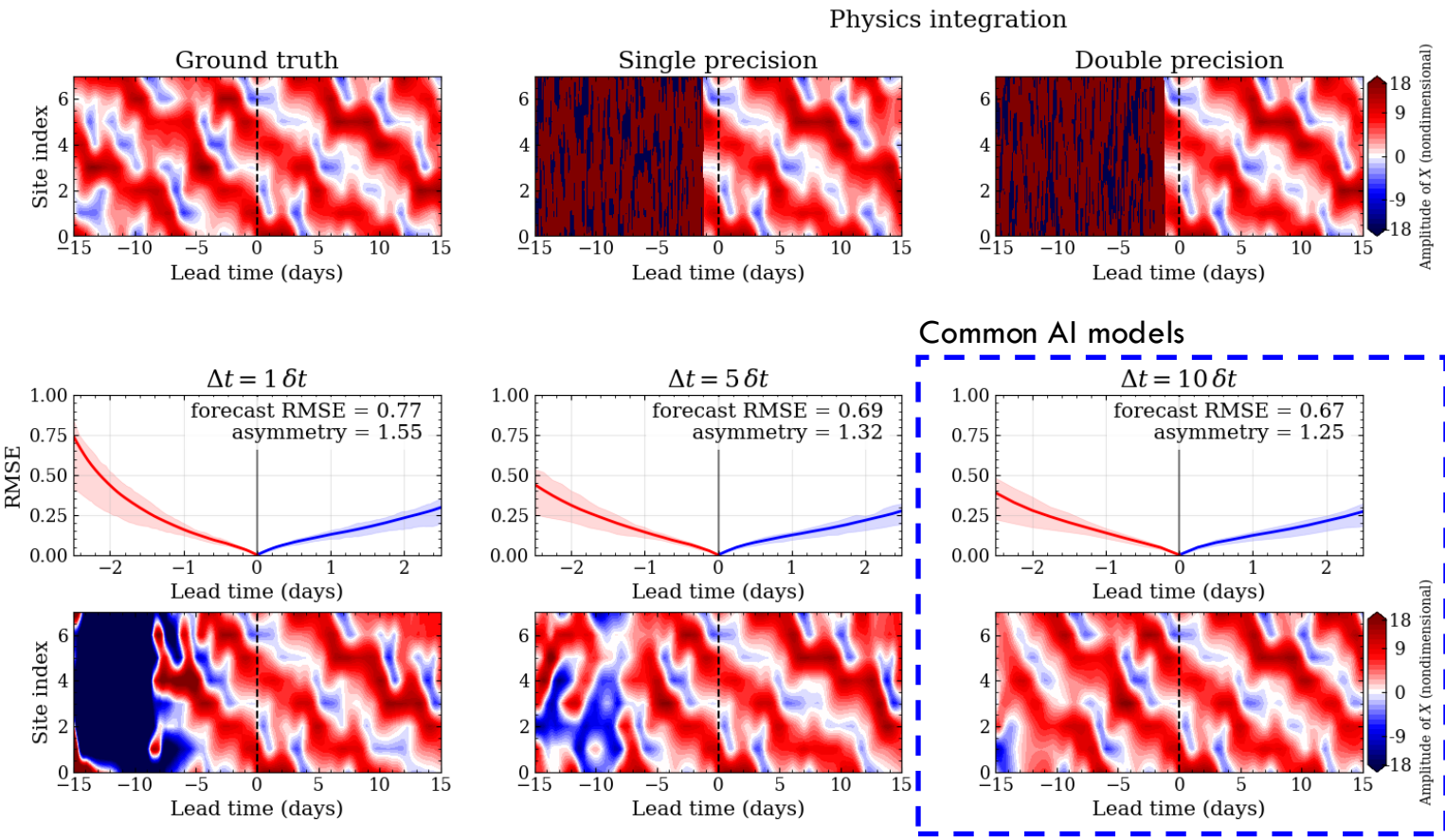
$$\boldsymbol{X} = \{X_k\}$$
$$k = 1 \ldots 8$$

$$\boldsymbol{X} = \{X_k, Y_{j,k}\}$$
$$j = 1 \ldots 32$$

Effects of Coarse-grained Training Data?

Filtered fast/small scales, missing variables

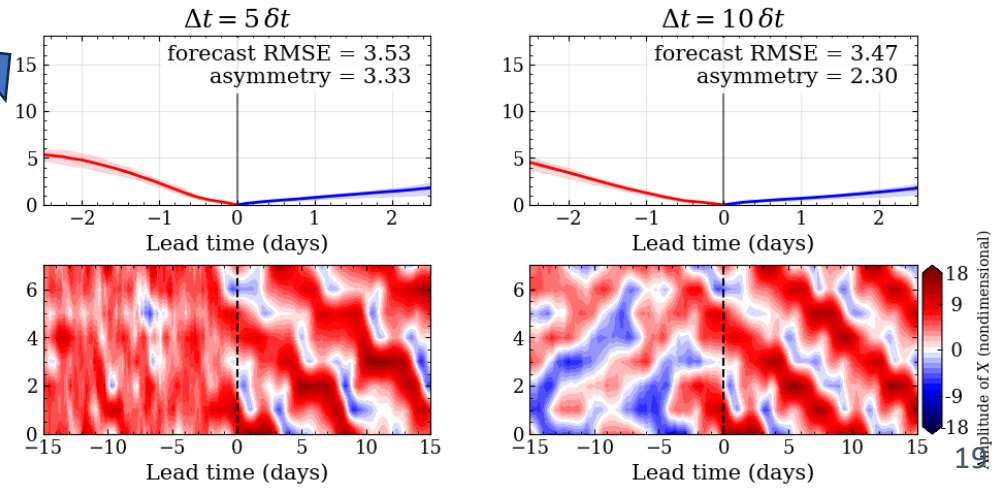
$$X = \{X_k\}$$
$$k = 1 \ldots 8$$



physics vs. performance

More accurate forecasts

More small/fast scales
More asymmetry
More like physics!

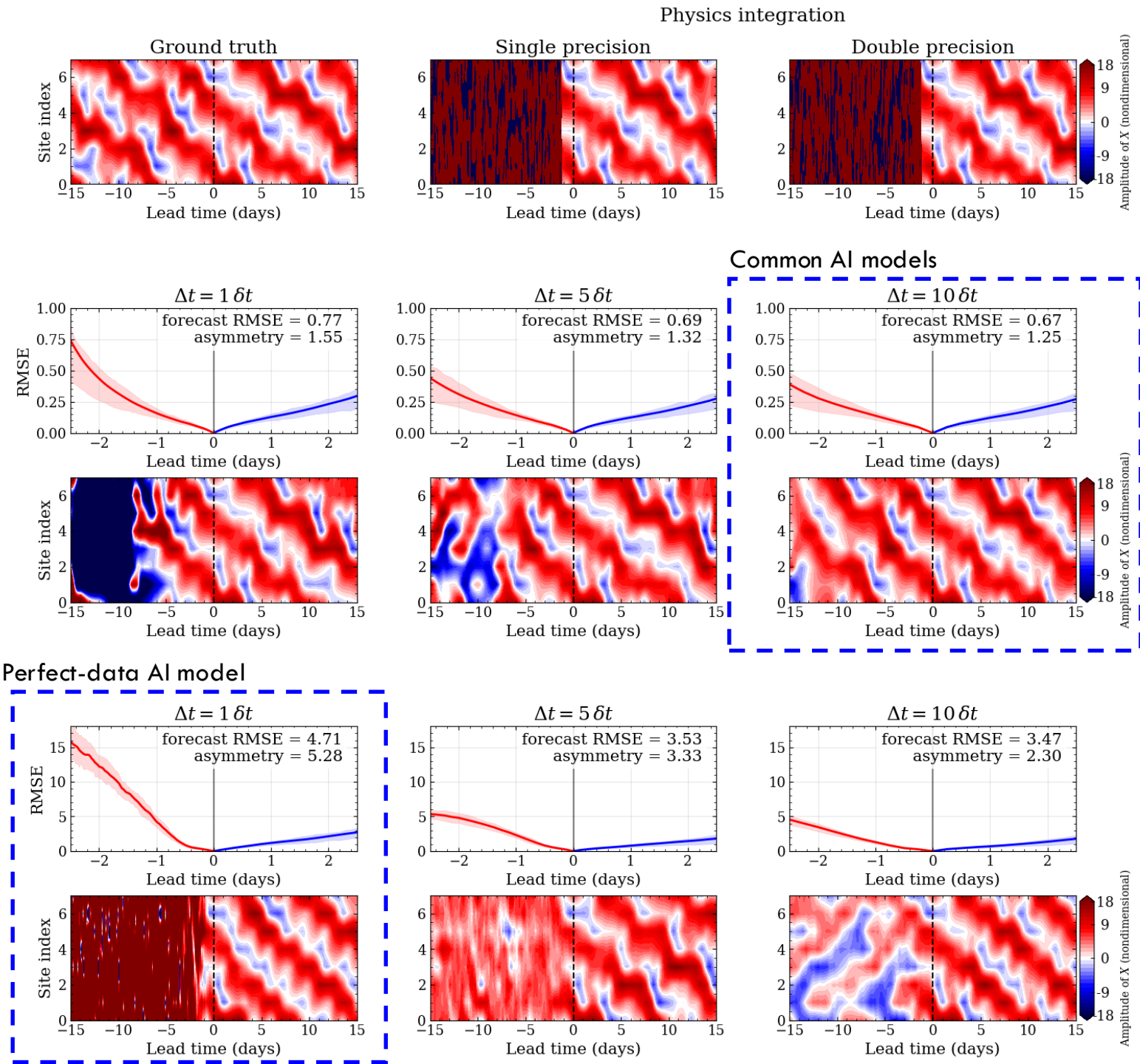


Effects of Coarse-grained Training Data?

Filtered fast/small scales, missing variables

$$\boldsymbol{X} = \{X_k\}$$
$$k = 1 \ldots 8$$

$$\boldsymbol{X} = \{X_k, Y_{j,k}\}$$
$$j = 1 \ldots 32$$



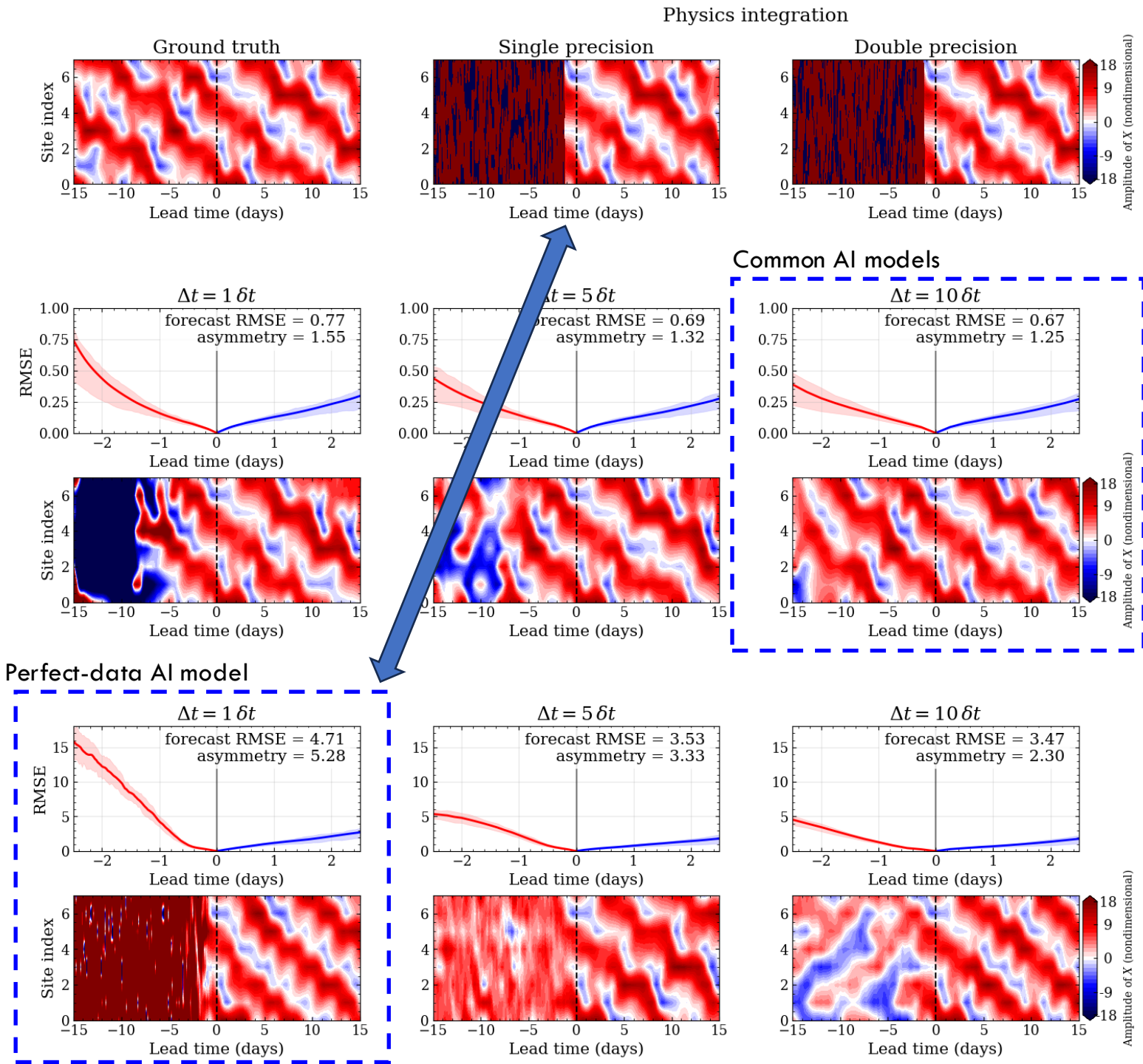
Effects of Coarse-grained Training Data?

Filtered fast/small scales, missing variables

Coarse-graining leads to:

1. Skillful backcasting (still less accurate than forecasting)

2. Enhanced AI forecast skill



Effects of Coarse-grained Training Data?

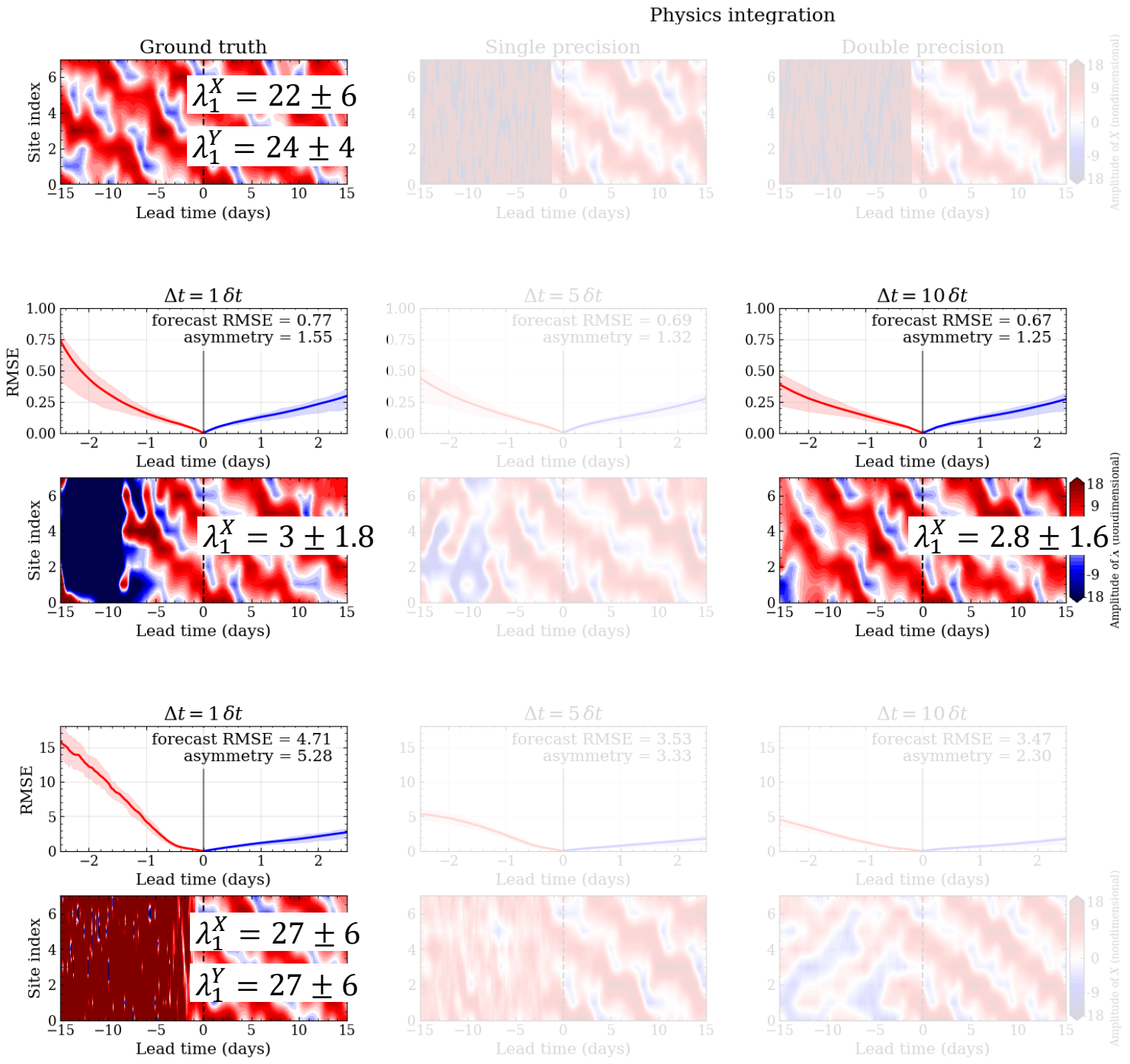
Filtered fast/small scales, missing variables

Coarse-graining leads to:

1. Skillful backcasting (still less accurate than forecasting)

2. Enhanced AI forecast skill

Large-scales' Lyapunov exponents are slaved to small-scales' Lyapunov exponents



Effects of Coarse-grained Training Data?

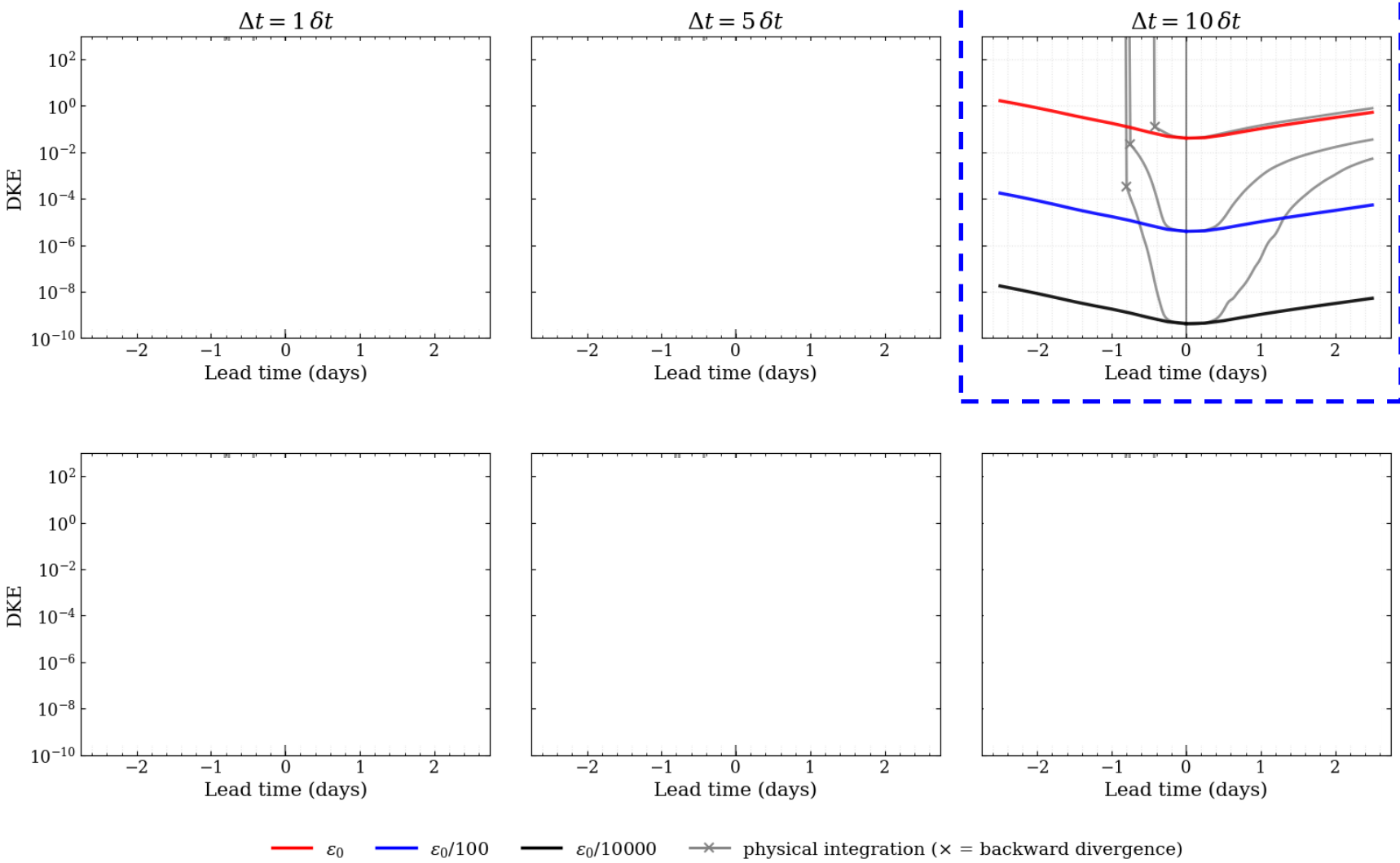
Filtered fast/small scales, missing variables



Common AI models

$\{X_k\}$
$k = 1 \ldots 8$

$\{X_k, Y_{j,k}\}$
$j = 1 \ldots 32$

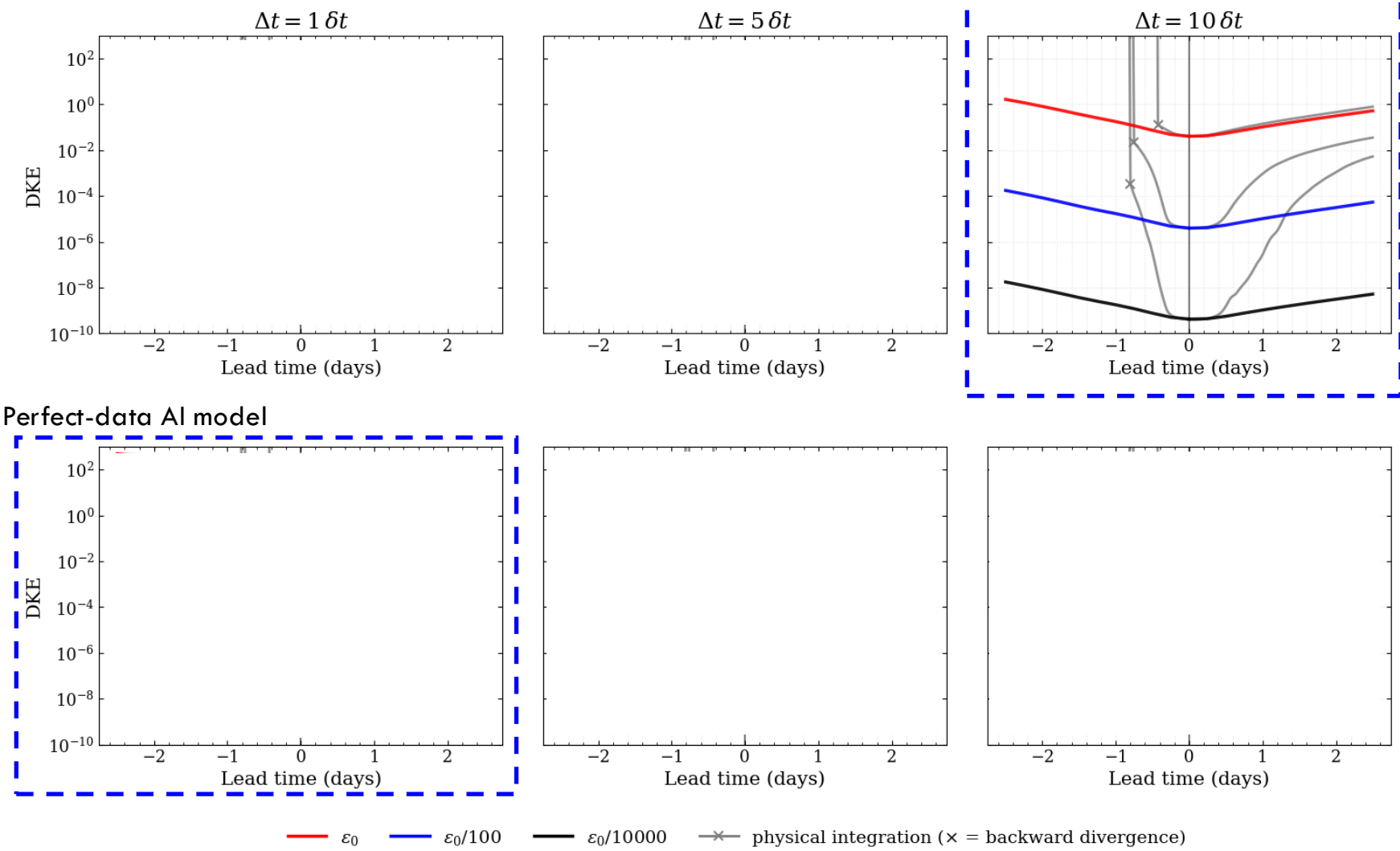


Effects of Coarse-grained Training Data?

Filtered fast/small scales, missing variables



Common AI models



Perfect-data AI model

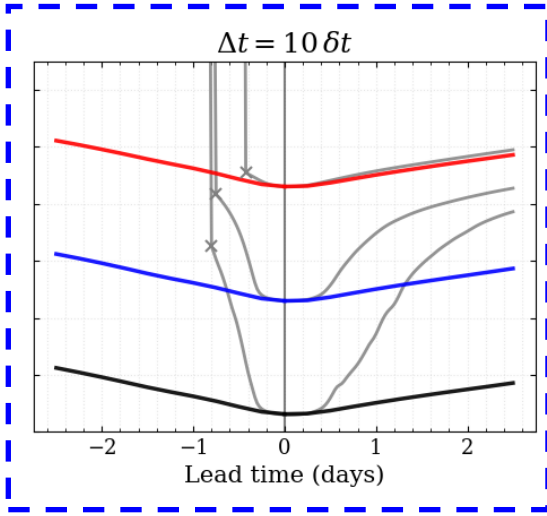
ε_0 $\varepsilon_0/100$ $\varepsilon_0/10000$ physical integration (× = backward divergence)

Effects of Coarse-grained Training Data?

Filtered fast/small scales, missing variables



Common AI models

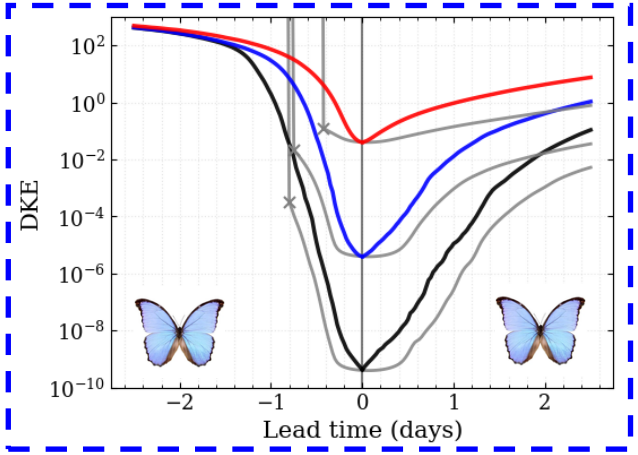


More accurate forecasts

physics vs. performance

More small/fast scales
More like physics!

Perfect-data AI model





Effects of Coarse-grained Training Data?

$$\boldsymbol{X}(t+\Delta t) = \mathbf{NN}(\boldsymbol{X}(t), \boldsymbol{\theta})$$

$$\mathcal{L} = \|\boldsymbol{X}(t+\Delta t) - \mathbf{NN}(\boldsymbol{X}(t), \boldsymbol{\theta})\|_2$$



Filtered fast/small scales, missing variables

Coarse-graining leads to:

1. Skillful backcasting (still less accurate than forecasting)

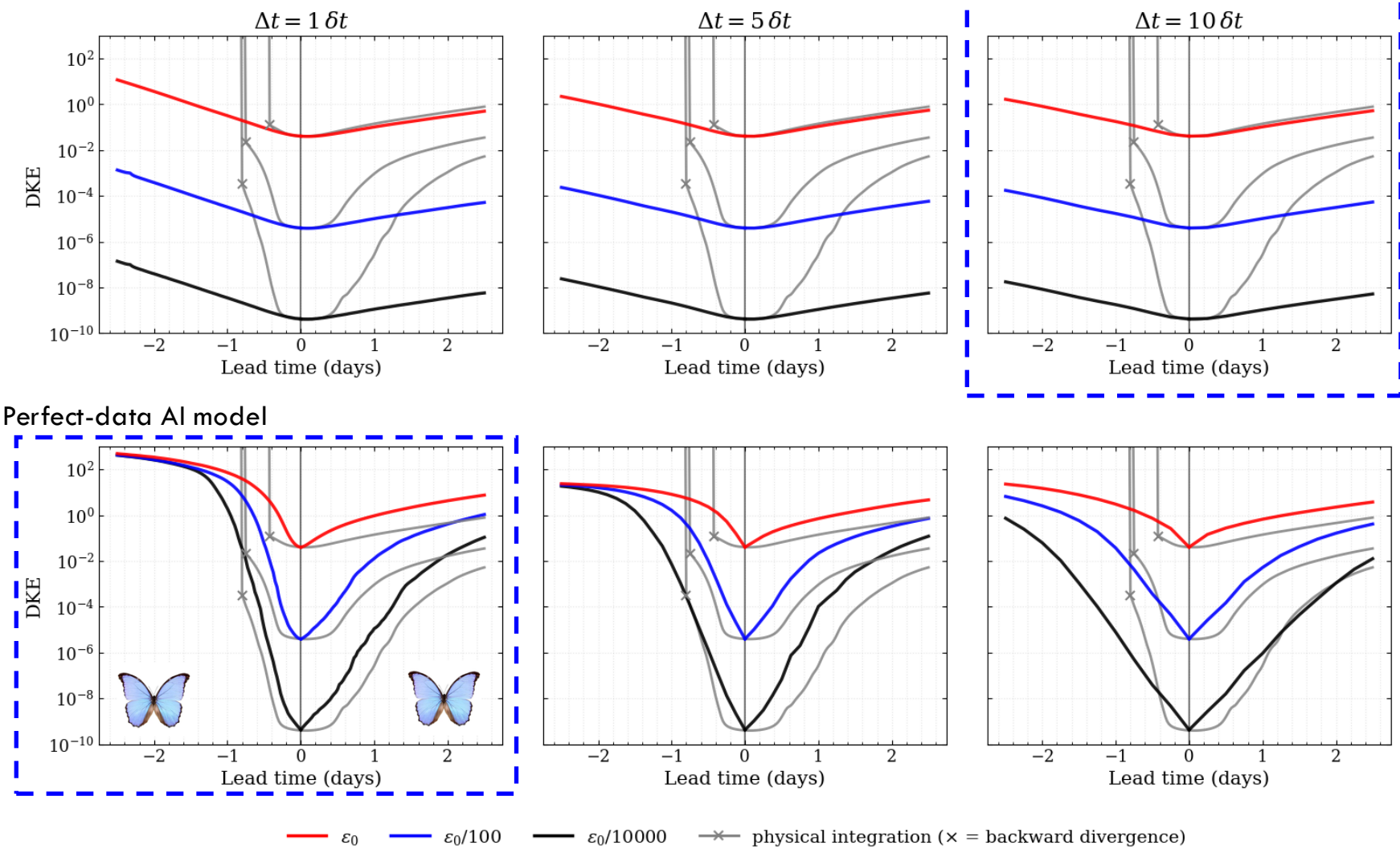
2. Enhanced AI forecast skill

3. Missing butterfly effect

(not MSE, not blurring etc.)

Common AI models

Perfect-data AI model



Effects of Coarse-grained Training Data?

Filtered fast/small scales, missing variables

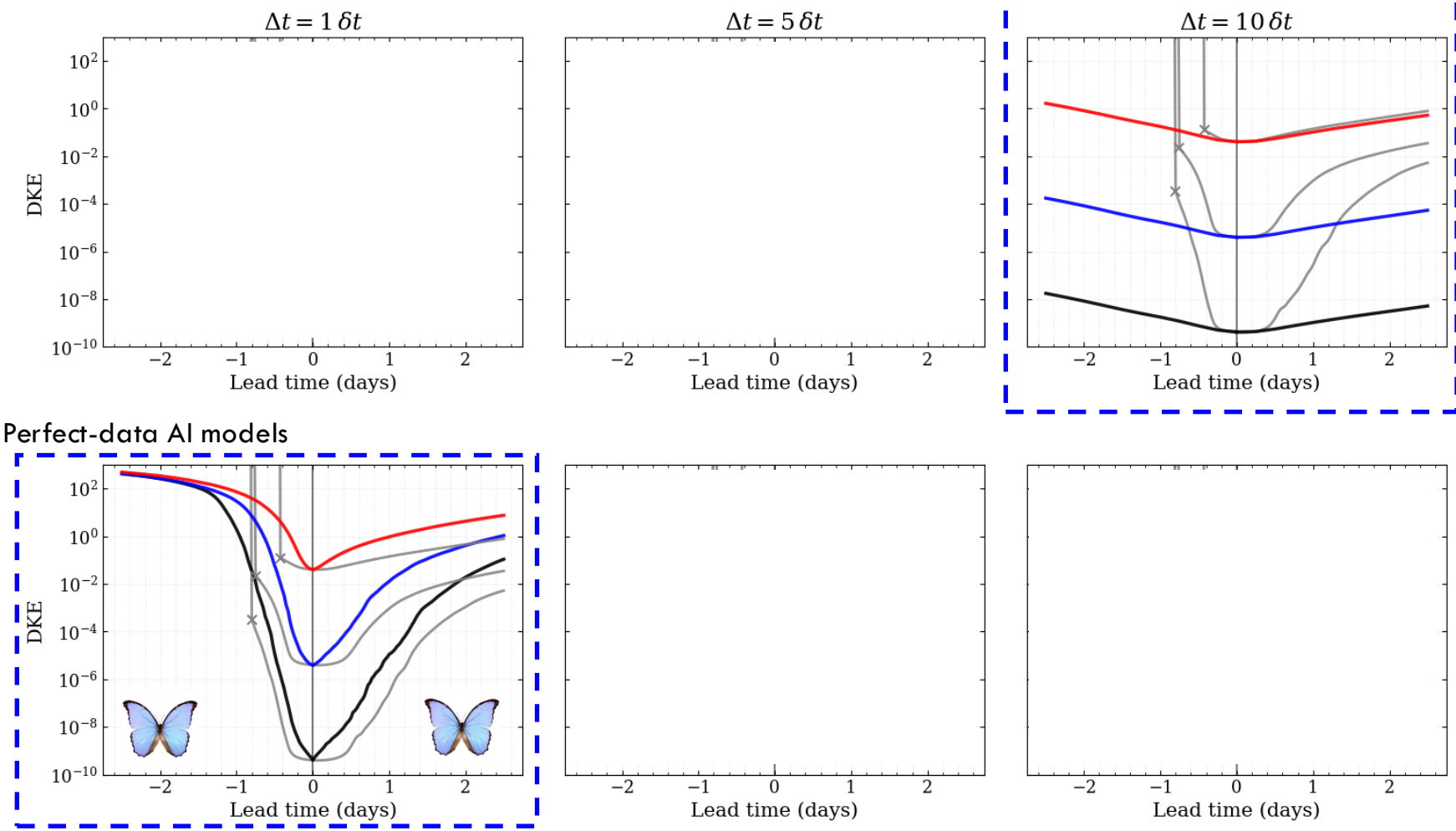


Interpreting the common AI models?

At best, we are learning about an *effective, smoothed version of the real world*, that has different physics:
wrong Lyap exponents, no butterflies, and weaker arrow of time!

video-like?

Common AI models



AI Forecasts of ERA5: We Can Get Butterfly-like Effects

Butterflies emerge when coarse-graining is reduced, but forecast accuracy drops!

Coarse-graining changes physics vs. performance:

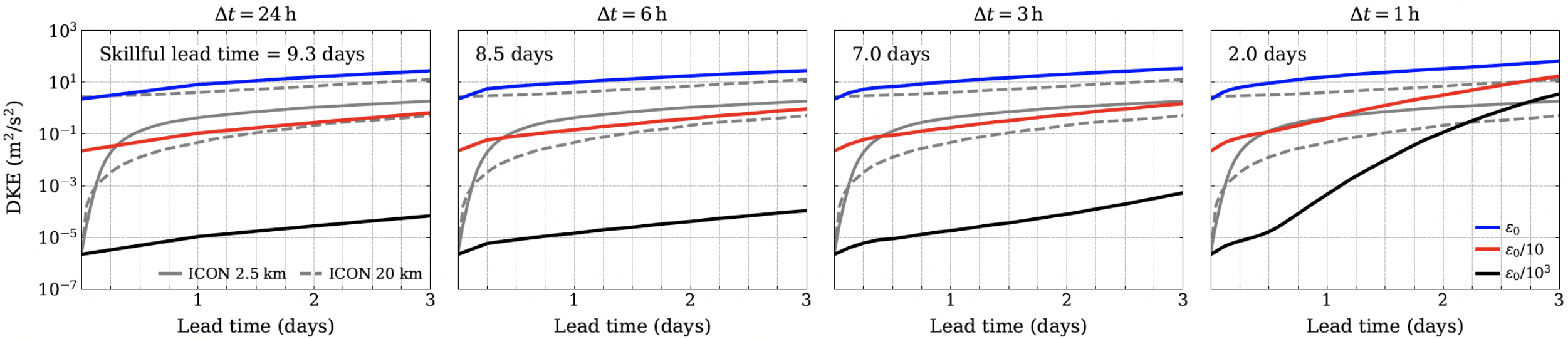
2. Enhanced AI forecast skill

3. Missing butterfly effect

Even full ERA5 data is already coarse-grained!

Next step: controlled spatio-temporal coarse graining of very high-resolution regional and global simulations

Official Pangu models (Bi et al, 2023 Nature): Fixed 0.25° resolution



Physics vs. Performance: It Can Come From the Training Data!

Do we want AI models to behave physics-like if that costs performance?

Inevitable coarse-graining of training data (filtered fast/small scales, missing variables) removes physics from AI's sight, leading to:

- Enhanced AI forecasting skill: AI weather models do "implicit subgrid-scale parameterization" (unlike NWP)

- Missing butterfly effect: But AI weather models that keep the butterfly forecast worse!

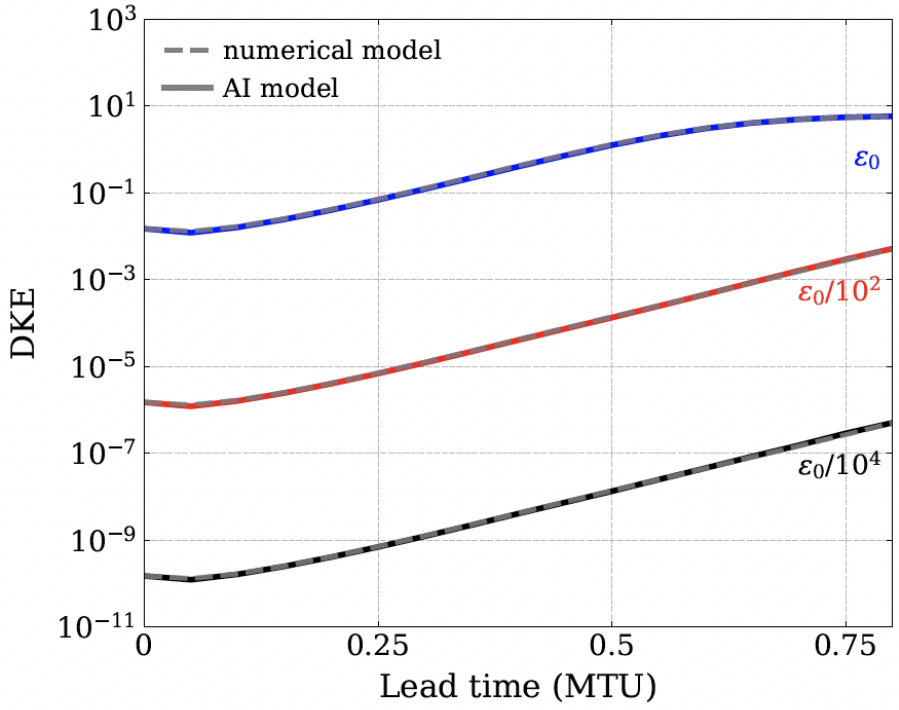
Is this stopping AI models from "learning" physics? Learning gray swans? Implications for ensemble S2S/probabilistic forecasting?

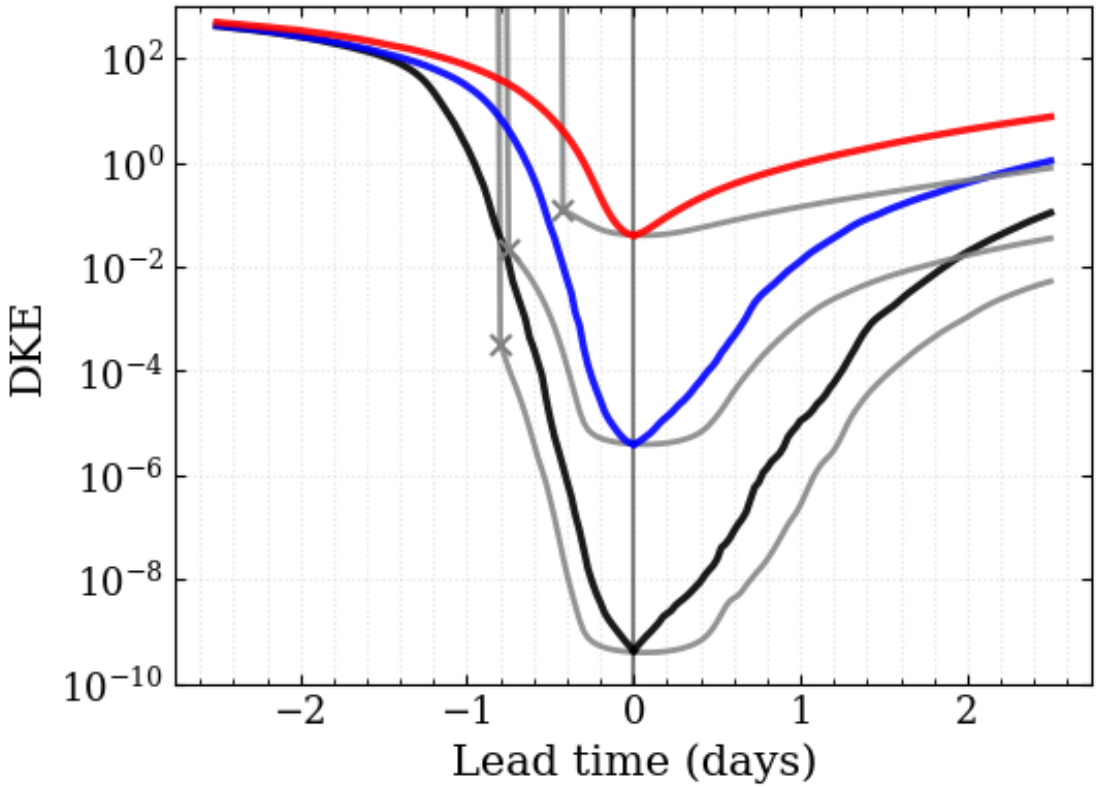
Fancier AI can help? (doubtful!) More math and physics are needed (Mori-Zwanzig?)

AI models learn an *effective smoothed dynamical system*, with key differences from the actual dynamical system. Implications for interpretability?

Lorenz 63

Chaotic? Yes. Real butterfly effect? No!





AI Forecasts of ERA5 and GCM: But Butterfly Costs Forecast Skill

Effects of numerical noise and unphysical behavior are significant

