Going with the flow

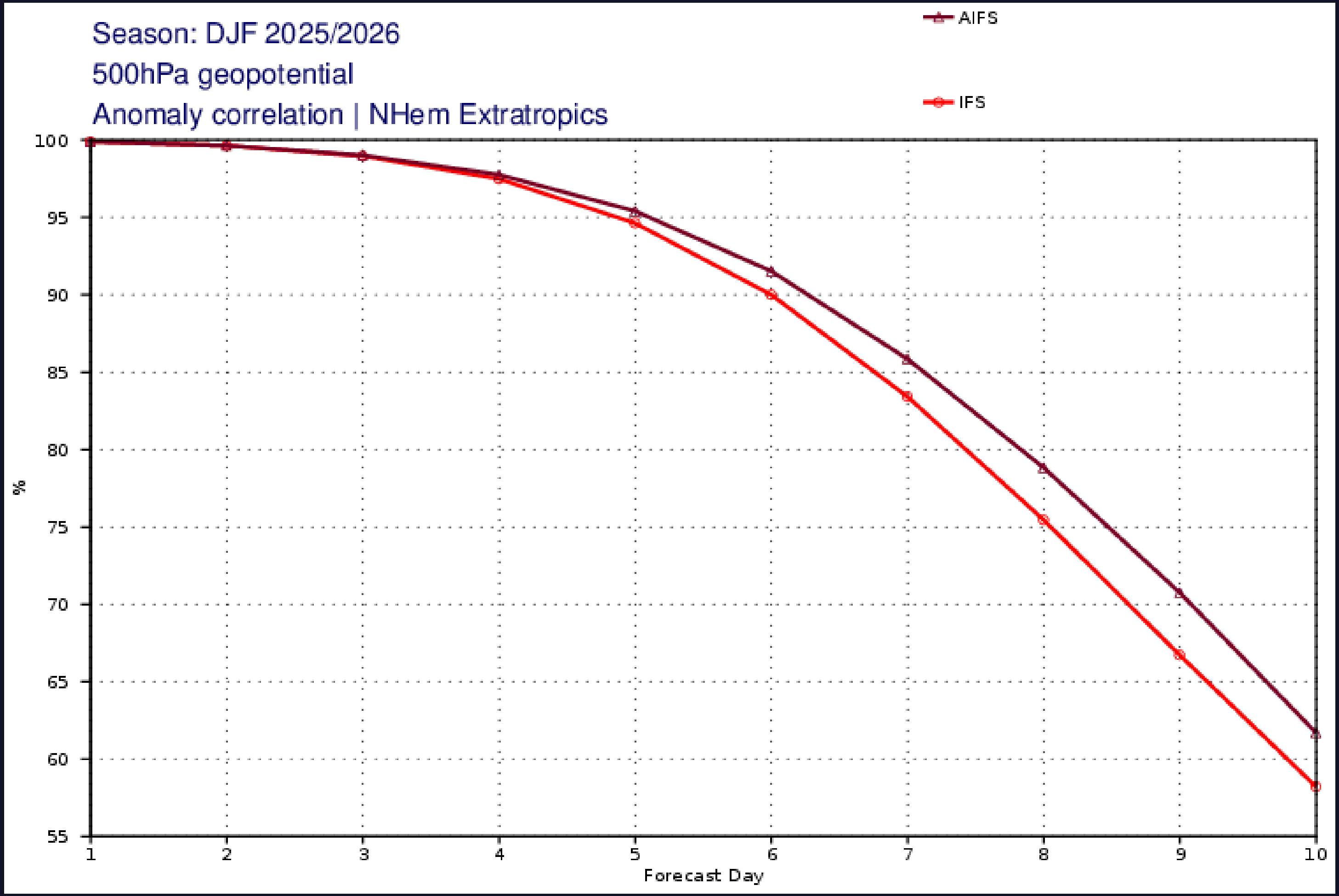
generalising physically consistent data-driven sea-ice models

Tobias Sebastian Finn

Marc Bocquet, Pierre Rampal, Freddy Bouchet, Alberto Carrassi, Charlotte Durand, Flavia Porro, Alban Farchi, et al.

This research was funded by Schmidt Sciences - a philanthropic initiative that seeks to improve societal outcomes through the development of emerging science and technologies

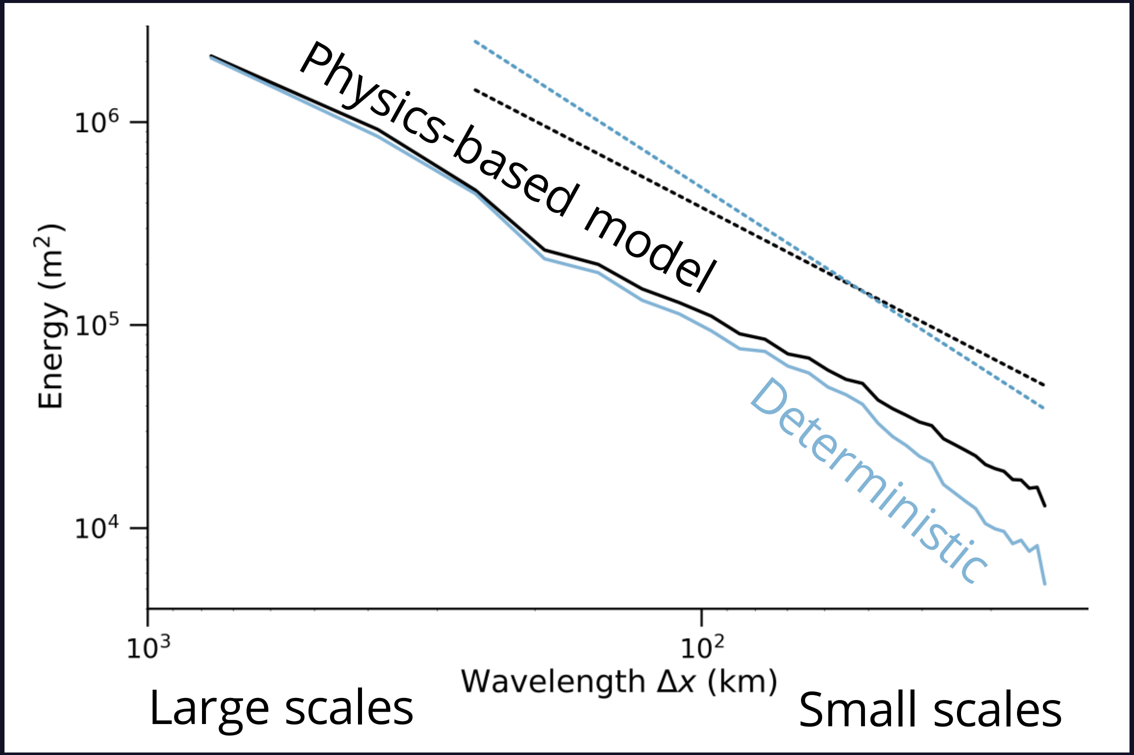
Deep learning based surrogate models are great ...



… but struggle with …

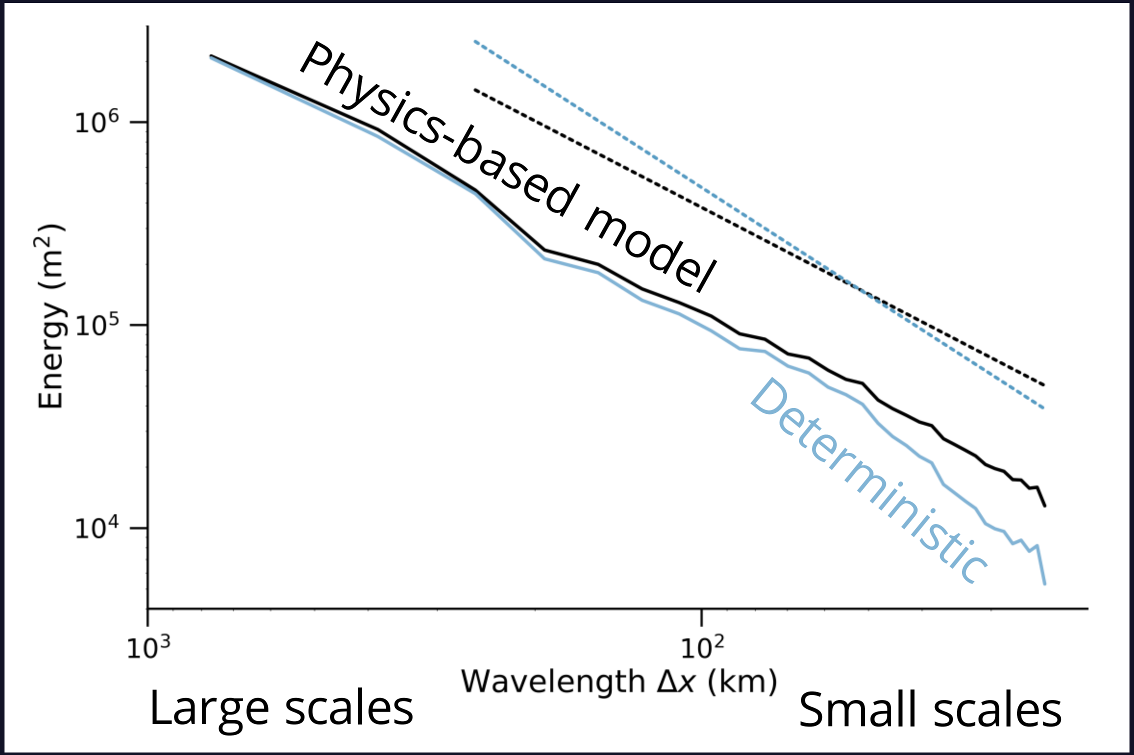
... but struggle with ...

... small-scale information

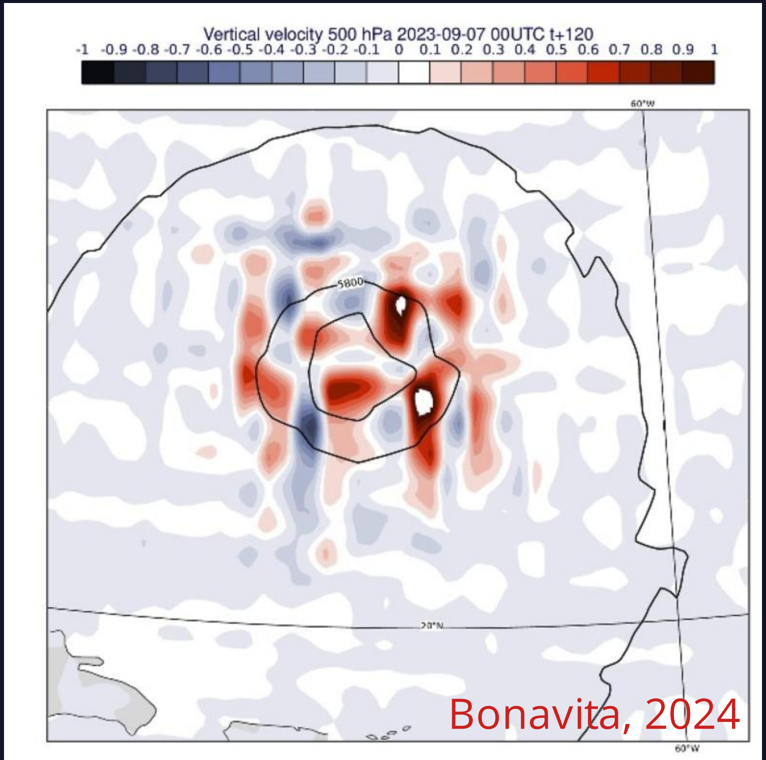


… but struggle with …

… small-scale information

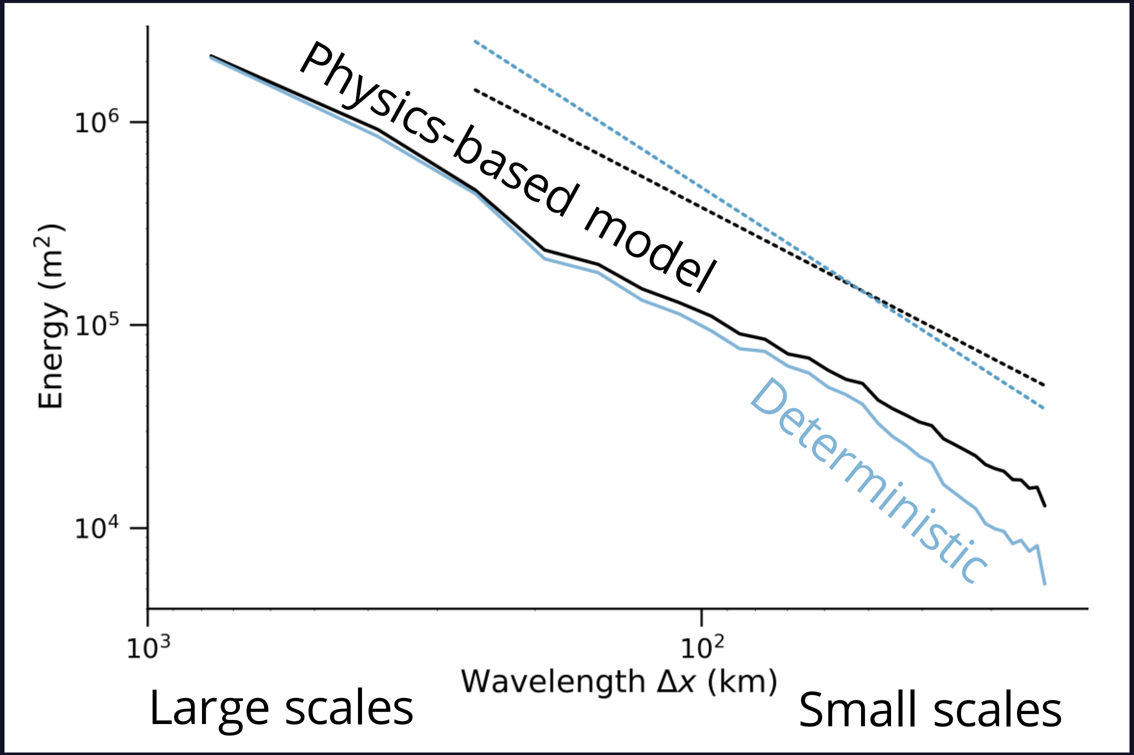


… physical realism

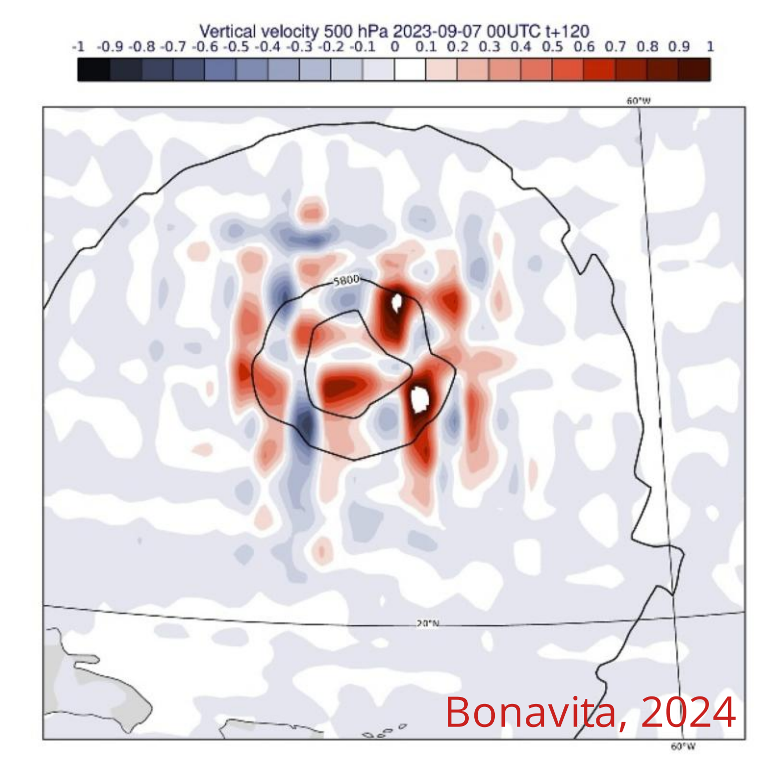


... but struggle with ...

... small-scale information



... physical realism



<u>Claim:</u> generative modelling is the right tool (at least for sea ice)

... the revolution of generative computer vision

Article

Probabilistic weathe
machine learning

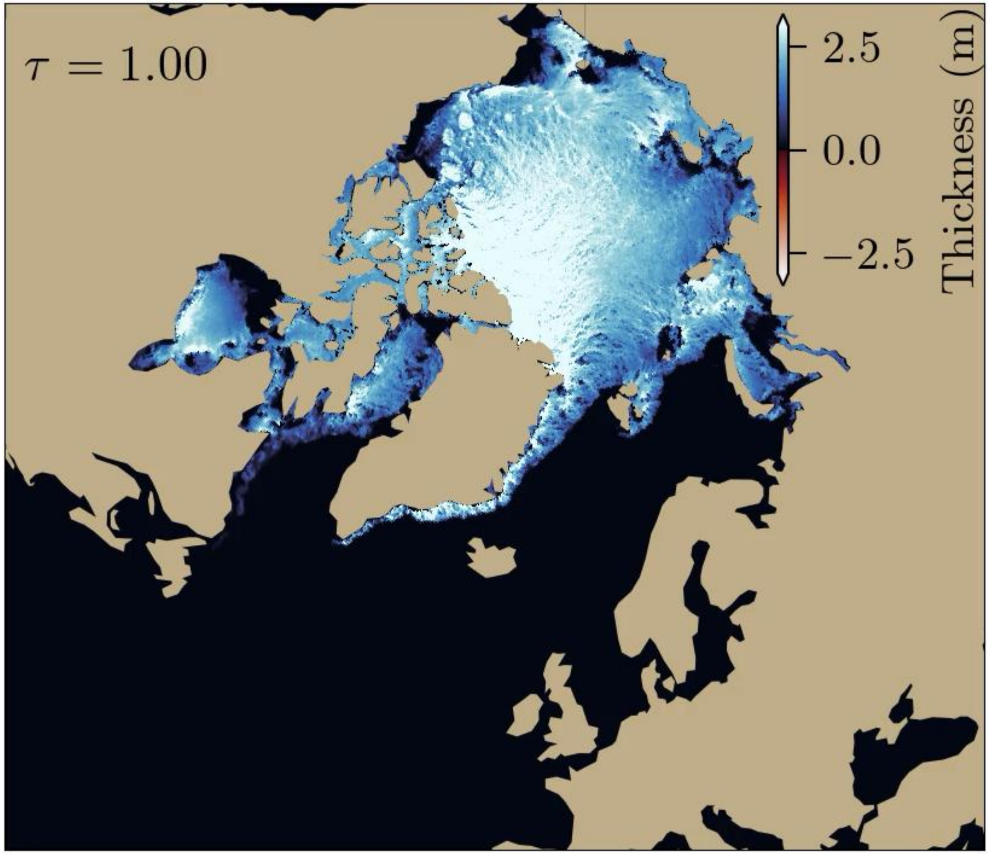
Climate in a Bottle: Towards a Generative Foundation Model for the Kilometer-Scale Global Atmosphere

Noah D. Brenowitz Tao Ge Akshay Subramaniam Aayush Gupta
David M.Hall Morteza Mardani Arash Vahdat Karthik Kashinath
Michael S. Pritchard

NVIDIA, Santa Clara, CA, USA

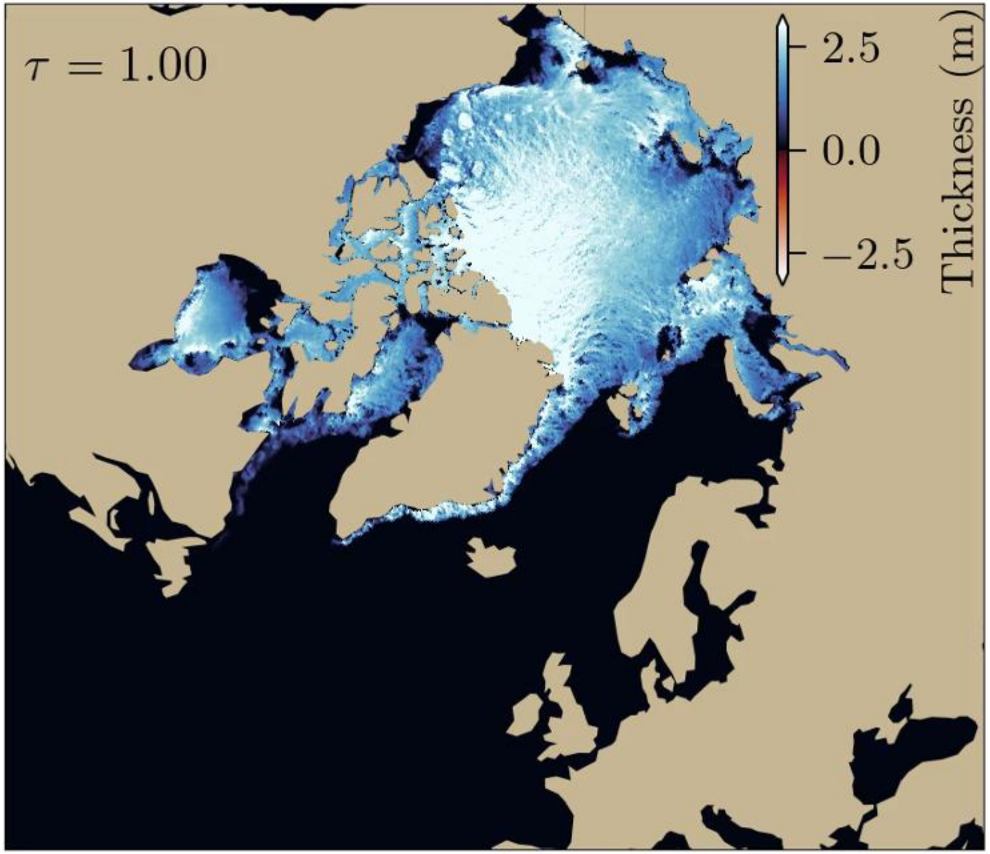
... the revolution of generative computer vision
starts also for Earth system modelling!

Destroy information



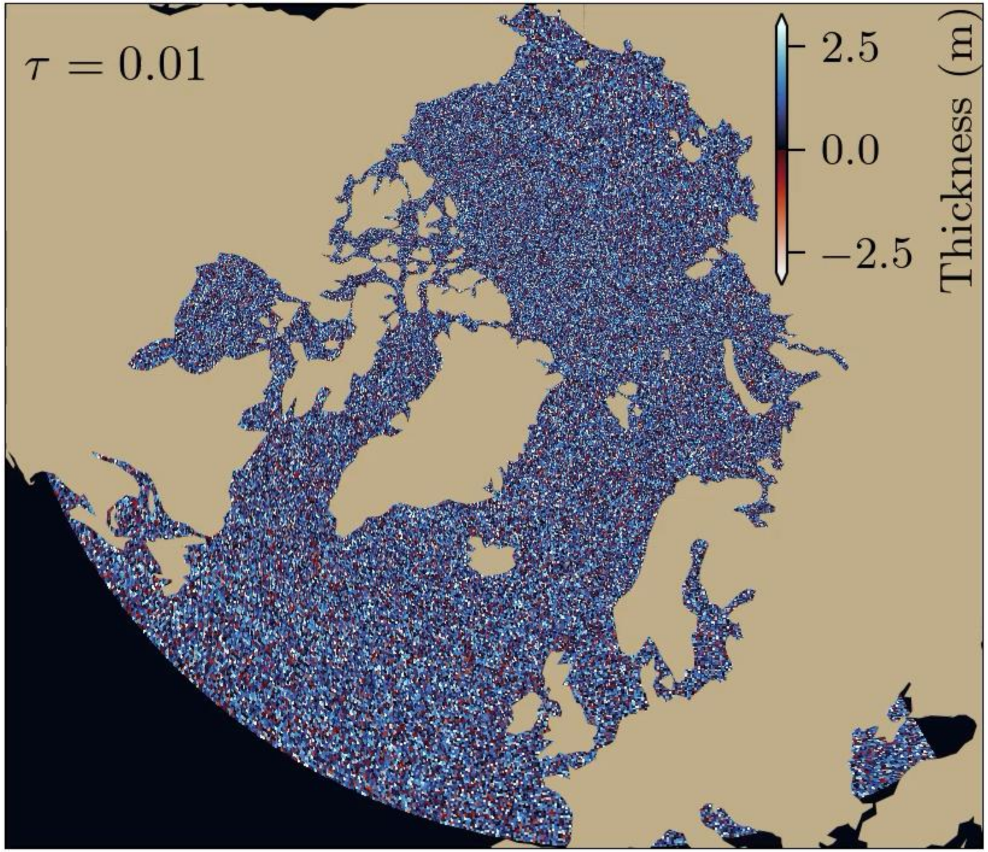
$$\boldsymbol{z}(\tau) = \tau\boldsymbol{x} + (1-\tau)\boldsymbol{\epsilon}$$

Destroy information



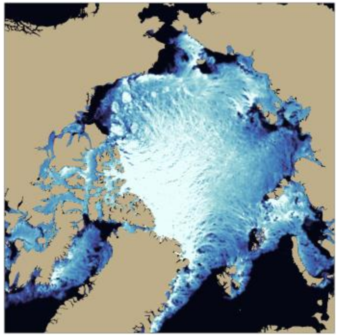
$$z(\tau) = \tau x + (1 - \tau)\epsilon$$

Learn to recover/denoise



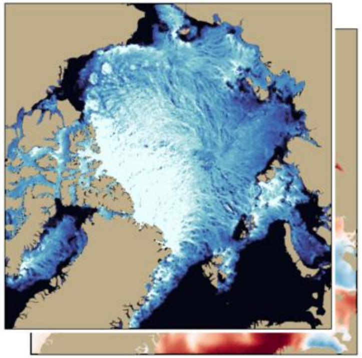
How to train a generative forecasting model?

Target $\boldsymbol{x_{t+\Delta t}}$



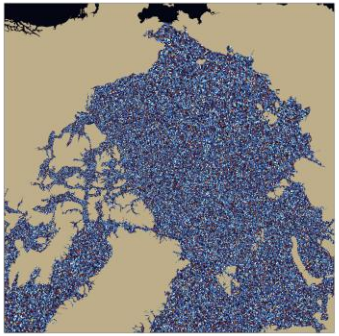
Initial conditions $\boldsymbol{x_t}$

\+ *external forcings*

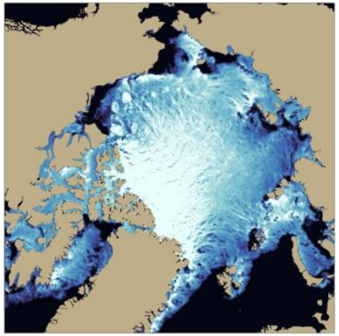


How to train a generative forecasting model?

$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

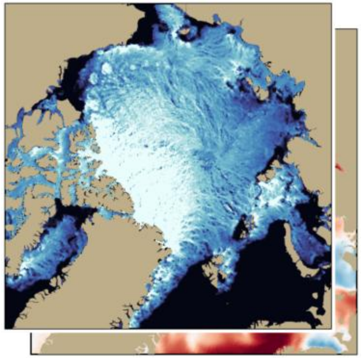


Target $\boldsymbol{x}_{\boldsymbol{t}+\Delta \boldsymbol{t}}$



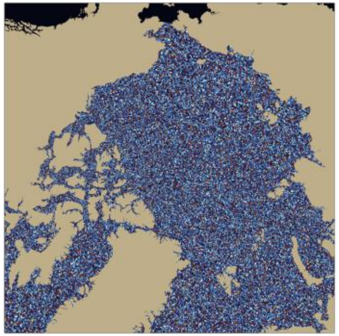
Initial conditions $\boldsymbol{x}_{\boldsymbol{t}}$

\+ *external forcings*



How to train a generative forecasting model?

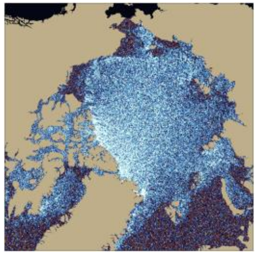
$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$



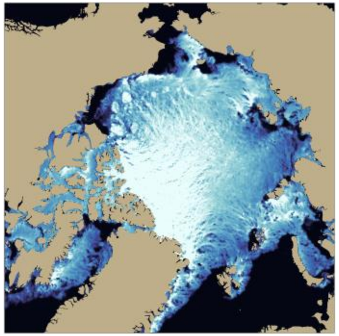
Interpolant

e.g., $\boldsymbol{z}(s) = s\boldsymbol{x}_{\boldsymbol{t}+\Delta \boldsymbol{t}} + (1 - s)\boldsymbol{\epsilon}$

$s \in [0, 1]$

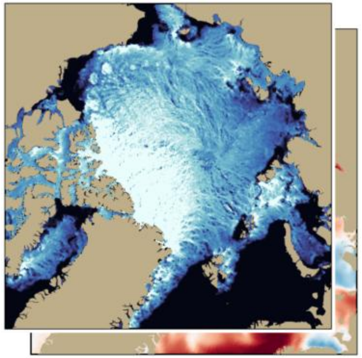


Target $\boldsymbol{x}_{\boldsymbol{t}+\Delta \boldsymbol{t}}$



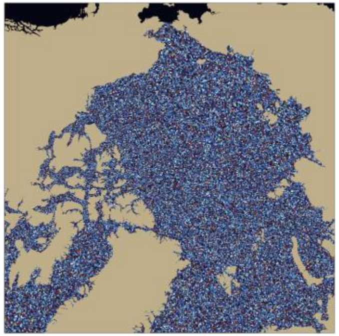
Initial conditions $\boldsymbol{x}_{\boldsymbol{t}}$

+ *external forcings*



How to train a generative forecasting model?

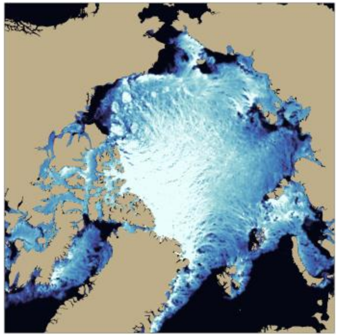
$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$



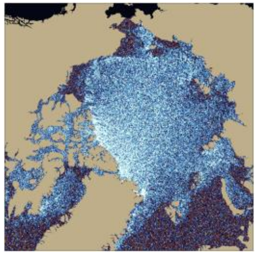
Interpolant

e.g., $\boldsymbol{z}(s) = s\boldsymbol{x}_{t+\Delta t} + (1-s)\boldsymbol{\epsilon}$

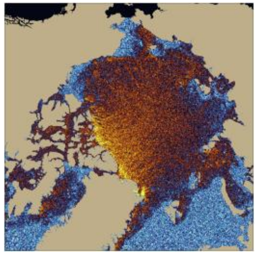
Target $\boldsymbol{x}_{t+\Delta t}$



$s \in [0, 1]$

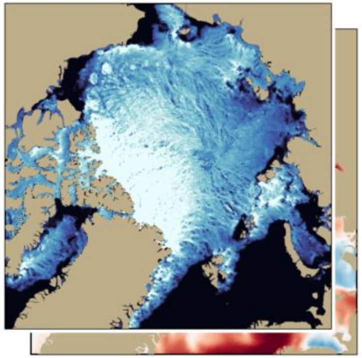


Velocity: $\frac{\partial \boldsymbol{z}(s)}{\partial \boldsymbol{s}} = \boldsymbol{x}_{t+\Delta t} - \boldsymbol{\epsilon}$

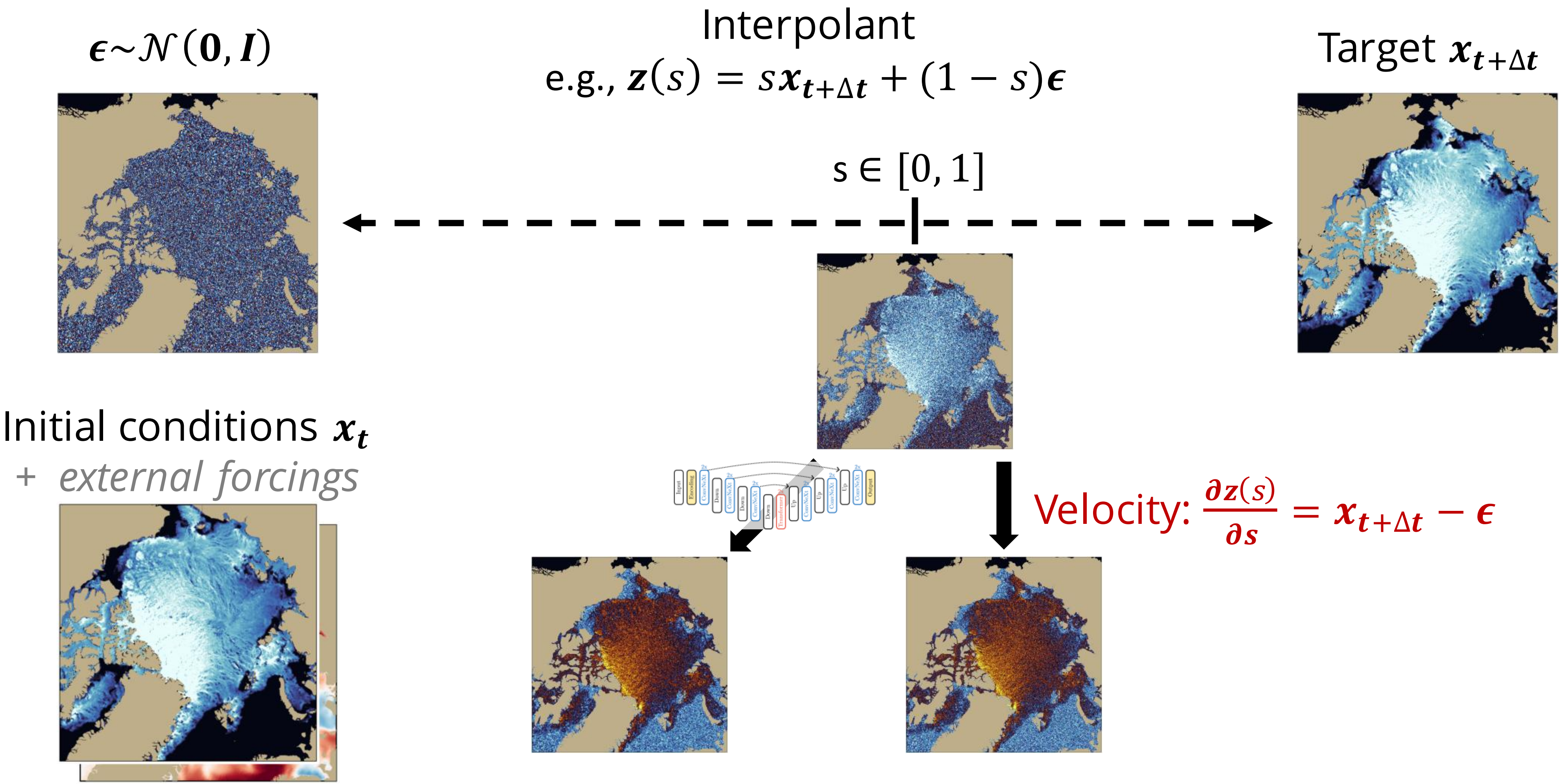


Initial conditions $\boldsymbol{x}_t$

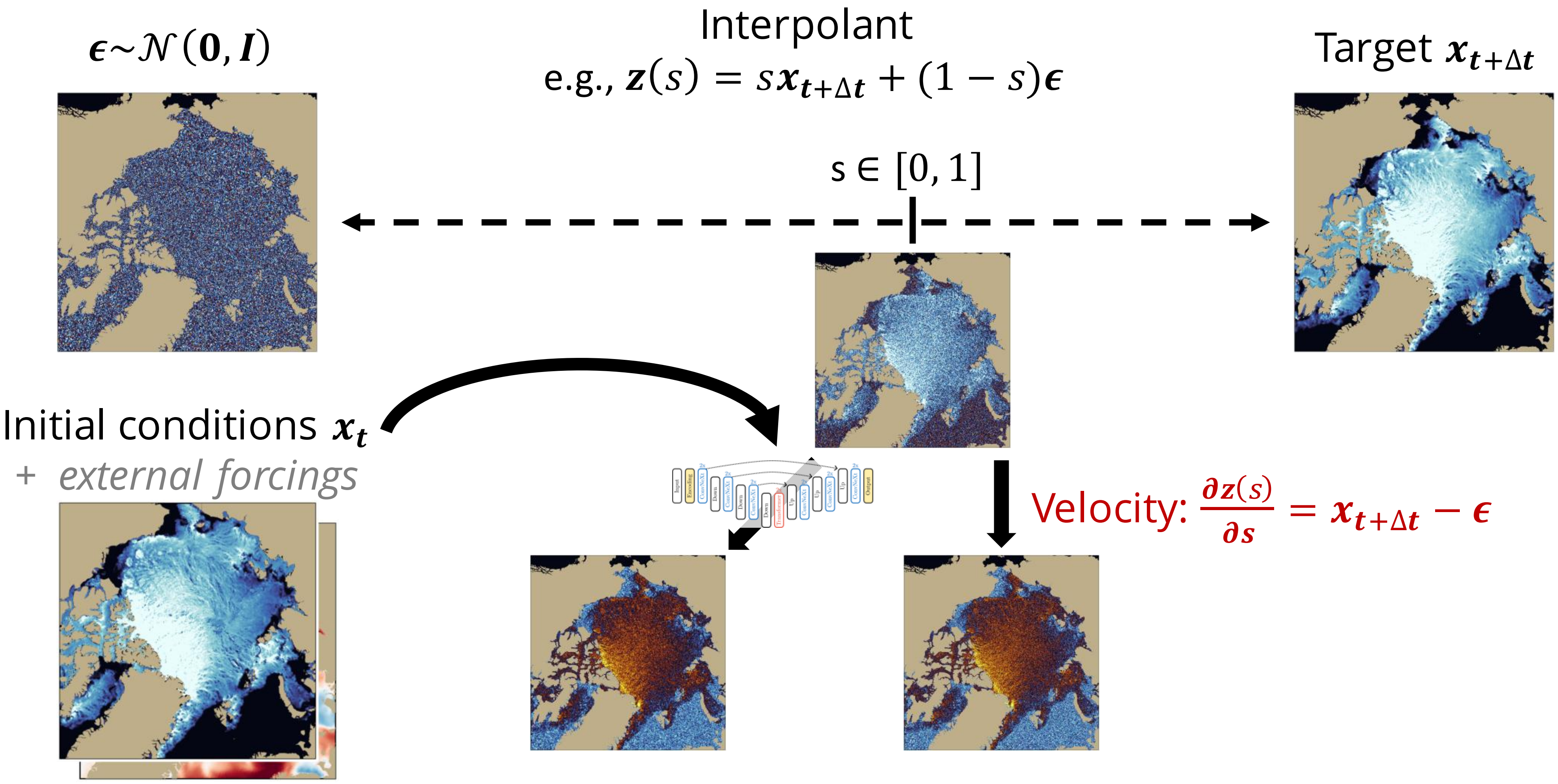
\+ *external forcings*



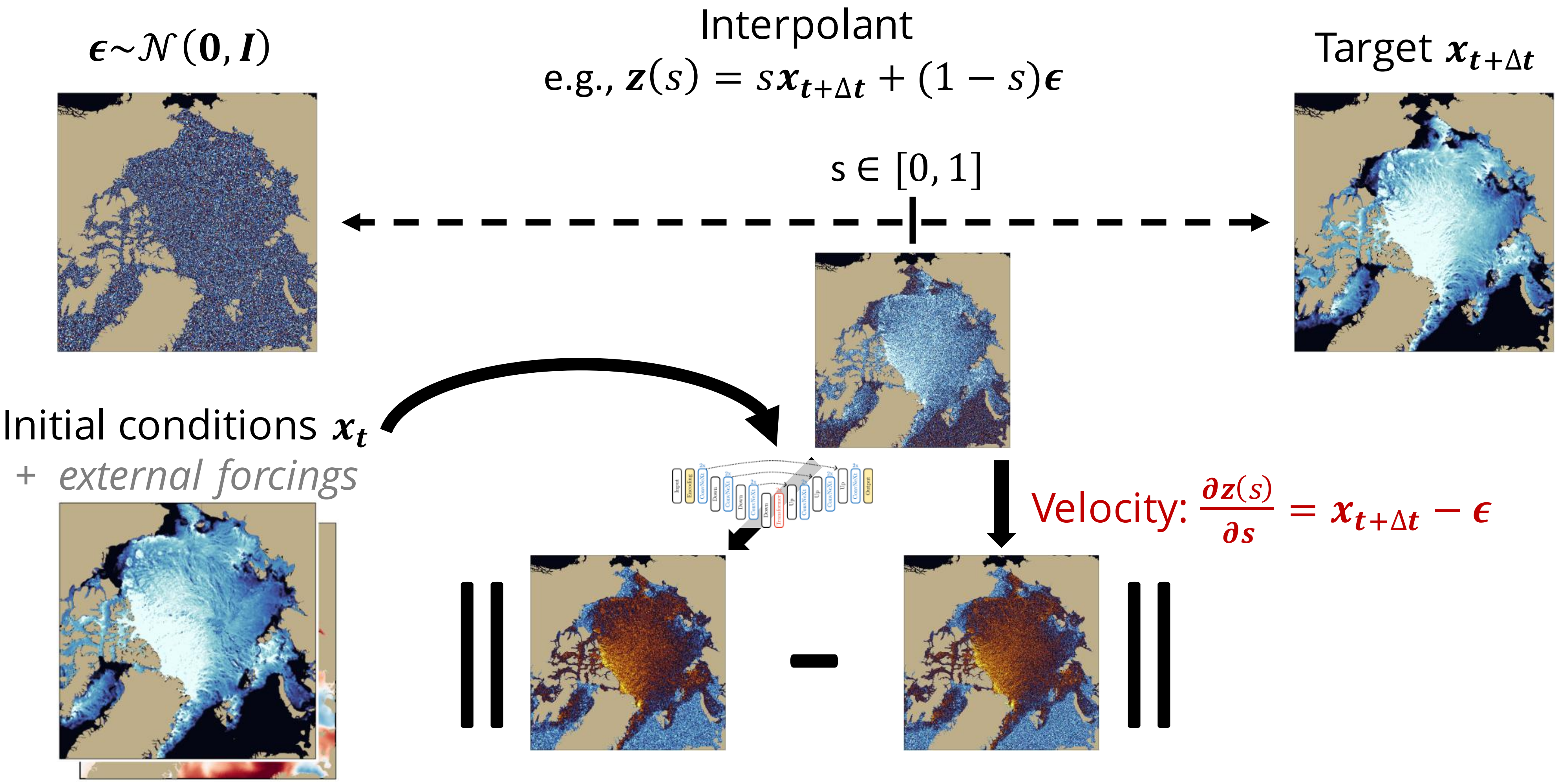
How to train a generative forecasting model?



How to train a generative forecasting model?

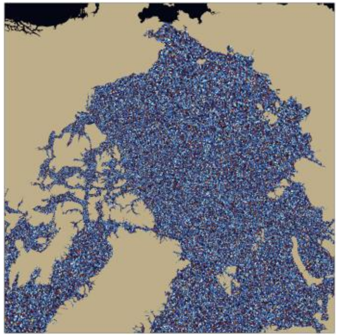


How to train a generative forecasting model?



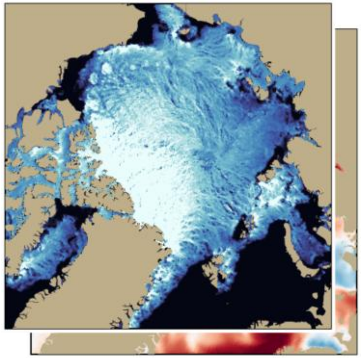
How to forecast with the model?

$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$



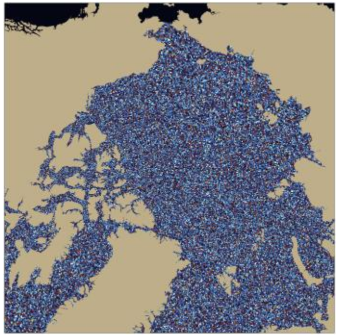
Initial conditions $\boldsymbol{x_t}$

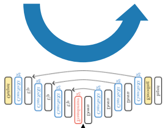
\+ *external forcings*



How to forecast with the model?

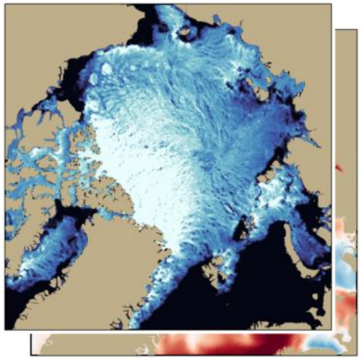
$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$



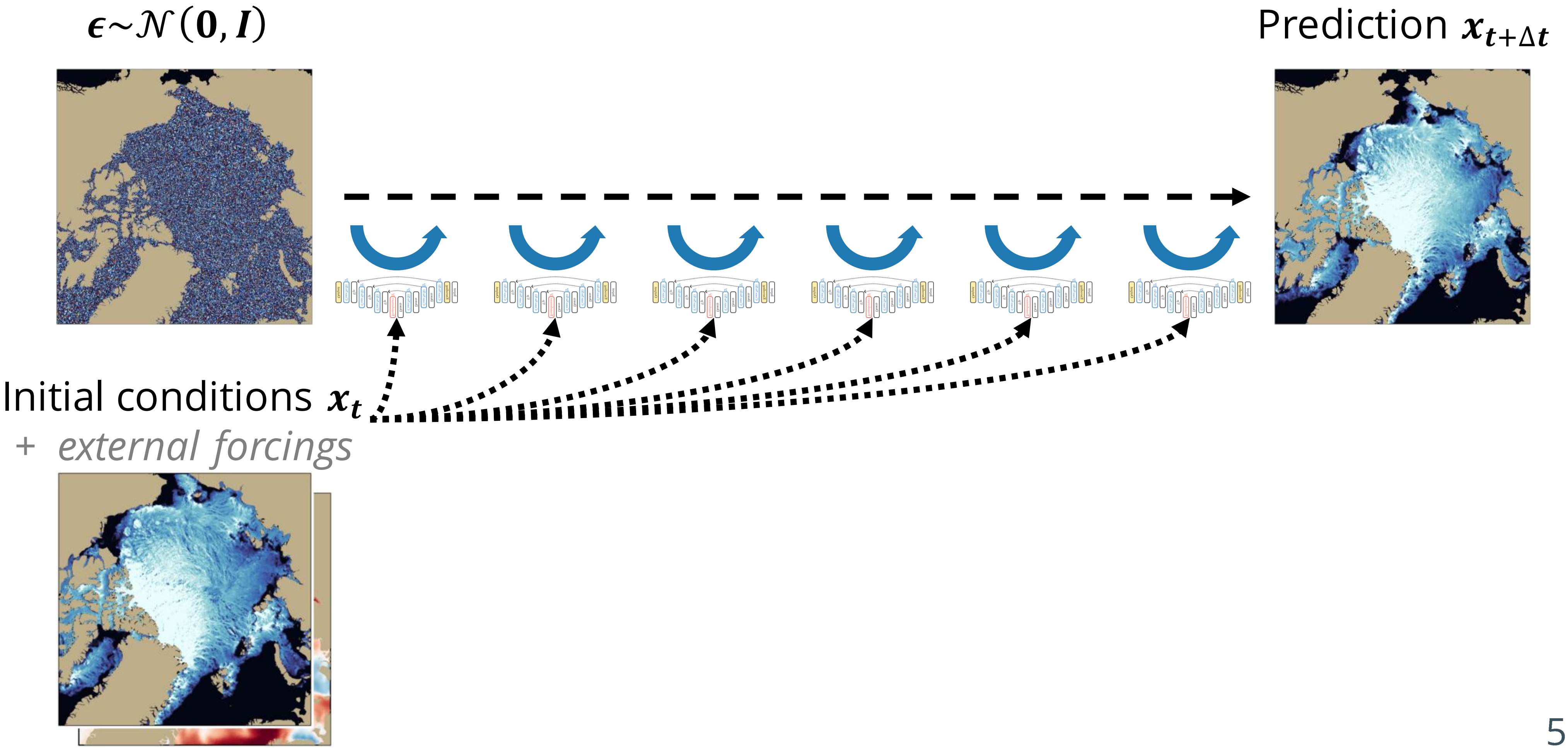


Initial conditions $\boldsymbol{x_t}$

+ *external forcings*

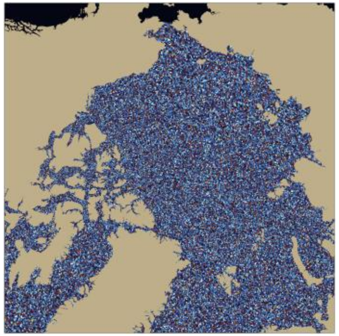


How to forecast with the model?



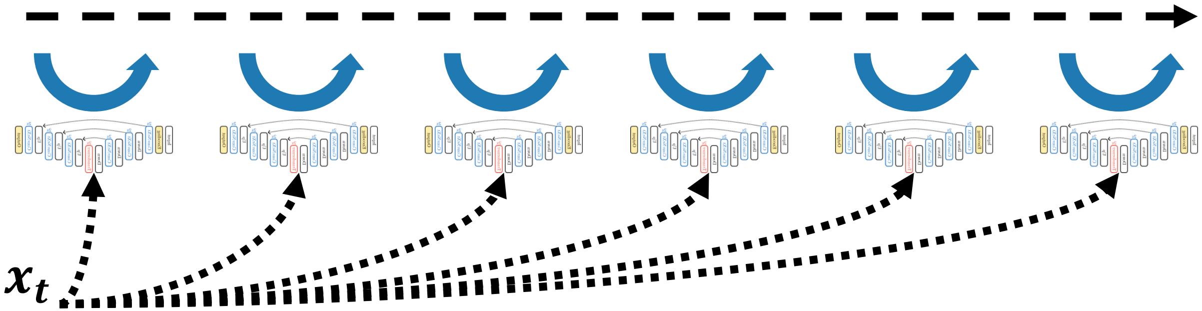
How to forecast with the model?

$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

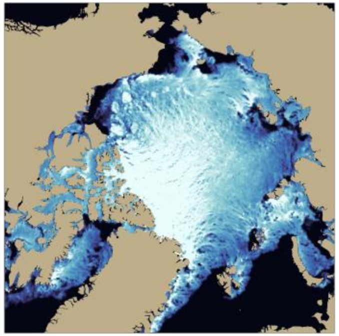


Flow ODE

$$\boldsymbol{x}_{t+\Delta t} = \boldsymbol{\epsilon} + \int_0^1 \widehat{\boldsymbol{V}}(\boldsymbol{z}(s), \boldsymbol{x}_t, s)\, ds$$

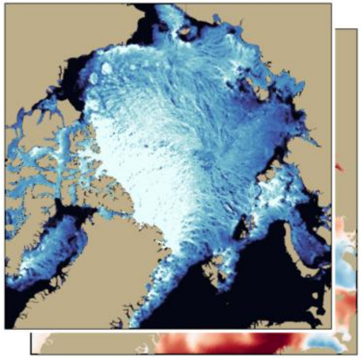


Prediction $\boldsymbol{x}_{t+\Delta t}$

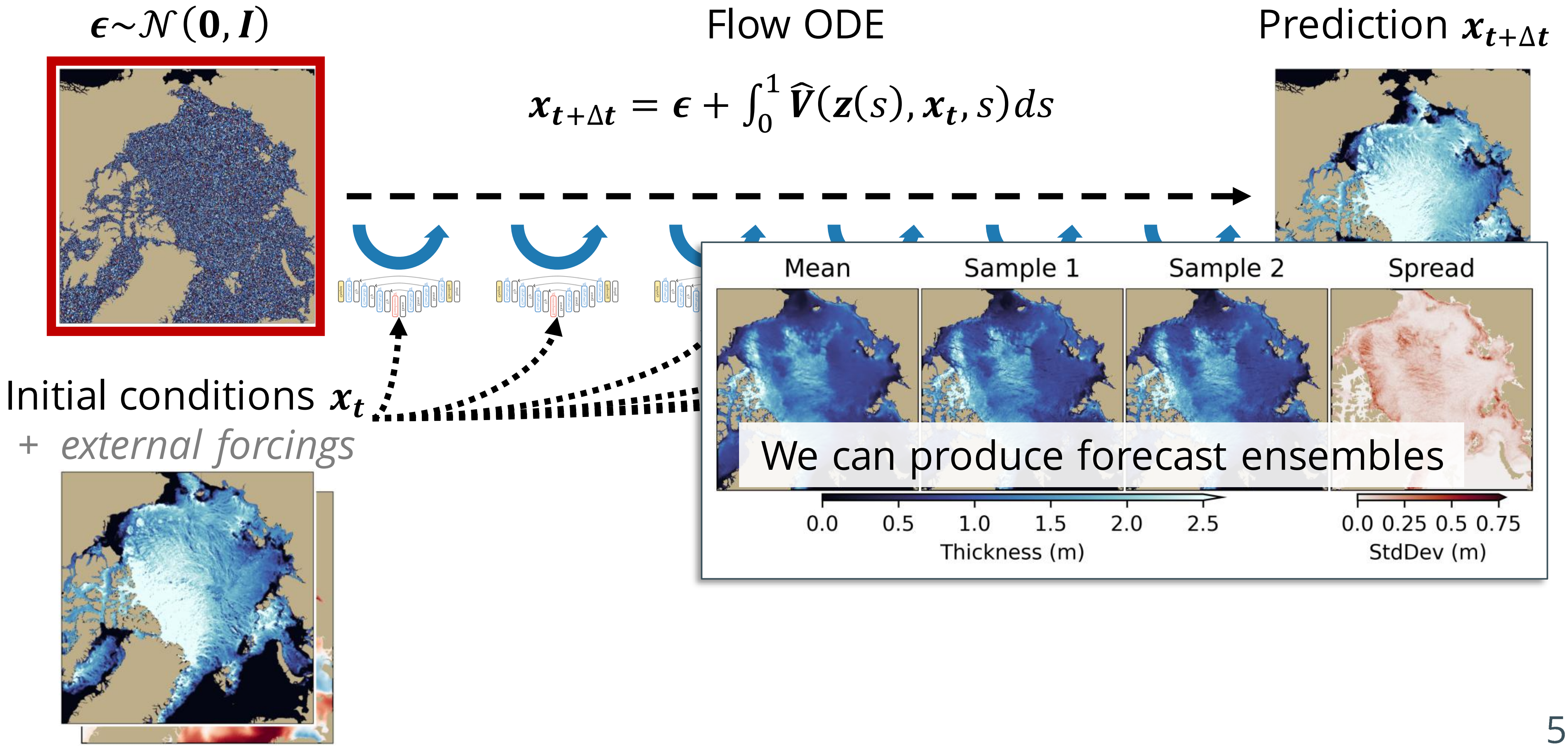


Initial conditions $\boldsymbol{x}_t$

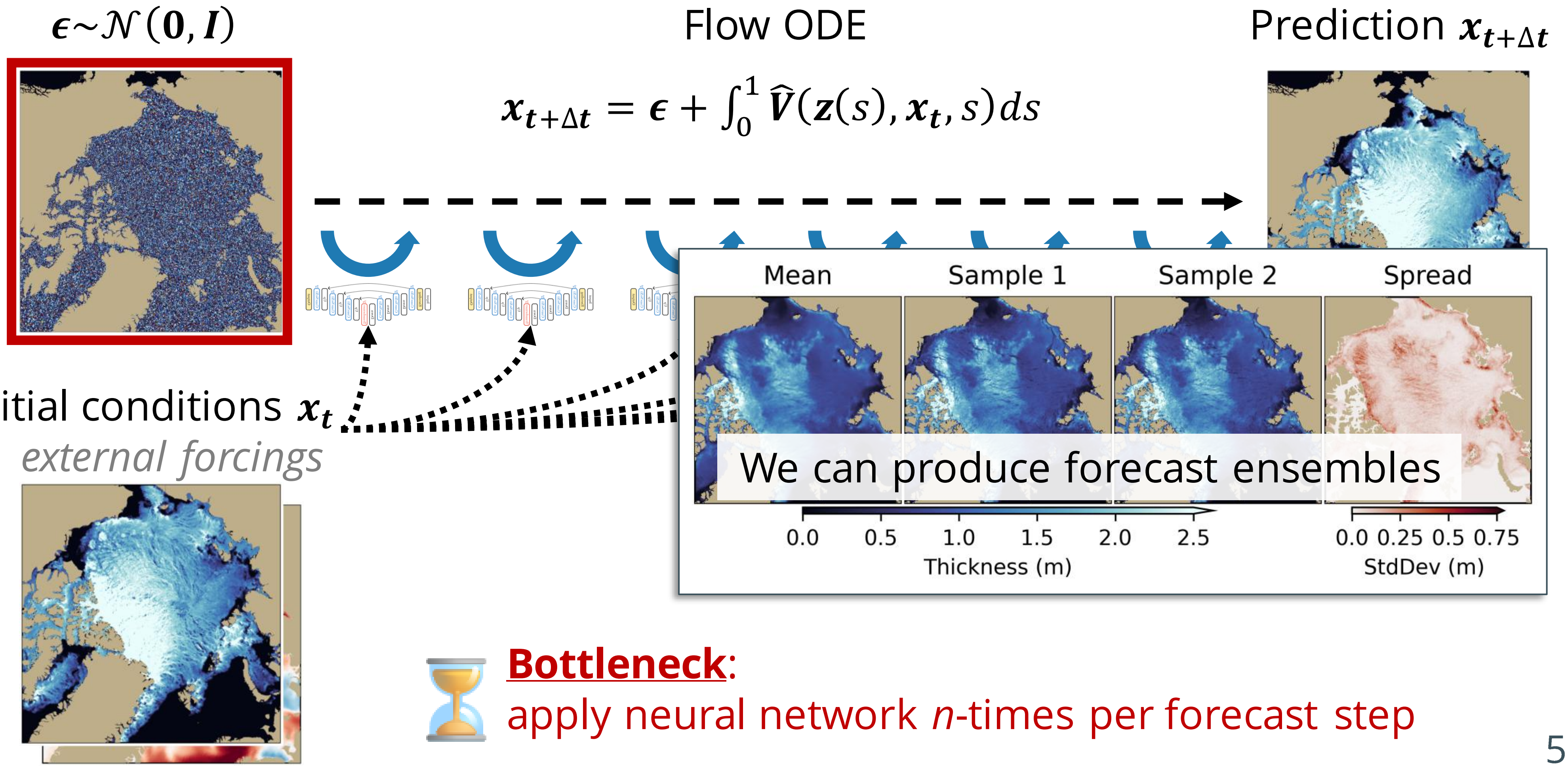
\+ *external forcings*



How to forecast with the model?

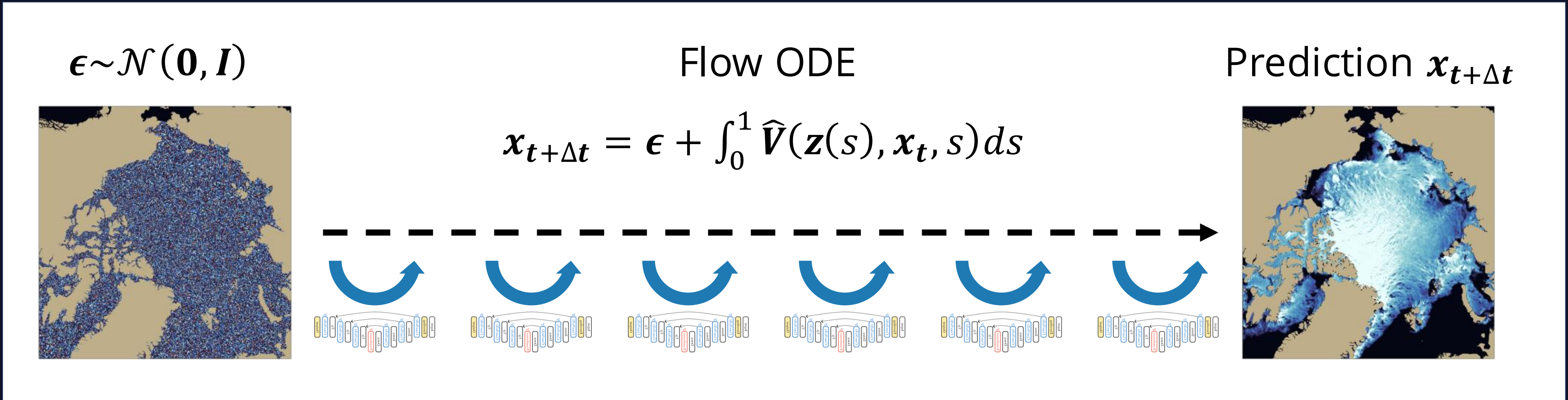


How to forecast with the model?



⌛ **Bottleneck**:
apply neural network *n*-times per forecast step

Generative flows work like numerical models

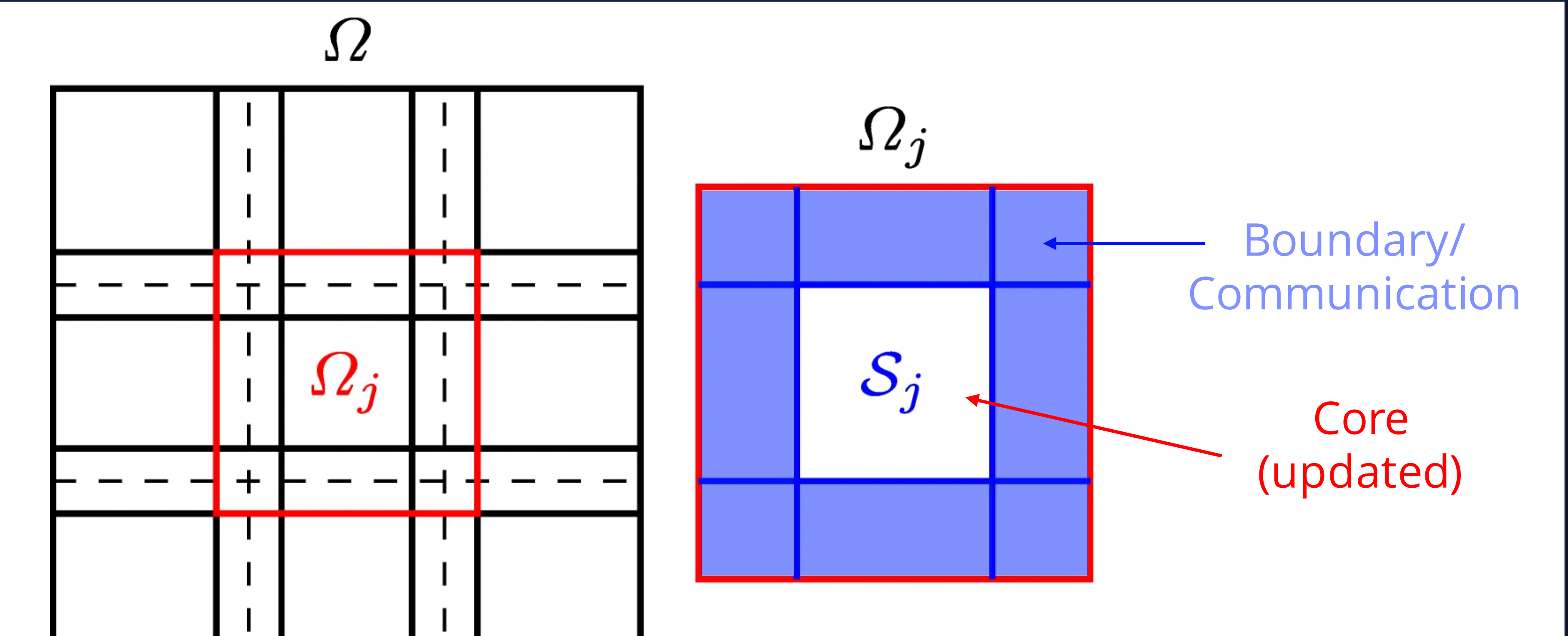


Integration of differential equation in pseudo time

Prior physical knowledge: <u>sensitivity for short term is localised</u>

Domain decomposition comes to rescue

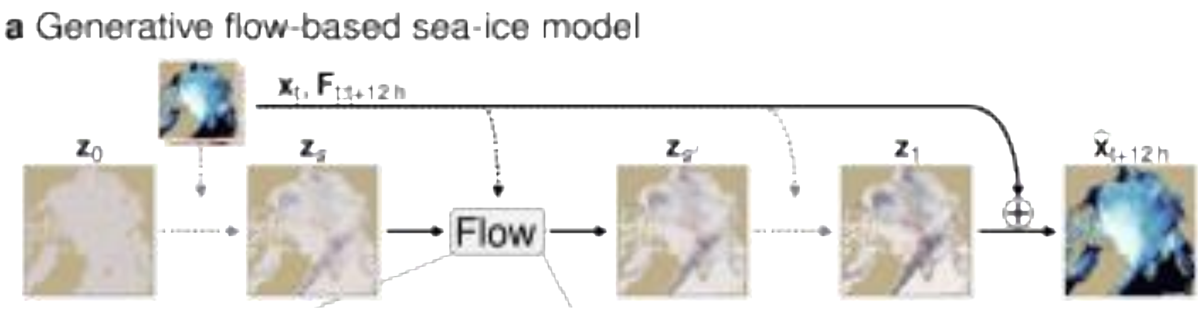
Allows efficient training and inference with different datasets



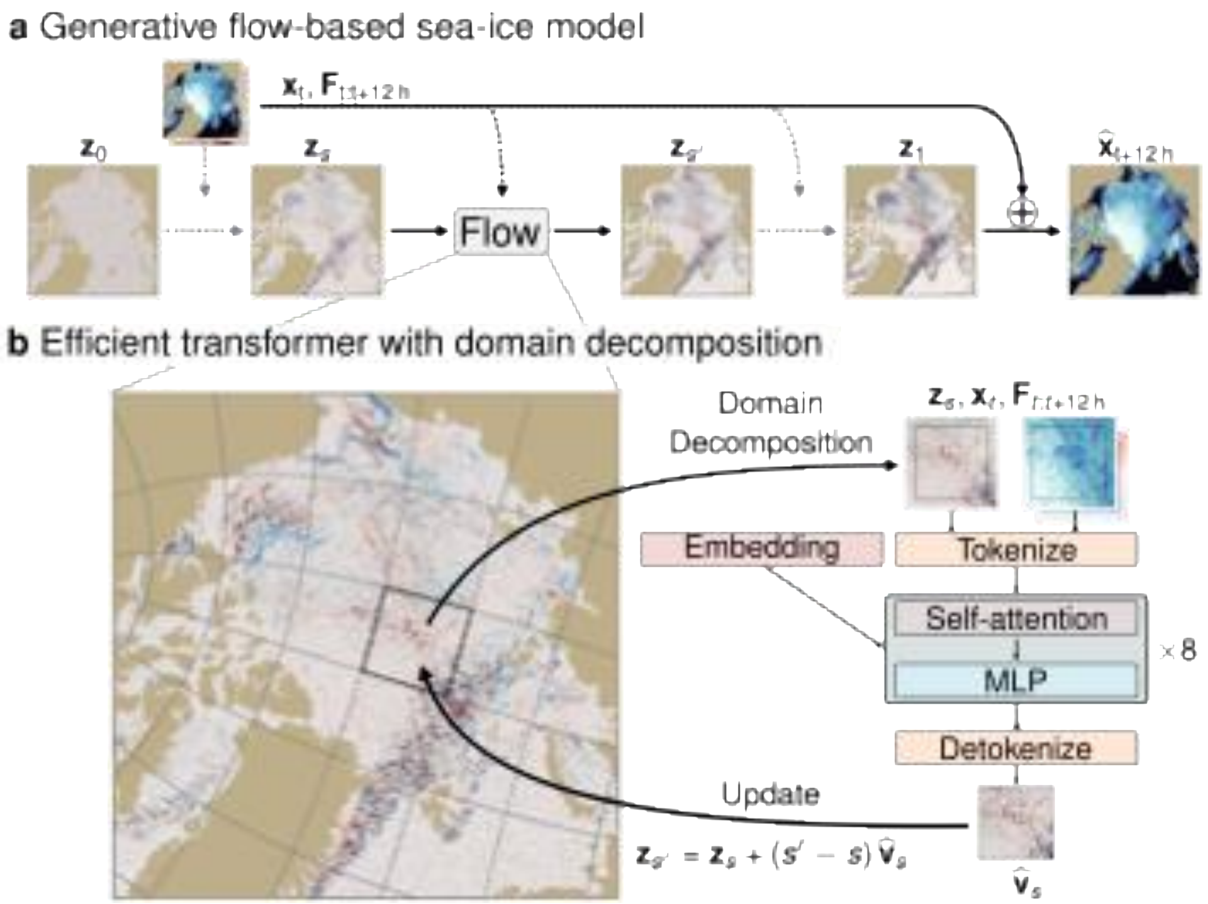
Similarly used by Nvidia (MultiDiffusion)

GenSIM – first efficient generative sea-ice model

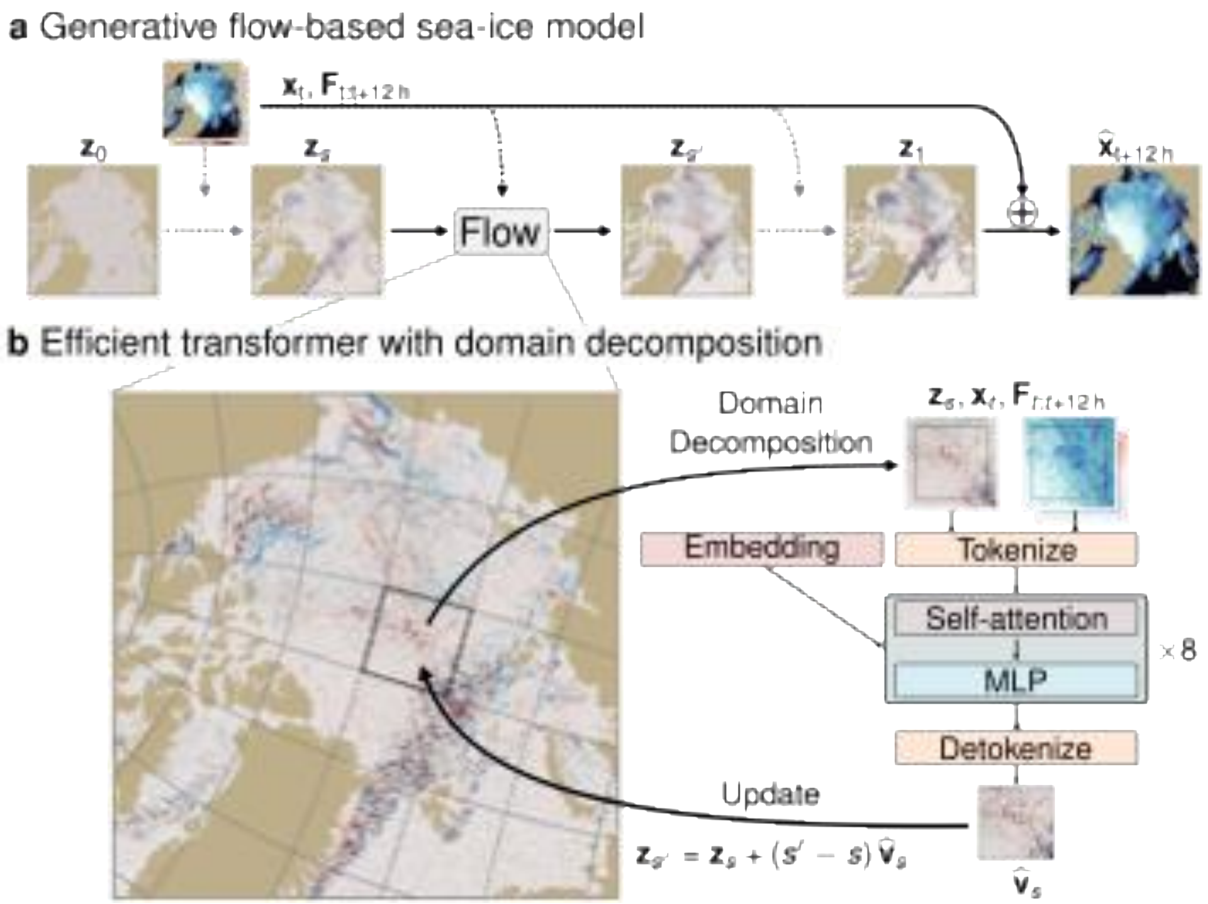
GenSIM – first efficient generative sea-ice model



GenSIM – first efficient generative sea-ice model

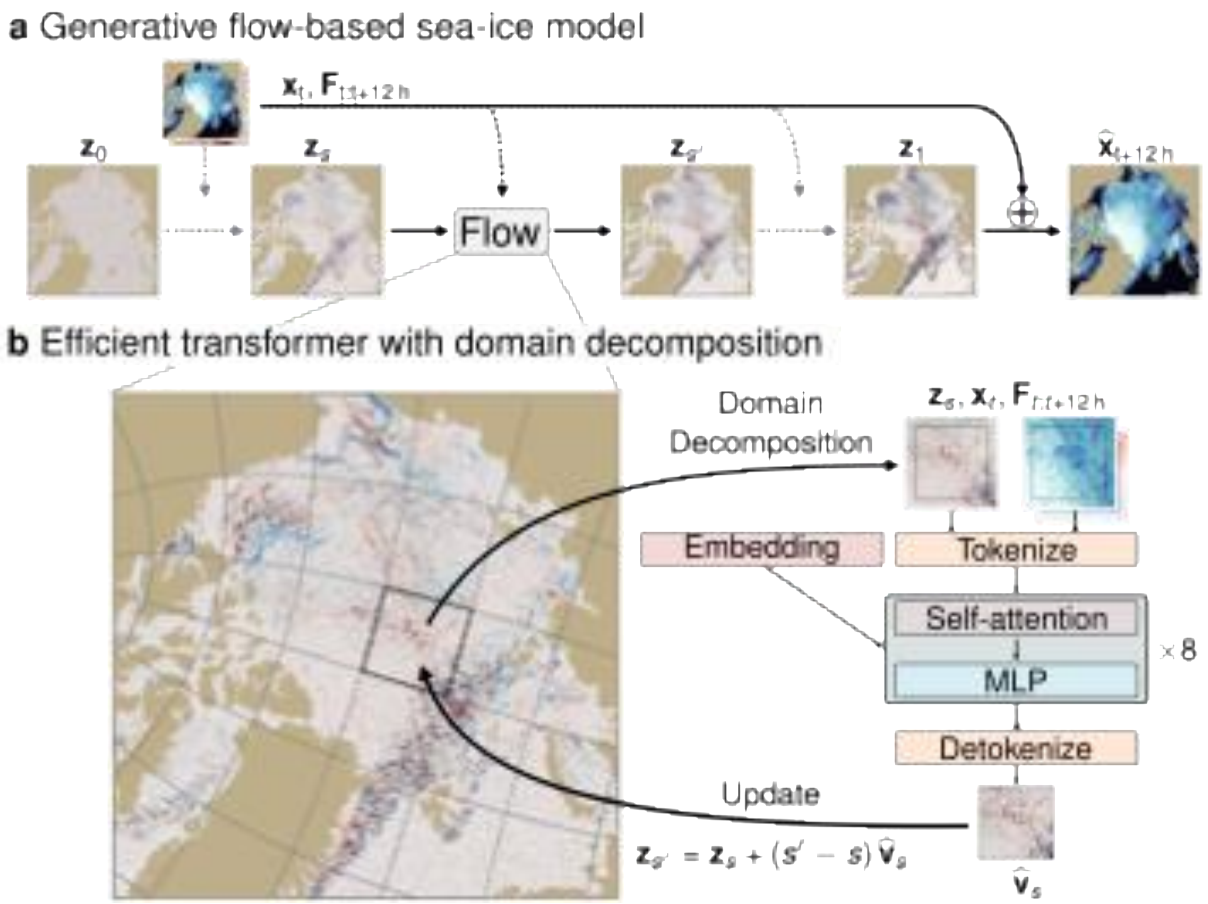


GenSIM – first efficient generative sea-ice model

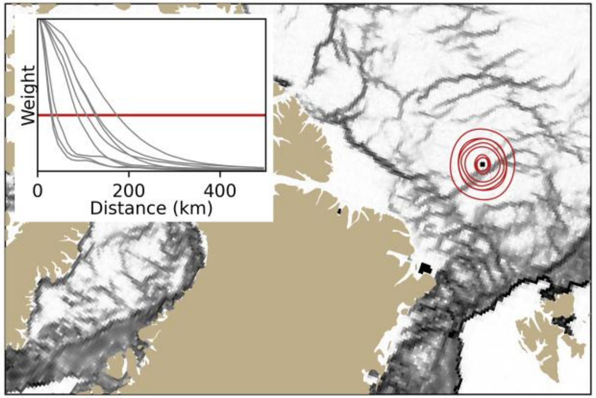


12 h lead time, neXtSIM-OPA (1995-2014), full state, ERA5 forcing

GenSIM – first efficient generative sea-ice model

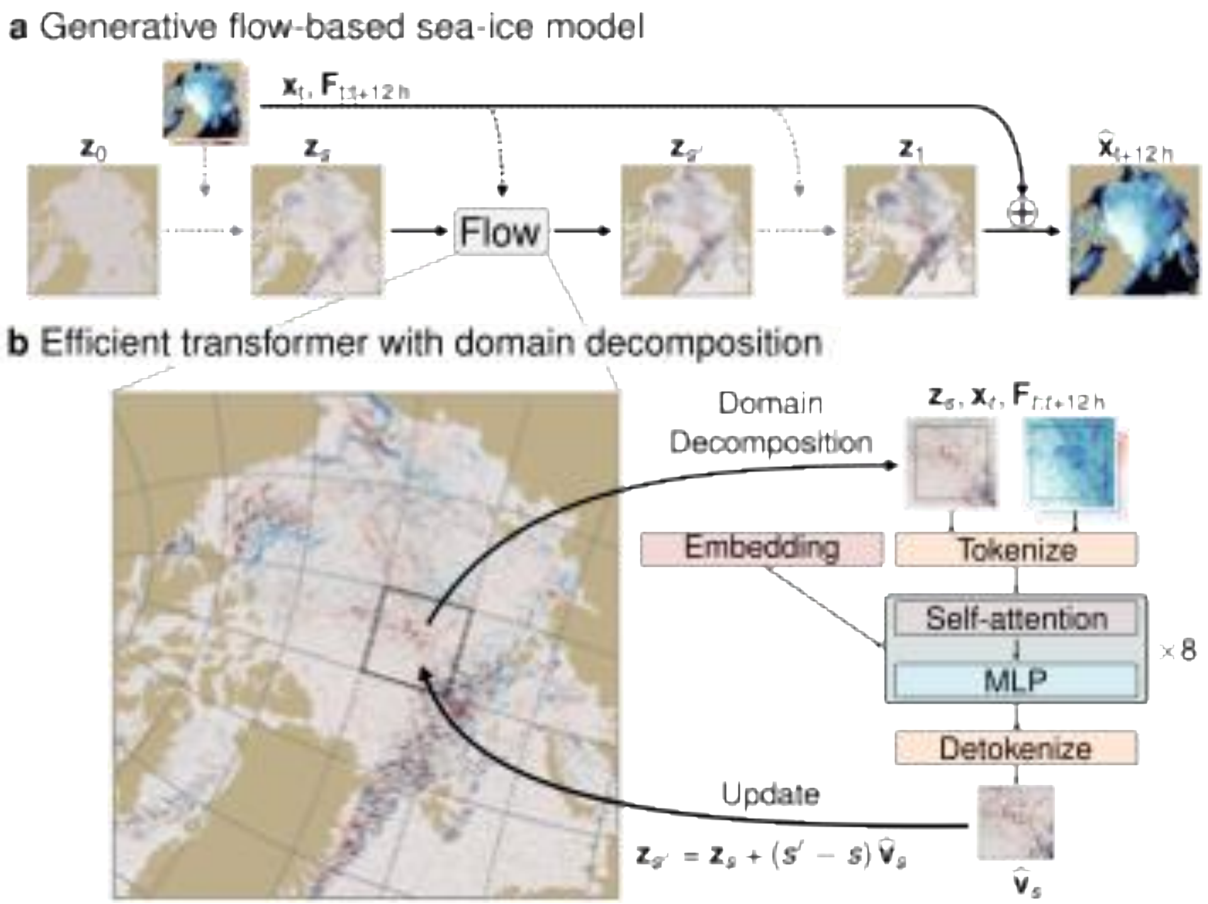


Learned soft localisation

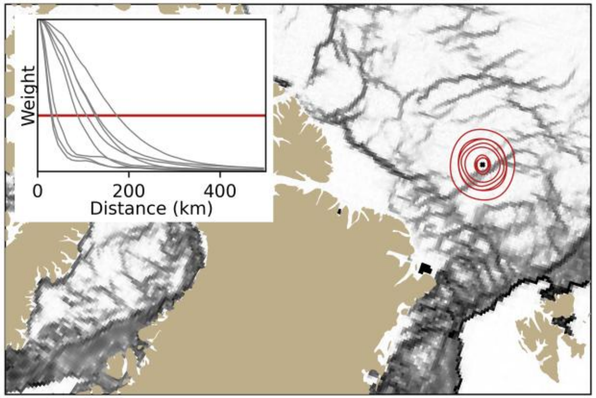


12 h lead time, neXtSIM-OPA (1995-2014), full state, ERA5 forcing

GenSIM – first efficient generative sea-ice model



Learned soft localisation



Resolution-agnostic

Mesh-agnostic

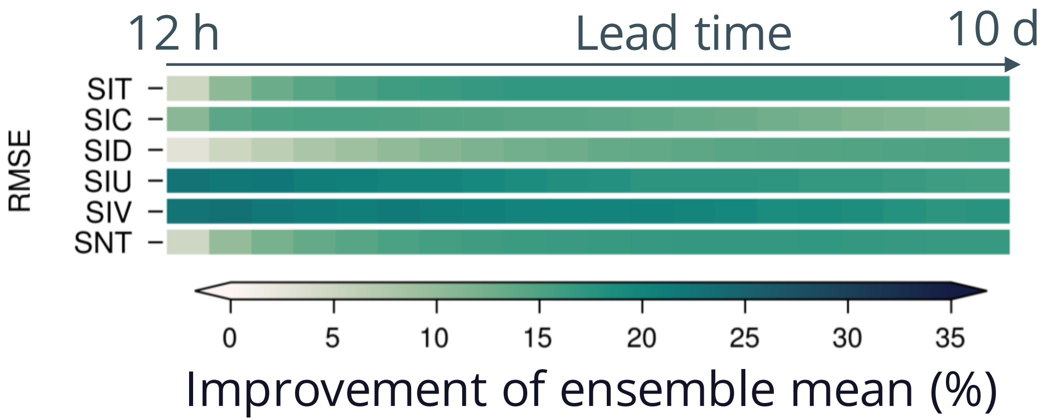
Accounts for bounds

Efficient (< 4 GB & 25 sy/d)

12 h lead time, neXtSIM-OPA (1995-2014), full state, ERA5 forcing

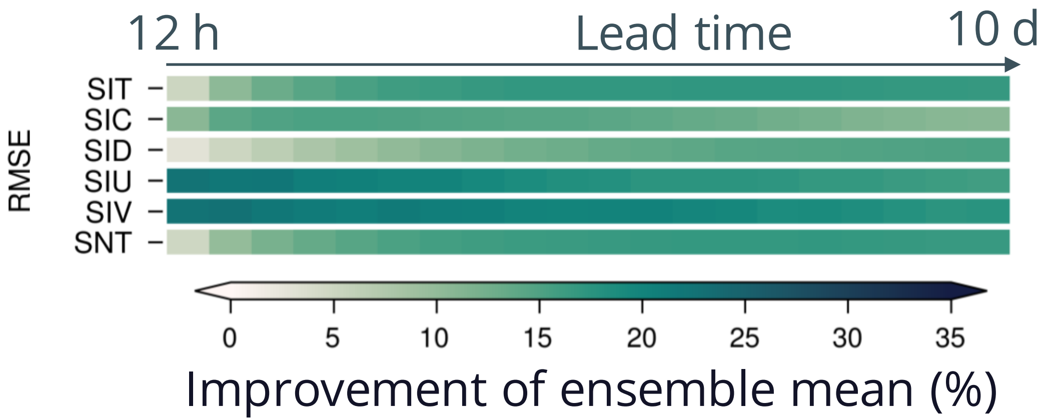
Outperforms deterministic baseline

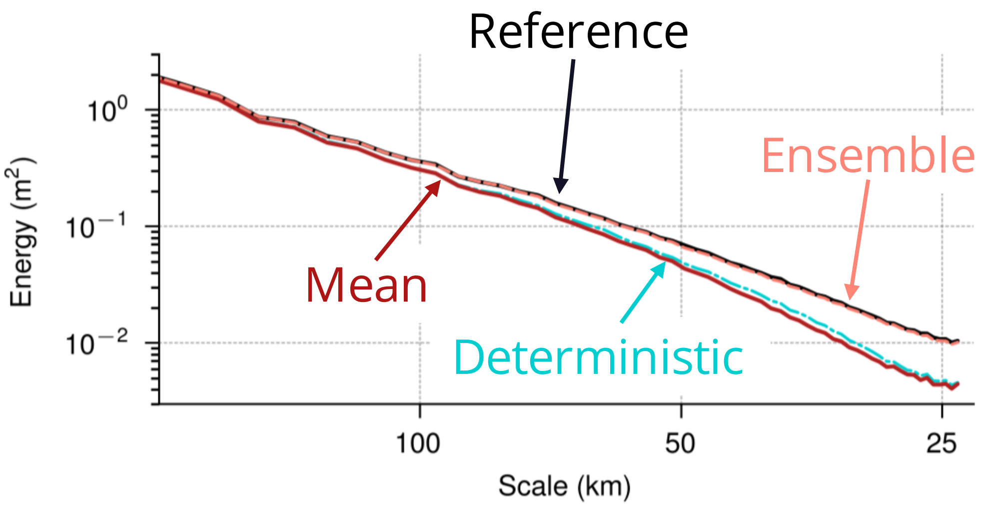
16 members, 2016-2018



Outperforms deterministic baseline

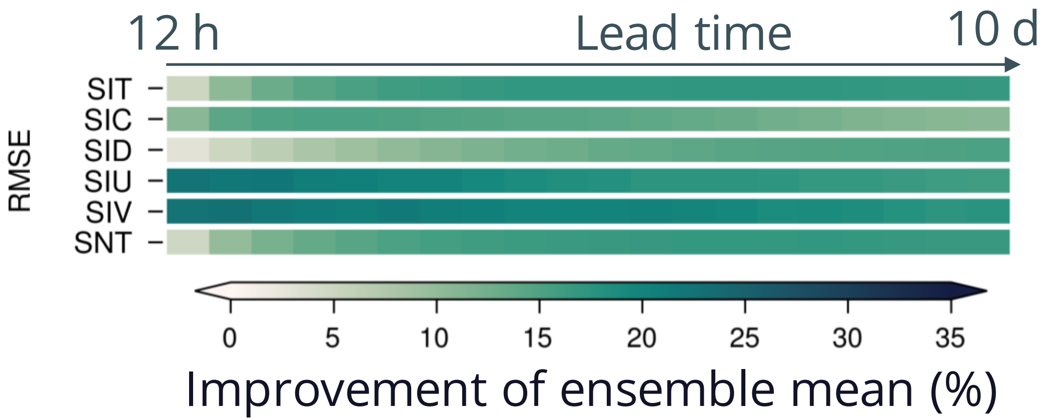
16 members, 2016-2018

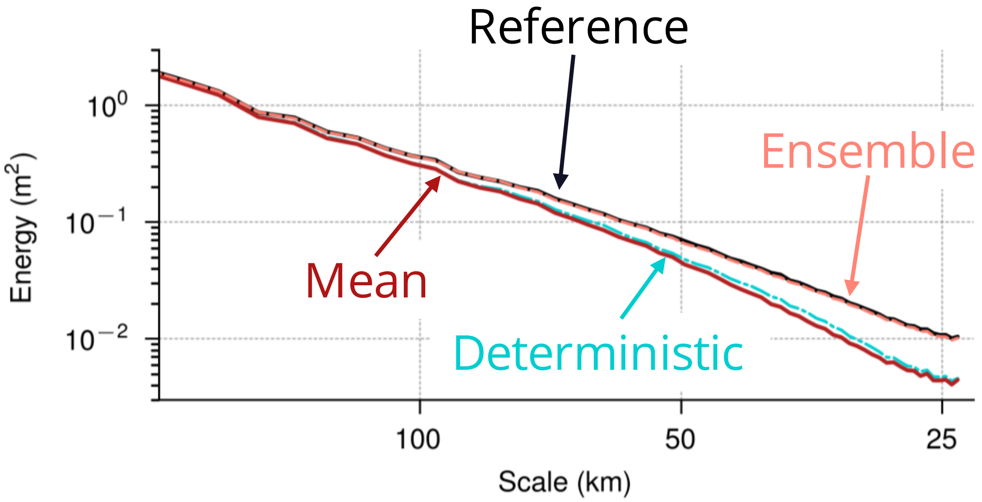


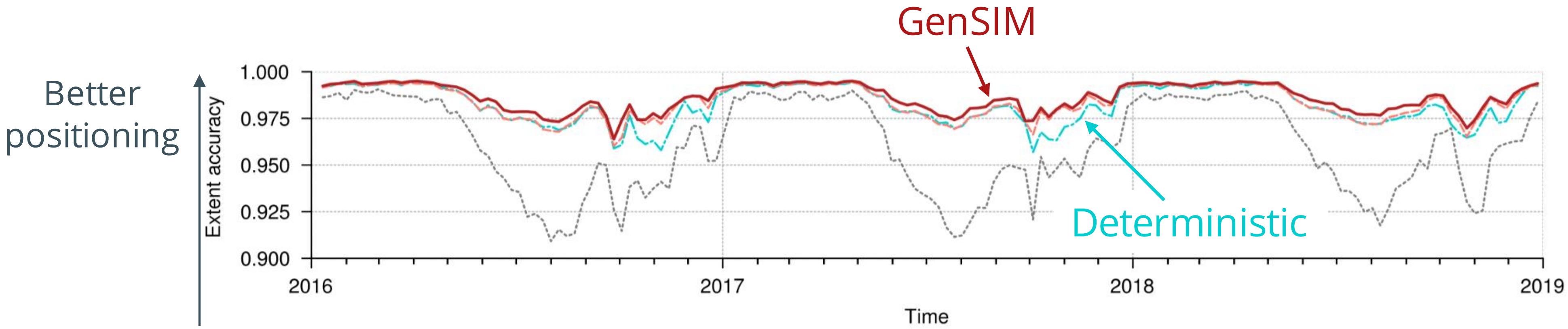


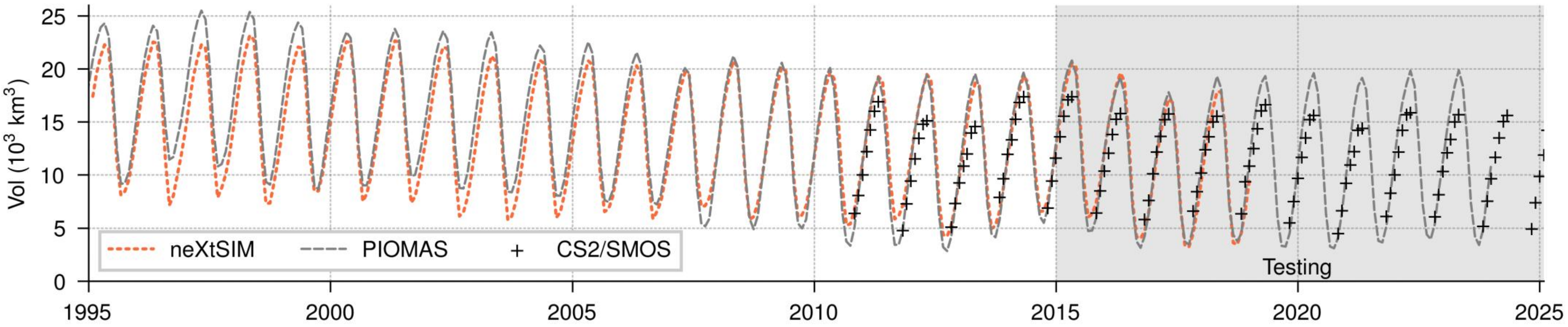
Outperforms deterministic baseline

16 members, 2016-2018



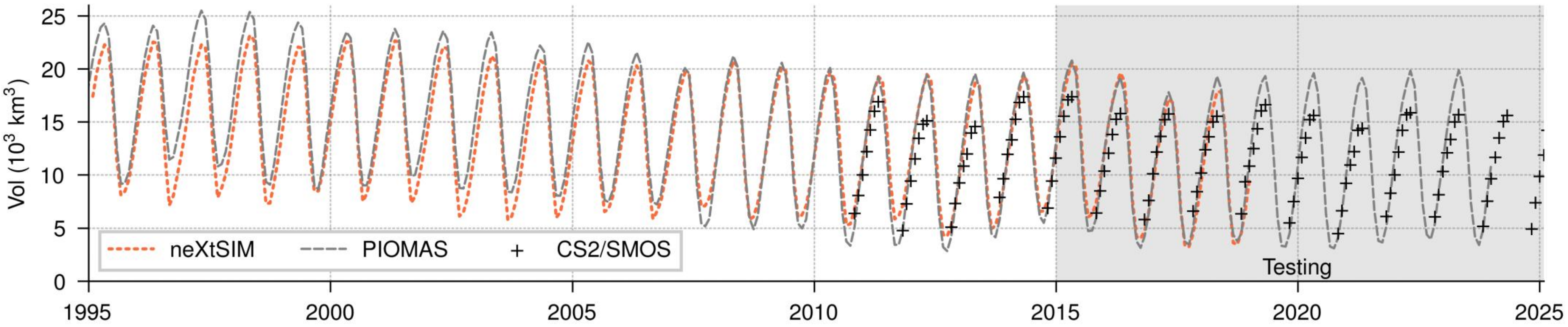






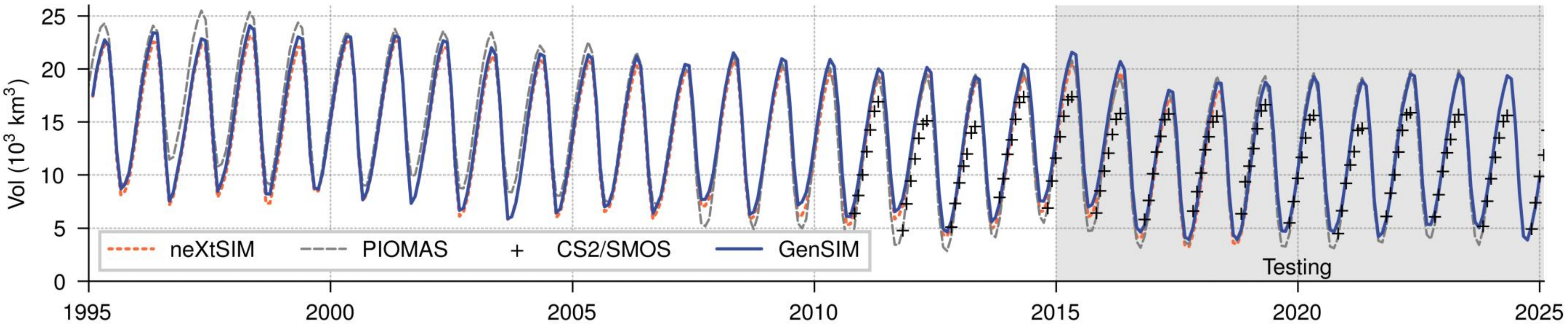
Vol (10^3 km^3)
0
5
10
15
20
25
1995
2000
2005
2010
2015
2020
2025
neXtSIM
PIOMAS
CS2/SMOS
Testing

Remember: only trained for 12 hour forecasts

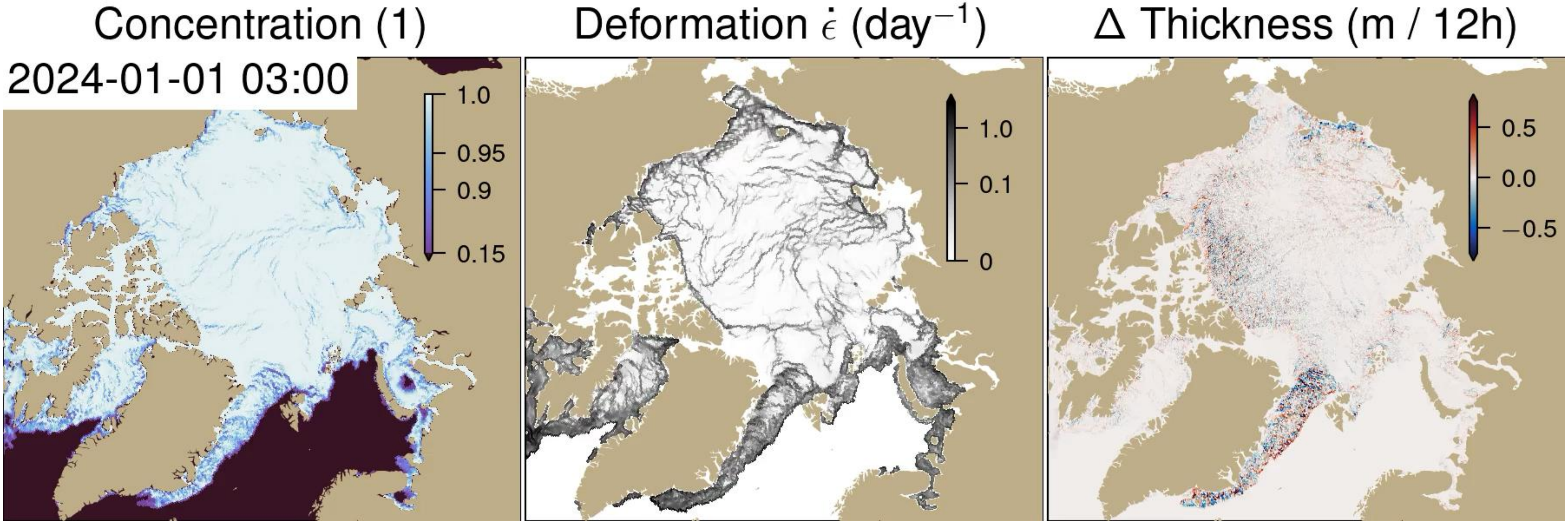


Stable extrapolation to decade-long simulations

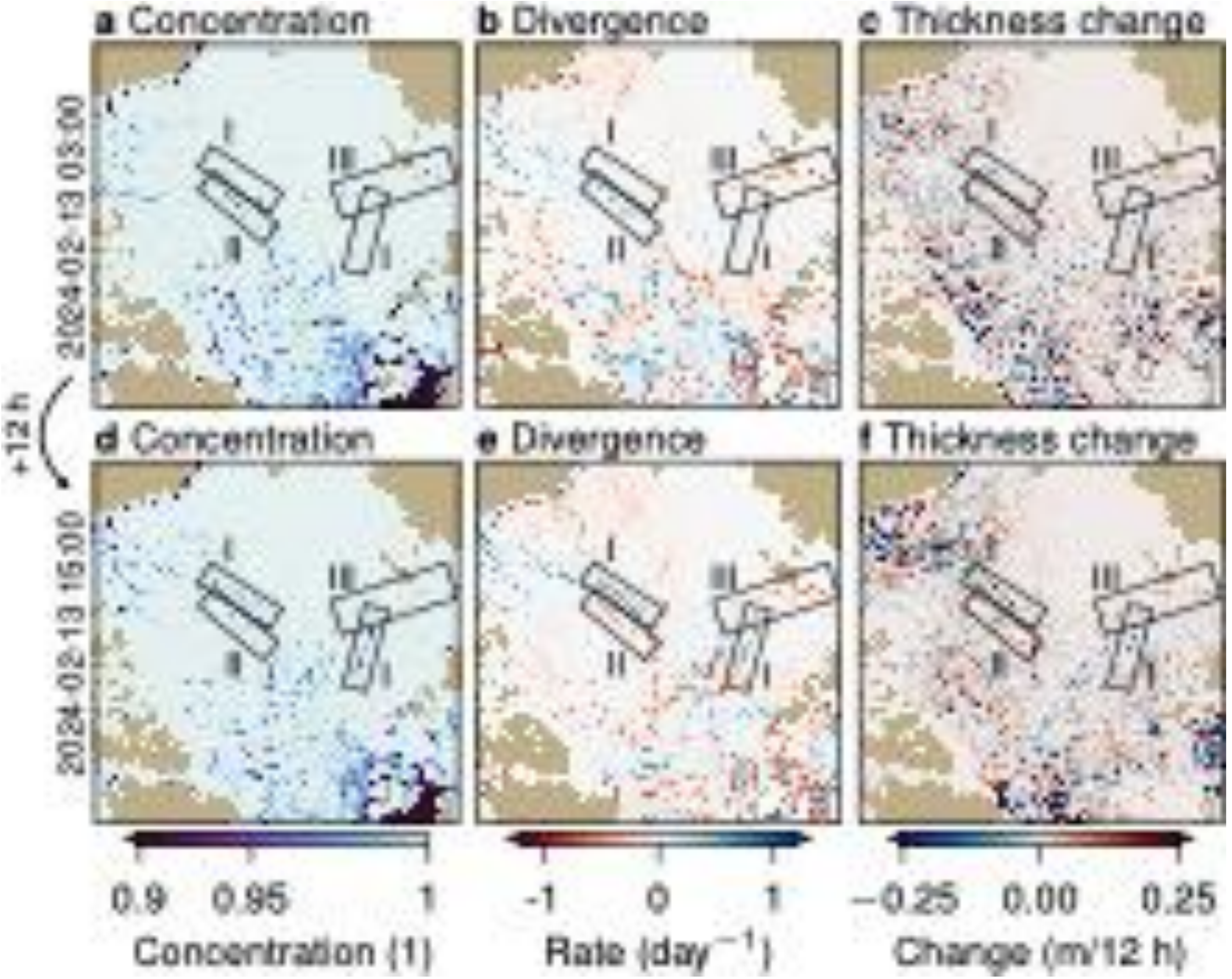
Remember: only trained for 12 hour forecasts



Temporal consistency even after nearly 30 years

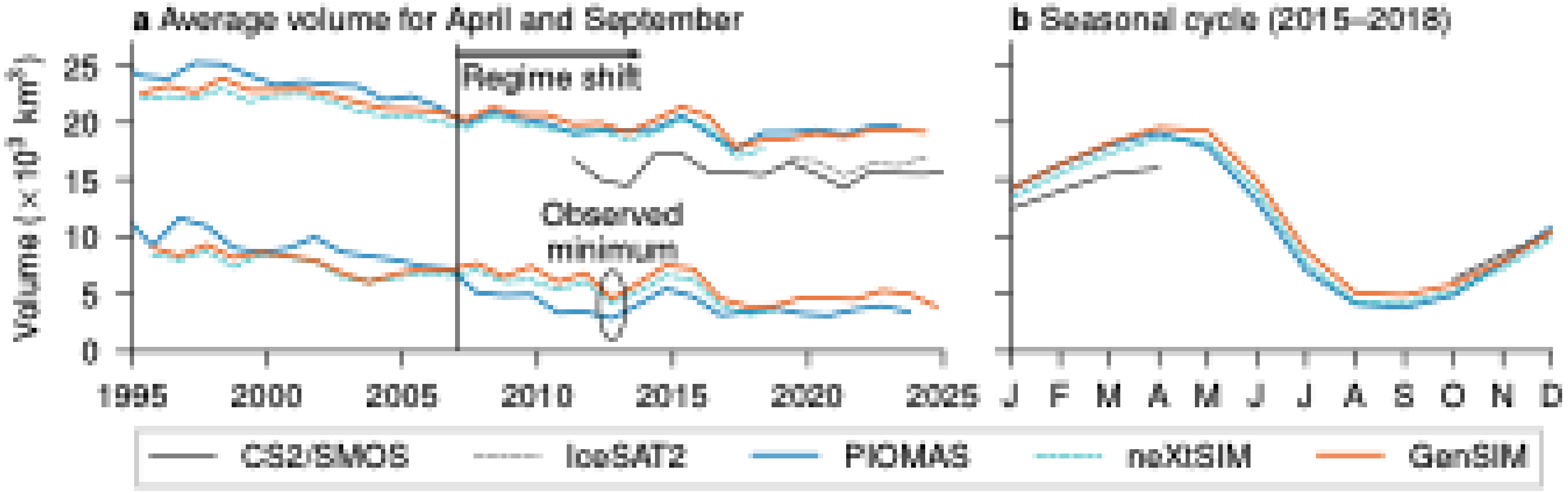


Implicit representation of resolved physical processes

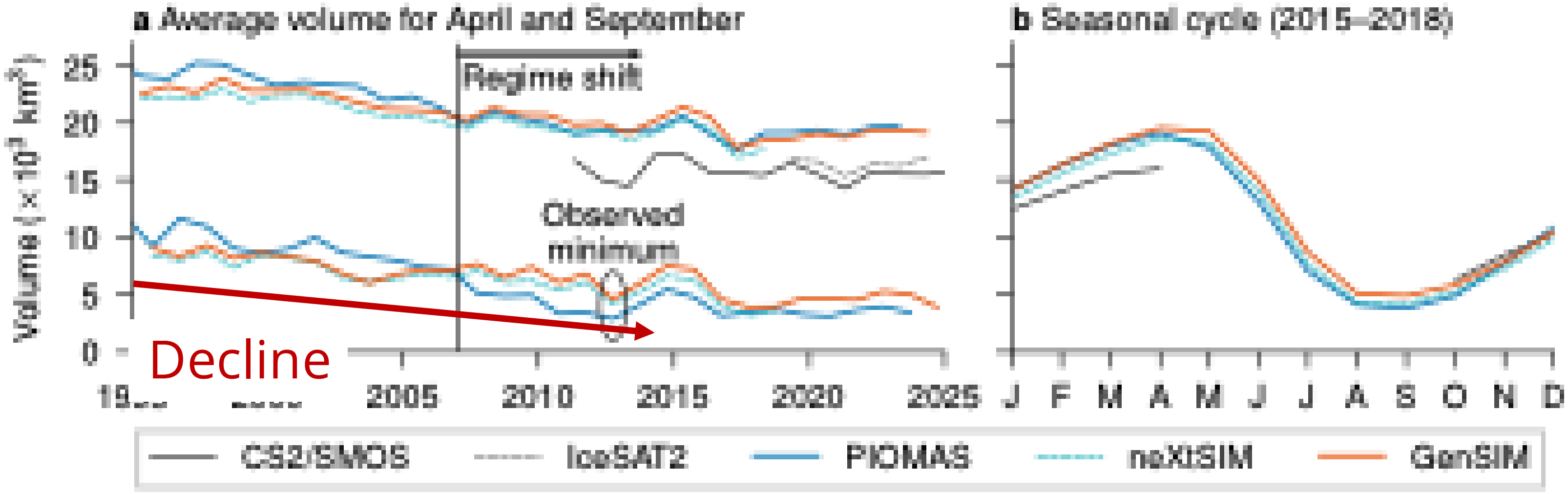


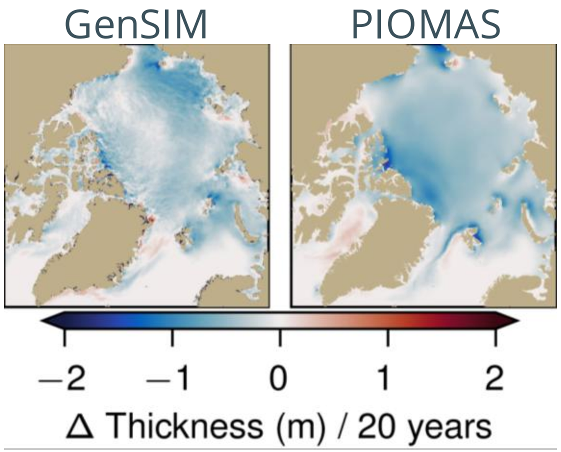
I: Opening of leads
II: Healing
III: Ridging

Correct thermodynamical evolution

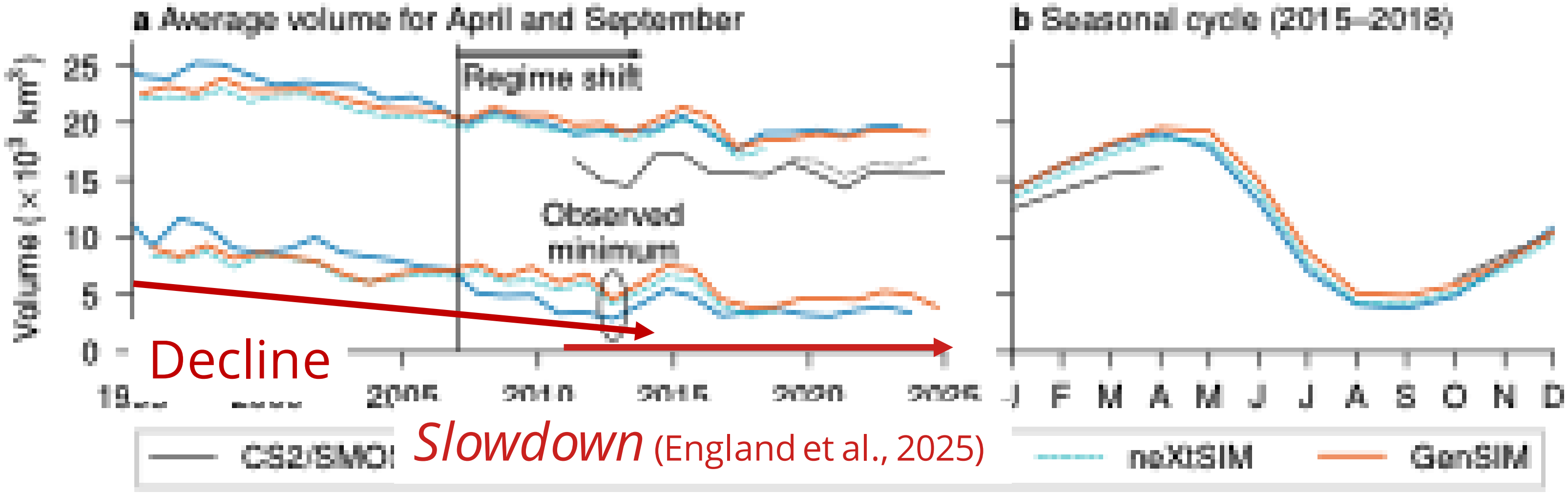


Correct thermodynamical evolution

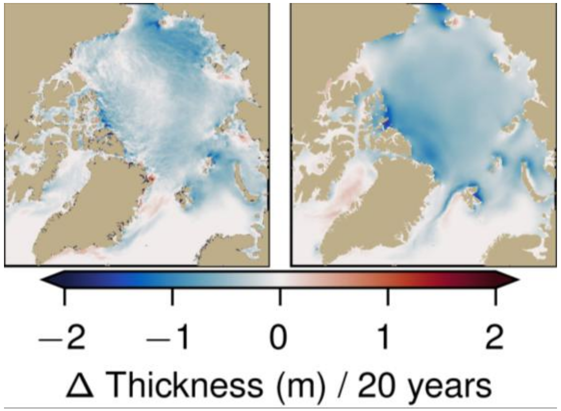


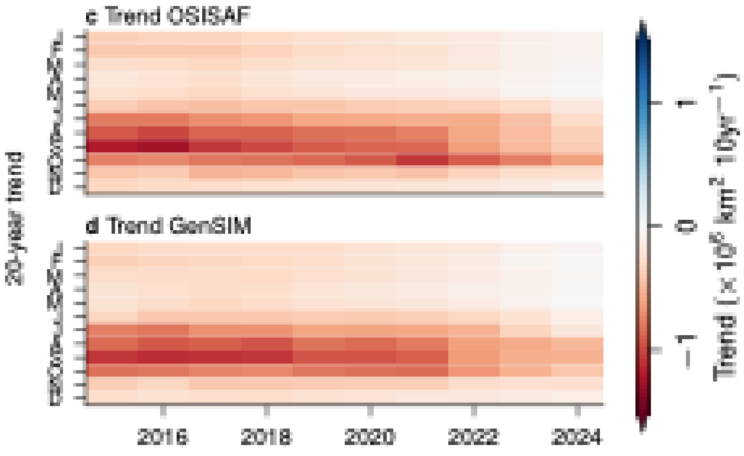


Correct thermodynamical evolution



GenSIM PIOMAS

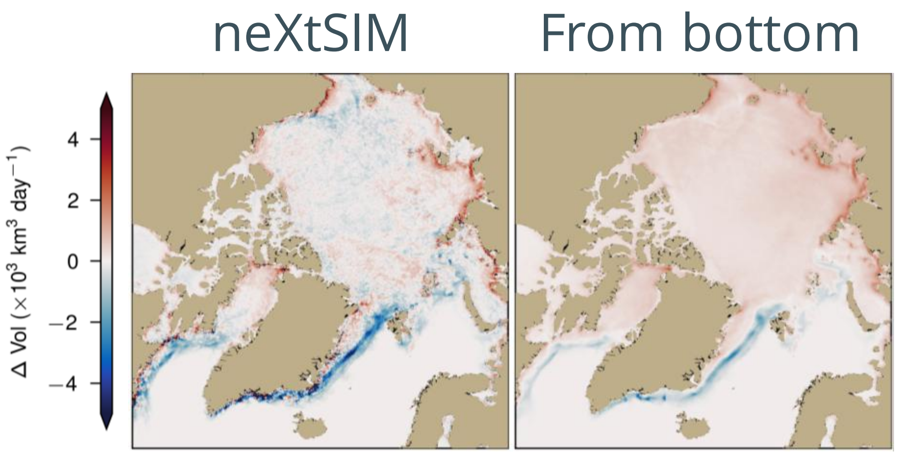




2015→2018

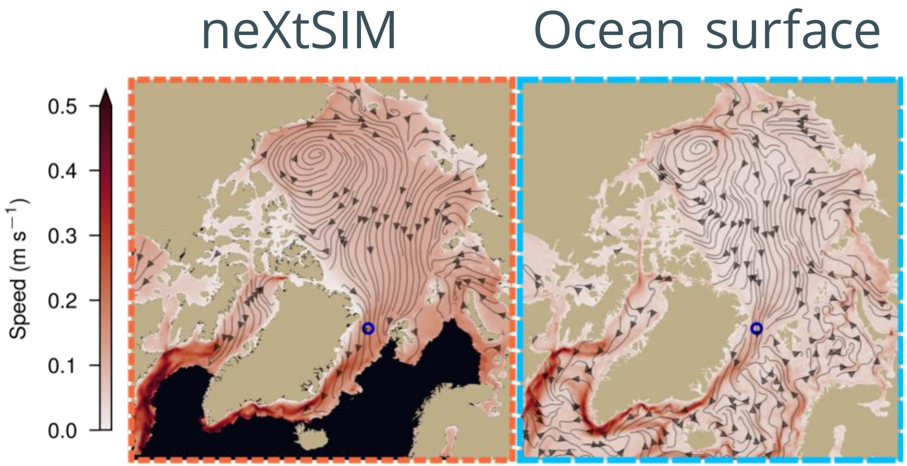
Thermodynamics

Avg vol change in 12 h



Dynamics

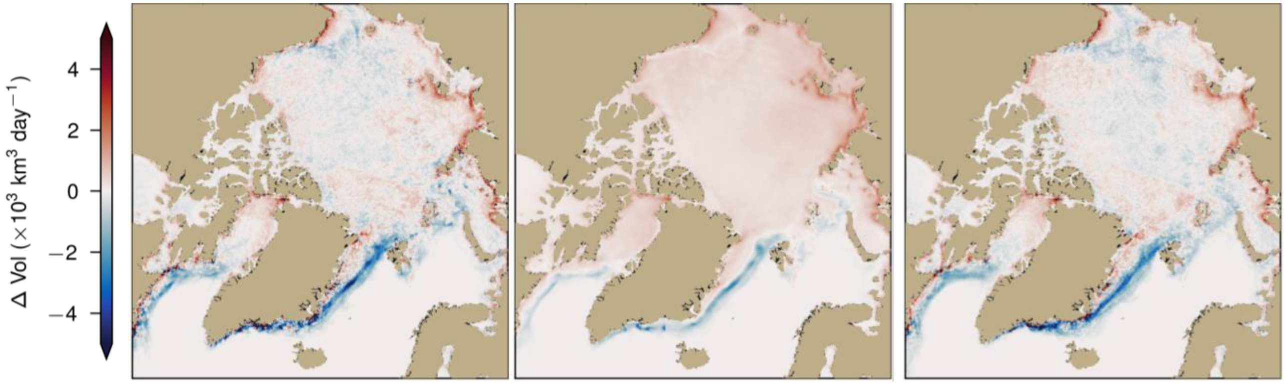
Average drift



2015→2018

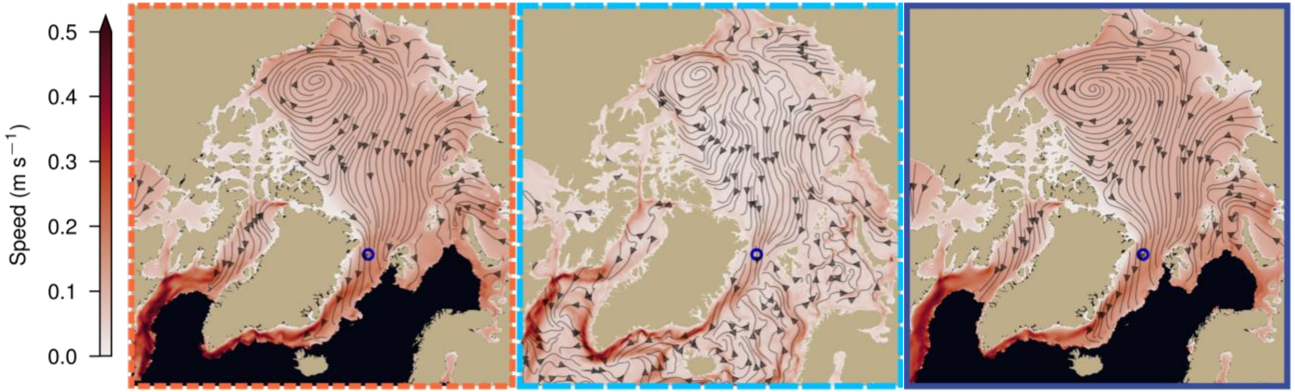
Thermodynamics
Avg vol change in 12 h

neXtSIM | From bottom | GenSIM



Dynamics
Average drift

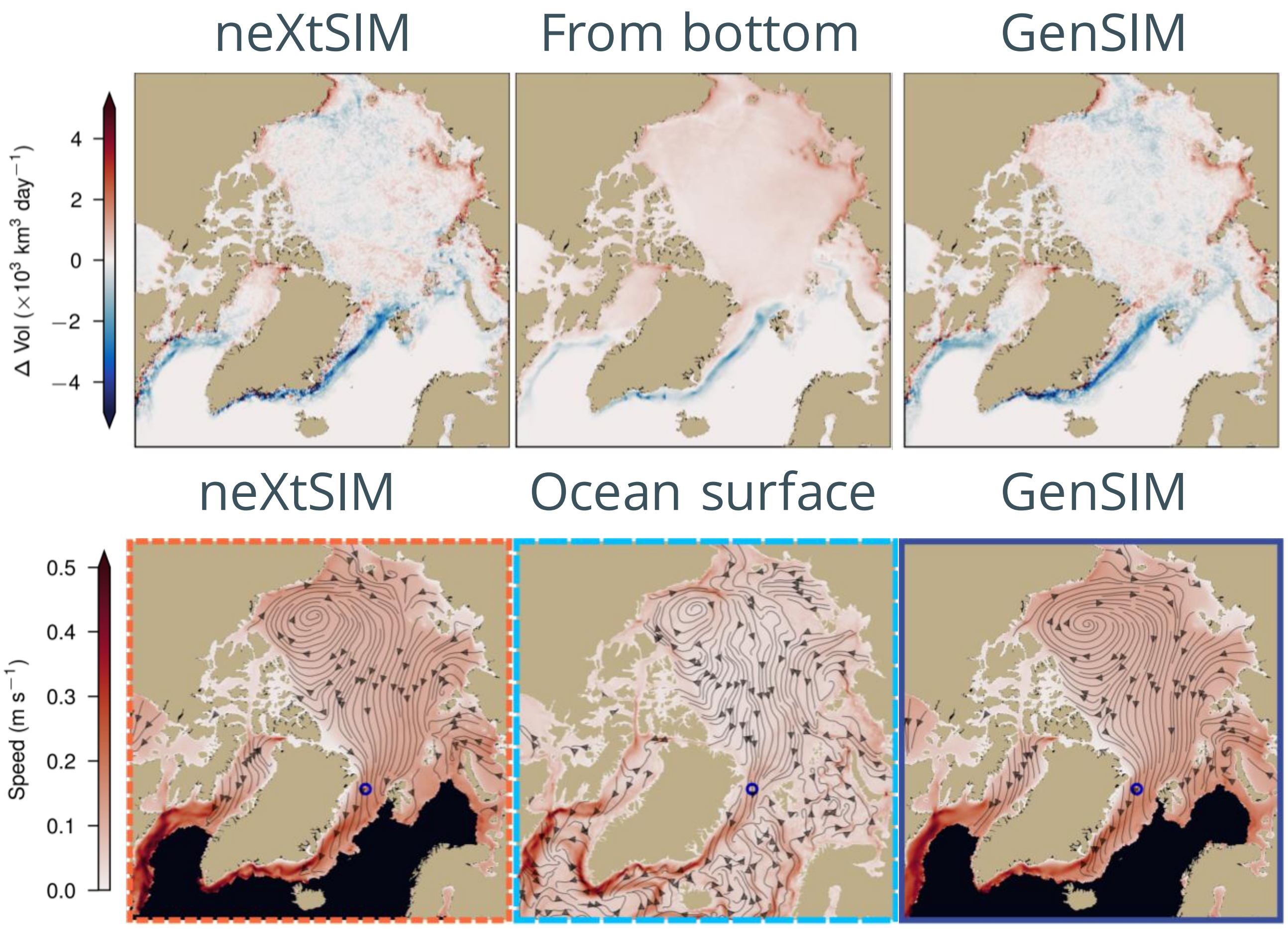
neXtSIM | Ocean surface | GenSIM



2015→2018

Thermodynamics
Avg vol change in 12 h

Dynamics
Average drift



Learning of ocean-like patterns from atmosphere only!

Implictations: Learning of slow component + Coupling in AI-ESM difficult

Claim: Generative flow models are the right tool (at least for sea ice)

Claim: Generative flow models are the right tool (at least for sea ice)

<u>Work</u> similar to classical numerical models
use domain decomposition to scale model up

<u>Unlock</u> improved data-driven modeling
exhibit physically realistic and consistent predictions

<u>Extrapolate</u> sub-daily to decade
implicit learning of slow component and hidden features

Claim: Generative flow models are the right tool (at least for sea ice)

Work similar to classical numerical models
use domain decomposition to scale model up

Unlock improved data-driven modeling
exhibit physically realistic and consistent predictions

Extrapolate sub-daily to decade
implicit learning of slow component and hidden features

Pre-print (under review)

Code + Weights

Claim: Generative flow models are the right tool (at least for sea ice)

Work similar to classical numerical models
use domain decomposition to scale model up

Unlock improved data-driven modeling
exhibit physically realistic and consistent predictions

Extrapolate sub-daily to decade
implicit learning of slow component and hidden features

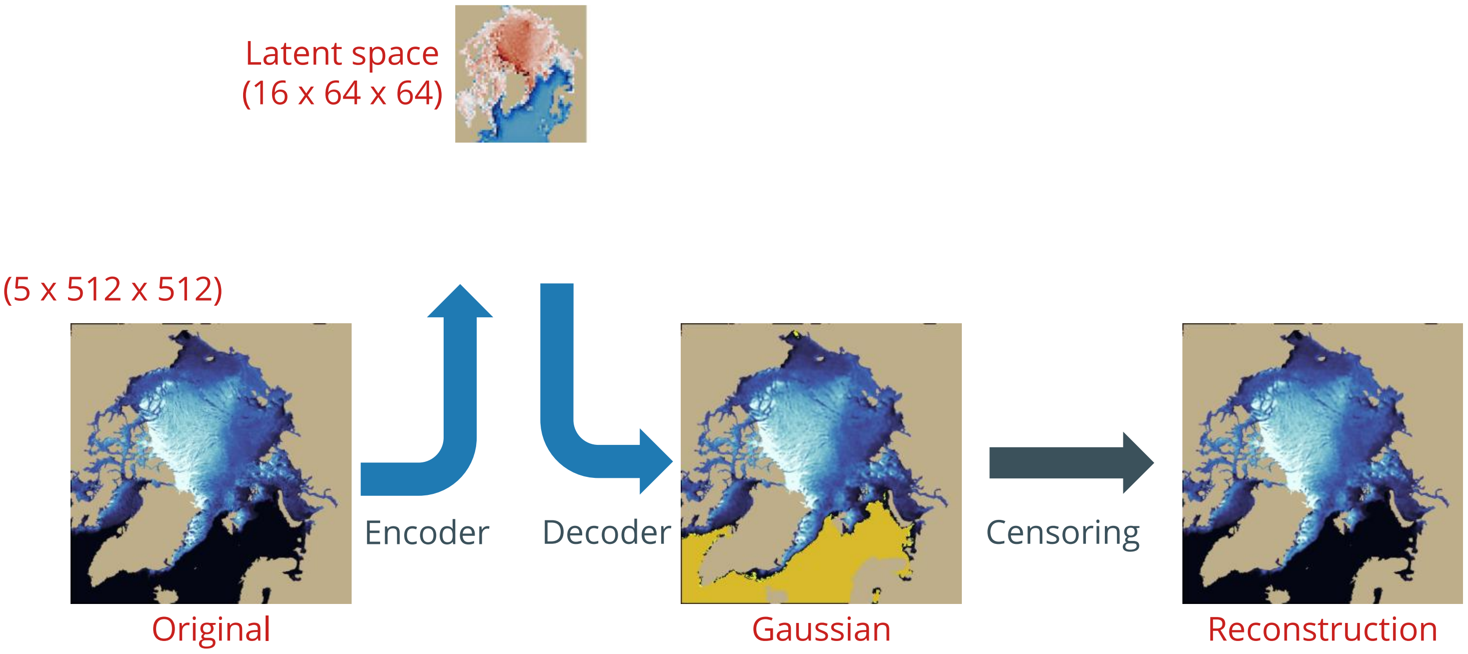
Pre-print (under review)

Code + Weights

Do you have questions?

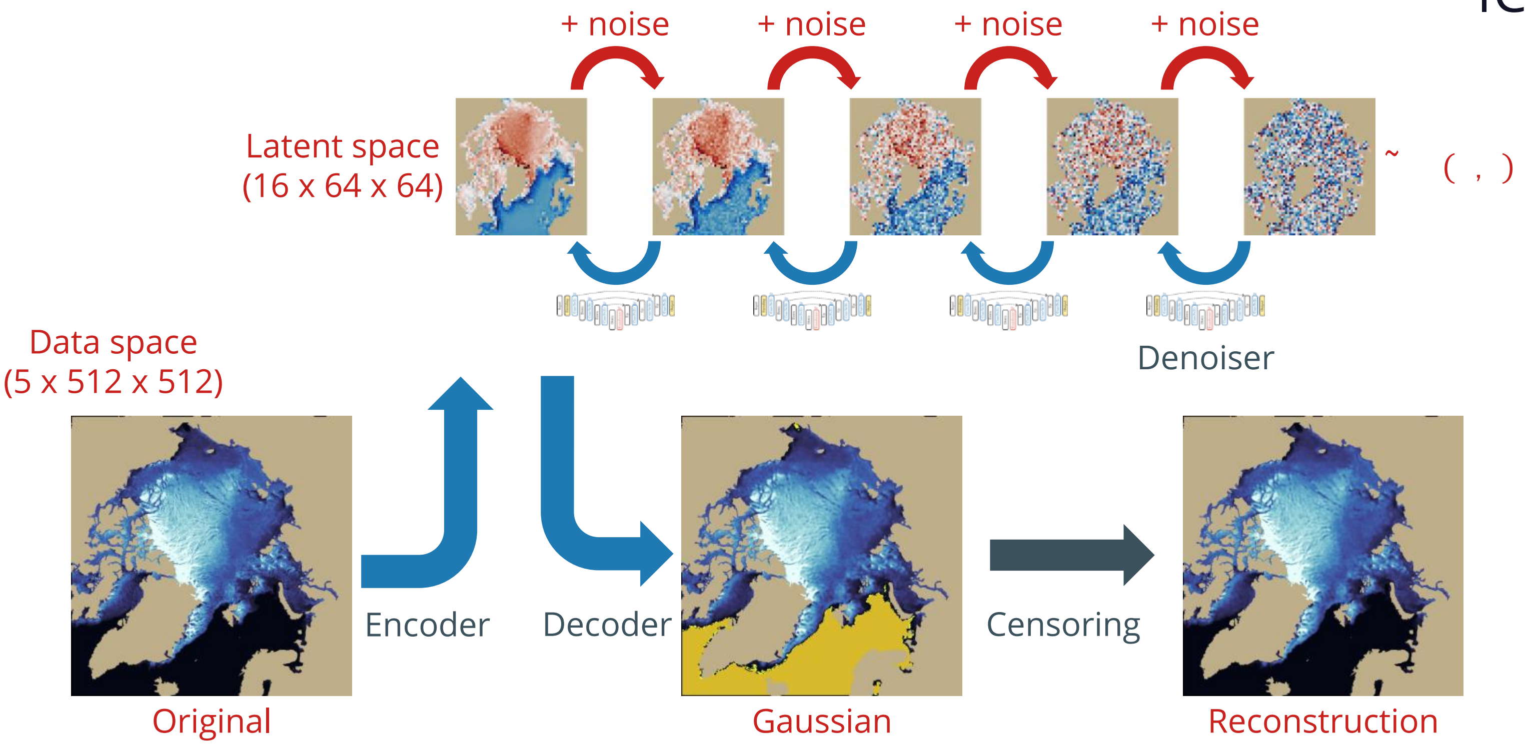
Scaling up: latent diffusion models

ICML '24



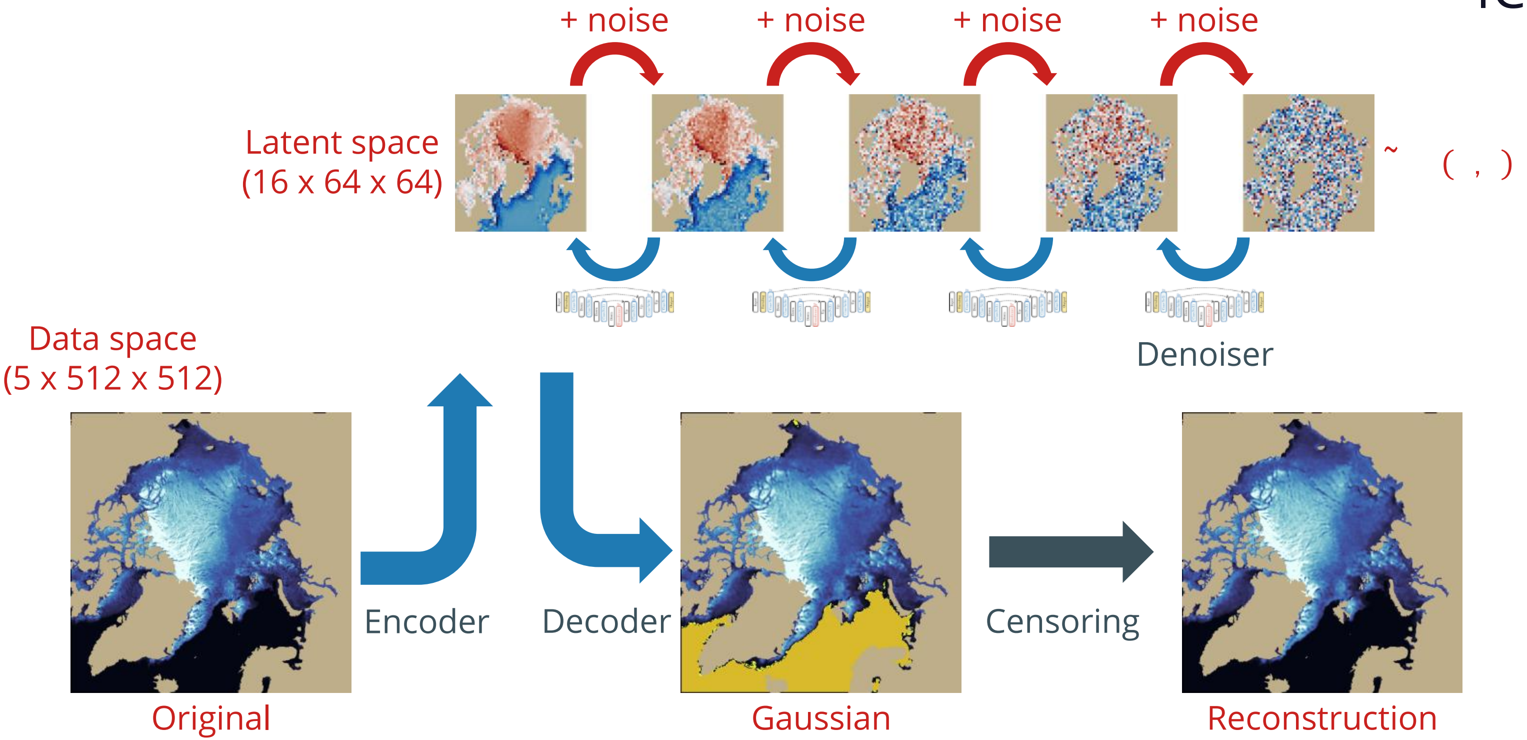
Scaling up: latent diffusion models

ICML ‘24



Scaling up: latent diffusion models

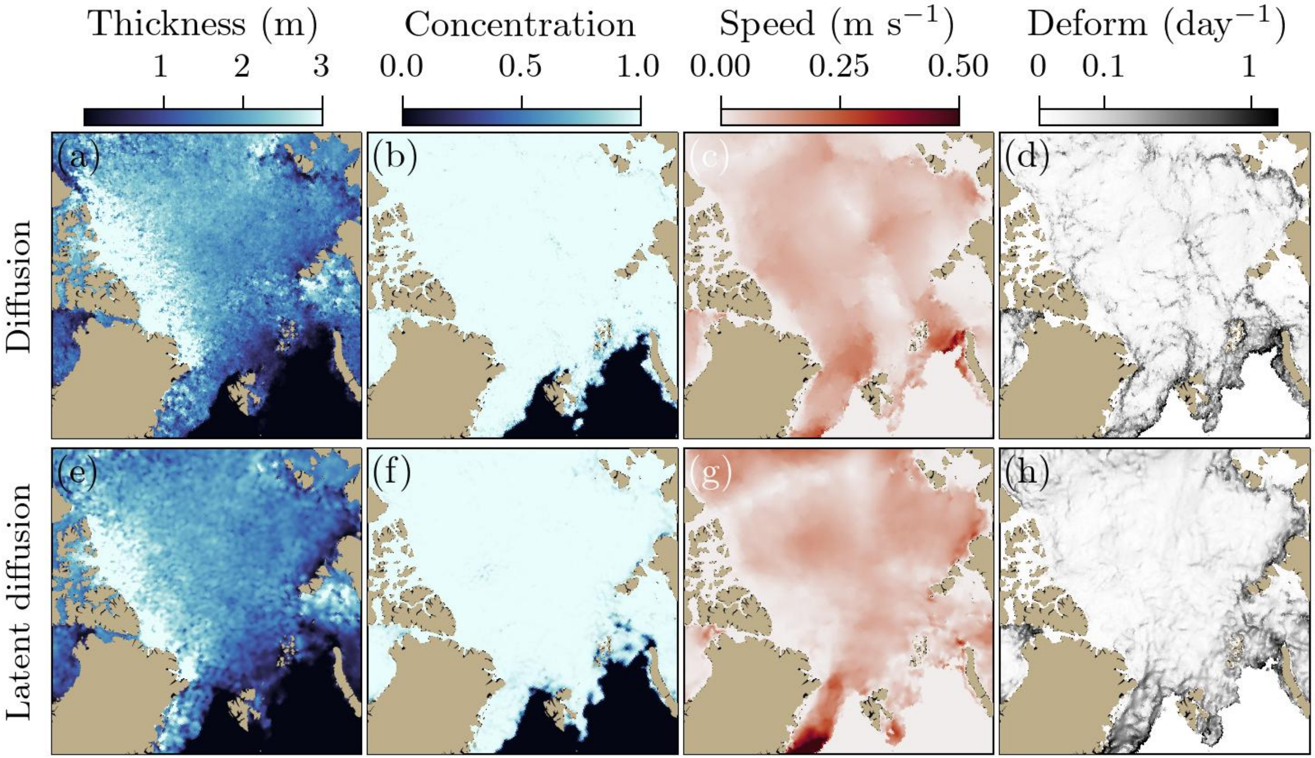
ICML '24



25x faster + Prior physical knowledge

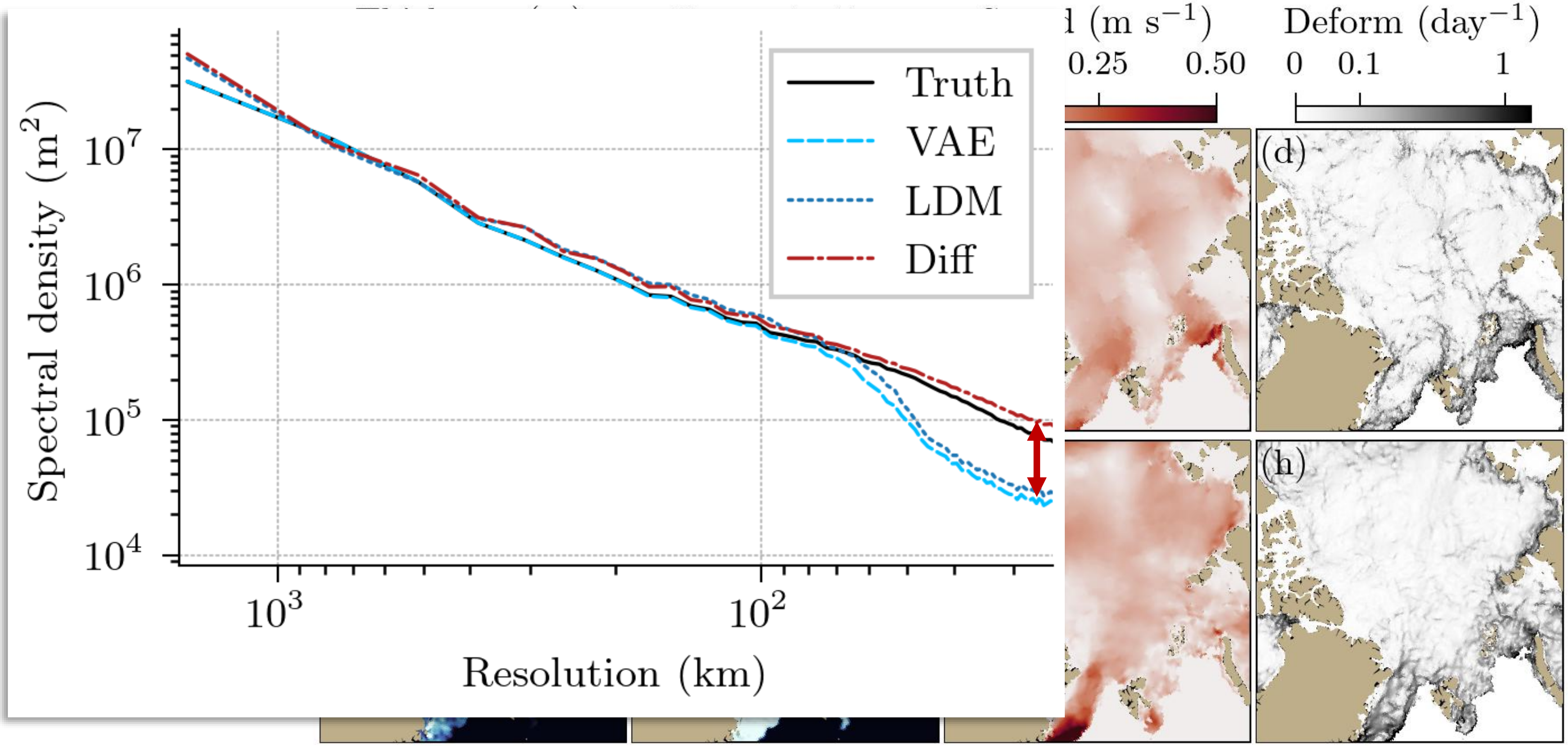
Scaling up: latent diffusion models

ICML '24



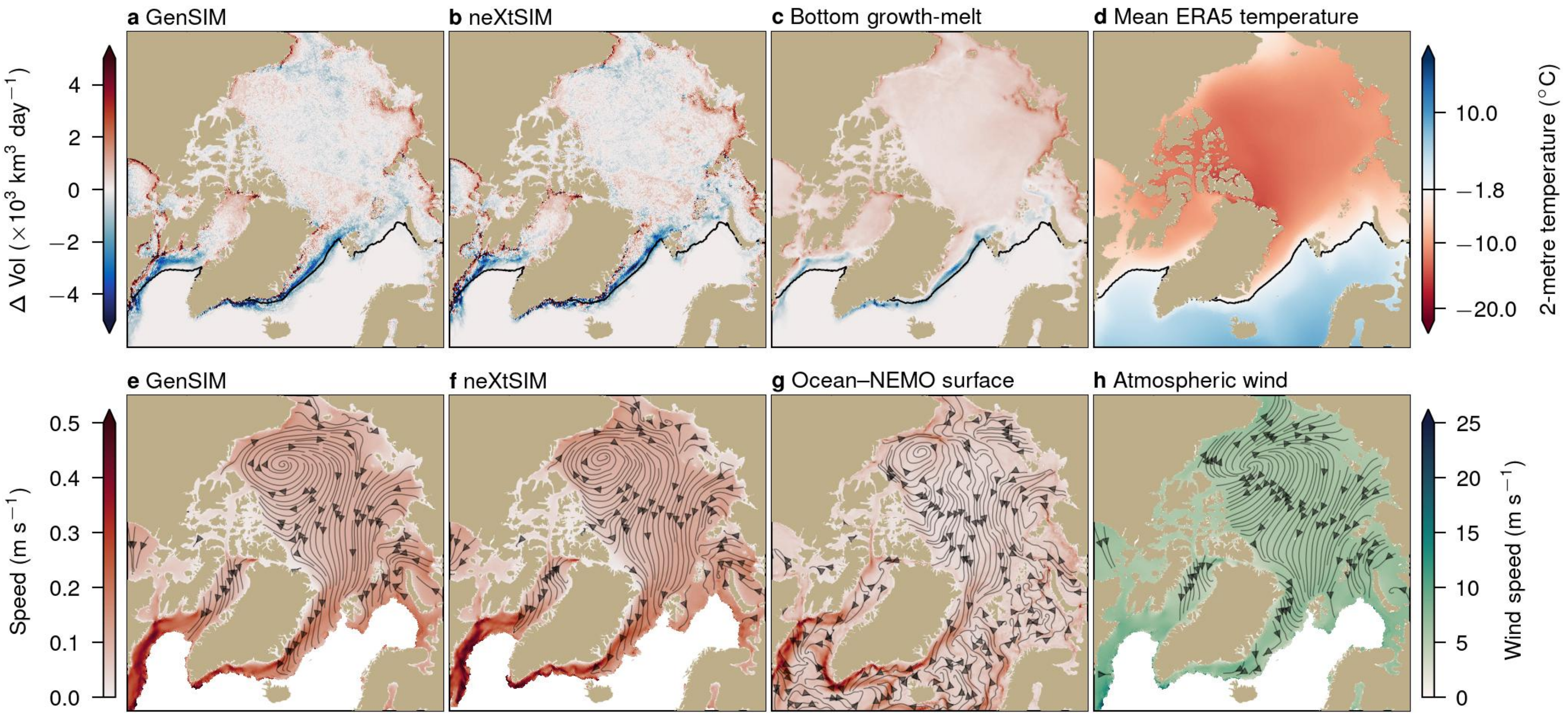
Scaling up: latent diffusion models

ICML ‘24

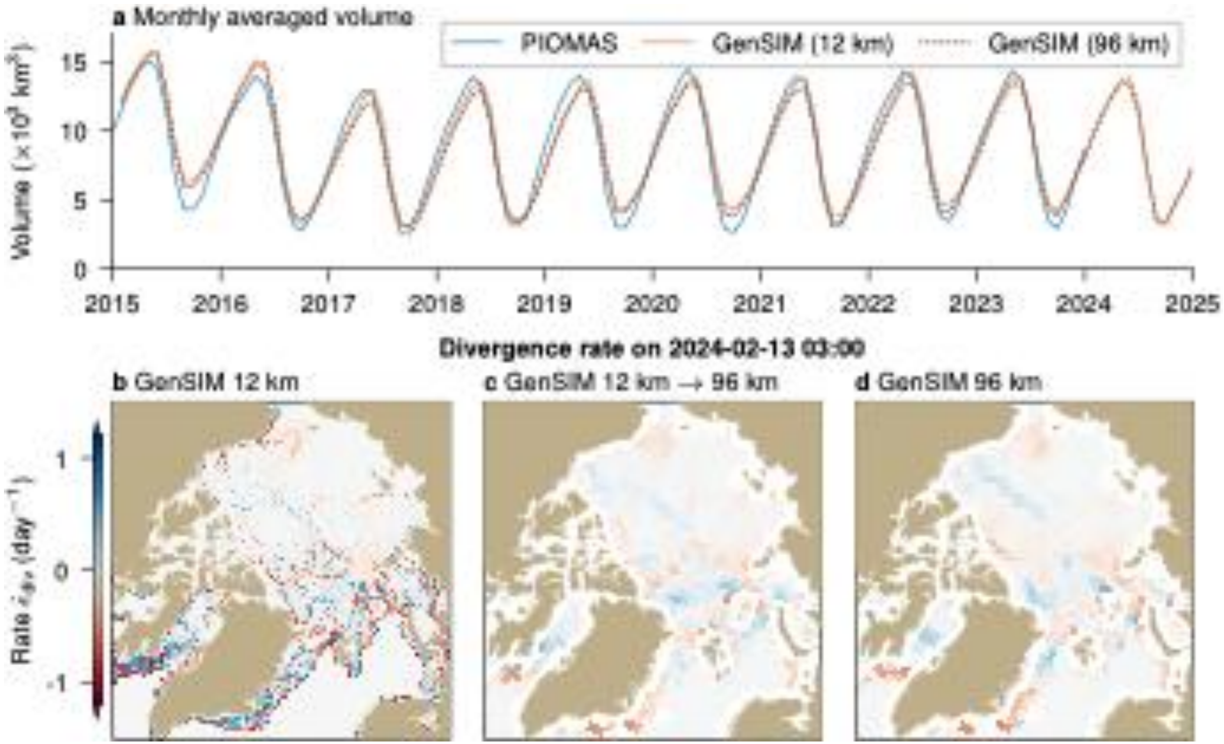


Reintroduce problems with smoothing due to *wrong* compression

Non-trivial proxy learning



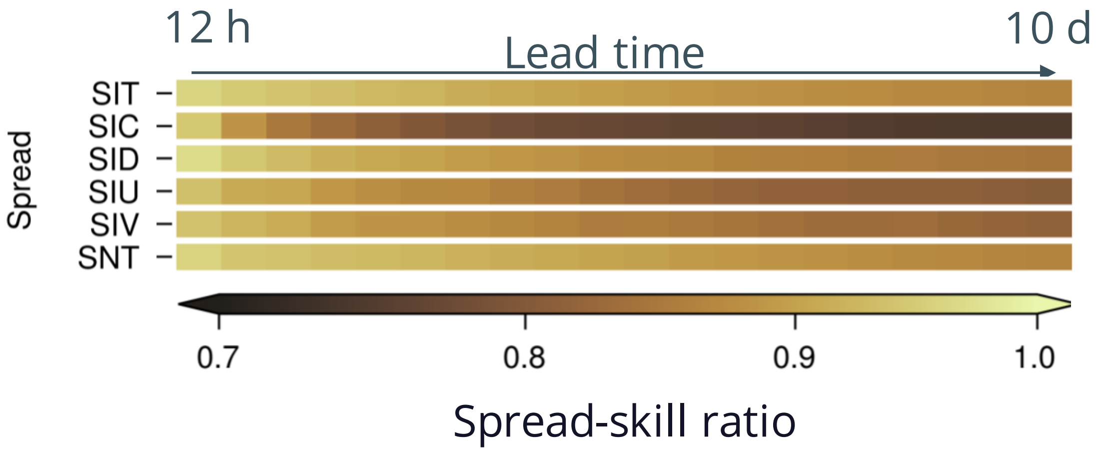
Generalisation to multiple resolutions and grids

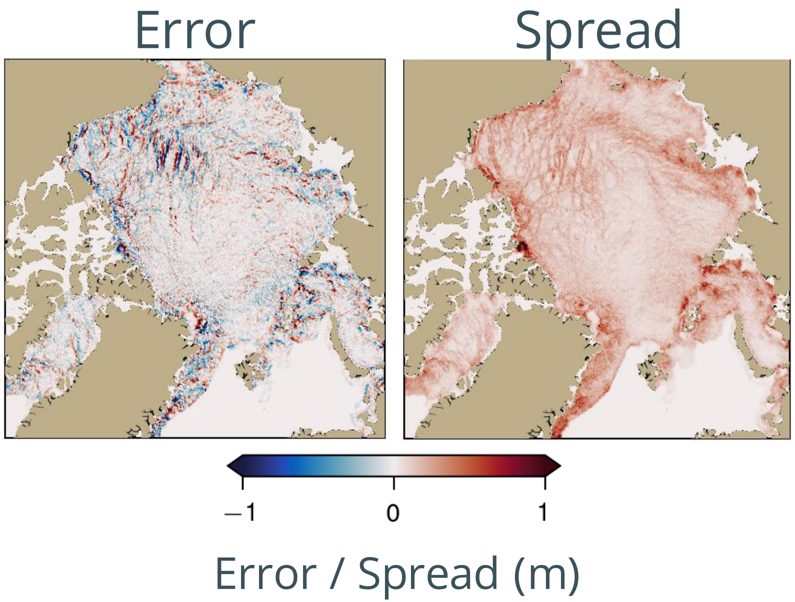


Idealised cyclone

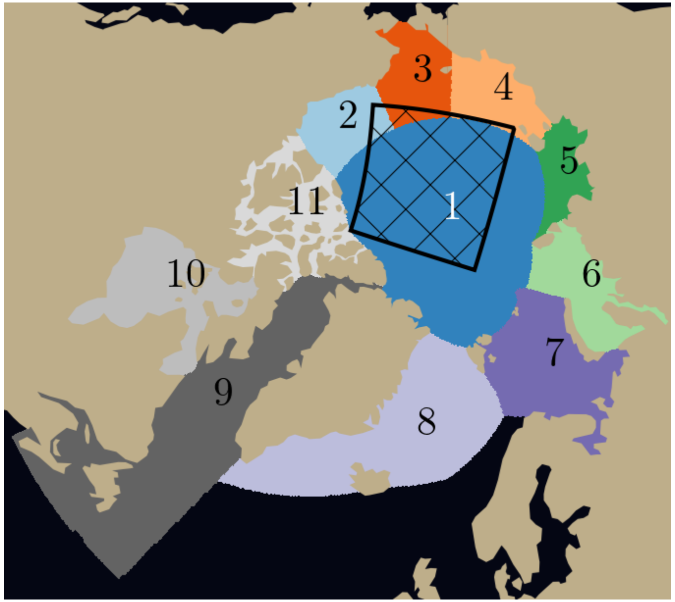
Realistic ensemble spread, yet slightly underdispersive

Quantitative evaluation of spread

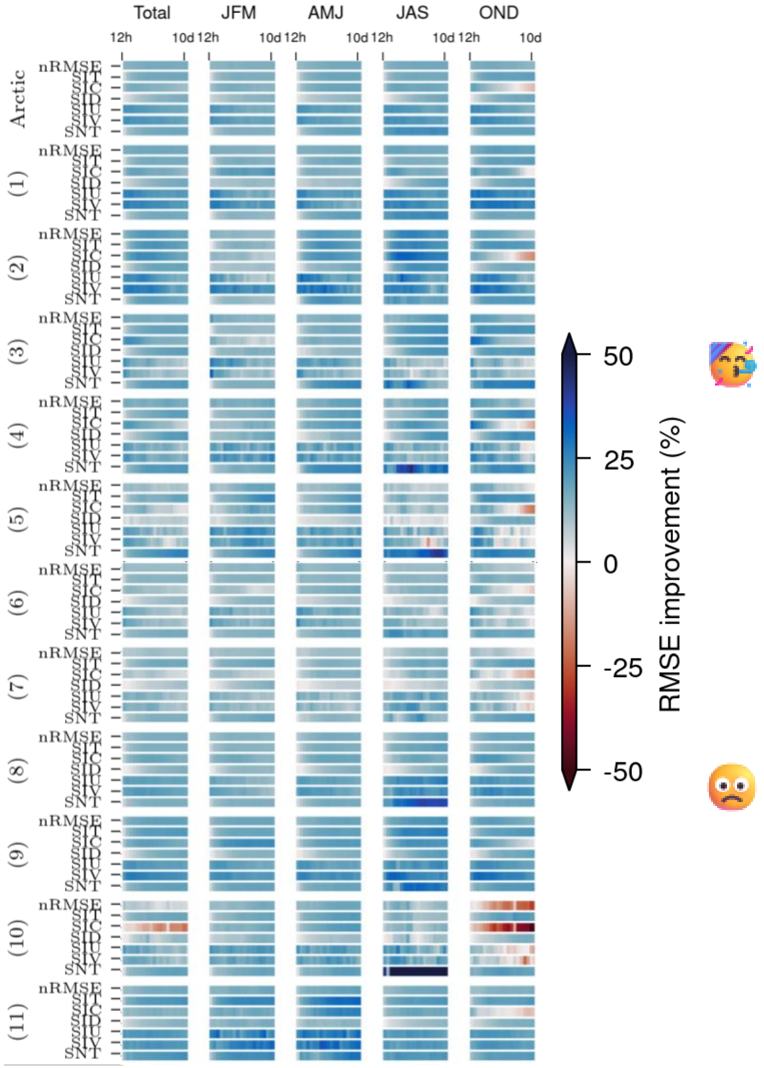




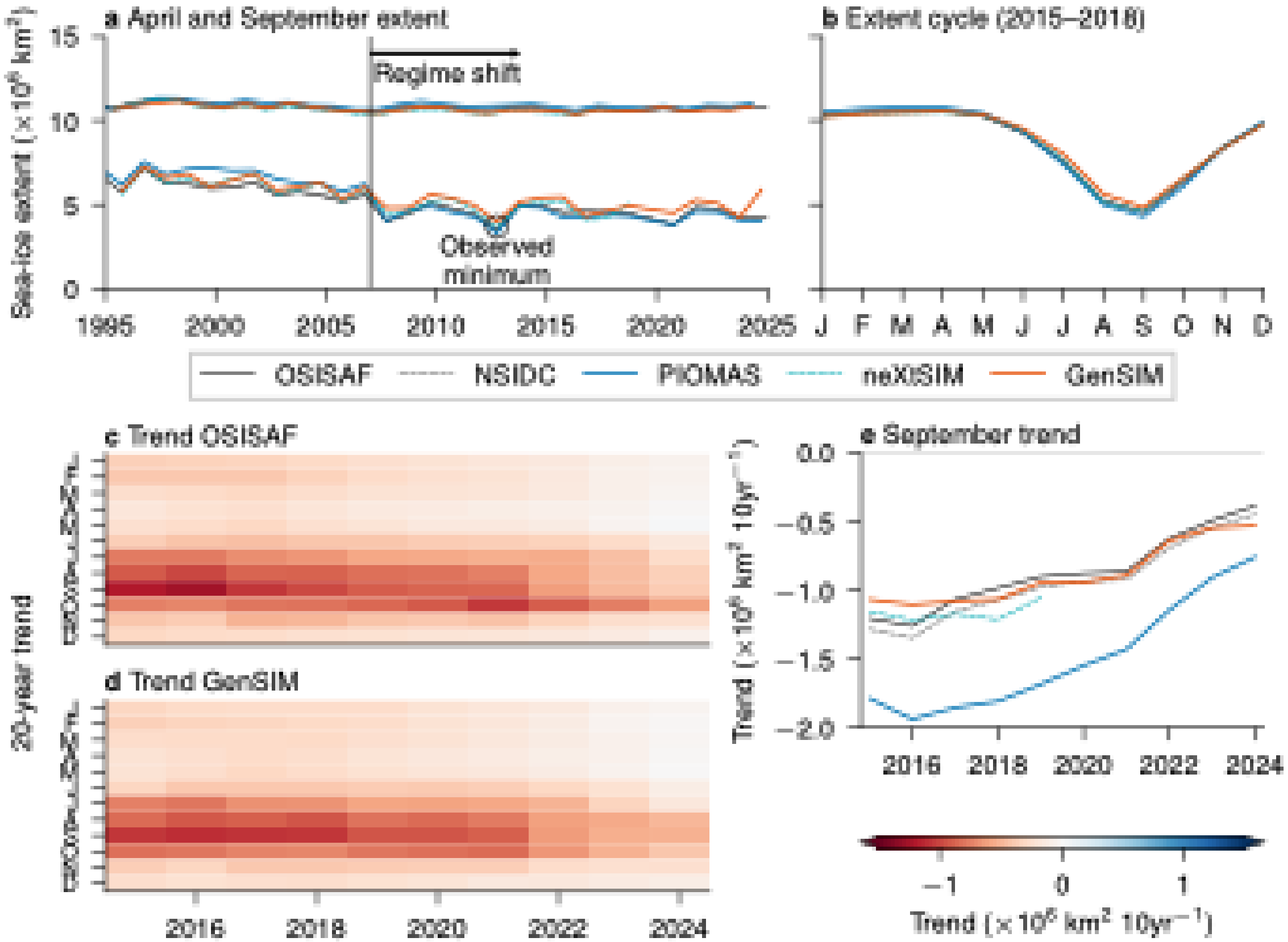
Better than Deterministic



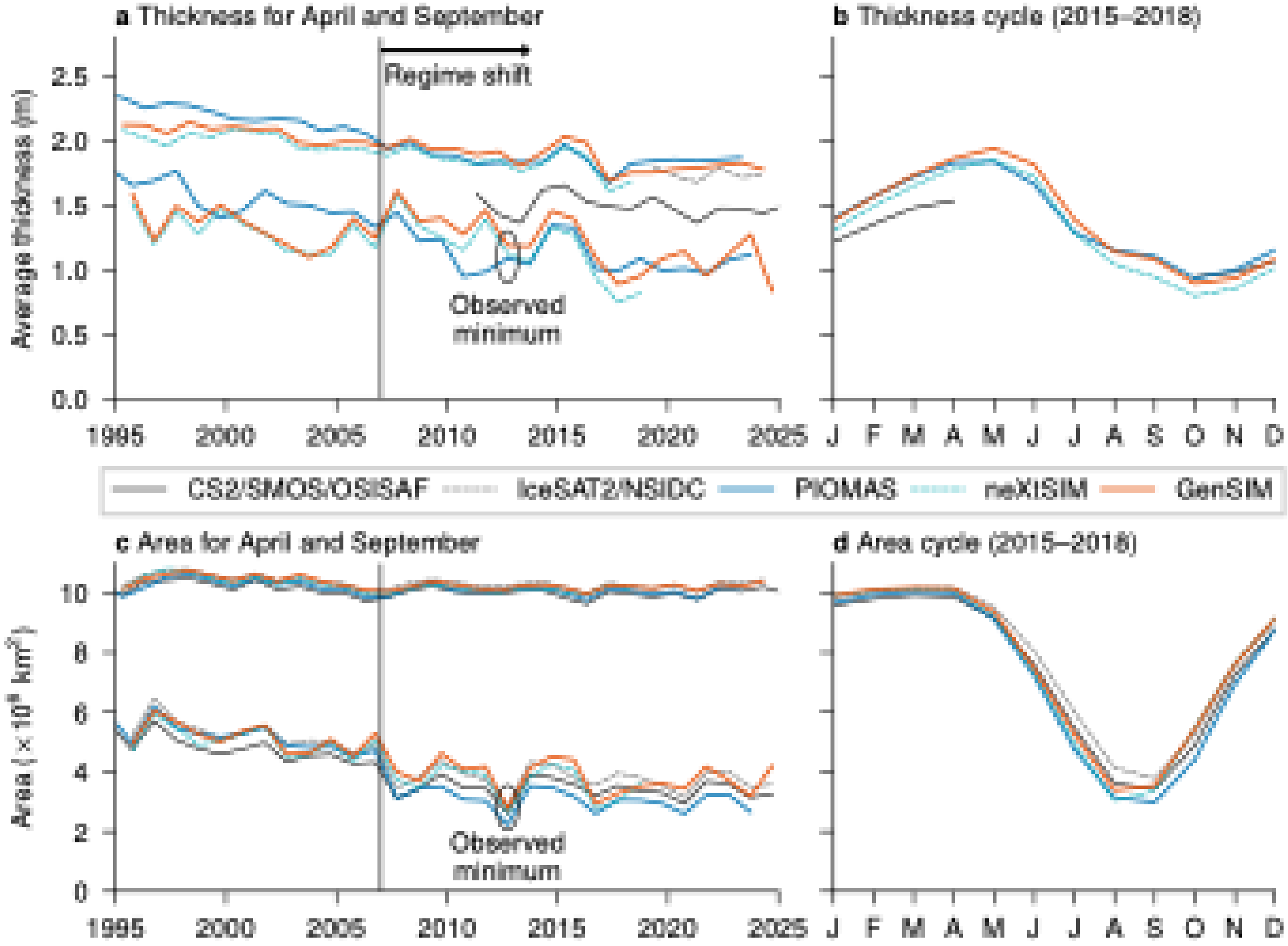
$$\frac{\text{RMSE of GenSIM}}{\text{RMSE of Deterministic}}$$



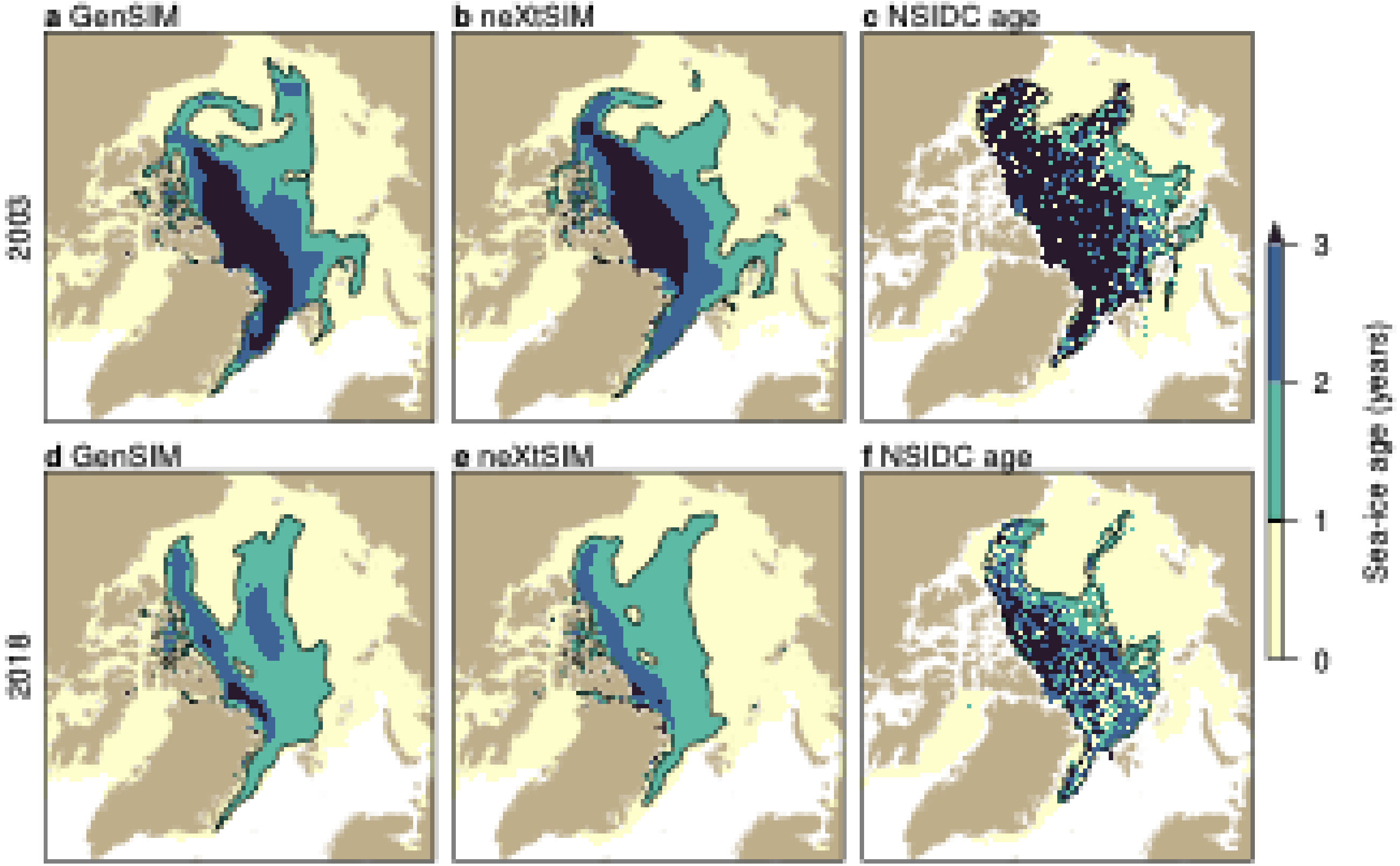
Capturing of sea-ice extent evolution



Small bias in thickness

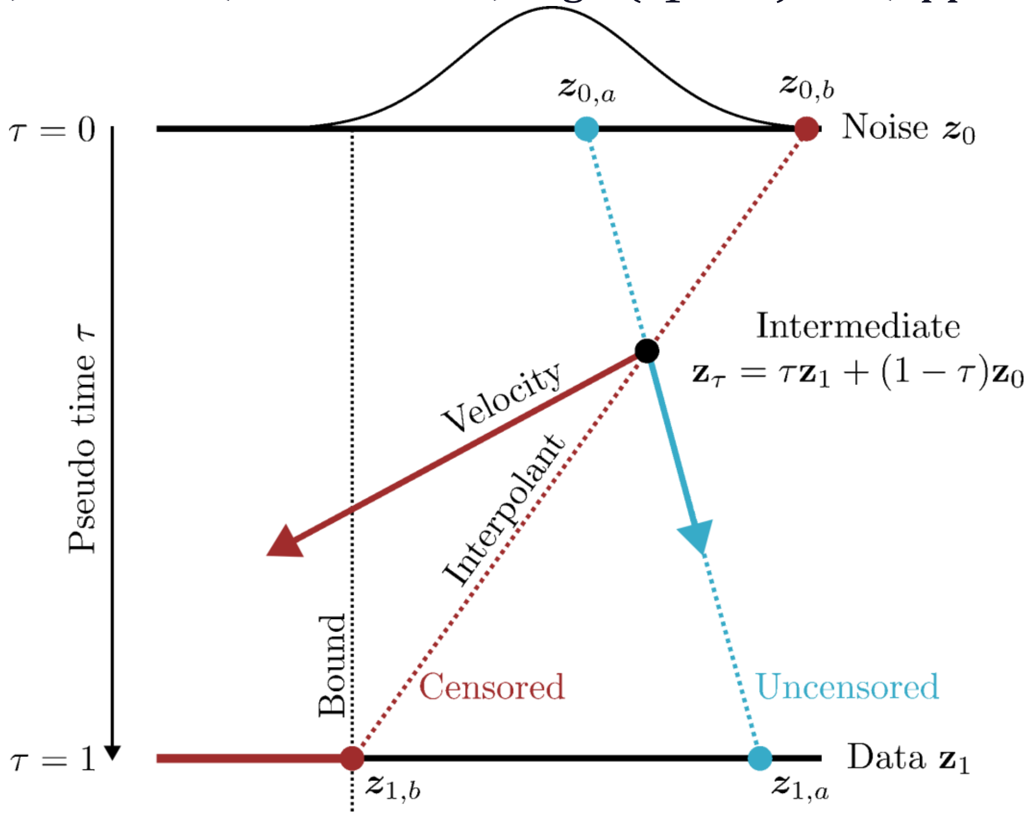


Transition signature in sea-ice age



Physical bounded predictions by censored flows

$$\text{Loss} = \mathbb{I}(\text{unbounded})\ \text{MSE} - \mathbb{I}(\text{lower bound}) \log P(z_1 \leq L) - \mathbb{I}(\text{upper bound}) \log P(z_1 \geq U)$$



Sampling with clipping into bounds at each step