



Défis théoriques pour les sciences du climat
Grenoble | 29 May 2024

Towards **AI-native** hybrid (ocean) models : The need and the challenges ahead

Julien Le Sommer (IGE, Grenoble)
with input from many in Brest, Paris, Toulouse and Grenoble





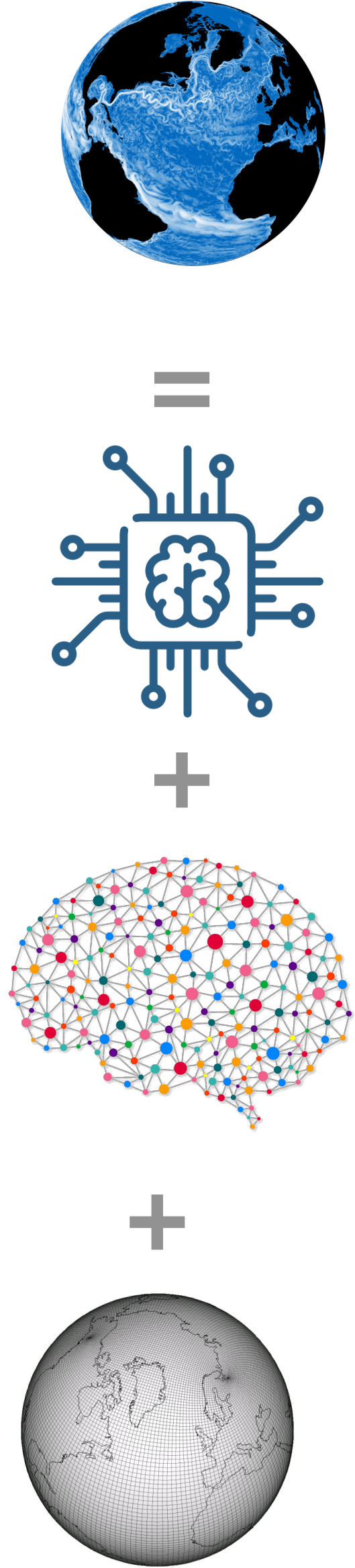
Défis théoriques pour les sciences du climat
Grenoble | 29 May 2024

Towards **AI-native** hybrid (geoscientific) models : The need and the challenges ahead

Julien Le Sommer (IGE, Grenoble)
with input from many in Brest, Paris, Toulouse and Grenoble



Objectives of this talk



- Illustrate why we explore **augmenting models** with ML
- Illustrate **how this is done** in practice today
- Advocate that a **deep recast** of our models is needed
- Discuss some steps towards **AI-native hybrid** models



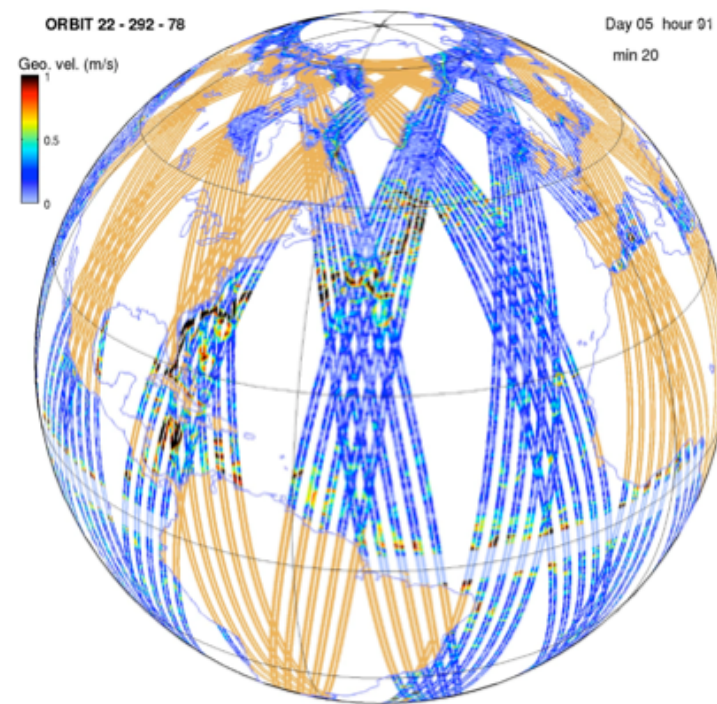
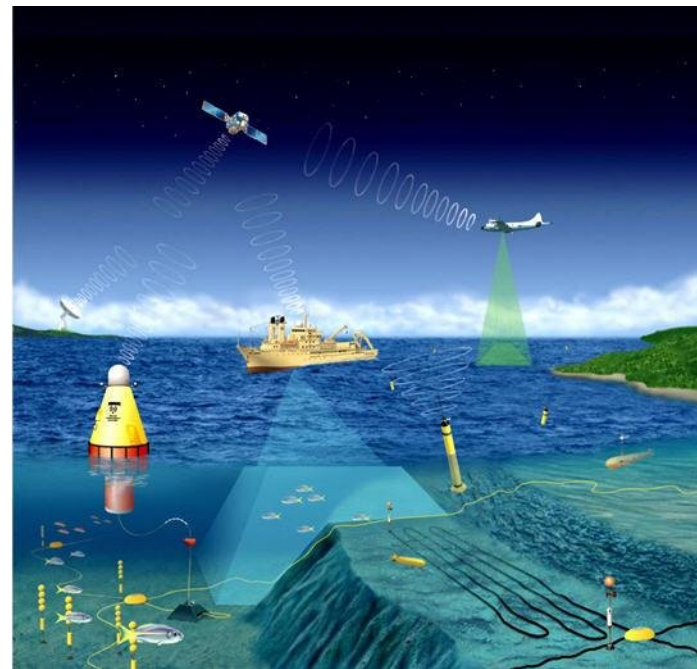


New toys in the oceanographer's **toolbox**

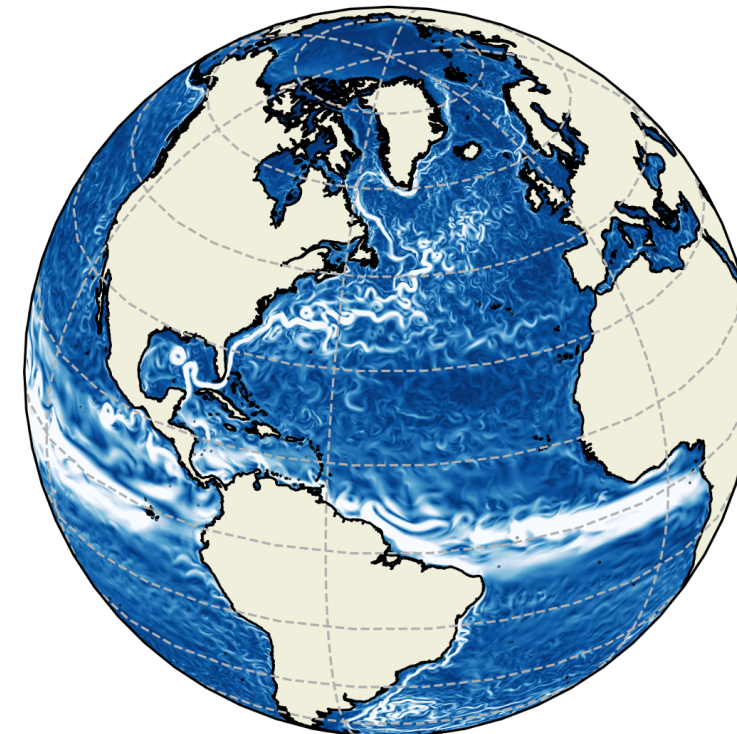
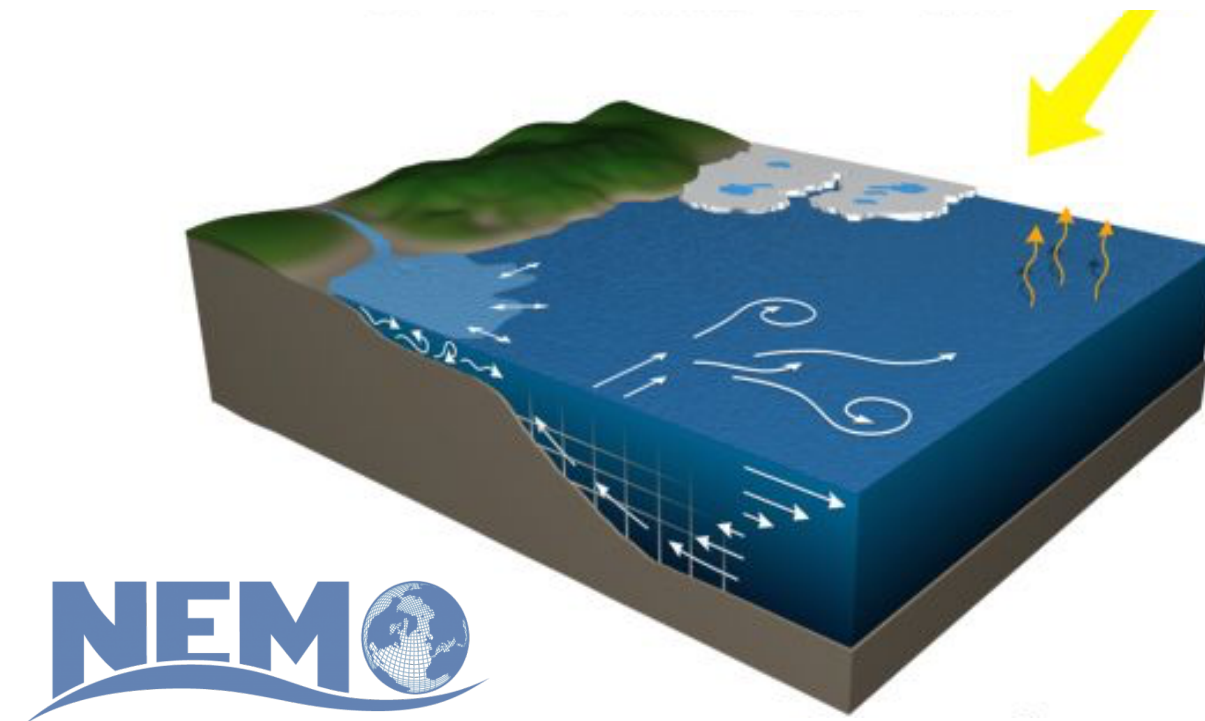


Computational oceanographer's toolbox

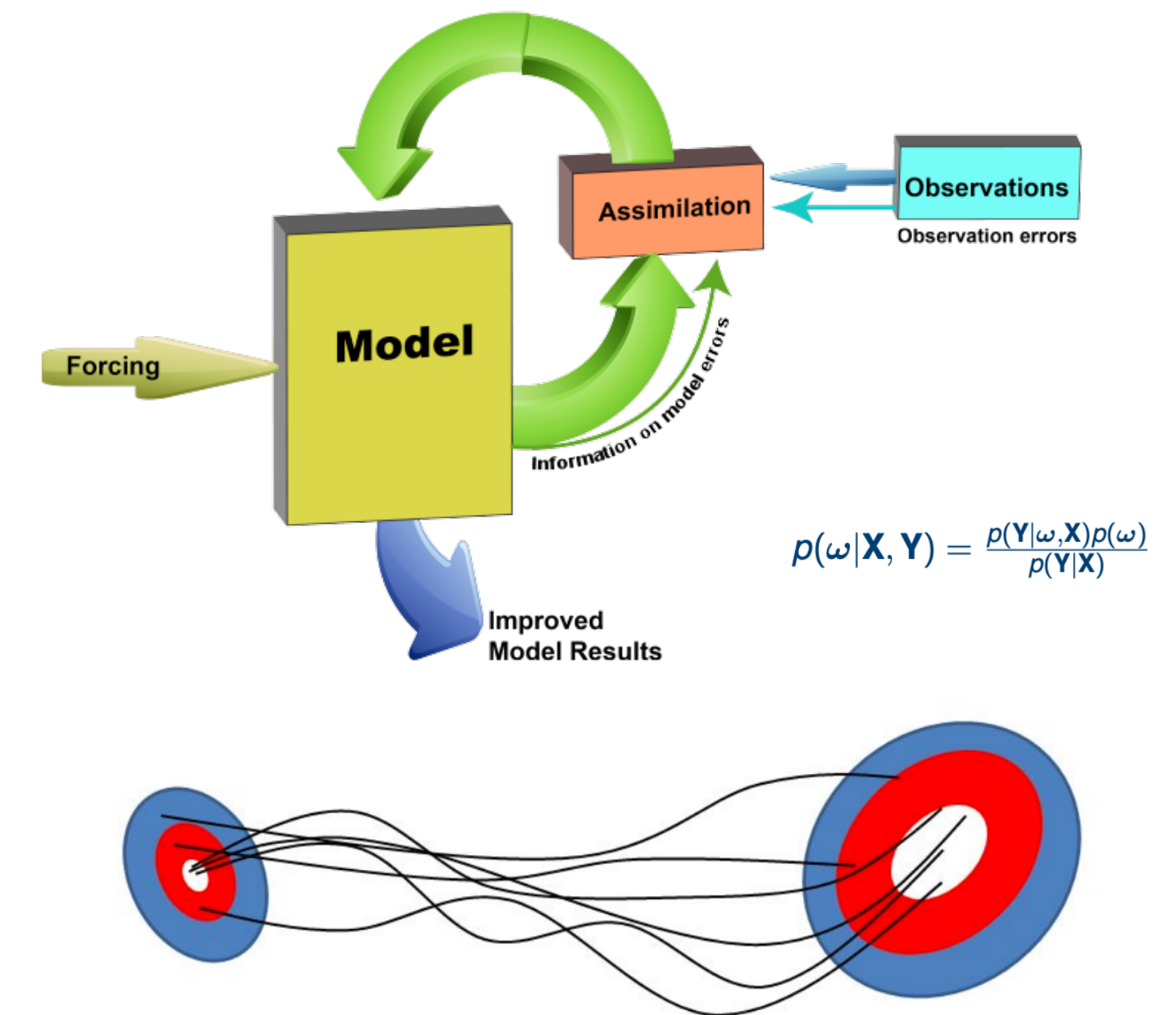
Observations (in situ/satellite)



Physical models (ocean circulation models)

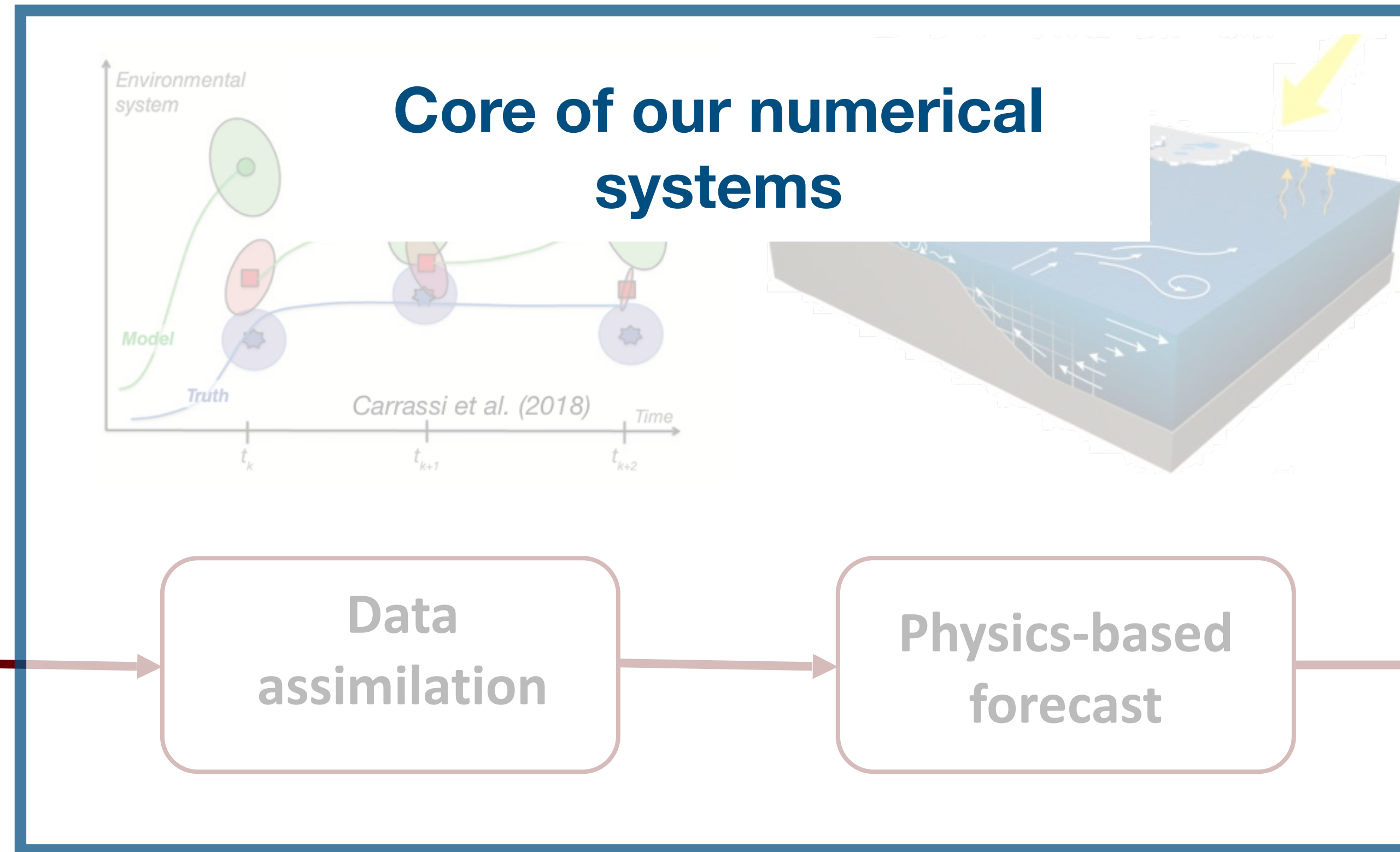
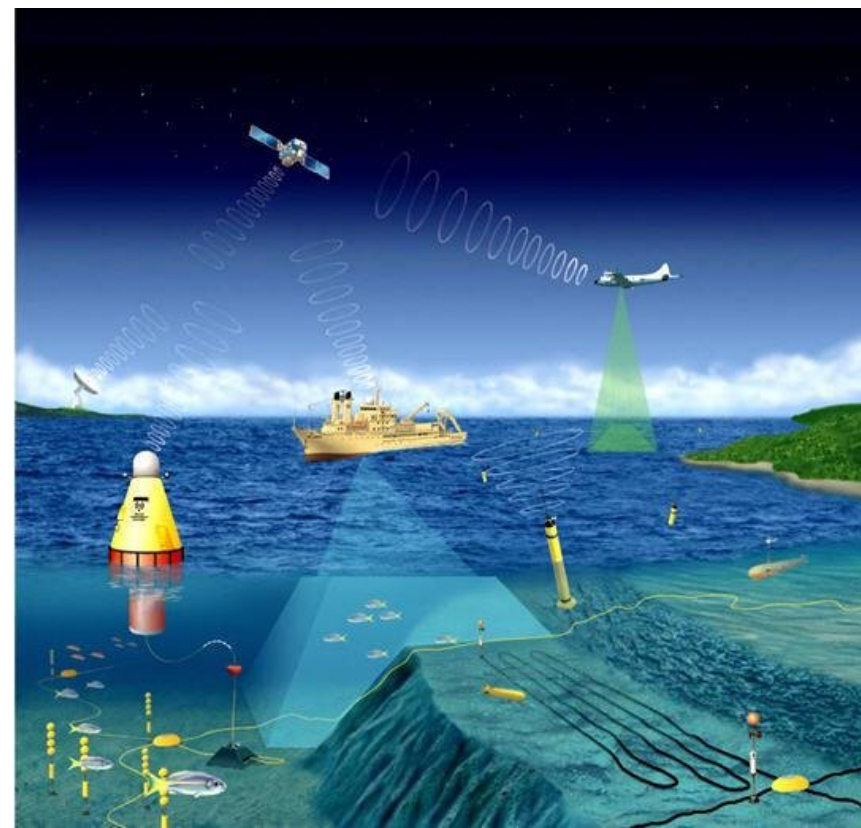


Inverse methods (data assimilation)



Tools for **understanding** but also monitoring and **forecasting** ocean circulation

How AI is affecting our numerical systems



Observations

**Data
assimilation**

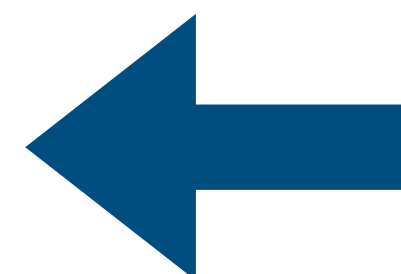
**Physics-based
forecast**

**Post-processing,
dissemination**

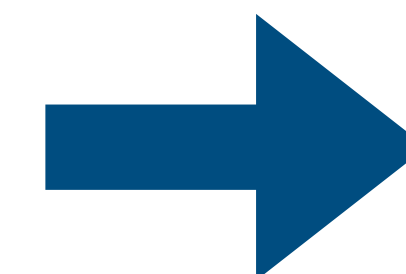
Upstream

Downstream

*denoising, inpainting
parameter retrieval
quality control*



**AI, machine learning &
data-driven approaches**



*data fusion,
tailored services
data mining*

COMING UP

AI-based ocean forecasting

MANUSCRIPT

XiHe: A Data-Driven Model for Global Ocean Eddy-Resolving Forecasting

Xiang Wang, Renzhi Wang, Ningzi Hu, Pinqiang Wang, Peng Huo, Guihua Wang, Huizan Wang, Senzhang Wang, Junxing Zhu, Jianbo Xu, Jun Yin, Senliang Bao, Ciqiang Luo, Ziqing Zu, Yi Han, Weimin Zhang, Kaijun Ren, Kefeng Deng, Junqiang Song

Abstract—Global ocean forecasting is fundamentally important to support marine activities. The leading operational Global Ocean Forecasting Systems (GOFs) use physics-driven numerical forecasting models that solve the partial differential equations with expensive computation. Recently, specifically in atmosphere weather forecasting, data-driven models have demonstrated significant potential for speeding up environmental forecasting by orders of magnitude, but there is still no data-driven GOFs that matches the forecasting accuracy of the numerical GOFs. In this paper, we propose the first data-driven $1/12^\circ$ resolution global ocean eddy-resolving forecasting model named *XiHe*, which is established from the 25-year France Mercator Ocean International's daily GLORYS12 reanalysis data. *XiHe* is a hierarchical transformer-based framework coupled with two special designs. One is the land-ocean mask mechanism for focusing exclusively on the global ocean circulation. The other is the ocean-specific block for effectively capturing both local ocean information and global teleconnection. Extensive experiments are conducted under satellite observations, *in situ* observations, and the IV-TT Class 4 evaluation framework of the world's leading operational GOFs from January 2019 to December 2020. The results demonstrate that *XiHe* achieves stronger forecast performance in all testing variables than existing leading operational numerical GOFs including Mercator Ocean Physical System (PSY4), Global Ice Ocean Prediction System (GIOPS), BLUEINK OceanMAPS (BLK), and Forecast Ocean Assimilation Model (FOAM). Particularly, the accuracy of ocean current forecasting of *XiHe* out to 60 days is even better than that of PSY4 in just 10 days. Additionally, *XiHe* is able to forecast the large-scale circulation and the mesoscale eddies. Furthermore, it can make a 10-day forecast in only 0.36 seconds, which accelerates the forecast speed by thousands of times compared to the traditional numerical GOFs.

Index Terms—Global Ocean Forecasting, Deep Learning, Eddy Resolving, Data-Driven, AI for Science

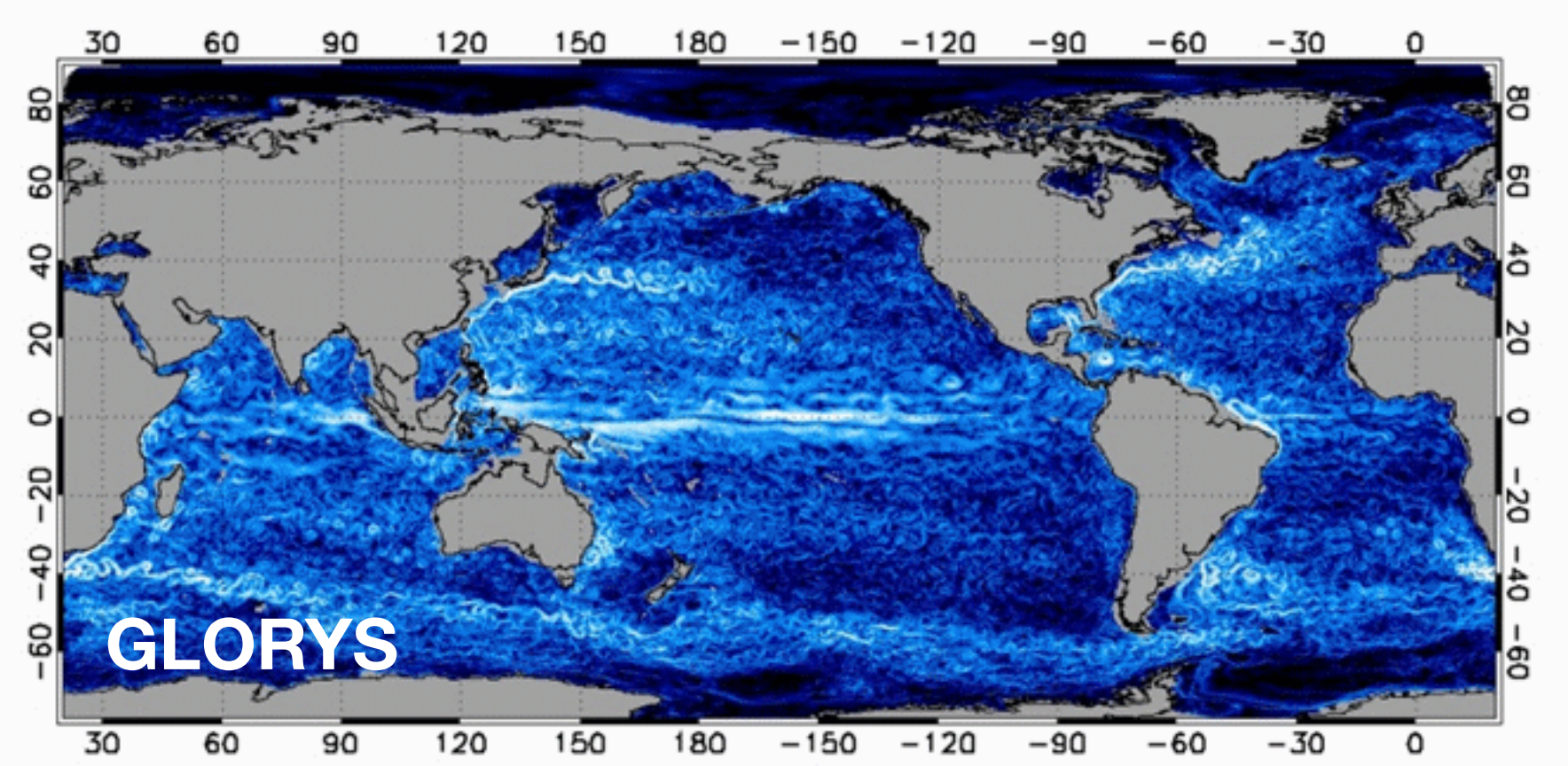
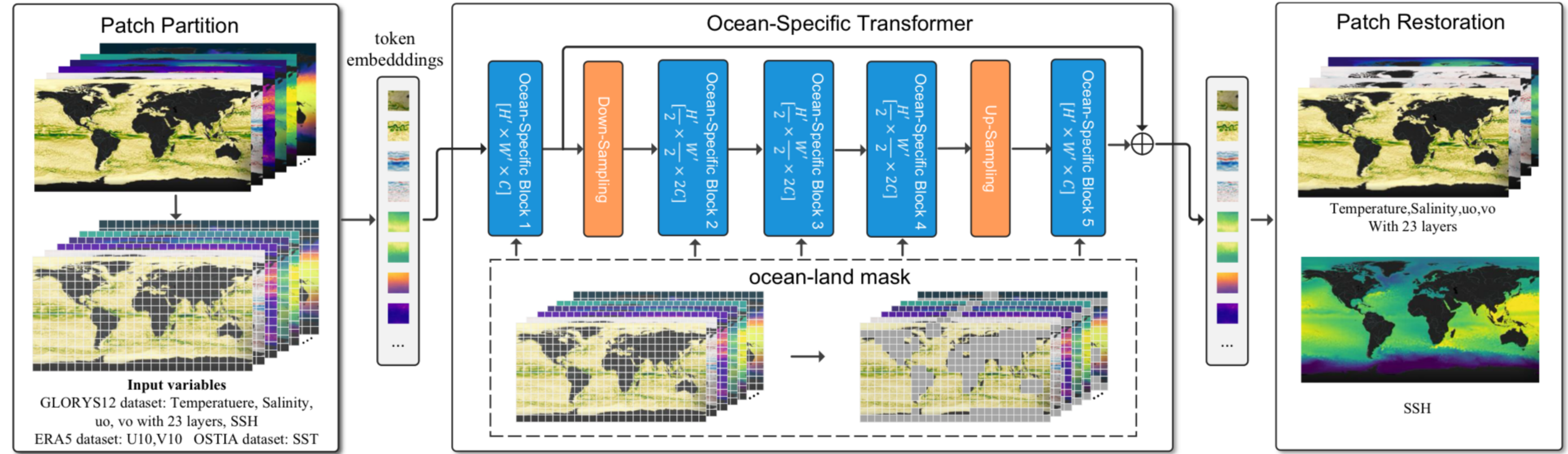
1 INTRODUCTION

Ocean forecasting is critically important for many marine activities. At present, the leading GOFs (e.g. Mercator Ocean Physical System (PSY4) and Real-Time Ocean Forecast System (RTOFS)) use physics-driven models in fluid mechanics and thermodynamics to predict future ocean motion states and phenomena based on current ocean conditions [1]. The GOFs adopt numerical methods that rely on supercomputers to solve the partial differential equations of the physical models. Due to their desirable performance, they are operationally run in different countries worldwide. However, numerical forecasting methods are usually computationally expensive and slow. For example, a single forecasting simulation in the numerical GOFs may take hours on a supercomputer with hundreds of computational nodes [2]. Besides, improving the forecasting accuracy of these methods is exceedingly challenging because they heavily rely on the human cognitive abilities in understanding the physical laws of the ocean environment [3].

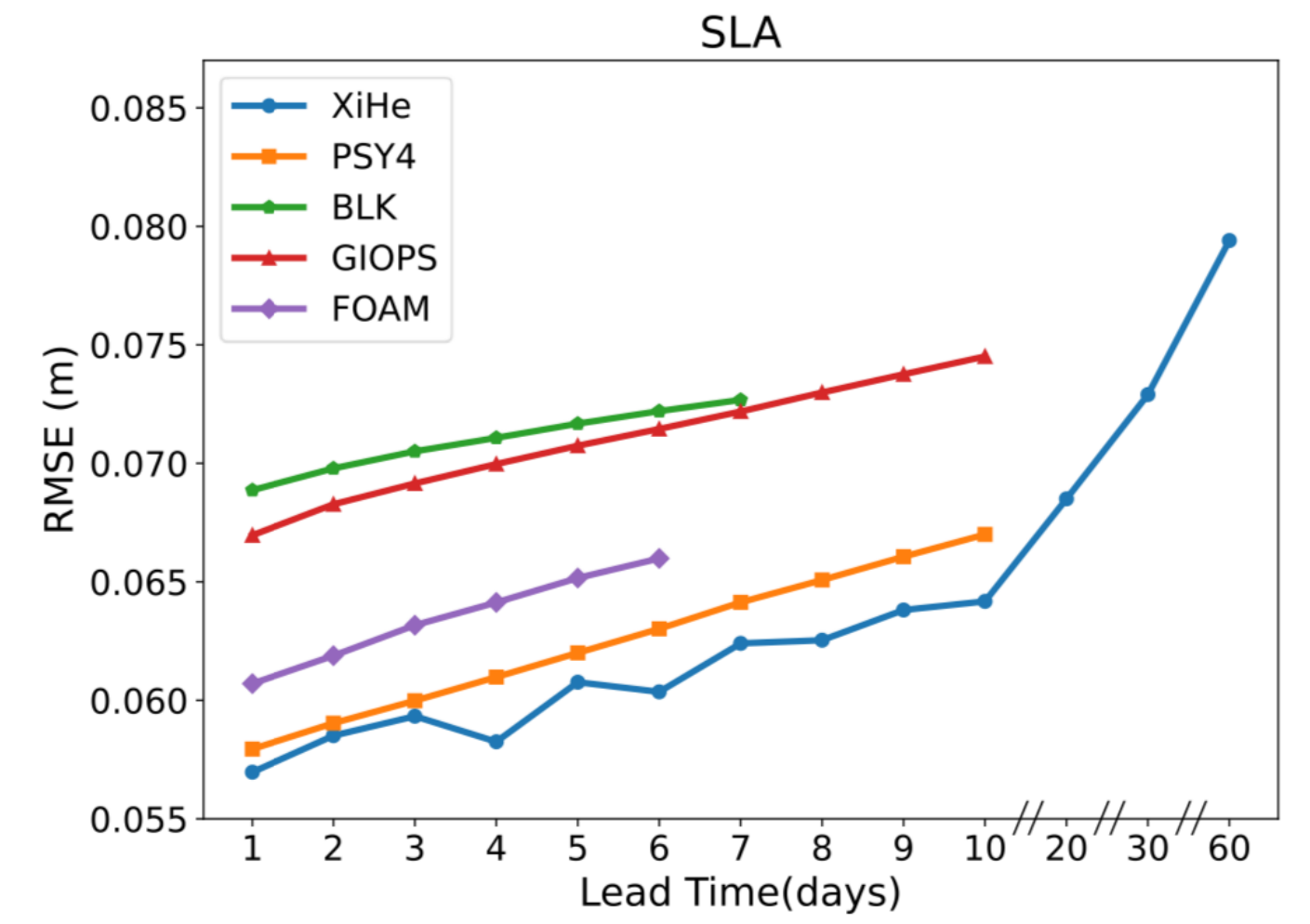
With the recent advances of Artificial Intelligence (AI) techniques, deep learning methods have been widely applied in various prediction/forecasting tasks of different fields and achieved great success. Particularly, some data-driven AI models have shown the potential in atmosphere weather forecasting like *Pangu-Weather* [4] and *GraphCast* [5]. They have achieved comparable or even better prediction results in global medium-range weather forecasting than current leading numerical weather prediction (NWP) methods [4, 5, 6, 7, 8, 9]. One significant advantage of data-driven models is that they can make the forecasting thousands or even tens of thousands of times faster than NWP methods [4]. Furthermore, they can automatically learn the spatial-temporal relationships from massive meteorological data, and effectively capture the rules of weather changing, without introducing the prior knowledge of physics mechanisms.

Although data-driven models have achieved promising results in atmosphere weather forecasting, how to build a more accurate and efficient data-driven ocean forecasting model remains an open research issue due to the following

- Xiang Wang, Pinqiang Wang, Huizan Wang, Junxing Zhu, Jianbo Xu, Senliang Bao, Ciqiang Luo, Yi Han, Weimin Zhang, Kaijun Ren, Kefeng Deng, and Junqiang Song are with the College of Meteorology and Oceanography, National University of Defense Technology, Changsha 410073, China.
- Renzhi Wang, Senzhang Wang and Jun Yin are with the School of Computer Science and Engineering, Central South University, Changsha 410083, China.
- Ningzi Hu is with the College of Oceanography and Space Informatics, China University of Petroleum (East China), Qingdao 266580, China.
- Peng Huo is with the College of Artificial Intelligence, Tianjin University of Science and Technology, Tianjin 300457, China.
- Guihua Wang is with the Department of Atmospheric and Oceanic Sciences, Fudan University, Shanghai 200438, China.
- Ziqing Zu, Key Laboratory of Marine Hazards Forecasting, National Marine Environmental Forecasting Center, Ministry of Natural Resources, Beijing 100081, China.
- Guihua Wang, Huizan Wang, Senzhang Wang, and Weimin Zhang are the corresponding authors.



Trained from ocean reanalyses



Short term forecast skill

<https://arxiv.org/abs/2402.02995>

Wang et al. (2024)

arXiv:2402.02995v2 [physics.ao-ph] 8 Feb 2024

COMING UP

AI-native hybrid geoscientific models

arXiv:2311.07222v3 [physics.ao-ph] 8 Mar 2024

Neural General Circulation Models for Weather and Climate

Dmitrii Kochkov^{1†}, Janni Yuval^{1†}, Ian Langmore^{1†}, Peter Norgaard^{1†}, Jamie Smith^{1†}, Griffin Mooers¹, Milan Klöwer⁴, James Lottes¹, Stephan Rasp¹, Peter Düben³, Sam Hatfield³, Peter Battaglia², Alvaro Sanchez-Gonzalez², Matthew Willson², Michael P. Brenner^{1,5}, Stephan Hoyer^{1†}

¹Google Research, Mountain View, CA.
²Google DeepMind, London, UK.
³European Centre for Medium-Range Weather Forecasts, Reading, UK.
⁴Earth, Atmospheric and Planetary Sciences, Massachusetts Institute of Technology.
⁵School of Engineering and Applied Sciences, Harvard University.

*Corresponding author(s). E-mail(s): dkochkov@google.com; janniyuval@google.com; shoyer@google.com;
†These authors contributed equally to this work.

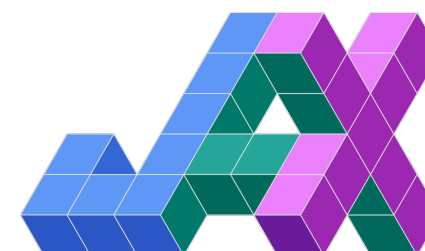
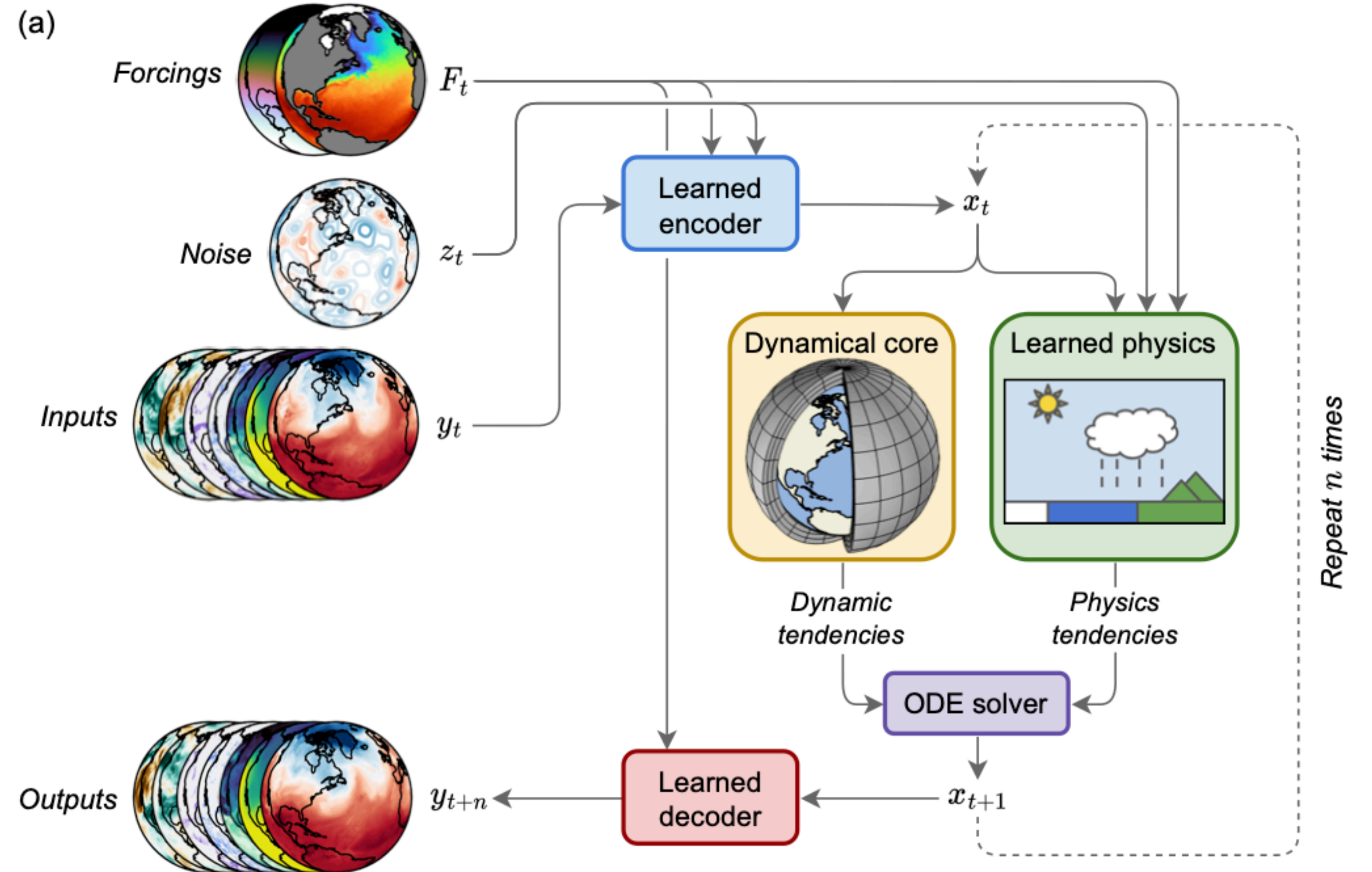
Abstract

General circulation models (GCMs) are the foundation of weather and climate prediction. GCMs are physics-based simulators which combine a numerical solver for large-scale dynamics with tuned representations for small-scale processes such as cloud formation. Recently, machine learning (ML) models trained on reanalysis data achieved comparable or better skill than GCMs for deterministic weather forecasting. However, these models have not demonstrated improved ensemble forecasts, or shown sufficient stability for long-term weather and climate simulations. Here we present the first GCM that combines a differentiable solver for atmospheric dynamics with ML components, and show that it can generate forecasts of deterministic weather, ensemble weather and climate on par with the best ML and physics-based methods. NeuralGCM is competitive with ML models for 1-10 day forecasts, and with the European Centre for Medium-Range Weather Forecasts ensemble prediction for 1-15 day forecasts. With prescribed sea surface temperature, NeuralGCM can accurately track climate metrics such as global mean temperature for multiple decades, and climate forecasts with 140

1

<https://arxiv.org/abs/2311.07222>

Kochkov et al. (2024)



<https://github.com/google-research/dinosaur>

<https://github.com/google-research/neuralgcm>



Hybrid models combining physics and ML

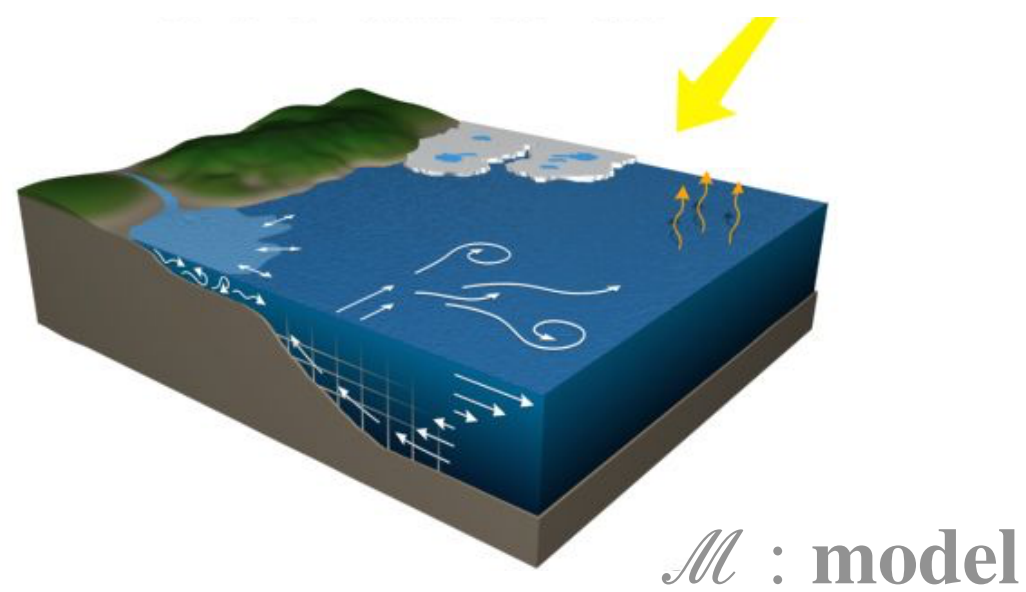


Augmenting ocean models with ML components

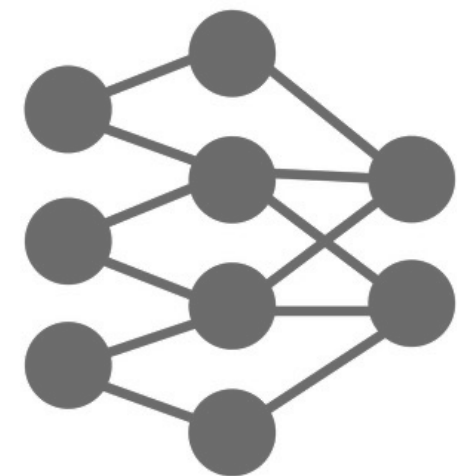
Input



step n

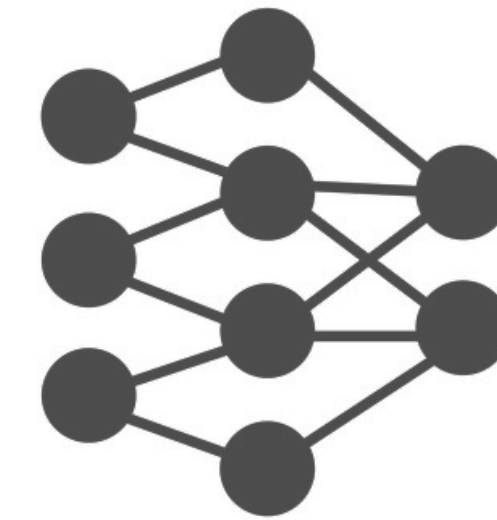


+



θ : parameters

with



θ : parameters

trained to minimise :

$\mathcal{L}(\theta) = \text{training objective}$

- correcting model errors (vs obs.)
- replacing some components (x10-100)
- improving physical consistency
- NB : does not have to be deterministic

Output

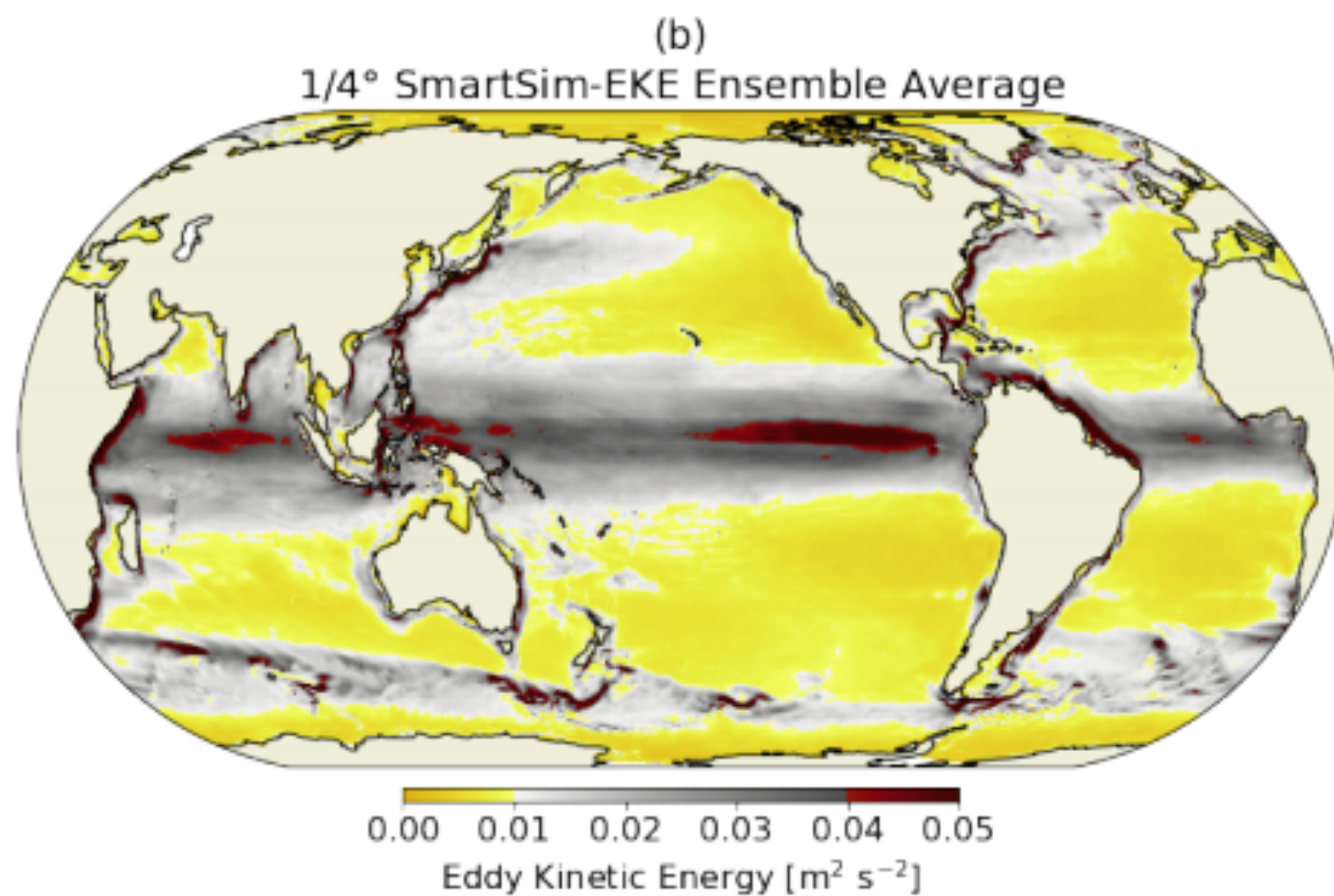
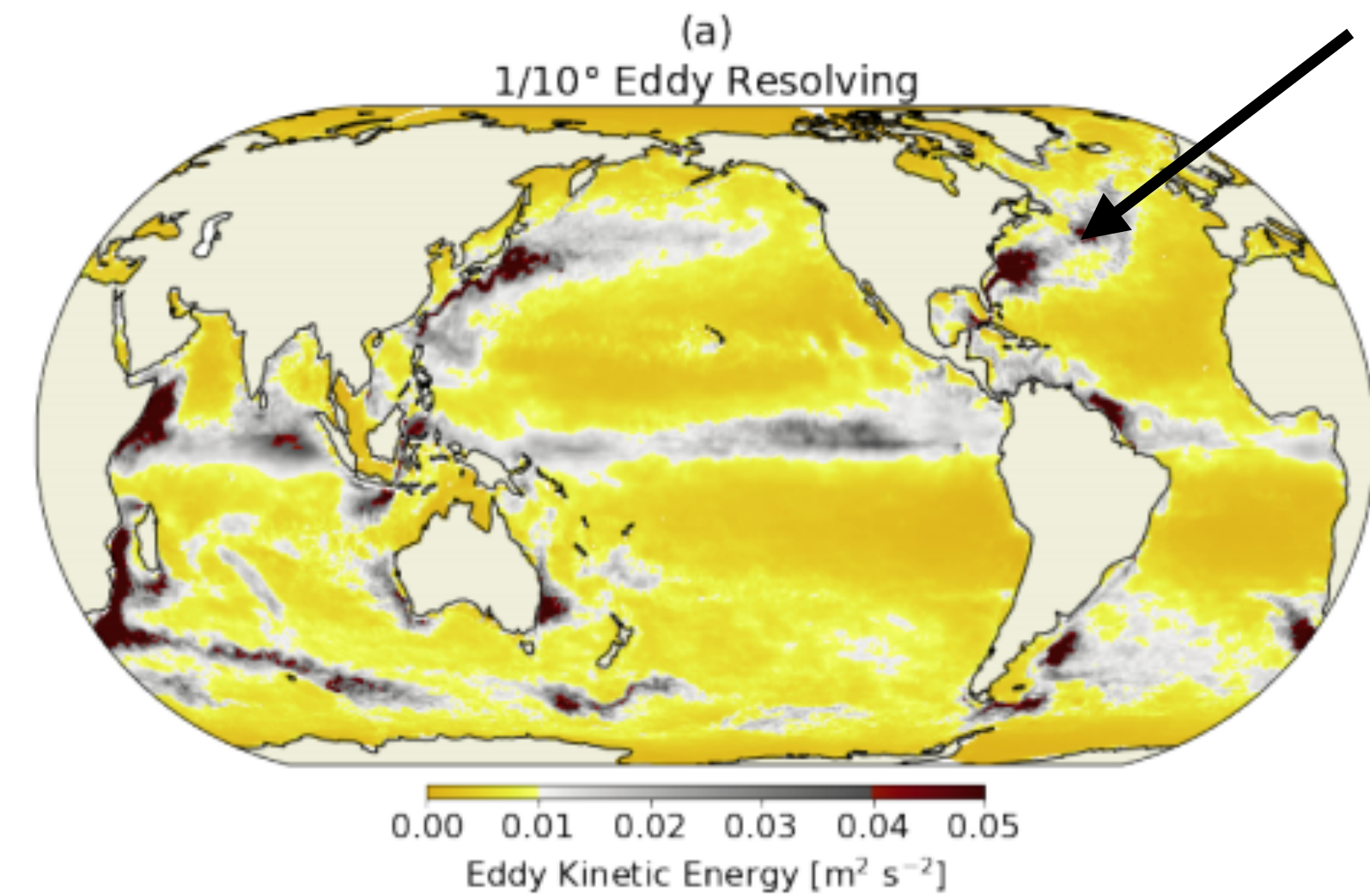
The model is augmented with a **trainable** component

step n+1

ML for ocean models subgrid physics (1/2)

 ocean
macro-turbulence

- missing terms from resolved quantities
- closures for **turbulent processes**
- leveraging **hi-res/process** model data
- encoded as **closed forms** or **ML models**
- a very active field (5-10 papers / months)

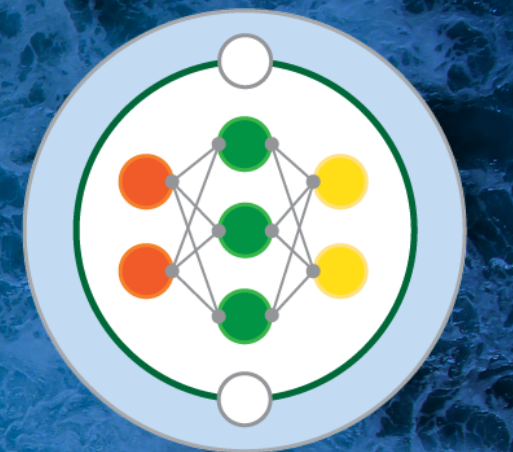


Partee et al. (2022)

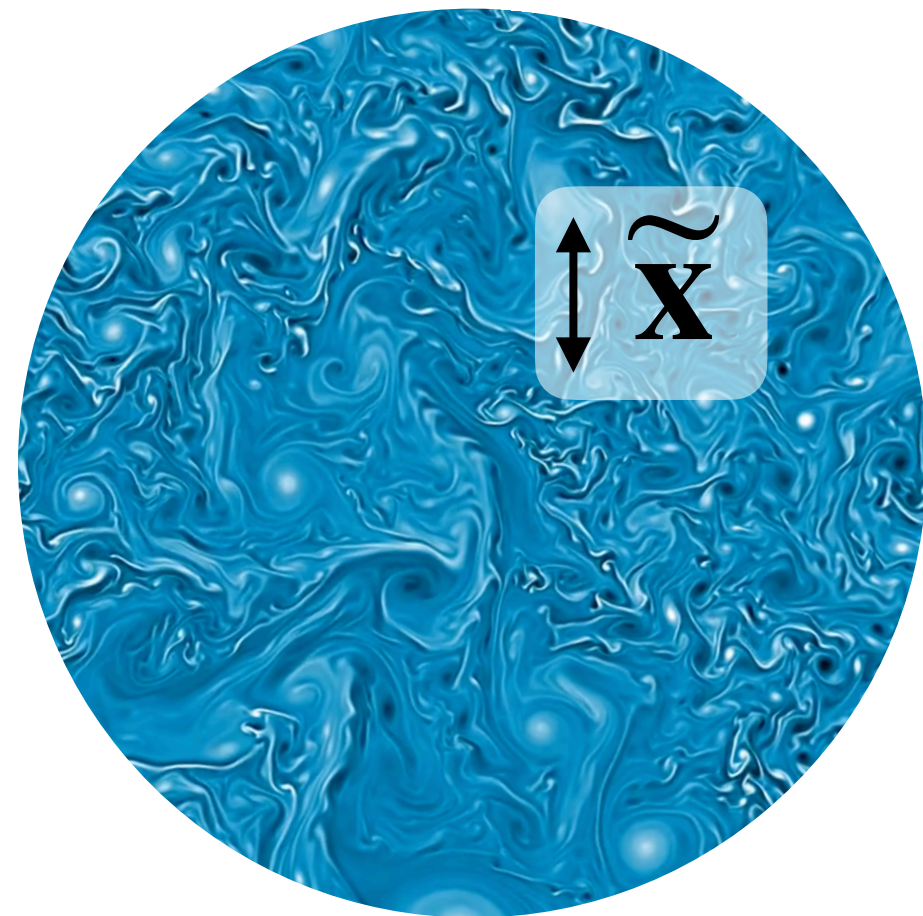
See for instance :
M2LInES consortium

<https://m2lines.github.io>

M²LInES - Multiscale
Machine Learning In
Coupled Earth System
Modeling



ML for ocean models subgrid physics (2/2)

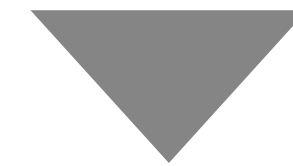


Dynamical system

$$\partial_t \mathbf{x} + \mathcal{L} \mathbf{x} + \mathcal{N}(\mathbf{x}) = 0$$

Resolved equations

$$\partial_t \tilde{\mathbf{x}} + \mathcal{L} \tilde{\mathbf{x}} + \mathcal{N}(\tilde{\mathbf{x}}) = \mathcal{N}(\tilde{\mathbf{x}}) - \overline{\mathcal{N}(\mathbf{x})}$$

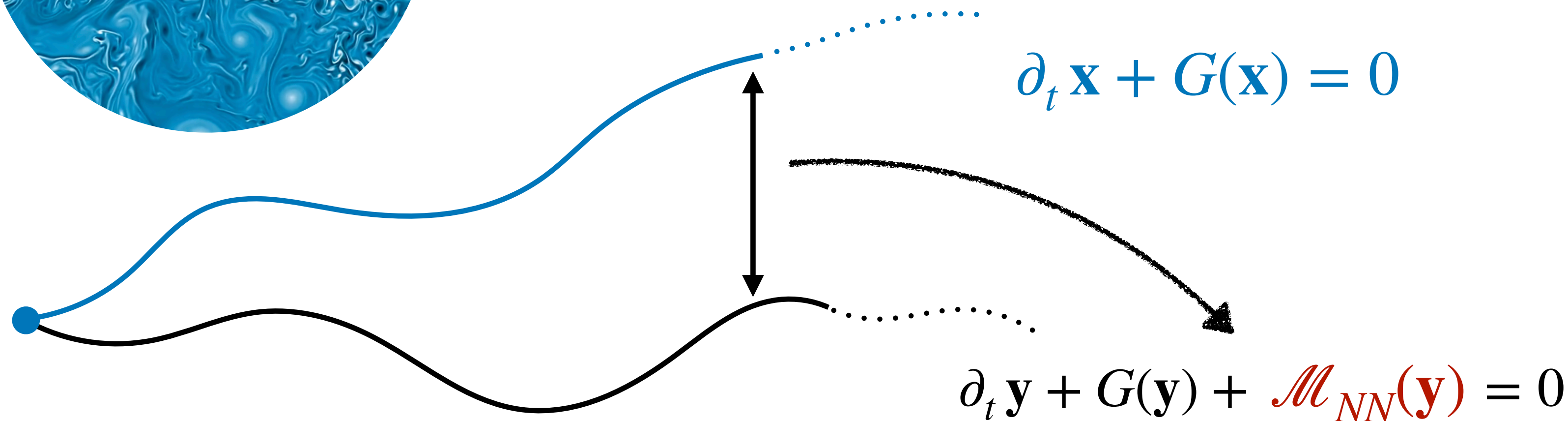


Subgrid closure

$$\mathcal{M}(\tilde{\mathbf{x}}) \simeq \mathcal{N}(\tilde{\mathbf{x}}) - \overline{\mathcal{N}(\mathbf{x})}$$

Learning the mapping

$$\tilde{\mathbf{x}}(t) \rightarrow \mathcal{M}(\tilde{\mathbf{x}}(t))$$



Frezat et al. (2021)

Physical consistency

Symmetries, invariances
loss function / architecture

Frezat et al. (2022)

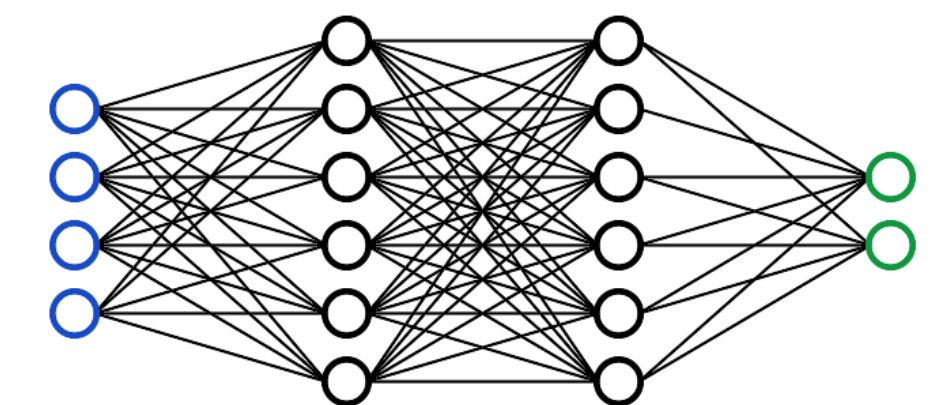
End-to-end training

Differentiable programming,
different loss function
w/ same architecture

Frezat et al. (2023)

Gradient-free training

training model emulator
for approx. gradient
wrt NN. parameters

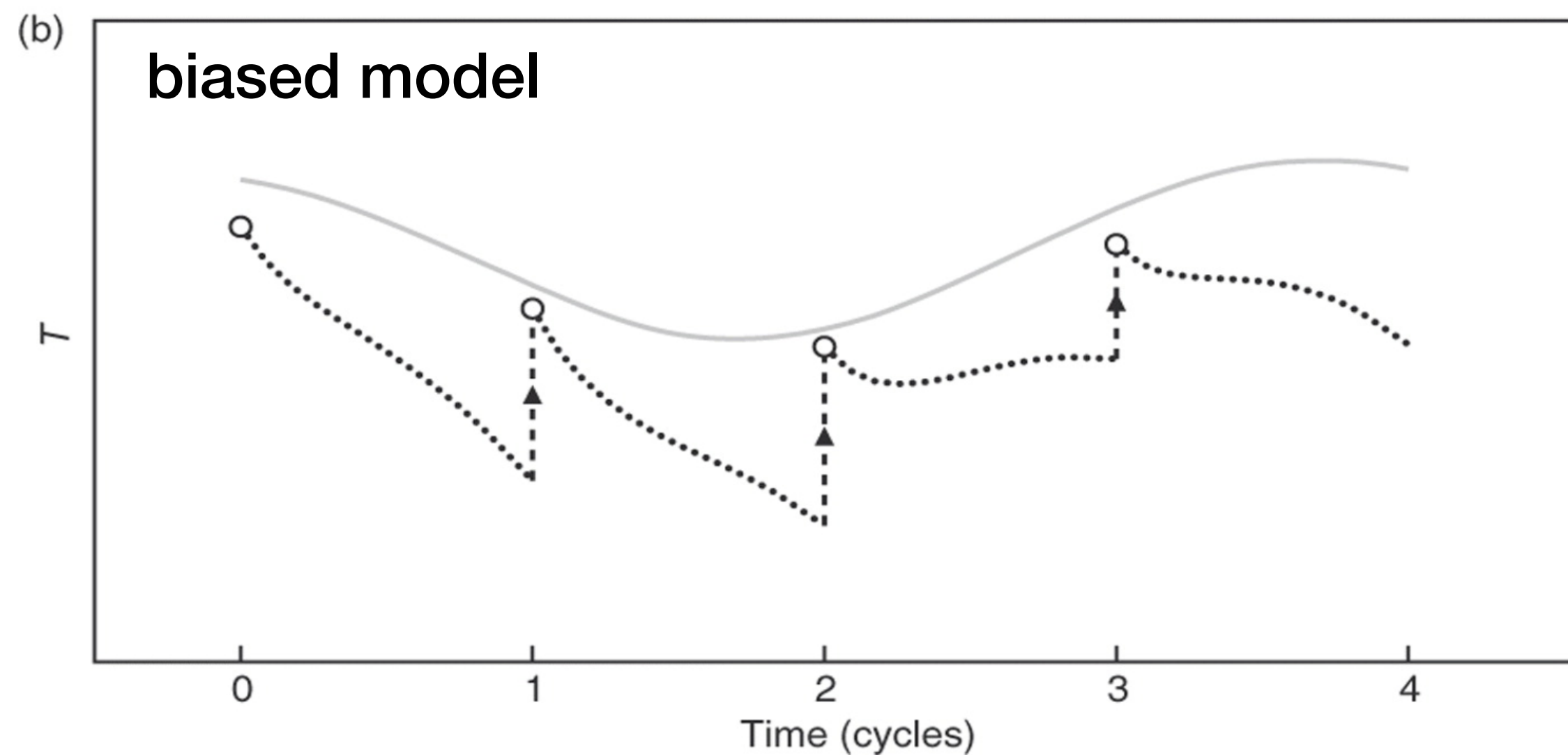
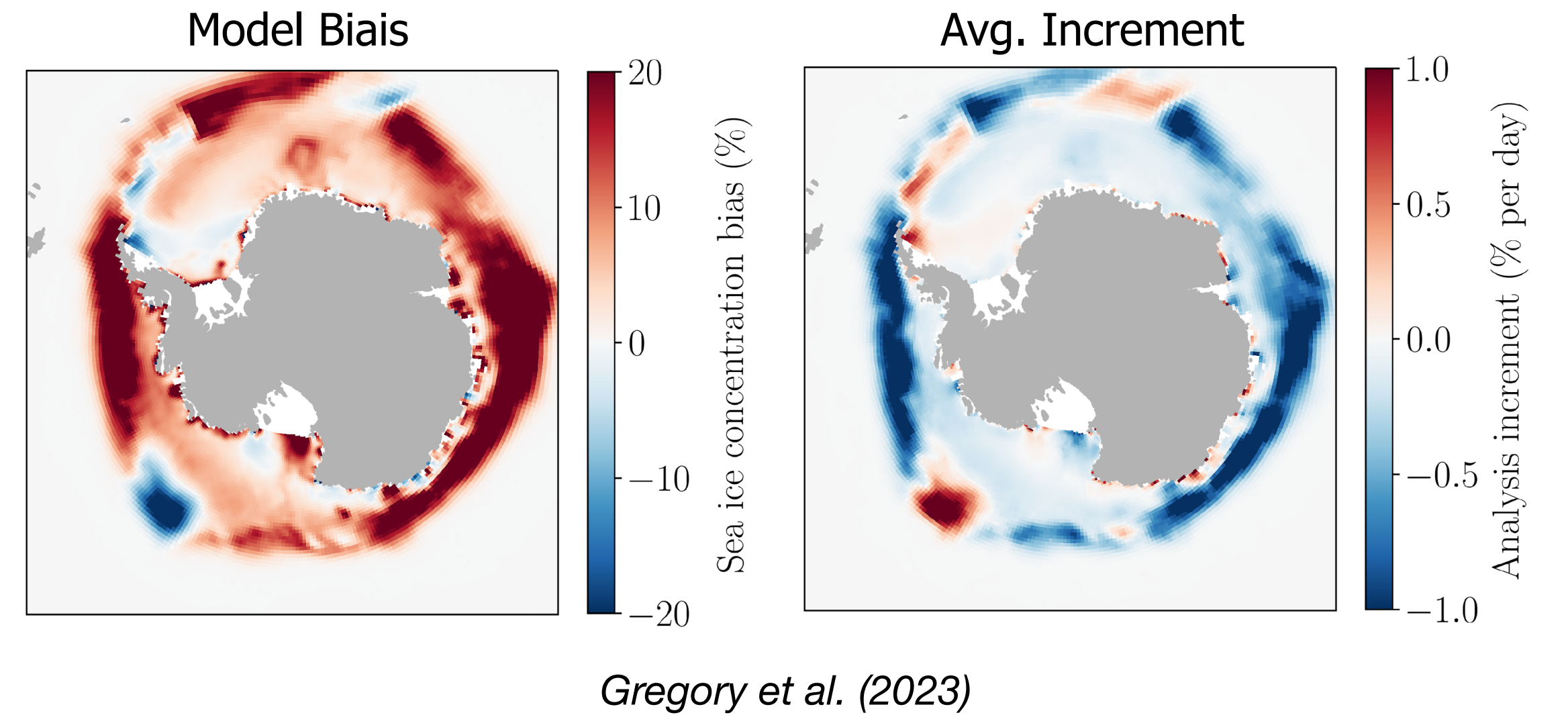
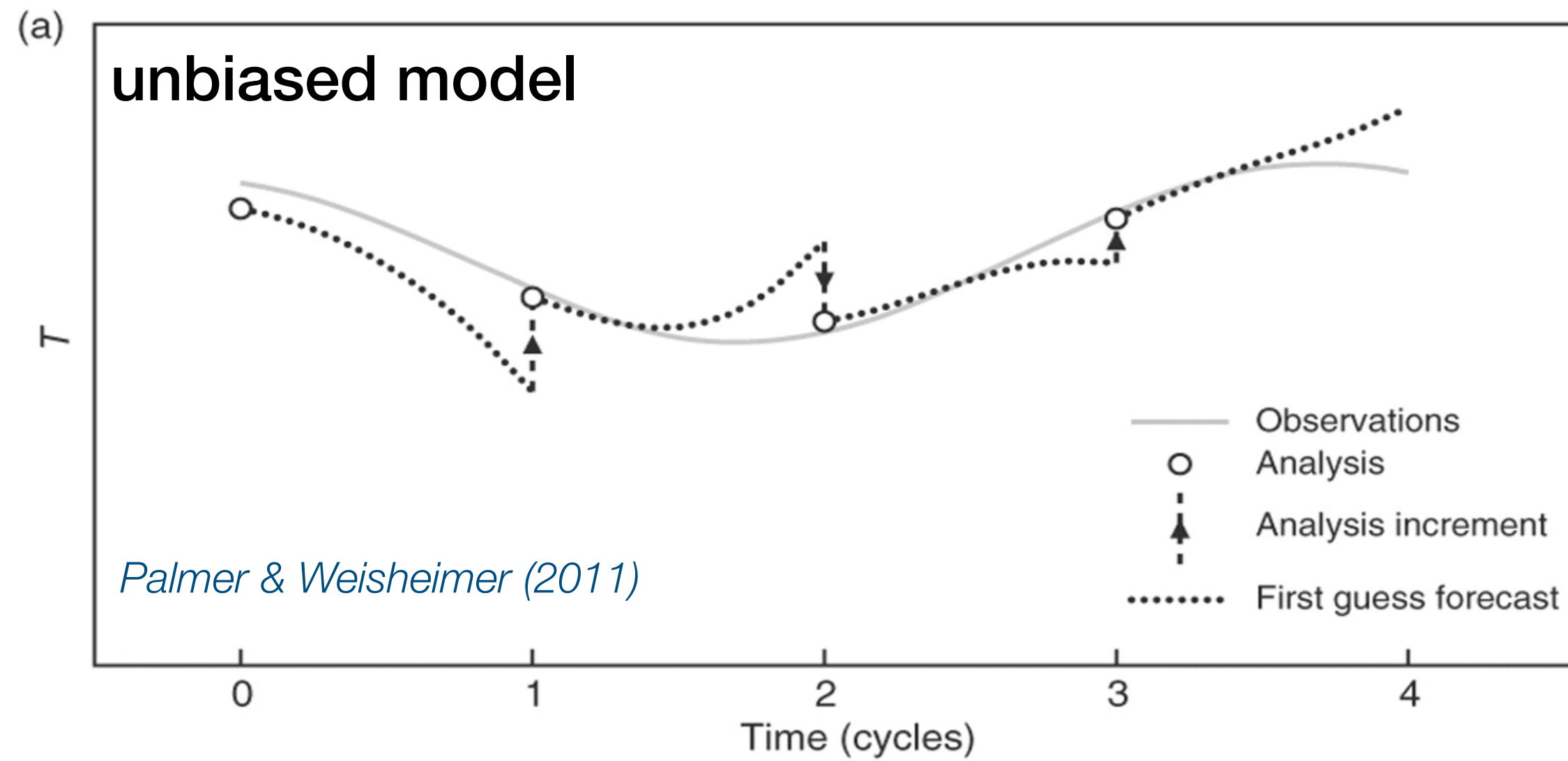


θ : parameters

Performance, stability

Generalisation, interpretability

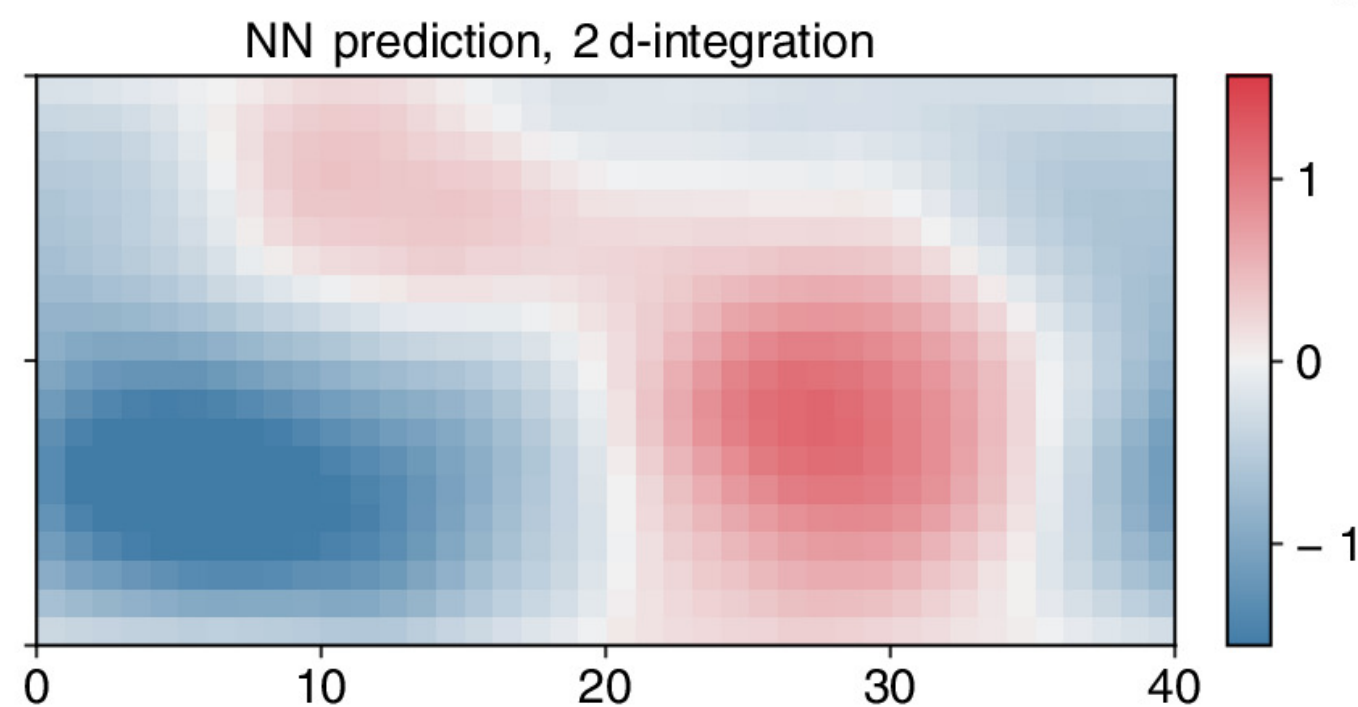
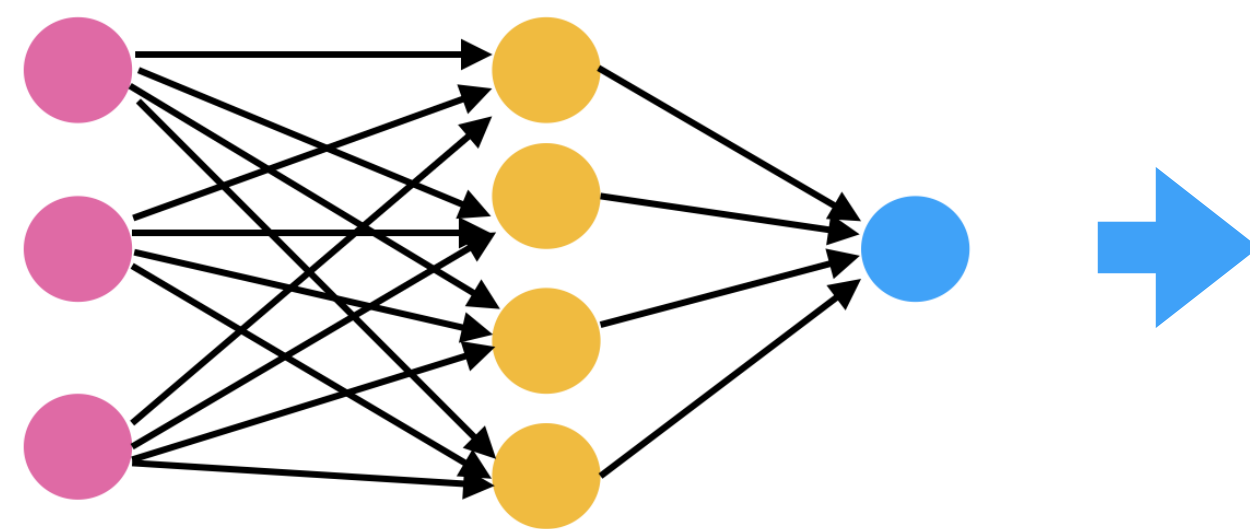
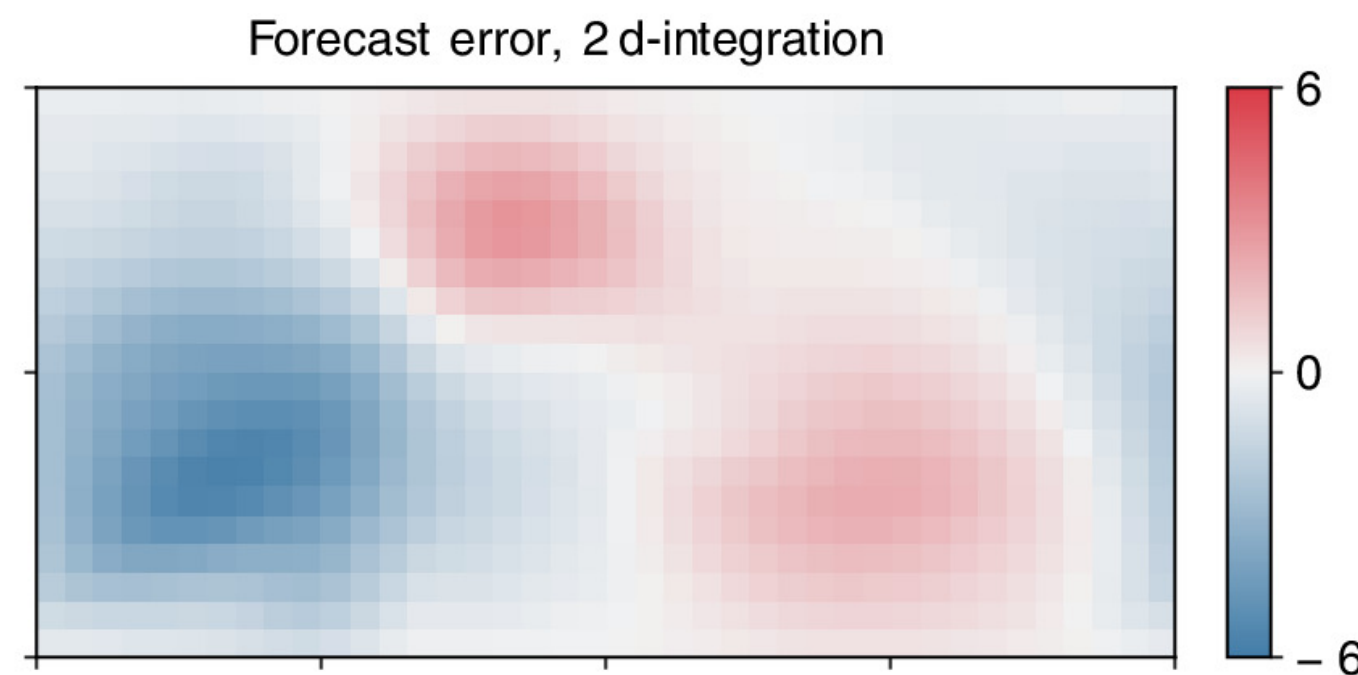
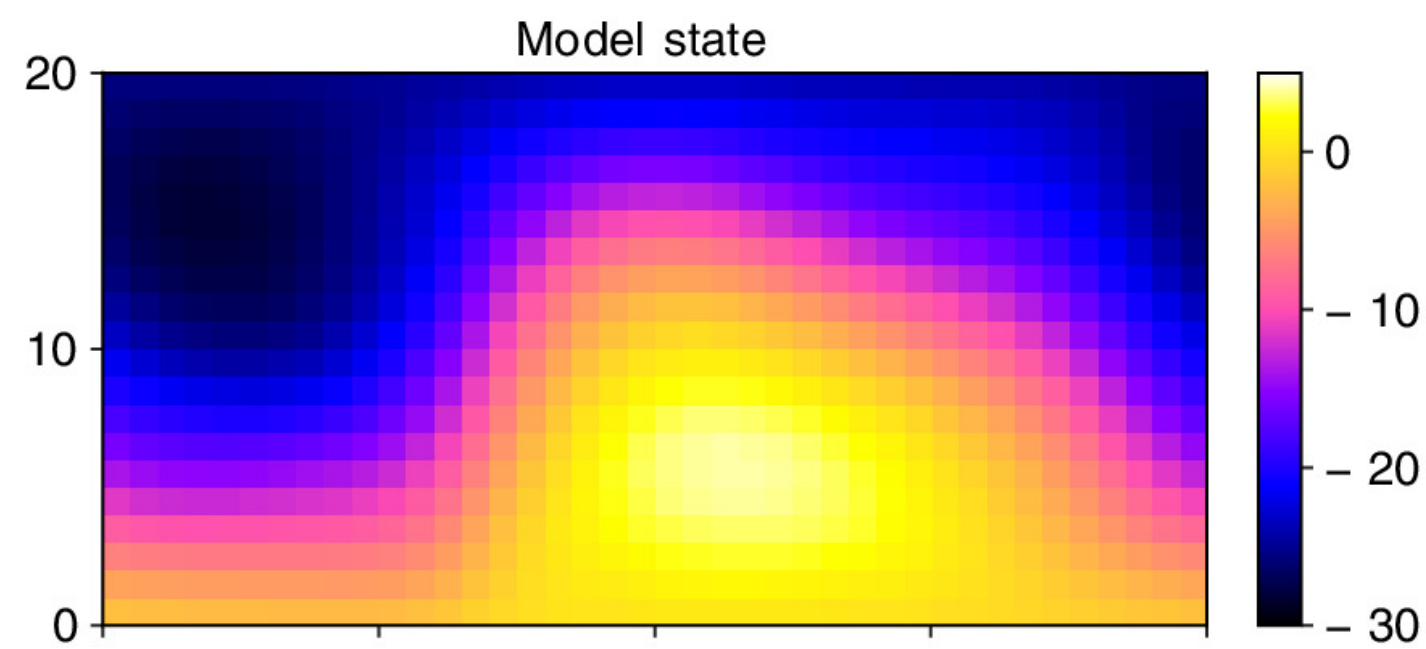
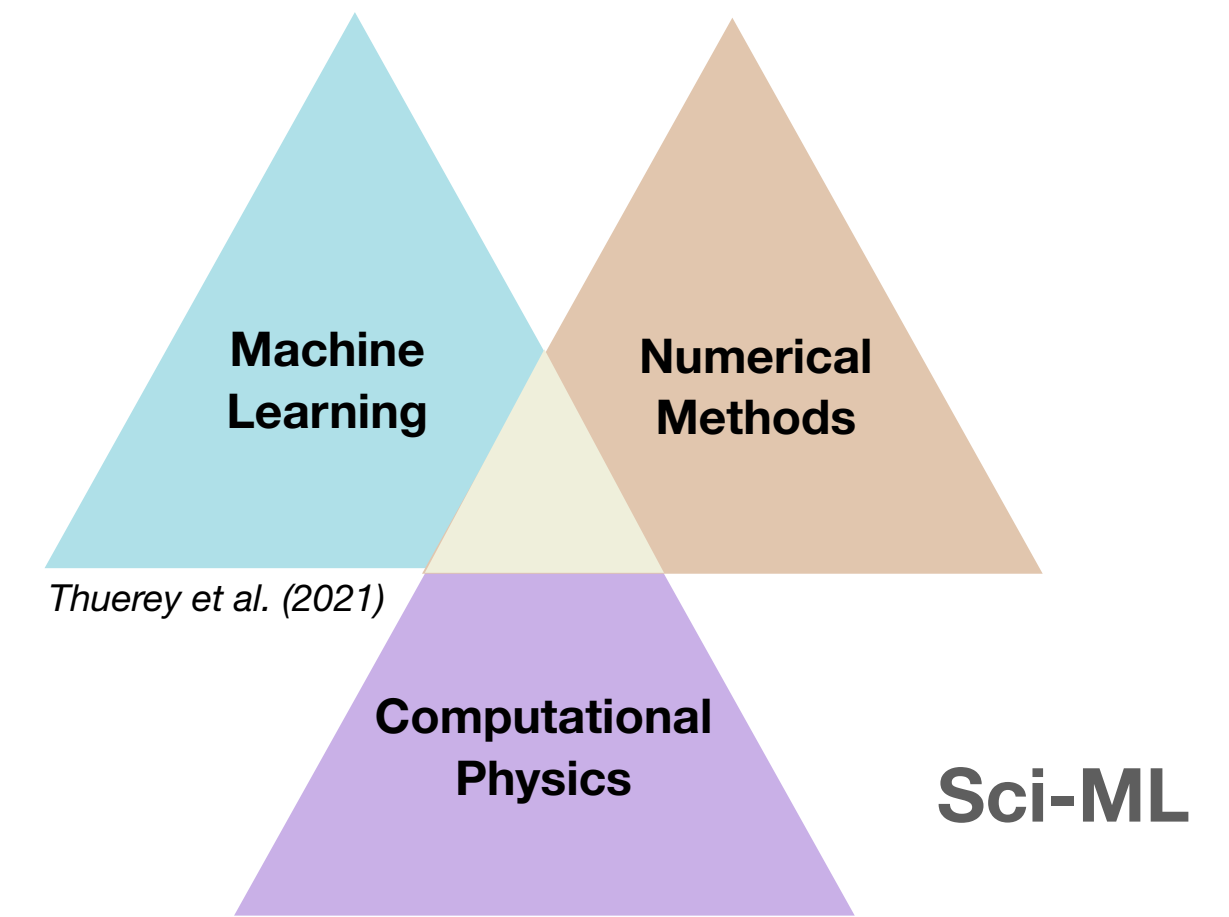
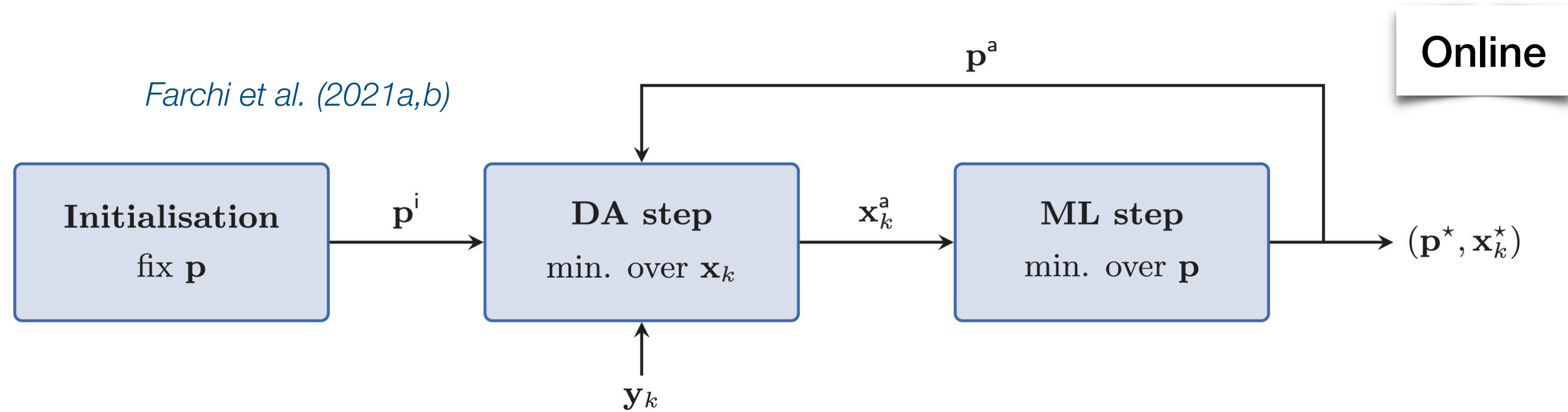
Learning model error from DA increments (1/3)



- w/ unbiased observations, analysis **increments** compensate for model **bias**
- estimating **state-dependent** bias corrections (Leith, 1978; Saha, 1992; DelSole and Hou, 1999)
- state-dependent **bias corrections** provide a representation of model errors

Learning model error from DA increments (2/3)

Farchi et al. (2021a,b)

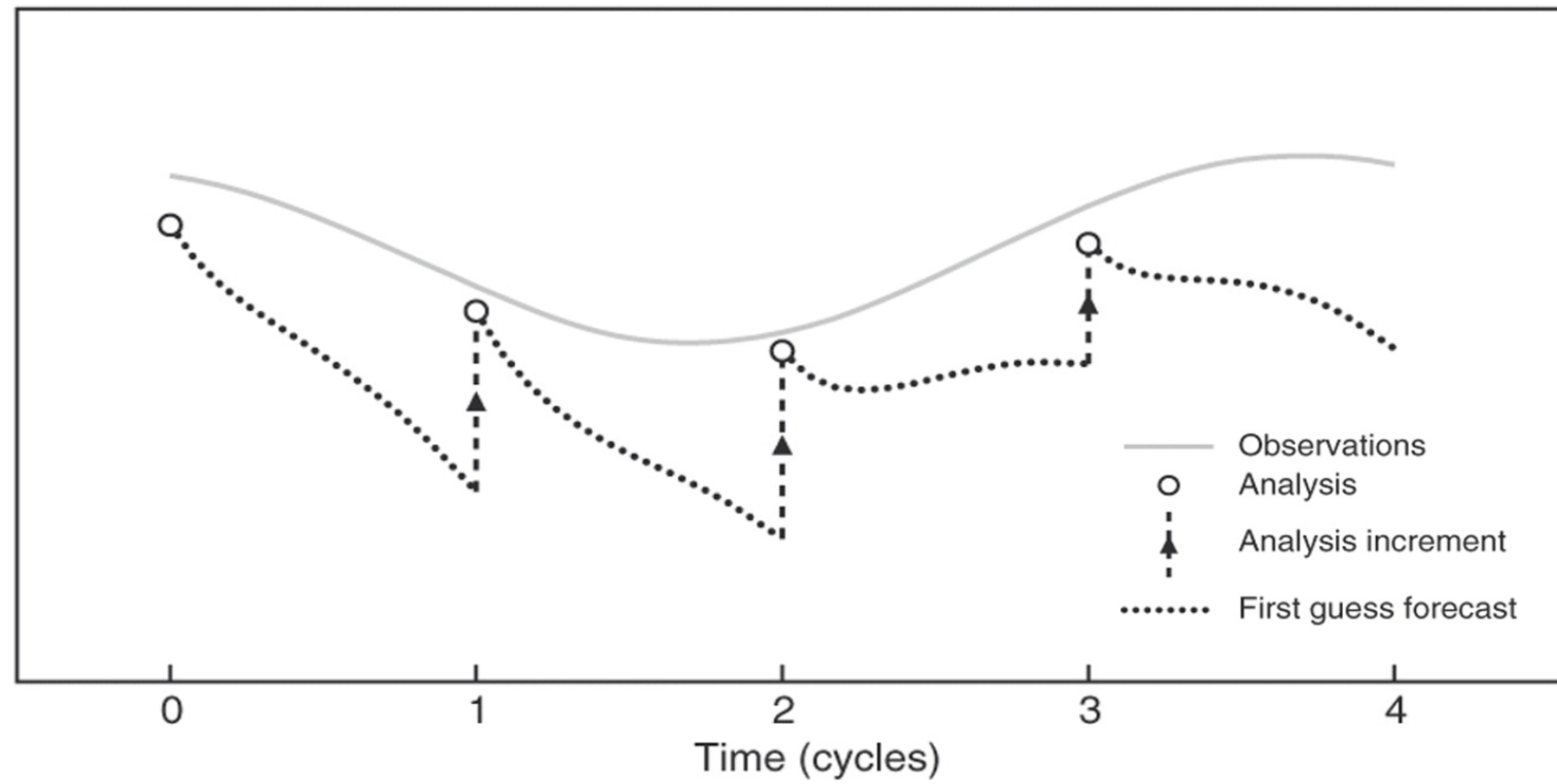


- renewed interest in bias corrections for DA systems over recent years
- joint DA-ML frameworks for estimating model errors (in idealised settings)
- iterative online training of NN with DA

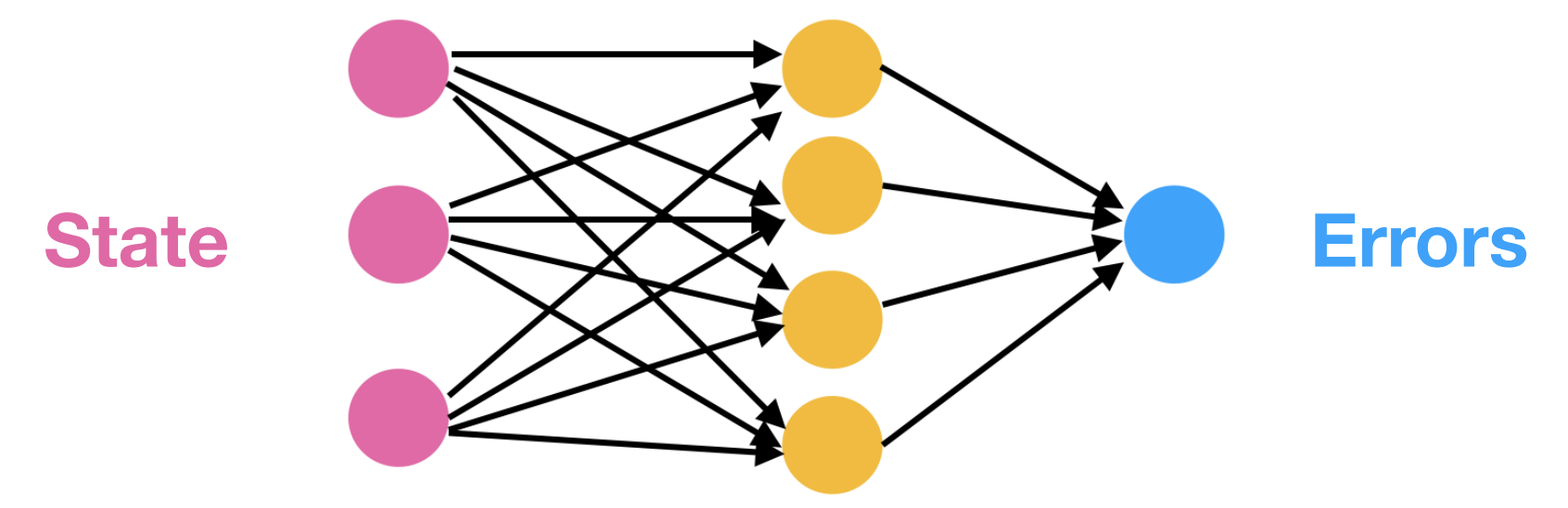
Online estimation of model errors w/ a joint DA-ML 4DVAR (weak)

(Farchi et al. 2021a,b; 2023; Brajard et al. 2021; Frezat et al. 2022)

Learning model error from DA increments (3/3)



Offline



- NN for learning state-dependant biases corrections from analysis increments
- w/ applications in GCMs (atmosphere and ocean/sea-ice)
- showing success in improving the modeled climate state & forecast skill

JAMES | Journal of Advances in Modeling Earth Systems*

RESEARCH ARTICLE
10.1029/2022MS003309

Correcting Systematic and State-Dependent Errors in the NOAA FV3-GFS Using Neural Networks

Tse-Chun Chen^{1,2}, Stephen G. Penny^{1,3}, Jeffrey S. Whitaker¹, Sergey Frolov², Robert Pincus⁴, and Stefan Tulchik^{1,2}

Abstract Weather forecasts made with imperfect models contain state-dependent errors. Data assimilation (DA) partially corrects these errors with new information from observations. As such, the corrections, or “analysis increments,” produced by the DA process embed information about model errors. An attempt is made here to extract that information to improve numerical weather prediction. Neural networks (NNs) are trained to predict corrections to the systematic error in the National Oceanic and Atmospheric Administration’s

JAMES | Journal of Advances in Modeling Earth Systems*

RESEARCH ARTICLE
10.1029/2023MS003757

Deep Learning of Systematic Sea Ice Model Errors From Data Assimilation Increments

William Gregory¹, Mitchell Bushuk², Alistair Adcroft¹, Yongfei Zhang¹, and Laure Zanna³

Abstract Data assimilation is often viewed as a framework for correcting short-term error growth in dynamical climate model forecasts. When viewed on the time scales of climate however, these short-term corrections, or analysis increments, can closely mirror the systematic bias patterns of the dynamical model. In this study, we use convolutional neural networks (CNNs) to learn a mapping from model state variables to analysis increments, in order to showcase the feasibility of a data-driven model parameterization which can predict state-dependent model errors. We undertake this problem using an ice-ocean data assimilation system within the Seamless system for Prediction and Earth system Research (SPEAR) model, developed at the

Ocean/sea-ice reanalyses (increments) will be used for estimating model errors

1. Introduction

Operational numerical weather prediction (NWP) models are inherently imperfect. Systematic errors result from approximations in deriving the governing equations, from their numerical implementation, and from conceptual and numerical errors in the parameterizations that represent subgrid scale physical and dynamical processes. Even small errors in any component of the NWP model can compound over time to produce errors that significantly degrade the forecasting skill.

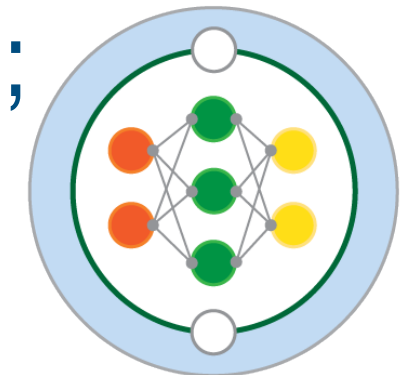
Systematic errors can be addressed with a wide range of approaches. One approach is to improve the model components—the dynamical core and subgrid scale physics parameterizations. The forecast system as a whole can be improved, say by adopting stochastic parameterizations that account for uncertainty, or by increasing spatial resolution. Model forecasts can also be further improved by an “offline” post-processing using statistical methods (e.g., Model Output Statistics) or machine learning (ML) methods applied to the model output after the completion of model forecast. However, the model errors may be convoluted over time and become more nonlinear as forecast progresses, leading to errors that are more difficult to represent.

CHEN ET AL. 1 of 17

1. Introduction

The influence of structural errors within climate models due to missing physics, imperfect parameterizations of subgrid-scale processes, as well as errors in the underlying numerics, leads to systematic biases across the atmosphere, land, sea ice, and ocean. Subsequently, our ability to diagnose and correct these biases ultimately governs the accuracy of numerical weather and climate predictions on different time scales (Stevens & Bonville, 2013). In the context of sea ice for example, much effort has been afforded to the improvement of model physics and subgrid parameterizations through the development of, for example, ice thickness distribution (Bitz et al., 2001; Thorndike et al., 1975) and floe-size distribution theory (Horvat & Tripperman, 2015; Rothrock & Thorndike, 1984), surface melt-pond (Flocco et al., 2012), ice drift (Tsamados et al., 2013) and lateral melt parameterizations (M. Smith et al., 2022), as well as sea ice rheology (Dunserau et al., 2016; Hibler, 1979; Ólason et al., 2022). Such studies have shown how the improved representation of sea ice physics produces model simulations which more closely reflect observations in terms of either their mean sea ice volume, drift, or ice thickness distribution. Despite this,

GREGORY ET AL. 1 of 23



(Bonavita and Laloyaux, 2020; Watt-Meyer et al., 2021; Chen et al., 2022; Gregory et al. 2023; Chapman and Berner 2023)



Hybrid modelling with existing codes

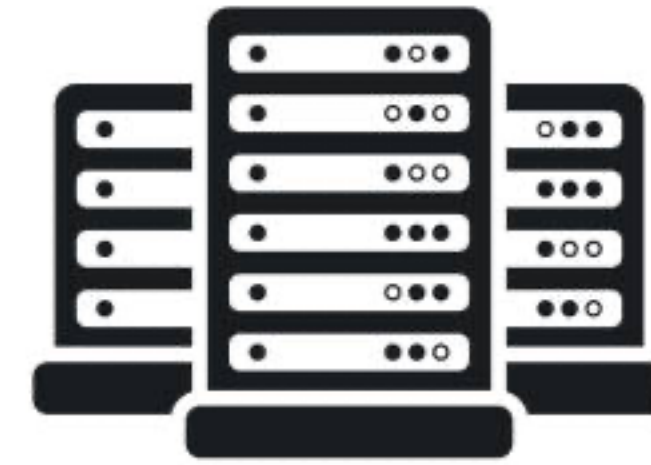


Interfacing ocean models with DL frameworks (1/3)

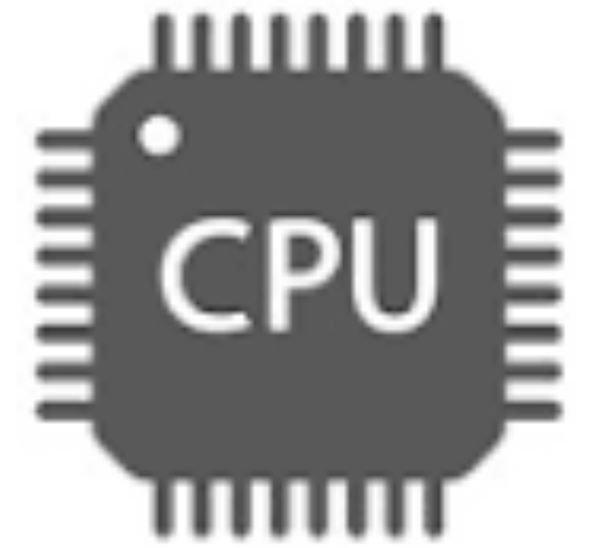


stable, robust, low abstraction languages

+



supercomputers



runs only on CPUs

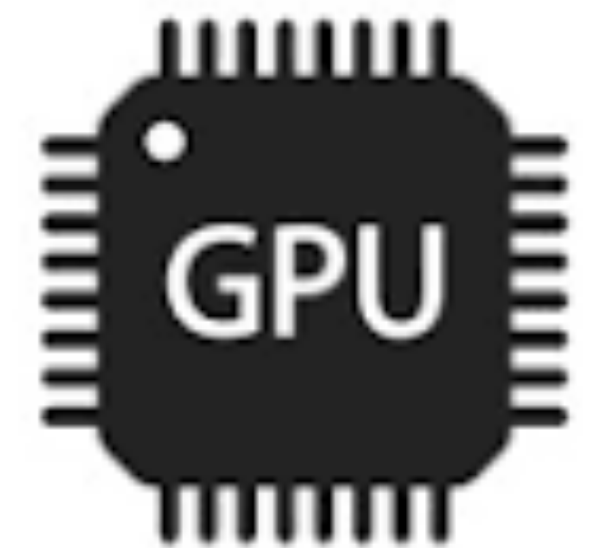


high abstraction, fast evolving languages

+

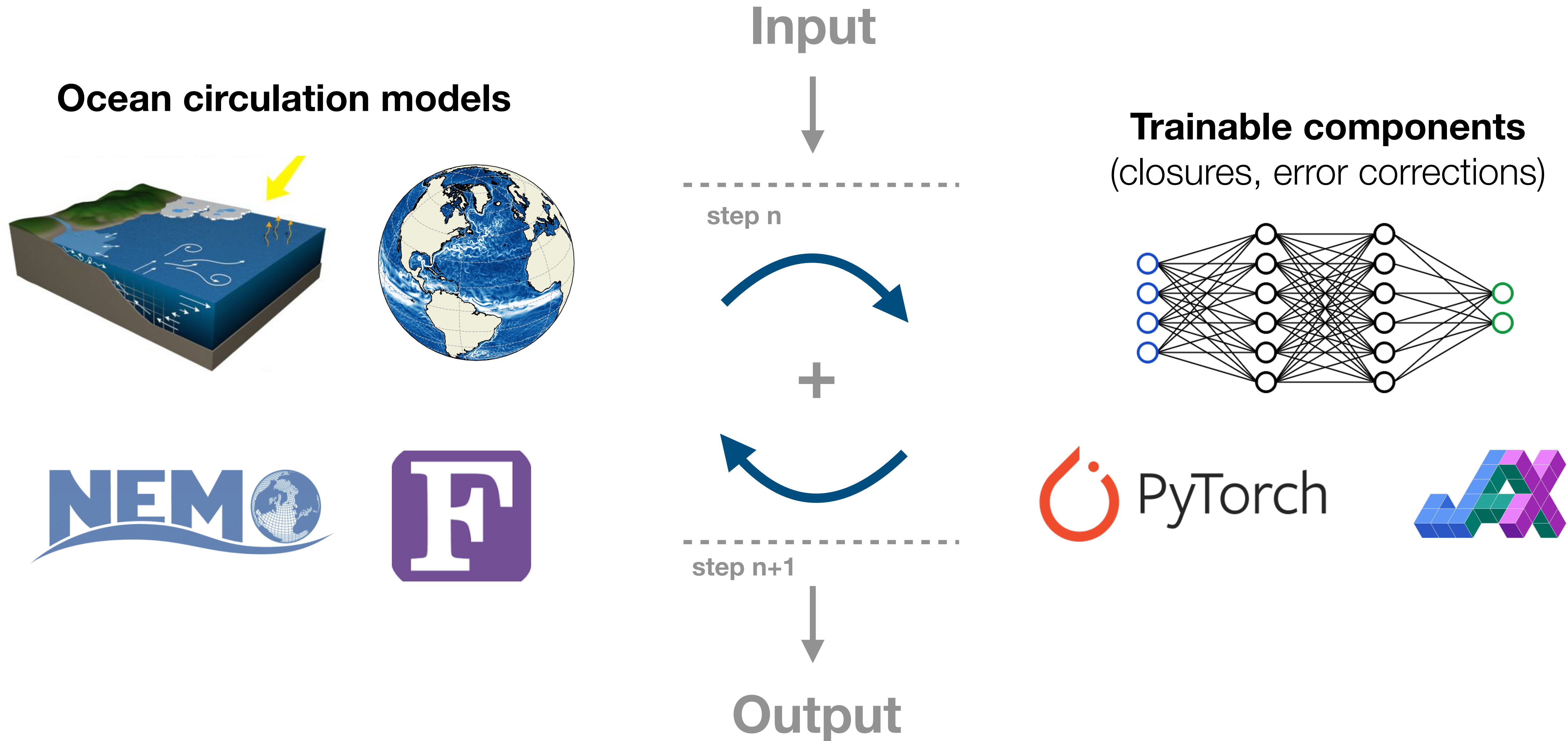


cloud ready

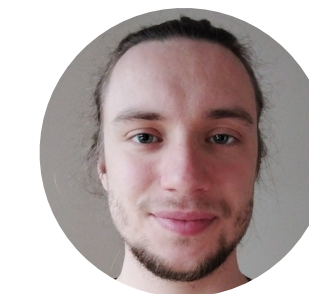
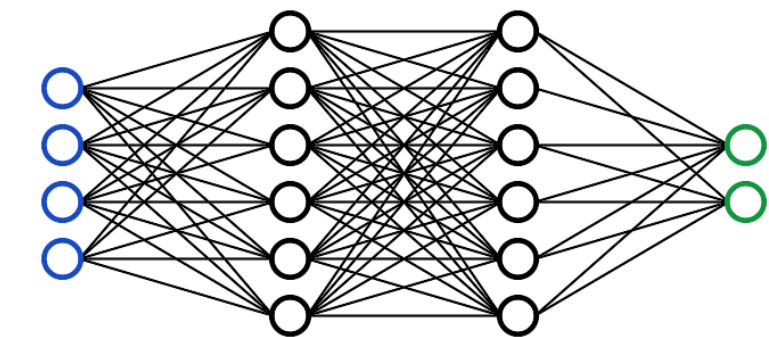
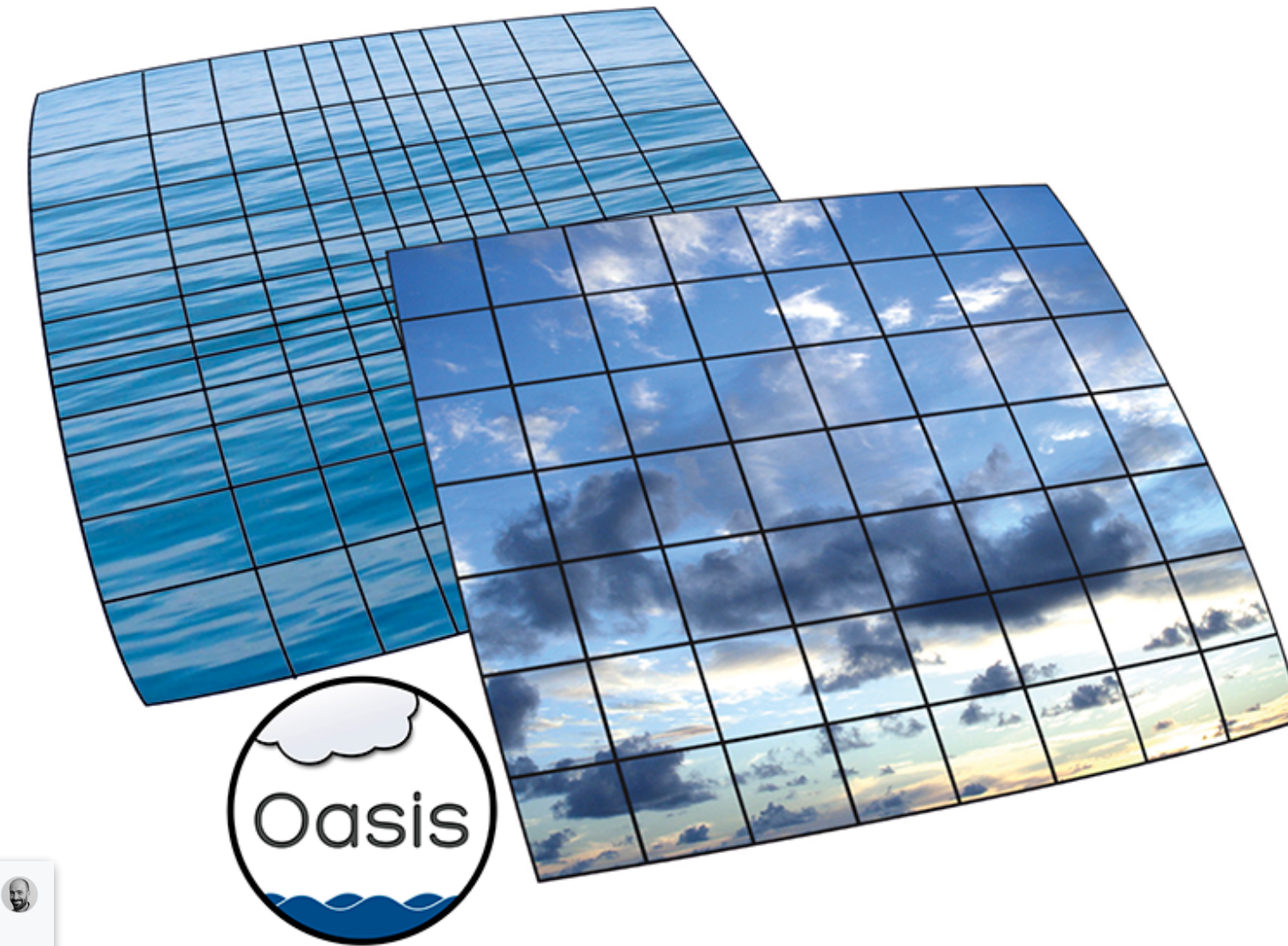
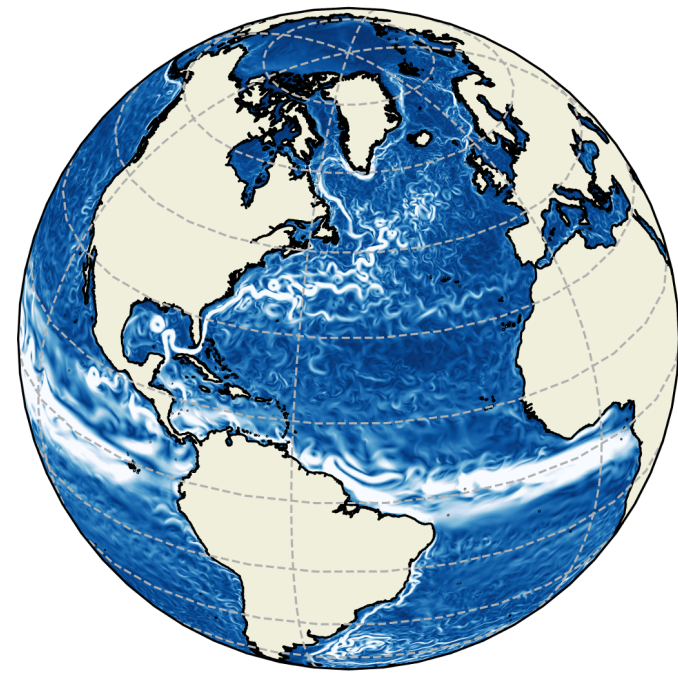


natively runs on GPUs

Interfacing ocean models with DL frameworks (2/3)

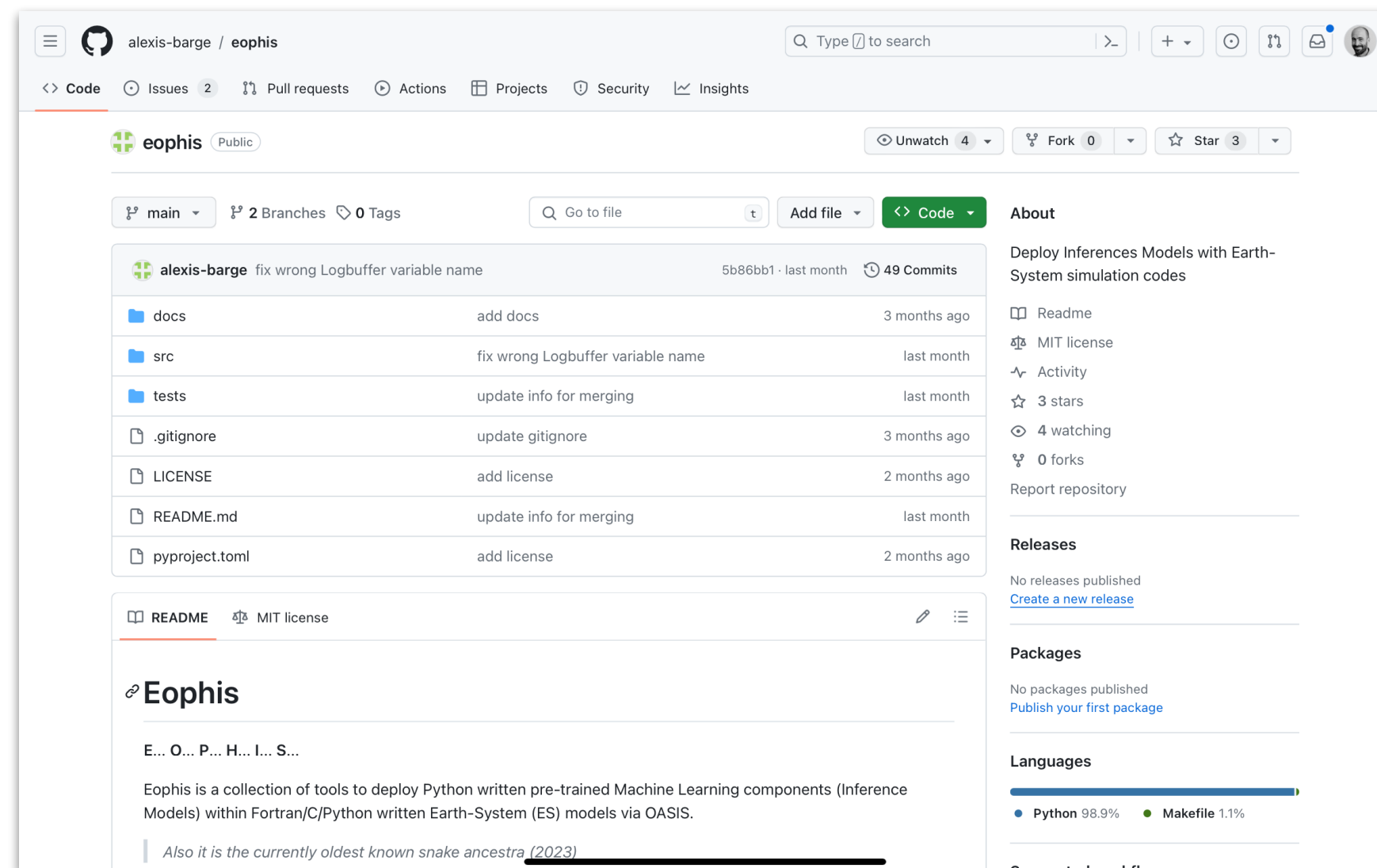


Interfacing ocean models with DL frameworks (3/3)



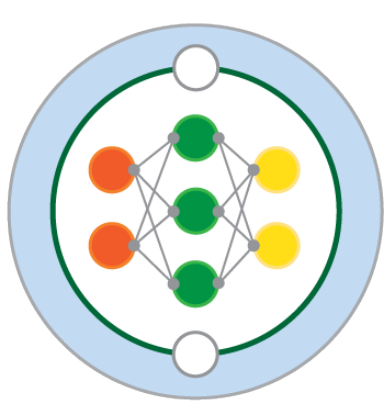
A. Barge

Work by Alexis Barge at IGE



- OASIS : exchange of 3D data between different codes
- Eophis : simplified deployment of ML models w/ OASIS
- Requires some change to the NEMO code
- Key : portability, domain decomposition

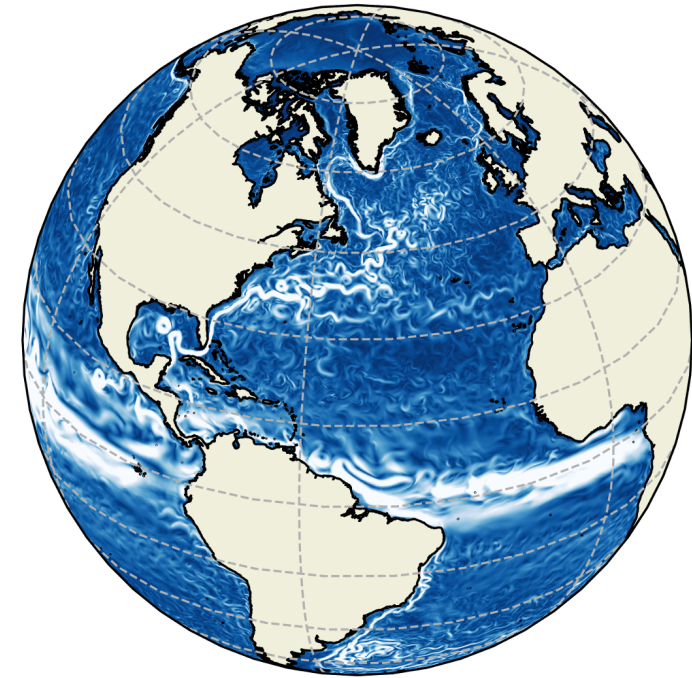
<https://github.com/meom-group/eophis>



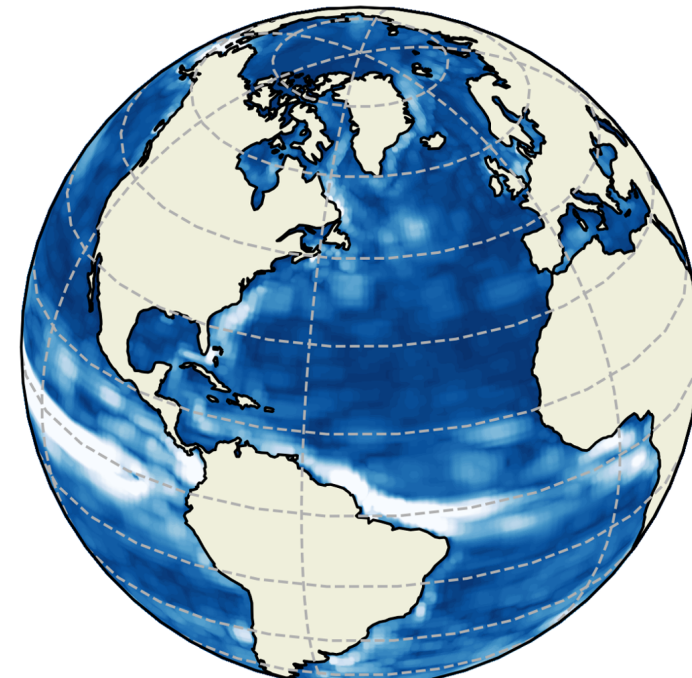
ML for ocean macro-turbulence



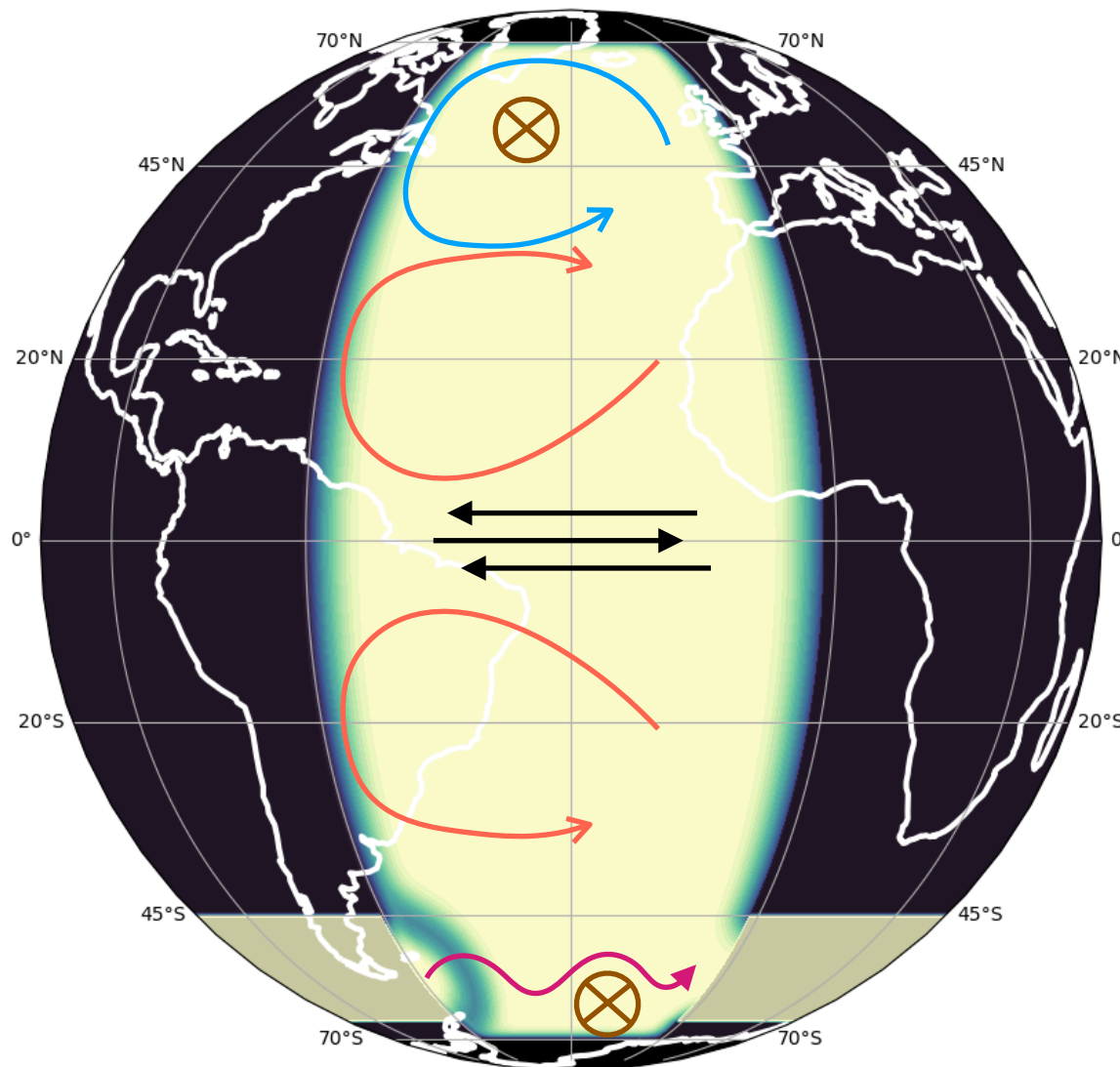
Target NEMO configurations



NEMO-eORCA025



NEMO-eORCA1



DINO : Diabatic Neverworld

Light-weight test-bed

resolutions $\sim 1^\circ, 1/4^\circ, 1\text{km}$

D. Kamm
E. Meunier
J. Deshayes

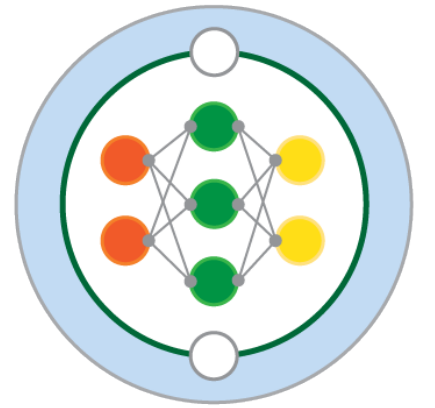


On-going work :

Subgrid momentum forcing due to mesoscale eddies

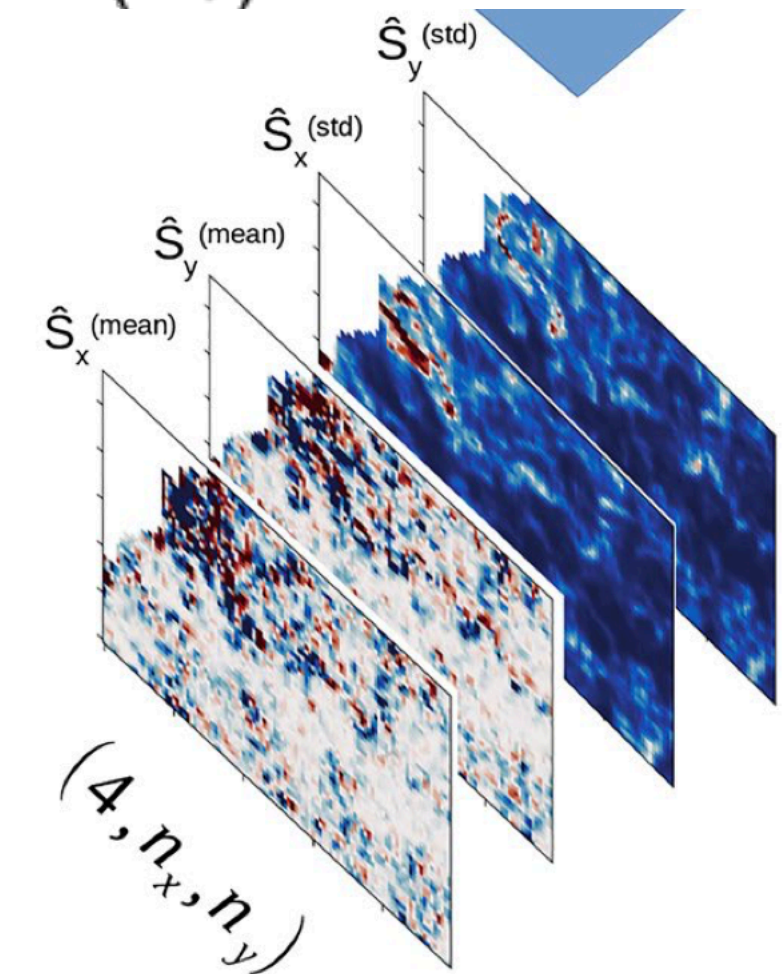
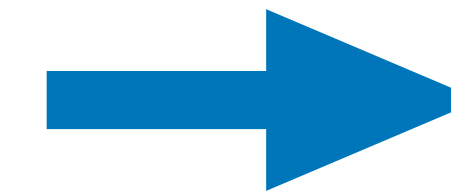
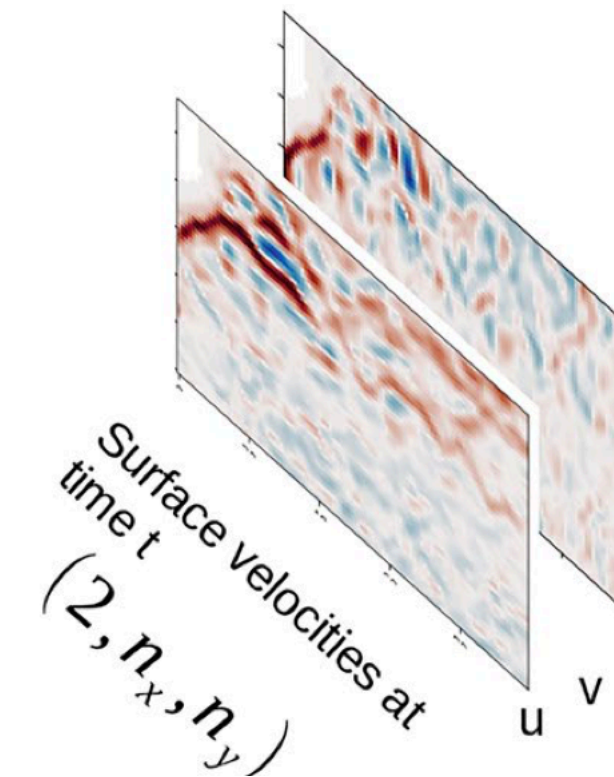
Work by D. Kamm, E. Meunier, A. Barge

Collaboration through M2LINES project



$$\frac{\partial \bar{\mathbf{u}}_k}{\partial t} + \frac{f + \bar{\zeta}_k}{\bar{h}_k} \hat{\mathbf{z}} \times \bar{h}_k \bar{\mathbf{u}}_k + \nabla \bar{K}_k + \nabla \bar{M}_k = \bar{\mathbf{F}}_k + \mathbf{S}_k$$

$$\mathbf{S}_k = \begin{pmatrix} S_{kx} \\ S_{ky} \end{pmatrix} = (\bar{\mathbf{u}}_k \cdot \nabla) \bar{\mathbf{u}}_k - \overline{(\mathbf{u}_k \cdot \nabla) \mathbf{u}_k}$$

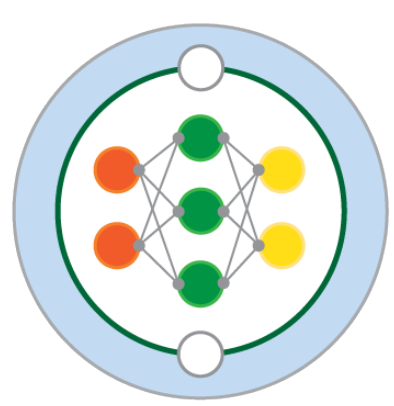


Guillaumin and Zanna 2021

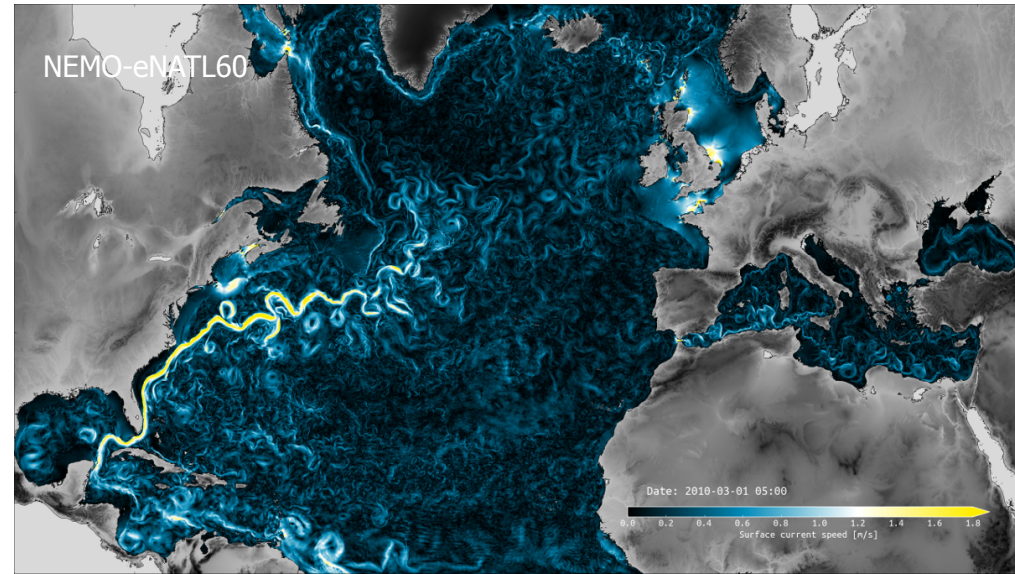
Zhang et al. 2023

<https://doi.org/10.1029/2021MS002534>

<https://doi.org/10.1029/2023MS003697>



Training from realistic ocean models

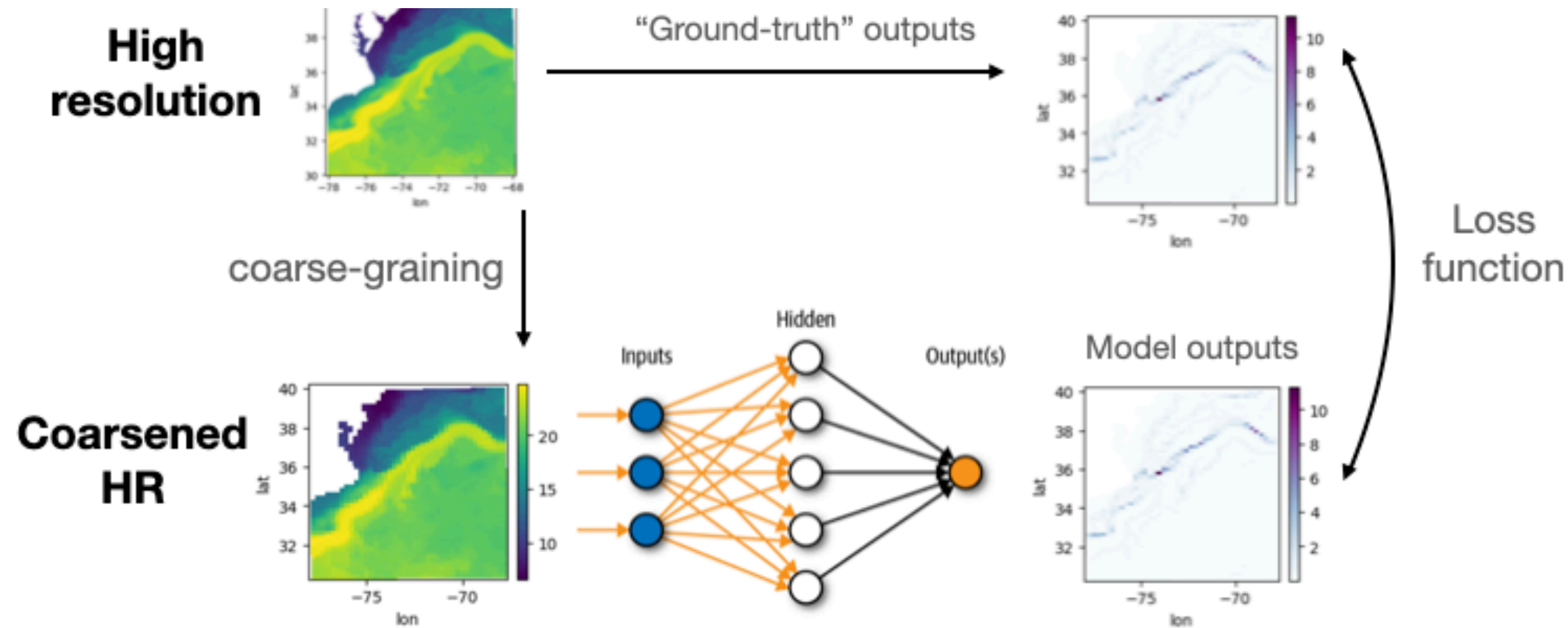
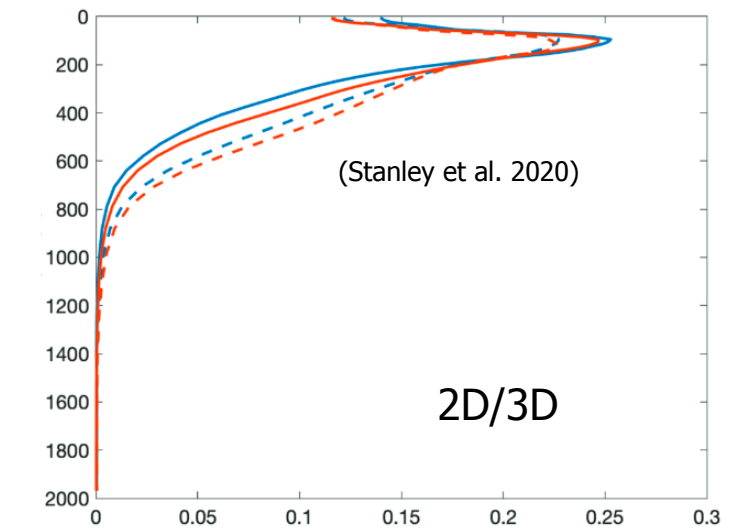


- geometry, grid, coastlines
- heterogeneous regimes
- optimal coarse graining
- data structure / orchestration

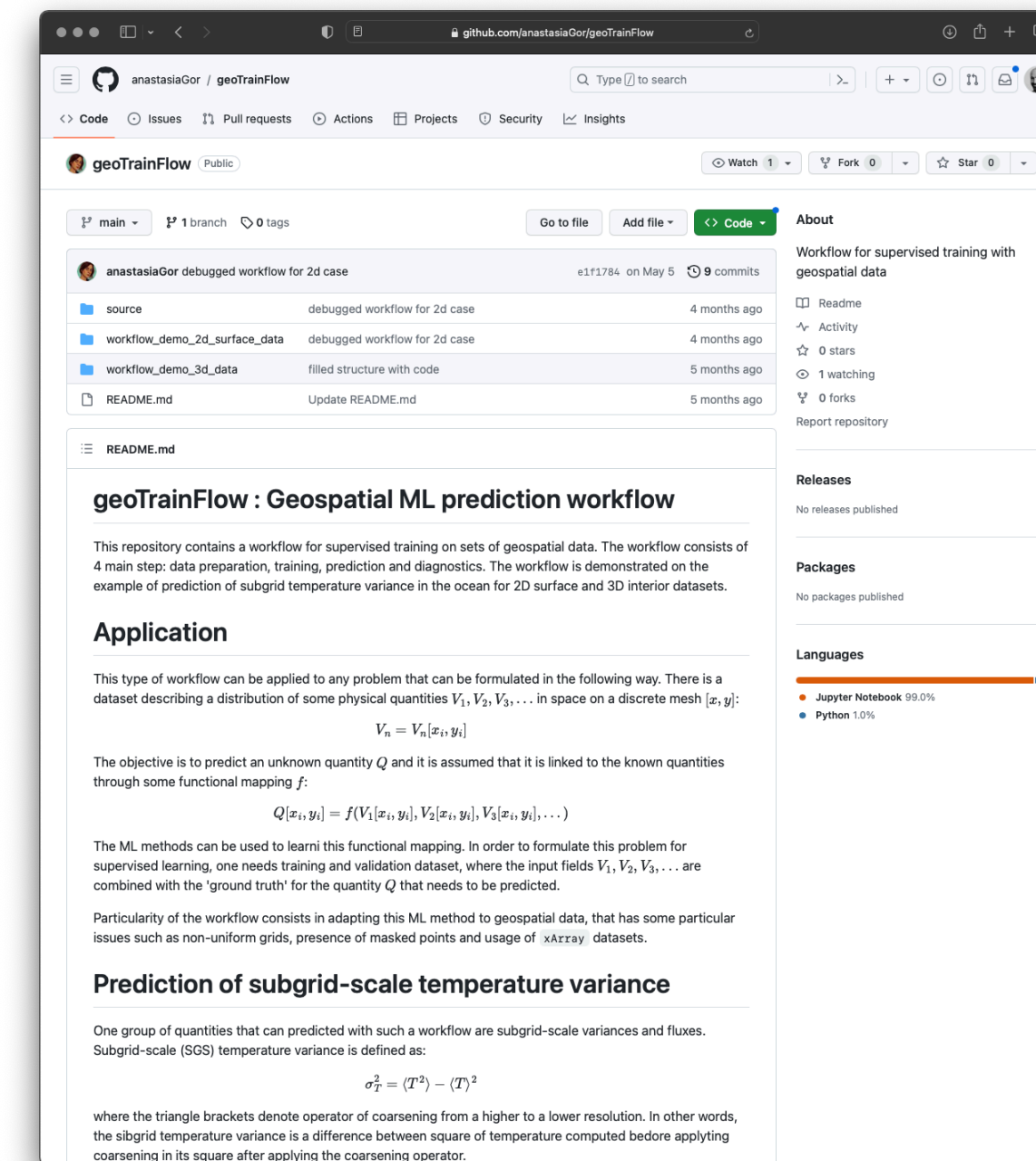
sub-mesoscale
temperature variance

$$\sigma_T^2 = \langle T^2 \rangle - \langle T \rangle^2$$

w/ simple baseline (eq. state)



Problem formulation



GeoTrainFlow training pipeline



A. Gorbunova



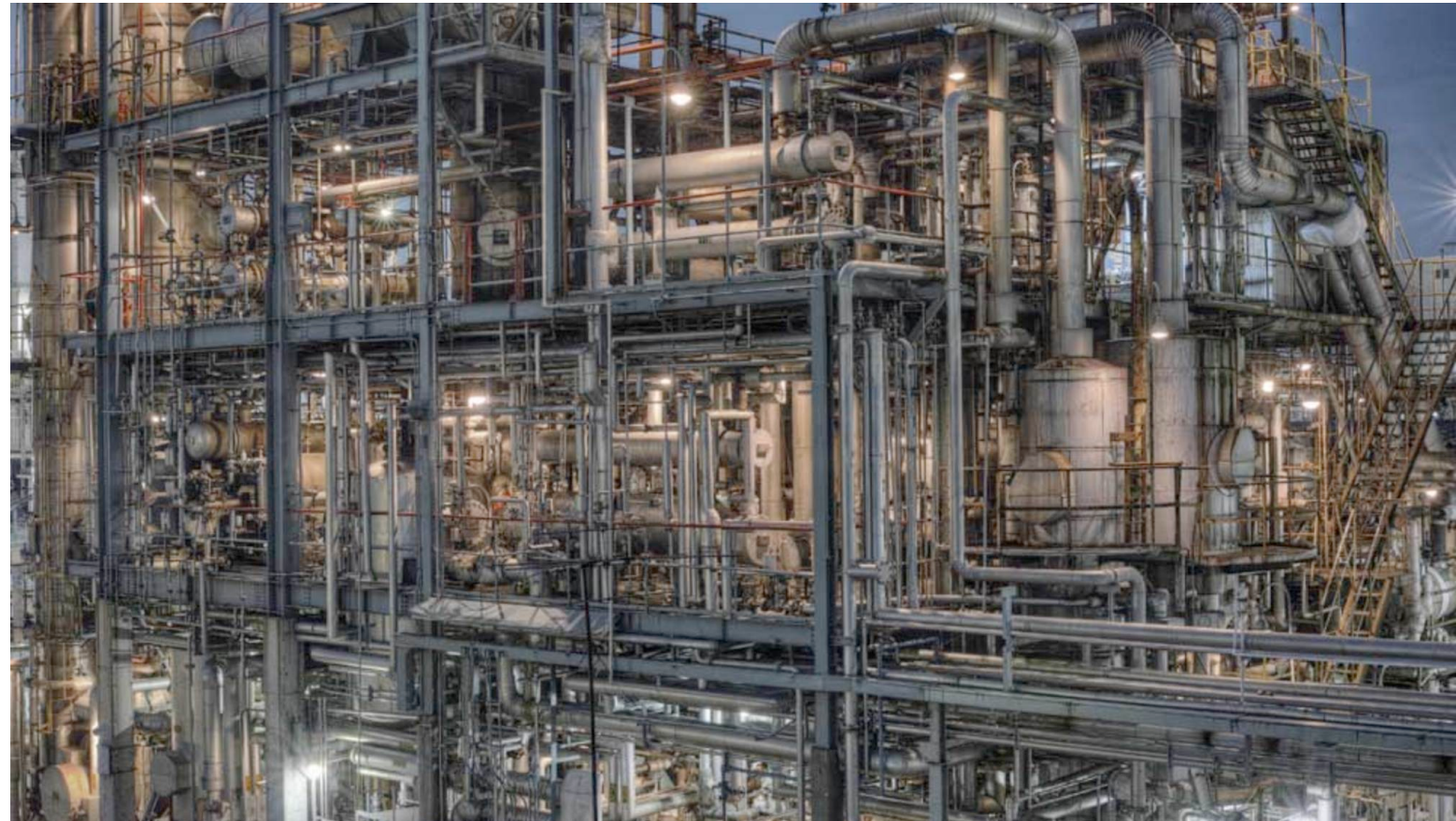
The need for **AI-native** hybrid models



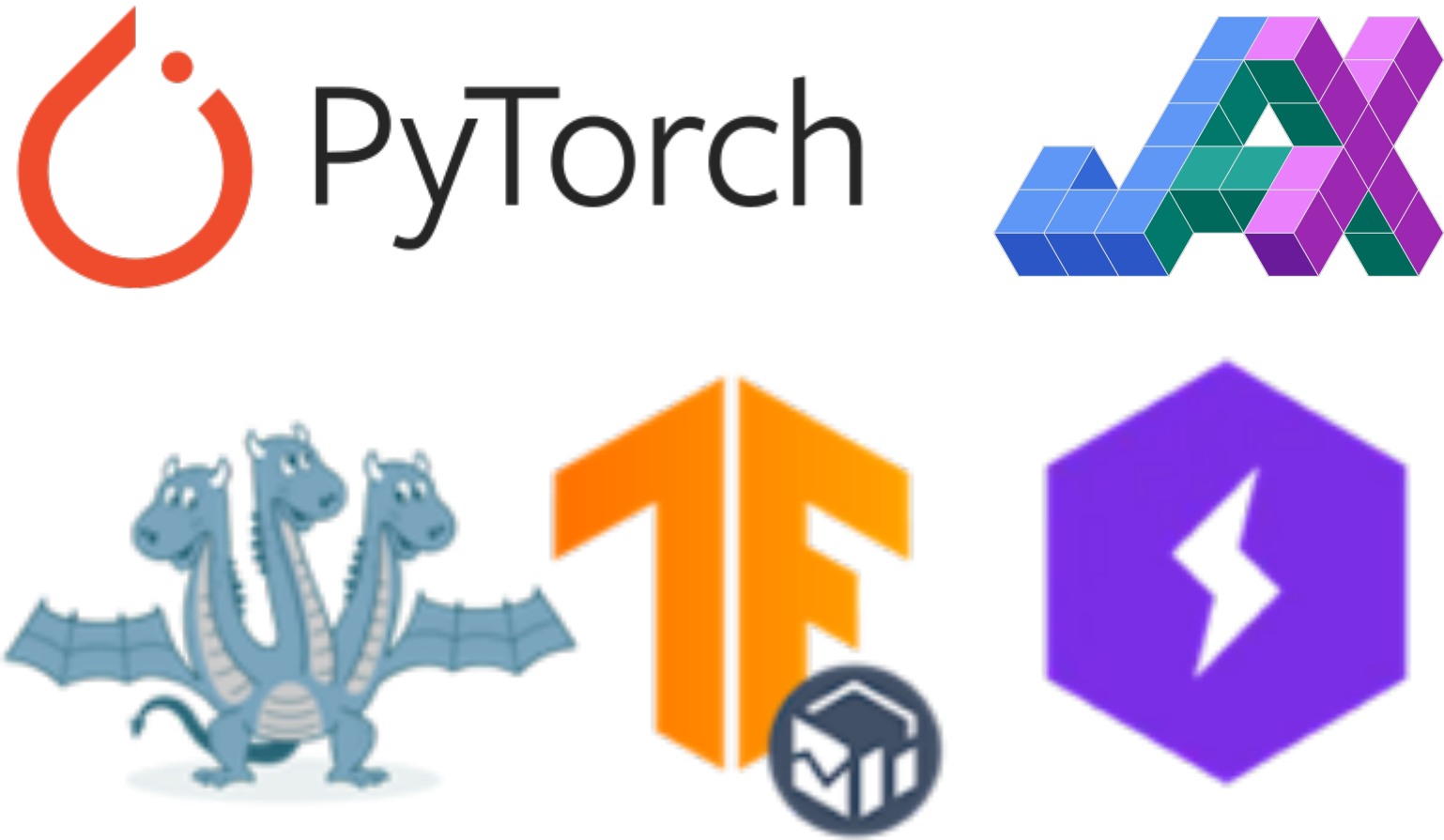
Avoid having to bridge the technological gap



stable, robust, low abstraction languages

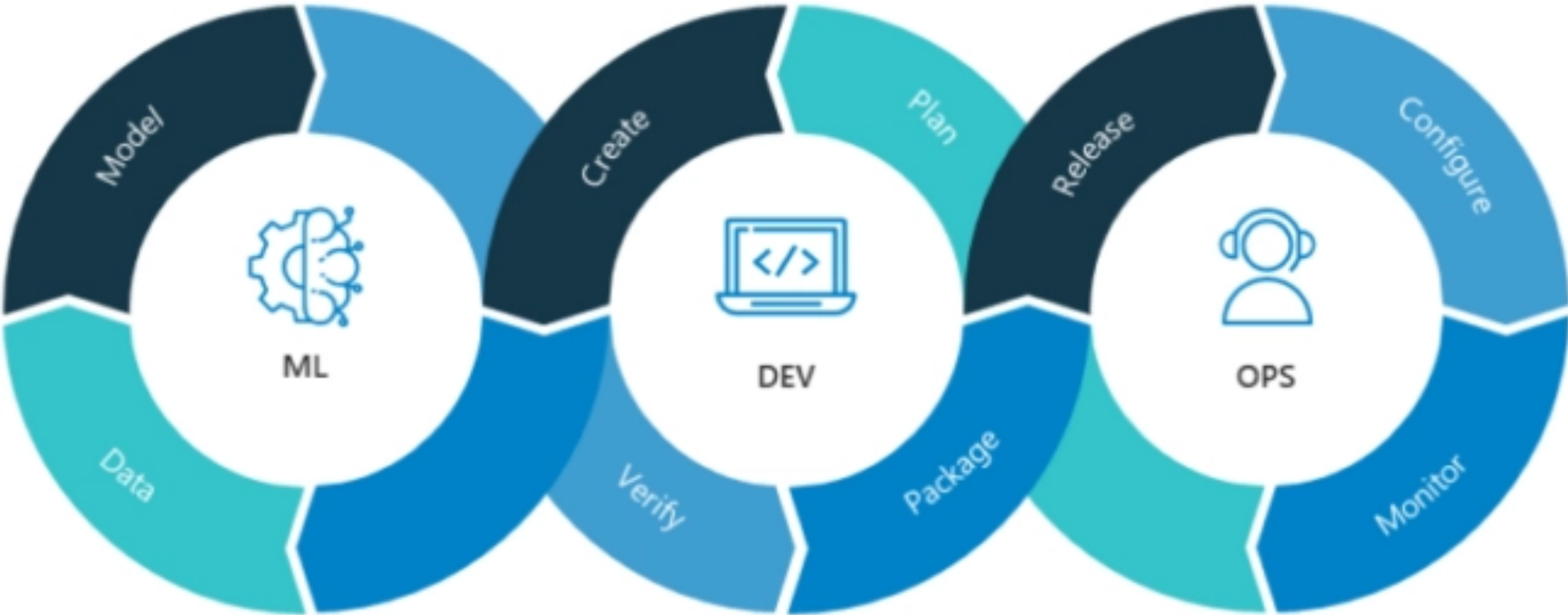


Less robust software design (APIs)



high abstraction, fast evolving languages

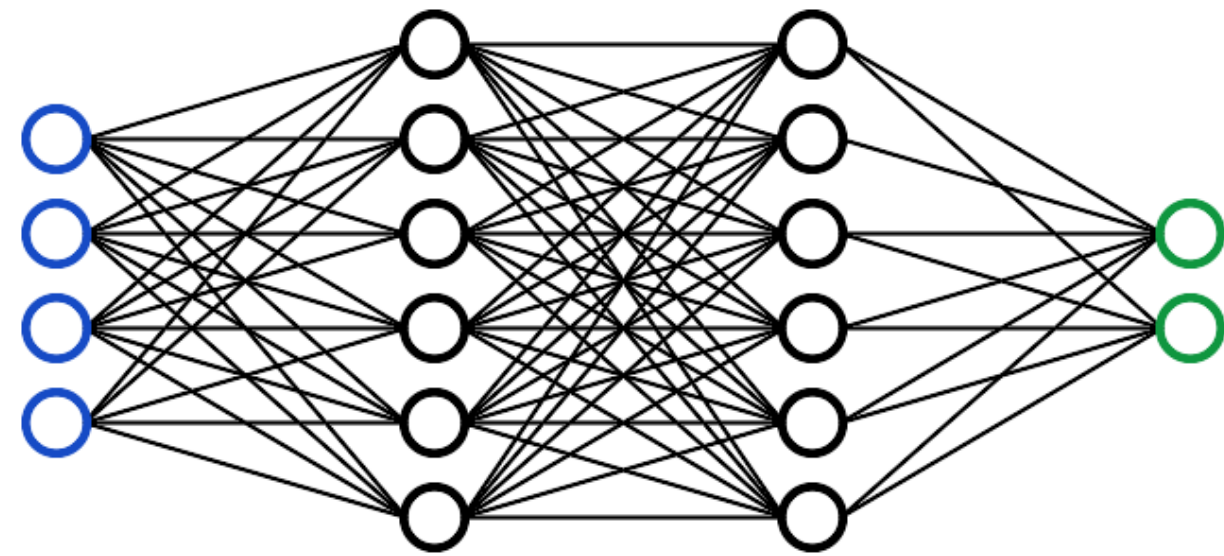
+



Clean APIs and MLOPs

Training ML components for physical models

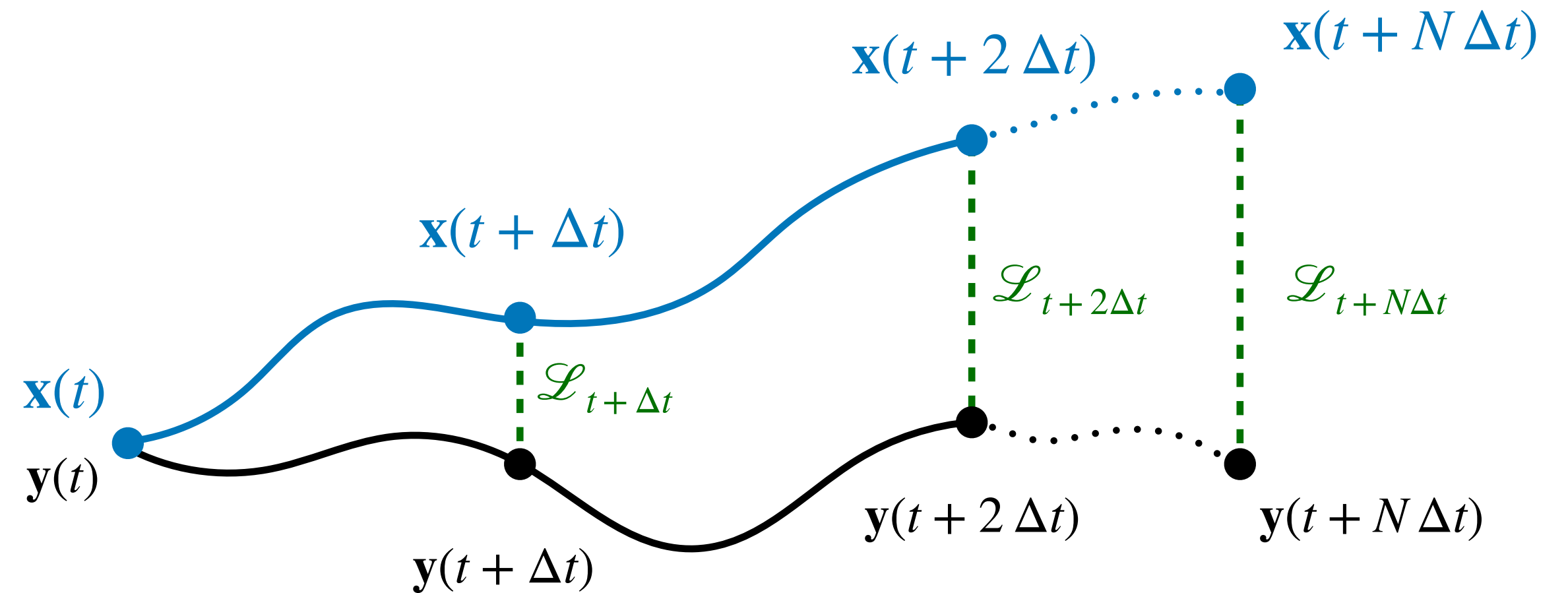
offline learning



mapping $\bar{\mathbf{x}} \rightarrow \overline{\mathcal{N}(\mathbf{x})}$

at fixed time t

online learning



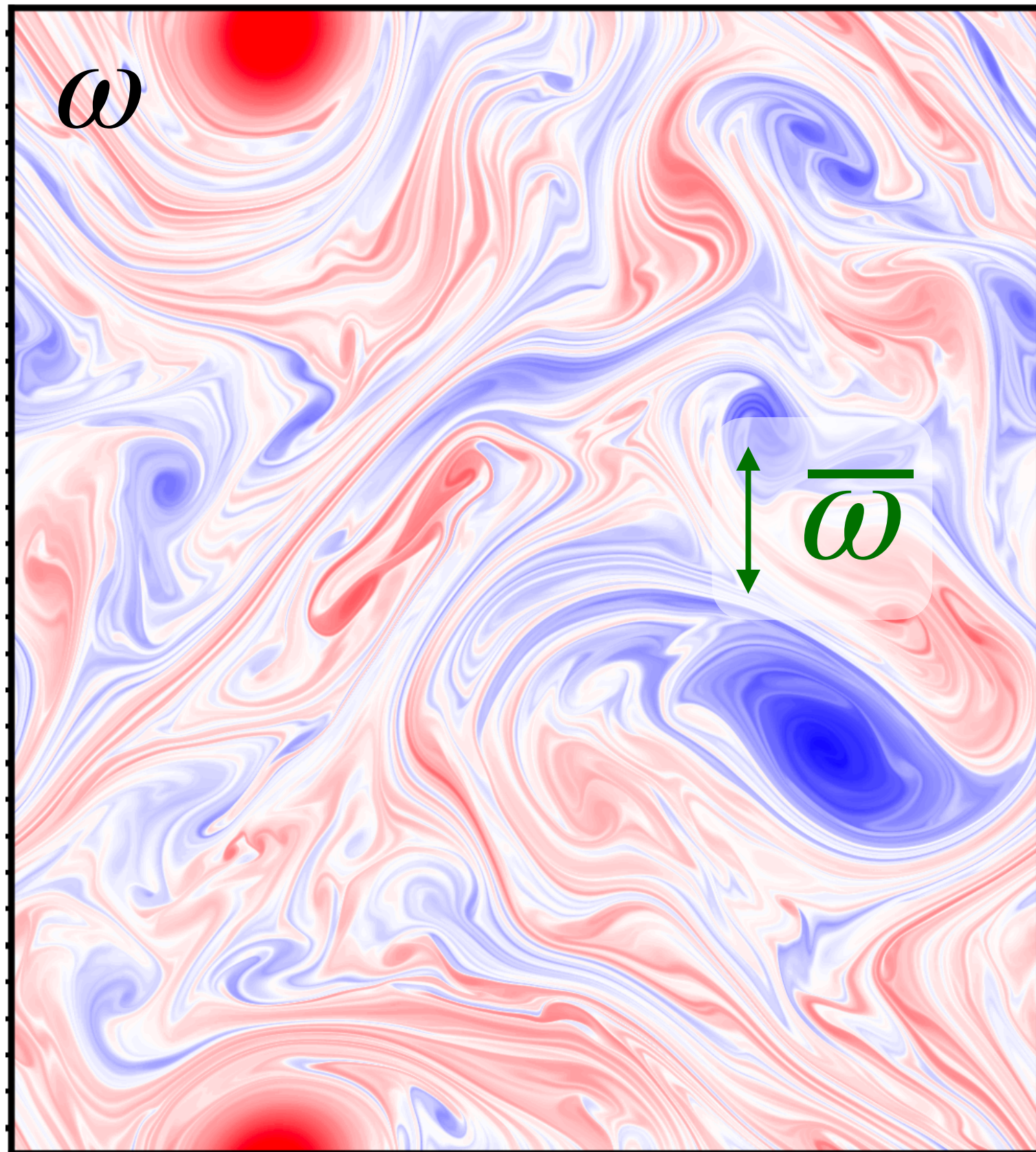
$$\partial_t \mathbf{y} + G(\mathbf{y}) + \mathcal{M}_{NN}(\mathbf{y}) = f$$

along a trajectory

(a.k.a : a posteriori, solver-in-the-loop, end-to-end)

ML closure for ocean macro-turbulence (1/3)

$$\partial_t \omega + J(\psi, \omega) = \nu \nabla^2 \omega - \mu \omega - \beta \partial_x \psi + F$$



$$\omega = \nabla^2 \psi$$

vorticity

$$\mathbf{u} = (-\partial_y \psi, \partial_x \psi)$$

velocity

Filtering $\bar{\omega} := \int \omega(\mathbf{x}') G(\mathbf{x} - \mathbf{x}') d\mathbf{x}'$

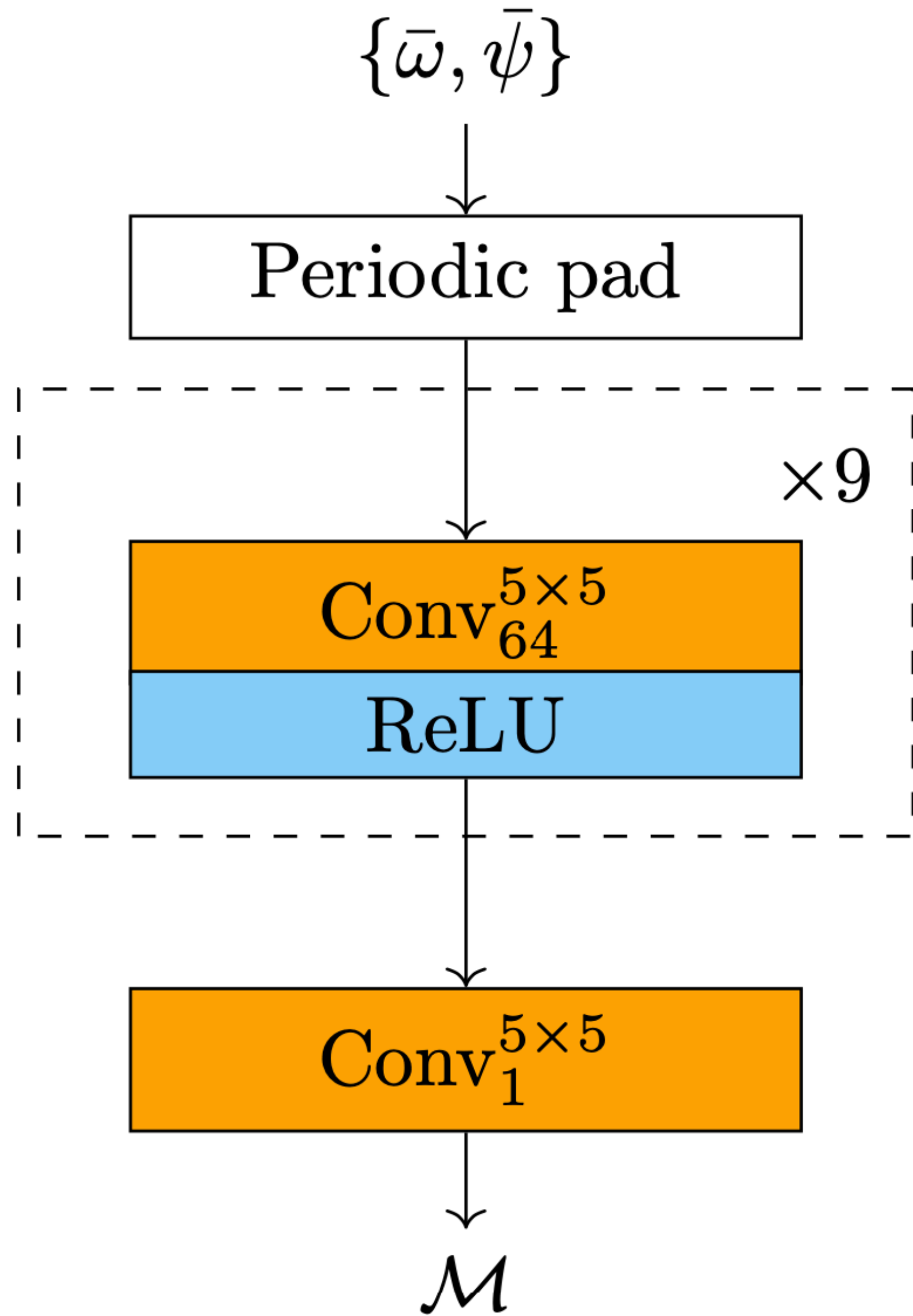
Filtered eq. $\partial_t \bar{\omega} + J(\bar{\psi}, \bar{\omega}) = rhs + R(\psi, \omega)$

SGS term $R(\psi, \omega) = \nabla \cdot (\bar{\mathbf{u}} \bar{\omega} - \bar{\mathbf{u}} \bar{\omega})$

See e.g. Graham and Ringler (2013)

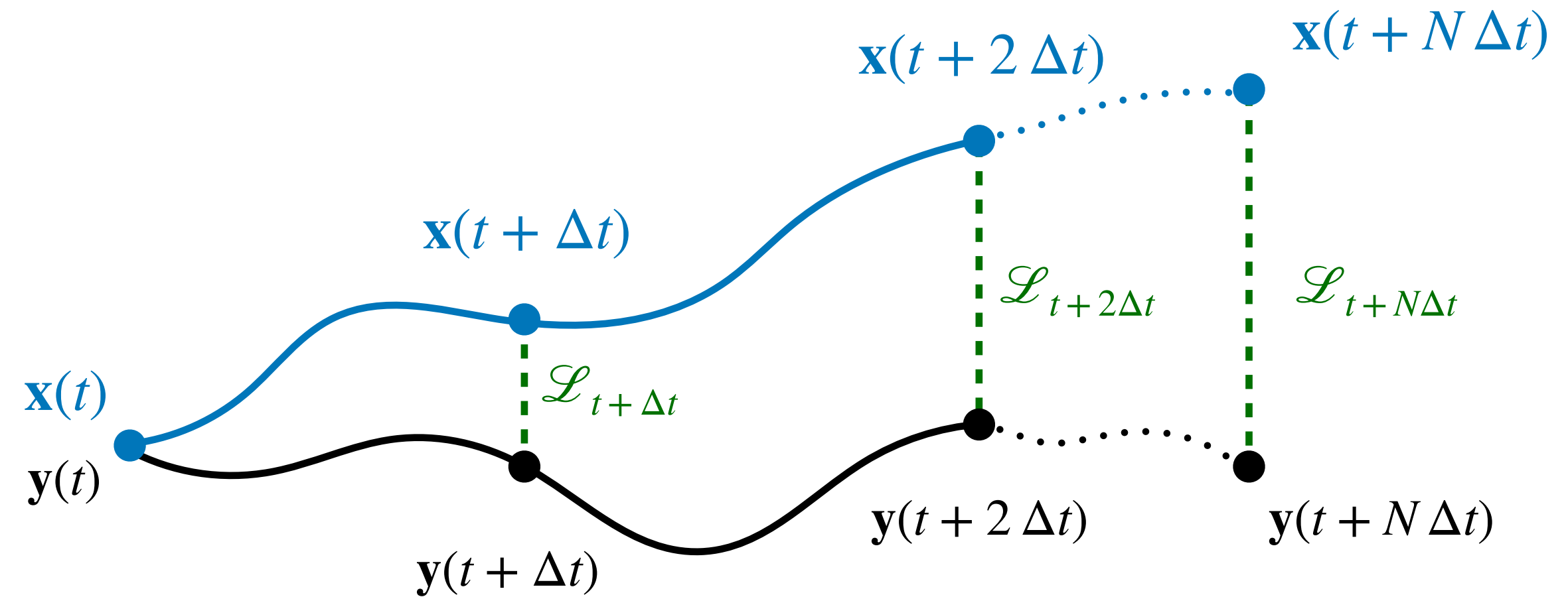
Closure pbm $R(\psi, \omega) \simeq \mathcal{M}_\theta^{NN}(\bar{\psi}, \bar{\omega})$

ML closure for ocean macro-turbulence (2/3)



Loss for a priori training

$$\mathcal{L}_{\text{prio}}(\mathcal{M}) := \frac{1}{S} \sum_{i=1}^S (R(\psi, \omega)_i - \mathcal{M}(\bar{\psi}_i, \bar{\omega}_i))^2$$

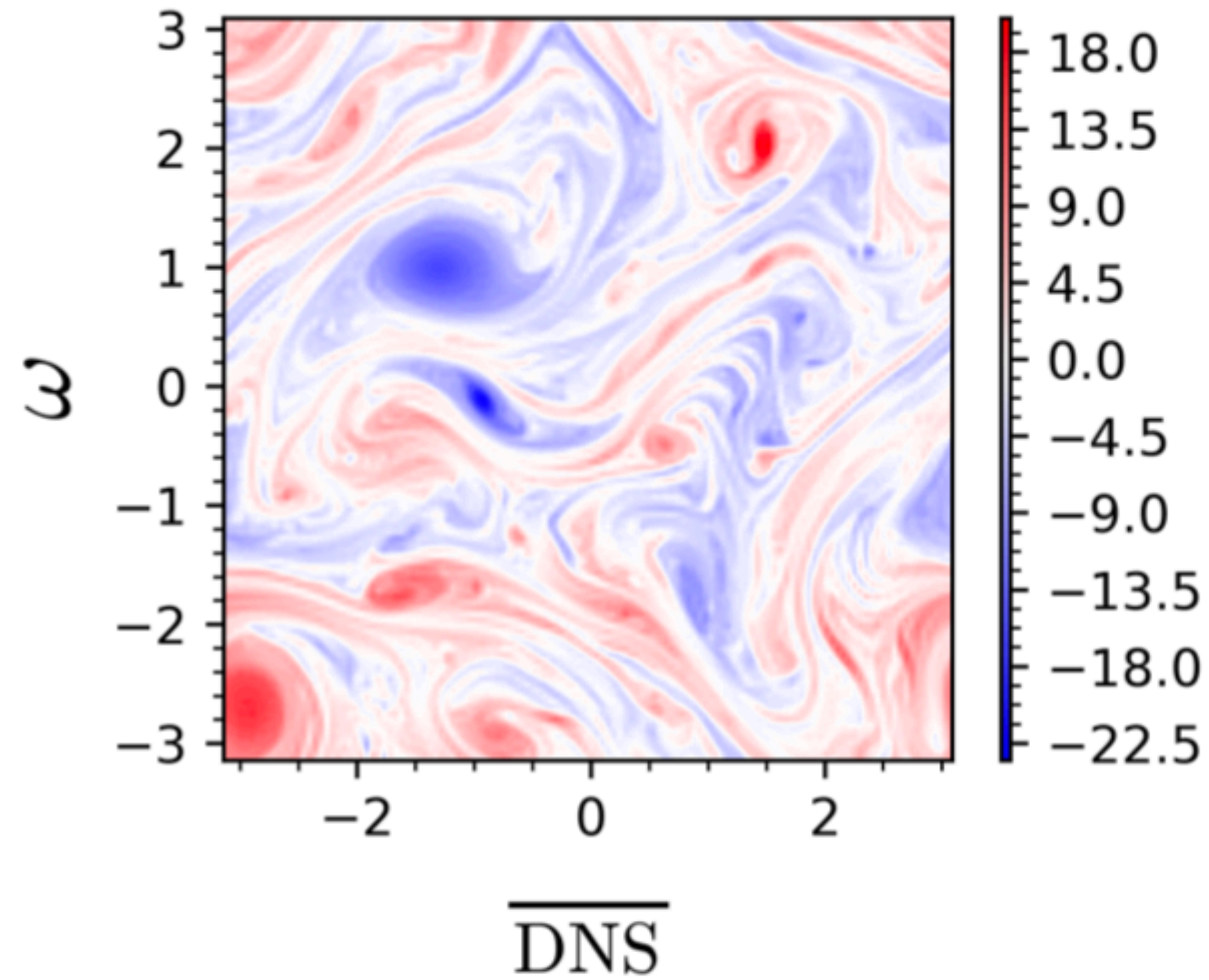


Loss for a posteriori training

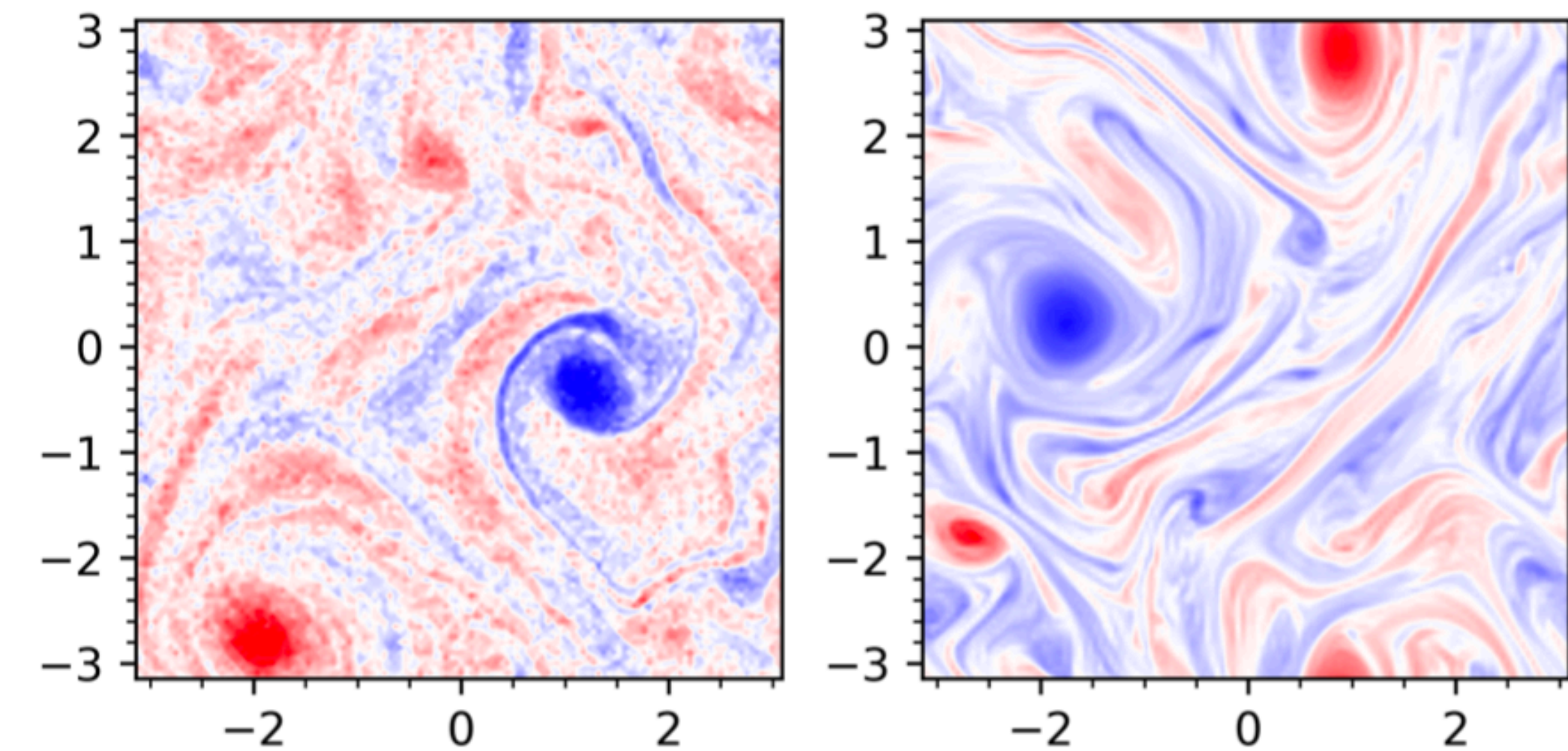
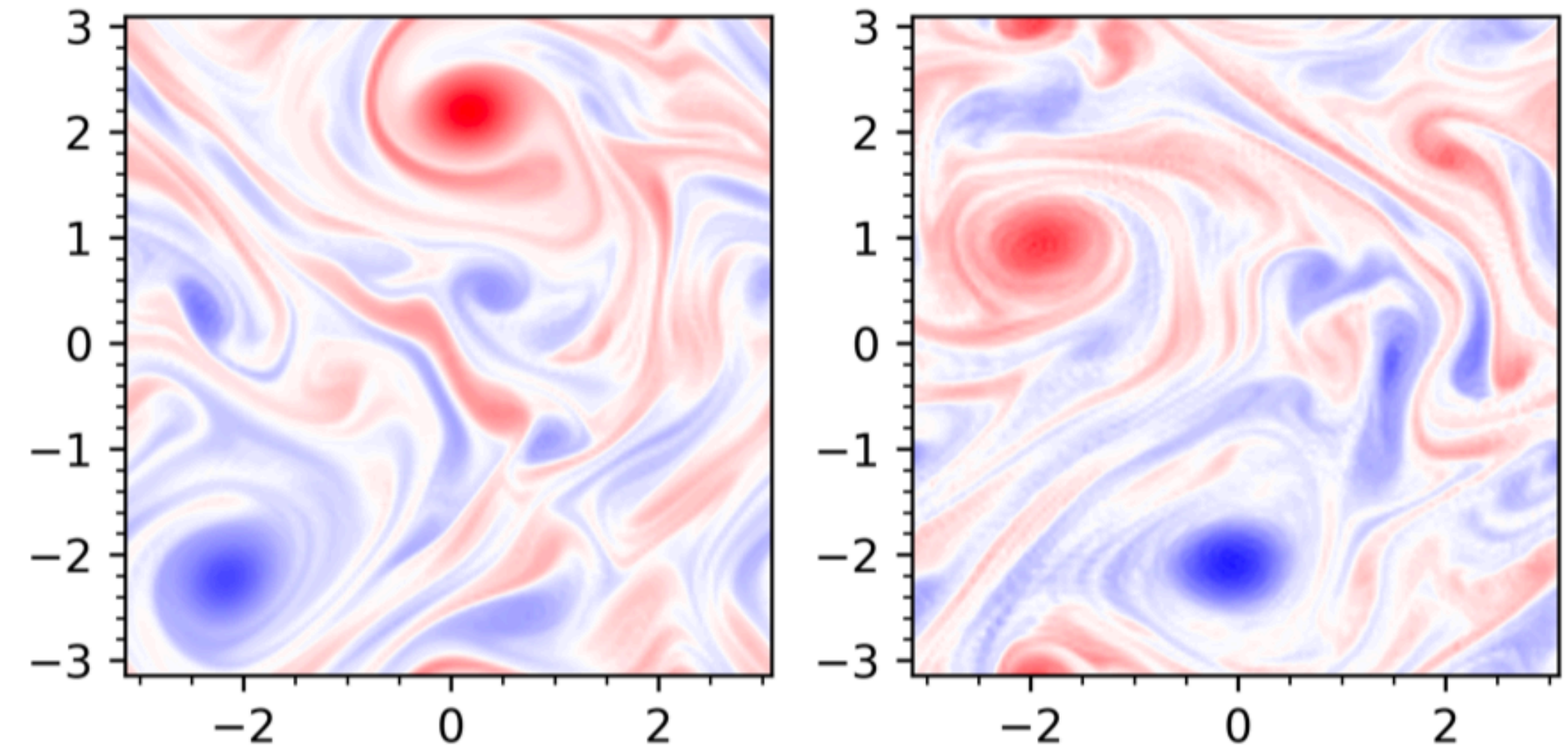
$$\mathcal{L}_{\text{post}}(\mathcal{M}) := \frac{1}{N} \sum_{i=1}^N (\mathcal{T}(\omega(i\Delta t)) - \bar{\omega}(i\Delta t))^2$$

ML closure for ocean macro-turbulence (3/3)

Frezat et al. 2022 JAMES



- baselines : over-diffusive
- offline learning : unstable
- online : stable and accurate



See also List et al. (2022, 2024)

The need for differentiable numerical solvers

$$\arg \min_{\theta} \mathcal{L}(\mathbf{z}, \mathcal{M}(\mathbf{y} | \theta))$$

target
prediction

$$\frac{\partial \mathcal{L}}{\partial \theta}(\mathbf{z}, \mathcal{M}(\mathbf{y} | \theta)) = \frac{\partial \mathcal{M}}{\partial \theta}(\mathbf{y} | \theta) \frac{\partial \mathcal{L}}{\partial \mathcal{M}}$$

gradient of the loss

For time evolving problems, with

$$\mathbf{y}(t + \Delta t) = E_m \circ \dots \circ E_1(\mathbf{y}(t))$$

$\mathcal{M} \equiv E$ temporal evolution operator

The gradient of the loss involves

tricky without Automatic Differentiation (AD)



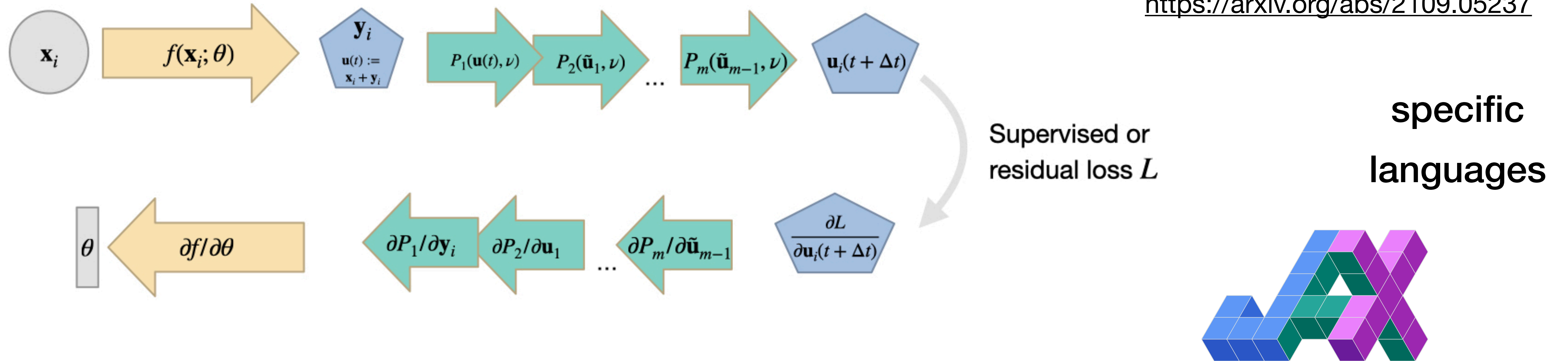
$$\frac{\partial \mathcal{M}}{\partial \theta} \equiv \frac{\partial E}{\partial \theta} = \frac{\partial(E_m \circ \dots \circ E_1)}{\partial \theta} = \frac{\partial E_m}{\partial E_{m-1}} \dots \frac{\partial E_2}{\partial E_1} \frac{\partial E_1}{\partial \theta}$$

But AD not available
in ocean models...

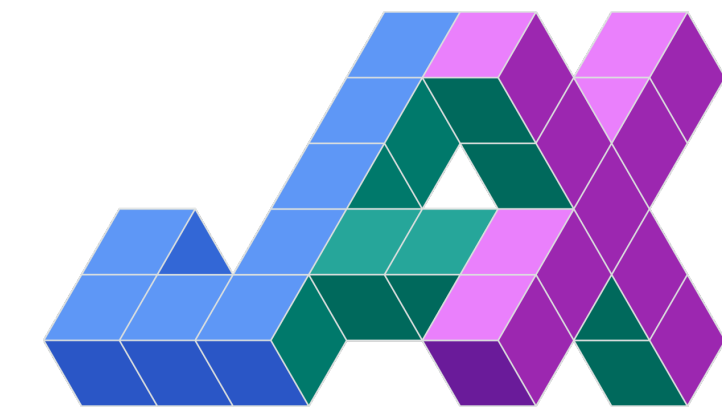
Leveraging differentiable programming

See eg Thuerey et al. 2021

<https://arxiv.org/abs/2109.05237>



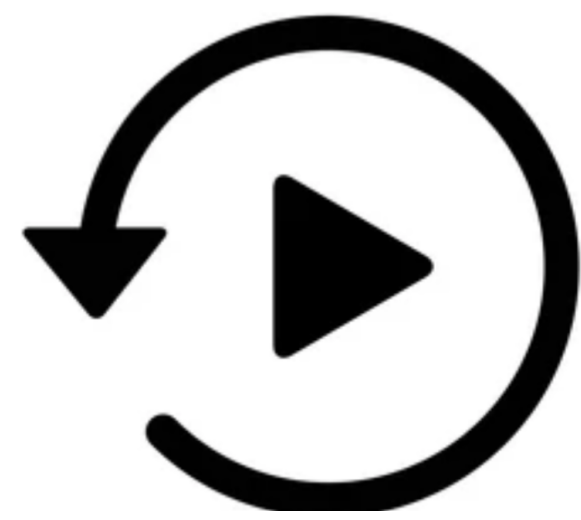
- programs composed of **differentiable** building blocks
- building blocks : trainable and procedural **code components**
- **trainable end-to-end** with gradient based optimisation



julia

a generalisation of
deep learning

Differentiable numerical
simulations of physical systems



AI-native hybrid geoscientific models

arXiv:2311.07222v3 [physics.ao-ph] 8 Mar 2024

Neural General Circulation Models for Weather and Climate

Dmitrii Kochkov^{1†}, Janni Yuval^{1†}, Ian Langmore^{1†}, Peter Norgaard^{1†}, Jamie Smith^{1†}, Griffin Mooers¹, Milan Klöwer⁴, James Lottes¹, Stephan Rasp¹, Peter Düben³, Sam Hatfield³, Peter Battaglia², Alvaro Sanchez-Gonzalez², Matthew Willson², Michael P. Brenner^{1,5}, Stephan Hoyer^{1†}

¹Google Research, Mountain View, CA. ²Google DeepMind, London, UK. ³European Centre for Medium-Range Weather Forecasts, Reading, UK. ⁴Earth, Atmospheric and Planetary Sciences, Massachusetts Institute of Technology. ⁵School of Engineering and Applied Sciences, Harvard University.

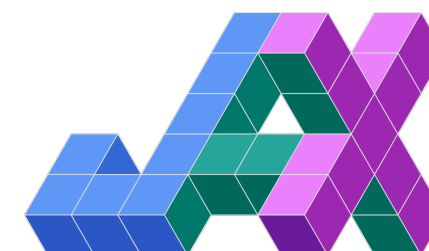
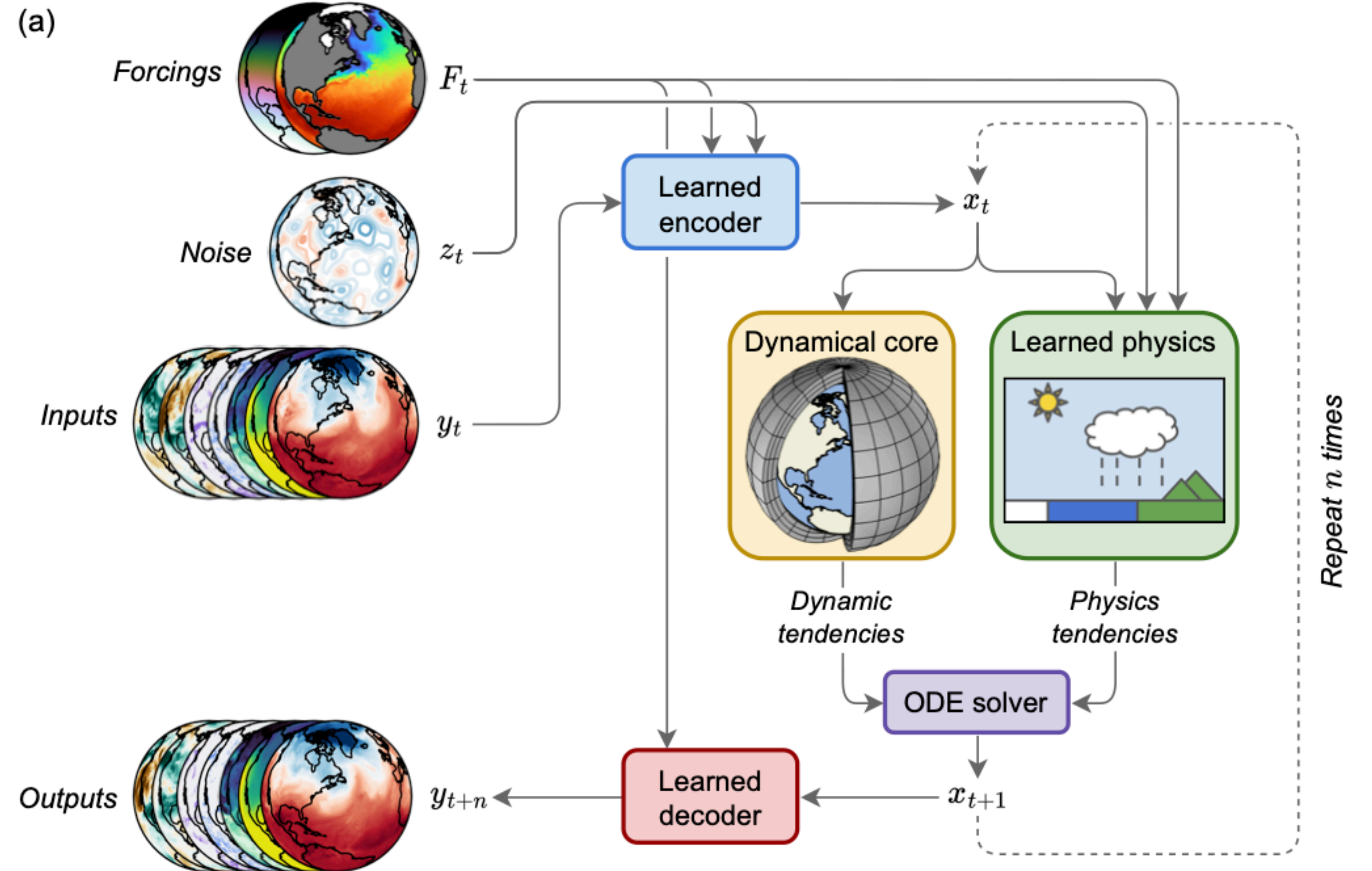
*Corresponding author(s). E-mail(s): dkochkov@google.com; janniyuval@google.com; shoyer@google.com; †These authors contributed equally to this work.

Abstract
General circulation models (GCMs) are the foundation of weather and climate prediction. GCMs are physics-based simulators which combine a numerical solver for large-scale dynamics with tuned representations for small-scale processes such as cloud formation. Recently, machine learning (ML) models trained on reanalysis data achieved comparable or better skill than GCMs for deterministic weather forecasting. However, these models have not demonstrated improved ensemble forecasts, or shown sufficient stability for long-term weather and climate simulations. Here we present the first GCM that combines a differentiable solver for atmospheric dynamics with ML components, and show that it can generate forecasts of deterministic weather, ensemble weather and climate on par with the best ML and physics-based methods. NeuralGCM is competitive with ML models for 1-10 day forecasts, and with the European Centre for Medium-Range Weather Forecasts ensemble prediction for 1-15 day forecasts. With prescribed sea surface temperature, NeuralGCM can accurately track climate metrics such as global mean temperature for multiple decades, and climate forecasts with 140

1

<https://arxiv.org/abs/2311.07222>

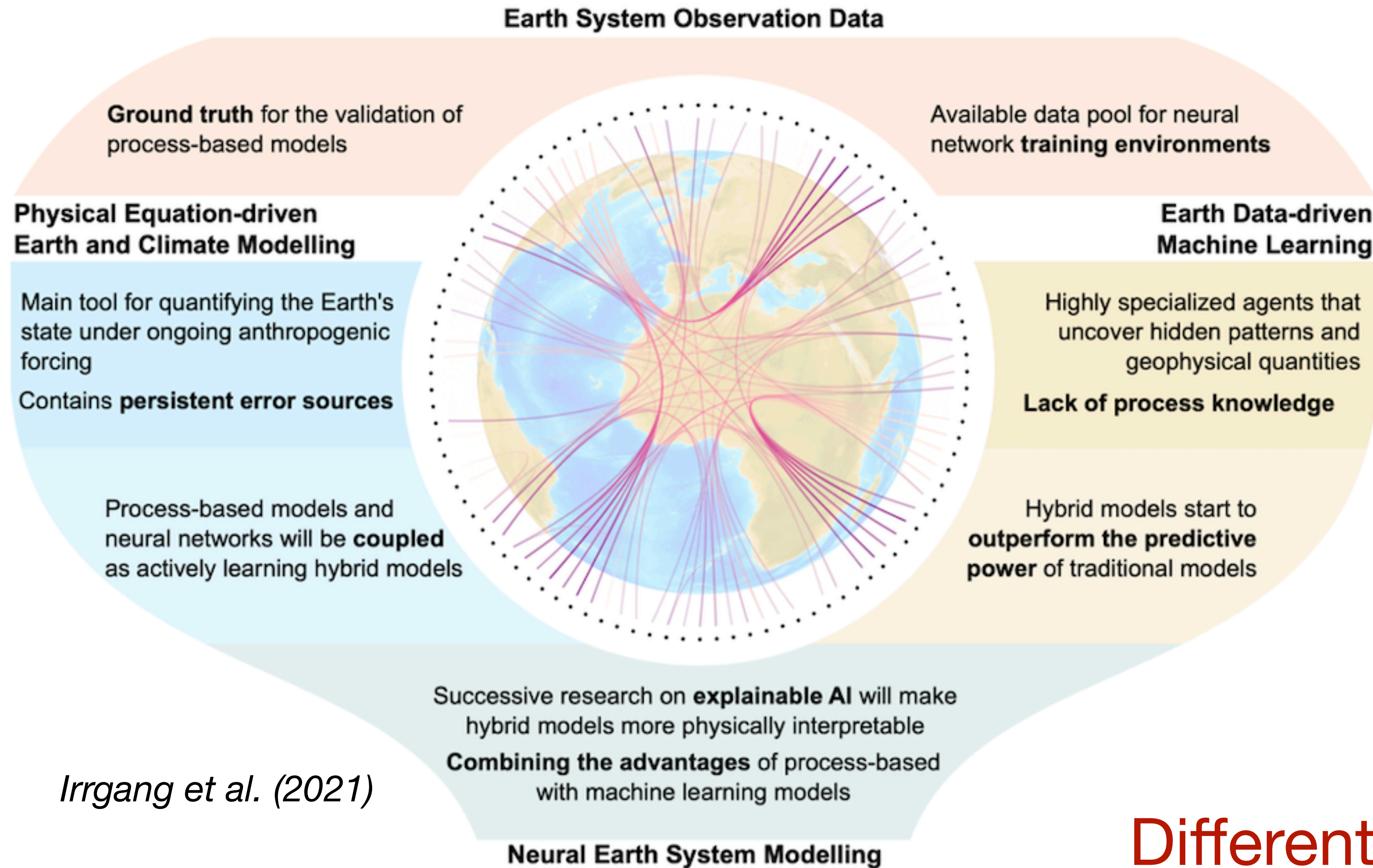
Kochkov et al. (2024)



<https://github.com/google-research/dinosaur>

<https://github.com/google-research/neuralgcm>

AI-native hybrid geoscientific models

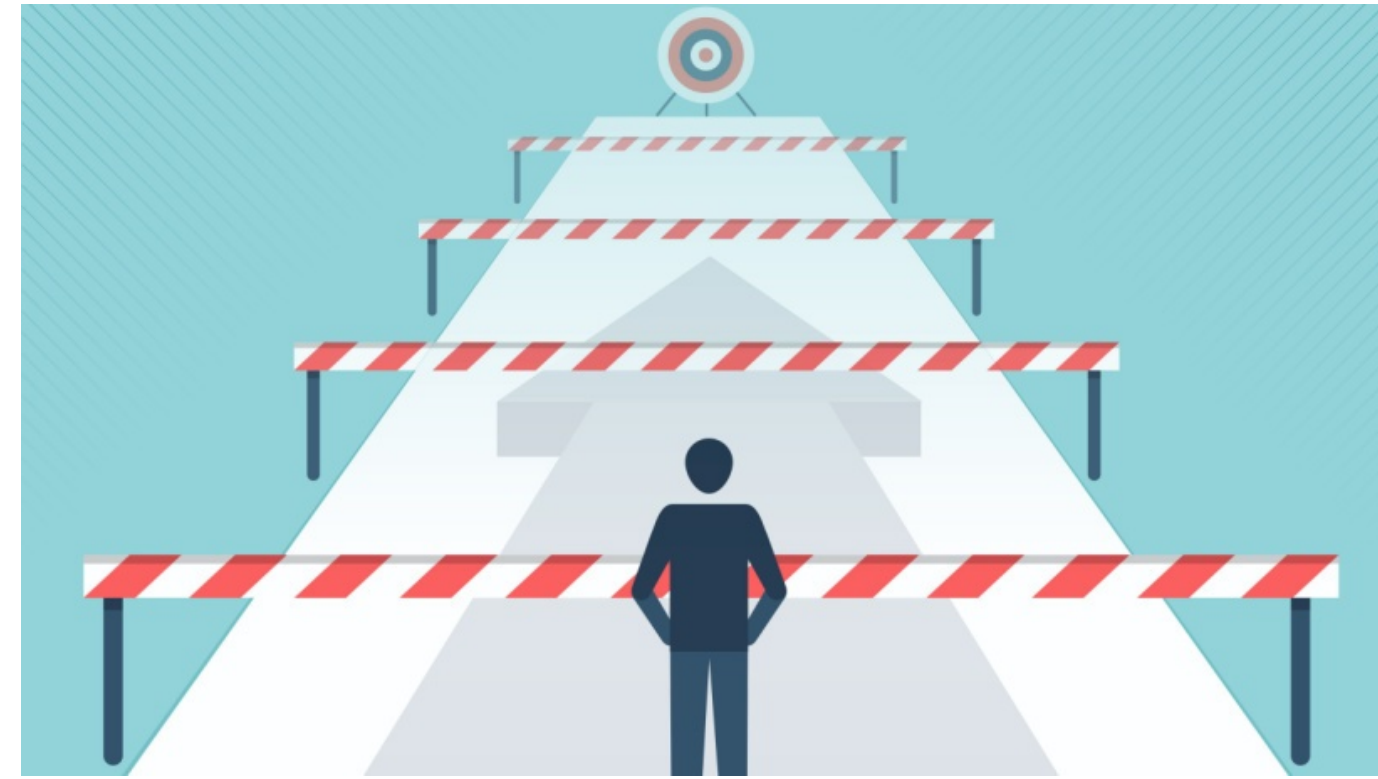


Irrgang et al. (2021)

- Allowing to optimise
- model parameters
 - numerical schemes
 - subgrid closures
 - ...

...and better exploit observations and hi-res simulations

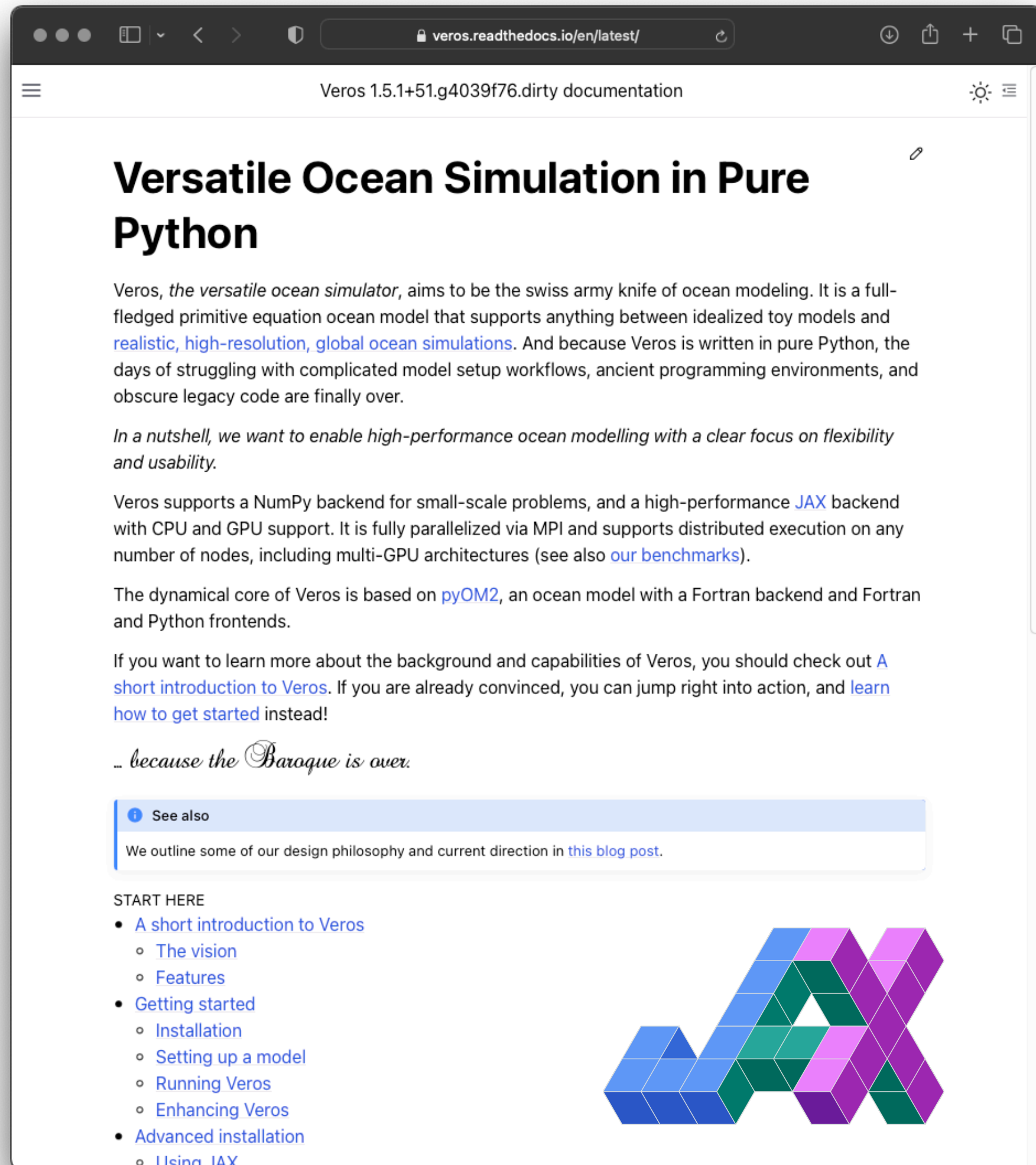
Differentiable programming
in earth system models



Path **towards** AI-native hybrid ocean models



The choice of the programming language



Veros 1.5.1+51.g4039f76.dirty documentation

Versatile Ocean Simulation in Pure Python

Veros, *the versatile ocean simulator*, aims to be the swiss army knife of ocean modeling. It is a full-fledged primitive equation ocean model that supports anything between idealized toy models and [realistic, high-resolution, global ocean simulations](#). And because Veros is written in pure Python, the days of struggling with complicated model setup workflows, ancient programming environments, and obscure legacy code are finally over.

In a nutshell, we want to enable high-performance ocean modelling with a clear focus on flexibility and usability.

Veros supports a NumPy backend for small-scale problems, and a high-performance [JAX](#) backend with CPU and GPU support. It is fully parallelized via MPI and supports distributed execution on any number of nodes, including multi-GPU architectures (see also [our benchmarks](#)).

The dynamical core of Veros is based on [pyOM2](#), an ocean model with a Fortran backend and Fortran and Python frontends.

If you want to learn more about the background and capabilities of Veros, you should check out [A short introduction to Veros](#). If you are already convinced, you can jump right into action, and [learn how to get started](#) instead!


... because the Baroque is over.

See also

We outline some of our design philosophy and current direction in [this blog post](#).

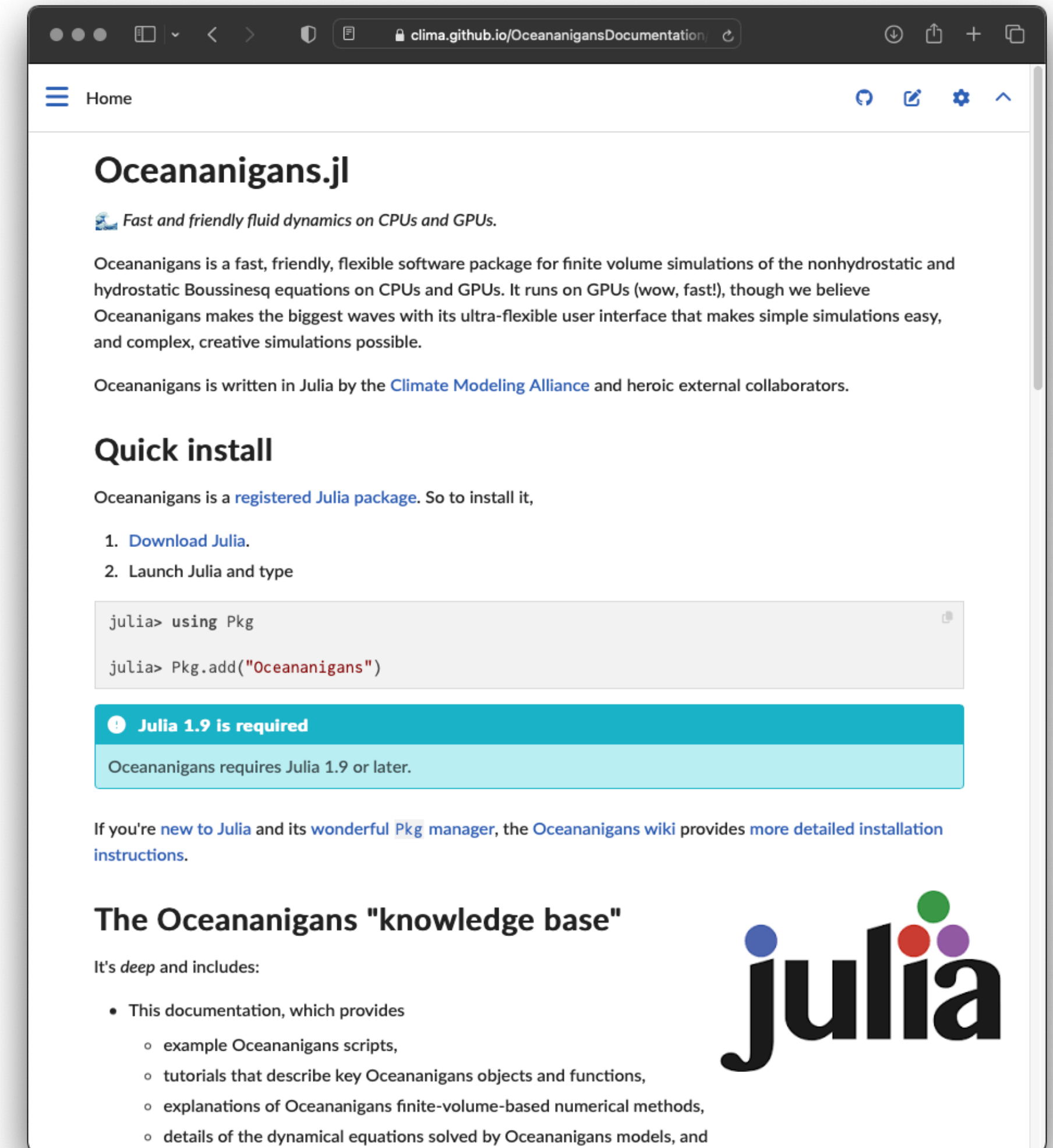
START HERE

- [A short introduction to Veros](#)
 - [The vision](#)
 - [Features](#)
- [Getting started](#)
 - [Installation](#)
 - [Setting up a model](#)
 - [Running Veros](#)
 - [Enhancing Veros](#)
- [Advanced installation](#)
 - [Using JAX](#)



Examples
of (almost) AI-native
ocean models

but not fully
AI-ready yet



Home

Oceananigans.jl

[Fast and friendly fluid dynamics on CPUs and GPUs.](#)

Oceananigans is a fast, friendly, flexible software package for finite volume simulations of the nonhydrostatic and hydrostatic Boussinesq equations on CPUs and GPUs. It runs on GPUs (wow, fast!), though we believe Oceananigans makes the biggest waves with its ultra-flexible user interface that makes simple simulations easy, and complex, creative simulations possible.

Oceananigans is written in Julia by the [Climate Modeling Alliance](#) and heroic external collaborators.

Quick install

Oceananigans is a [registered Julia package](#). So to install it,

1. [Download Julia](#).
2. Launch Julia and type

```
julia> using Pkg  
julia> Pkg.add("Oceananigans")
```


Julia 1.9 is required
Oceananigans requires Julia 1.9 or later.

If you're [new to Julia](#) and its wonderful [Pkg manager](#), the [Oceananigans wiki](#) provides [more detailed installation instructions](#).

The Oceananigans "knowledge base"

It's *deep* and includes:

- This documentation, which provides
 - example Oceananigans scripts,
 - tutorials that describe key Oceananigans objects and functions,
 - explanations of Oceananigans finite-volume-based numerical methods,
 - details of the dynamical equations solved by Oceananigans models, and

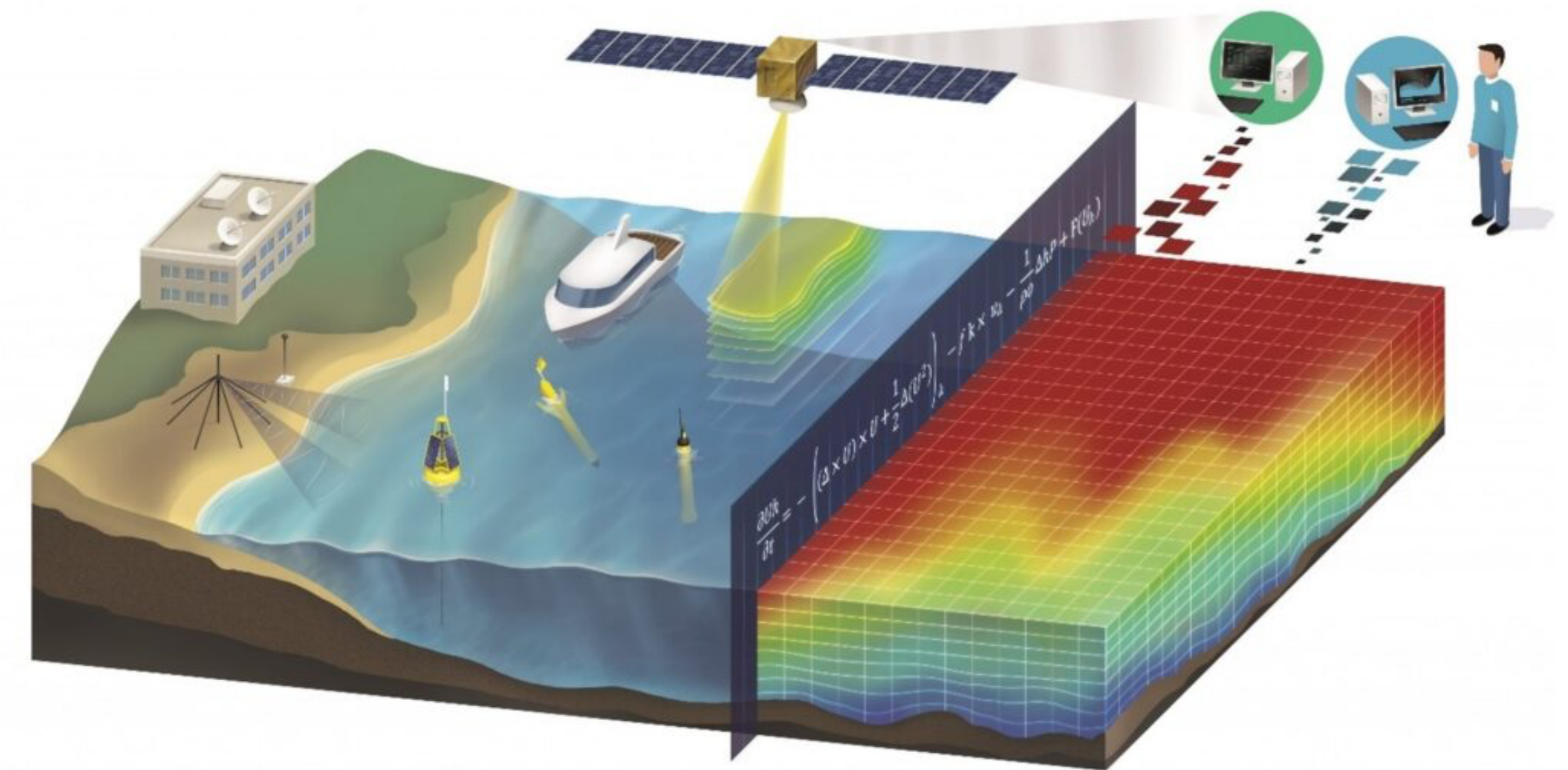
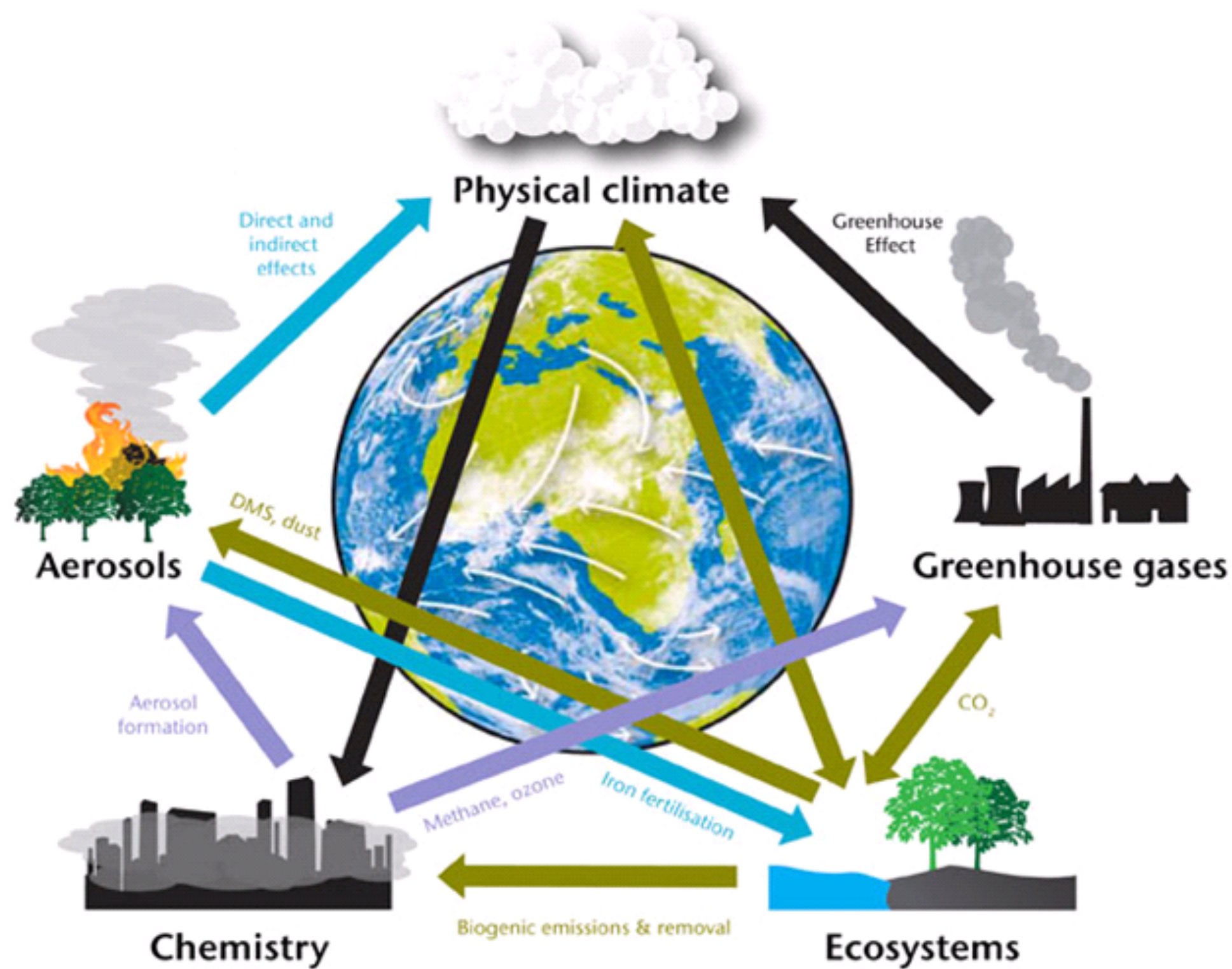


Our Graal : AI-ready, differentiable, fast, high-level abstraction, long-lasting.

Revisiting our systems' APIs



Copernicus
Marine Service



©Mercator Océan International

Earth System models
(IPCC)

Operational prediction systems
(Copernicus)

Systems build over decades, based on low level abstraction

No clearly defined APIs for ocean models.

The need for cross-disciplinary efforts



Use cases



Physics



AI



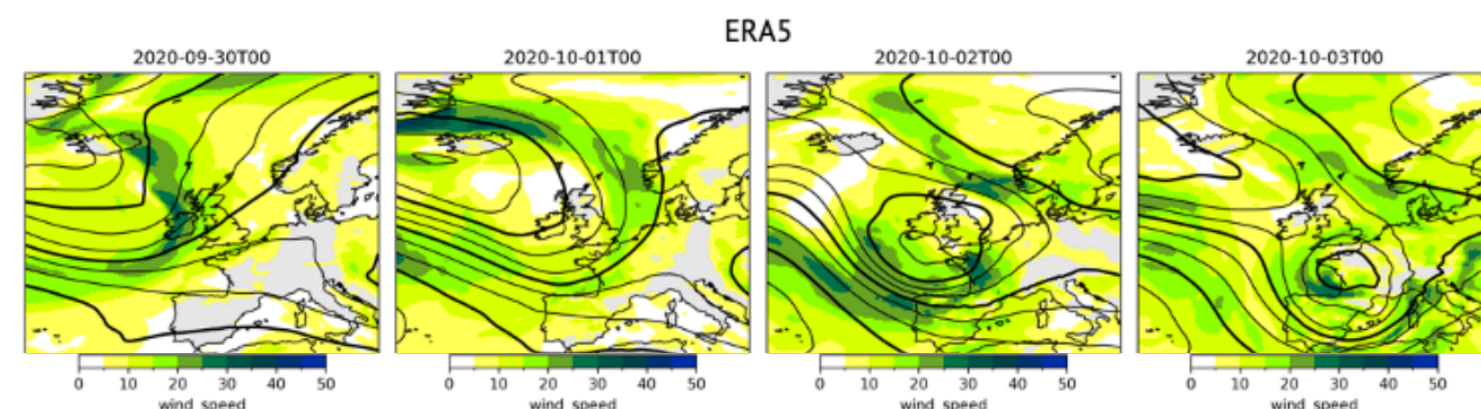
**Computer
science**



Applied maths

Accelerating progress with open benchmarks (1/2)

<https://github.com/google-research/weatherbench2>



data source : ERA5 reanalysis

Keisler et al. (2022)

ClimaX (Microsoft)

PanguWeather (Huawei)

GraphCast (Google Deep Mind)

WeatherBench 2

- 3D, more variables
- more metrics
- probabilistic forecasts

<https://doi.org/10.1038/s41586-023-06185-3>

<https://doi.org/10.1038/s41586-023-06185-3>

<https://doi.org/10.1038/s41586-023-06185-3>

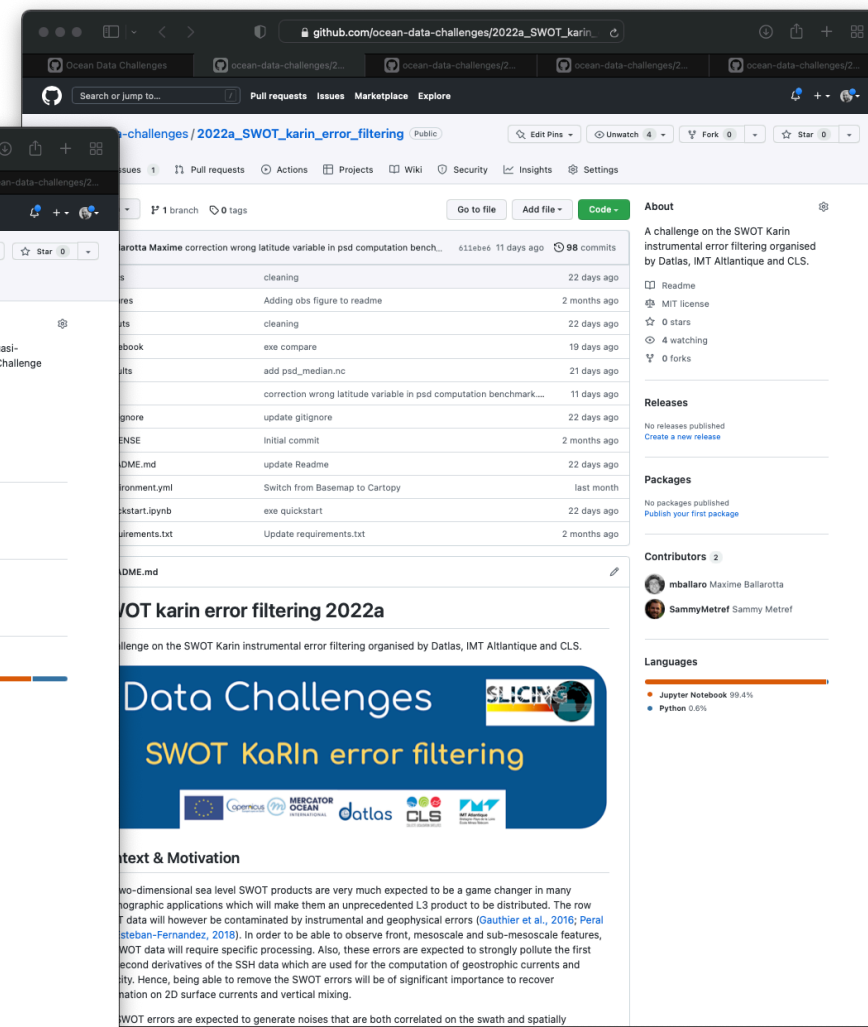
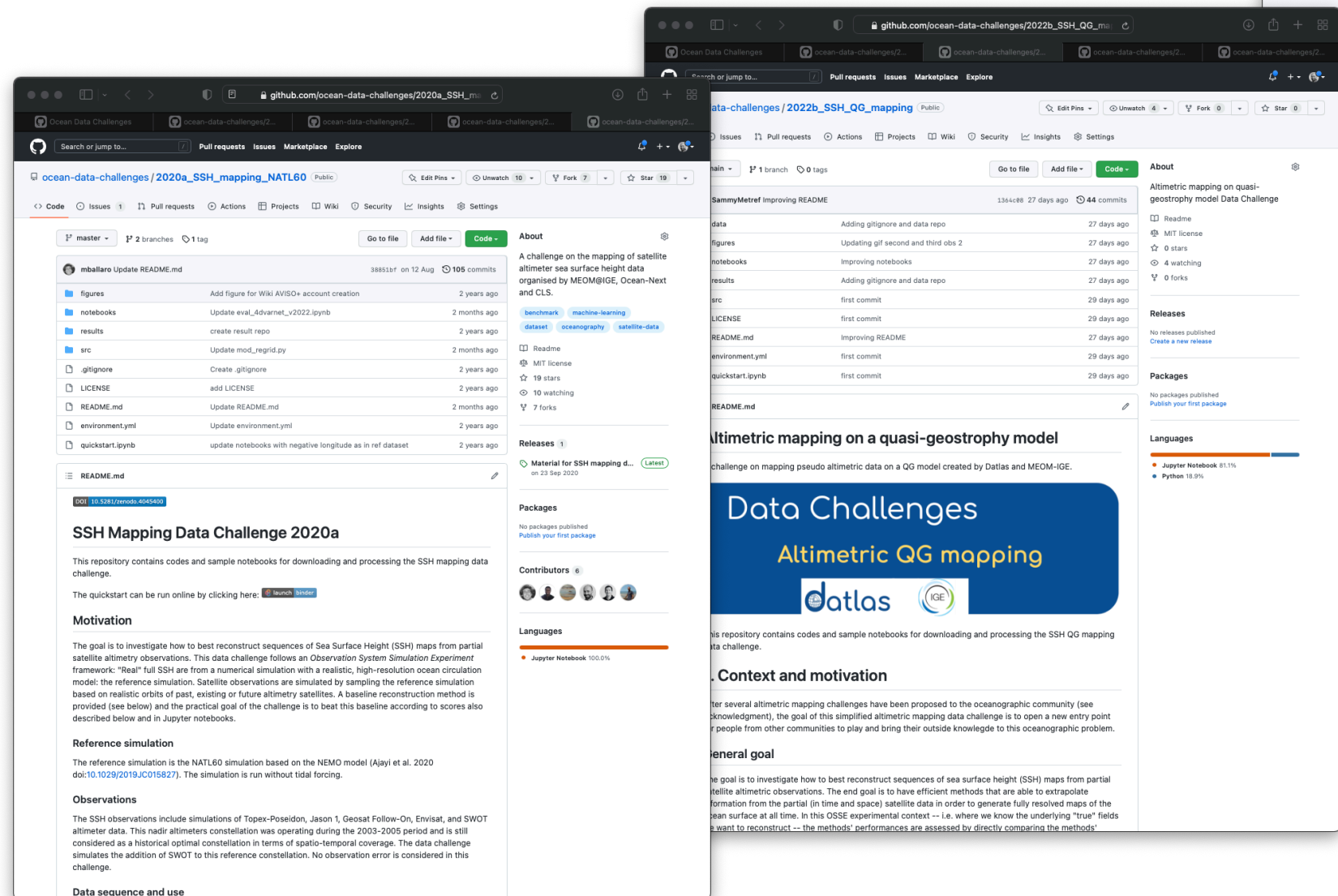
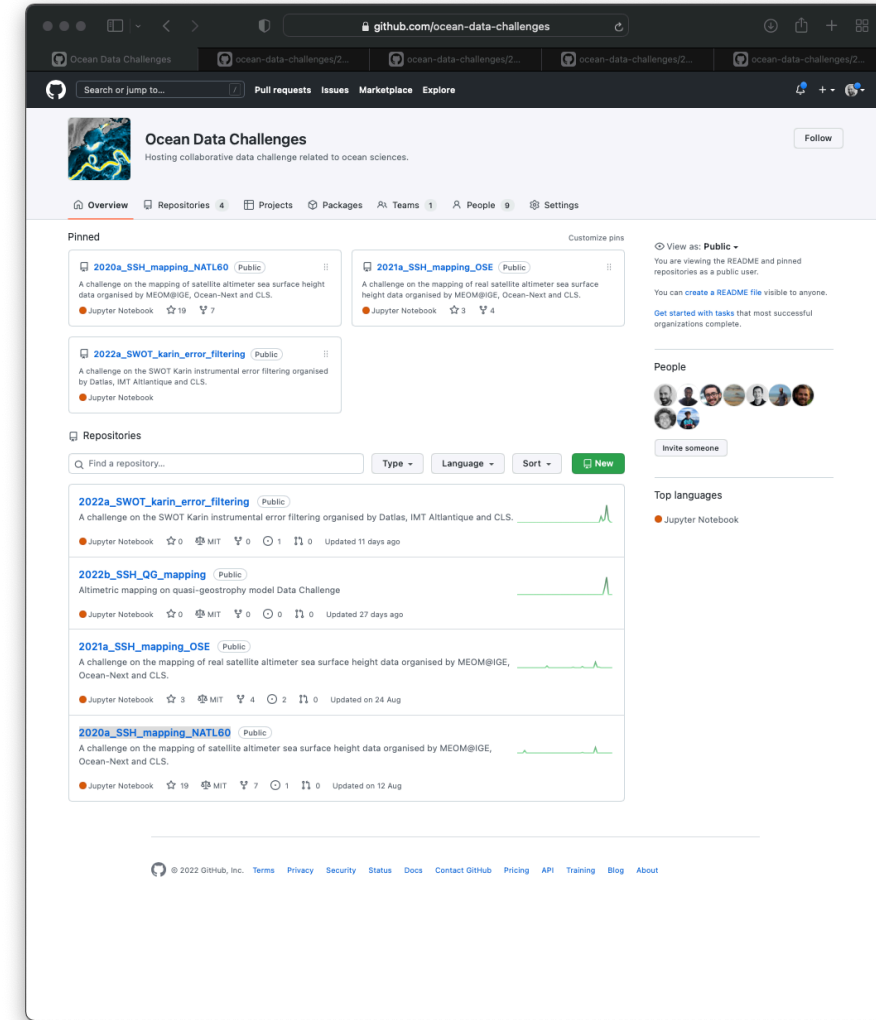
<https://doi.org/10.48550/arXiv.2212.12794>

Accelerating progress with open benchmarks (2/2)

Collaborative data-challenges



- problem description + baseline
- data, metrics (with codes)
- tools for collaboration and papers



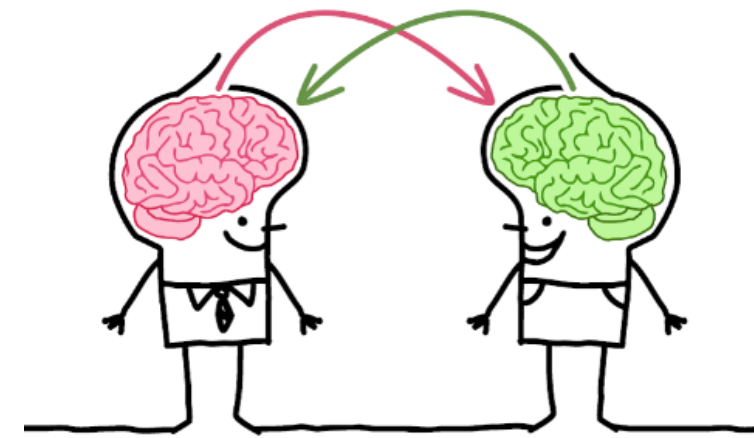
model data / obs data

Leaderboard

Method	μ (RMSE)	σ (RMSE)	λ_x (degree)	λ_t (days)	Notes	Reference
baseline OI 1 nadir	0.69	0.03	3.31	33.32	Covariances not optimized	quickstart.ipynb
baseline OI 4 nadirs	0.83	0.04	2.25	15.67	Covariances not optimized	quickstart.ipynb
baseline OI 1 swot	0.85	0.05	1.22	12.38	Covariances not optimized	quickstart.ipynb
duacs 4 nadirs	0.92	0.01	1.42	12.0	Covariances DUACS	eval_duacs.ipynb
bfn 4 nadirs	0.92	0.02	1.23	10.6	QG Nudging	eval_bfn.ipynb
dymost 4 nadirs	0.91	0.01	1.36	11.79	Dynamic mapping	eval_dymost.ipynb
miost 4 nadirs	0.93	0.01	1.35	10.19	Multiscale mapping	eval_miost.ipynb
4DVarNet 4 nadirs	0.94	0.01	1.18	10.34	4DVarNet mapping	eval_4dvarnet.ipynb
duacs 1 swot + 4 nadirs	0.92	0.02	1.22	11.15	Covariances DUACS	eval_duacs.ipynb
bfn 1 swot + 4 nadirs	0.93	0.02	0.8	10.09	QG Nudging	eval_bfn.ipynb
dymost 1 swot + 4 nadirs	0.93	0.02	1.2	10.07	Dynamic mapping	eval_dymost.ipynb
miost 1 swot + 4 nadirs	0.94	0.01	1.18	10.0		
4DVarNet 1 swot + 4 nadirs	0.95	0.01	0.82	6.0		

μ (RMSE): average RMSE score.
 σ (RMSE): standard deviation of the RMSE score.
 λ_x : minimum spatial scale resolved.
 λ_t : minimum time scale resolved.

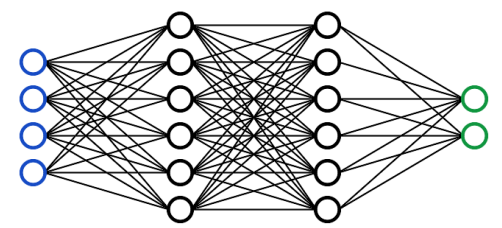
Le Guillou et al. 2021
 Febvre et al. 2021
 Beauchamp et al. 2022



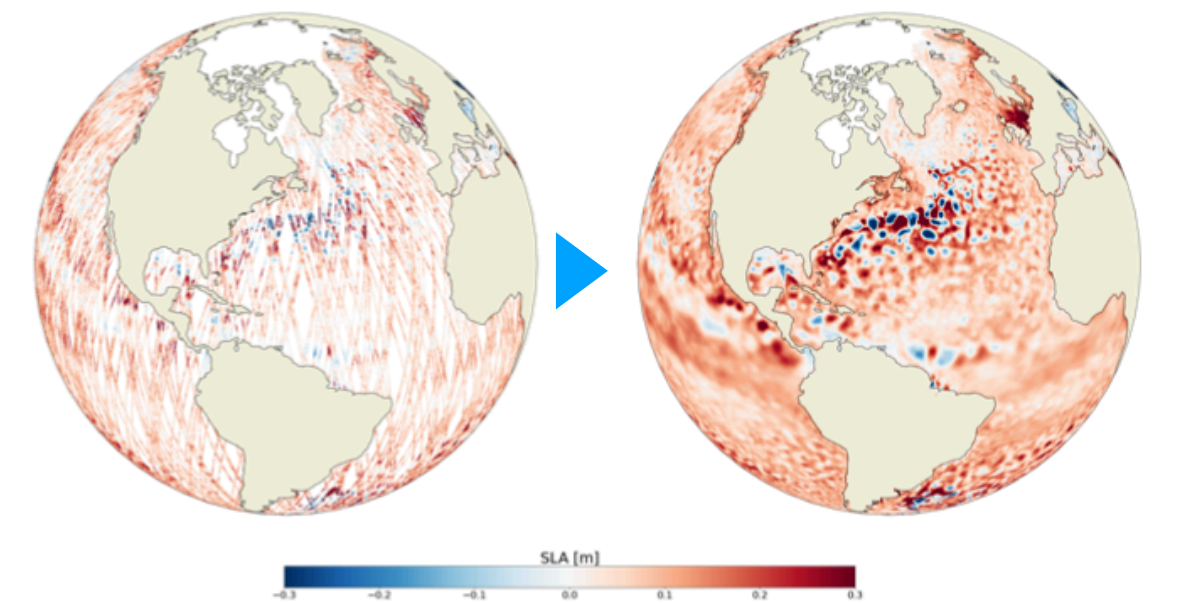
interdisciplinary

collab.
 IGE, IMT-Atl,
 Datlas, CLS

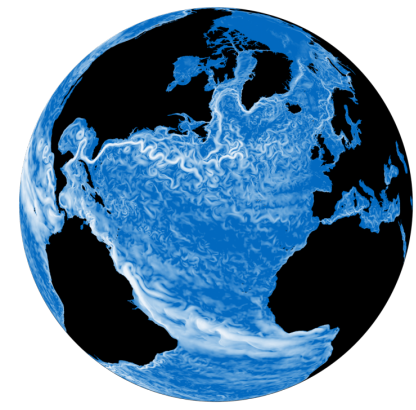
supported by
 CNES, CMEMS



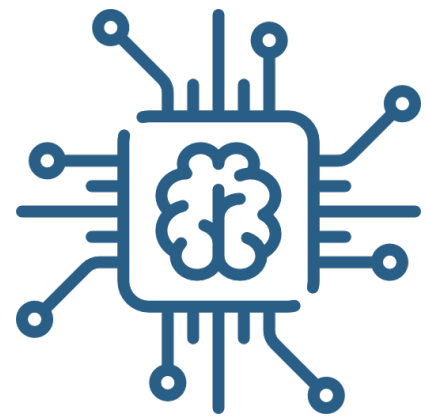
SSH mapping



Summary



=



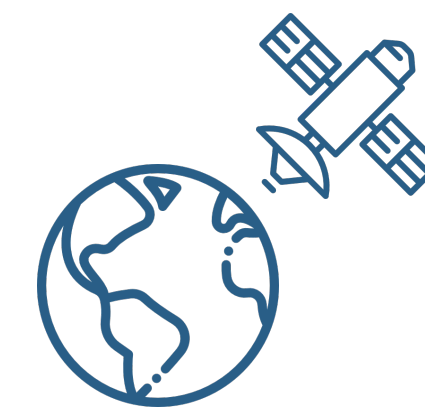
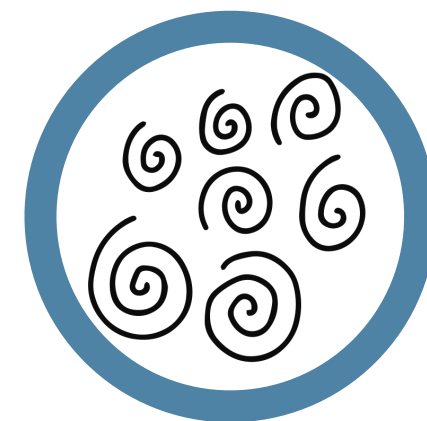
+



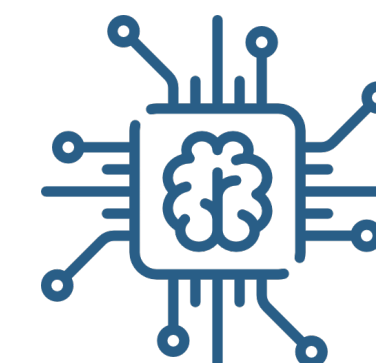
+



- Illustrated why we are **augmenting models** with ML
- Described how this is **done in practice** today
- Advocated that a **deep recast** of our models is needed
- Discussed some steps towards **AI-native hybrid** models
- Including the need for large **cross-disciplinary** efforts



Observations



Models / AI

A cross-disciplinary event later today



GAP 2024 Grenoble Artificial Intelligence for Physical Sciences Workshop

Scientific seminars from 29 to 30 May 2024 at MaCi, Grenoble, France

Julia tutorial on 31 May 2024 at IMAG building, Grenoble, France

<https://gap2024.sciencesconf.org>

Starting today at 1:30PM