Modélisation et Observation des Océans: Enjeux et convergence avec l'IA

R. Fablet, Prof. IMT Atlantique
Lab-STICC, INRIA Team Odyssey

Joint work with many, many, … colleagues (Lab-STICC, LOPS, IGE, LEGI, LEMAR

Overview of the talk

- *Challenges in ocean observation, modeling and forecasting*

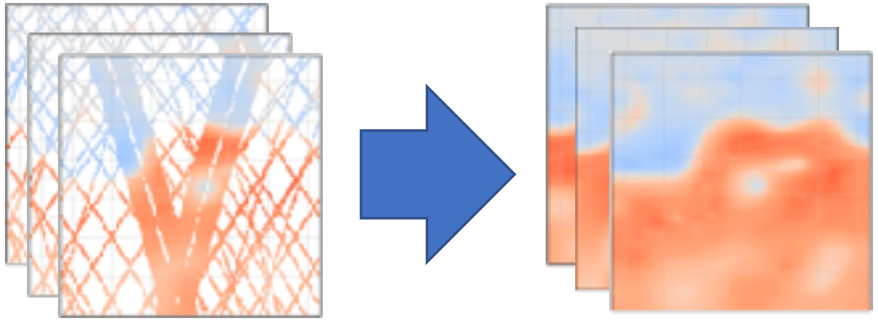
- *Beyond Neural Networks regarded as black boxes*

- *End-to-end learning can make a difference*

Challenges in Ocean Modeling and Forecasting

Challenges in AI & Ocean Modeling/Forecasting

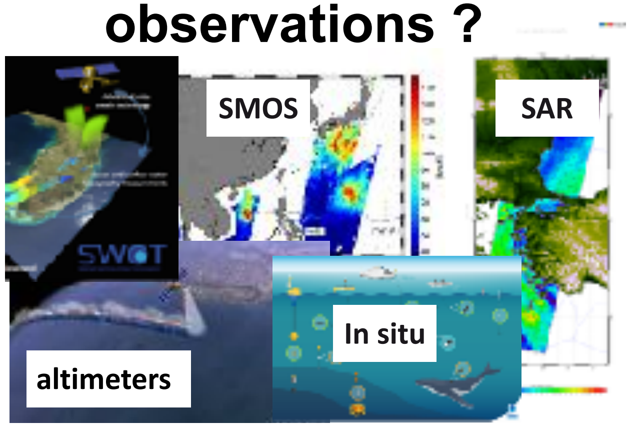
How to reconstruct processes ?



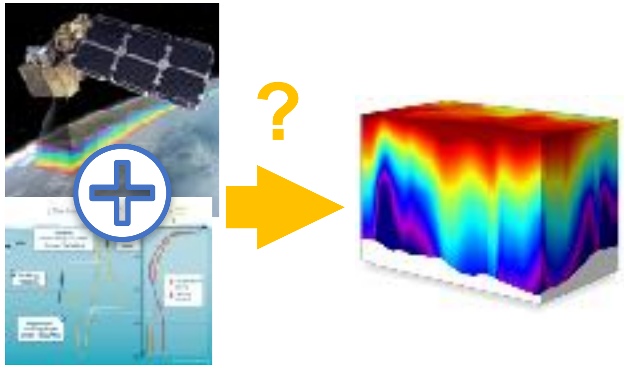
Observations y

Gap-free states x

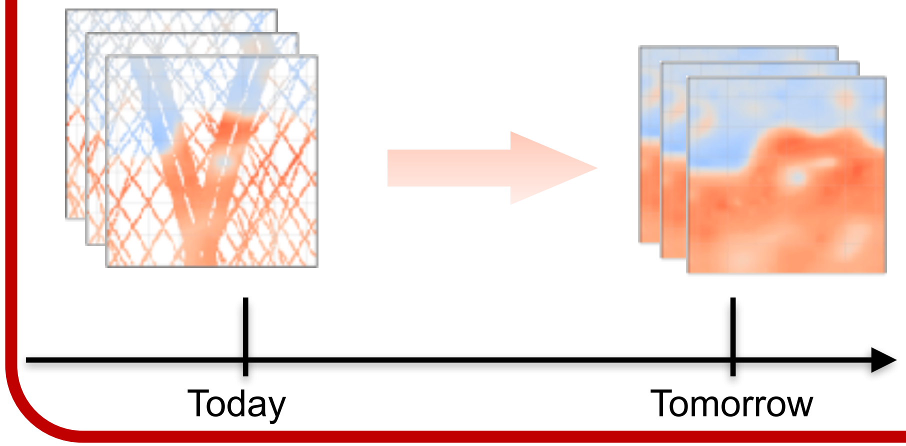
Using multi-source observations ?



From surface to interior ?



How to simulate/forecast?

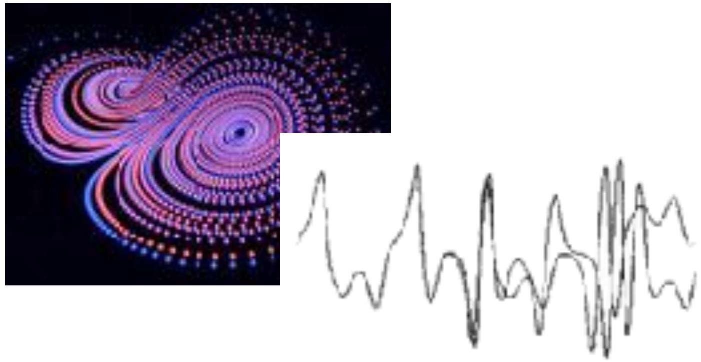


Where and what to sample ?

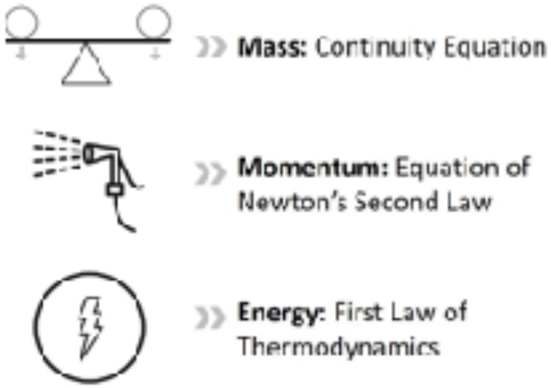
Beyond geophysical processes ?

Related methodological challenges

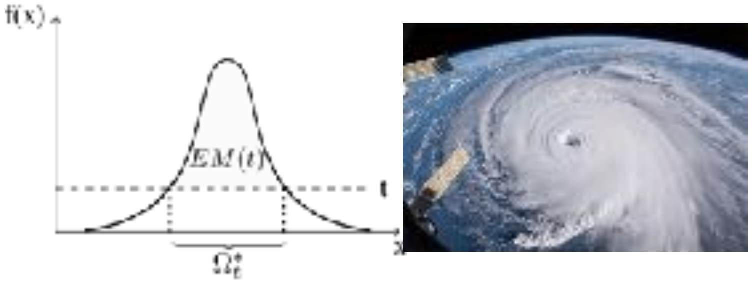
Dealing with chaos



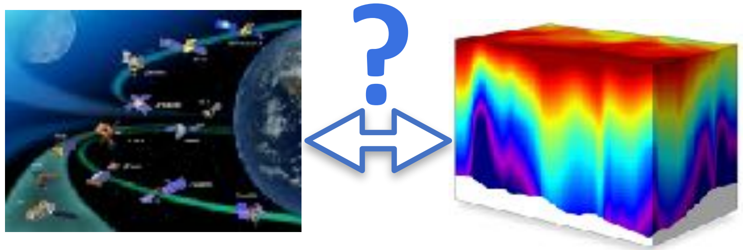
First principles



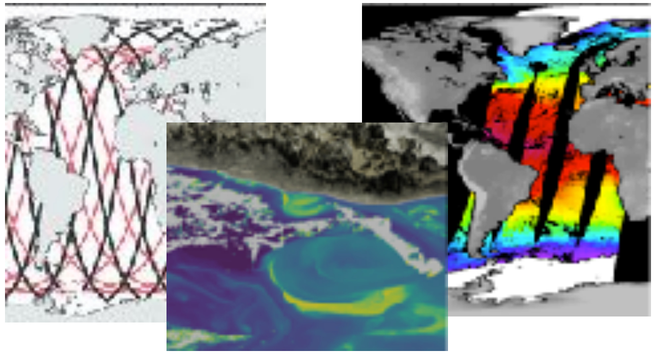
Dealing with extremes



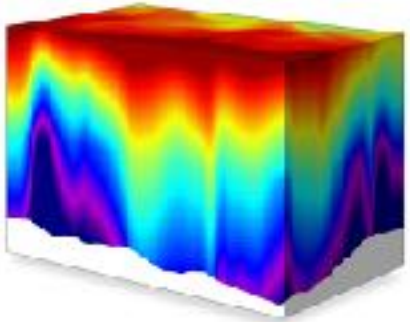
Bridging models and obs.



Irregular sampling

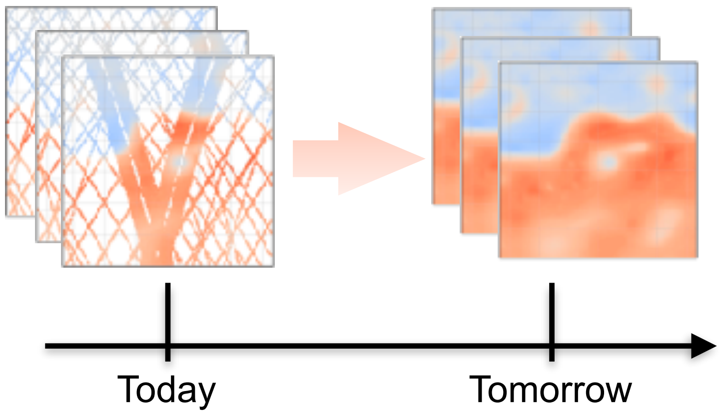


Identification of partially-observed systems

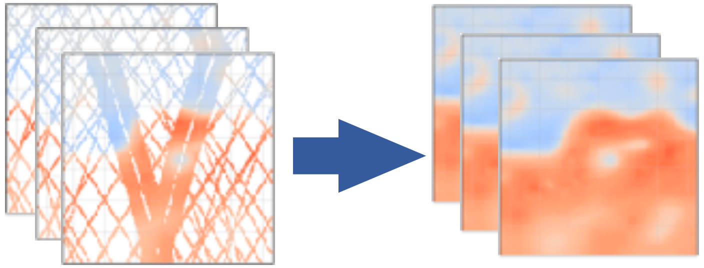


Why end-to-end learning and differentiable frameworks matter ?

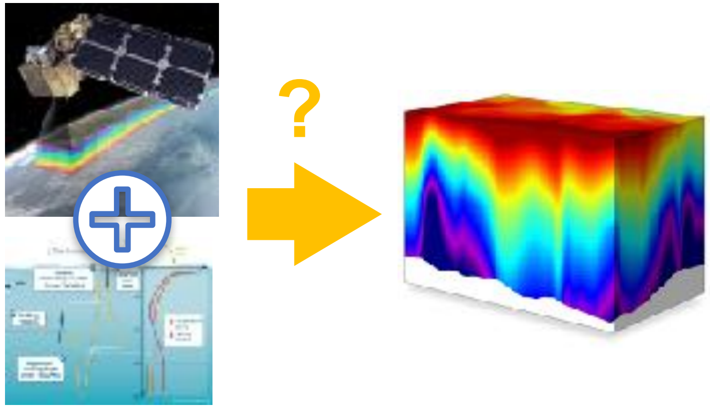
How to simulate/forecast?



How to reconstruct?



From surface to interior?



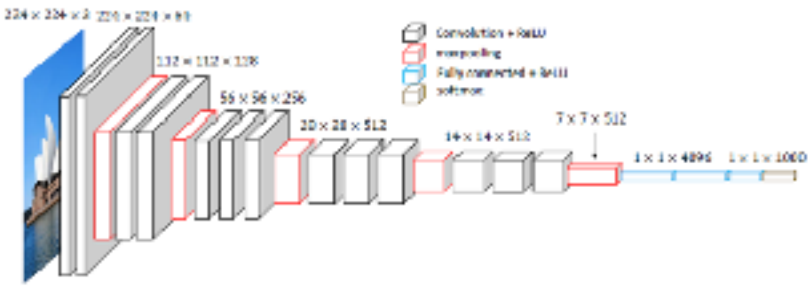
Deep Learning and Earth Sciences

What's (deep) learning (for a computer) ?

Input data

$$x_i$$

Deep neural network f_θ



Targeted output

$$y_i$$

Mathematical formulation:

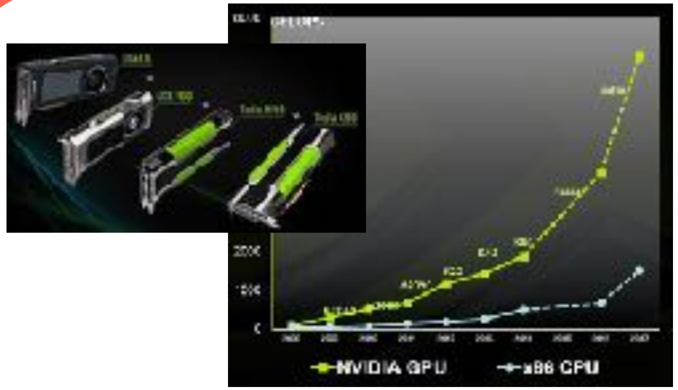
Learning comes to minimising some loss function given w.r.t. model parameters and training data

$$\widehat{\theta} = \arg\min_{\theta} \mathcal{L}\left(\{x_i, y_i\}_{i \in \{1,..,N\}}; f_\theta\right)$$

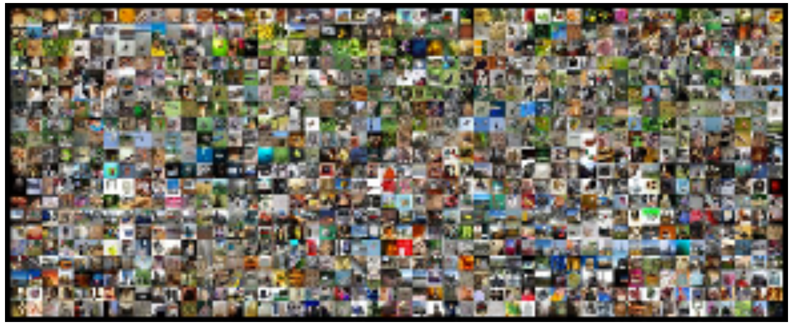
Neural networks as composition of **differentiable elementary functions**

$$f_\theta(x) = f_{\theta_N} \circ \ldots \circ f_{\theta_2} \circ f_{\theta_1}(x)$$

Key reasons for the emergence of DL

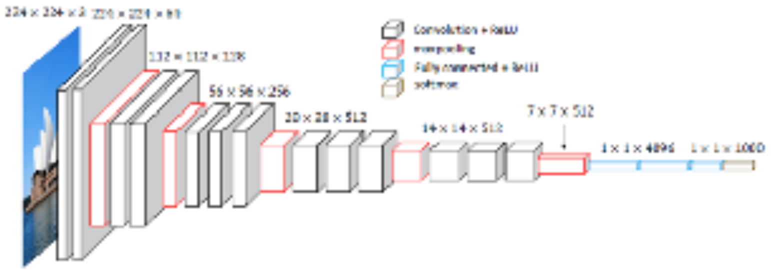


High-performance computing (GPU)



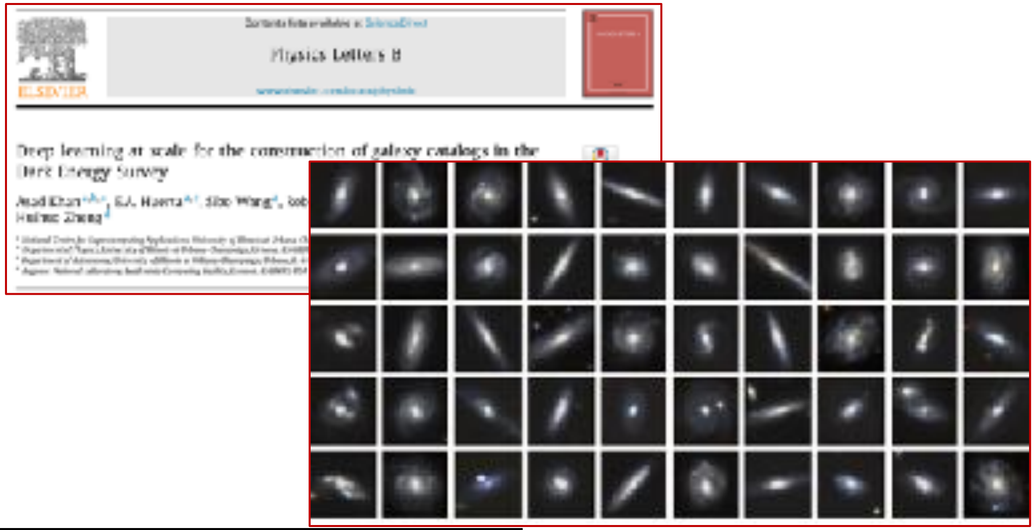
Large annotated dataset (> 1M)

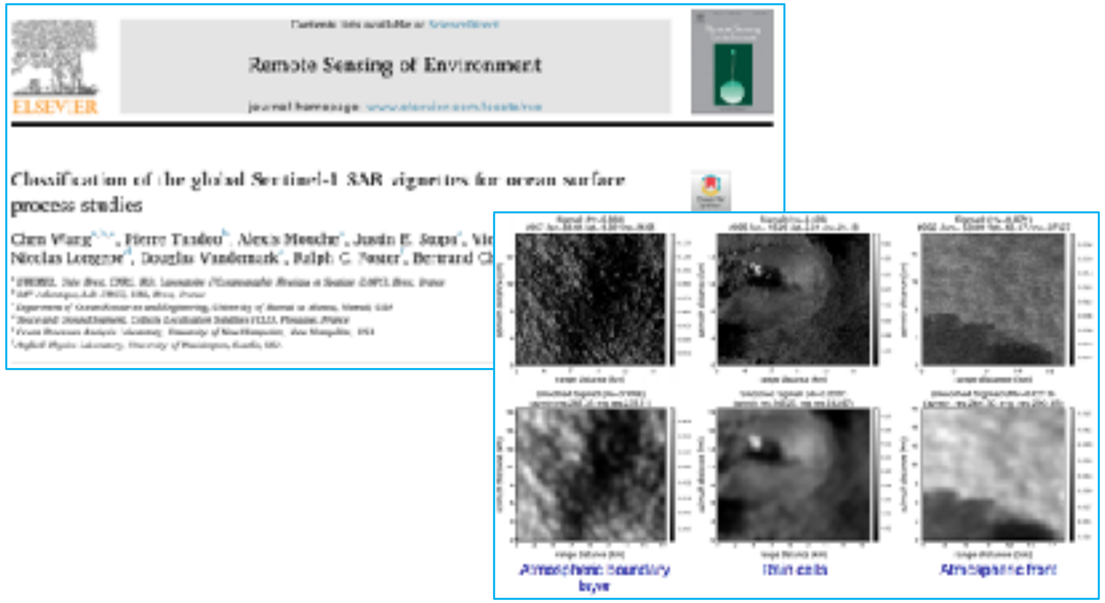
Efficient & easy-to-use frameworks

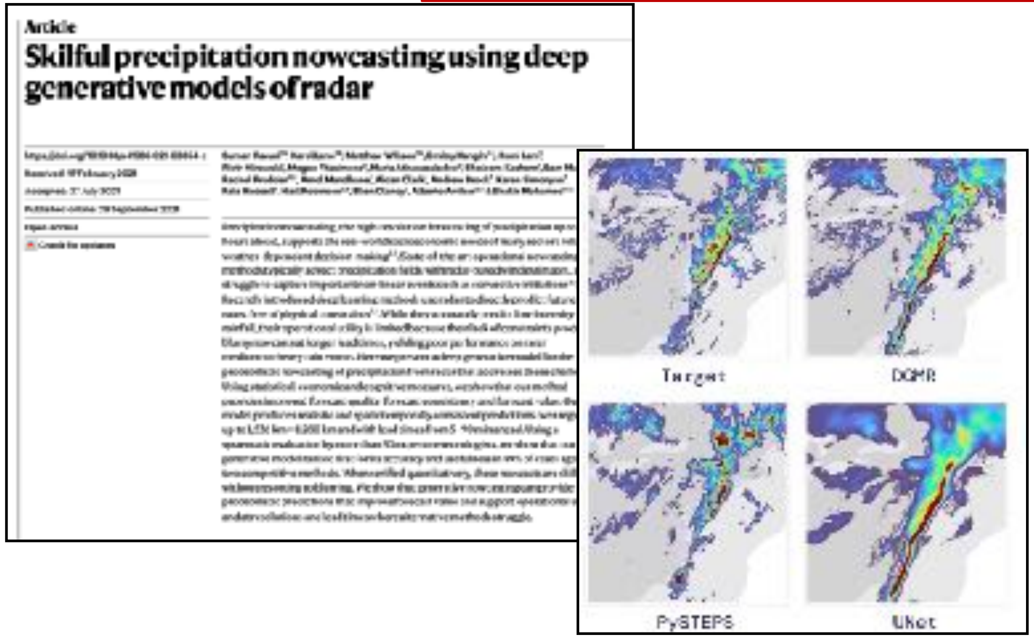


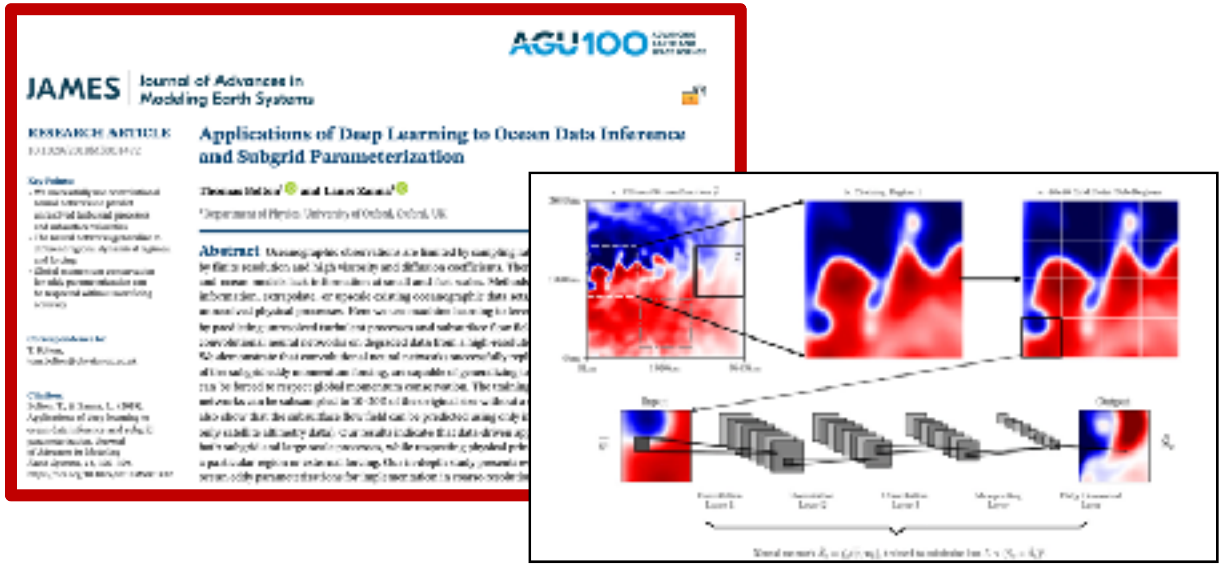
End-to-end learning

“State-of-the-art” DL schemes applied to physics-related problems



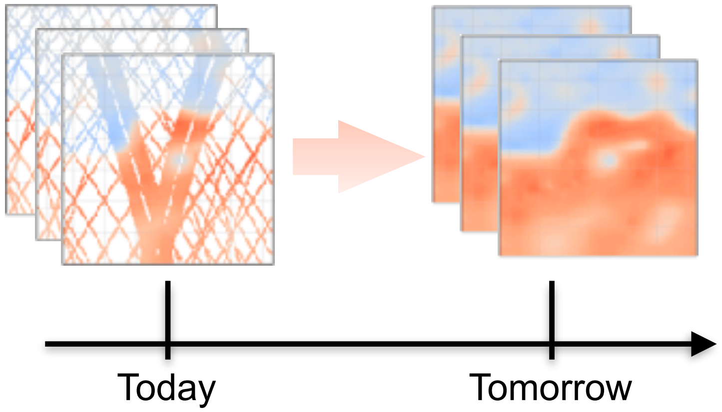




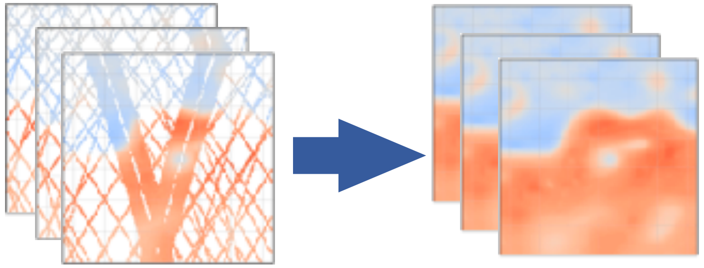


Why end-to-end learning and differentiable frameworks matter ?

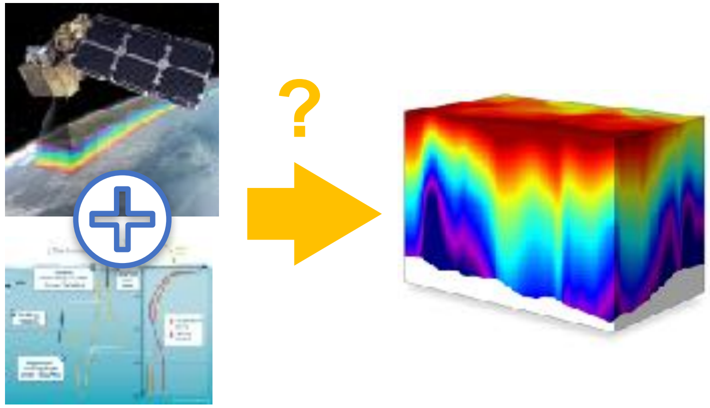
How to simulate/forecast?



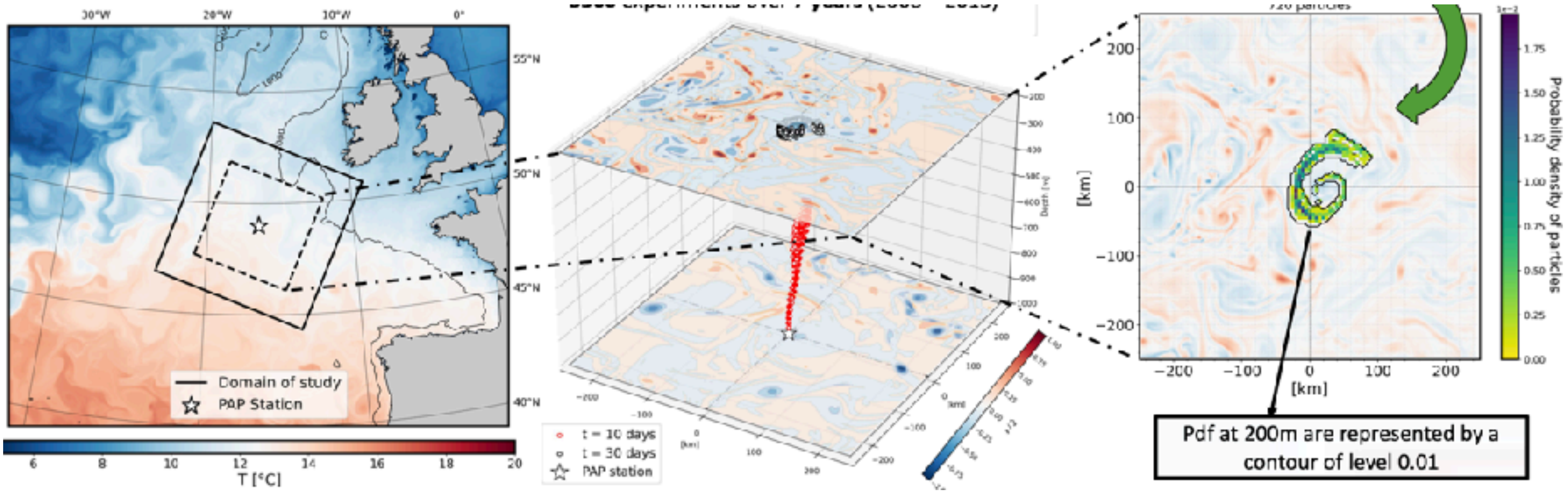
How to reconstruct?



From surface to interior?

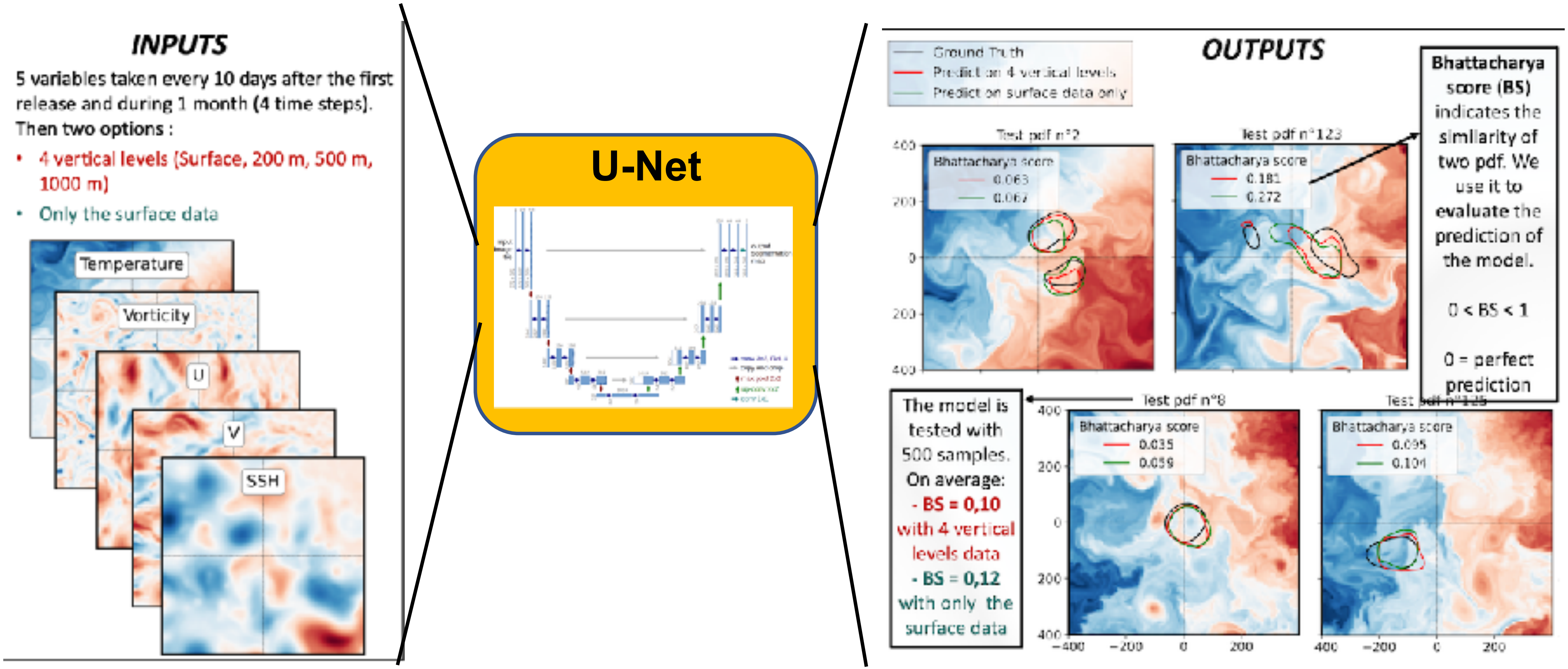


Ocean carbon cycle: where deep ocean particles come from ? [*Picard et al., In Prep*]

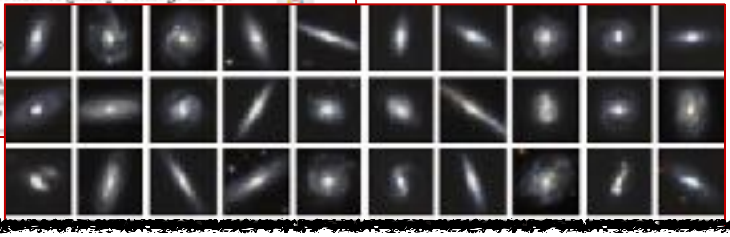


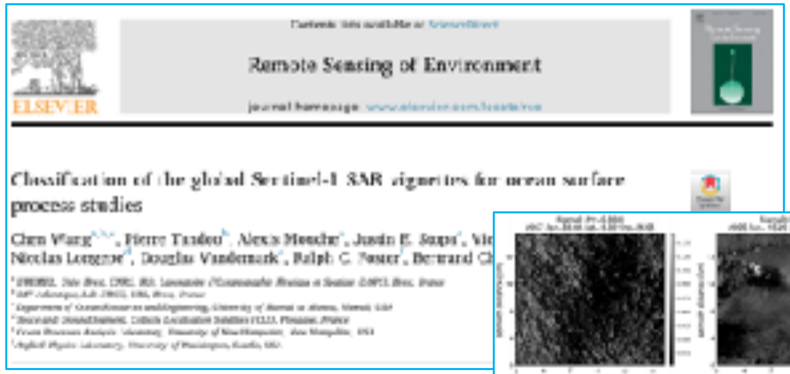
Can we learn to predict the origin of deep ocean particles from sea surface dynamics?

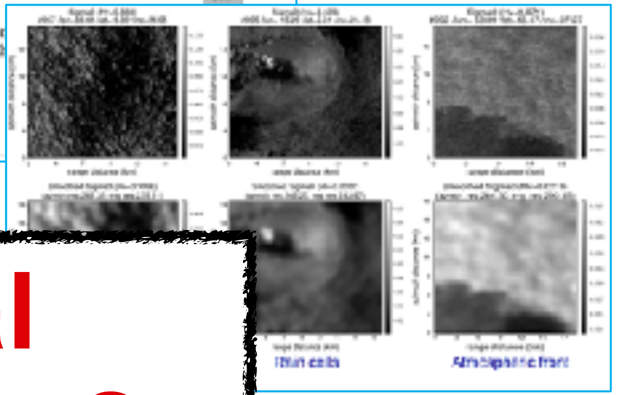
End-to-end learning from backward trajectory simulations [*Picard et al., In Prep*]



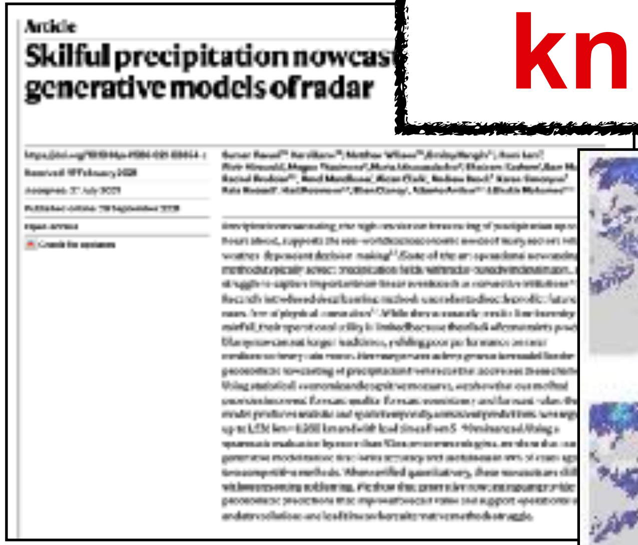
“State-of-the-art” DL schemes applied to physics-related problems

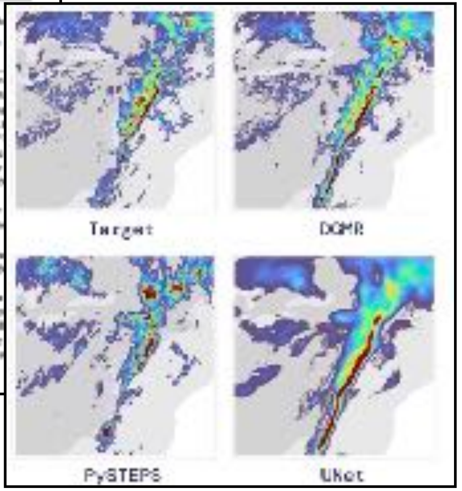


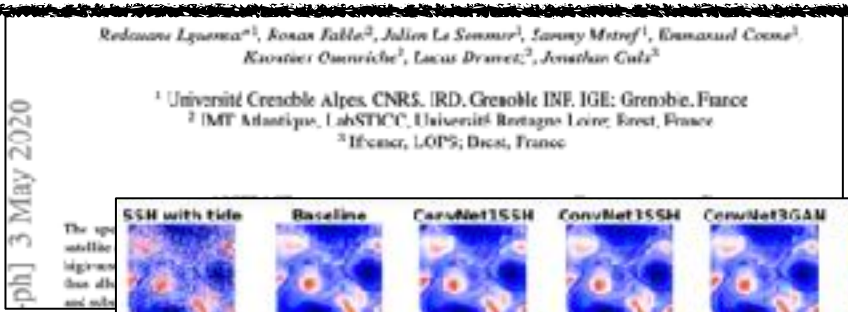


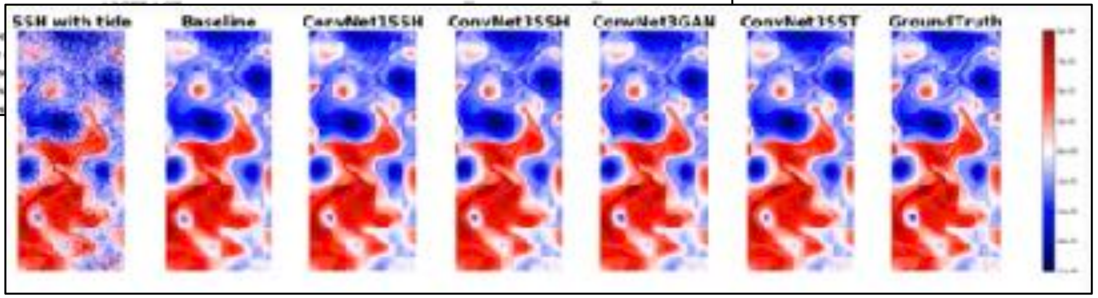


How to embed physical knowledge in DL schemes ?







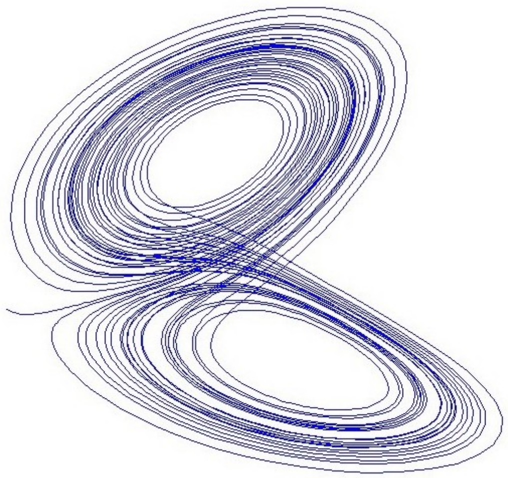


How to embed physics-driven priors in DL models ?

An illustration through neural ODE for Lorenz-63 dynamics (Fablet et al., 2018)

Can we learn a forecasting model from data ?

Can we exploit prior knowledge on the observed system ?



$$
\begin{aligned}
\frac{\mathrm{d}x(t)}{\mathrm{d}t} &= \sigma\,(y(t) - x(t)) \\
\frac{\mathrm{d}y(t)}{\mathrm{d}t} &= x(t)\,(\rho - z(t)) - y(t) \\
\frac{\mathrm{d}z(t)}{\mathrm{d}t} &= x(t)\,y(t) - \beta\,z(t)
\end{aligned}
$$

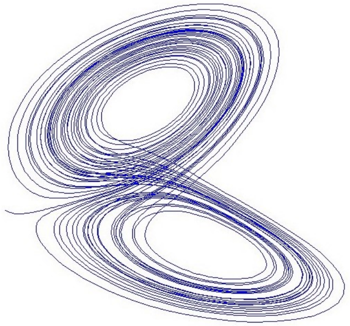
Lorenz-63 equations

How to embed physics-driven priors in DL models ?

An illustration through neural ODE for Lorenz-63 dynamics (Fablet et al., 2018)

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

Lorenz-63 equations

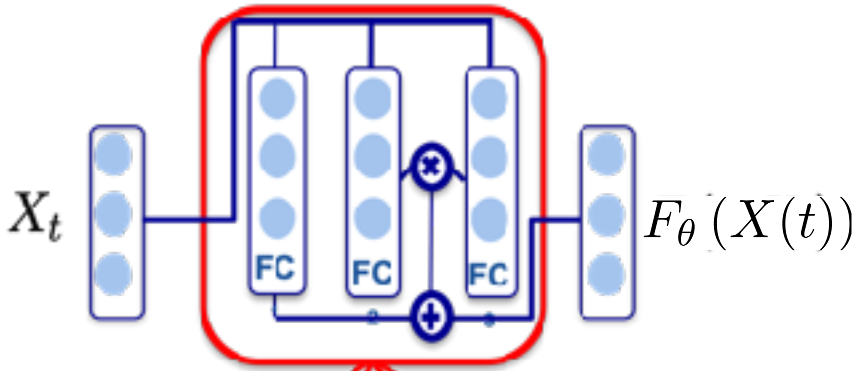


Associated Euler integration scheme for the ODE

$$d_t X(t) = F_\theta\left(X(t)\right)$$

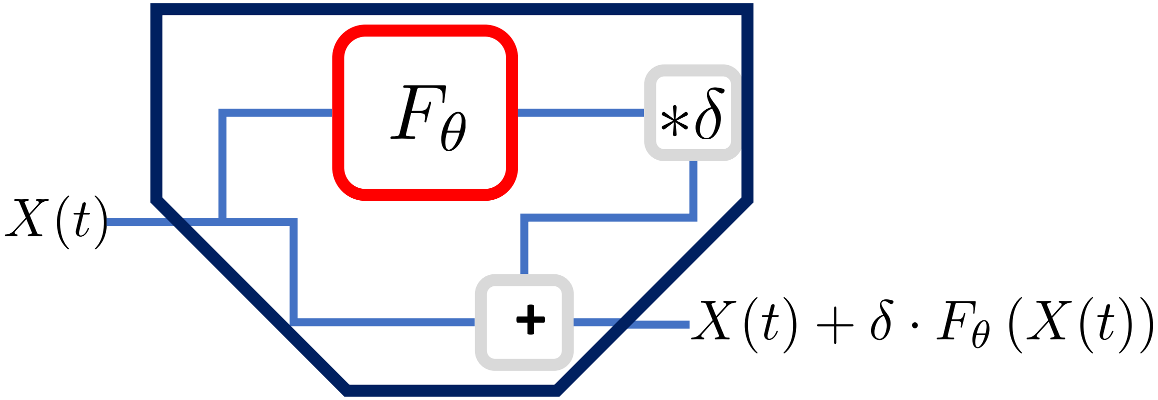
$$X(t+\delta) = X(t) + \delta \cdot F_\theta\left(X(t)\right)$$

NN architecture for differential operator



Bilinear architecture

NN architecture for integration scheme



ResNet architecture (Residual Network)

How to embed physics-driven priors in DL models ?

An illustration through L63 dynamics: numerical experiments

Identification of the governing equation from data

Forecasting experiments

Noise-free training data

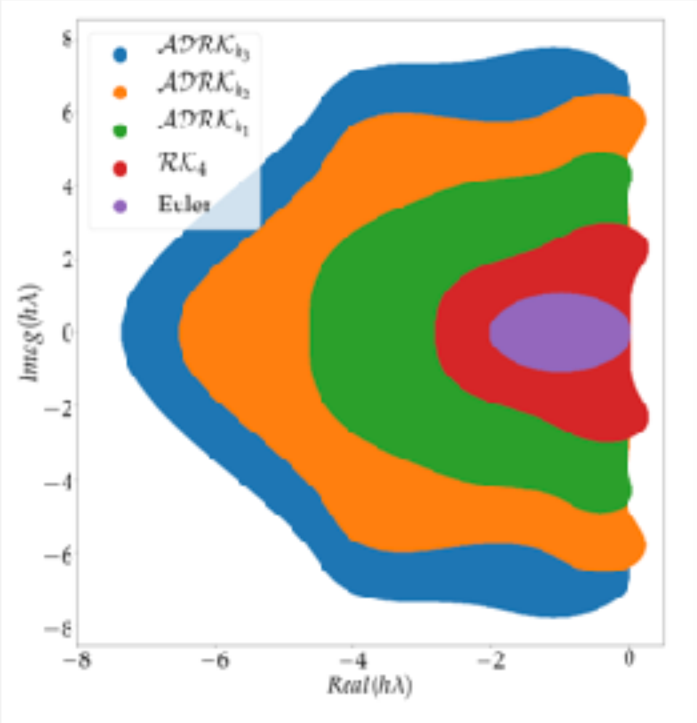
Forecasting time step	t_0+h	t_0+4h	t_0+8h
Analog forecasting	$<10^{-6}$	0.002	0.005
Sparse regression	$<10^{-6}$	0.002	0.006
MLP	$<10^{-6}$	0.018	0.044
Bi-NN(4)	***$<10^{-6}$***	***$<10^{-6}$***	***$<10^{-6}$***

Noisy training data (σ=0.5)

Forecasting time step	t_0+h	t_0+4h	t_0+8h
Analog forecasting	$<10^{-6}$	2.01	2.2
Bi-NN(4)	**$<10^{-6}$**	***0.054***	***0.14***

Fablet et al., 2018

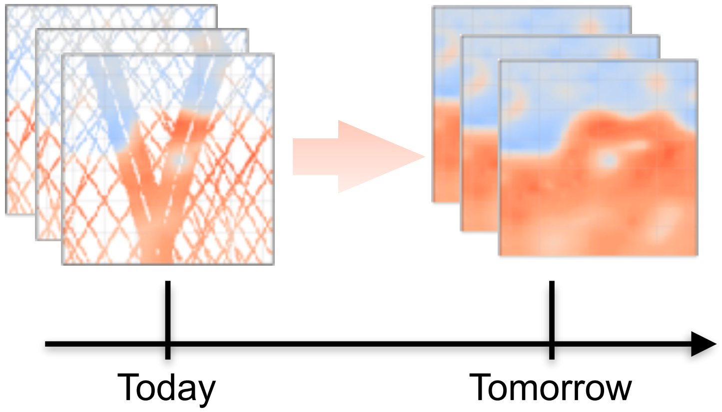
Stability of the learned RK schemes



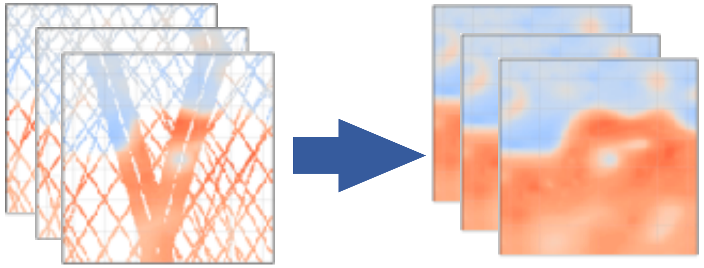
Ouala et al., 2020

Why end-to-end physics-informed learning and differentiable frameworks matter ?

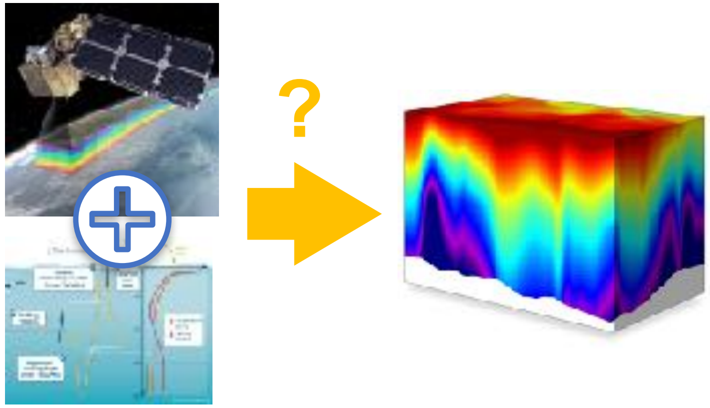
How to simulate/forecast?



How to reconstruct?



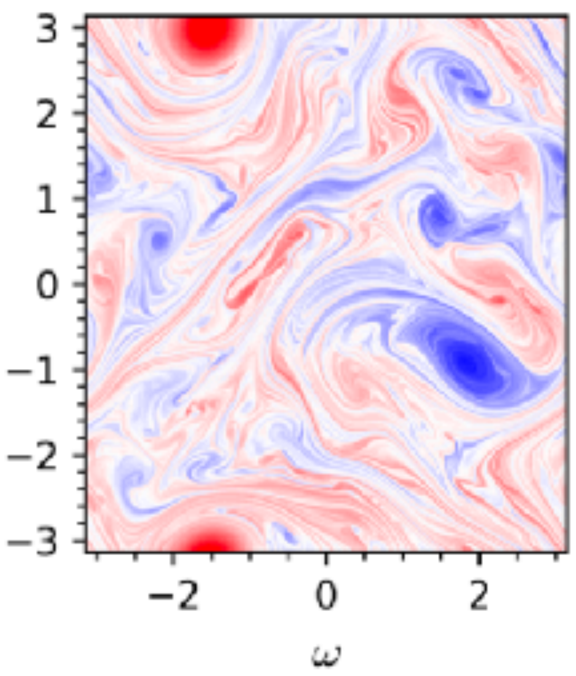
From surface to interior?



A posteriori learning for subgrid-scale parameterisation [Frezat et al., 2022]

Example: quasi-geostrophic turbulence

$$\partial_t \omega + J(\psi, \omega) = \nu \nabla^2 \omega - \mu \omega + F \qquad (1)$$

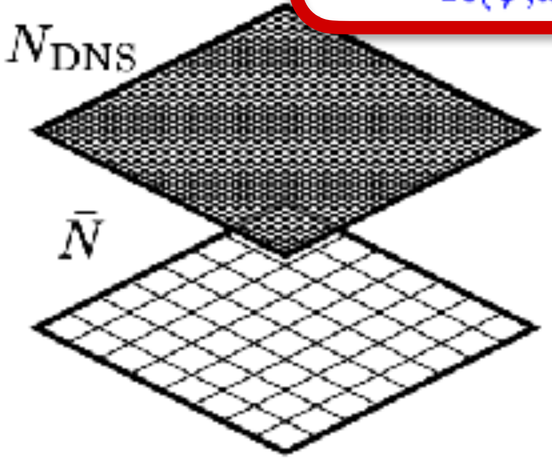


Vorticity field in quasi-geostrophic DNS.

Reduced equations

- Applying projection $\mathcal{T}(\mathbf{y}) = \bar{\mathbf{y}}$.
- Additional term must be modeled:

$$\partial_t \bar{\omega} + J(\bar{\psi}, \bar{\omega}) = \qquad (2)$$
$$\nu \nabla^2 \bar{\omega} - \mu \bar{\omega} + \bar{F} + \boxed{\underbrace{J(\bar{\psi}, \bar{\omega}) - \overline{J(\psi, \omega)}}_{\mathbf{R}(\psi, \omega)}}$$



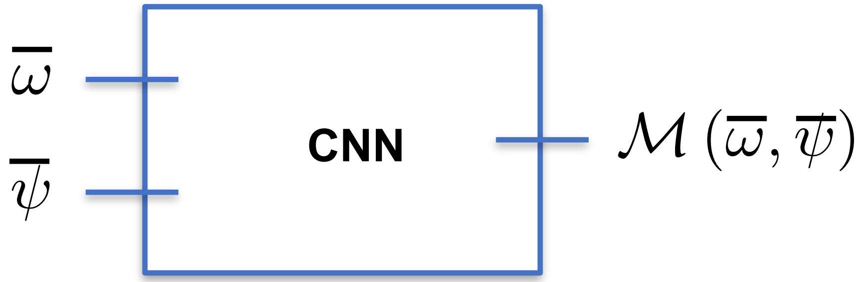
Grid sizes on domain length L.

Learning paradigms for subgrid-scale parameterisation

A priori learning [eg, Maulik et al., 2019; Zanna and Bolton, 2020]

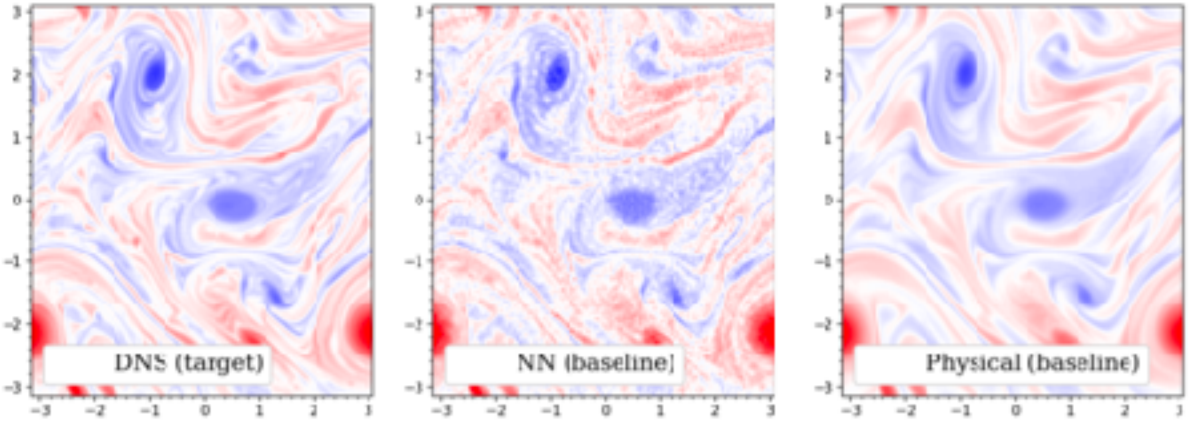
$$\min_{\mathcal{M}} \sum_{k} \left\| \mathcal{R}\left(\omega_k, \psi_k\right) - \mathcal{M}\left(\overline{\omega}_k, \overline{\psi}_k\right) \right\|^2$$

Empirical subgrid-scale term from a DNS dataset



No evaluation of the impact of the SGS term onto the simulated states

May lead to stable or unstable simulations

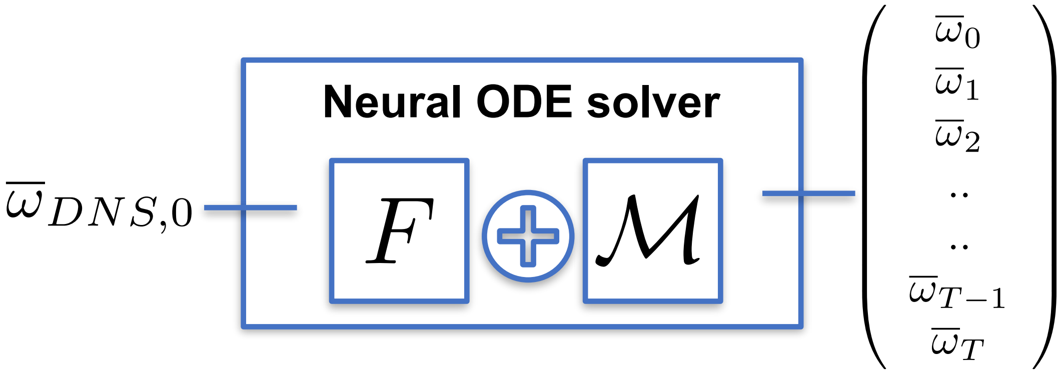


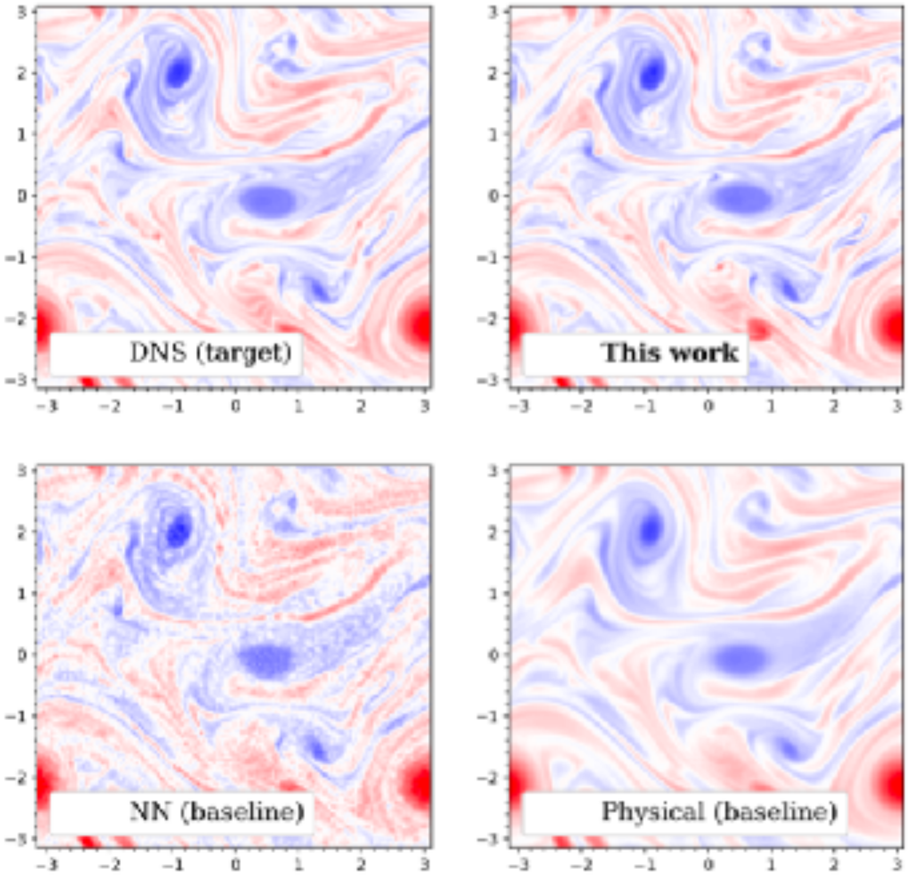
Learning paradigms for subgrid-scale parameterisation

A posteriori (end-to-end) learning [Frezat at al. 2022]

$$\min_{\mathcal{M}} \sum_{t,k} \left\| \mathcal{T}_k\left(\overline{\omega}_t\right) - \mathcal{T}_k\left(\overline{\omega}_{DNS,t}\right)\right\|^2$$

$$\begin{cases} \overline{\omega}_0 & = & \overline{\omega}_{DNS,0} \\ \partial_t \overline{\omega}_t & = & F\left(\overline{\omega}_t, \overline{\psi}_t\right) + \mathcal{M}\left(\overline{\omega}_t, \overline{\psi}_t\right) \end{cases}$$

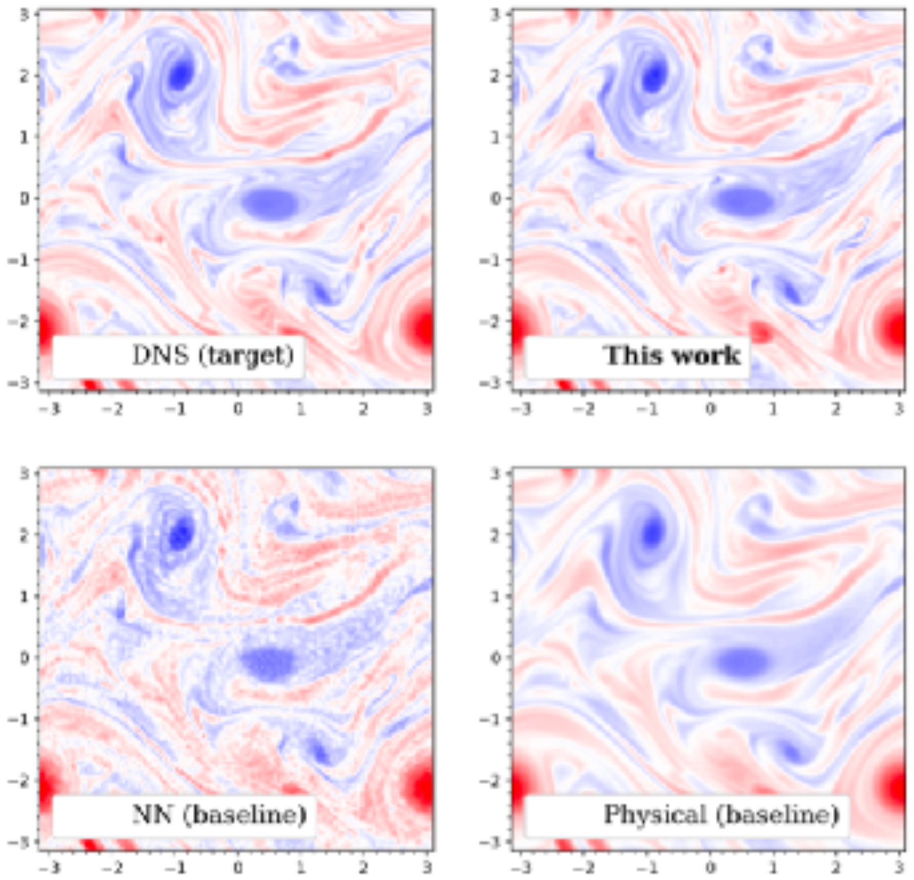




More details: https://arxiv.org/abs/2204.03911

Key messages

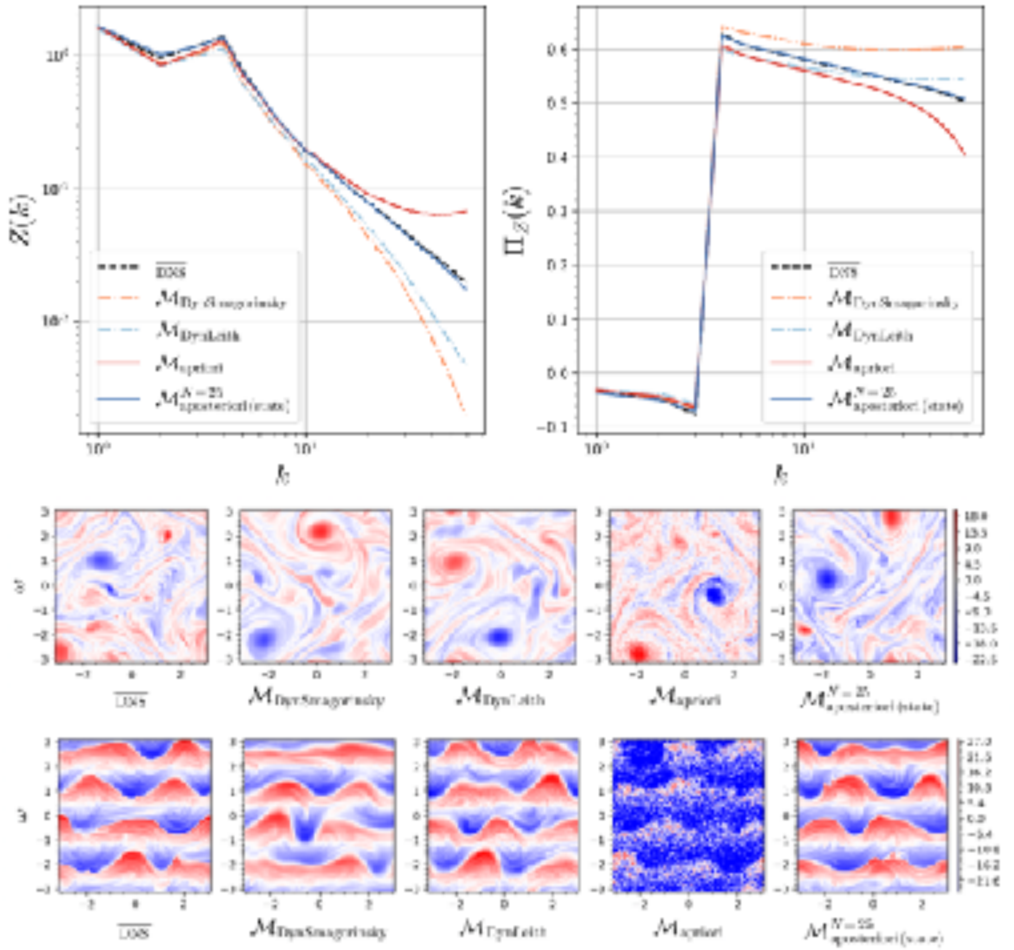
- NN-based parameterisation and calibration of subgrid-scale model
- End-to-end learning to account for a posteriori learning metrics
- Requirement for a differentiable version of the forward model
- How to address SGS model for ODGCM (Nemo, Croco,…)?



Paper (doi): https://arxiv.org/abs/2204.03911
Code: coming soon

Key messages

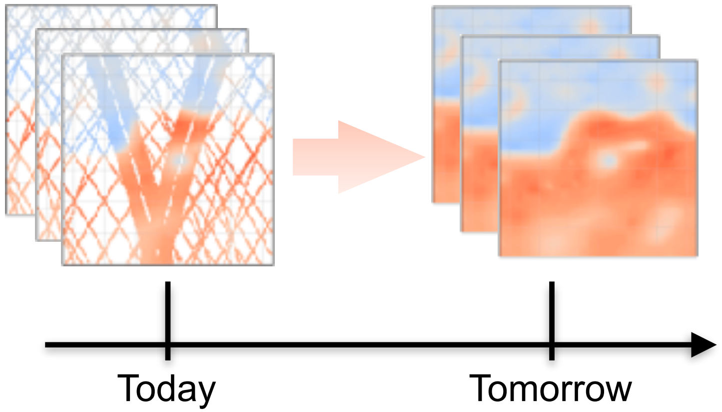
- NN-based parameterisation and calibration of subgrid-scale model
- End-to-end learning to account for a posteriori learning metrics
- Requirement for a differentiable version of the forward model
- How to address SGS model for ODGCM (Nemo, Croco,…)?



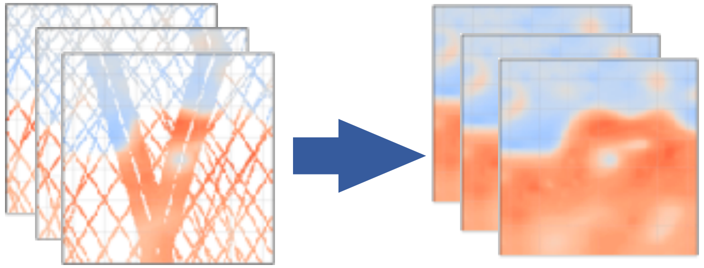
Paper (doi): https://arxiv.org/abs/2204.03911
Code: coming soon

Why end-to-end physics-informed learning and differentiable frameworks matter ?

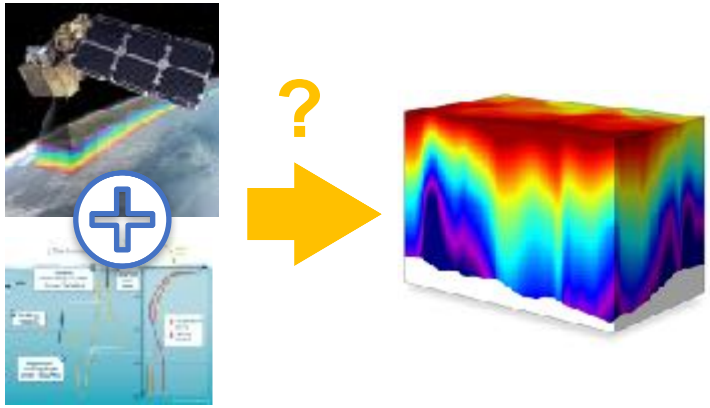
How to simulate/forecast?



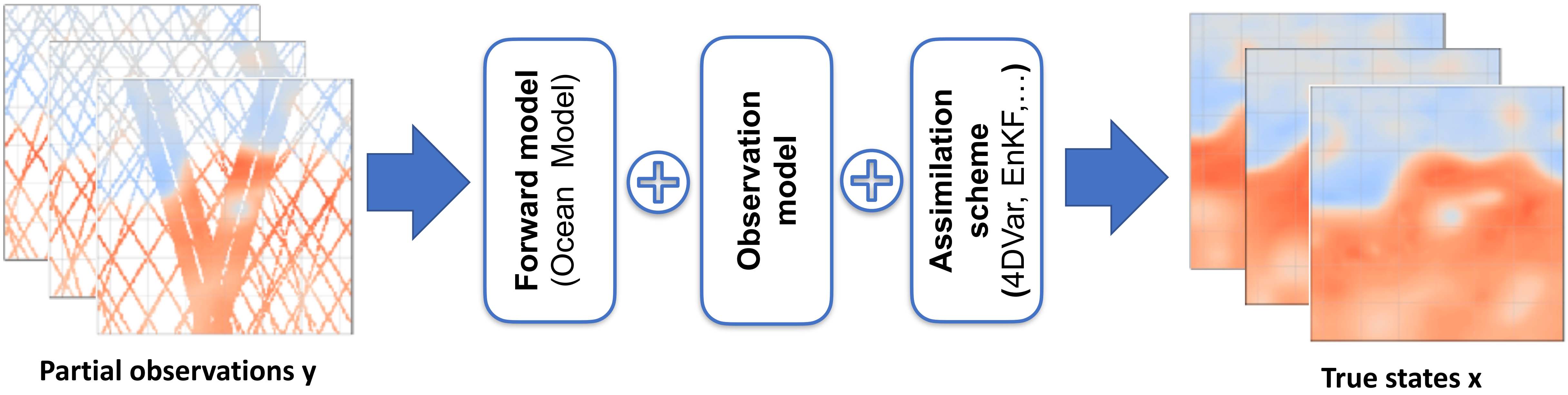
How to reconstruct?



From surface to interior?

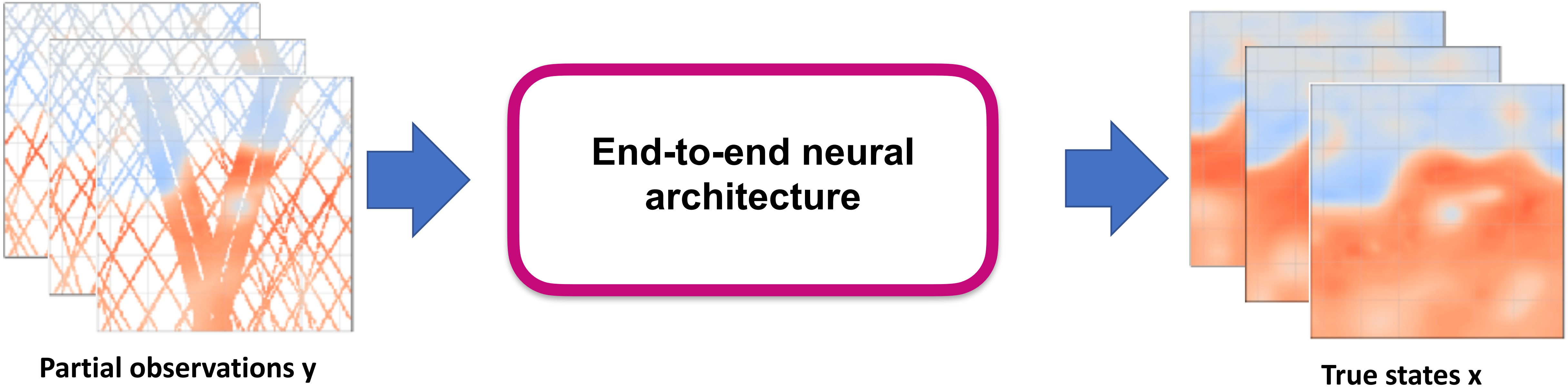


Data assimilation in geoscience [Evensen, 2000]



Each component designed using model-driven principles and mostly independently…. But limited ability to fully exploit observation datasets.

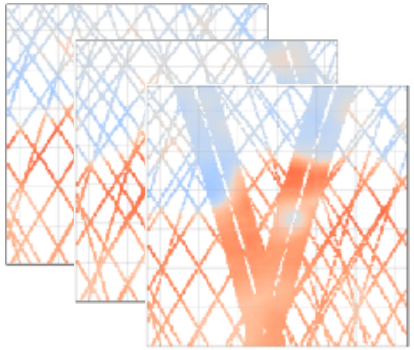
What about end-to-end learning for data assimilation?



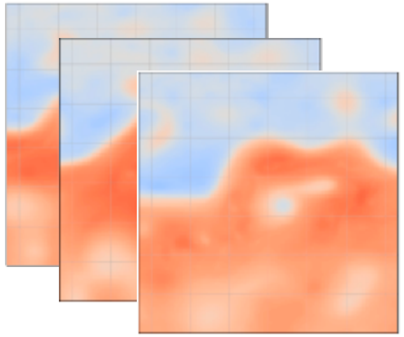
Can we exploit prior knowledge ?

Can we calibrate all the components of a DA scheme at once?

(Weak constraint) **4DVar Data Assimilation (DA) formulation**



Partial observations y



True states x

State-space formulation:

$$\begin{cases} \dfrac{\partial x(t)}{\partial t} & = & \mathcal{M}\left(x(t)\right) \\[2ex] y(t) & = & x(t) + \epsilon(t), \forall t \in \{t_0, t_0 + \Delta t...., , t_0 + N\Delta t\} \end{cases}$$

Associated variational formulation:

$$\arg\min_x \lambda_1 \sum_i \|x(t_i) - y(t_i)\|^2_{\Omega_{t_i}} + \lambda_2 \sum_n \|x(t_i) - \Phi(x)(t_i)\|^2$$

with $$\Phi(x)(t) = x(t-\Delta) + \int_{t-\Delta}^{t} \mathcal{M}\left(x(u)\right) du$$

$$\boxed{\arg\min_x \lambda_1 \|x - y\|^2_\Omega + \lambda_2 \|x - \Phi(x)\|^2}$$

4DVarNet: Learning 4DVar models and solvers

Trainable Variational DA formulation

$$\widehat{x} = \arg\min_{x} \|y - H(x)\|^2 + \lambda \|x - \Phi(x)\|^2$$

Trainable or pre-defined observation model

Trainable or pre-defined prior

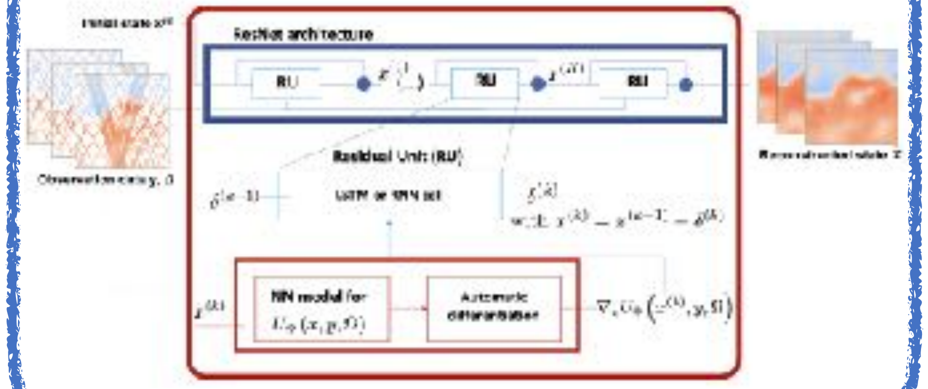
Trainable solver

$$x^{(k+1)} = x^{(k)} - \mathcal{H}\left[\nabla_x U_\Phi\left(x^{(k)}, y\right)\right]$$

RNN

Automatic differentiation

End-to-end architecture



Preprint: https://arxiv.org/abs/2006.03653
Code: https://github.com/CIA-Oceanix/DinAE_4DVarNN_torch

4DVarNet: Learning 4DVar models and solvers

Trainable Variational DA formulation

$$\widehat{x} = \arg\min_x \|y - H(x)\|^2 + \lambda \|x - \Phi(x)\|^2$$

Trainable or pre-defined observation model

Trainable or pre-defined prior

Learning criterion

Variational cost (non-supervised)

Reconstruction error (supervised)

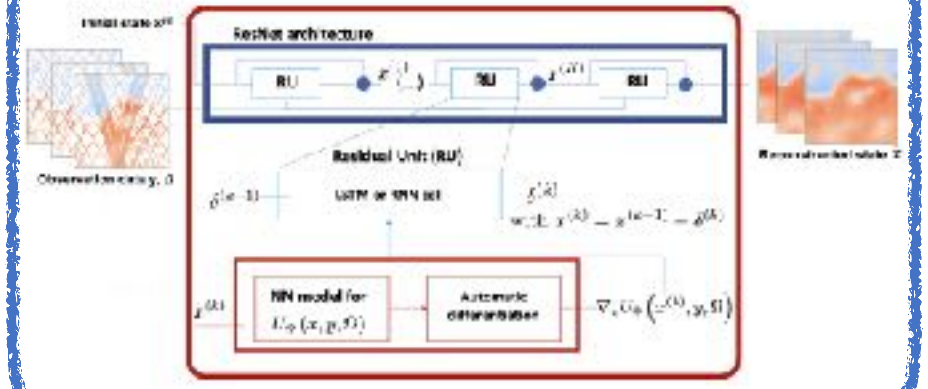
Trainable solver

$$x^{(k+1)} = x^{(k)} - \mathcal{H}\left[\nabla_x U_\Phi \left(x^{(k)}, y\right)\right]$$

RNN

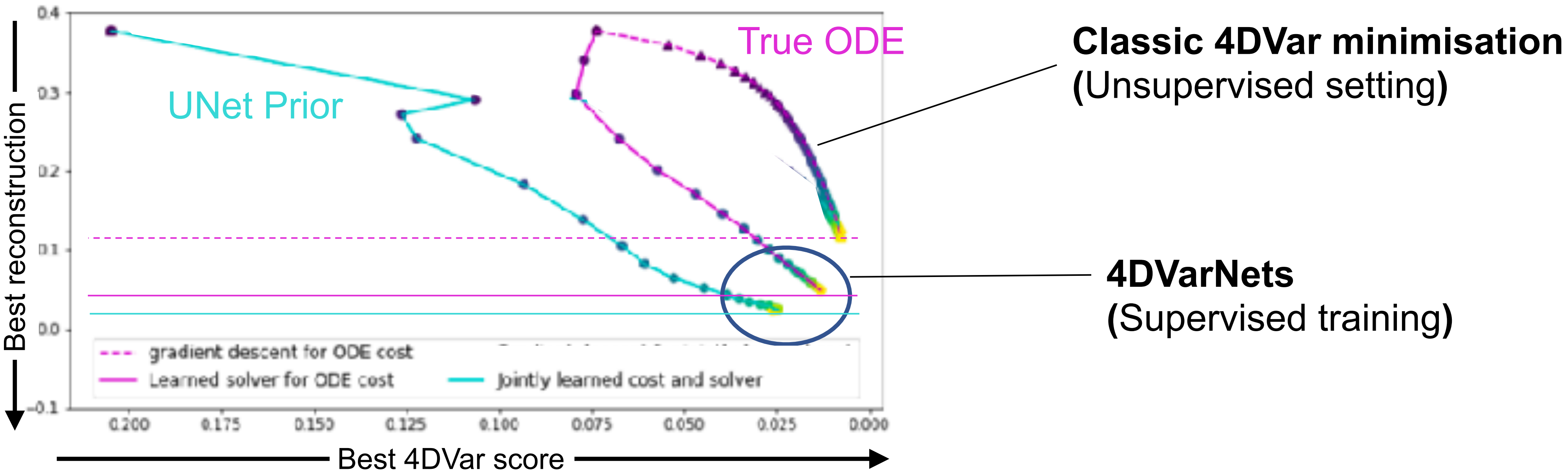
Automatic differentiation

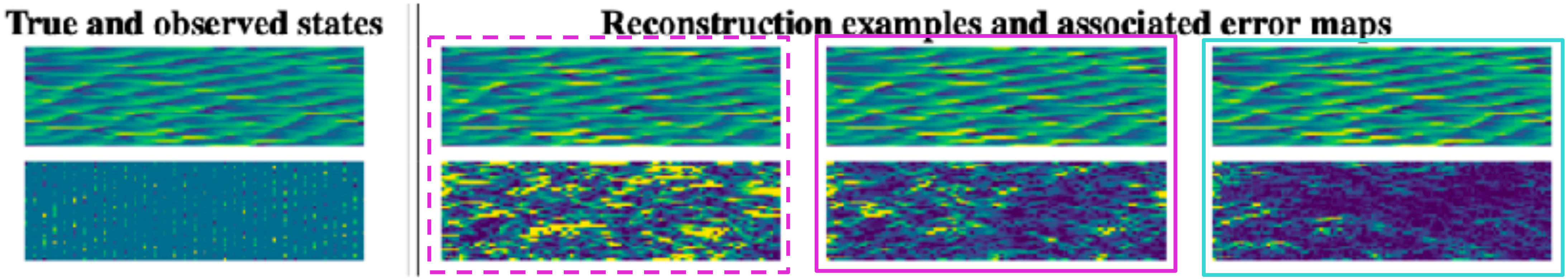
End-to-end architecture



Preprint: https://arxiv.org/abs/2006.03653
Code: https://github.com/CIA-Oceanix/DinAE_4DVarNN_torch

Application to Lorenz-96 system





The true dynamical model may not be the best prior for DA ?

4DVarNet: Application to sea surface dynamics

Method

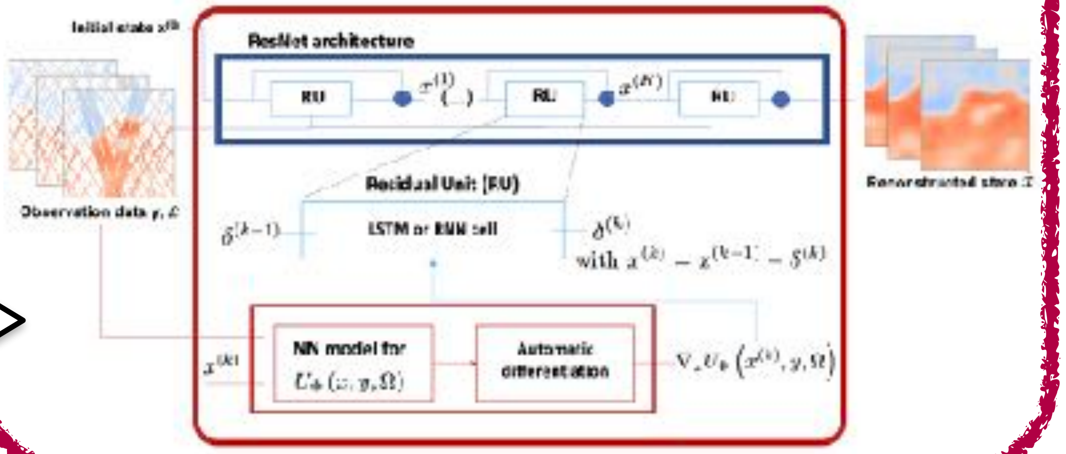
From a Variational DA formulation

$$\widehat{x} = \arg\min_x \|y - H(x)\|^2 + \lambda \|x - \Phi(x)\|^2$$

Trainable variational model

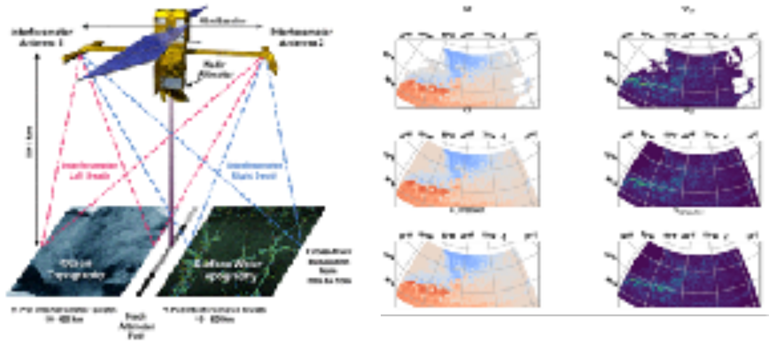
Trainable gradient-based solver

Associated end-to-end scheme

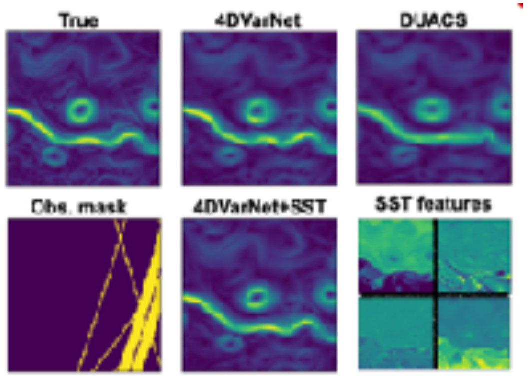


Applications

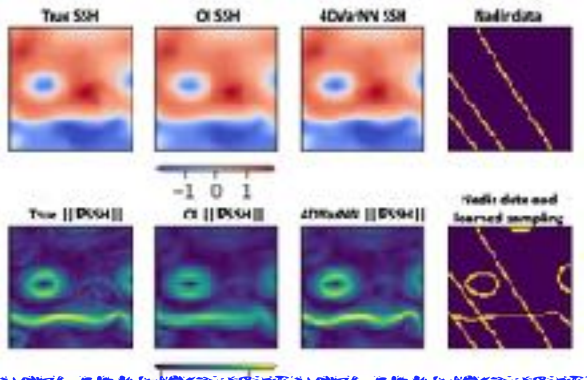
Reconstruction & Forecasting



Multimodal DA

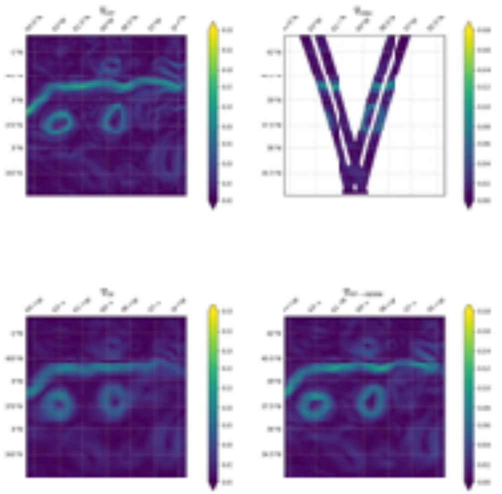


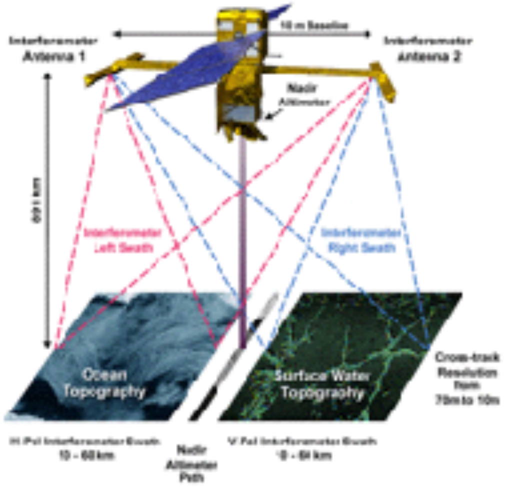
Learning where to sample ?



Reconstruction of satellite-derived sea surface dynamics

Satellite altimetry





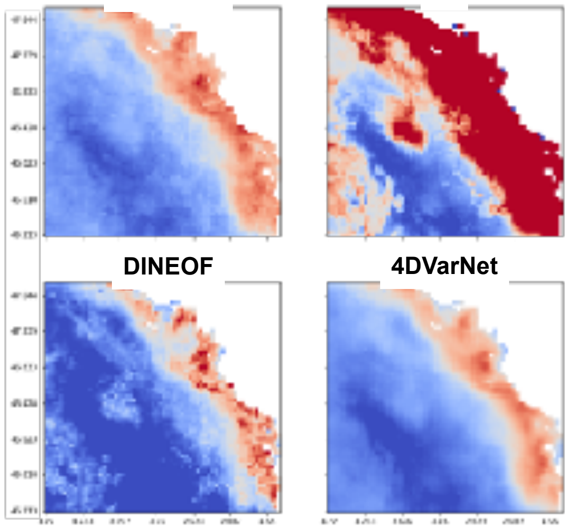
Best score for BOOST-SWOT SLA Data Challenge

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	10.14	Multiscale mapping	eval_miost.ipynb
4DVarNet 1 swot + 4 nadirs 🏆	0.95	0.01	0.82	6.67	4DVarNet mapping	eval_4dvarnet.ipynb

https://github.com/ocean-data-challenges/2020a_SSH_mapping_NATL60

Sea surface suspended sediments

Metric	Dataset	Unit	Samp. Strat	OI	DinEOF	4DVarNet
RMSE	OSSE	$\log_{10}[g/L]/m$	-	0.176	0.167	**0.104**
	MODIS	$\log_{10}[g/L]/m$	Random	0.304	0.237	**0.156**
	MODIS	$\log_{10}[g/L]/m$	Patch	0.346	0.253	**0.168**
R-score	OSSE	%	-	90.4	91.3	**96.6**
	MODIS	%	Random	60.5	76.4	**89.5**
	MODIS	%	Patch	56.5	73.8	**87.3**



Mean gradient norm

Simulation and real data

Learning from real gappy data only

Collab. IMT Atl./SHOM/ LGO

4DVarNet: Dealing with uncertainties [Lafon et al., in prep]

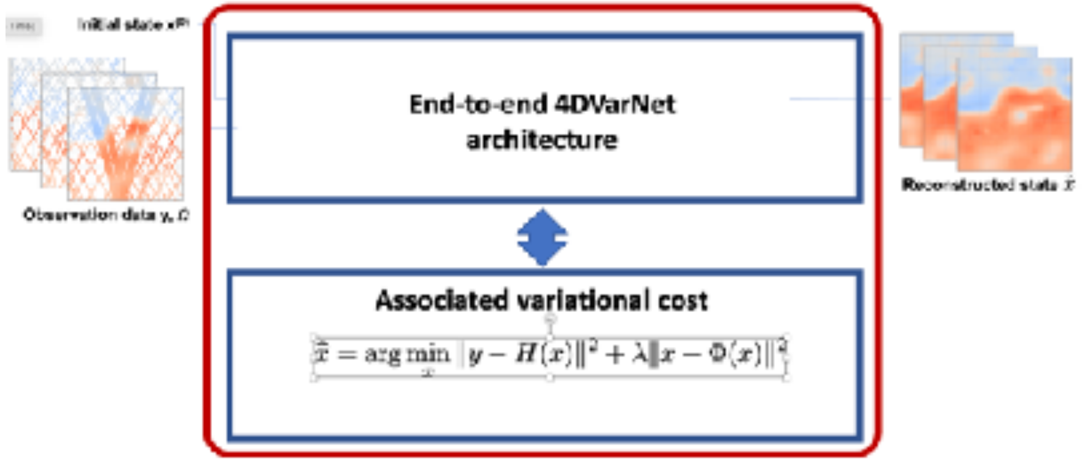
Physical state-space

$$(x, y)$$

4DVar formulation

$$\widehat{x} = \arg\min_{x} \|y - H(x)\|^2 + \lambda \|x - \Phi(x)\|^2$$

Associated end-to-end scheme



"Pdf" state-space

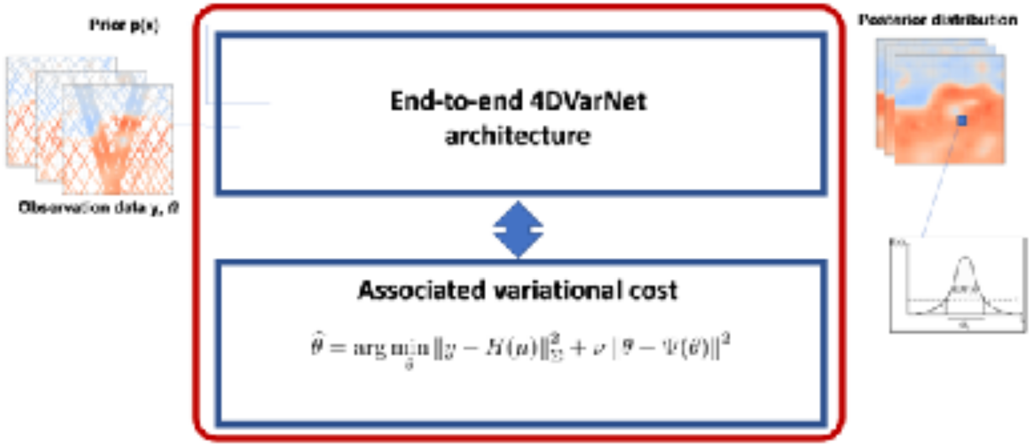
$$p(x|y) \approx q_\theta(x), y$$

4DVar formulation (ELBO)

$$\widehat{\theta} = \arg\min_{\theta} E_{q_\theta(x)}\left[p\left(y|x\right)\right] + KL\left[q_\theta(x)|p(x)\right]$$

$$\widehat{\theta} = \arg\min_{\theta} \|y - H(\mu)\|^2_{\Sigma} + \nu\, \|\theta - \Psi(\theta)\|^2$$

Associated end-to-end scheme



4DVarNet: Dealing with uncertainties [Lafon et al., in prep]

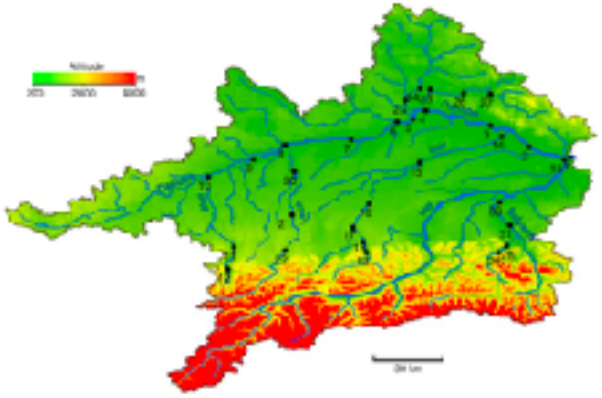


Figure: 31 gauging stations on the Danube river network (*Asadi et al.*, *2015*), with 50 years of daily measurements (1960-2010)

Application to the monitoring of the Danube river

- 15 stations over 31 with observations

- Reconstruction of the river flow at unobserved stations

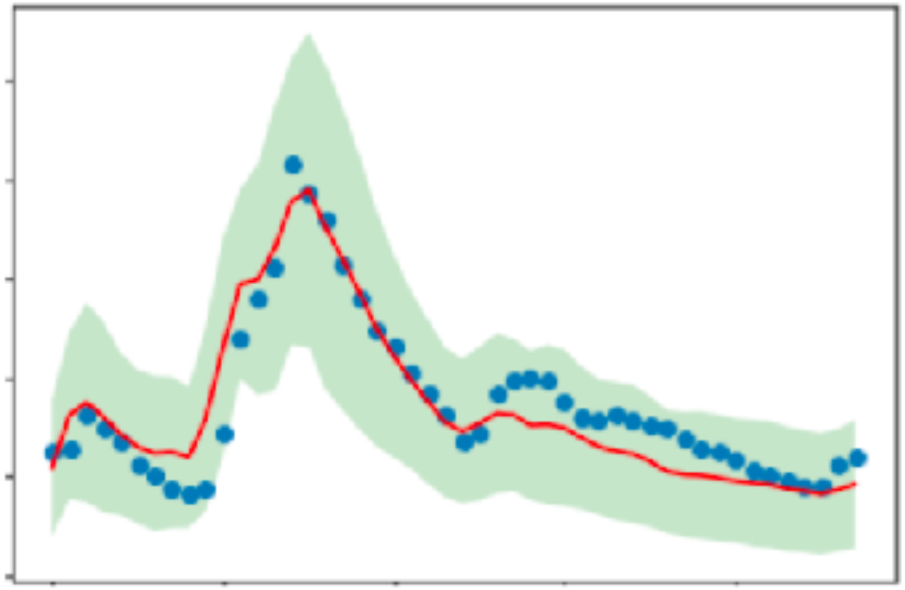
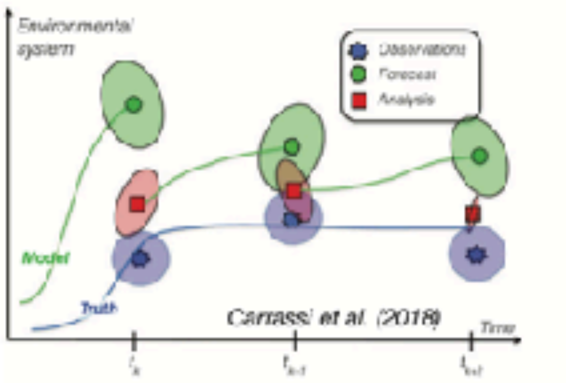


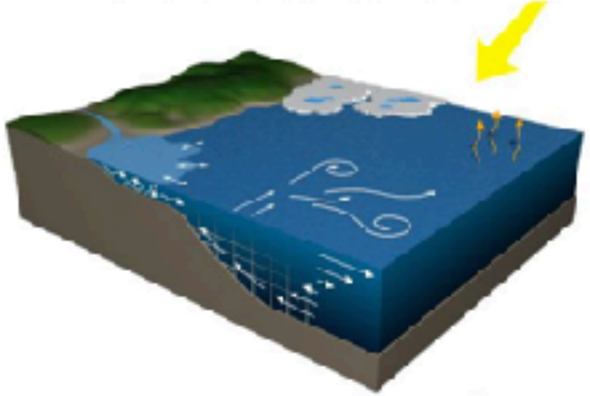
Figure: Hidden observation (blue dots), estimated mean (red curve) and 95% confidence interval

Take-home messages

- ***NNs as numerical schemes/solvers for ODE/PDE/energy-based representations of geophysical flows***
- ***End-to-end learning of governing equations and/or unknown terms in governing equations (e.g., closures)***
- ***Deep learning as a lego game breaking a given problem into a set of connected trainable differentiable components***

Earth System Models / Digital Twins / AI





Observations → Data assimilation → Physics-based forecast → Post-processing, dissemination

Hybrid vs. native AI core engines ?

From model-centric systems to process-centric/problem-centric ones ?

So what for climate/ocean science?

Free your mind from unnecessary computation-related constraints ….

…. stick to relevant targeted concepts and questions

And think end-to-end

AI Chair OceaniX 2020-2024

Thank you.

Physics-informed AI for Observation-Driven Ocean AnalytiX

R. Fablet, Prof. IMT Atlantique, Brest
Web: https://cia-oceanix.github.io/

References

- **Model calibration/identification**
  - Frezat et al. Physics-informed neural networks for sub-grid scale modeling in filtered turbulence. PRF, 2021. https://arxiv.org/abs/2010.04663
  - Frezat et al. A posteriori learning for quasi-geostrophic turbulence parametrization. arXiv, 2022. https://arxiv.org/abs/2204.03911
  - Pannekoucke et al. PDE-NetGen 1.0: from symbolic PDE representations of physical processes to trainable neural network representations. GMD, 2020. https://gmd.copernicus.org/articles/13/3373/2020/
  - Ouala et al. Learning Runge-Kutta Integration Schemes for ODE Simulation and Identification, aRxiv, 2021. https://arxiv.org/abs/2105.04999
- **Data assimilation and Forecasting**
  - Ouala et al. Learning Latent Dynamics for Partially-Observed Chaotic Systems. Chaos, 2020. https://arxiv.org/abs/1907.02452
  - Fablet et al. Joint interpolation and representation learning for irregularly-sampled satellite-derived geophysical fields. FAMS, 2021. https://doi.org/10.3389/fams.2021.655224
  - Fablet et al. Learning Variational Data Assimilation Models and Solvers. JAMES, 2021. https://arxiv.org/abs/2007.12941
- **Trajectory data analysis and modeling**
  - Nguyen et al. GeoTrackNet-A Maritime Anomaly Detector using Probabilistic NN Representation of AIS Tracks and A Contrario Detection. IEEE TITS, 2020. https://arxiv.org/abs/1912.00682
  - Roy et al. Deep Learning and Trajectory Representation for the Prediction of Seabird Diving Behaviour. biorRxiv, 2020. https://www.biorxiv.org/content/10.1101/2021.04.19.438554v1
  - Nguyen et al. TrAISformer-A generative transformer for AIS trajectory prediction, arXiv, 2021. https://arxiv.org/abs/2109.03958
- **Applications to upper ocean dynamics**
  - Cazau et al. Multimodal deep learning for cetacean distribution modeling of fin whales in the western Mediterranean Sea. ML, 2021. https://link.springer.com/article/10.1007/s10994-021-06029-z
  - Benaichouche et al. Unsupervised Reconstruction of Sea Surface Currents from AIS Maritime Traffic Data Using Trainable Variational Models. RS, 2021. https://www.mdpi.com/2072-4292/13/16/3162
  - Vient et al. Data-Driven Interpolation of Sea Surface Suspended Concentrations Derived from Ocean Colour Remote Sensing Data. RS 2021. https://www.mdpi.com/2072-4292/13/17/3537
  - Beauchamp, et al. Intercomparison of data-driven and learning-based interpolations of along-track nadir and wide-swath altimetry observations. RS, 2020. https://www.mdpi.com/2072-4292/12/22/3806