Neural networks & dynamical systems: dealing with partially-observed systems

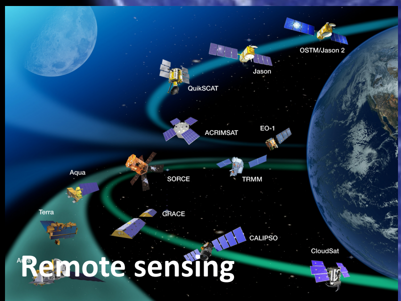
R. Fablet et al.
ronan.fablet@imt-atlantique.fr
web: rfablet.github.io

« Saisir le mouvement », Nov 2020

Lab-STICC
IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

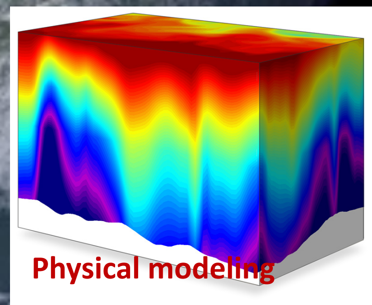
General question

How to solve sampling gaps and extract high-level information for ocean monitoring and surveillance ?

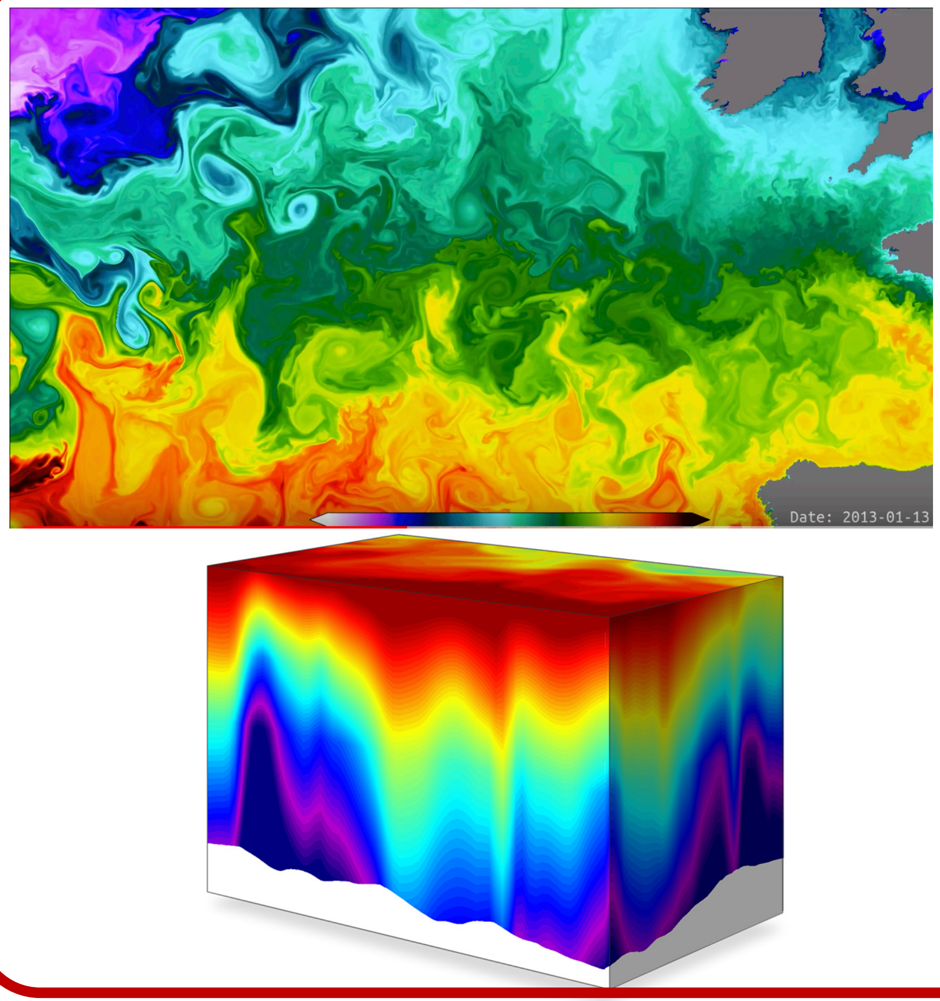


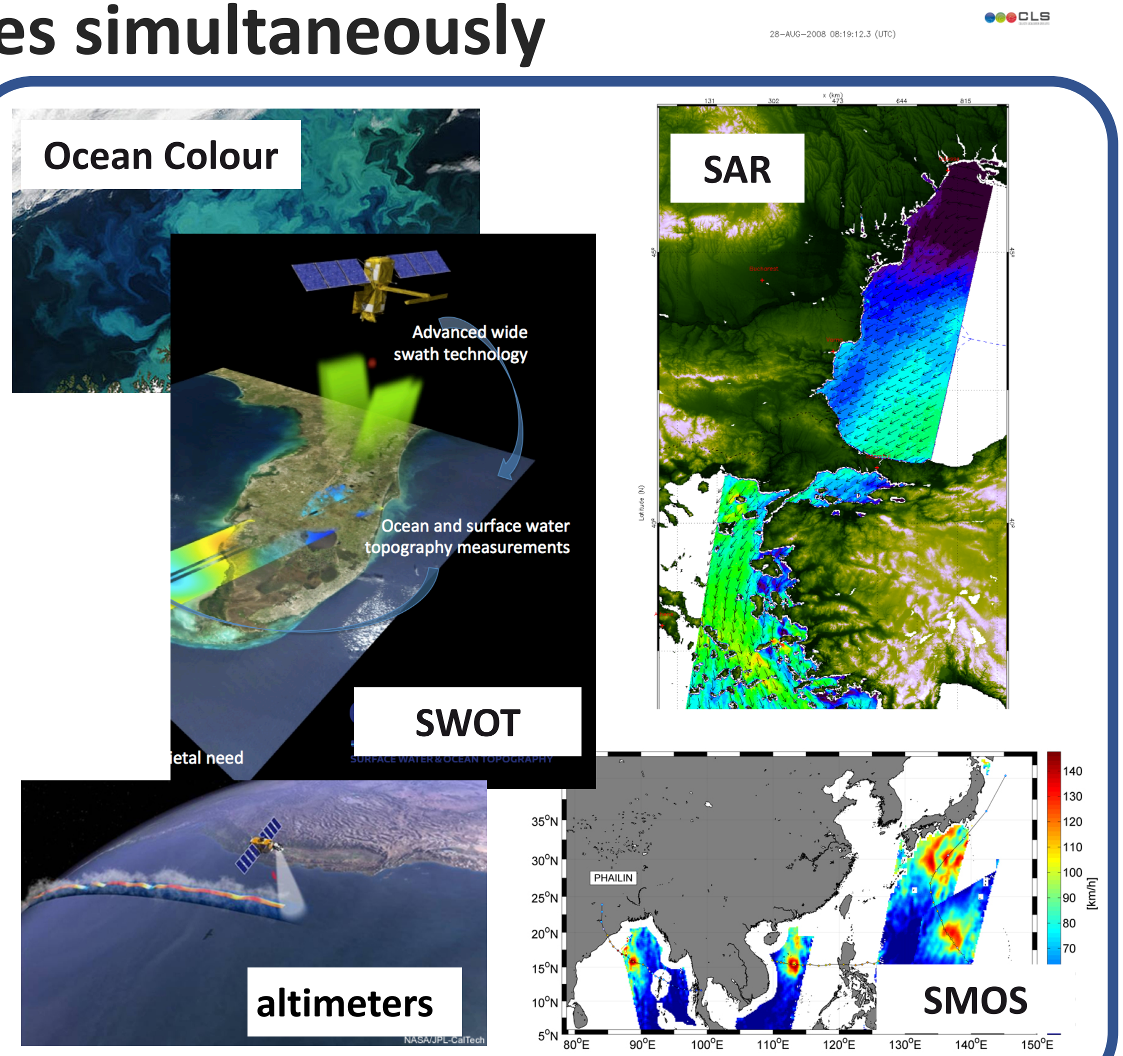
+

+



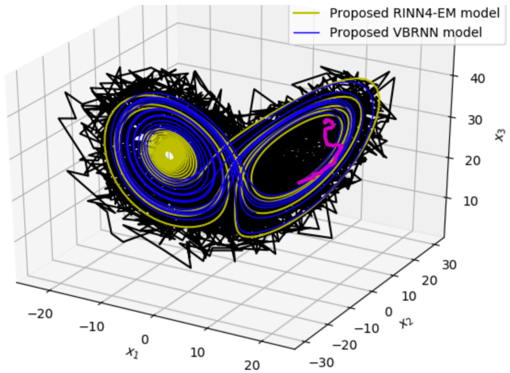
Context: No observation / simulation system to resolve all scales and processes simultaneously





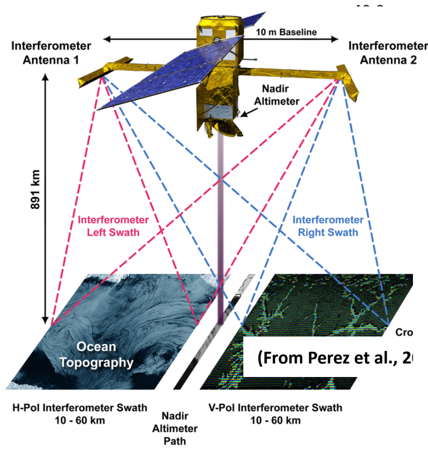
Learning for partially-observed systems / from irregularly-sampled data ?

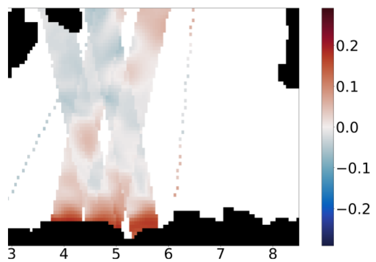
Scarce time sampling



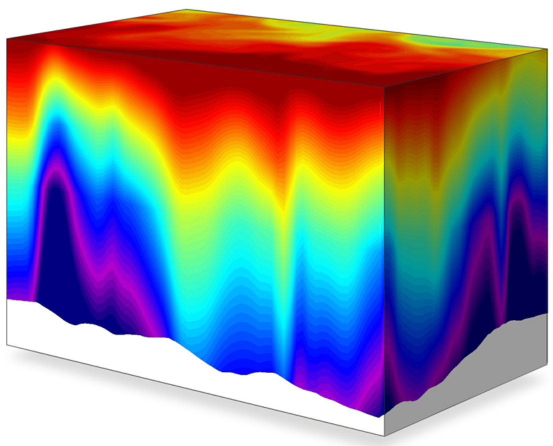
Ouala, ICASSP'19

Noisy and irregular sampling





Fablet, arXiv'2020
Nguyen, ICASSP'20

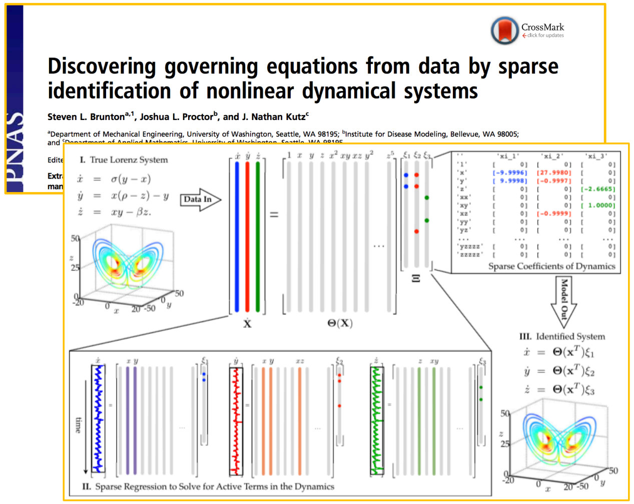


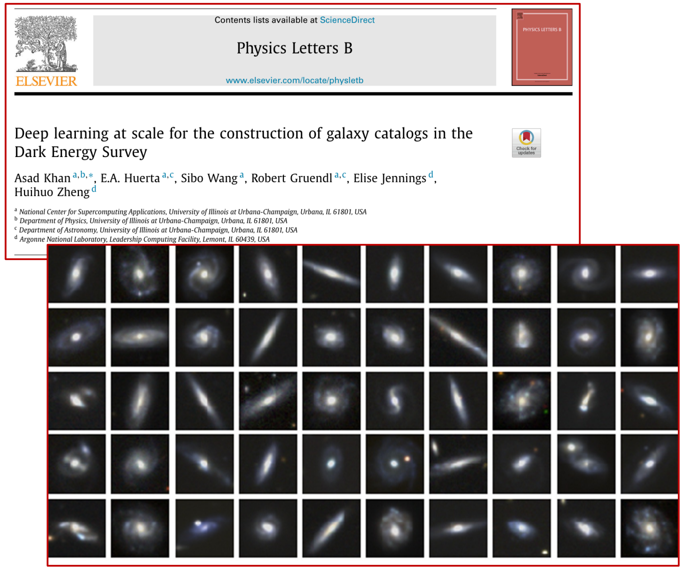
Partially-observed system

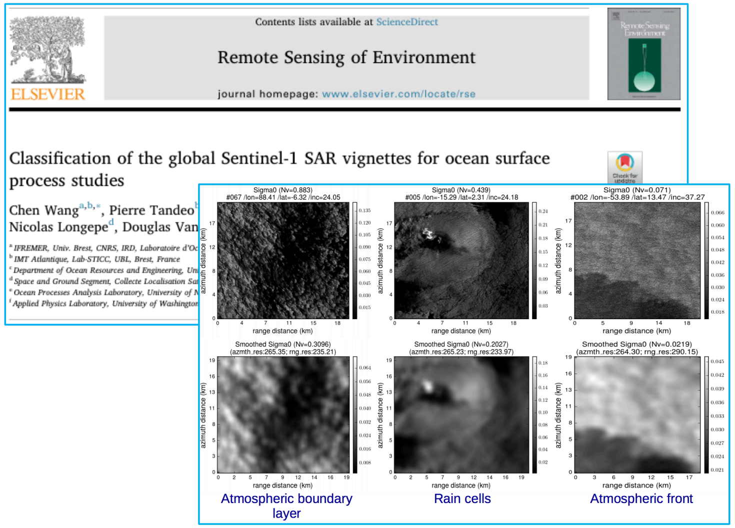
Ouala, preprint 2019

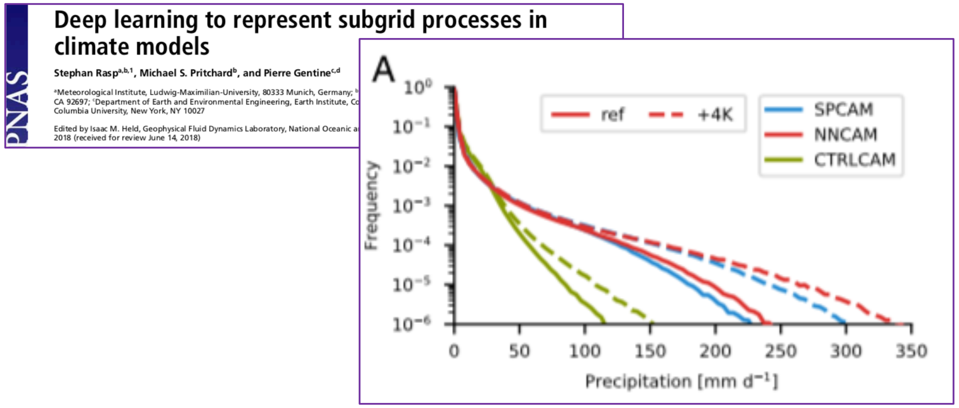
Briding Physics and Deep Learning

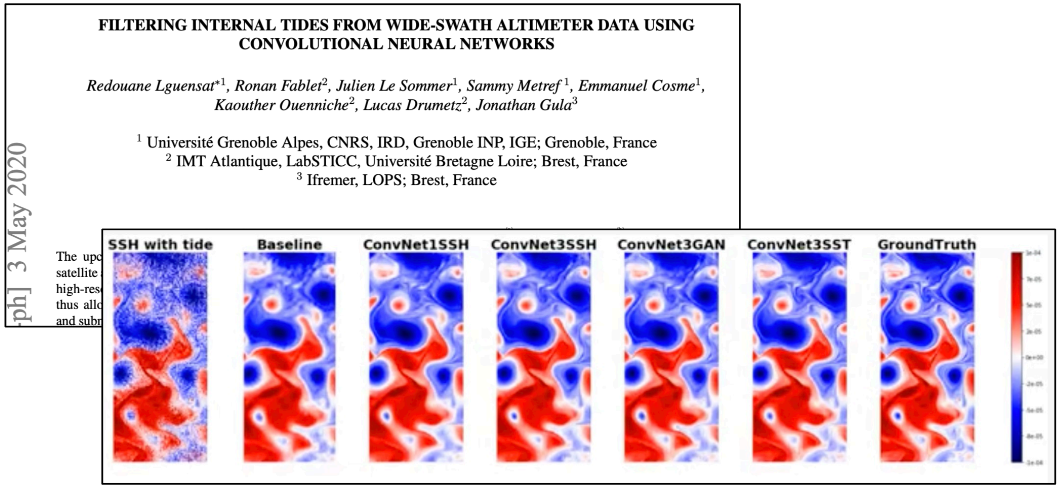
Examples of DL schemes applied to physics-related issues





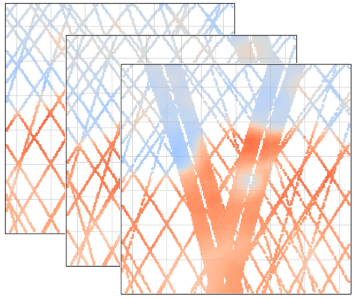




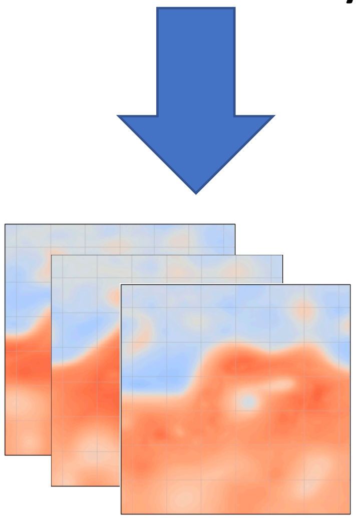


An example of deep learning for inverse problems

(Lguensat et al., 2020)

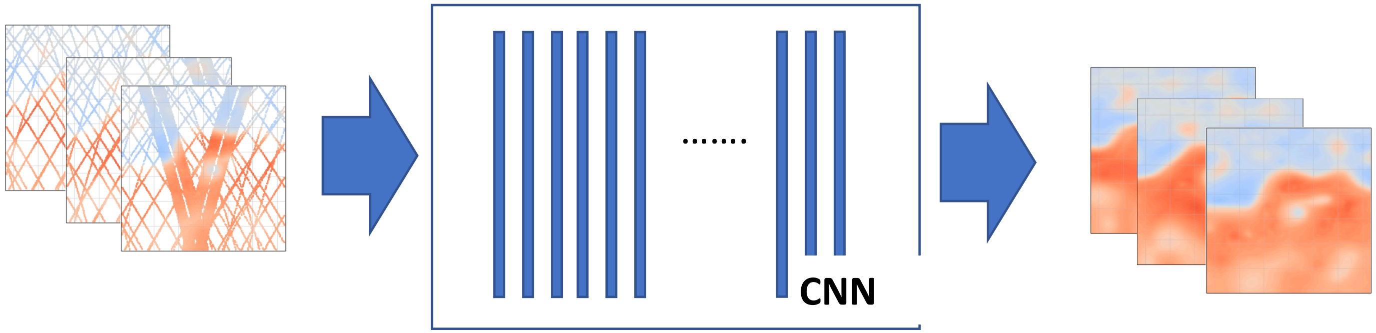


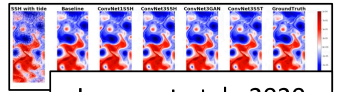
Partial observations y



True states x

Direct learning for inverse problems: $\widehat{x} = \Psi(y)$ $\quad y \longrightarrow$ CNN $\longrightarrow x$





Lguensat etal., 2020

Examples of CNN architectures: classic CNN architectures, architectures derived from inver problem algorithms, eg Reaction-Diffusion architectures, ADMM-inspired architectures,…

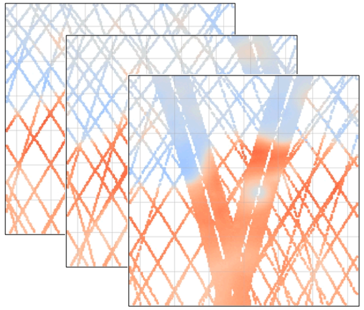
Good performance but possibly weak interpretability/generalization capacities of the solution byeond the training cases

May reinvent the wheel and forget to exploit the available (physical) knowledge

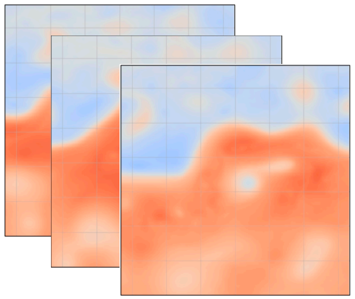
Bridging Deep Learning and Variational Data Assimilation

Lab-STICC

Starting from a (Weak constraint) 4DVar Data Assimilation (DA) formulation



Partial observations y



True states x

State-space formulation:

$$\left\{ \begin{array}{lll} \frac{\partial x(t)}{\partial t} & = & \mathcal{M}\left(x(t)\right) \\ y(t) & = & x(t) + \epsilon(t), \forall t \in \{t_0, t_0 + \Delta t ...., , t_0 + N\Delta t\} \end{array} \right.$$

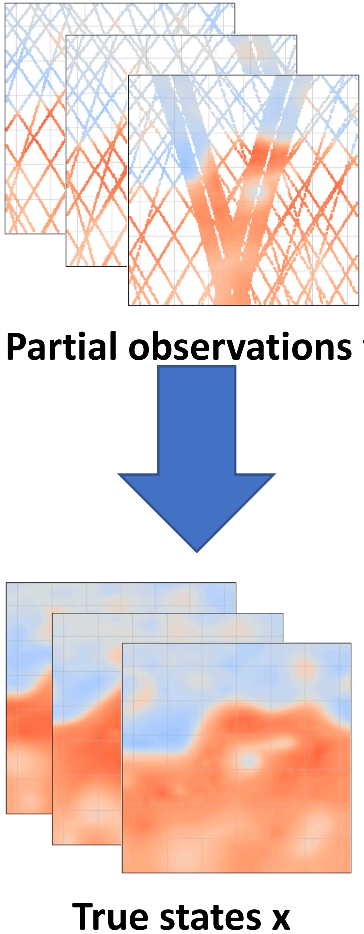
Associated variational formulation:

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

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

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

Bridging 4DVar DA and Deep Learning [Fablet et al., 2020]



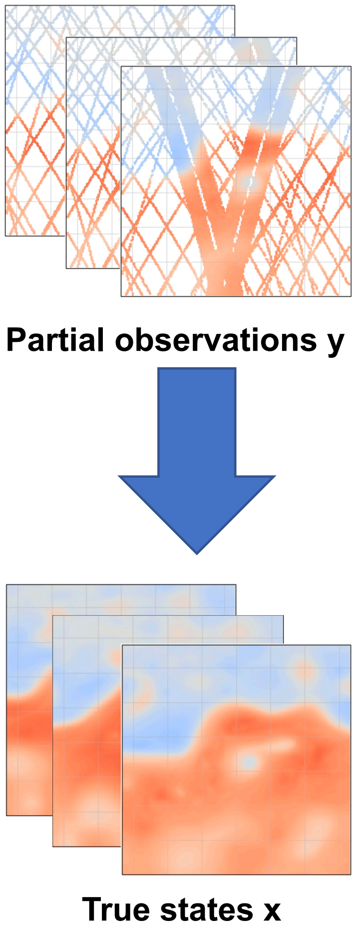
Model-driven schemes: $$\widehat{x} = \arg\min_x \underbrace{\lambda_1 \left\| x - y \right\|_\Omega^2 + \lambda_2 \left\| x - \Phi(x) \right\|^2}_{U_\Phi\left(x^{(k)}, y, \Omega\right)}$$

Gradient-based solver (adjoint/Euler-Lagrange method):

$$x^{(k+1)} = x^{(k)} - \alpha \nabla_x U_\Phi \left(x^{(k)}, y, \Omega \right)$$

Possibly implemented within a DL framework to benefit from automatic differentiatio tools to design gradient-based solver

Bridging 4DVar DA and Deep Learning (Fablet et al., 2020)



Model-driven schemes: $\widehat{x} = \arg\min_{x} \lambda_1 \|x - y\|_\Omega^2 + \lambda_2 \|x - \Phi(x)\|^2$

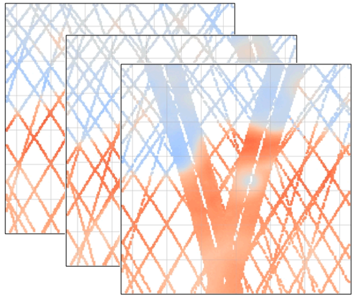
Direct learning for inverse problems: $\widehat{x} = \Psi(y)$ $\quad y \longrightarrow$ **CNN** $\longrightarrow x$

Proposed scheme: joint learning of the variational model and solver

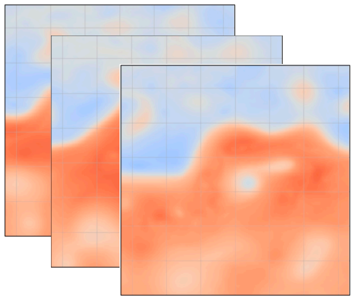
- **Theoretical bi-level optimization**

$$\arg\min_{\Phi} \sum_n \|x_n - \tilde{x}_n\|^2 \text{ s.t. } \tilde{x}_n = \arg\min_{x_n} U_\Phi(x_n, y_n, \Omega_n)$$

Bridging 4DVar DA and Deep Learning (Fablet et al., 2020)



Partial observations y



True states x

Model-driven schemes: $\widehat{x} = \arg\min_x \lambda_1 \|x - y\|^2_\Omega + \lambda_2 \|x - \Phi(x)\|^2$

Direct learning for inverse problems: $\widehat{x} = \Psi(y)$ $\quad y \rightarrow$ **CNN** $\rightarrow x$

Proposed scheme: **joint learning of the variational model and solver**

- **Theoretical bi-level optimization**

$$\arg\min_\Phi \sum_n \|x_n - \tilde{x}_n\|^2 \text{ s.t. } \tilde{x}_n = \arg\min_{x_n} U_\Phi(x_n, y_n, \Omega_n)$$

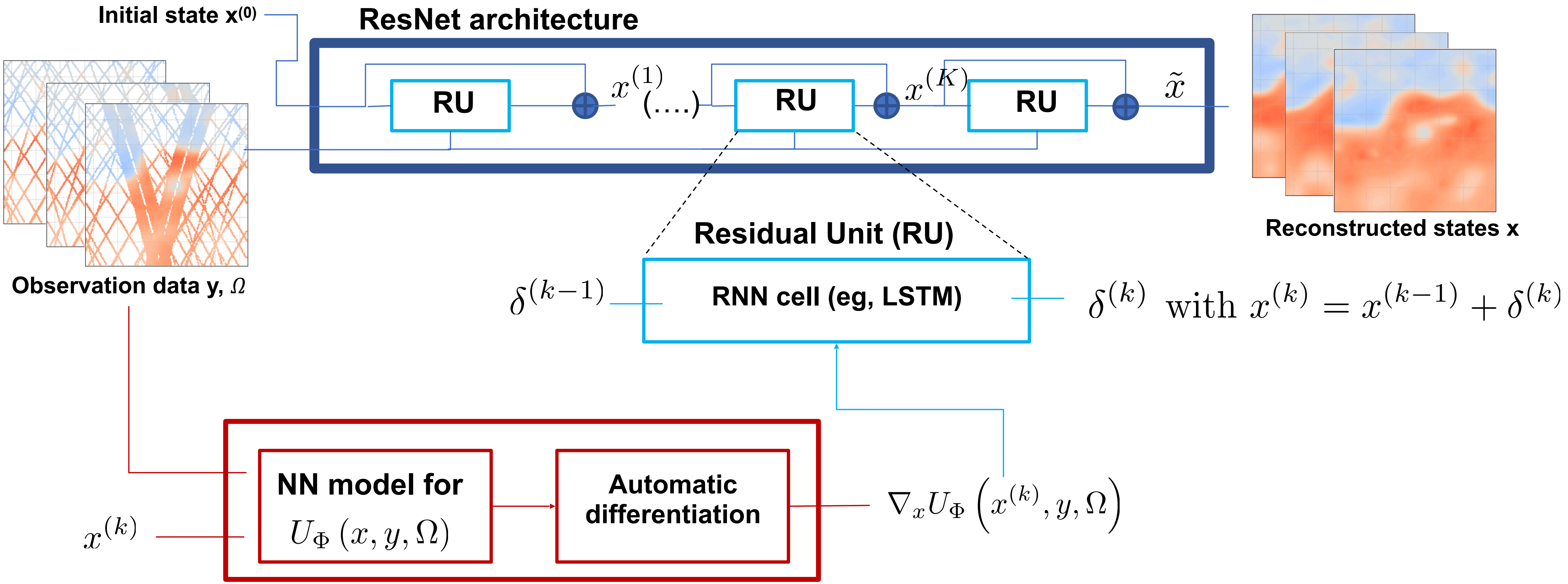
- **Restated with a gradient-based NN solver for inner minimization**

$$\arg\min_{\Phi,\Gamma} \sum_n \|x_n - \tilde{x}_n\|^2 \text{ s.t. } \tilde{x}_n = \Psi_{\Phi,\Gamma}(x_n^{(0)}, y_n, \Omega_n)$$

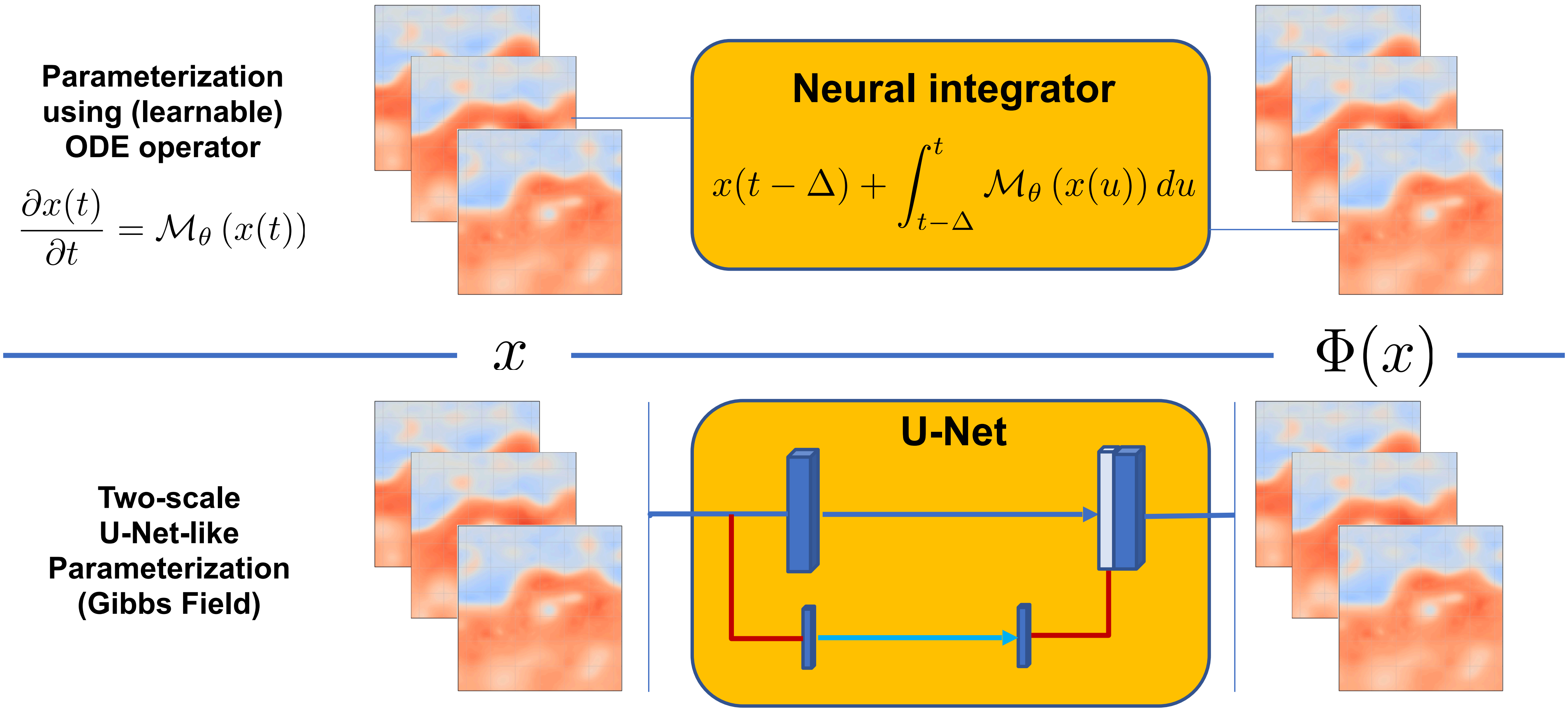
Iterative NN solver using automatic differentiation to compute gradient $\nabla_x U_\Phi\left(x^{(k)}, y, \Omega\right)$

Bridging 4DVar DA and Deep Learning (Fablet et al., 2020)

Proposed scheme: associated NN architecture

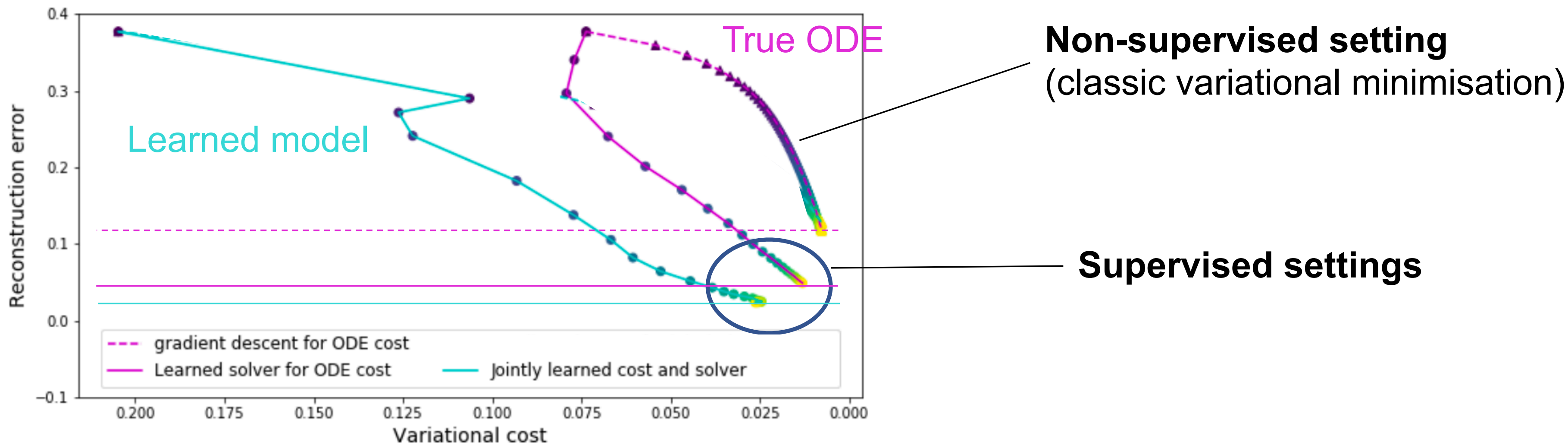


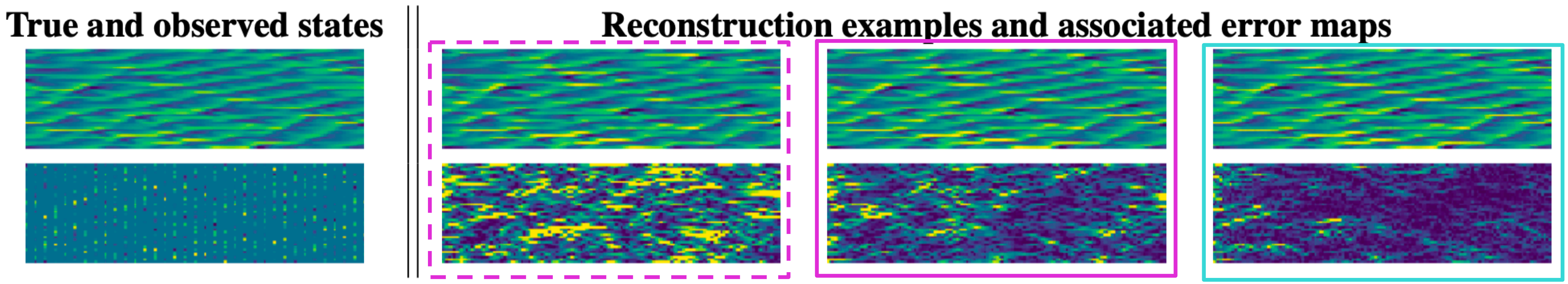
End-to-end learning for 4DVar DA: projection operator Φ



End-to-end learning for inverse problems (Fablet etal., 2020)

Illustration on Lorenz-96 dynamics (Bilinear ODE)

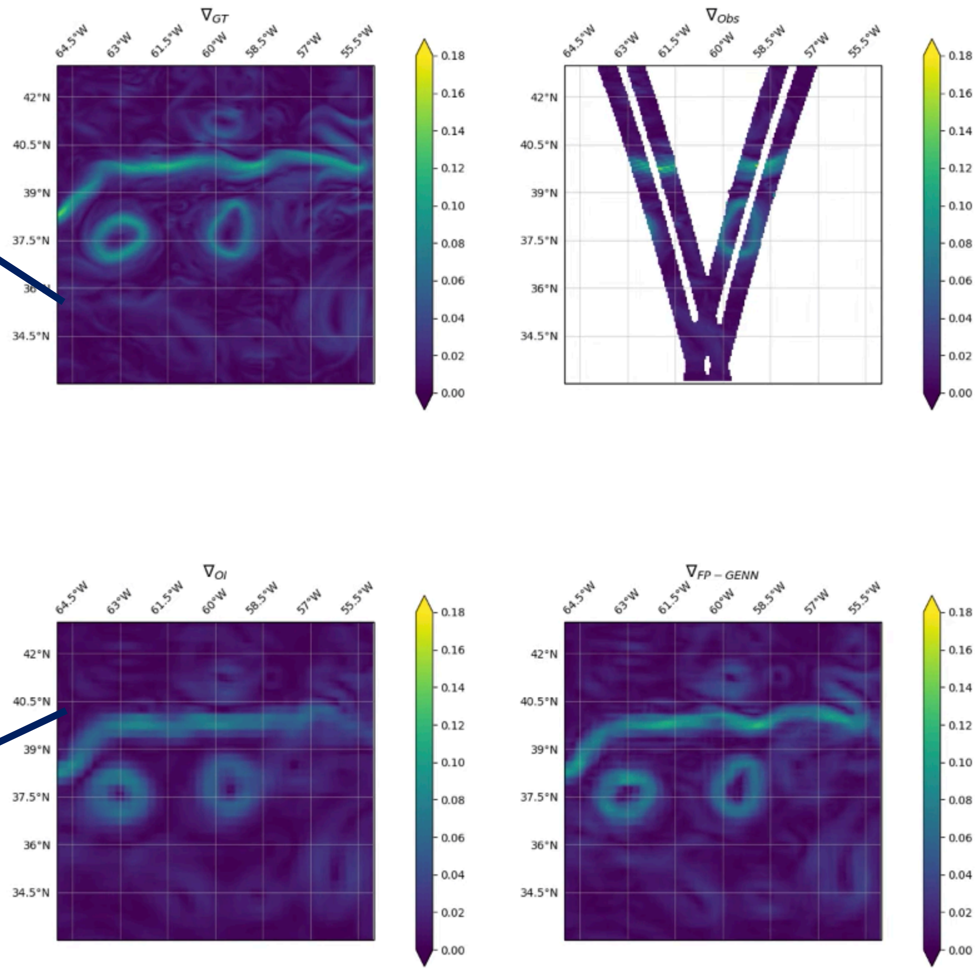


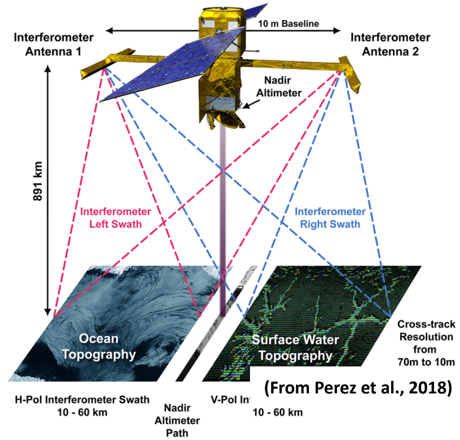


An example for upcoming SWOT mission

Groundtruth

State-of-the-art
operational processing





(From Perez et al., 2018)

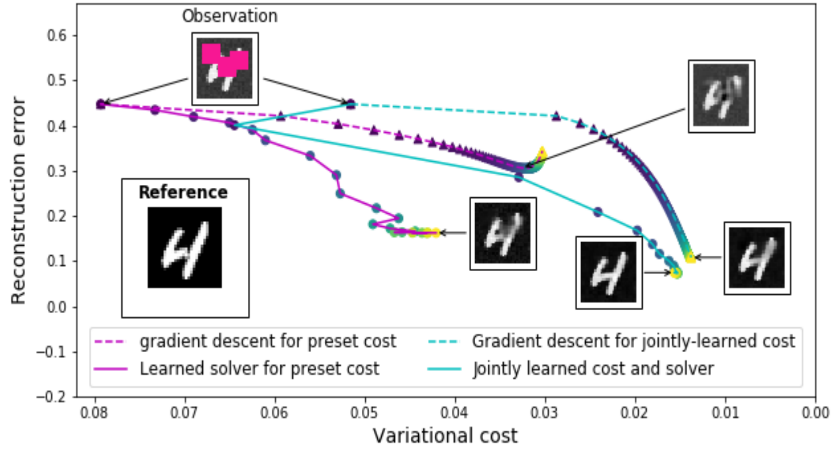
Proposed NN framework
(Fablet et al., 2019)

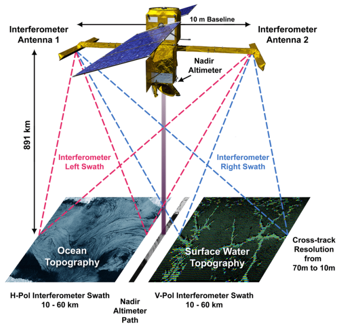
Beauchamp et al., 2020
https://doi.org/10.3390/rs12223806

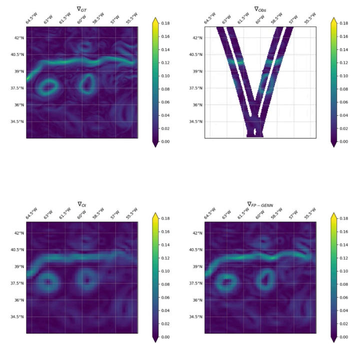
End-to-end learning for inverse problems (Fablet et al., 2020)

Key messages

- **We can bridge DNN and variational models to solve inverse problems**
- **Learning both variational priors and solvers using groundtruthed (simulation) or observation-only data**
- **The best model may not be the TRUE one for inverse problems**
- **Generic formulation/architecture beyond space-time dynamics**





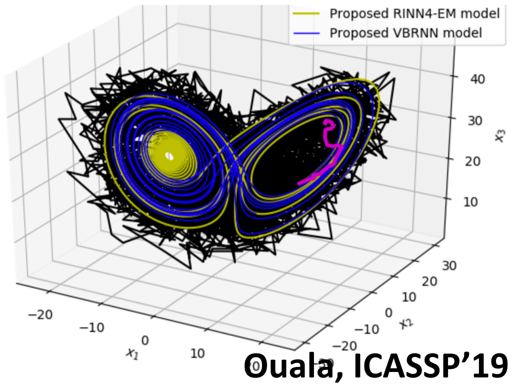


Preprint: https://arxiv.org/abs/2006.03653

Code: https://github.com/CIA-Oceanix

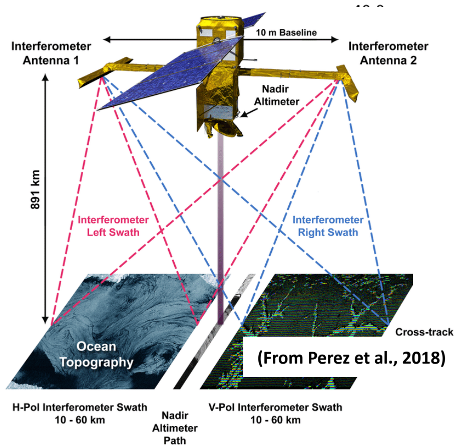
End-to-end learning from real observation data ?

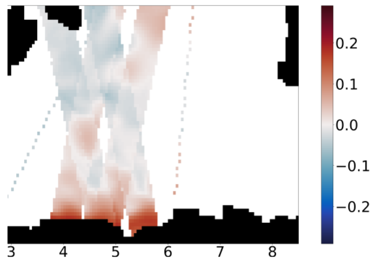
Scarce time sampling



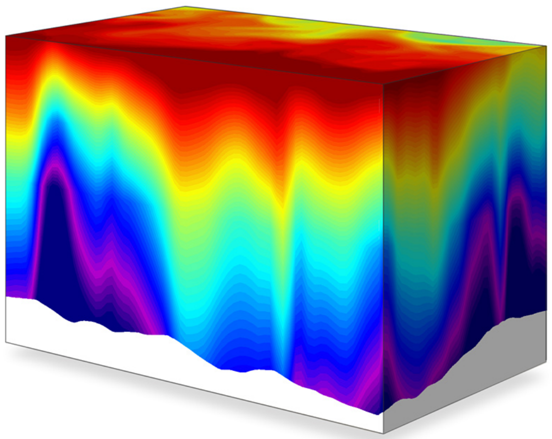
Ouala, ICASSP'19

Noisy and irregular sampling





Nguyen, ICASSP'20

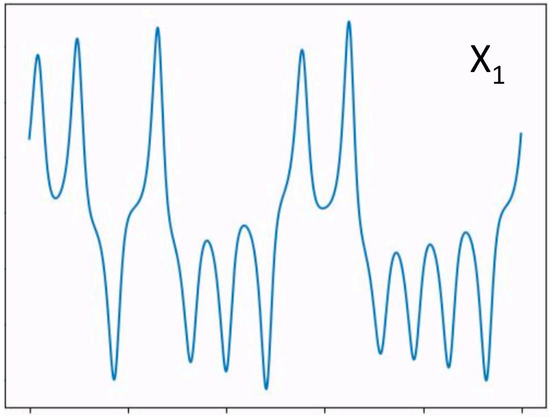


Partially-observed system

Ouala, preprint 2019

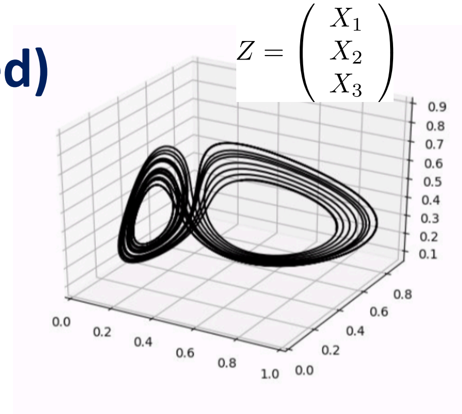
Neural ODE for partially-observed systems [Ouala et al., 2020]

Illustration for L63 assuming only the first components is observed

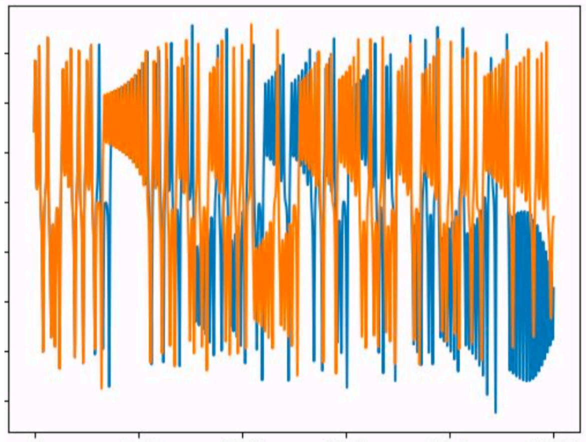


Learning Latent (unobserved) dynamics

$$d_t Z_t = \Phi_\theta(Z_t)$$



Objectives: acccurate short-term forecast and realistic « long-term » patterns for X_1



Approach: trainable variational formulation with latent dynamics

Neural ODE for partially-observed systems [Ouala et al., 2019]

Problem statement: **end-to-end learning of the latent (augmented) space and of the associated dynamics**

$$X_t = \begin{pmatrix} x_t \\ z_t \end{pmatrix}$$

Observed variables

Unknown variables

Dynamical model in the latent space

$$\partial_t X_t = f_\theta \left(X_t \right)$$

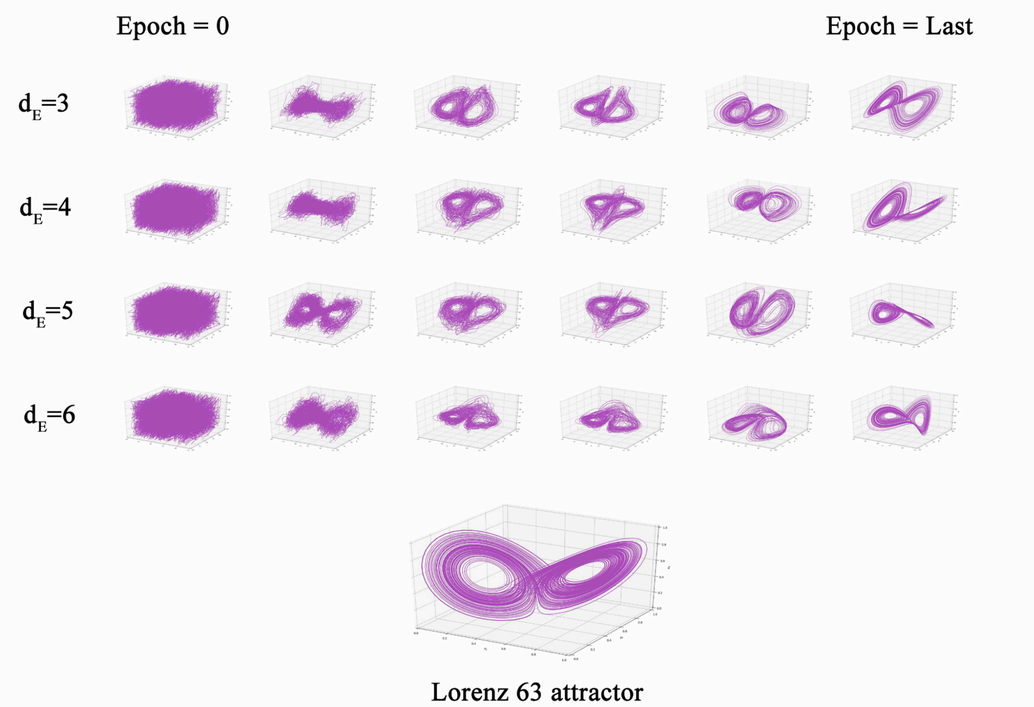
Goals:

1. **Learn model parameters θ from observed time series**
2. **Forecast future observed states given previous ones**

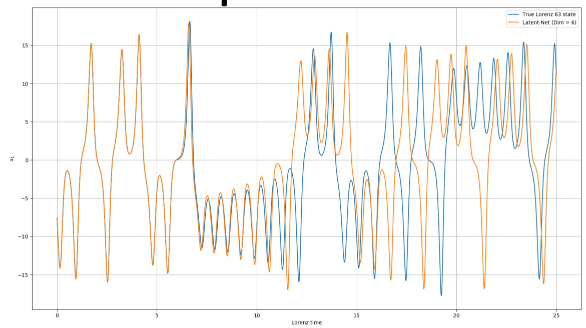
Proposed approach: WC 4DVar formulation with an unknown dynamical model

Neural ODE for partially-observed systems [Ouala et al., 2020]

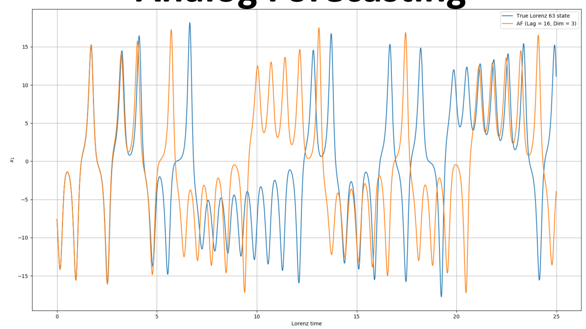
Illustration on Lorenz-63 dynamics



Proposed model



Analog Forecasting



Summary

- ***NNs as numerical schemes for ODE/PDE/energy-based representations of geophysical flows (***ie, not only Black Boxes***)***
- ***Embedding geophysical priors in NN representations*** (e.g., Lguensat et al., 2019; Ouala et al., 2019)
- ***End-to-end learning of (latent) representations (***eg, ODE***) and solvers*** (e.g., Fablet et al., 2020; Ouala et al., 2020)
- ***Towards stochastic representations embedded in NN architectures*** (e.g., Pannekoucke et al., 2020, Nguyen etal., 2020)

Making the most of model-driven and data-driven approaches

Model-driven

ODEs/PDEs
Variational models
State-space models
Data Assimilation
....

++
Use of prior knowledge
Interpretability
Generalization

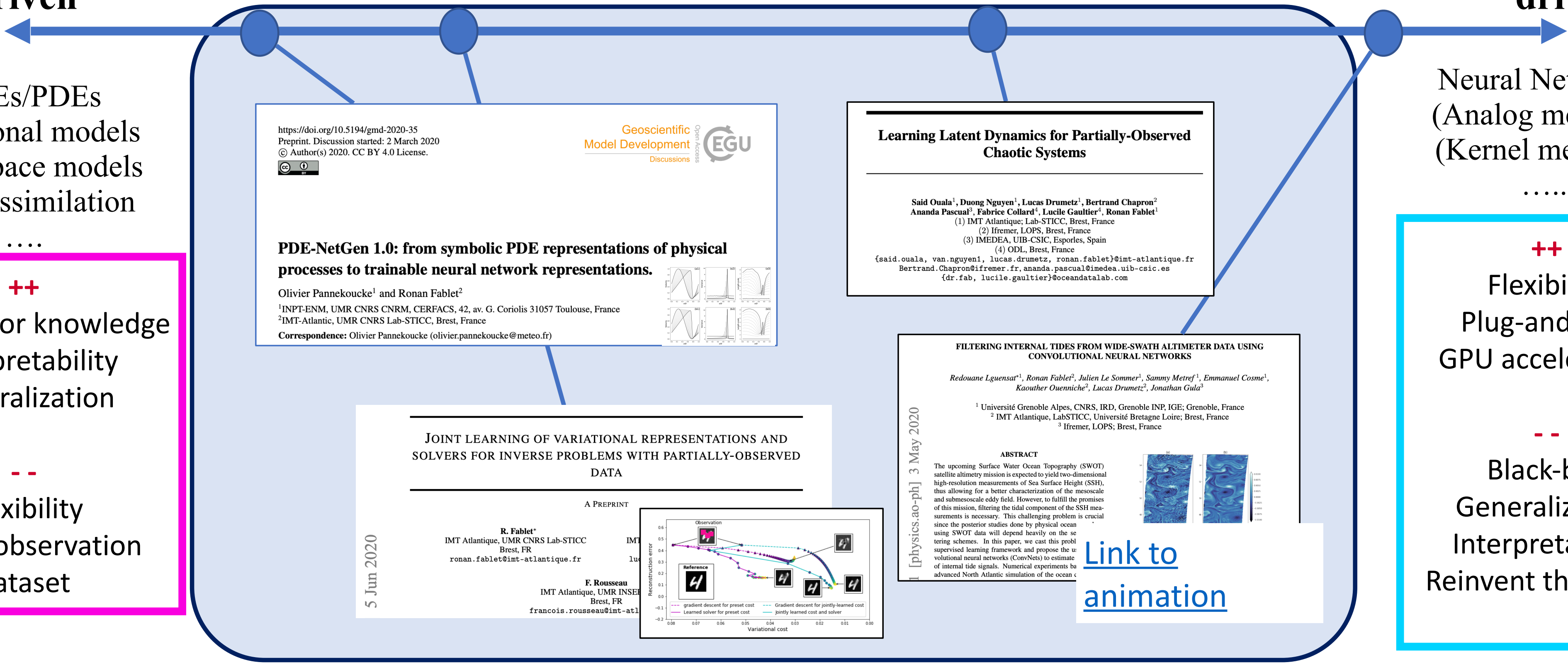
- -
Flexibility
Use of observation dataset

Data-driven

Neural Networks
(Analog methods)
(Kernel methods)
.....

++
Flexibility
Plug-and-play
GPU acceleration

- -
Black-box
Generalization
Interpretability
Reinvent the wheel



Physics-informed & Data-constrained

Data-driven & Physics-aware

AI Chair OceaniX 2020-2024

Physics-informed AI for Observation-Driven Ocean AnalytiX

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

Thank you.

Internship, PhD and postdoc opportunities