Deep Learning for surrogate modeling

Alessandro Longhi
&
D. Lathouwers, Z. Perkó

IMORN-31, 12th of September 2024

Department of Radiation Science and Technology
Delft University of Technology
Delft, The Netherlands

TUDelft

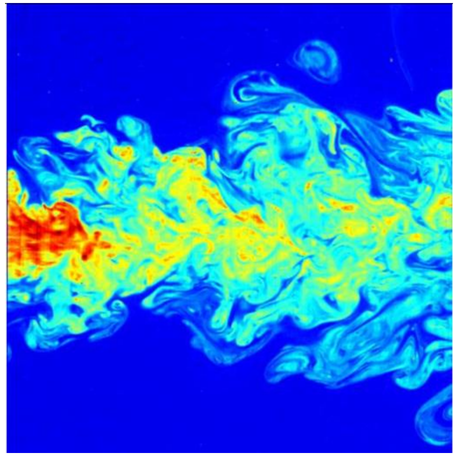
Programme

- What is *Deep Learning (DL) for Surrogate modeling*?
- Which methods are used? What am I focusing on?
- Examples of large scale applications.
- ASSAS project.
- The future of the field and challenges.

What is a surrogate model?

A subset of DL for Science is concerned with surrogate modeling

- Context: physical simulations
  - Fluid dynamics
  - Nuclear physics
  - Climate science
  - ...



Described by set of Partial Differential Equations (PDEs)/Ordinary Differential Equations (ODEs)

Solvers are **expensive**!
(in time)

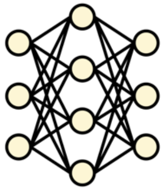
Build a **simpler** (faster to solve)
but still **accurate** model

Surrogate model

What is Deep Learning?

- The main objective of Deep Learning is the **approximation of functions** (which we do not know)

$$\mathcal{X} \xrightarrow{f_{\boldsymbol{\theta}}(\mathcal{X})} \mathcal{Y}$$



$\longrightarrow$ 'Cat'

In DL one overuses the term 'mapping'

f_θ is usually **overparametrized** (by vector of weights θ)

$$y = f_\theta(x) = \boxed{W}x + \boxed{\mathrm{b}}$$

Weights to be found! (During training)

In practice, **deep** and **nonlinear**

$$y = f_\theta(x) = \underbrace{g_N \circ \cdots \circ g_2 \circ g_1(x)}_{\textbf{Deep}}$$

'Neural Networks'

TU Delft

Why Deep Learning?

- Universal approximation theorem (?)
  - Extremely flexible
  - Scalability (to data)
  - Better generalization

- Approximation of f_{θ} is done in a **data-driven** manner (data from the simulations)

θ found during training by looking at ground truth

Fast computation thanks to GPUs (easily parallelizable operations)

Standard numerical solvers will never be beaten in accuracy (?)

What are the data used?

There are 3 possibilities

- Purely *Supervised* learning

  Data from simulations

- Purely *Unsupervised* learning

  No data, uses equations from PDEs/ODEs

- A mixture of the two

  Data + equations + known properties of the system

TU Delft

Problem statement

- **Time dependent**, **parametric** and **non linear** PDEs

$$
\begin{cases}
\hat{\mathcal{N}}(s(\mathbf{x}, t|\boldsymbol{\mu}), \mathbf{x}, t, \boldsymbol{\mu}) = g(\mathbf{x}, t, \boldsymbol{\mu}) \\
s(\tilde{\mathbf{x}}, t|\boldsymbol{\mu}) = v(\tilde{\mathbf{x}}, t, \boldsymbol{\mu}) \\
s(\mathbf{x}, t = 0|\boldsymbol{\mu}) = s^0(\mathbf{x}, \boldsymbol{\mu})
\end{cases}
$$

- $\hat{\mathcal{N}}$ is a nonlinear (integro-) differential operator
- v is the boundary condition
- s is the solution
- s^0 is the initial condition
- $\tilde{\mathbf{x}}$ is the variable on the boundary
- $\boldsymbol{\mu}$ is the vector of parameters

Learn the mapping

$$
(s^0(\mathbf{x}, \boldsymbol{\mu}), v(\tilde{\mathbf{x}}, t, \boldsymbol{\mu}), \boldsymbol{\mu}) \rightarrow s(\mathbf{x}, t|\boldsymbol{\mu})
$$

NB: is this a function?

Neural **Operators**

Reduced Order Modeling (ROM)

- Main assumption: → *A system determined by **N degrees of freedom** can be projected into **a lower dimensional space** of dimension n, with* $n \ll N$

- In most cases, Reduced Basis methods →

$$s(\mathbf{x}, t|\boldsymbol{\mu}) \approx \sum_{k=1}^{n} a_k(t, \boldsymbol{\mu})\, V_k(\mathbf{x})$$

- Gaussian Processes, Radial Basis functions, PCA

- *Intrusive methodology* vs *non-intrusive methodology*

- Analogy in DL with **manifold hypothesis:** → *High-dimensional data actually lie in low dimensional manifolds*

PDEs' solution, image, medical record → Correlations, symmetries, noise...

Reduced/Latent space

- In **Proper Orthogonal Decomposition** (POD) the vectors $V_k(\boldsymbol{x})$ are found using **Singular Value Decomposition** (SVD) on the *snapshot matrix*

$$M = \left[s(\mathbf{x}, t_1|\boldsymbol{\mu}_1) | \cdots | s(\mathbf{x}, t_{N_t}|\boldsymbol{\mu}_1) | \cdots | s(\mathbf{x}, t_1|\boldsymbol{\mu}_{N_{\boldsymbol{\mu}}}) \cdots s(\mathbf{x}, t_{N_t}|\boldsymbol{\mu}_{N_{\boldsymbol{\mu}}}) \right]$$

- SVD gives the optimal basis (for a linear projection)
- Equivalent to finding a new vector basis with eigenvectors pointing in direction of maximum variance

- The above mapping is **linear**! DL gives an handy way to compute **nonlinear dimensionality reduction**

During the training an *optimal latent space* is found

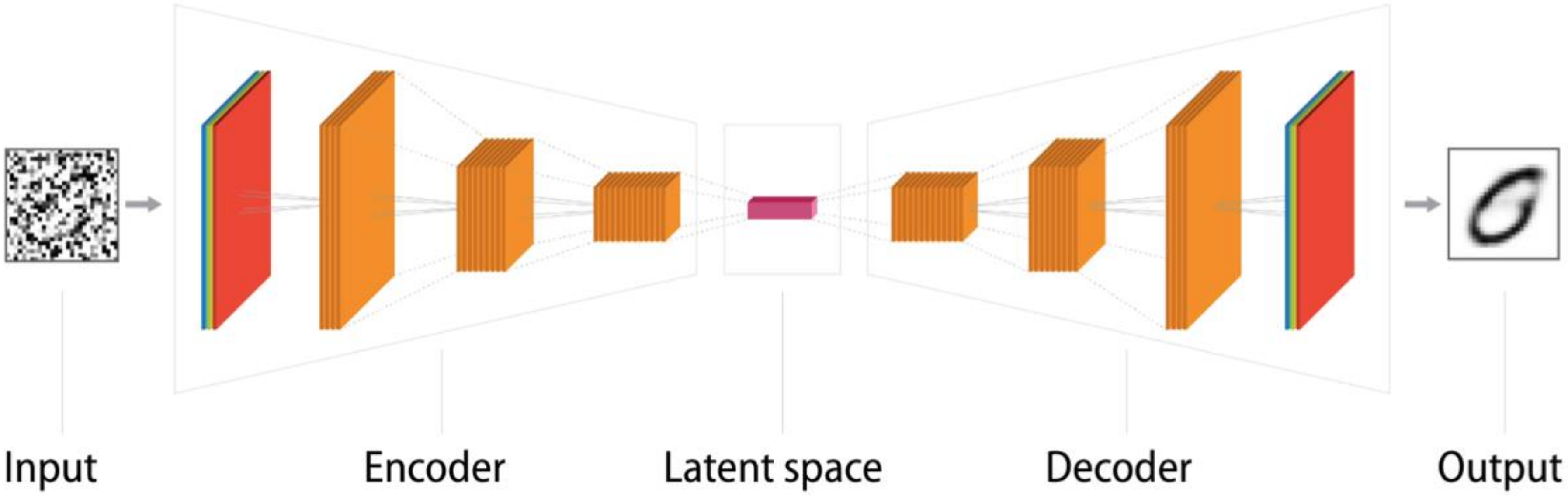
- Are there any desiderata for this very abstract vector?

Independent
Interpretable

Disentangled

Very open and not well-defined area of research

AutoEncoder



Some of the available methods

Supervised

- Reduced Order Modeling enhanced by DL
  - POD for $V_k(\boldsymbol{x})$ followed by DL to predict α_k
- Sparse Identification of Nonlinear Dynamics
  - AutoEncoder + latent dynamics modeled by pre-determined library of functions
- Neural Operators
  - How to approximate an Operator

Unsupervised

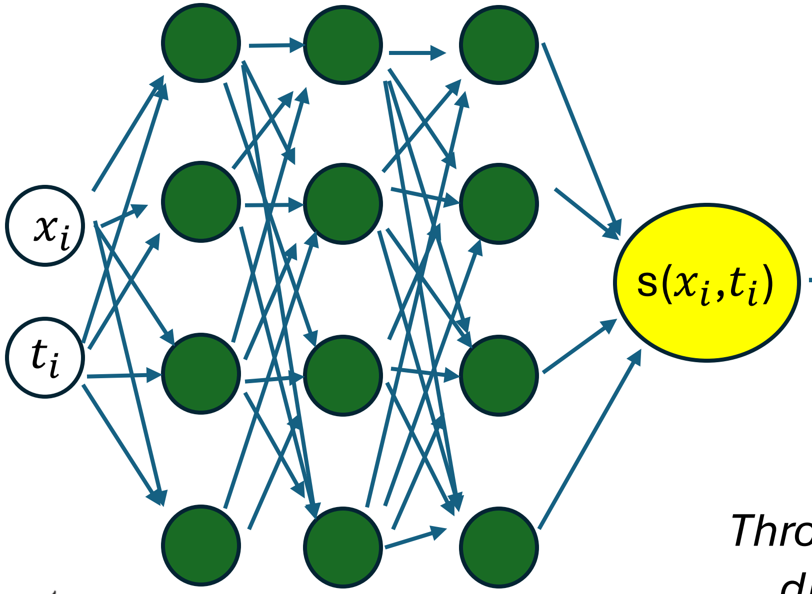
- Physics Informed NNs (PINNs) (although **not limited** to unsupervised)

The known PDEs are used in the loss function to drive the learning process

Approximating the **Solution** vs approximating the **Operator**

Approximating the **Solution** through *Physics-Informed NNs* (not the only way)

Define a set of *collocation points* $\{x_i, t_i\}$



$$\left\{ \begin{array}{l} \left.\dfrac{\partial s}{\partial x}\right|_{x=x_i} \\ \left.\dfrac{\partial s}{\partial t}\right|_{t=t_i} \end{array} \right.$$

Through automatic differentiation

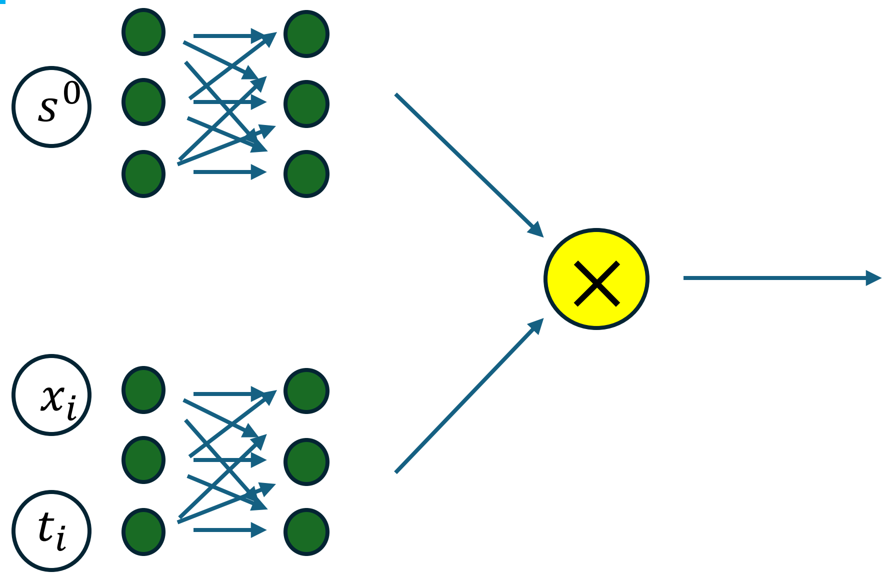
$$\mathcal{L} = \left\| \left.\frac{\partial s}{\partial t}\right|_{t=t_i} + \left.\frac{\partial s}{\partial x}\right|_{x=x_i} \right\|_2$$

Plus **initial conditions** and **boundary conditions**

Here, the NN is approximating directly the solution s(x,t) !

Approximating the **Solution** vs approximating the **Operator**

Approximating the **Operator** through Physics-Informed DeepONet



$$s(x_i, t_i | s^0) \begin{cases} \left.\dfrac{\partial s}{\partial x}\right|_{x=x_i} \\ \left.\dfrac{\partial s}{\partial t}\right|_{t=t_i} \end{cases}$$

$$\mathcal{L} = \left\| \left.\frac{\partial s}{\partial t}\right|_{t=t_i} + \left.\frac{\partial s}{\partial x}\right|_{x=x_i} \right\|_2$$

Plus **initial conditions** and **boundary conditions**

Here, the NN is approximating directly the Solution Operator !

Neural Operators

- We want to model the mapping $v : D \to \mathbb{R}^{d_v} \longrightarrow z : D \to \mathbb{R}^{d_z}$
- For simplicity, $d_v = d_z = 1$
- What if we apply a simple NN? $\longrightarrow \mathbf{z} \approx \sigma(\mathbf{B}\, v + b)$ with $\mathbf{B} \in \mathbb{R}^{J\times J},\ b \in \mathbb{R}^{J}$

Then it is clear that the size of the NN depends on J

With J being the size of the spatial discretization

Operator layer, generalization of the usual layer for mappings between **finite dimensional spaces** to mappings between **infinite dimensional spaces**

$$(\mathcal{O}v)(x) = z(x) = \sigma\left(\int_D k(x,y)\, v(y)dy + g(x) + \mathbf{W}v(x)\right)$$

$k(x,y) : D\times D \to \mathbb{R}^{d_z\times d_v}$ $\quad g : D \to \mathbb{R}^{d_z}$ $\quad$ K, g and **W** parametrized by NNs!

Autoregressive or global?

How is the full spatio-temporal domain of the solution predicted?

Global approach

$$(s^0(\mathbf{x}, \boldsymbol{\mu}), \boldsymbol{\mu}, T) \rightarrow s(\mathbf{x}, t = T|\boldsymbol{\mu})$$

- Time is used as an input parameter.
- High number of Neural Networks (NN) weights needed.
- Difficult to generalize outside the largest T used in training.

Autoregressive approach

$$(s(\mathbf{x}, t|\boldsymbol{\mu}), \boldsymbol{\mu}, \Delta t) \rightarrow s(\mathbf{x}, t + \Delta t|\boldsymbol{\mu})$$

- Δt is used to march in time **as in numerical solvers.**
- Previous state contains more information than IC.
- The full temporal solution is obtained autoregressively starting with the initial condition s^0.
- There is accumulation of errors.

TU Delft

Examples of large scale applications

AURORA: A FOUNDATION MODEL OF THE ATMOSPHERE

Cristian Bodnar[*,1], Wessel P. Bruinsma[*,1], Ana Lucic[*,1], Megan Stanley[*,1],
Johannes Brandstetter[3,†], Patrick Garvan[1], Maik Riechert[1], Jonathan Weyn[2], Haiyu Dong[2],
Anna Vaughan[4], Jayesh K. Gupta[5,†], Kit Tambiratnam[2], Alex Archibald[4], Elizabeth Heider[1],
Max Welling[6,†], Richard E. Turner[1,4], and Paris Perdikaris[1]

- Foundation model
- 1.3 billion NN's weights

POSEIDON: Efficient Foundation Models for PDEs

Maximilian Herde[1,*], Bogdan Raonić[1,2,*], Tobias Rohner[1], Roger Käppeli[1],
Roberto Molinaro[1], Emmanuel de Bézenac[1], and Siddhartha Mishra[1,2]

[1]Seminar for Applied Mathematics, ETH Zurich, Switzerland
[2]ETH AI Center, Zurich, Switzerland
Correspondence to herdem@ethz.ch

- Foundation model
- Training on small set of PDEs and see how it generalize to others
- Can generalize to unseen physical processes
- From 21 to 629 million NN's weights
- 1 to 5 orders of magnitude gain in speed

ASSAS project

CODE developed by IRSN, **severe accidents simulator,** various types of Generation II reactor

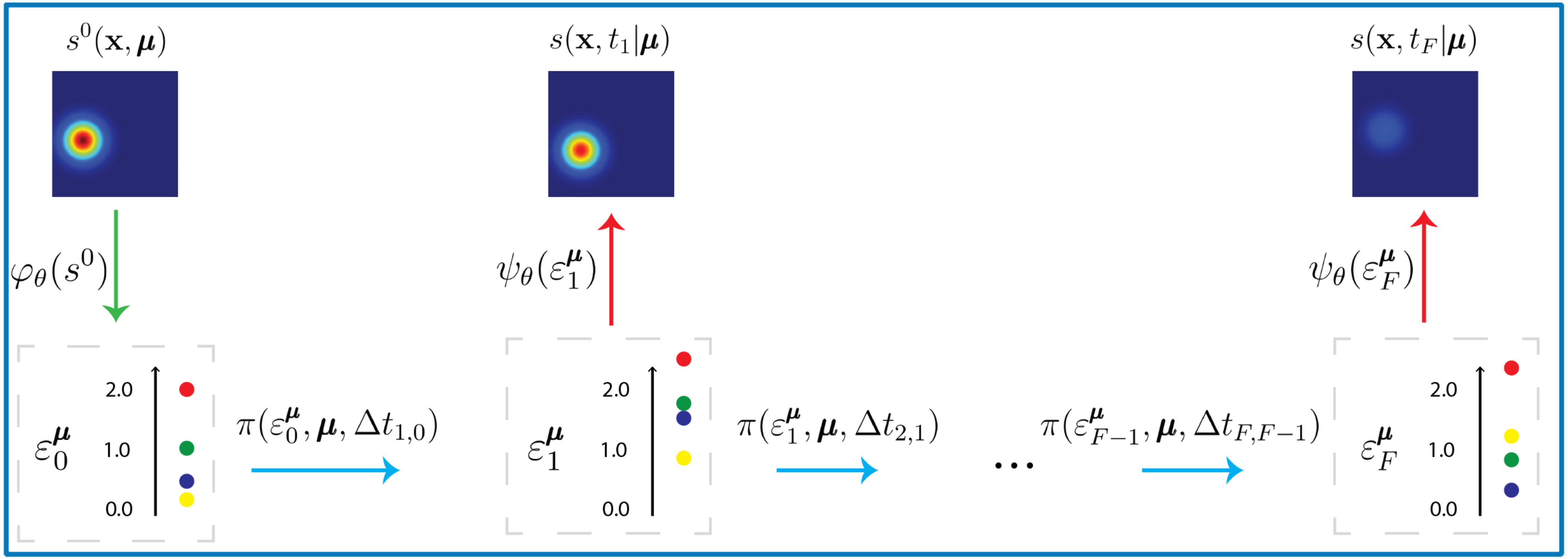
11 modules which manage different physics (some of them are coupled)

The objective is to create a ‘simulator’ using surrogate models.

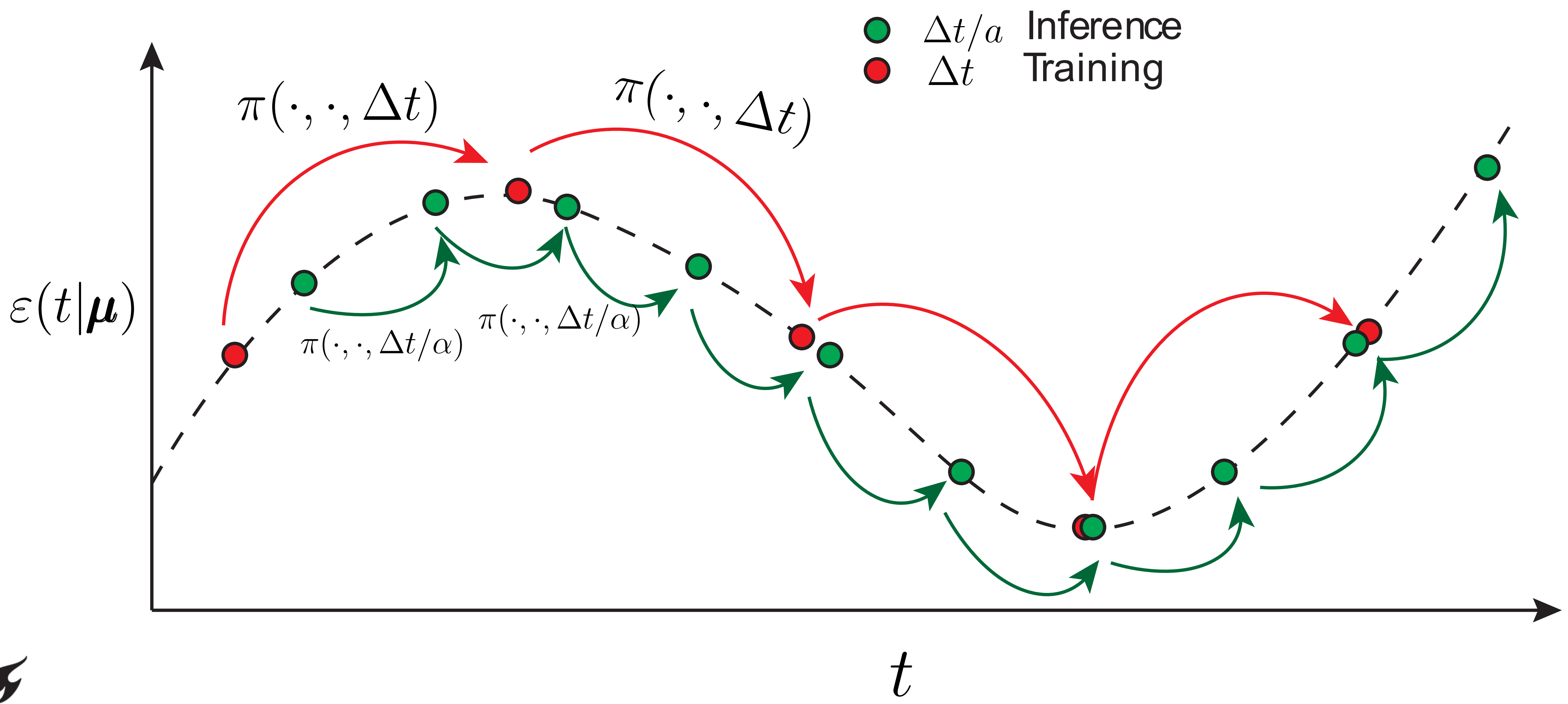
The difficulty here is dealing with the *operator actions* that may cause *discontinuities* in the solution.

TU Delft

AutoEncoding and Neural ODEs



Time generalization



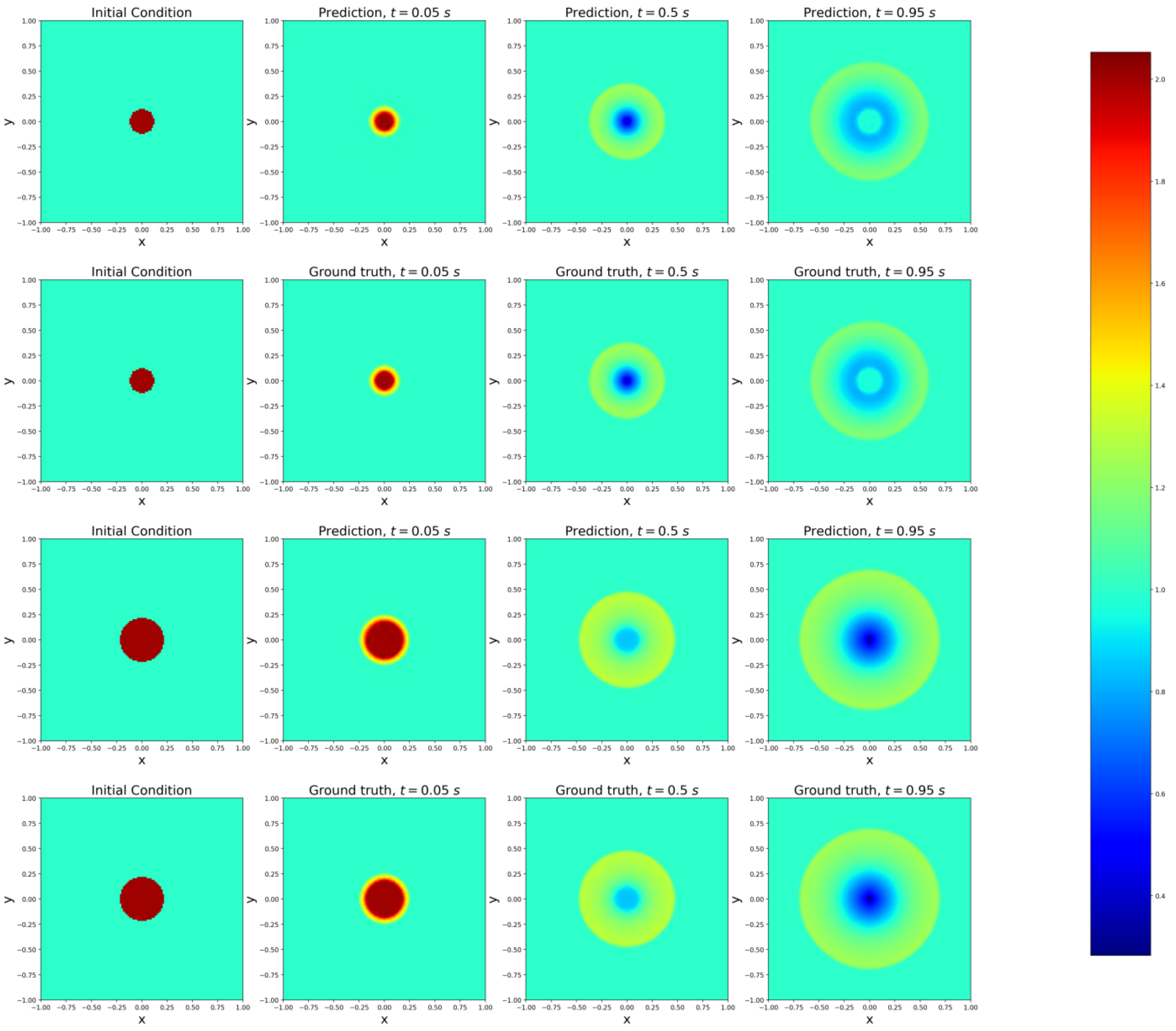
Some examples

2D Shallow-Water Equations

$$\begin{cases} \partial_t s + \partial_x su + \partial sv = 0 \\ \partial_t su + \partial_x \left(u^2 s + \frac{1}{2} g_r s^2 \right) + \partial_y uvs = -g_r s \partial_x b \\ \partial_t sv + \partial_y \left(v^2 s + \frac{1}{2} g_r s^2 \right) + \partial_x uvs = -g_r s \partial_y b \end{cases}$$

	U-NET	FNO	PINN
RMSE	8.6×10^{-2}	4.5×10^{-3}	1.7×10^{-2}
nRMSE	8.3×10^{-2}	4.4×10^{-3}	1.7×10^{-2}
max error	4.4×10^{-1}	4.5×10^{-2}	1.3×10^{-3}
cRMSE	1.3×10^{-2}	2.0×10^{-4}	1.7×10^{-2}
bRMSE	4.2×10^{-3}	1.4×10^{-3}	1.5×10^{-1}
fRMSE low	2.0×10^{-2}	2.6×10^{-4}	5.9×10^{-3}
fRMSE mid	7.0×10^{-3}	3.1×10^{-4}	1.9×10^{-3}
fRMSE high	8.6×10^{-4}	2.5×10^{-4}	6.0×10^{-4}

Our nRMSE: 2.8×10^{-3}



Some examples

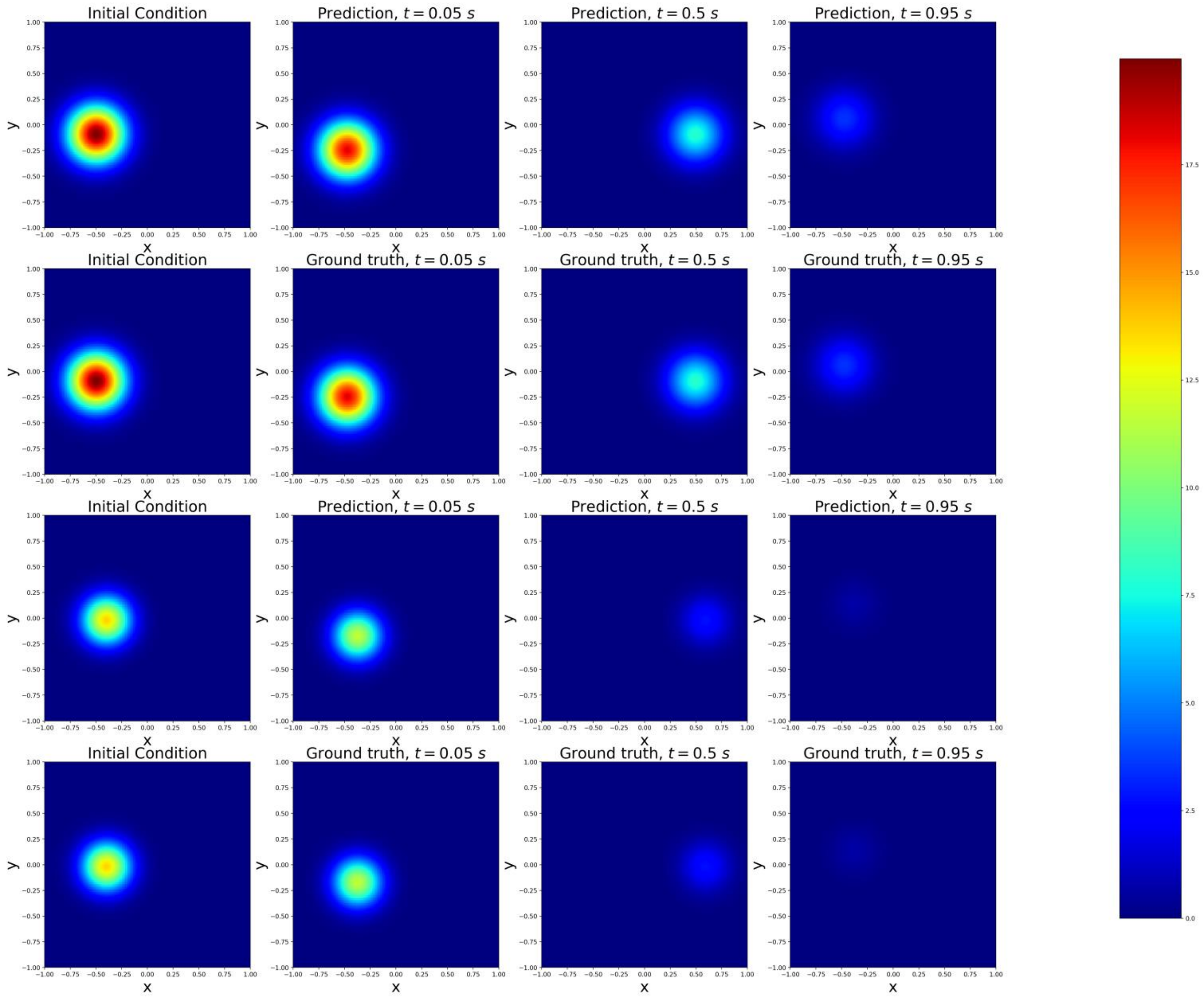
2D Molenkamp Test

$$\begin{cases} \partial_t s(x,y,t|\boldsymbol{\mu}) + u\,\partial_x s(x,y,t|\boldsymbol{\mu}) + v\,\partial_y s(x,y,t|\boldsymbol{\mu}) + \mu_3\, s(x,y,t|\boldsymbol{\mu}) = 0 \\ s(x,y,0|\boldsymbol{\mu}) = \mu_1\, 0.01^{\mu_2\, h(x,y,0|\boldsymbol{\mu})^2}, \quad h(x,y,0|\boldsymbol{\mu}) = \sqrt{(x-\mu_4+\frac{1}{2})^2 + (y-\mu_5)^2} \end{cases}$$

5 parameters

TU Delft

Some examples



Conclusions and future challenges

- Deep Learning in practice is useful in surrogate modeling because of **speed**, standard numerical solvers will always be needed for **accuracy** ad **data**

- Can we gain more insights about the learned **reduced representations,** and what properties should they respect?

- There are a lot of methods available and it is often difficult to choose one or to understand why one works better than another: can we find a **unified framework** for DL methods for PDEs?

- **Data scarcity**. Sometimes because of memory or time resources we cannot use a lot of data, especially for large cale problems: how can we **determine which data-points are most needed** for an optimal training?

TU Delft

Some useful references

Slide 10/11:

- Alfio Quarteroni and Gianluigi Rozza. Reduced Order Methods for Modeling and Computational Reduction. Springer International Publishing, 01 2014. ISBN 978-3-319-02089-1. doi: 10.1007/978-3-319-02090-7
- Irina Higgins, David Amos, David Pfau, Sebastien Racaniere, Loıc Matthey, Danilo Jimenez Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. ArXiv, abs/1812.02230, 2018b. URL https://api.semanticscholar.org/CorpusID:54447715

Slide 13:

- Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proceedings of the National Academy of Sciences, 113(15):3932–3937, March 2016. ISSN 1091-6490. doi: 10.1073/pnas.1517384113. URL http://dx.doi.org/10.1073/pnas.1517384113.
- Kaushik Bhattacharya, Bamdad Hosseini, Nikola B Kovachki, and Andrew M Stuart. Model reduction and neural networks for parametric pdes. The SMAI journal of computational mathematics,
- Stefania Fresca and Andrea Manzoni. Pod-dl-rom: Enhancing deep learning-based reduced order models for nonlinear parametrized pdes by proper orthogonal decomposition. Computer Methods in Applied Mechanics and Engineering

TU Delft

Some useful references

Slide 14:

- M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics, 378:686–707, February 2019. ISSN 0021-9991. doi: 10.1016/j.jcp.2018.10.045. URL http://dx.doi.org/10.1016/j.jcp.2018.10.045.

Slide 15:

Sifan Wang, Hanwen Wang, and Paris Perdikaris. Learning the solution operator of parametric partial differential equations with physics-informed deeponets. Science Advances, 7(40), October 2021. ISSN 2375-2548. doi: 10.1126/sciadv.abi8605. URL http://dx.doi.org/10.1126/sciadv.abi8605.

Slide 16:

- Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces. arXiv preprint arXiv:2108.08481, 2021
- Francesca Bartolucci, Emmanuel de Bezenac, Bogdan Raonic, Roberto Molinaro, Siddhartha Mishra, and Rima Alaifari. Representation equivalent neural operators: a framework for alias-free operator learning. In Thirty-seventh Conference on Neural Information Processing Systems, 2023b. URL https://openreview.net/forum?id=7LSEkvEGCM.

TU Delft

Deep Learning (DL) for Science

In the last decade a lot of bridges between DL and Science

nature

Explore content ⌄ About the journal ⌄ Publish with us ⌄

nature > articles > article

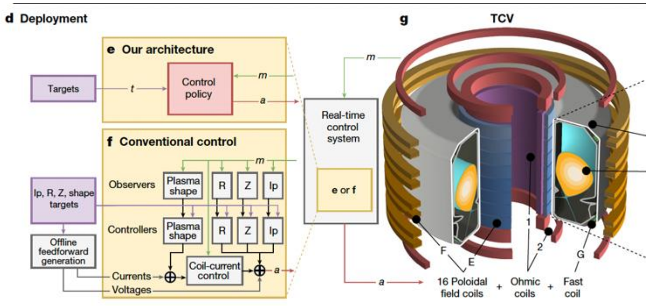
Article | Open access | Published: 16 February 2022

Magnetic control of tokamak plasmas through deep reinforcement learning

Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de las Casas, Craig Donner, Leslie Fritz, Cristian Galperti, Andrea Huber, James Keeling, Maria Tsimpoukelli, Jackie Kay, Antoine Merle, Jean-Marc Moret, Seb Noury, Federico Pesamosca, David Pfau, Olivier Sauter, Cristian Sommariva, ... Martin Riedmiller + Show authors

Nature 602, 414–419 (2022) | Cite this article

223k Accesses | 311 Citations | 2432 Altmetric | Metrics



Science

Current Issue First release papers Archive About ⌄ Submit manuscript

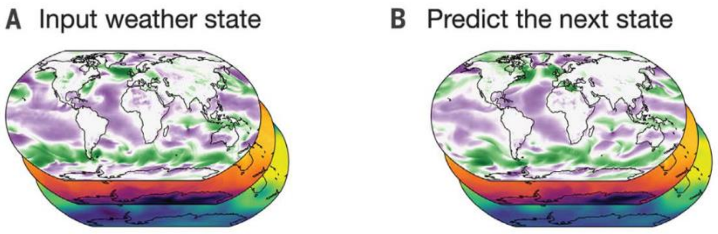
HOME > SCIENCE > VOL. 382, NO. 6677 > LEARNING SKILLFUL MEDIUM-RANGE GLOBAL WEATHER FORECASTING

RESEARCH ARTICLE | WEATHER FORECASTING

Learning skillful medium-range global weather forecasting

REMI LAM, ALVARO SANCHEZ-GONZALEZ, MATTHEW WILLSON, PETER WIRNSBERGER, MEIRE FORTUNATO, FERRAN ALET, SUMAN RAVURI, TIMO EWALDS, ZACH EATON-ROSEN, [...], AND PETER BATTAGLIA +8 authors Authors Info & Affiliations

SCIENCE • 14 Nov 2023 • Vol 382, Issue 6677 • pp. 1416-1421 • DOI: 10.1126/science.adi2336



nature

Explore content ⌄ About the journal ⌄ Publish with us ⌄

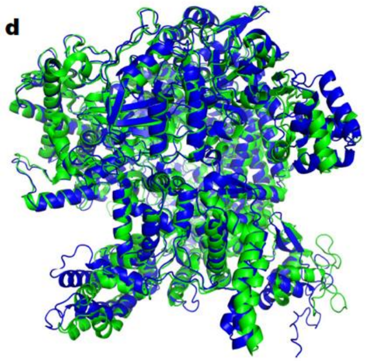
nature > articles > article

Article | Open access | Published: 15 July 2021

Highly accurate protein structure prediction with AlphaFold

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, ... Demis Hassabis + Show authors

d



My Background

- Bachelor in *Physics* at Universitá di Milano-Bicocca in 2020.

- Joint Master of Science in *Physics of Complex Systems* at Politecnico di Torino & Université Paris-Saclay in 2022.

- Started my PhD in 2023 in *Deep Learning for Surrogate Modeling*.

TUDelft