CERFACS

CENTRE EUROPÉEN DE RECHERCHE ET DE FORMATION AVANCÉE EN CALCUL SCIENTIFIQUE

Works in AI @ CERFACS

Luciano Drozda & Sébastien Villon

AIRBUS cnes EDF METEO FRANCE ONERA THE FRENCH AEROSPACE LAB SAFRAN TotalEnergies

AI @ CERFACS

Working topics

- LLMs for HPC (A. Dauptain et al. (18/02))
- Scientific Machine Learning
  - ✓ Data Assimilation
  - ✓ Statistical Downscaling
  - ✓ Extreme and rare event prediction
  - ✓ Ensemble generation
  - ✓ Subgrid-scale modeling
  - ✓ Numerical Methods for PDEs

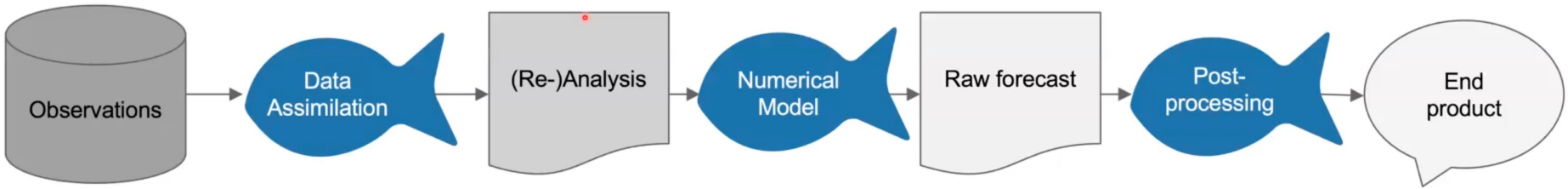
Full-time AI researchers

- **Sébastien Villon**
  - Expert in AI for Marine Ecology
- **Luciano Drozda**
  - Expert in AI for Numerical Methods

CERFACS

Data Assimilation and AI

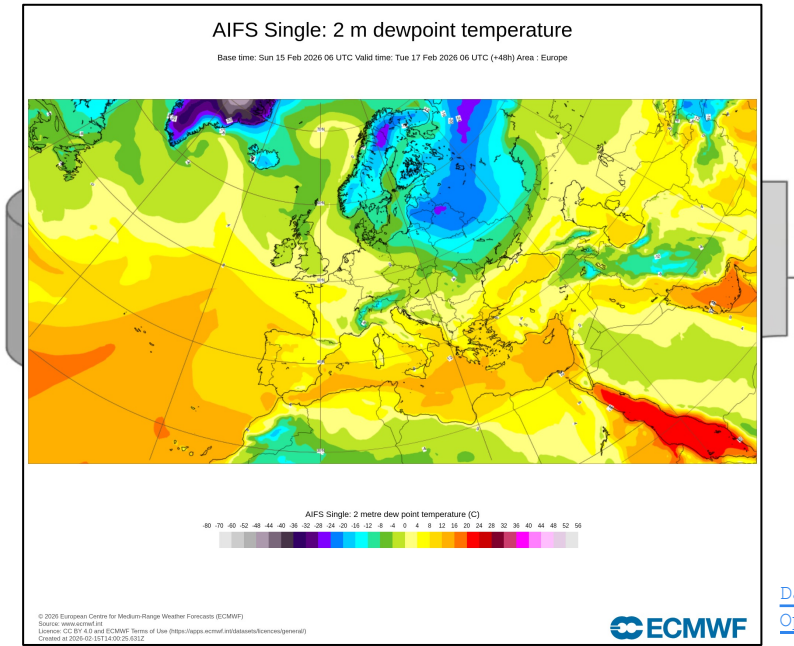
- Contact: **Selime Gürol**
- Will AI replace DA in the **Numerical Weather Prediction** chain ?

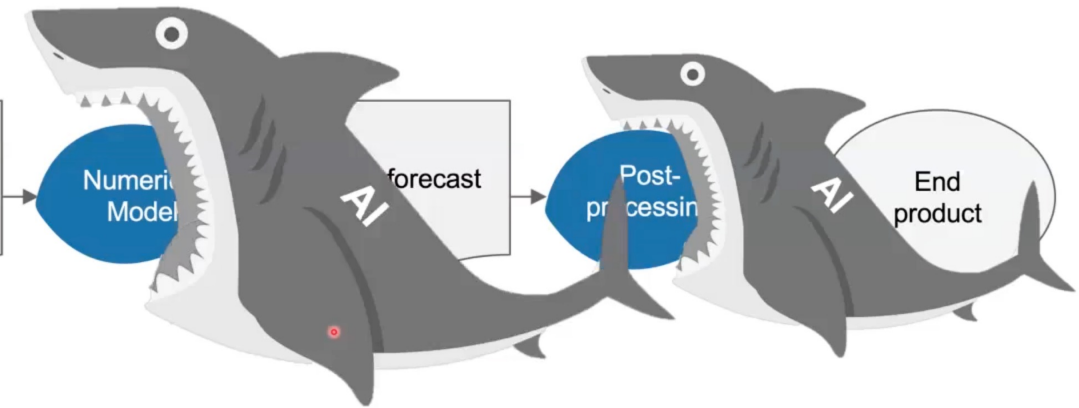


Data Assimilation training course (CERFACS), ECMWF's OpenCharts, and Stephan Rasp presentation in ISDA 2024

Data Assimilation and AI

- Contact: **Selime Gürol**
- Will AI replace DA in the **Numerical Weather Prediction** chain ?



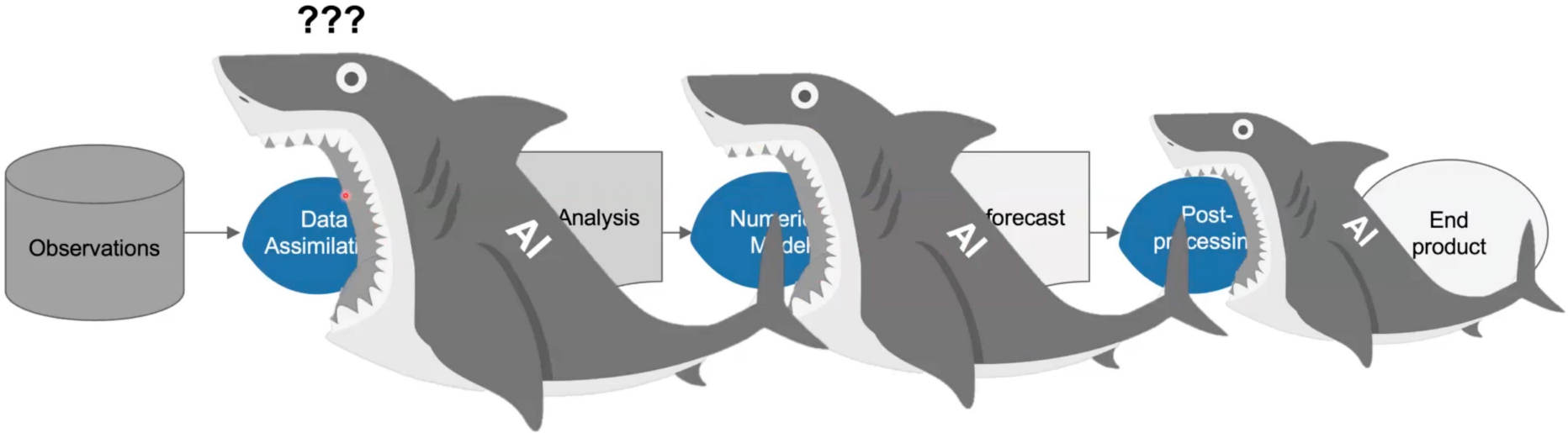


Data Assimilation training course (CERFACS), ECMWF's OpenCharts, and Stephan Rasp presentation in ISDA 2024

CERFACS

Data Assimilation and AI

- Contact: **Selime Gürol**
- Will AI replace DA in the **Numerical Weather Prediction** chain ?

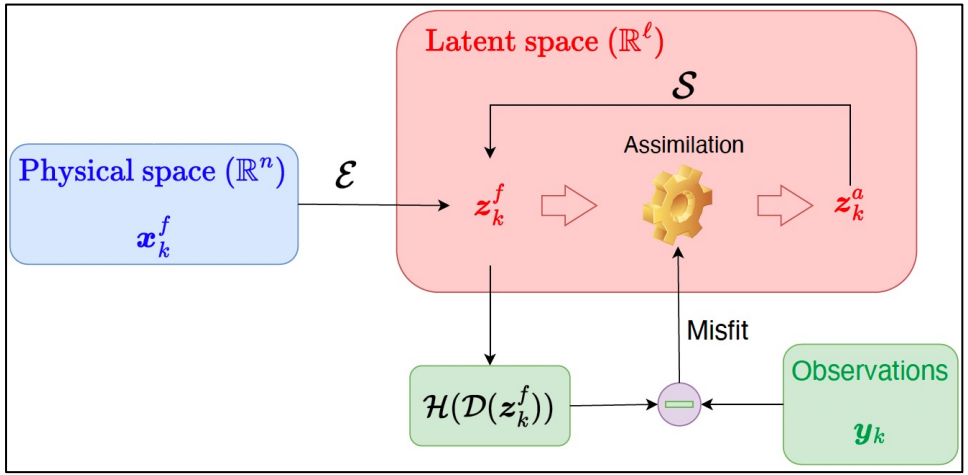


Data Assimilation training course (CERFACS), ECMWF's OpenCharts, and Stephan Rasp presentation in ISDA 2024

CERFACS

Data Assimilation and AI

- Contact: **Selime Gürol**
- **Latent Space** Data Assimilation by using Deep Learning[*]
- Proposed to perform ensemble filtering in the latent space obtained by an autoencoder
- **Reduced computational cost** and **better accuracy** levels than state-of-the-art algorithms



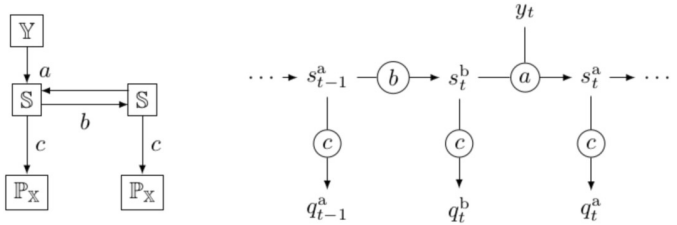
[*] Peyron, M., Fillion, A., Gürol, S., Marchais, V., Gratton, S., Boudier, P., & Goret, G. (2021). Latent space data assimilation by using deep learning. *Quarterly Journal of the Royal Meteorological Society*, *147*(740), 3759-3777. https://doi.org/10.1002/qj.4153

Data Assimilation and AI

- Contact: **Selime Gürol**
- Collaboration with IRIT

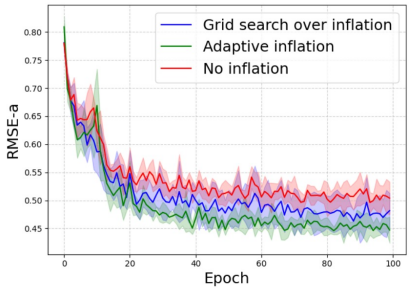
Data Assimilation Networks (DAN)[*]

Emulation of Data Assimilation:
A fully data-driven deep learning framework that generalizes traditional data assimilation algorithms.



Differentiable DA:

- **Improved inference based on approximate data log-likelihood**
- Joint learning of dynamical model parameters and DA regularization parameters

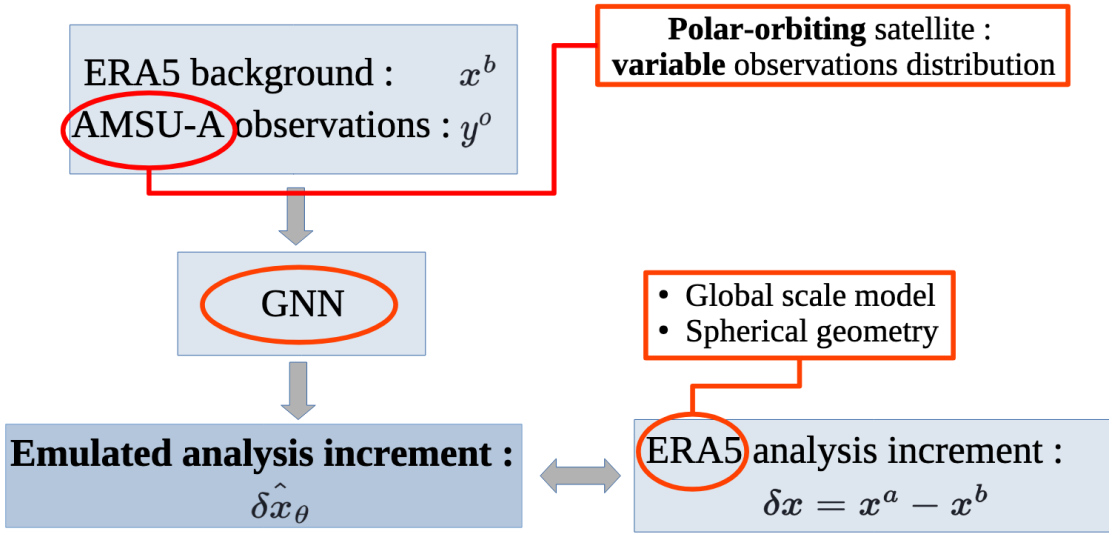


[*] Boudier, P., Fillion, A., Gratton, S., Gürol, S., & Zhang, S. (2023). Data assimilation networks. *Journal of Advances in Modeling Earth Systems*, 15, e2022MS003353. https://doi.org/10.1029/2022MS003353

CERFACS

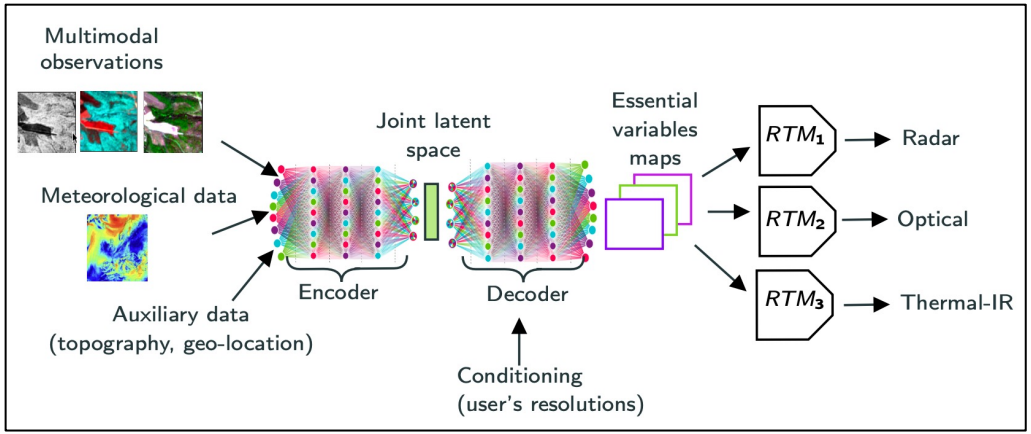
Data Assimilation and AI

- Contact: **Selime Gürol**
- **Data assimilation emulation** in a global spherical framework
- Co-supervision of Météo-France FCPLR PhD thesis



Data Assimilation and AI

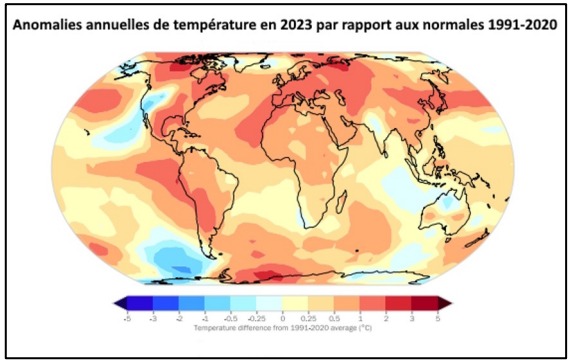
- Contact: **Selime Gürol**
- REpresentation Learning for Earth Observation (RELEO)
- **Goal:** to develop **probabilistic self-supervised** learning methods that produce semantically meaningful representations from **multi-modal time series** data



- Contact: **Sébastien Villon**

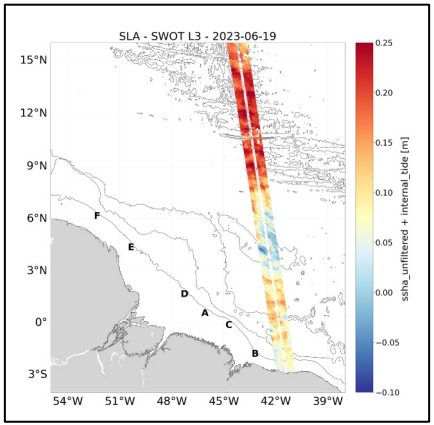
Climate model emulation

- **Generation** of large ensemble members
- Focus on climate extrema



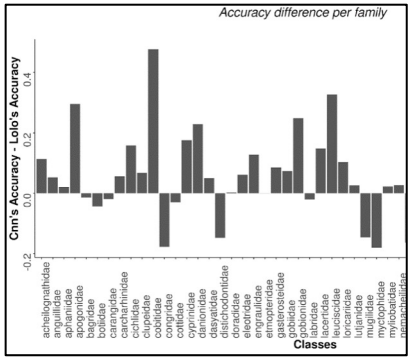
Detection of complex marine patterns

- **Detection** and characterization of internal solitary waves by processing SWOT signals



Marine Ecology

- **Monitoring**, forecasting
- Focus on rare species

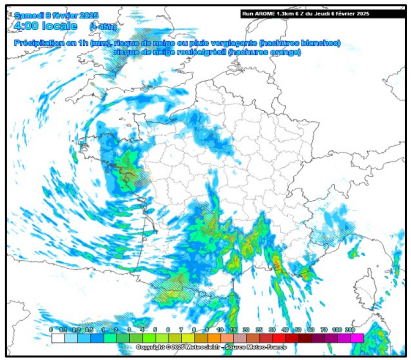


- Contact: **Sébastien Villon**

FRANCE 2030 PROGRAMME DE RECHERCHE CLIMAT

Weather forecasting

- **Downscaling**
- **Emulation** of weather forecasting models
- Focus on extrema: cold/heat waves, precipitation, and cyclones

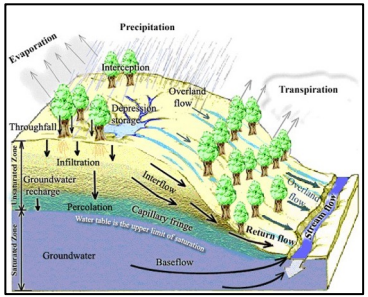


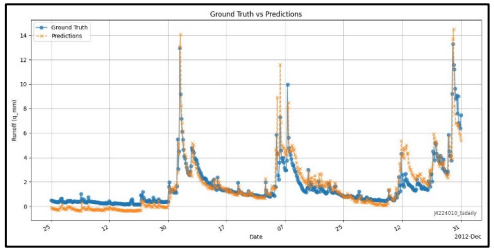
METEO FRANCE

CNRM

Hydrology

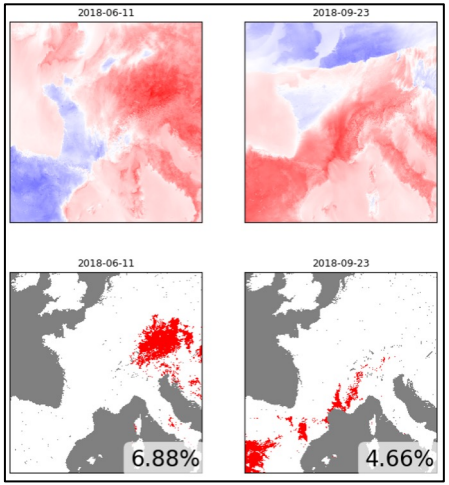
- Modeling run-off over time
- Focus on floods and sediment transport





Extreme detection and data enrichment

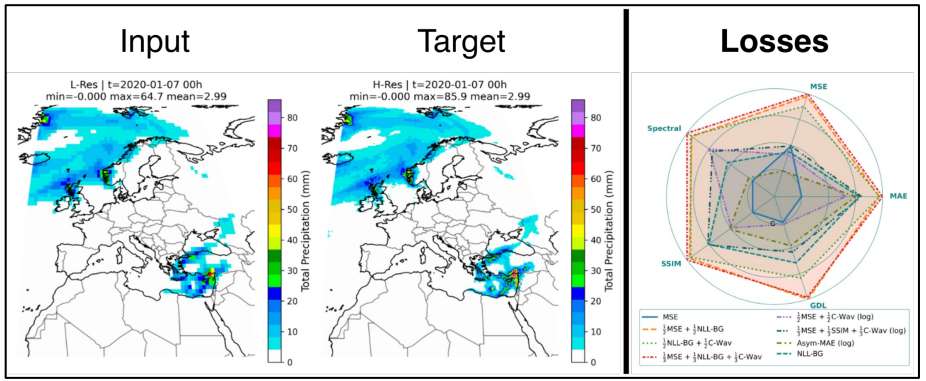
- Extreme hot days detection (AROME area)
- **Generation** of new realistic extreme hot days



CERFACS

Statistical downscaling

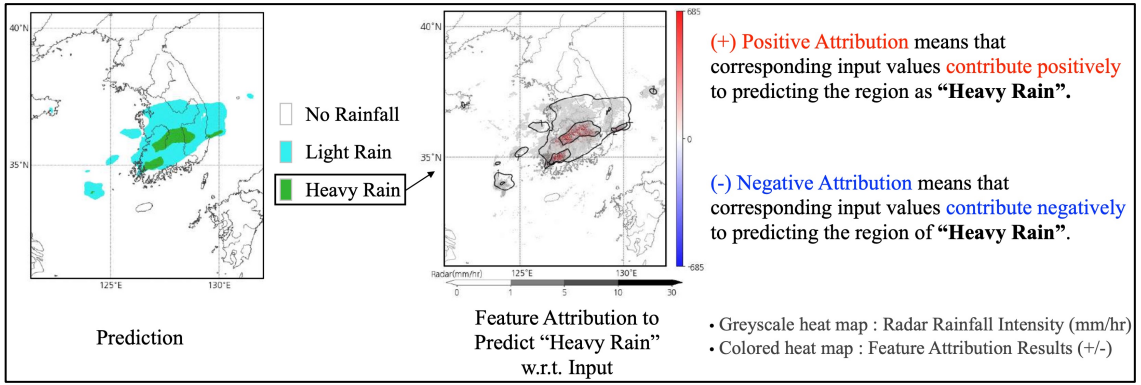
- Can AI models perform **reliable future climate projections** by downscaling coarse-grained simulations?
- At CERFACS, first works benchmarked the impact of the choice of **loss function** in downscaling wind and precipitation fields using Vision Transformers
- Preprint available[*]



[*] Montassir, R. E., Drozda, L., Pannekoucke, O., Terray, L., Omrani, H., & Verny, L. (2025). *Training loss matters: Exploring new fit-for-purpose loss combinations for statistical downscaling*. https://doi.org/10.22541/essoar.175977614.45331660/v1

Explainable AI

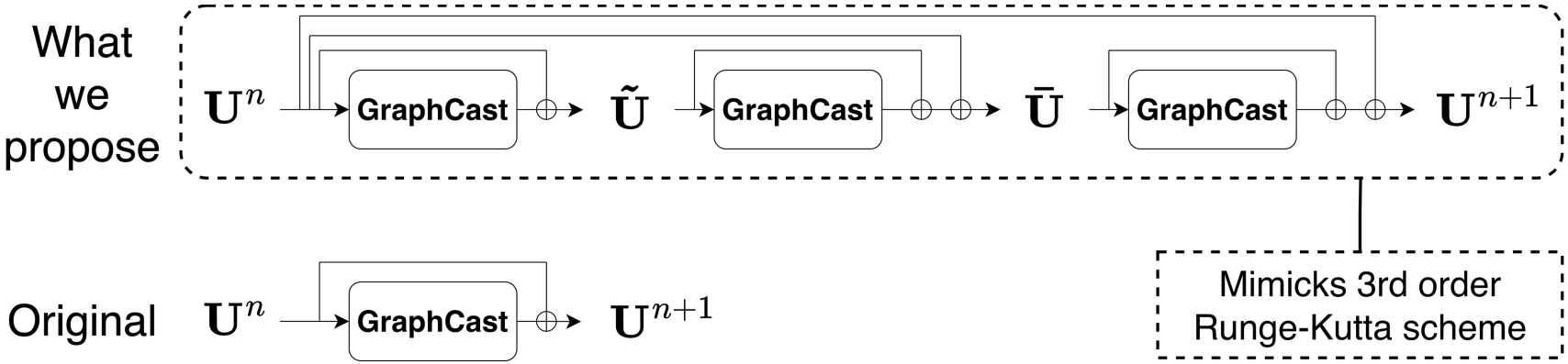
- **Why** does an AI model predict precipitation at a given area?
- **Gradients** of target variables with respect to each input are locally computed to measure which inputs are relevant to the prediction
- **Goal:** to bring **trustworthiness** to AI-based weather forecasting



Kim, S., Choi, Junho, Choi, Y., Lee, S., Stitsyuk, A., Park, M., Jeong, S., Baek, Y.-H., & Choi, Jaesik. (2023). Explainable ai-based interface system for weather forecasting model. In H. Degen, S. Ntoa, & A. Moallem (Eds), *HCI International 2023 – Late Breaking Papers* (Vol. 14059, pp. 101–119). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-48057-7_7

Hybridization

- Can **PDE theory** design **more accurate** AI models for weather forecasting?
- GraphCast[*] training relates to a first order time-stepping scheme
- **Idea:** to train GraphCast mimicking **high-order time-stepping** schemes



[*] Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Alet, F., Ravuri, S., Ewalds, T., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Vinyals, O., Stott, J., Pritzel, A., Mohamed, S., & Battaglia, P. (2023). Learning skillful medium-range global weather forecasting. *Science*, *382*(6677), 1416-1421. https://doi.org/10.1126/science.adi2336

PDE solvers in the era of AI

- Three paradigms for solving PDEs

CERFACS
CENTRE EUROPÉEN DE RECHERCHE ET DE FORMATION AVANCÉE EN CALCUL SCIENTIFIQUE

Discretization schemes

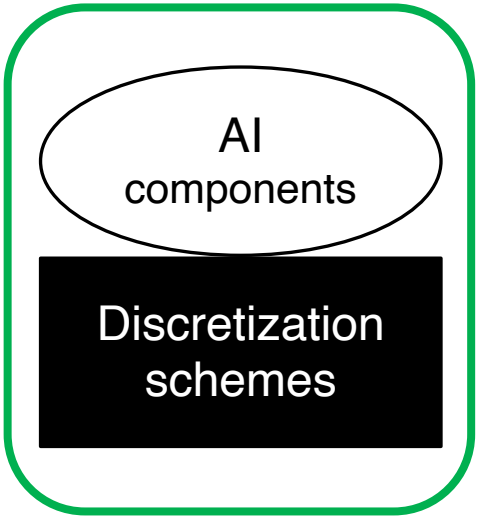
✅ Pros
1. Based on rigorous numerical analysis
2. Interpretable

❌ Cons
1. Expensive
2. Ad-hoc assumptions

AI surrogate

✅ Pros
1. Run fast on specific hardware (GPU)
2. Easy to implement

❌ Cons
1. Hard to interpret
2. Data-hungry

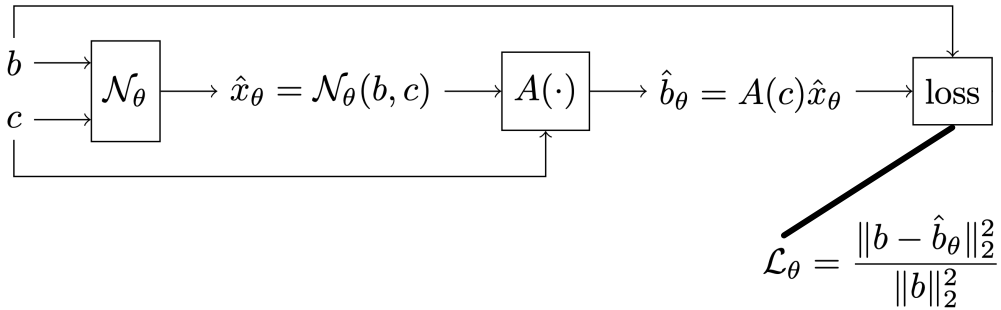


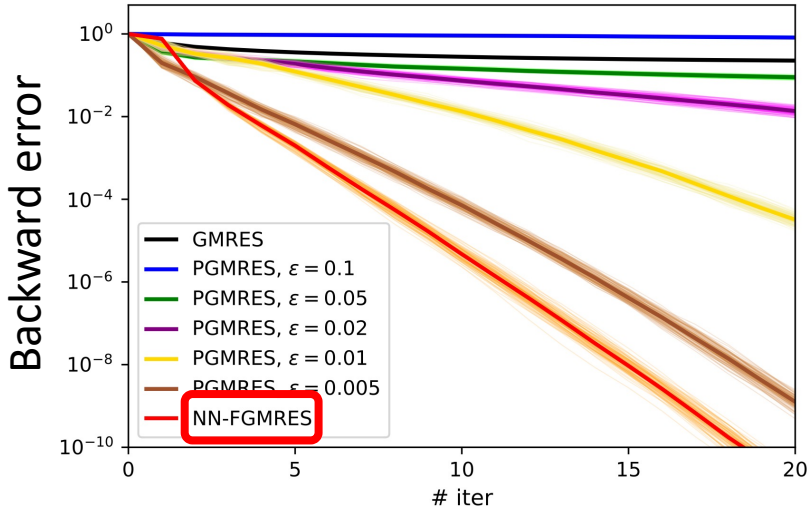
Physics-AI solvers

Numerical Linear Algebra and AI

- Contacts: **Luc Giraud** (Inria CONCACE team), **Paul Mycek**, **Carola Kruse**
- **Preconditioning** of linear systems arising from the discretization of the two-dimensional parametric **Helmholtz** equation using a **U-Net** model

$$A(c)x = b$$

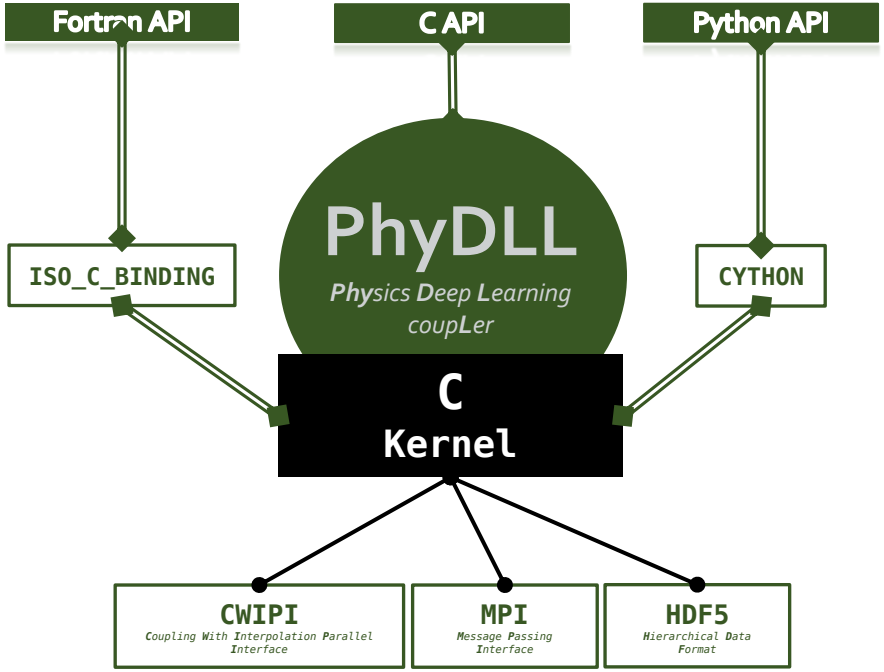




Luc Giraud, Carola Kruse, Paul Mycek, Maksym Shpakovych, Yanfei Xiang. Neural network preconditioning: a case study for the solution of the parametric Helmholtz equation. RR-9593, Inria Centre at the University of Bordeaux, France. 2025. ⟨hal-05157038⟩

Coupling AI models with legacy PDE solvers

- **PhyDLL** is the coupler developed at CERFACS and used by European partners (BSC, RWTH Aachen, Météo-France)
- Neural networks coded in **PyTorch** exchange fields with legacy physical solvers in **Fortran, C**
- Added value: handles **mesh partitioning**



Serhani, A., Xing, V., Dupuy, D., Lapeyre, C., & Staffelbach, G. (2024). Graph and convolutional neural network coupling with a high-performance large-eddy simulation solver. *Computers & Fluids*, *278*, 106306. https://doi.org/10.1016/j.compfluid.2024.106306

PhyDLL + CERFACS CFD solver AVBP

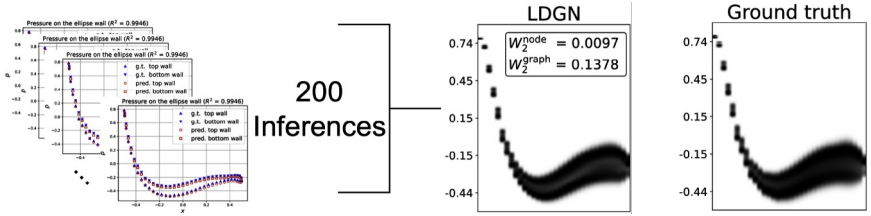
Wall modeling
using Graph Neural Networks[*]

Surrogate modeling for wall shear stress or wall conductive heat flux

Near-wall flow reconstruction
using Generative AI

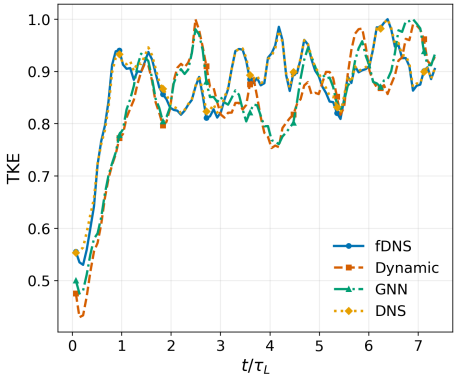
Recent works on Latent Diffusion Graph Networks (LDGN) for reconstructing quantities in converged flows presented promising results

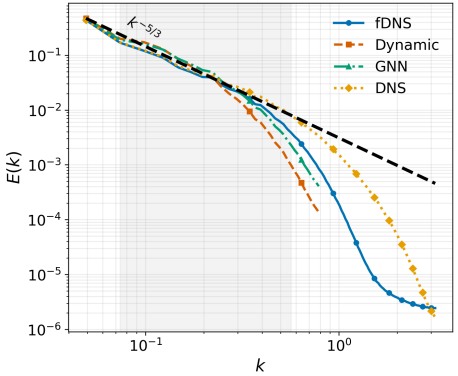
Ongoing extension to forced Homogeneous Isotropic Turbulence and Channel flows



Subgrid-scale modeling
using Graph Neural Networks

First a posteriori tests show improvement over standard models such as Dynamic Smagorinsky for Homogeneous Isotropic Turbulence



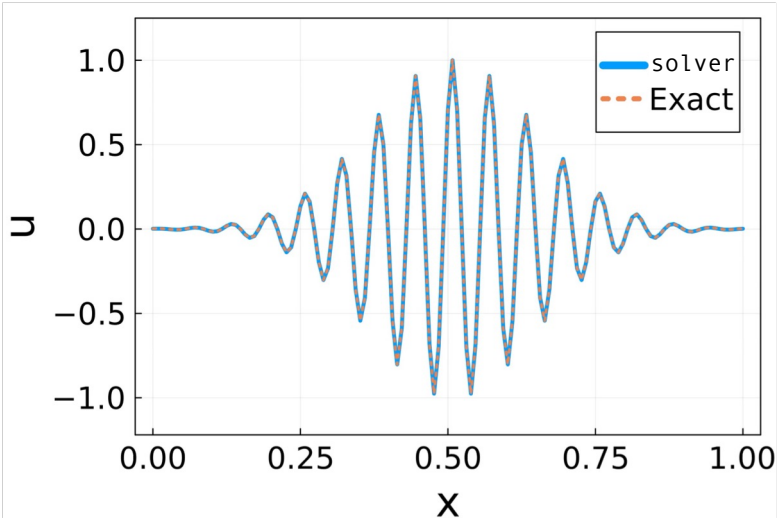


[*] Dupuy, D., Odier, N., & Lapeyre, C. (2024). Using graph neural networks for wall modeling in compressible anisothermal flows. *Data-Centric Engineering*, *5*, e10. https://doi.org/10.1017/dce.2024.7

Building hybrid Physics-AI solvers at ease

- **Goal:** to **generate** (GPU-ported) adjoint code from scalar primal one

$$u_t + c\, u_x = 0$$

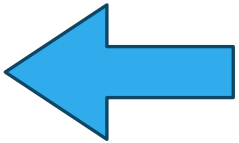



```
function solver(...)

  # Loop over time steps
  for i_seq_ = 1:i_nstep

    # Compute finite-differences
    for i_x = 2:i_nnode
      du[i_x] = u[i_x] - u[i_x - 1]
    end

    # Update solution
    for i_x = 2:i_nnode
      u[i_x] = u[i_x] - c * dt * du[i_x] / dx
    end

  end

end
```

Building hybrid Physics-AI solvers at ease

- **Goal:** to **generate** (GPU-ported) adjoint code from scalar primal one

```
function solver_b(...)
    i_du_stack = 1
    for i_seq_ = 1:i_nstep
        du_stack[i_du_stack, :] .= du
        [...]
    end
    for i_seq_ = i_nstep:-1:1
        i_du_stack -= 1
        du .= du_stack[i_du_stack, :]
        for i_x = 2:i_nnode
            temp = du[i_x] / dx
            cb = cb - dt * temp * ub[i_x]
            dtb = dtb - c * temp * ub[i_x]
            tempb = -((c * dt * ub[i_x]) / dx)
            dub[i_x] = dub[i_x] + tempb
            dxb = dxb - temp * tempb
        end
        i_du_stack -= 1
        du .= du_stack[i_du_stack, :]
        for i_x = 2:i_nnode
            ub[i_x] = ub[i_x] + dub[i_x]
            ub[i_x - 1] = ub[i_x - 1] - dub[i_x]
            dub[i_x] = 0.0
        end
    end
    return (cb, dxb, dtb)
end
```



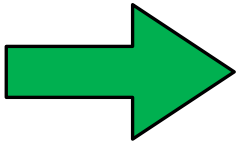
Generate!
(adjoint)

```
function solver(...)

  # Loop over time steps
  for i_seq_ = 1:i_nstep

    # Compute finite-differences
    for i_x = 2:i_nnode
      du[i_x] = u[i_x] - u[i_x - 1]
    end

    # Update solution
    for i_x = 2:i_nnode
      u[i_x] = u[i_x] - c * dt * du[i_x] / dx
    end

  end

end
```

Building hybrid Physics-AI solvers at ease

- **Goal:** to **generate** (GPU-ported) adjoint code from scalar primal one

```
function solver_b(...)
    i_du_stack = 1
    for i_seq_ = 1:i_nstep
        du_stack[i_du_stack, :] .= du
        [...]
    end
    for i_seq_ = i_nstep:-1:1
        i_du_stack -= 1
        du .= du_stack[i_du_stack, :]
        for i_x = 2:i_nnode
            temp = du[i_x] / dx
            cb = cb - dt * temp * ub[i_x]
            dtb = dtb - c * temp * ub[i_x]
            tempb = -((c * dt * ub[i_x]) / dx)
            dub[i_x] = dub[i_x] + tempb
            dxb = dxb - temp * tempb
        end
        i_du_stack -= 1
        du .= du_stack[i_du_stack, :]
        for i_x = 2:i_nnode
            ub[i_x] = ub[i_x] + dub[i_x]
            ub[i_x - 1] = ub[i_x - 1] - dub[i_x]
            dub[i_x] = 0.0
        end
    end
    return (cb, dxb, dtb)
end
```



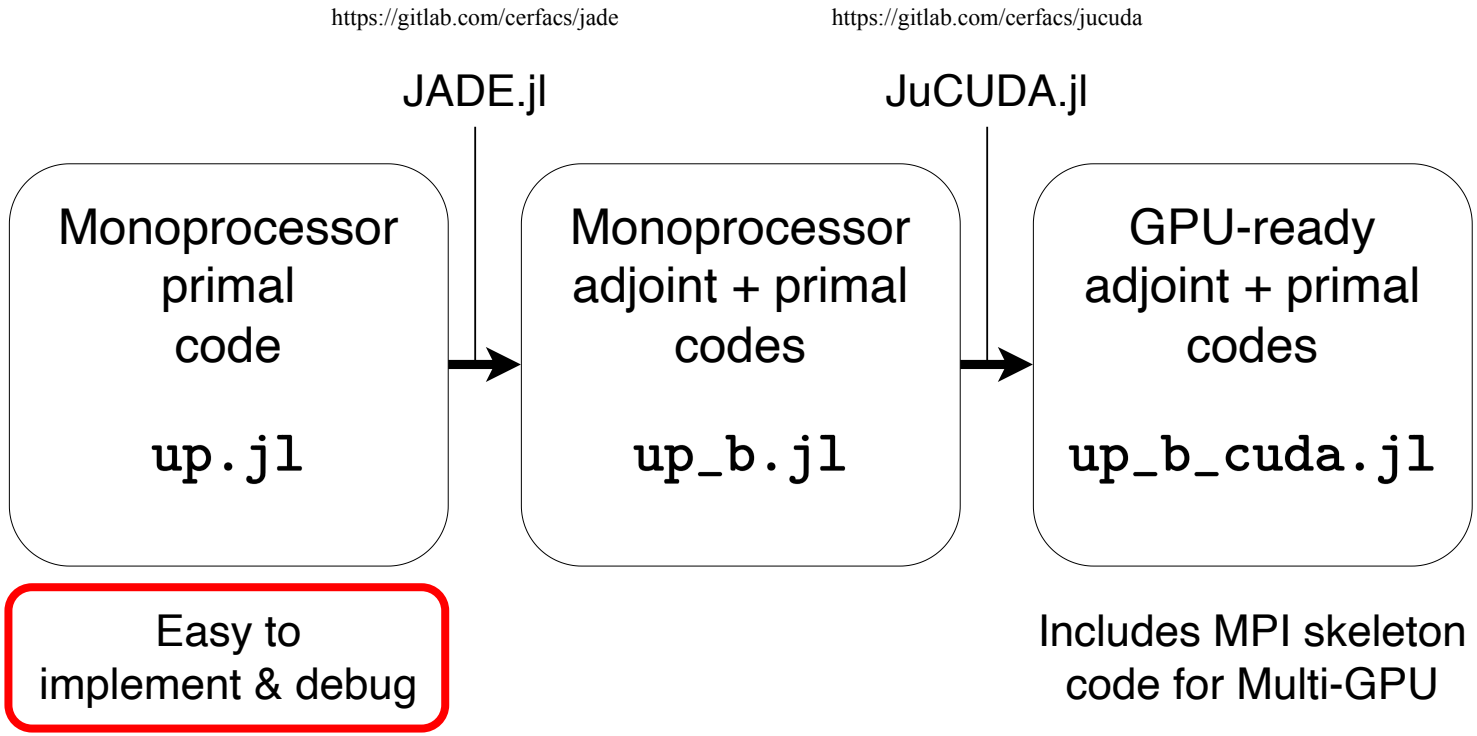
Generate!

(GPU-ported)

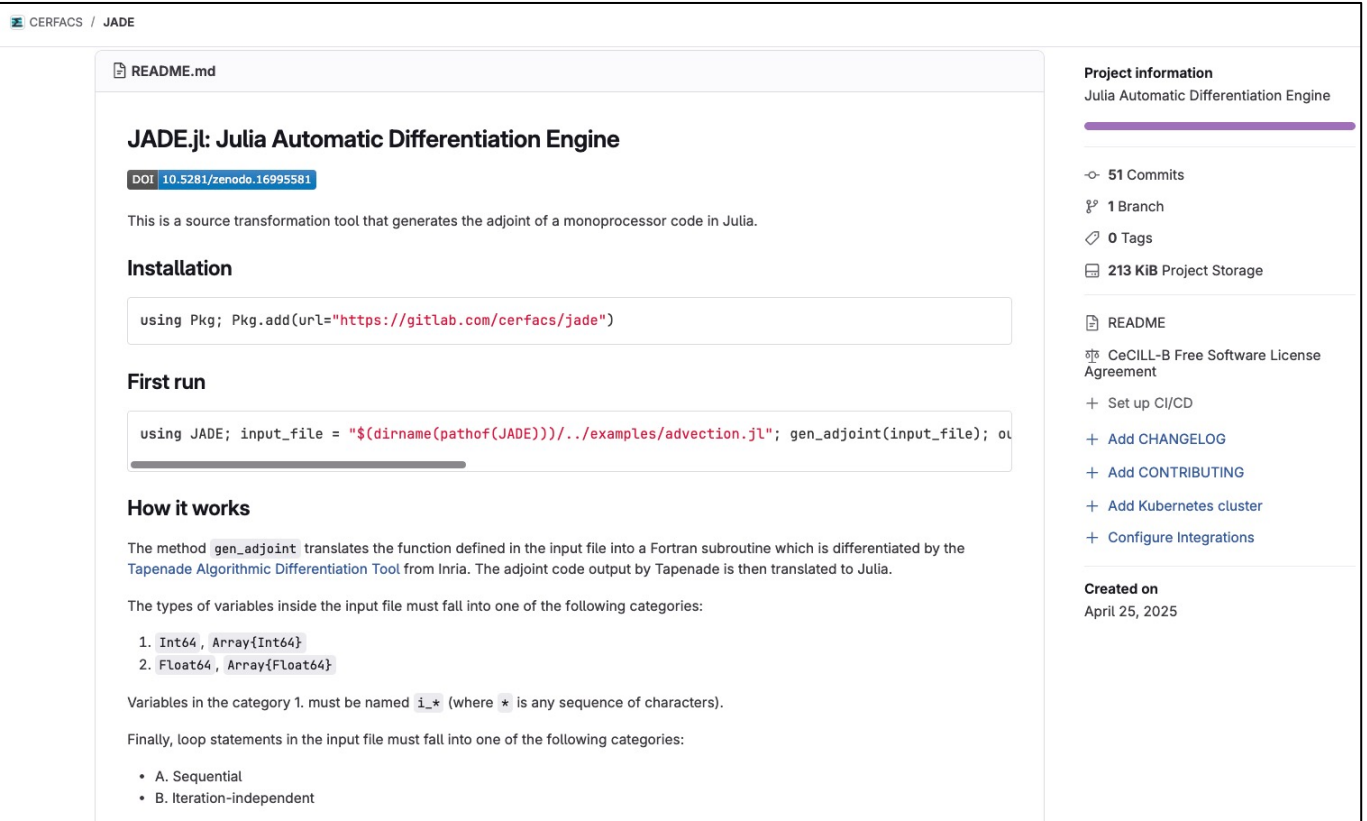
```
function solver_b_cuda(...)
    nthread_per_block = 256
    i_du_stack = 1
    for i_seq_ = 1:i_nstep
        du_stack[i_du_stack, :] .= du
        [...]
    end

    for i_seq_ = i_nstep:-1:1
        i_du_stack -= 1
        du .= du_stack[i_du_stack, :]
        @cuda threads = nthread_per_block blocks =
ceil(Int, ((i_nnode - 2) + 1) / nthread_per_block)
func_b_loop3!(...)
        i_du_stack -= 1
        du .= du_stack[i_du_stack, :]
        @cuda threads = nthread_per_block blocks =
ceil(Int, ((i_nnode - 2) + 1) / nthread_per_block)
func_b_loop4!(...)
    end

    return (cb, dxb, dtb)
end
```

Verifiable and optimizable by the end user!

Building hybrid Physics-AI solvers at ease



Building hybrid Physics-AI solvers at ease



CERFACS / JADE

README.md

JADE.jl: Julia Automatic Differentiation Engine

DOI 10.5281/zenodo.16995581

This is a source transformation tool that generates the adjoint of a monoprocessor code in Julia.

Installation

```
using Pkg; Pkg.add(url="https://gitlab.com/cerfacs/jade")
```

First run

```
using JADE; input_file = "$(dirname(pathof(JADE)))/../examples/advection.jl"; gen_adjoint(input_file); ou
```

How it works

The method `gen_adjoint` translates the function defined in the input file into a Fortran subroutine which is differentiated by the Tapenade Algorithmic Differentiation Tool from Inria. The adjoint code output by Tapenade is then translated to Julia.

The types of variables inside the input file must fall into one of the following categories:

1. `Int64`, `Array{Int64}`
2. `Float64`, `Array{Float64}`

Variables in the category 1. must be named `i_*` (where `*` is any sequence of characters).

Finally, loop statements in the input file must fall into one of the following categories:

- A. Sequential
- B. Iteration-independent

Project information
Julia Automatic Differentiation Engine

51 Commits
1 Branch
0 Tags
213 KiB Project Storage

README
CeCILL-B Free Software License Agreement
Set up CI/CD
Add CHANGELOG
Add CONTRIBUTING
Add Kubernetes cluster
Configure Integrations

Created on
April 25, 2025

You can try them!

https://gitlab.com/cerfacs/jade

https://gitlab.com/cerfacs/jucuda

Building hybrid Physics-AI solvers at ease

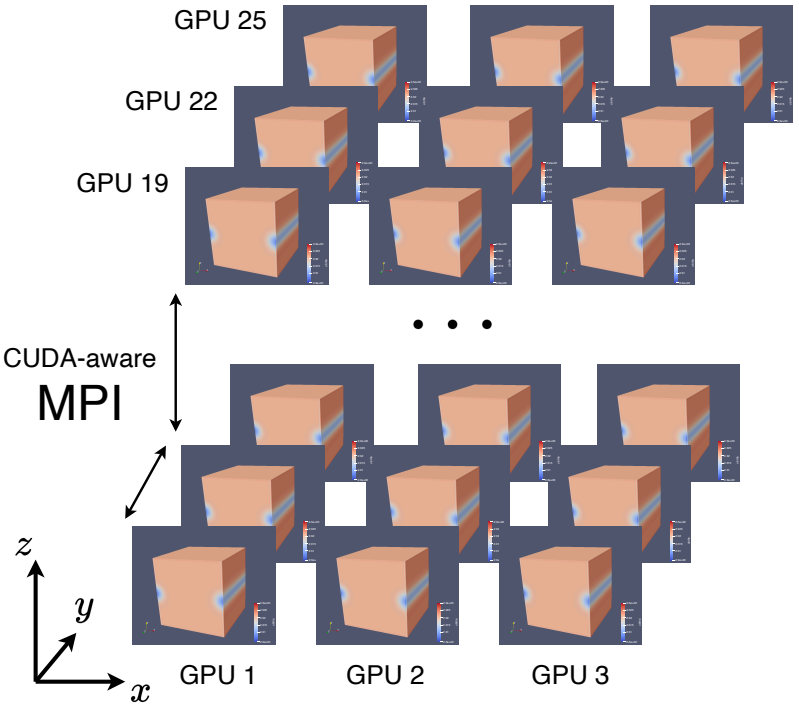
- A **demonstrator** is available at https://gitlab.com/cerfacs/ttgc

$$\text{Loss}\left(\mathbf{U}^n\right) := \|\rho^{n+1} - \rho_{\text{ref}}^{n+1}\|_2^2$$

$$\frac{\partial}{\partial \rho^n}\text{Loss}\left(\mathbf{U}^n\right)$$

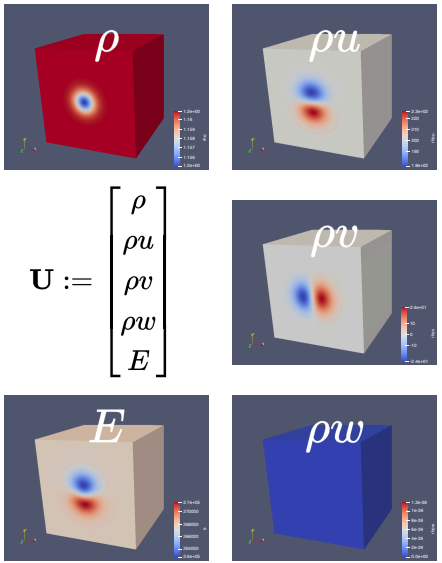
computed across
27 GPUs NVIDIA A30

$$\#\text{ Mesh cells} = 27 \times 15.09\text{ M} \approx 407\text{ M}$$



$$\mathbf{U}^{n+1} = \text{TTGC}\left(\mathbf{U}^n\right)$$

$$\mathbf{U} := \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix}$$



- CERFACS aims at providing **trustworthy and industry-grade AI** to their partners
- Hybridization, Explainability, Multi-modality, LLMs for HPC (**A. Dauptain et al. (18/02)**)

1. **Data Assimilation**: Emulation using Anemoi framework; Learning dynamics and analysis directly from observations
2. **Weather, climate, environment**: Leveraging existing data and AI architectures to provide solutions with improved runtime-accuracy tradeoff
3. **Coupling**: Enhancing PhyDLL library to support online learning
4. **Numerical methods for PDEs**: Employing JADE + JuCUDA libraries to democratize building of scalable Physics-AI solvers

CERFACS

CENTRE EUROPÉEN DE RECHERCHE ET DE FORMATION AVANCÉE EN CALCUL SCIENTIFIQUE

Thank you

AIRBUS cnes EDF METEO FRANCE ONERA THE FRENCH AEROSPACE LAB SAFRAN TotalEnergies