Data-driven machine learning techniques for wind farm-scale flow simulations

– Implementing MCIA HPC resources

Nishant KUMAR

MEMPHIS Group
Inria, Bordeaux – Sud Ouest

JSMCIA2022 : Journée Scientifique 2022 du Mésocentre de Calcul Intensif Aquitain
Institut de Mathématiques de Bordeaux, Talence (FRANCE)

21 October 2022

ARIA Project

ARIA (Accurate ROMs for Industrial Applications) project to develop computationally accurate, robust and efficient predictive Reduced-Order Models for applications with complex physical phenomena.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 872442 (ARIA).

Focus on renewable energy: Joint collaboration Inria-IFPen

Model large-size offshore wind turbines: 6-12 MW

- Data-driven machine learning techniques for wind farm-scale flow simulations.
- Coupling actuator lines (SOWFA) with data-driven machine learning for the wake.

Context

Motivation

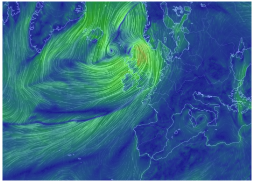
Reduction of the computational cost of high fidelity, wind farm scale simulations.

Approach

Collate several single wind turbine models and an appropriate propagation model.

Simulations with realistic flow conditions $\longrightarrow$ Data $\longrightarrow$ Data-driven modeling

Mesoscale flow

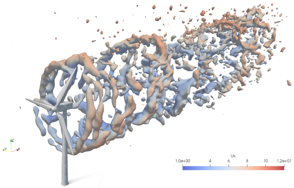


Turbine

Plant Array

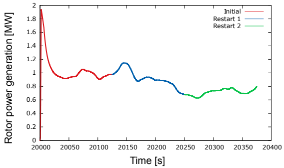
MCIA resources

Simulation



OpenFOAM
SOWFA
...

Post-processing

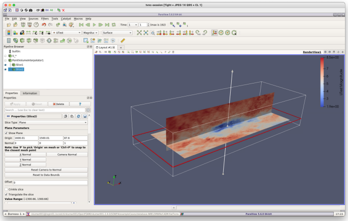


Python
C++
...

HPC tools

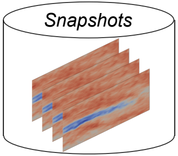
SLURM, bash, Modules, Redmine,
HPC support, ...

Visualization



TurboVNC
gnuplot
ParaView
...

Storage



/gpfs
/scratch
iRODS
...

Outline

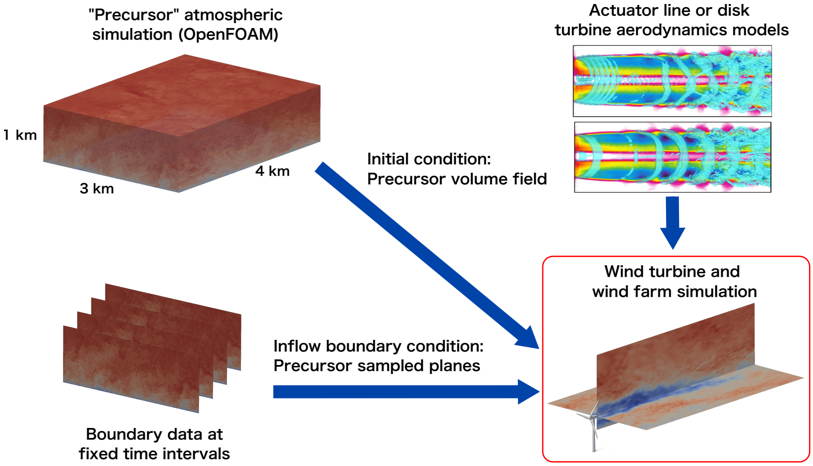
1. Flow simulation
2. Post-processing
3. In-sample prediction
4. Conclusion

Outline

1. Flow simulation
2. Post-processing
3. In-sample prediction
4. Conclusion

Wind turbine simulation

- Simulator fOr Wind Farm Applications
- Solver to investigate wind turbine and wind plant performance under the full range of atmospheric conditions and terrains.



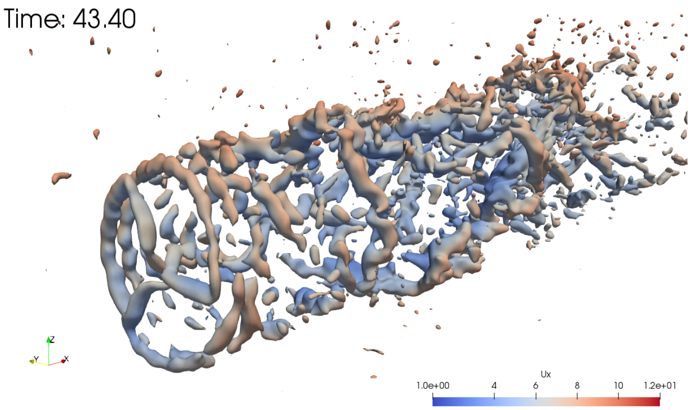
Wind turbine simulation from precursor fields:

- $\sim$**50 million** cells
- **180** Intel®Xeon®Gold SKL-6130 2.1 GHz processors on texttttcompute SD530 nodes of the MCIA *Curta* cluster
- $\sim$**6.5 days** of computation time to simulate $\sim$**400 s** of flow

Simulation output

- **Computational domain**:
  4 km × 3 km × 1 km
- **Atmospheric boundary layer**:
  Neutral
  LES SGS Model: One-equation eddy viscosity model
  Solver: PISO
  Streamwise velocity magnitude at hub-height = 8 m/s
  Aerodynamic roughness height = 1E-2 m (level grass plain)
- **Wind turbine**: NREL 5 MW Ref.
  Turbine models:
  - Actuator line model - Advanced
  - Actuator disc model

Postprocess in *ParaView* on *Curta*'s `visu` node

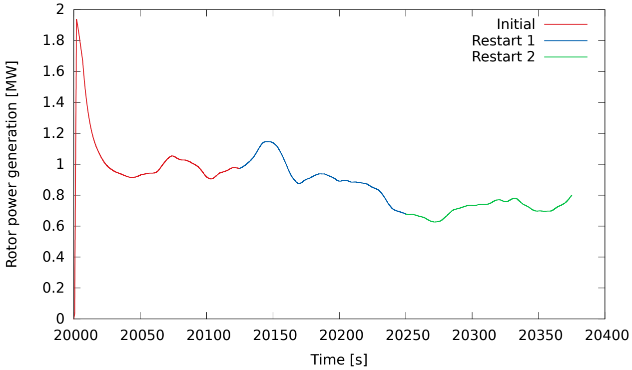


Iso-surface of Q-criterion (Q=0.012 s^{-2}) colored by streamwise velocity magnitude

Simulation output

Sampling of field variables:

- ABL develops up to 20000 s after which precursor data is collected.
- ADM simulation is initiated and sampling is performed:
  - Fields: `U T p p_rgh kSGS nuSgs`
  - Sampling time step: 0.35 s (equivalent to $\sim$ 14 snapshots per rotation)
  - Sampling time window: 20000 – 20400 s



Simulation output

Snapshot generation:

- $\sim$1 million cells near the wind turbine.
- 1150 snapshots account for a disk space of $\sim$150 GB.

iRODS storage:

- Consistent I/O with `tar` compression.

```
$ tar -cvf <tar-file.tar> <source-directory> # Create archive
$ iput -fPvDtar <tar-file.tar> <irods-destination> # Upload archive
```

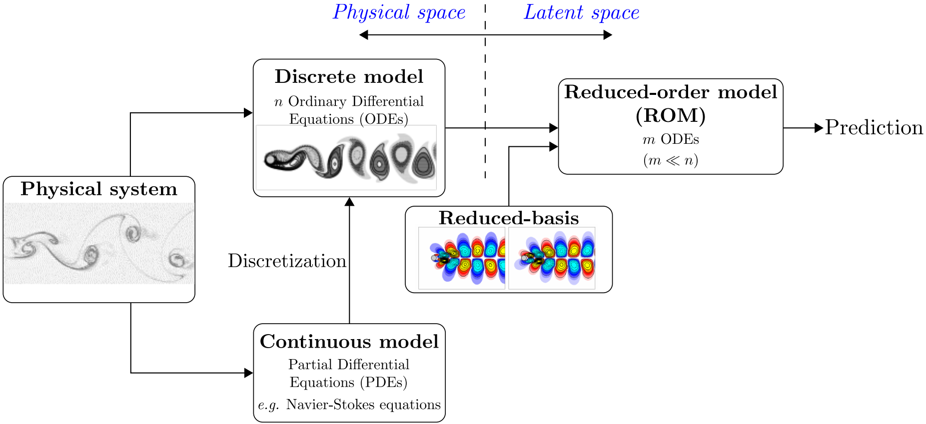
- Access the archived directory by remotely mounting it on *iRODS*.

```
$ imcoll -m tar <tar-file.tar> <mount-directory> # Mount
$ imcoll -Usp <mount-directory> # Unmount
```

Outline

1. Flow simulation
2. Post-processing
3. In-sample prediction
4. Conclusion

Model order reduction



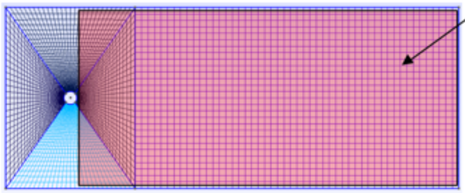
Motivation: Representation of complex high-dimensional data in a simpler low-dimensional space such that the dominant characteristics are preserved.

Proper Orthogonal Decomposition (POD)

POD modal description of velocity $\widetilde{\mathbf{u}}$ field in a low-fidelity domain (Ω_{LF}):

$$\widetilde{\mathbf{u}}(\mathbf{x},t) = \mathbf{f}_u(\mathbf{x},t) + \sum_{i=1}^{N} \widehat{u}_i(t)\boldsymbol{\Phi}_i(\mathbf{x}), \quad \widehat{u}_i \in \mathbb{R}, \boldsymbol{\Phi}_i \in \mathbb{R}^d, \mathbf{x} \in \Omega_{LF}.$$

Example: Laminar vortex shedding over a cylinder at $Re = 100$

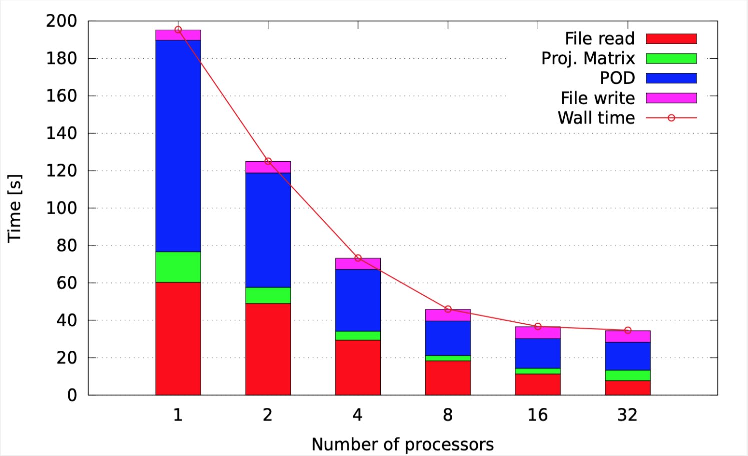


POD - Laminar vortex shedding

MPI-based parallel POD implementation in C++:

parallel-pod – Source code: https://gitlab.ifpen.fr/supercalcul/rom4wt/-/tree/main/POD

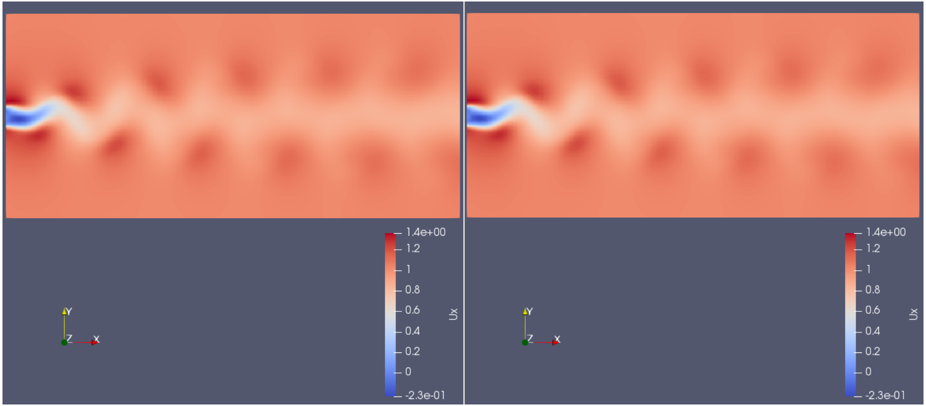
Runtimes of POD calculation



POD - Laminar vortex shedding

Normalized RMSE = 1.21E-5 (for all configurations):

Simulation Reconstruction

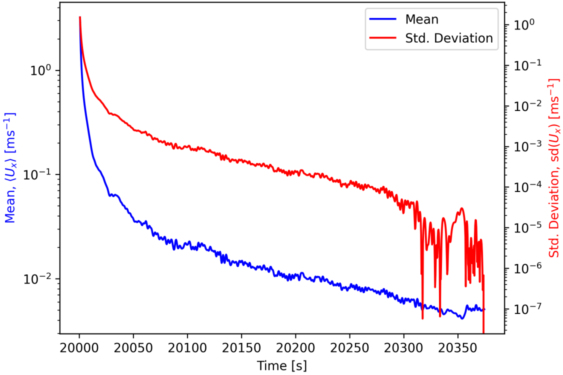


Streamwise velocity at $t = 200$ s.

POD of SOWFA simulation dataset

Statistical stationarity of snapshots:

- Streamwise velocity probed at 15D downstream of the turbine.
- Snapshot files read in parallel using `mpi4py` module.
  Speedup of 15x with 32 processors as compared to serial implementation.



POD of SOWFA simulation dataset

Dataset:

- $N_t = 370$ snapshots
- Each snapshot contains three components of velocity vector (u_x, u_y, u_z) stored at $N_p = 1081608$ data points.

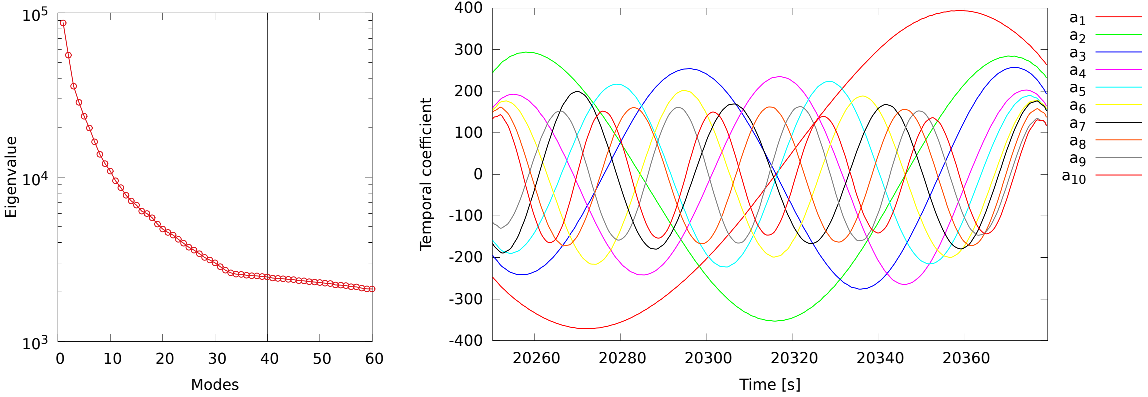
MPI-based parallel POD implementation in C++:

Task (32 processors)	Time
Reading snapshots...	43.7s
Computing projection matrix...	29.2s
Computing POD modes...	105.1s
Writing 40 POD modes...	179.8s
Wall time	357.9s

POD of SOWFA simulation dataset

Snapshot POD of SOWFA simulation dataset:

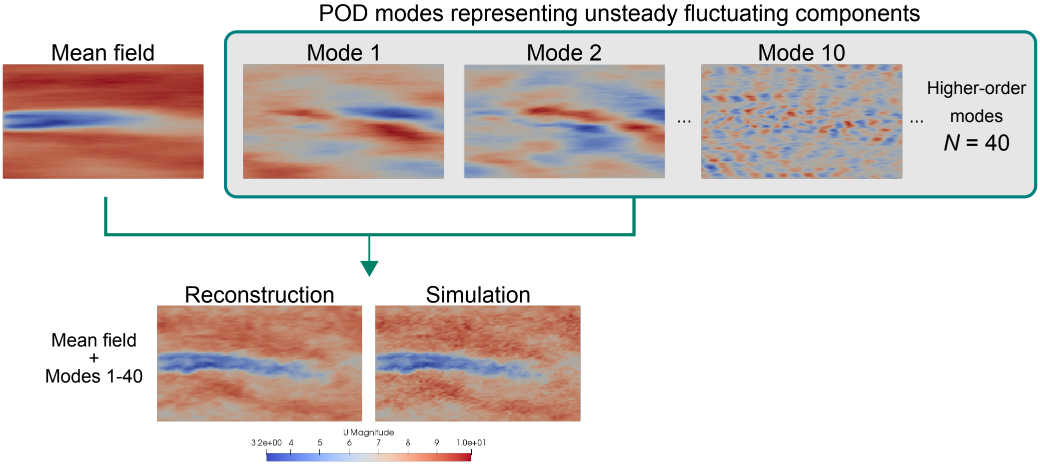
- Relative information content, $RIC \approx 70\%$ corresponding to 40 most energetic modes.



POD of SOWFA simulation dataset

Snapshot POD of SOWFA simulation dataset:

- In-sample reconstruction: Average $NRMSE = 1.419 \times 10^{-4}$, Max $NRMSE = 0.1406$.



Outline

In-sample prediction

The time-series data is split into training and test subsets (70:30 ratio in general).
The training subset is used to build the POD basis $\mathbf{\Phi}_i(\mathbf{x})$, used to project the test snapshots:

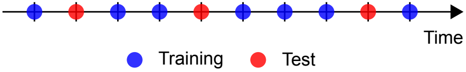
$$\mathcal{P}(\mathbf{u}_j(\mathbf{x})) = \mathbf{\Phi}_0(\mathbf{x}) + \sum_{i=1}^{N} \langle \mathbf{u}_j(\mathbf{x}) - \mathbf{\Phi}_0(\mathbf{x}), \mathbf{\Phi}_i(\mathbf{x}) \rangle_\Omega \, \mathbf{\Phi}_i(\mathbf{x}).$$

Projection error:

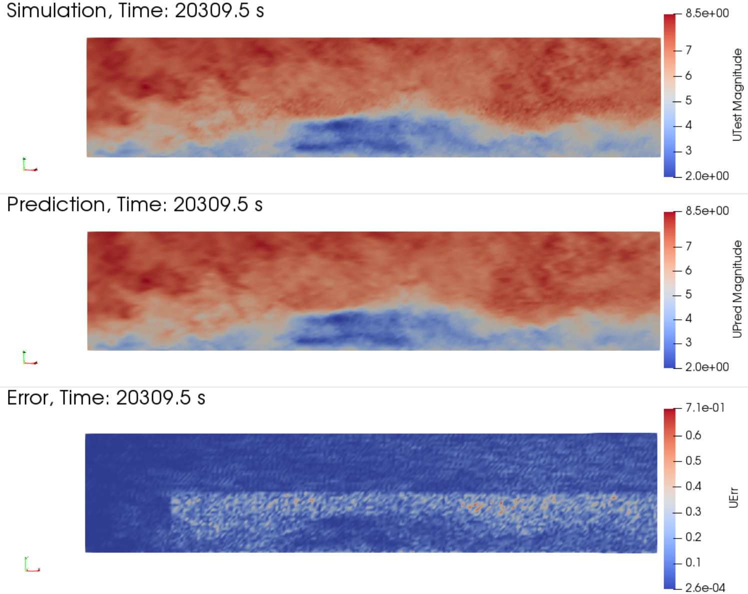
$$e_j(\mathbf{x}) = ||\mathbf{u}_j(\mathbf{x})| - |\mathcal{P}(\mathbf{u}_j(\mathbf{x}))|| \, .$$

Random selection from SOWFA simulation dataset to create training and test subsets:

- 450 snapshots in the time window [20150.2, 20374.95] s selected to build the basis, and
- the remaining 193 snapshots selected to test the prediction error using the basis.



In-sample prediction



Number of modes, N=80.

Prediction performed in parallel in time using `mpi4py`.

Instantaneous velocity magnitude plots showcase the ability of the POD-basis to provide fair in-sample prediction.

Outline

1. Flow simulation
2. Post-processing
3. In-sample prediction
4. Conclusion

Conclusion

- MCIA's *Curta* cluster provides the necessary resources to perform scientific research.
- Parallelized implementation provides significant reduction in the code runtimes - $\sim 6\times$ faster POD calculation, $\sim 15\times$ faster file I/O.
- Remote visualization tools facilitate plotting of results.
- *iRODS* offers a resilient archive storage of the simulation results.

Thank you for your attention.

Special acknowledgements to:
Michel BERGMANN (INRIA), Frédéric BLONDEL (IFPEN), Adrià BORRAS-NADAL (IFPEN), Guillaume ENCHERY (IFPEN), Laurent FACQ (IMB), Cédric GAZOPPI (IFPEN), Angelo IOLLO (IMB), Angela SCARDIGLI (Optimad), and the HPC support team!