AEROELASTIC MODELLING OF LARGE TRANSFORMATIONS USING PARTITIONED COUPLING: APPLICATION TO LARGE WIND TURBINES.

Caroline Le Guern

6th of December 2024

THESIS DIRECTOR:	MICHEL BERGMANN	**REVIEWERS:**	JORIS DEGROOTE
			NIELS NØRMARK SØRENSEN
SUPERVISORS:	FRÉDÉRIC BLONDEL	**EXAMINERS :**	AXELLE VIRÉ
	JEAN-LOU PFISTER		MATTEO CAPALDO
	NICOLAS GUY		ANGELO IOLLO

CURRENT DESIGN TOOL: DEEPLINES WIND™

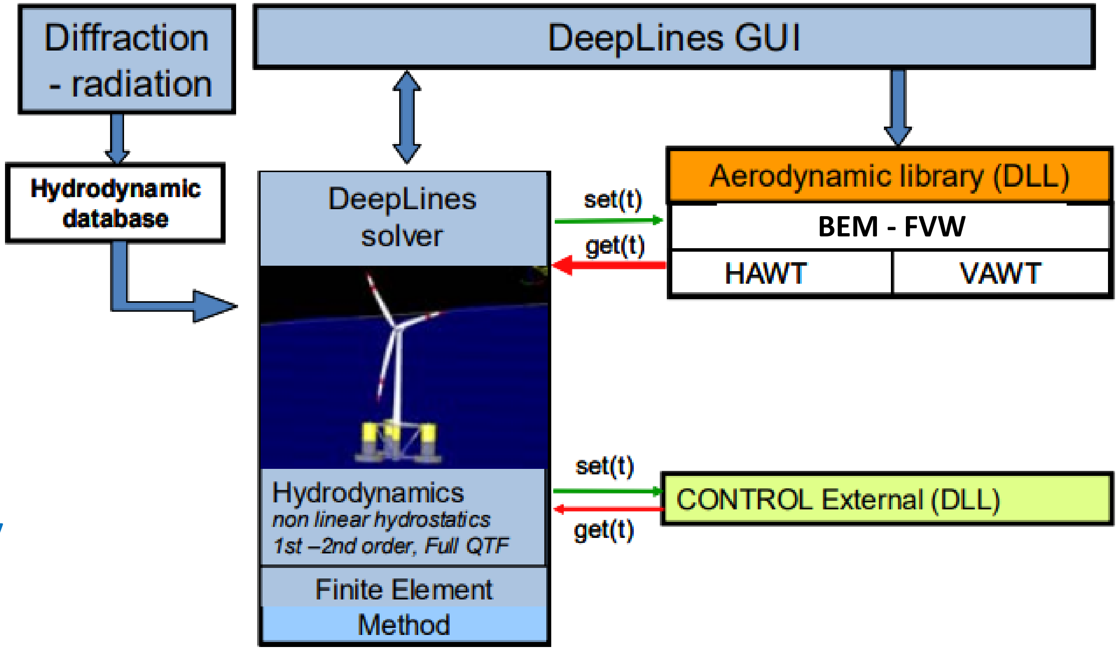
Complexity of wind power modelling -> Offshore wind turbines involve multi-physics interaction

- Aerodynamics
- Structural dynamics
- Hydrodynamics
- Control systems

Focus on **aero-elasticity** in this work.

Current design tool at IFPEN -> DeepLines Wind™ [2]

- Developed by Principia
- Purpose: multi-physics simulations for wind turbines
- Capabilities: aero-hydro-servo-elastic modelling with various aerodynamic methods of differing fidelity
- Limitations: high computational costs



[2] *C. Le Cunff, J.-M. Heurtier, L. Piriou, C. Berhault, T. Perdrizet, D. Teixeira, G. Ferrer, and J.-C. Gilloteaux, "Fully coupled floating wind turbine simulator based on nonlinear finite element method: Part i—methodology," in International Conference on Offshore Mechanics and Arctic Engineering, vol. 55423, p. V008T09A050, American Society of Mechanical Engineers, 2013*

THESIS OBJECTIVES AND CHALLENGES

Objectives :

Aeroelastic modelling of large wind turbine under operational conditions:

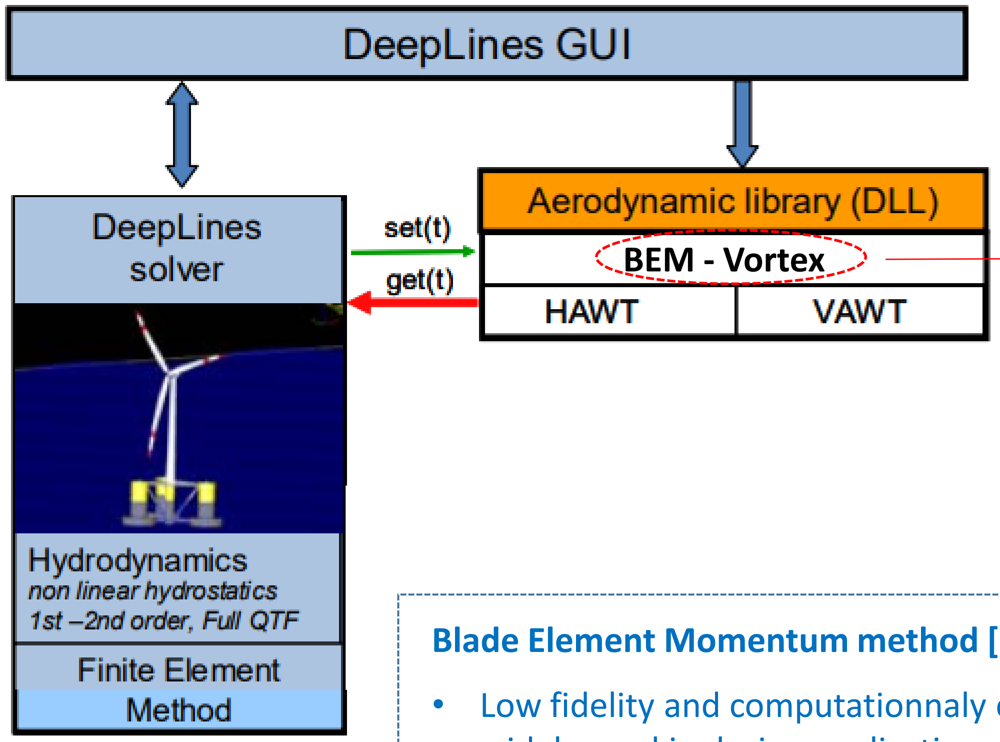
- Transition from low-fidelity Blade Element Momentum (BEM) to higher-fidelity Free Vortex Wake (FVW) methods.
- Develop alternative coupling techniques for aeroelastic computations.
- Implement partitioned coupling in the DeepLines Wind framework.
- Reduce computational costs of aeroelastic modelling with FVW methods

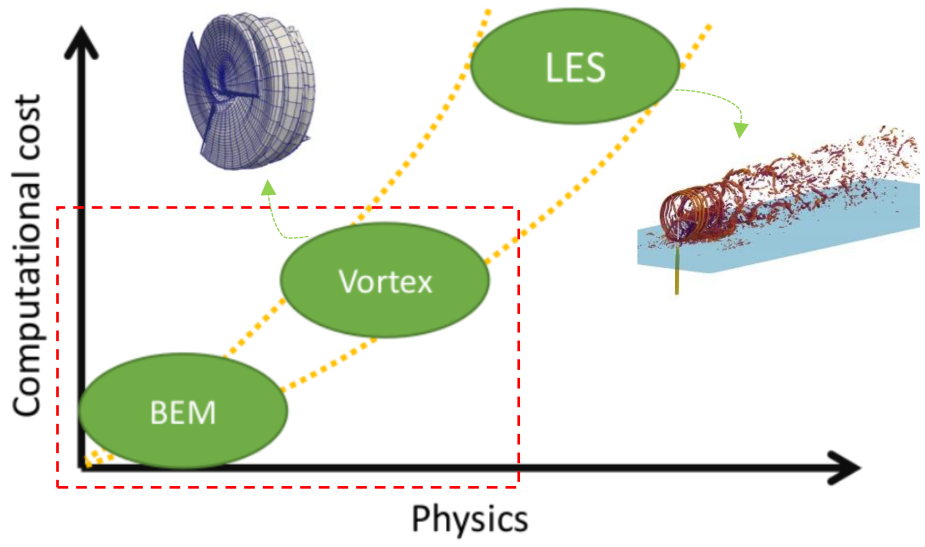
Main challenges:

- Time-scale difference: Fluid and structural problems may require different resolution orders.
- Over-resolved aerodynamics: FVW methods significantly increase computational coss.
- Coupling techniques: Can alternative methods reduce computational cost while maintaining numerical stability and accuracy?

PRESENTATION OUTLINE

AERODYNAMIC MODELLING TECHNIQUES FOR WIND TURBINE SIMULATION





Blade Element Momentum method [3]:

- Low fidelity and computationnaly efficient: widely used in design applications.
- Assumes axisymmetric, steady inflow.
- Relies on empirical corrections: tip-loss, dynamic inflow.
- Limited accuracy for unsteady or non-uniform inflow conditions

Free Vortex Wake methods [4]: → **Focus of this part**

- Intermediate-fidelity, physics-based approach.
- Models wake as discrete vortical structures evolving over time.
- Captures unsteady effects, wake interactions and non-uniform inflow.
- More computationally intensive, especially for aeroelastic simulations.

[3] *T. Burton, N. Jenkins, D. Sharpe, and E. Bossanyi, Wind energy handbook. John Wiley & Sons, 2011.*
[4] *E. Branlard, "Wind turbine aerodynamics and vorticity-based methods," 2020.*

PITCHOU AND CASTOR: TWO GPU ACCELERATED FVW CODES

Feature	Pitchou	CASTOR [5]
Language	Python	C++
Development Context	Developed during this thesis as a training tool	Pre-existing at IFPEN
Acceleration	GPU-accelerated	GPU-accelerated
Primary Application	Simplified studies and testing	More complex aerodynamic and aeroelastic simulations
Integration	Standalone testing framework for aerodynamic simulations	Integrated with DeepLines Wind™
Specifications	Filament wake discretisation	Filament discretisation + merging methods

→ **Focus on the underlying theory and development process**

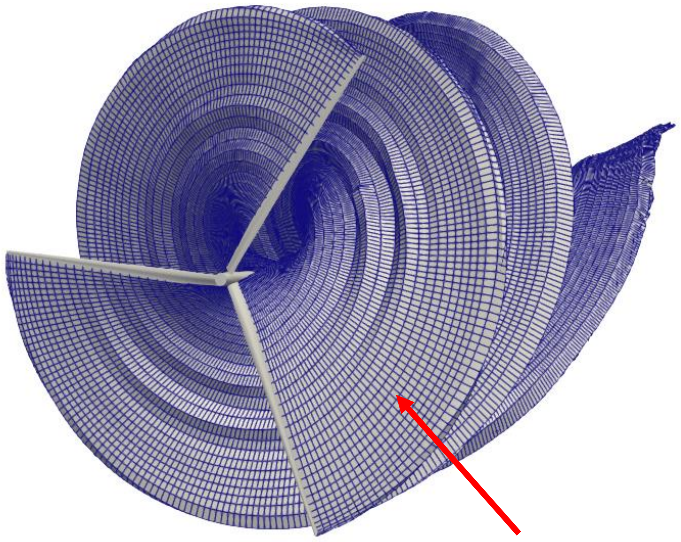
[5] *F. Blondel, P.-A. Joulin, and C. Le Guern, "Towards vortex-based wind turbine design using gpus and wake accommodation," in Journal of Physics: Conference Series, vol. 2767, p. 052016, IOP Publishing, 2024.*

Navier Stokes equation in $(\mathbf{u} - \boldsymbol{\omega})$ formulation

$$\vec{\nabla} \cdot \vec{u} = 0$$

$$\frac{\partial \vec{\omega}}{\partial t} + \underbrace{\left(\vec{u} \cdot \vec{\nabla}\right) \vec{\omega} = \left(\vec{\omega} \cdot \vec{\nabla}\right) \vec{u}}_{\text{Convection}} + \underbrace{\nu \Delta \vec{\omega}}_{\text{Diffusion}}$$

Convection **Diffusion**



Lagrangian framework formulation for an inviscid flow:

$$\frac{D\vec{\omega}}{Dt} = \left(\vec{\omega} \cdot \vec{\nabla}\right) \vec{u}$$

Biot-Savart law [6] -> compute the vorticity induced velocity:

$$\vec{u}^{\Psi}(\vec{x}, t) = \iiint_{\vec{y} \in \mathbb{R}^3} \vec{K}(\vec{x} - \vec{y}) \times \vec{\omega}(\vec{y}, t) \, \mathrm{d}v(\vec{y})$$

Volume of integration

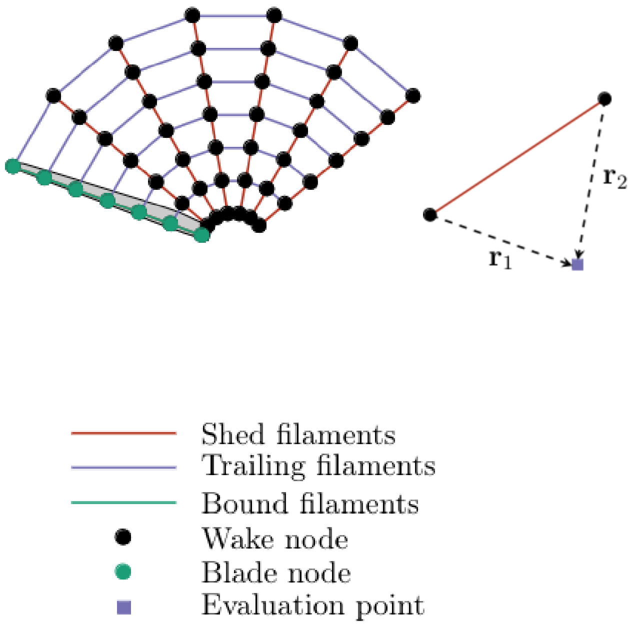
Biot-Savart kernel:

$$\vec{K}(\vec{x}) = \vec{\nabla} G(\vec{x}) = -\frac{1}{4\pi} \frac{\vec{x}}{\|\vec{x}\|^3}$$

[6] *E. S. P. Branlard, “Flexible multibody dynamics using joint coordinates and the rayleigh-ritz approximation: The general framework behind and beyond ex,” Wind Energy, vol. 22, no. 7, pp. 877–893, 2019.*

FREE VORTEX WAKE METHODS: FILAMENT DISCRETISATION AND LIFTING LINE METHOD

Overall discretisation



Wake discretisation

$$\frac{\partial \vec{\omega}}{\partial t} + \left(\vec{u} \cdot \vec{\nabla}\right) \vec{\omega} = \left(\vec{\omega} \cdot \vec{\nabla}\right) \vec{u}$$

Stretching term: numerically challenging

Filament based approach: solving Kelvin's circulation theorem

$$\frac{D\Gamma}{Dt} = 0$$

Biot-Savart law using circulation

$$\vec{u}_{induced}(\vec{x}_p) = \frac{\Gamma}{4\pi} \frac{(|\vec{r_1}| + |\vec{r_2}|)(\vec{r_1} \times \vec{r_2})}{|\vec{r_1}||\vec{r_2}|(|\vec{r_1}||\vec{r_2}| + \vec{r_1} \cdot \vec{r_2})}$$

Trail and shed filaments defined by circulation via Kelvin's theorem

$$\Gamma_{trail}(r,t) = \Gamma_{bound}(r,t) - \Gamma_{bound}(r - \Delta r, t)$$

$$\Gamma_{shed}(r,t) = \Gamma_{bound}(r,t) - \Gamma_{bound}(r, t - \Delta t)$$

Blade discretisation and litfting line [7]

Source of bound circulation generating lift

Kutta-Joukowski

$$L = \rho |\vec{u}_{eff}| \Gamma_{bound}$$

Blade-element theory

$$L = \frac{1}{2} \rho |\vec{u}_{eff}|^2 c C_L$$

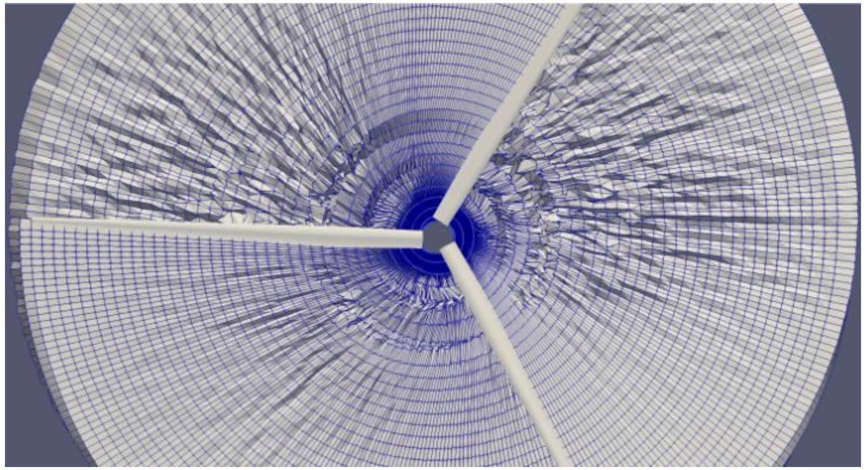
$$\Gamma_{bound} = \frac{1}{2} |\vec{u}_{eff}| c C_l$$

$$\vec{u}_{eff} = \vec{u}_{wind} + \vec{u}_{geom} + \vec{u}_{induced}$$

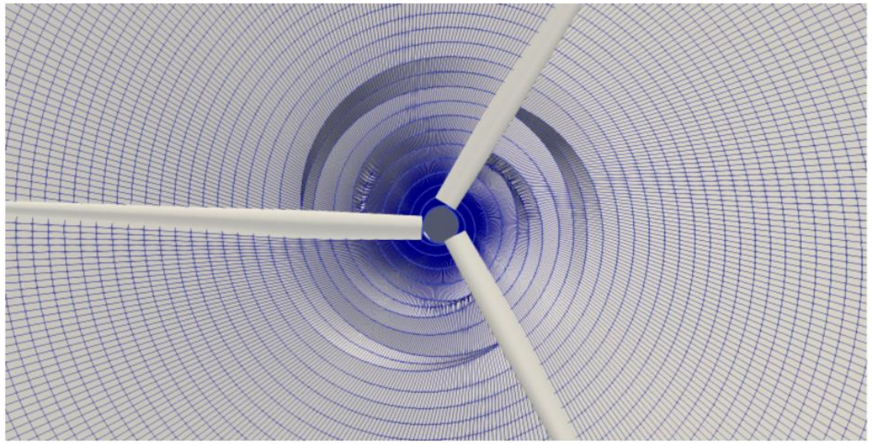
[7] A. van Garrel, "Development of a wind turbine aerodynamics simulation module," 2003.

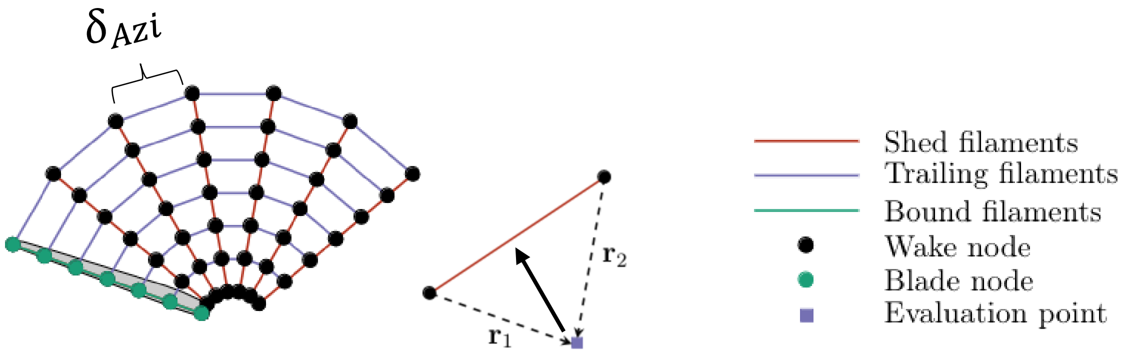
CHALLENGES IN FVW METHODS: WAKE DISCRETIZATION AND DESINGULARIZATION METHODS

Offset method



Vatistas method





$$\vec{u}_{induced}(\vec{x}_p) = \frac{\Gamma}{4\pi}\frac{(|\vec{r_1}|+|\vec{r_2}|)(\vec{r_1}\times\vec{r_2})}{|\vec{r_1}||\vec{r_2}|(|\vec{r_1}||\vec{r_2}|+\vec{r_1}\cdot\vec{r_2})}$$

Desingularisation methods [8]

Offset method

$$\vec{u}_{induced}(\vec{x}_p) = \frac{\Gamma}{4\pi}\frac{(|\vec{r_1}|+|\vec{r_2}|)(\vec{r_1}\times\vec{r_2})}{|\vec{r_1}||\vec{r_2}|(|\vec{r_1}||\vec{r_2}|+\vec{r_1}\cdot\vec{r_2})+(r_c l_0)^2}$$

Vatistas method

$$\vec{u}_{induced}(\vec{x}_p) = K(\rho)\frac{\Gamma}{4\pi}\frac{(|\vec{r_1}|+|\vec{r_2}|)(\vec{r_1}\times\vec{r_2})}{|\vec{r_1}||\vec{r_2}|(|\vec{r_1}||\vec{r_2}|+\vec{r_1}\cdot\vec{r_2})}$$

$$K(\rho) = \frac{\frac{\rho^2}{r_c^2}}{\sqrt{1+\frac{\rho^4}{r_c^4}}}$$

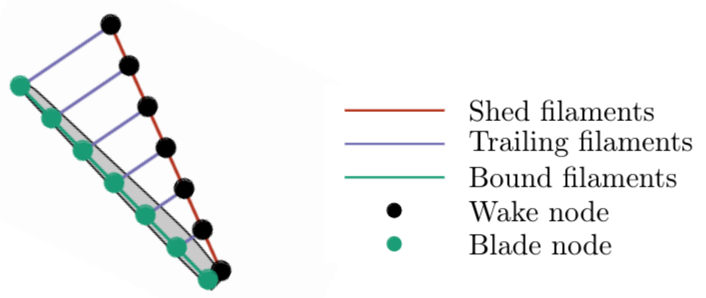
[8] G. H. Vatistas, V. Kozel, and W. Mih, "A simpler model for concentrated vortices," Experiments in Fluids, vol. 11, pp. 73–76, 1991.

FVW CODE – ALGORITHM OVERVIEW

Algorithm 1: Vortex method algorithm overview

Data: Turbine geometry and kinematic: blades, tower, struts element data
Result: Turbine loads and wake geometry

```
Add a new row of shed and trail filaments in the wake
Copy filaments' information on GPU, clear GPU memory
Compute filament inductions on wake and blade nodes
    Copy wake and blade nodes information on GPU
    Apply Biot-Savart and perform the reduction at every wake node on
  GPU
    Copy the induced velocities back to CPU
while |Γ̃_b^new − Γ̃_b^old| > ϵ do
    Compute new induced velocities on the lifting line on CPU using
      updated Γ̃_b^new and newly created Γ_s and Γ_t
    Compute Γ_b and apply relaxation to obtain Γ̃_b^new
    Update newly created Γ_s and Γ_t
end
Advect all wake nodes and apply any stretching corrections on CPU
Apply wake accommodation techniques on CPU
```



$$\vec{u}_{induced}(\vec{x}_p) = \frac{\Gamma}{4\pi} \frac{(|\vec{r_1}| + |\vec{r_2}|)(\vec{r_1} \times \vec{r_2})}{|\vec{r_1}||\vec{r_2}|(|\vec{r_1}||\vec{r_2}| + \vec{r_1} \cdot \vec{r_2})}$$

$$\Gamma_{bound} = \frac{1}{2}|\vec{u}_{eff}|cC_l$$

$$\Gamma_{trail}(r,t) = \Gamma_{bound}(r,t) - \Gamma_{bound}(r-\Delta r,t)$$

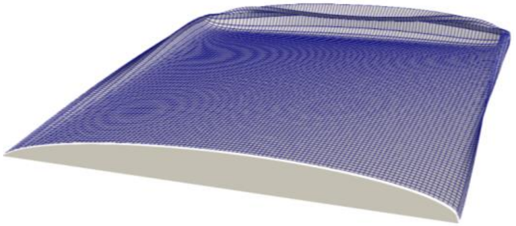
$$\Gamma_{shed}(r,t) = \Gamma_{bound}(r,t) - \Gamma_{bound}(r,t-\Delta t)$$

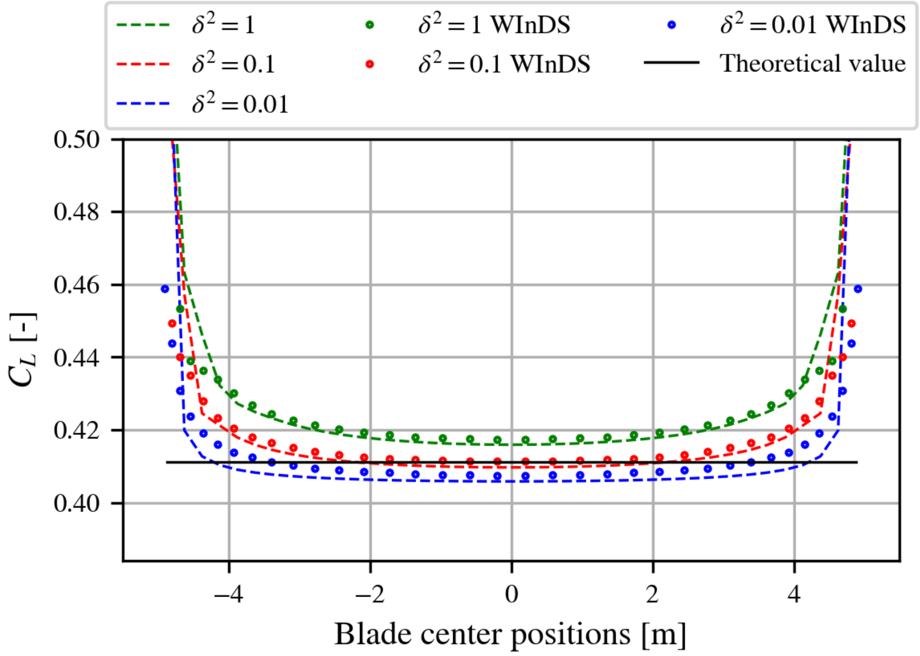
$$\vec{x}(t+\Delta t) = \vec{x}(t) + \vec{u}_{eff}(t)\Delta t$$

PITCHOU VALIDATION: ELLIPTICAL WING CASE – COMPARISON TO WINDS FVW CODE

Static elliptical wing

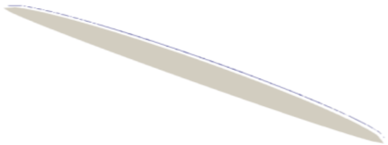
- 40 spanwise nodes
- Blade pitch = 5°
- AR = 6
- $U_\infty = 1\ m/s$
- Total time = 10s
- $\Delta t = 0.1$s
- Offset method ($r_c^2 = \delta^2$)

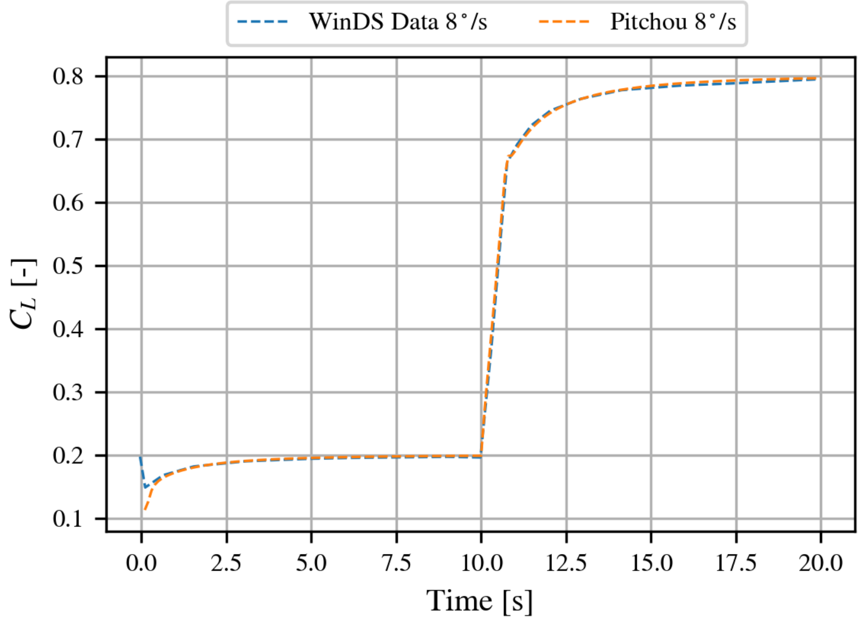




Elliptical wing subject to pitch change

- AR = 18
- Initial blade pitch = 2°
- Final blade pitch = 8°
- Pitch rate= 8°/s
- Total time = 20s
- Pitching start time = 10s
- Offset method : $\delta^2 = 0.1$

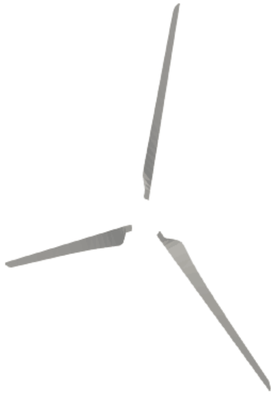


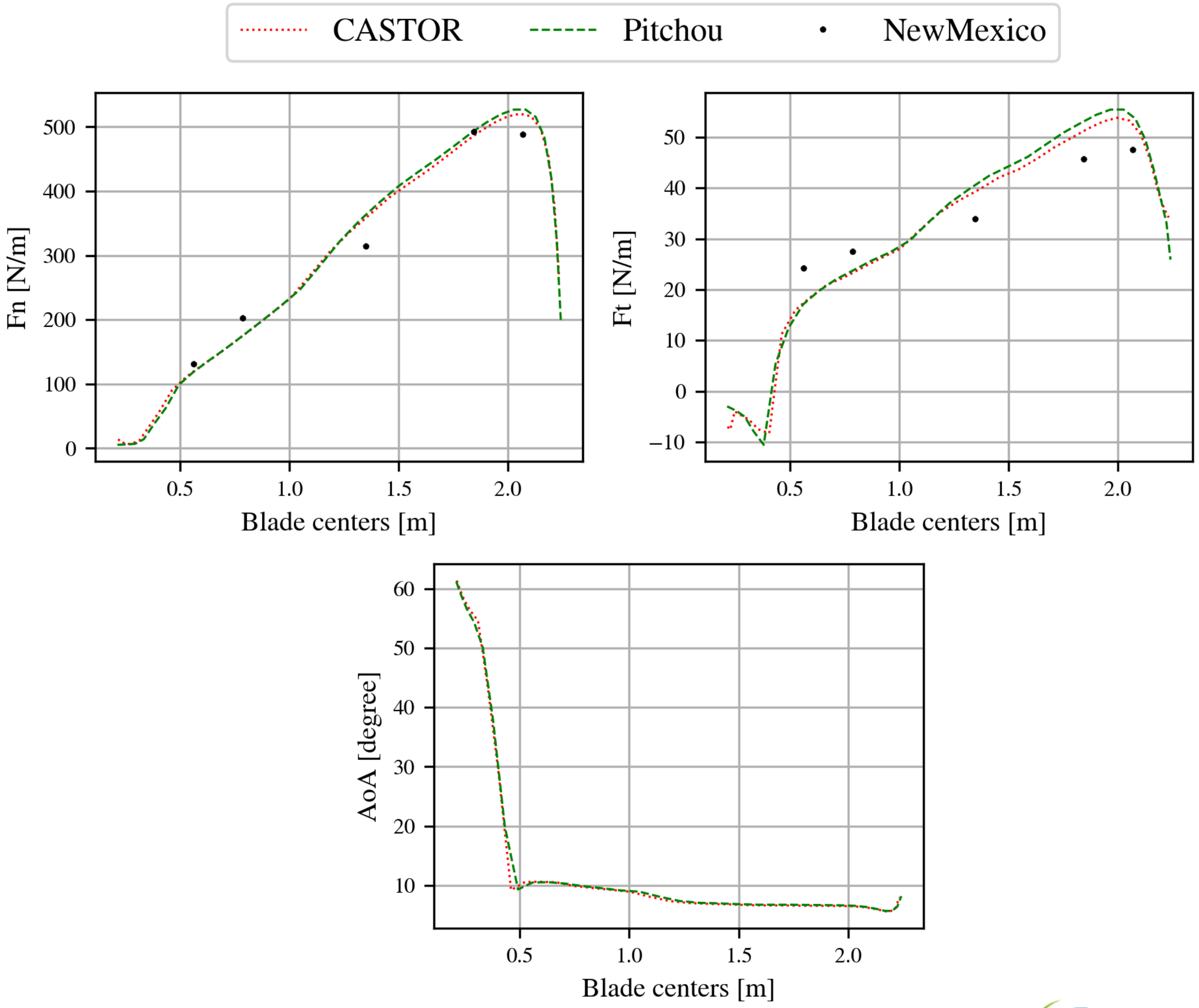


PITCHOU VALIDATION: NEW MEXICO TURBINE – COMPARISON TO CASTOR CODE

NewMexico [9] wind turbine:

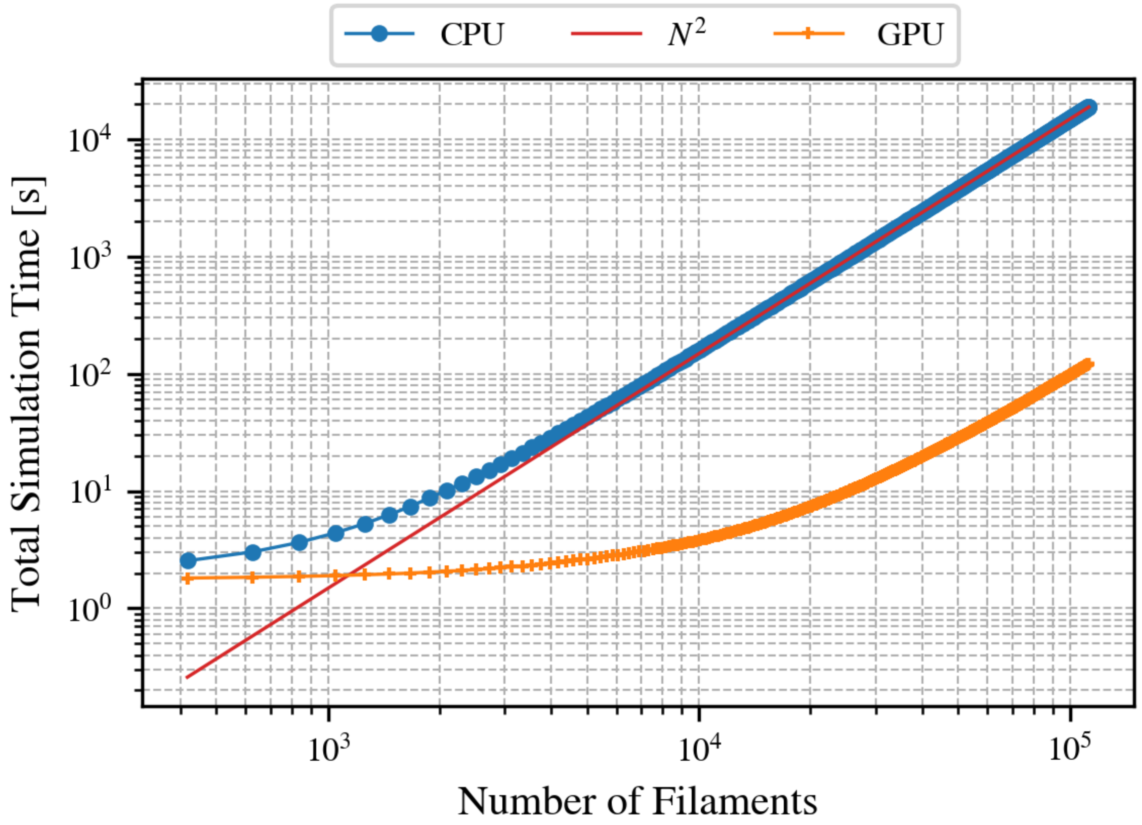
- $U_\infty = 15.06$ m/s
- Azimuthal time step: 10°
- Total rotations: 15
- Offset method: $\delta^2 = 0.01$





[9] *H. Snel, J. Schepers, and B. Montgomerie, "The mexico project (model experiments in controlled conditions): The database and first results of data processing and interpretation," in Journal of Physics: Conference Series, vol. 75, p. 012014, IOP Publishing, 2007.*

PITCHOU VALIDATION: EFFECTS OF GPU ACCELERATION

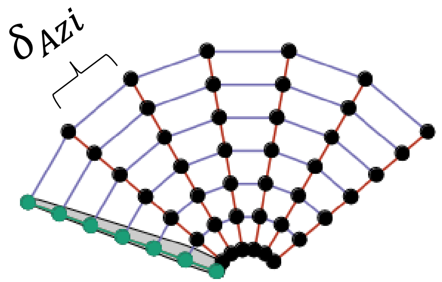


Static elliptical wing -> wake length equivalent to that required for convergeed induced velocity in wind turbine simulations.

- 105 spanwise nodes
- Blade pitch = 5°
- AR = 6
- $U_\infty = 1\ m/s$
- Total time = 54s
- $\Delta t = 0.1s$

- The total simulation time **scales quadratically** with the number of filaments.
- Achieves **a speedup of two orders** of magnitude with GPU acceleration compared to CPU simulations.

SENSITIVITY ANALYSIS USING CASTOR: EFFECTS OF AZIMUTHAL WAKE DISCRETISATION AND DESINGULARISATION METHODS

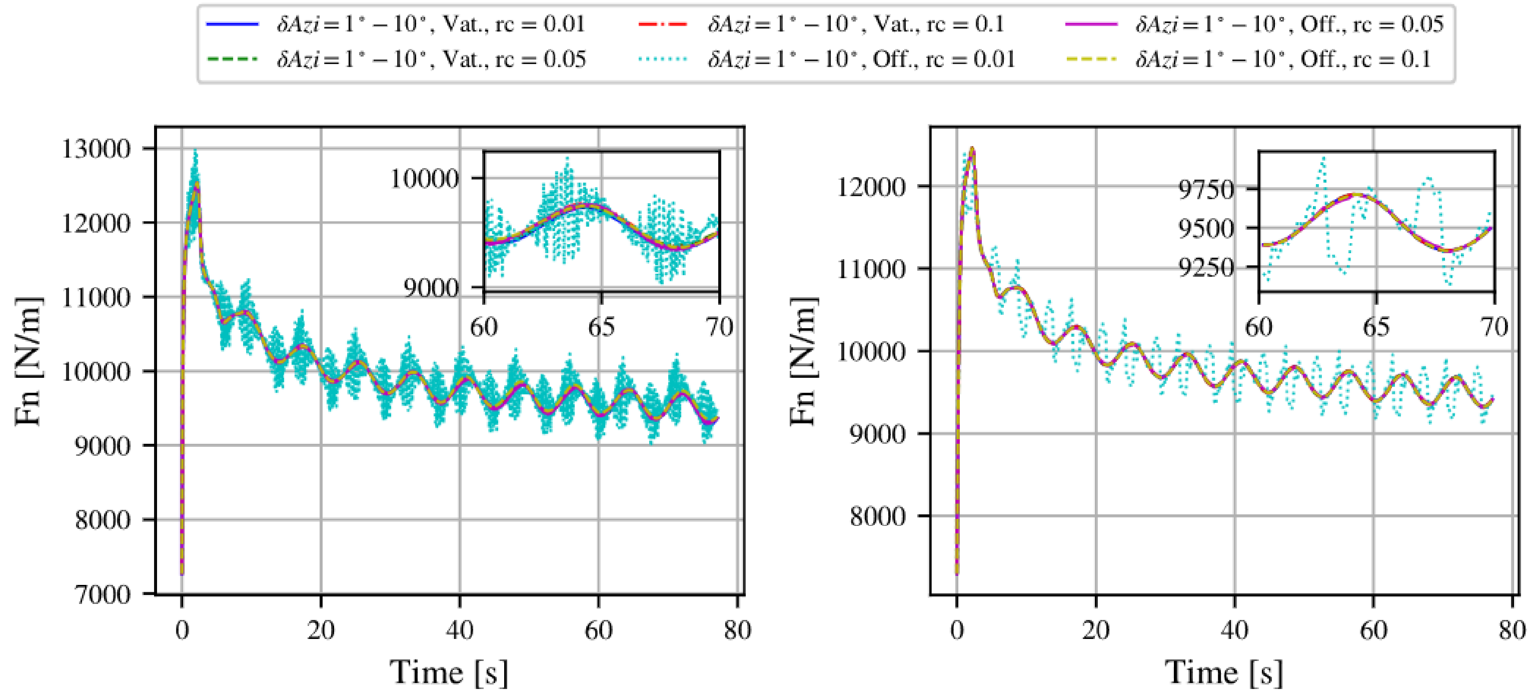


Offset method

$$\vec{u}_{induced}(\vec{x}_p) = \frac{\Gamma}{4\pi}\frac{(|\vec{r_1}|+|\vec{r_2}|)(\vec{r_1}\times\vec{r_2})}{|\vec{r_1}||\vec{r_2}|(|\vec{r_1}||\vec{r_2}|+\vec{r_1}\cdot\vec{r_2})+(r_c l_0)^2}$$

Vatistas method

$$\vec{u}_{induced}(\vec{x}_p) = K(\rho)\frac{\Gamma}{4\pi}\frac{(|\vec{r_1}|+|\vec{r_2}|)(\vec{r_1}\times\vec{r_2})}{|\vec{r_1}||\vec{r_2}|(|\vec{r_1}||\vec{r_2}|+\vec{r_1}\cdot\vec{r_2})}$$



$\delta_{Azi} = 1°$ $\delta_{Azi} = 10°$

Azimuthal discretization in litterature:

- Recommended range: 5° to 10° per time step [10]

Finer wake discretization:

- Combined with small core radii
- Results in non-smooth time evolution of aerodynamic forces

Impact on aeroelastic simulations:

- Potentially detrimental due to force irregularities

[10] R. Corniglion, aero-elastic modeling of floating wind turbines with vortex methods. PhD thesis, Marne-la-vallée, ENPC, 2022.

KEY TAKEAWAYS AND UPCOMING FOCUS

Takeaways:

- FVW methods:
  - Higher fidelity than BEM, captures unsteady effects (**wake interactions, inflow**).
  - Potential as design tool for offshore wind turbines.
- Pitchou FVW code:
  - **Python-based**, designed for testing (single blades/full rotors).
  - Quadratic scaling with filament count.
  - **GPU:** ~100x faster than CPU.
- Sensitivity of FVW methods:
  - Key parameters: **wake discretization, desingularization**.
  - Poor parameter choices -> non-smooth forces -> aeroelastic challenges.

Upcoming focus:

- Adressing time scale differences (structural vs. aerodynamic solvers).
- Exploring alternative coupling techniques: **simplified linear model**.

PRESENTATION OUTLINE

1. Aerodynamic simulation methods

2. Aeroelastic modeling

3. Partitioned coupling in wind turbine aeroelastic problem

4. Conclusion and perspectives

Current aeroelastic partitioned coupling technique in DeepLines Wind™

Current numerical tool

Structural solver

DeepLines
Finite Element method

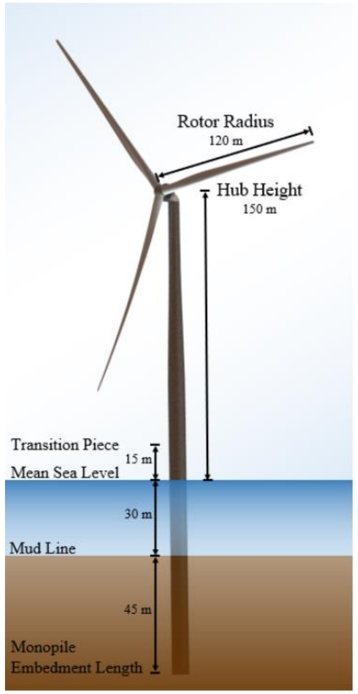
Fluid/structure coupling : Conventional serial staggered partitioned scheme

Aerodynamic Solvers

AeroDeep
BEM Method

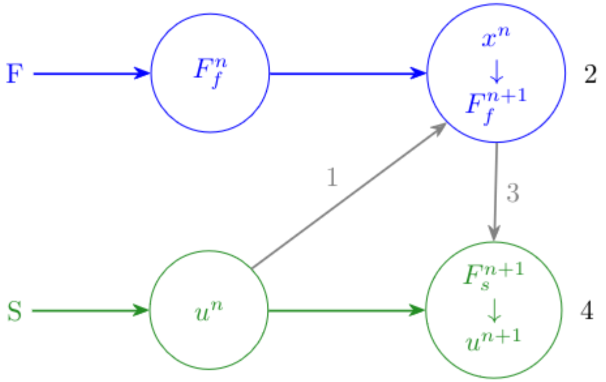
CASTOR
Free Vortex Wake Method

IEA 15MW



First blade mode's frequency

- ~ 0.5 Hz → T=2s
- $\Delta t_s \approx 0.02s$

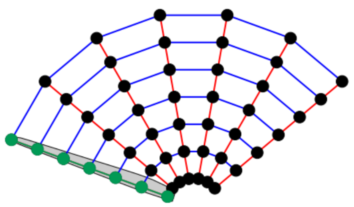


Constant wind simulation U=10 m/s:

- Steady-state rotation speed: 45deg/s
- For a coupling time step Δt_c= 0.01s

→ Δt_f = 0.45 deg per time step

Wake using FVW method

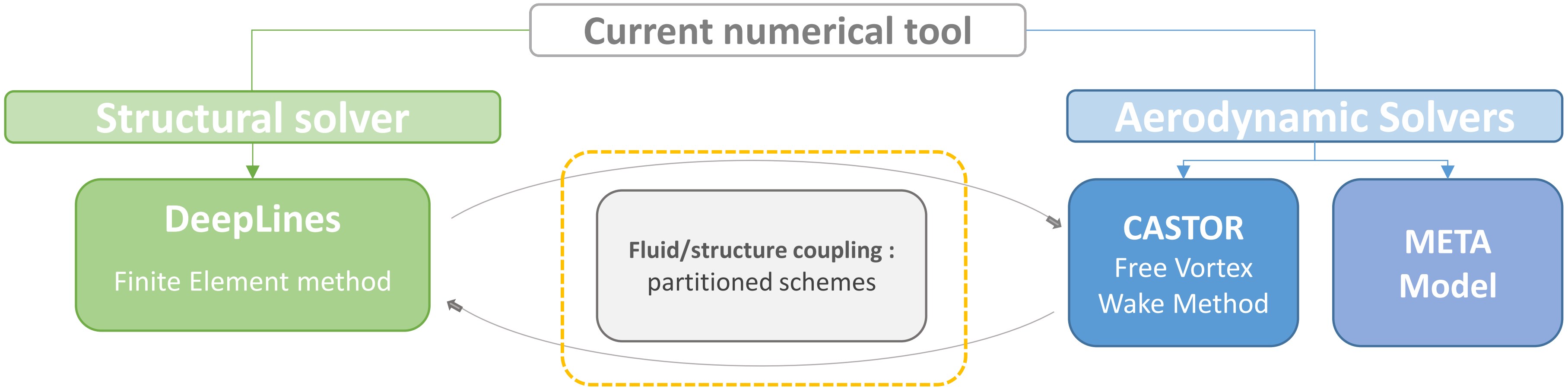


	Fluid	Structure
Time step order (s)	10^{-1}	10^{-2}

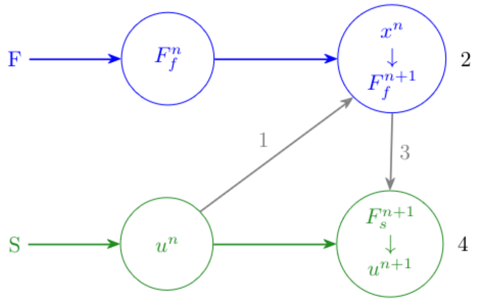
→ **Over-resolved aerodynamic problem**

Inria IFP Energies nouvelles

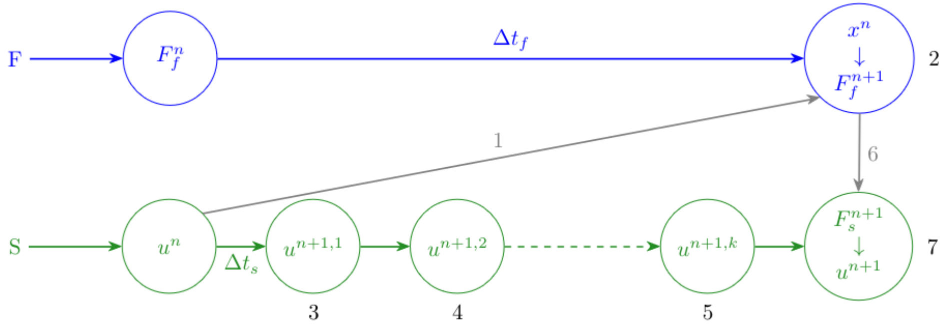
Coupling schemes for aeroelastic modelling



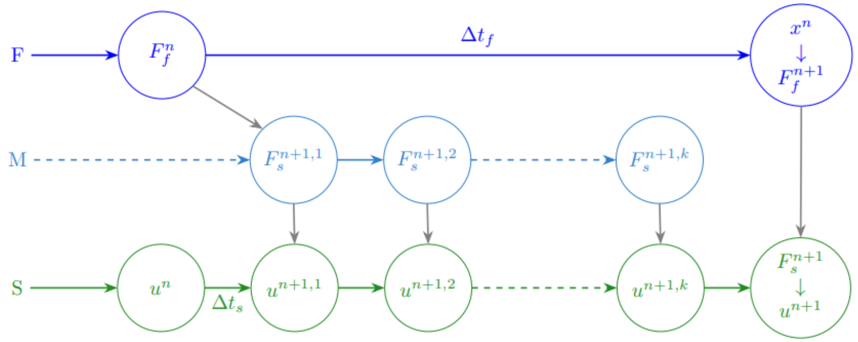
« Conventional Serial Staggered (CSS) »



Subcycling scheme [11]



Adapted subcycling



Aim: study the effects of partitioned schemes on numerical properties using a simplified linear coupled oscillator model first.

[11] S. Piperno, C. Farhat, and B. Larrouturou, "Partitioned procedures for the transient solution of coupled aroelastic problems part i: Model problem, theory and two-dimensional application," Computer methods in applied mechanics and engineering, vol. 124, no. 1-2, pp. 79–112, 1995.

TWO COUPLED LINEAR OSCILLATOR MODEL: PHYSICAL DESCRIPTION

Structural linear oscillator equation

$$\ddot{y} + d_s \dot{y} + \delta^2 y = Mq$$

- y structural displacement
- d_s structural damping term
- δ oscillation frequency
- M added coupling stiffness

Fluid Van der Pol linear equation

$$\ddot{q} + d_a \dot{q} + q = A_0 y + A_1 \dot{y}$$

- q fluid force
- d_a fluid damping term
- A_0 added coupling stiffness
- A_1 added coupling damping

Van der Pol equation used for VIV studies [12]: an aeroelastic instability modeled in large wind turbines.

Coupled matrix compact form

$$\begin{bmatrix} \dot{Q} \\ \delta\dot{W} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} Q \\ \delta W \end{bmatrix} \quad \delta W = \begin{bmatrix} X_f \\ \dot{X}_f \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \end{bmatrix} \quad Q = \begin{bmatrix} X_s \\ \dot{X}_s \end{bmatrix} = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}$$

[12] E. De Langre, “Frequency lock-in is caused by coupled-mode flutter,” Journal of fluids and structures, vol. 22, no. 6-7, pp. 783–791, 2006.

TWO COUPLED LINEAR OSCILLATOR MODEL: NUMERICAL PROPERTIES

Check physical stability

$$\begin{bmatrix} \dot{X}_s \\ \dot{X}_f \end{bmatrix} = \begin{bmatrix} A_{ss} & A_{sf} \\ B_{fs} & B_{ff} \end{bmatrix} \begin{bmatrix} X_s \\ X_f \end{bmatrix}$$

Physical problem

- Structural oscillator equation: inject **Newmark** scheme.
- Fluid oscillator equation: inject **Forward Euler** scheme.
- Linear multi-step method [13]: separate states at n+1 from states at n.

Check numerical stability

$$\begin{bmatrix} X_s^{n+1} \\ X_f^{n+1} \end{bmatrix} = \begin{bmatrix} C_{ss} & C_{sf} \\ D_{fs} & D_{ff} \end{bmatrix} \begin{bmatrix} X_s^{n} \\ X_f^{n} \end{bmatrix}$$

Time discretised problem: CSS

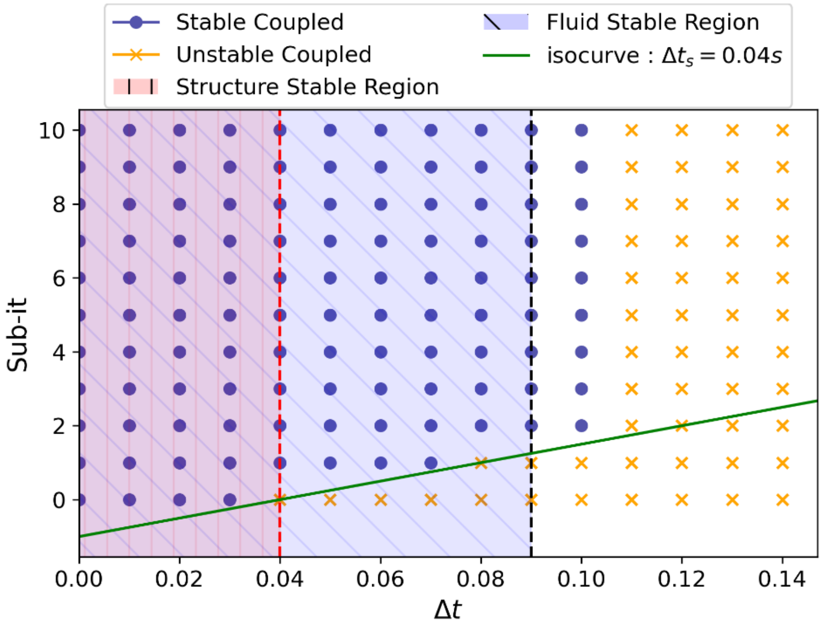
Method used for numerical stability study: for CSS and subcycling comparison.

[13] *J. C. Butcher, "General linear methods," Acta Numerica, vol. 15, pp. 157–256, 2006.*

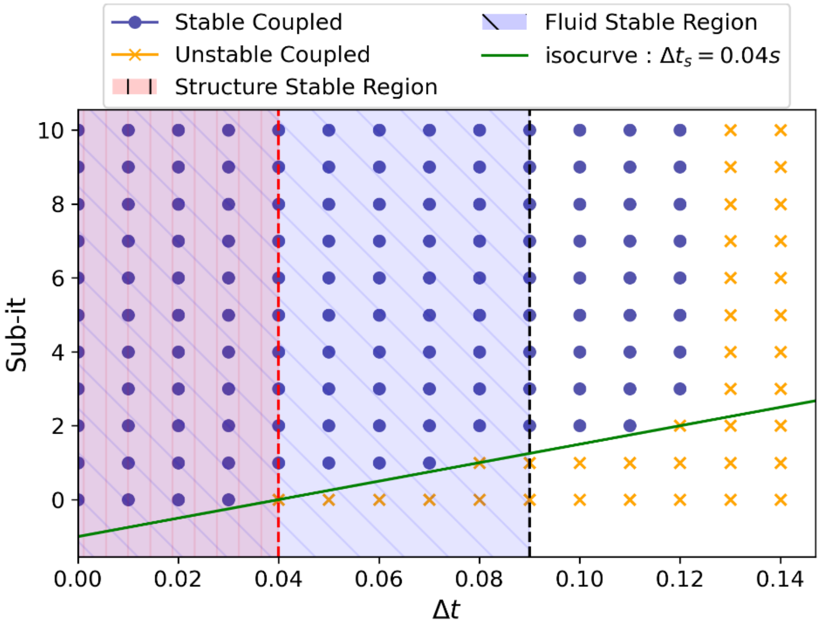
NUMERICAL STABILITY: COUPLING CASE WITH DAMPING EFFECTS

Subcycling using constant regression

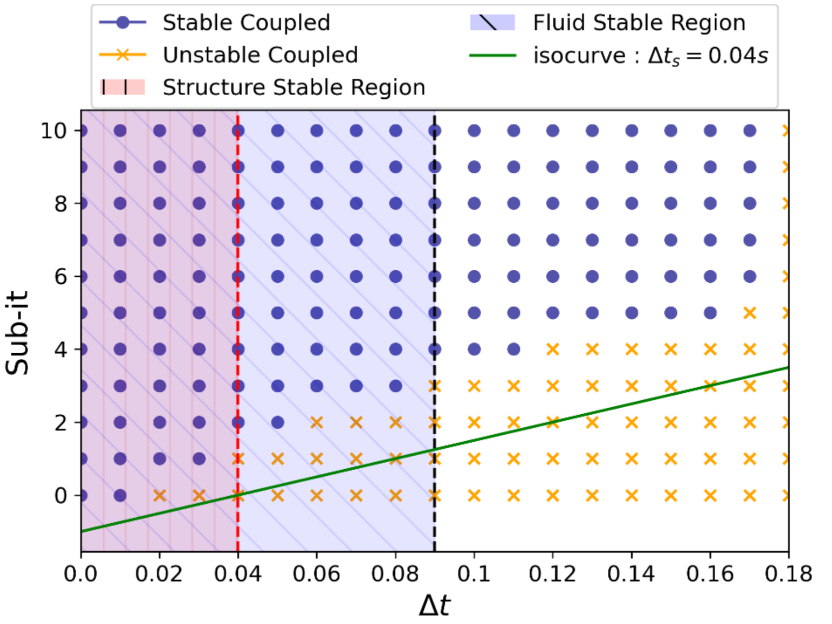
Weak coupling



Medium coupling

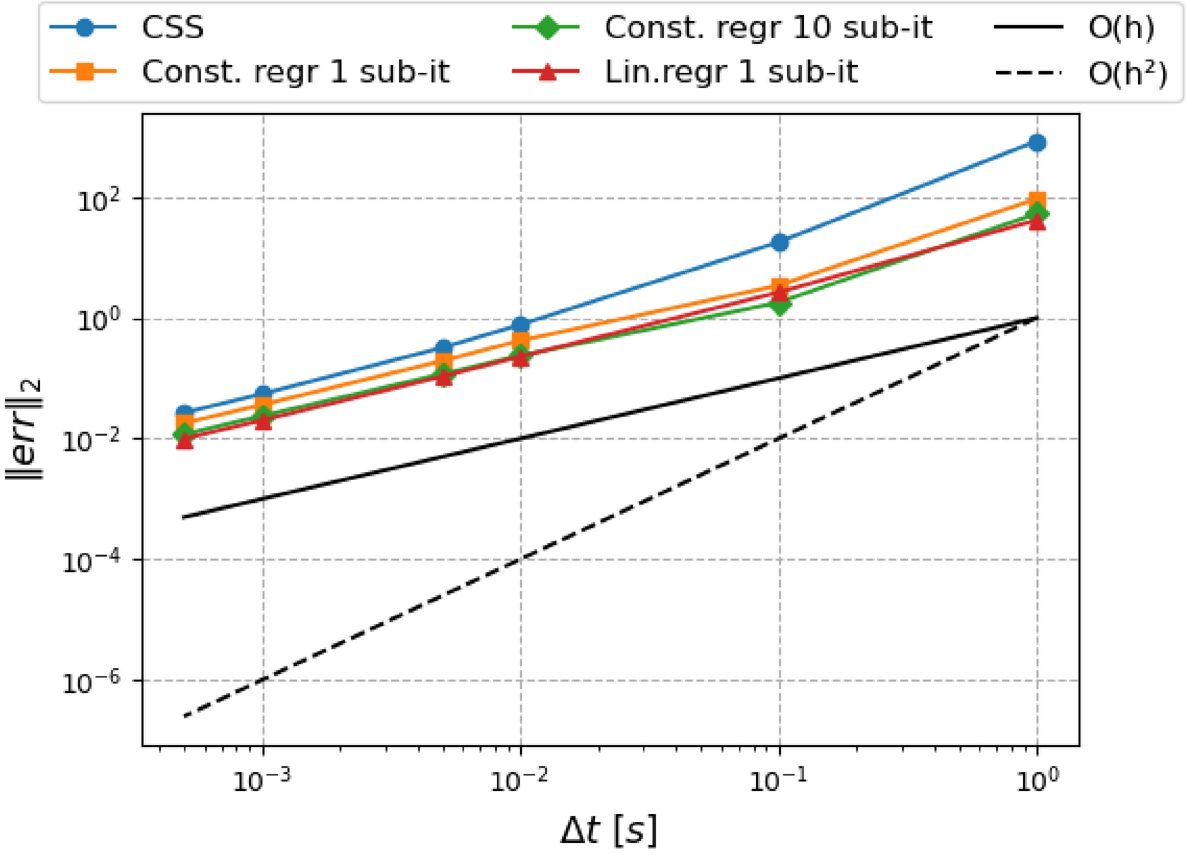


Strong coupling



Extended stability region when using subcycling compared to CSS.

ACCURACY ANALYSIS



Order of the coupling scheme:

- Error computation with respect to reference monolithic solution.
- The CSS coupling scheme using FE/Newmark is of order 1.
- Subcycling doesn't affect the order.
- The error seems to be less using subcycling in comparison to CSS.

KEY TAKEAWAYS AND UPCOMING FOCUS

Takeaways

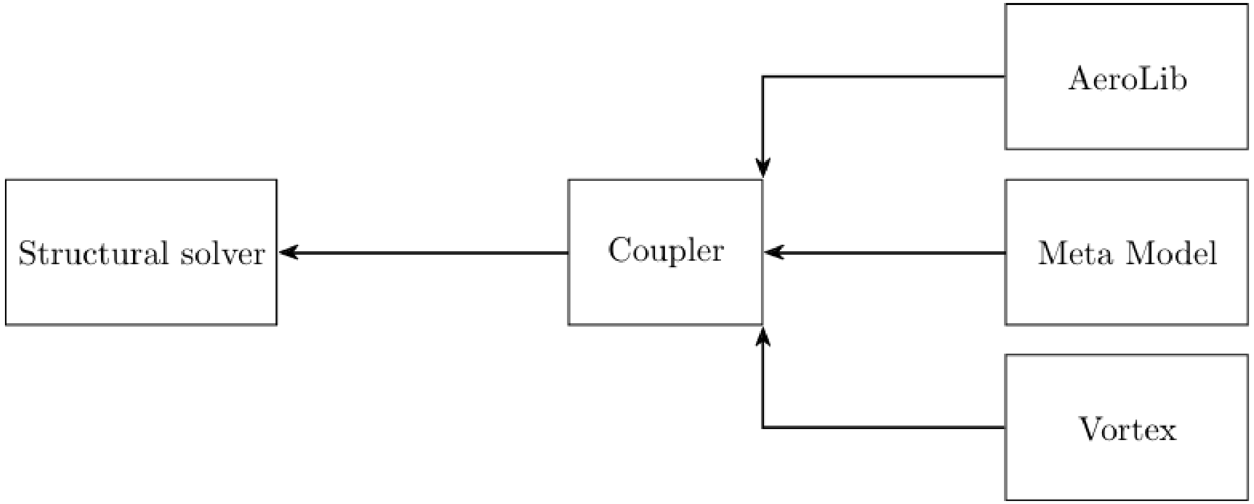
- Comparison of **CSS** and **Subcycling** schemes in a **linear coupled oscillator model:**
  - Insights into effects of partitioned schemes in coupled problems.
  - Subcycling **extends stability** in damping-dominated cases.
  - Order of **accuracy preserved** in subcycling.

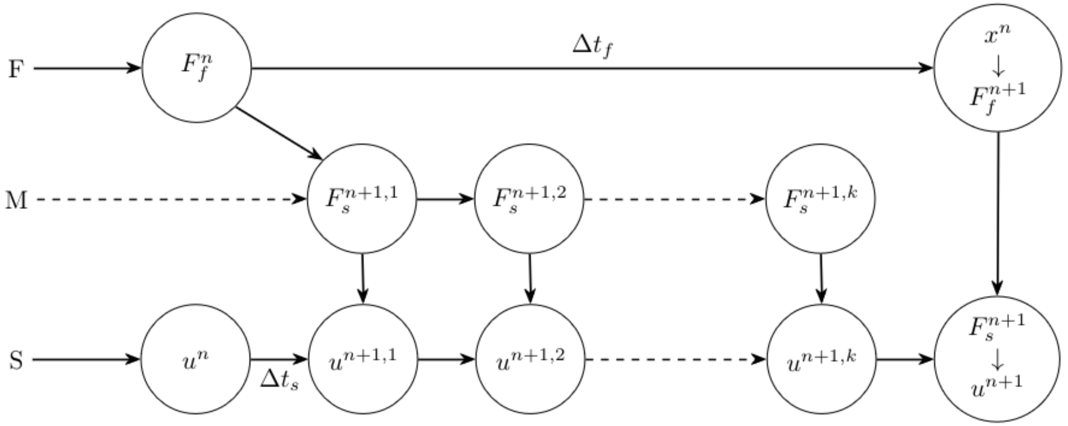
Upcoming focus

- Partitioned schemes for realistic **aeroelastic problems:**
  - Implementing **subcycling** in **DeepLines Wind™.**
  - Aeroelastic effects: **damping** vs natural **frequency.**
  - Comparing subcycling vs. CSS results.

PRESENTATION OUTLINE

IMPLEMENTATION OF PARTITIONED COUPLING IN DEEPLINES WIND FRAMEWORK





Architecture with coupler module that connects structural solver to two aerodynamic libraries:

- « High-fidelity solver » (e.g., CASTOR, AeroDeep): compute precise aerodynamic forces.
- « Low-fidelity meta model »: provide approximate aerodynamic forces for faster computations at intermediate time steps.

The coupling scheme uses two distinct time steps:

- Structural solver time step Δt_s : higher frequency updates for the structure using meta model outputs
- Fluid solver time step Δt_f : lower rate aerodynamic resolution to reduce computational costs.

EFFECTS OF PARTITIONED COUPLING SCHEMES ON WIND TURBINE PROBLEM

Objective: Assess subcycling scheme impact on aeroelastic simulations.

Setup:

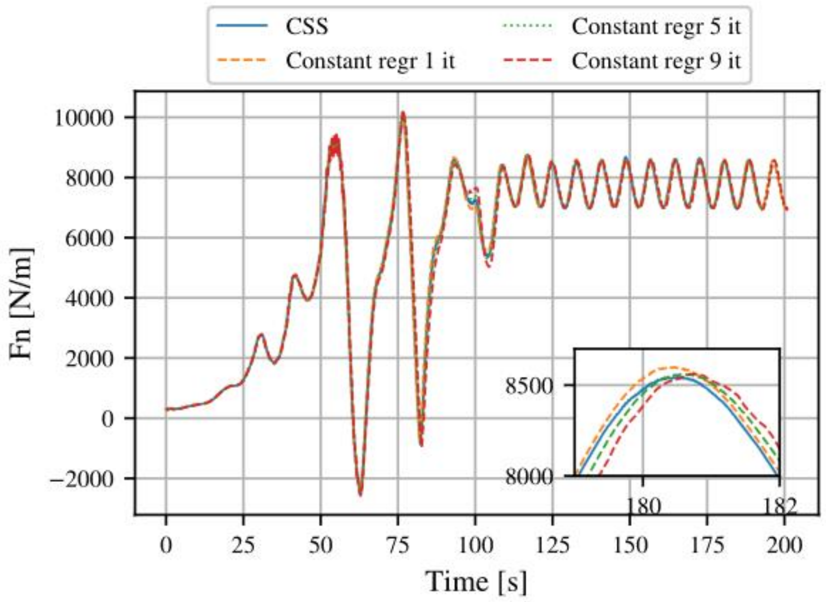
- IEA 15MW turbine, constant wind $U_\infty = 10.6$ m/s, TSR $\lambda = 8.5$
- 200 s simulation using DeepLines and CASTOR with **constant regression**
- CSS reference case: time step $\Delta t_c = 0.01$s

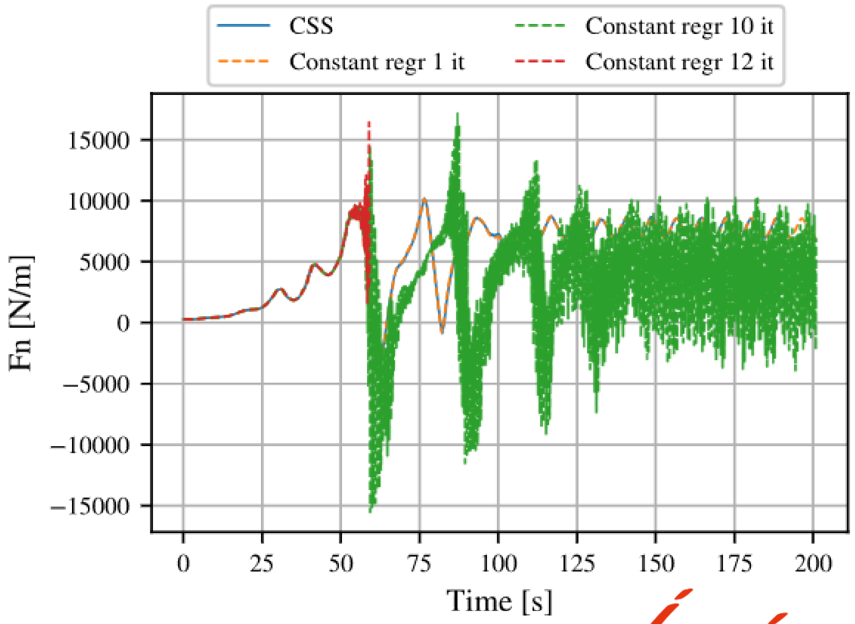
Results:

- Subcycling scheme:
  - Stable up to 9 sub-iterations ($\Delta t_f = 0.1$s).
  - Slight amplitude/phase differences compared to CSS.
- Efficiency gains:
  - Subcycling: **400x faster**
  - Reduced wake filament emissions and meta-model use.

Method	CSS	1 sub-it	2 sub-it	4 sub-it	7 sub-it	9 sub-it
Total comp. time	10d16h	35h06min	11h24min	2h55min	57min	36min

400x faster





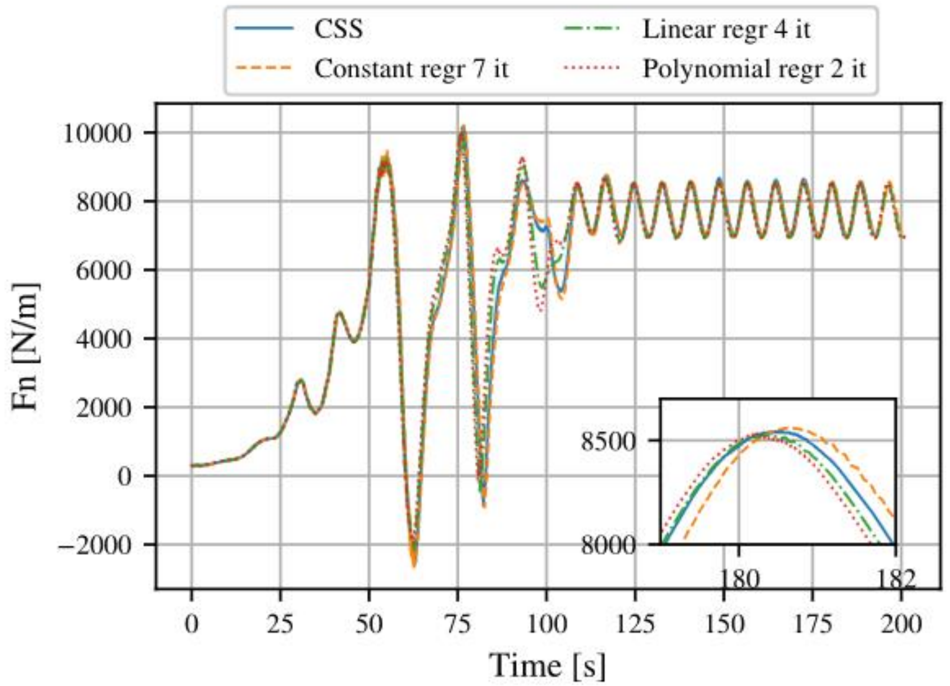
Objective: Assess subcycling scheme impact on aeroelastic simulations.

Setup:

- IEA 15MW turbine, constant wind $U_\infty = 10.6$ m/s, TSR $\lambda = 8.5$
- 200 s simulation using DeepLines and CASTOR with **linear and polynomial regression**
- CSS reference case: time step $\Delta t_c = 0.01$s

Results:

- Subcycling stability:
  - Linear: 4 subcycles max.
  - Polynomial: 2 subcycles max.
- Relative errors:
  - All schemes: slightly underestimate force amplitude.
  - Constant regression: lowest error – Polynomial regression: highest error.



Sub-it number	1	2	4	7	9
Constant regr err (%)	0.78	0.77	0.75	0.79	0.72
Linear regr err (%)	1.09	1.33	1.30	-	-
Polynomial regr err (%)	1.72	1.55	-	-	-

EFFECTS OF PARTITIONED COUPLING SCHEMES ON WIND TURBINE PROBLEM

Objective: Assess the impact of subcycling scheme on aeroelastic wind turbine problem with **turbulent wind**.

Simulation setup:

- IEA 15MW wind turbine subjected to **turbulent wind conditions.**
- TubSim generated.
- Mean wind speed: 8 m/s, turbulence intensity: 8%.

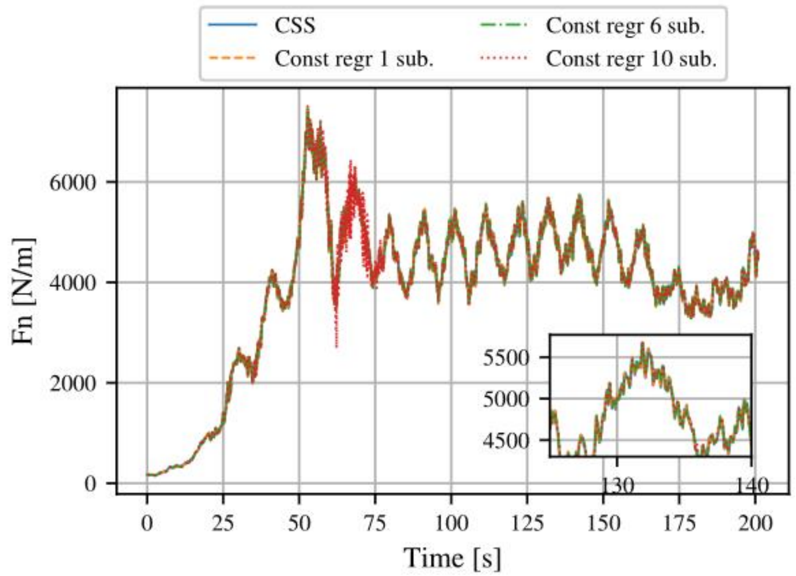
Performance

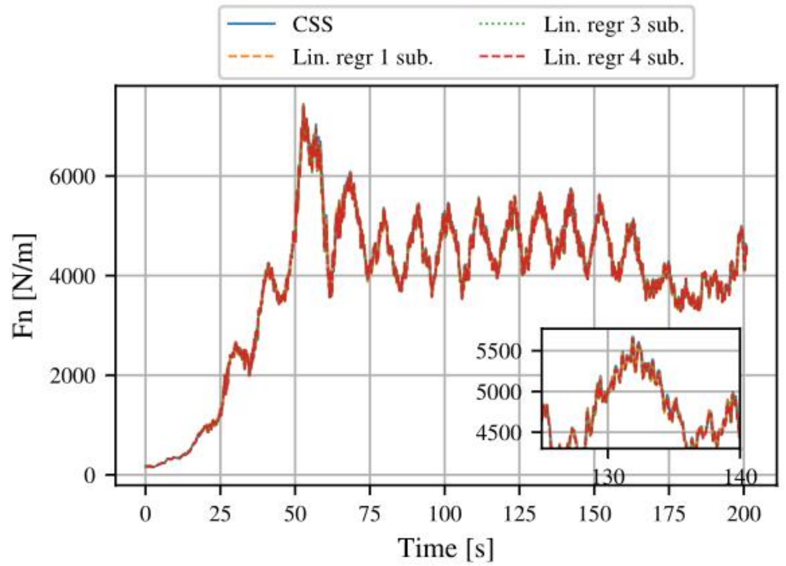
- Constant, linear, and polynomial regressions -> Comparable to constant wind case.

Error analysis:

- Relative amplitude errors ≤ 6% (constant regression with maximum of sub-iterations).

Sub-it number	1	2	4	6	10
Constant regr err (%)	1.01	0.16	1.23	2.09	6.75
Linear regr err (%)	0.11	0.63	0.52	-	-
Polynomial regr err (%)	1.2	1.2	-	-	-





PRESENTATION OUTLINE

CONCLUSIONS

Aerolastic modelling

- Offshore turbine size/complexity -> Need for accuracte aeroelastic models.
- Challenge: Balance computational cost vs. accuracy.

FVW method

- Higher fidelity than BEM, capturing unsteady effects.
- Feasible alternative to CFD but still resource-intensive.
- GPU acceleration: essential to manage large wake structures.
- Python-based solver enables flexible aerodynamic analysis.

Solver coupling

- Subcycling schemes reduce costs while maintaining accuracy.
- 400x speed-up in DeepLines Wind (vs. CSS).

PERSPECTIVES

Extending coupled simulations

- Apply subcycling/partitioned schemes for fully coupled offshore simulations.

Optimizing FVW method in aeroelastic simulations

- Investigate wake coarsening to further reduce computational costs.
- Transition vortex methods to practical design tools.

Refining Python-based FVW tool

- Integrate advanced wake coarsening/desingularization techniques.
- Expand usability for fully coupled aeroelastic simulations.

Enhancing meta-model forecasting

- Use machine learning (e.g., LSTM) for meta-model forecasting.

Thank you for you attention !

Innovating for energy

Find us on:

www.ifpenergiesnouvelles.com

@IFPENinnovation

