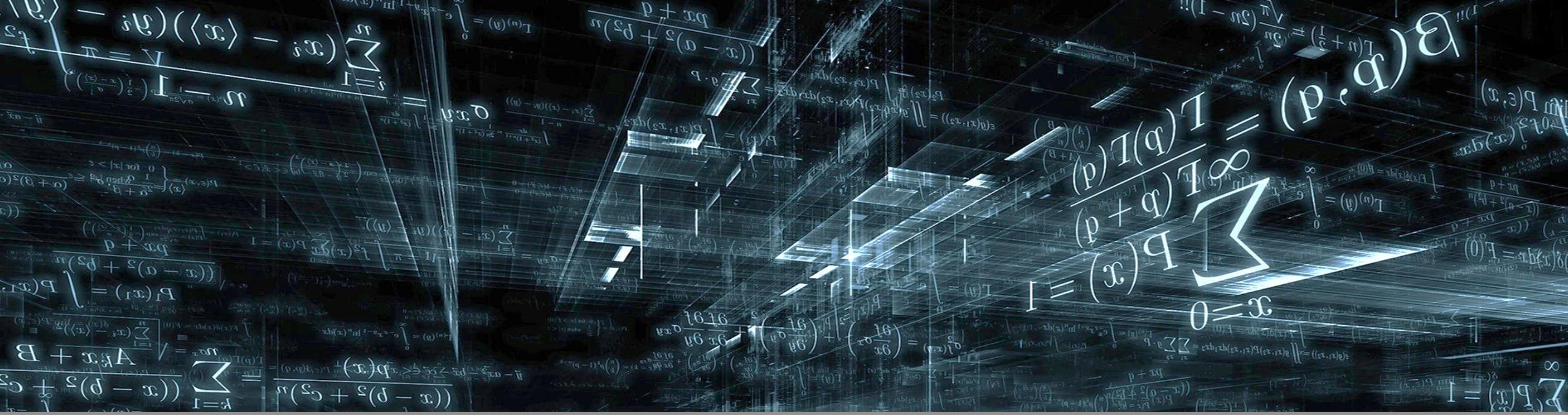


Industry 4.0: Challenges and opportunities for HPC and data-intensive applications

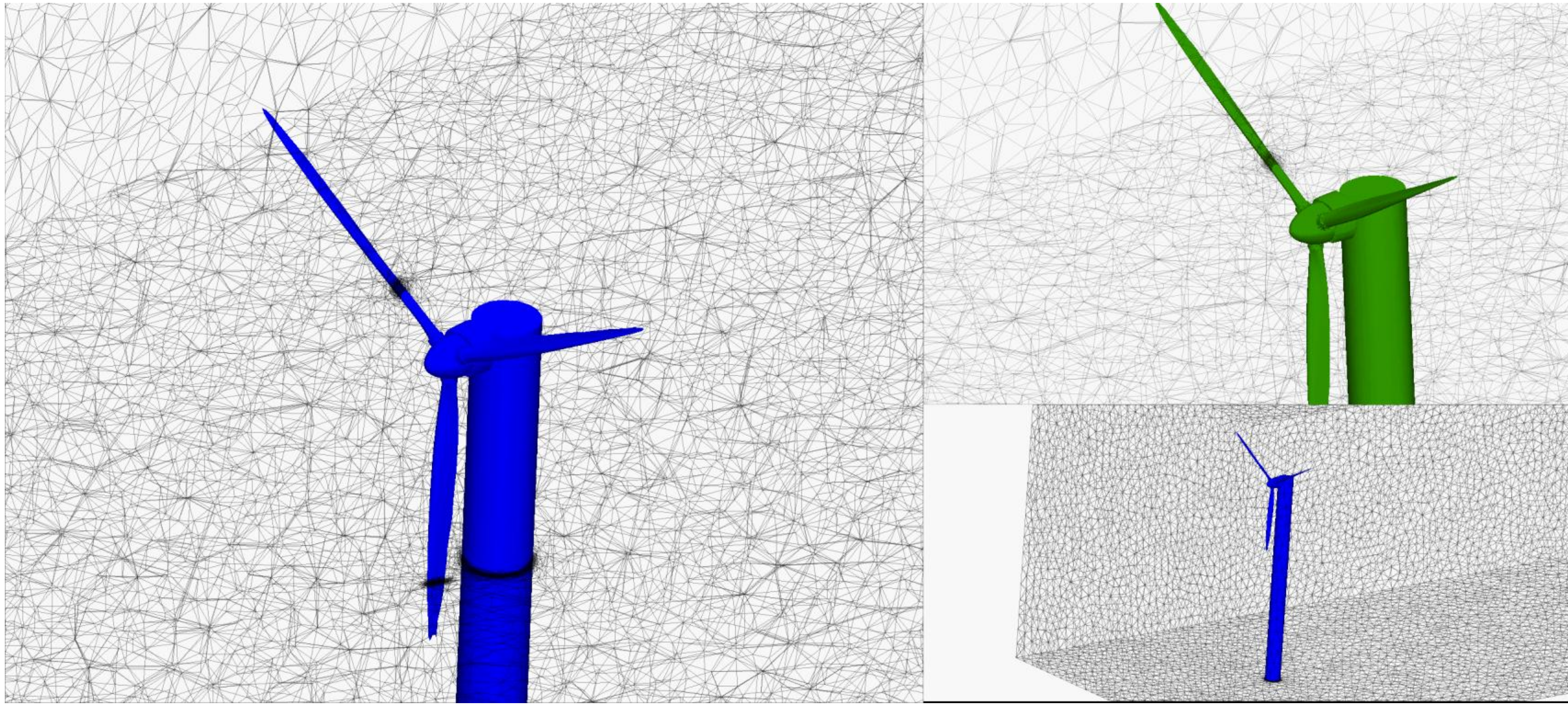
YOUSSEF MESRI

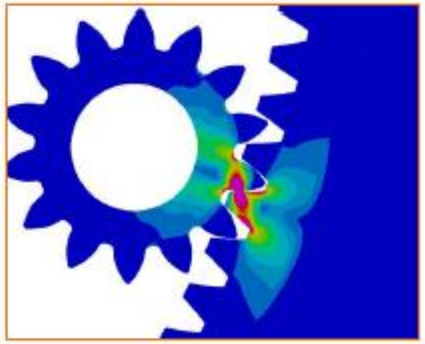
A. BAZILE, C. BEHBAH, E. HACHEM, M. KHALLOUFI, G. MANZINALI, C. MENSAH, R. VALETTE

MINES PARISTECH, PSL UNIVERSITY

CNRS ARMINES Cemef MINES ParisTech

Motivation : Industry 4.0





CAE: Computer Aided Engineering

Outline

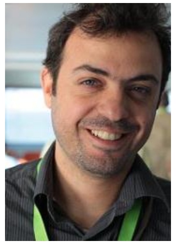
- Context
- Advanced Numerical Methods for CFD
- HPC and Big Data/AI Convergence
- Adaptive Algorithms based Error Estimation
- Conclusions

Applied Maths, High Performance Computing (HPC) and Fluid Modeling

Elie Hachem

HDR 2014

Finite elements and Fluid-Structure Interaction



Youssef Mesri

HDR 2016

Finite elements, HPC & meshing

Rudy Valette

HDR 2014

Rheology & complex fluids

Thierry Coupez

HDR 2000

Finite elements and fluids

Philippe Meliga

HDR 2018 – CNRS

Instabilities & control

RMP, SP2, TMP, CSM,…

Cemef

MINES ParisTech

PERSEE, CDM, CES, CTP, CRC et GeoSciences

ESPCI et IPGG (PSL)
INPHYNI, OCA, INRIA (UCA)

PSL RESEARCH UNIVERSITY PARIS

UNIVERSITÉ CÔTE D'AZUR

Stanford University, FRG
CIMNE UPC
DAMPT Cambridge
Barcelona SuperComputing
M2P2 Aix Marseille

Stanford University

Advanced numerical methods

Advanced Numerical Methods

❑ Anisotropic and parallel 3D mesh adaptation

- A priori and a posteriori error estimators
- Boundary layers & multicriteria adaptation
- Conservative Interpolation

❑ Advanced Finite elements methods

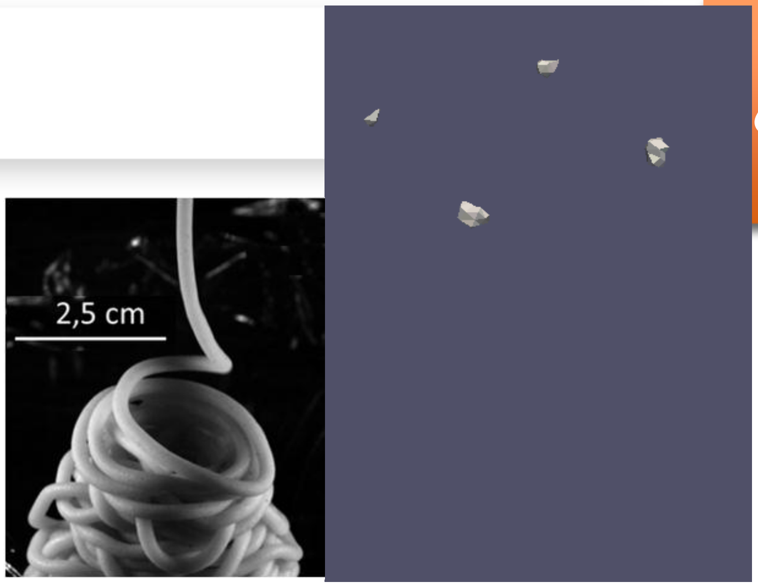
- Adaptive & monolithic framework
- Multi-scale appraoch VMS
- Conservative Levelset methods

❑ Massively parallel computing

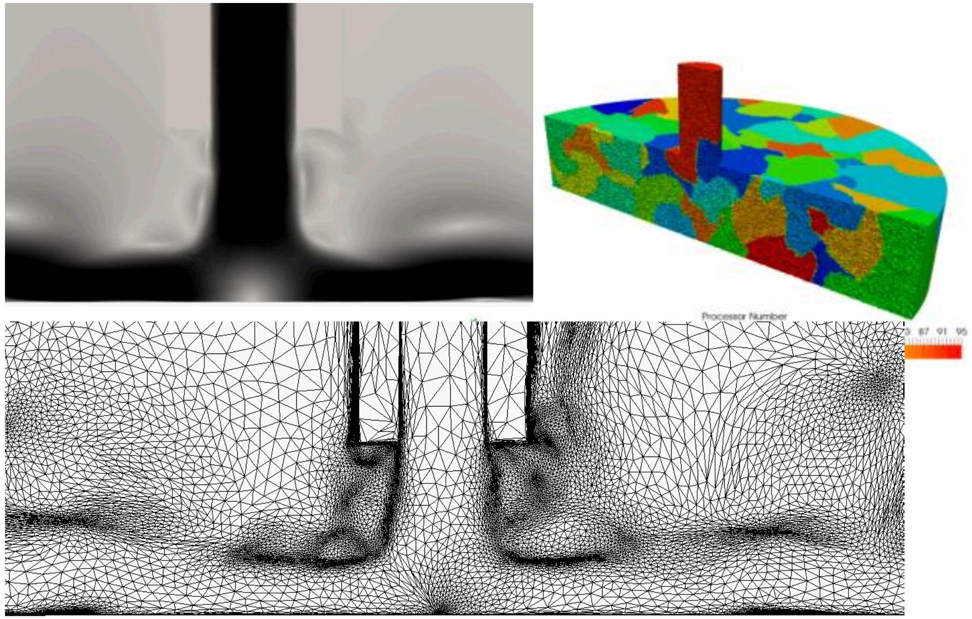
- Hierarchical mesh/domain decomposition
- Load balancing

❑ Immersive methods

- Fluid-Structure Interaction
- Immersed domains: surface mesh & NURBS
- Extension to moving domains.

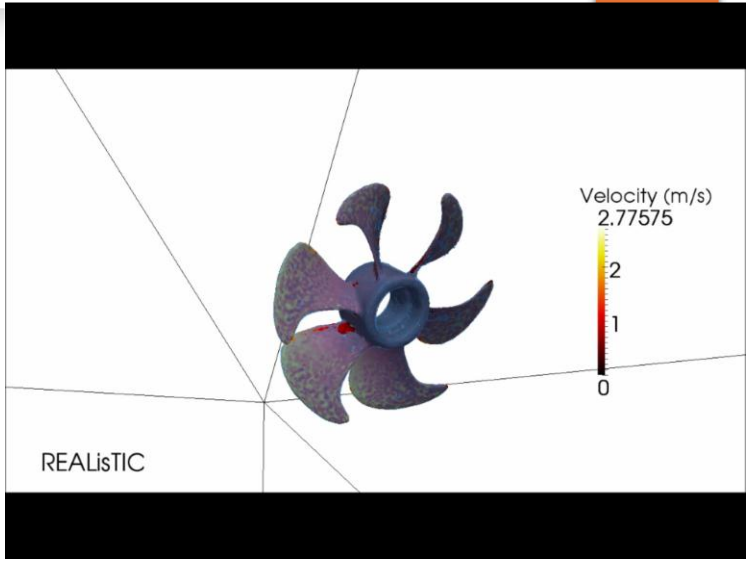


Experimental and numerical characterization of dense suspensions



Impingement jet cooling

Fluid Structure Interaction & Moving Mesh



Anisotropic large mesh deformation

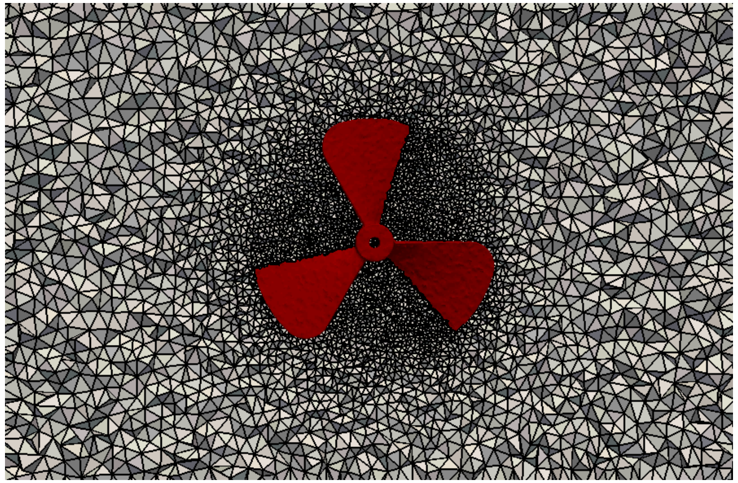
- Inverse Wighting Distance
- Mesh quality-based appraoch
- Conservative a priori geometric features (i.e. boundary layers, …)

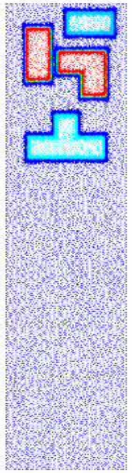
Multi-body configurations

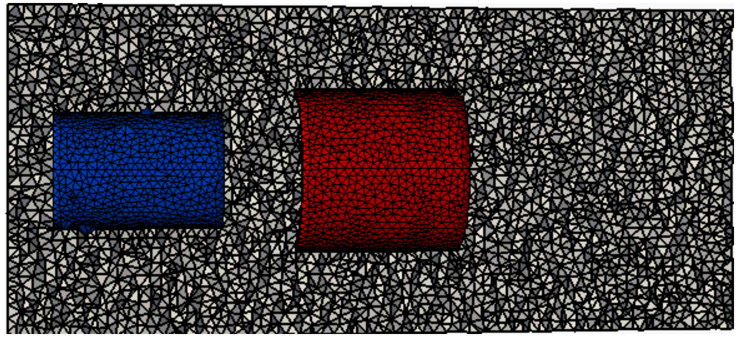
- Adaptive & monolithic framework
- Multi-scale appraoch VMS (iLES)

Massively parallel computing

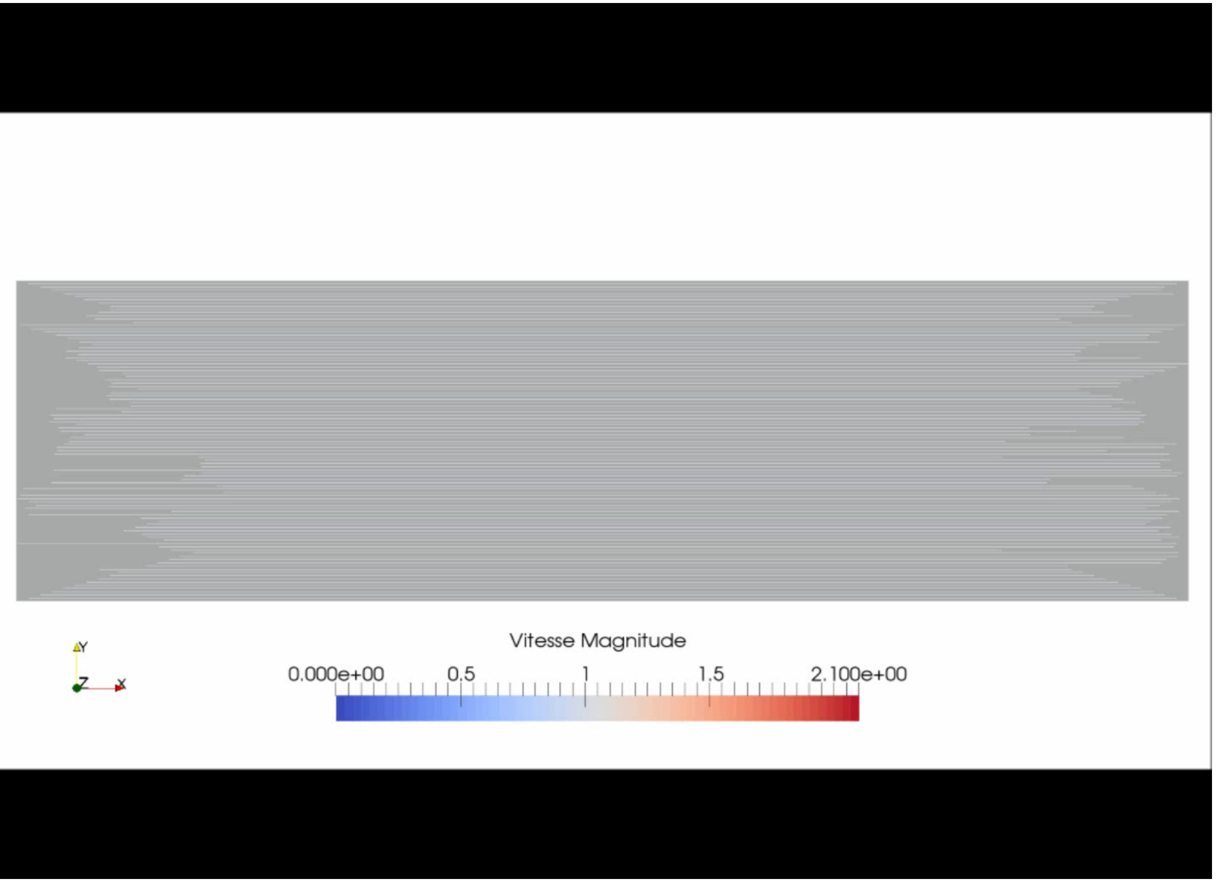
- Hierarchical algorithms
- Load balancing

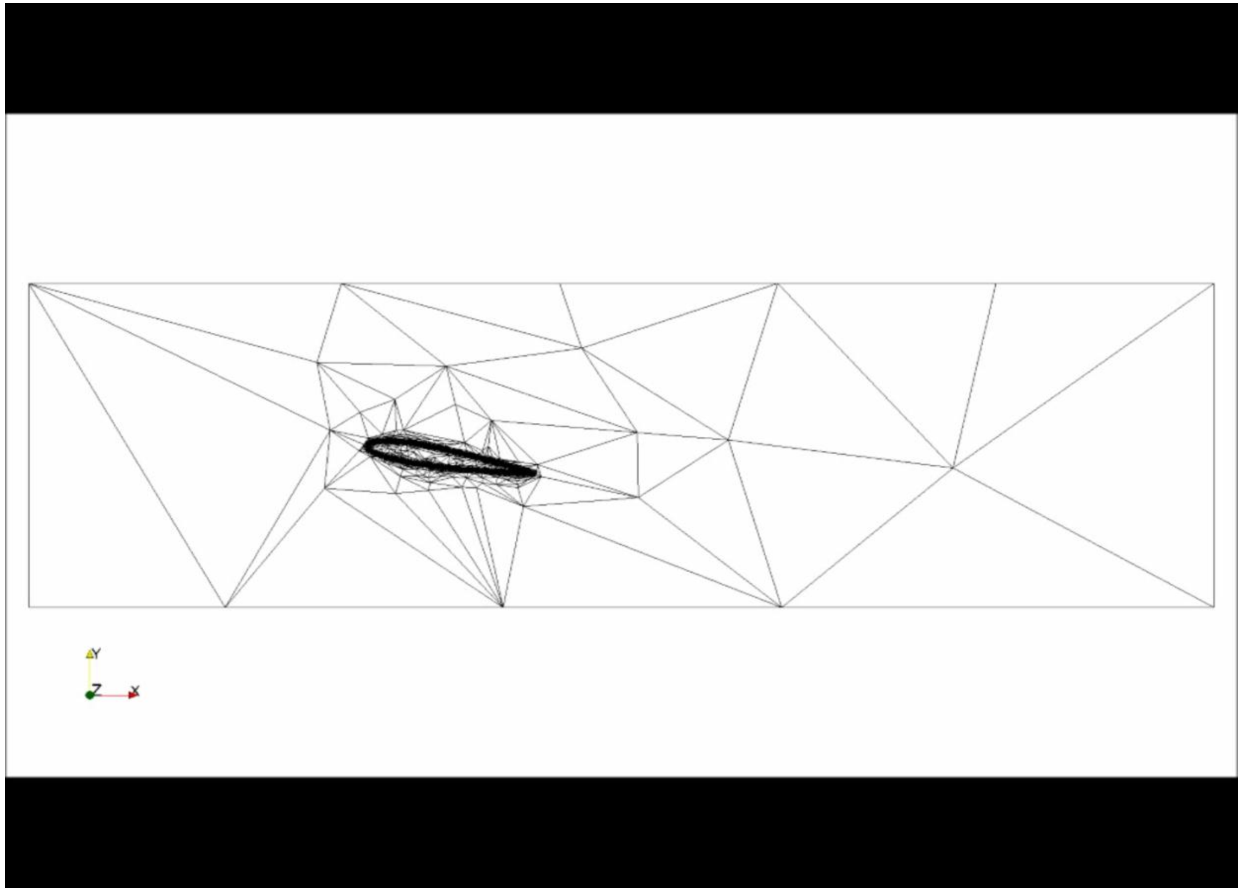






Turbulent incompressible Navier-Stokes flow



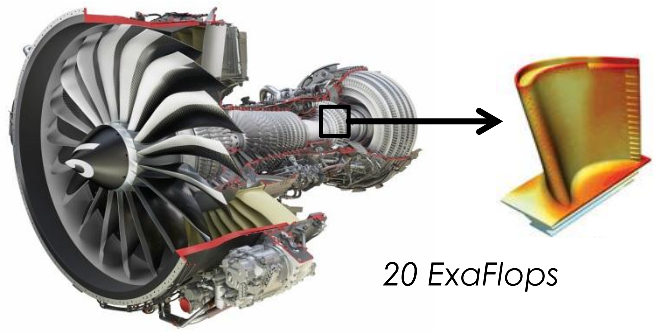


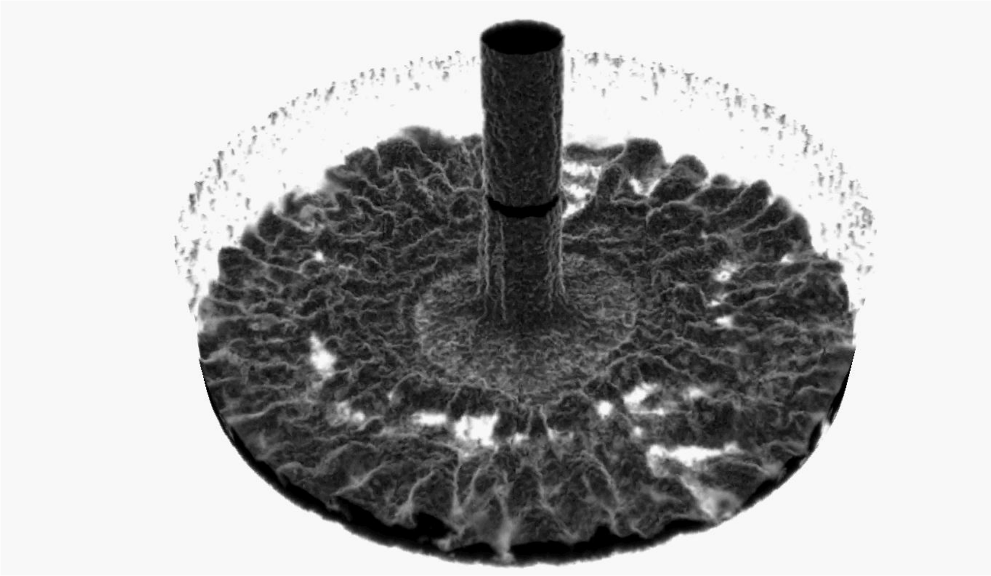
Vortex-Induced Vibration

Computing and simulation challenges examples

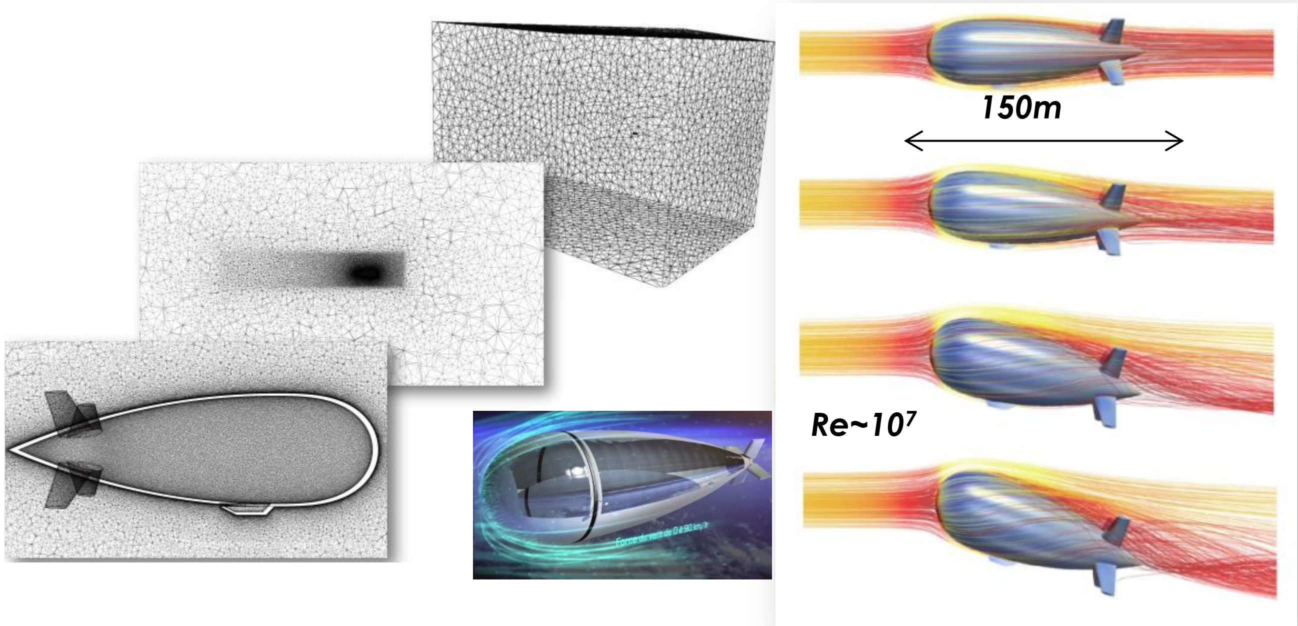
***Challenge 1:** Reduce environment impact of the civil aviation*

SAFRAN AIRCRAFT ENGINES

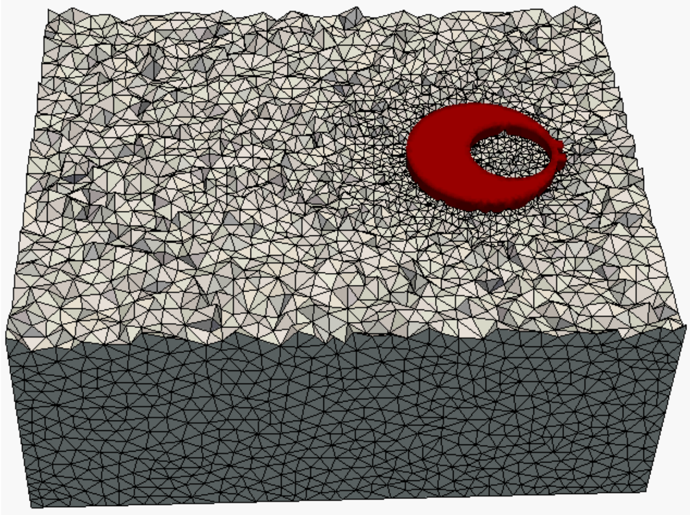


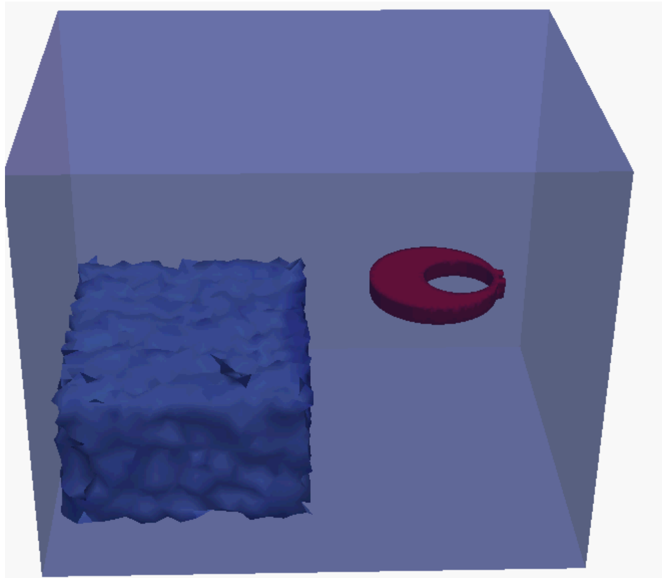


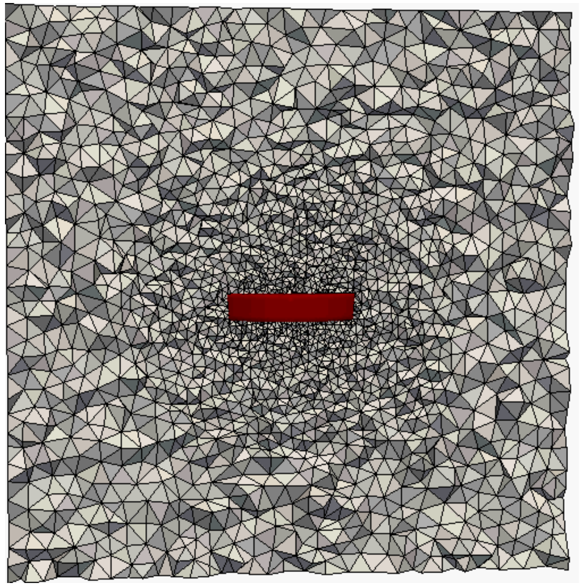
***Challenge 2:** Design of new stratospheric plateforms*

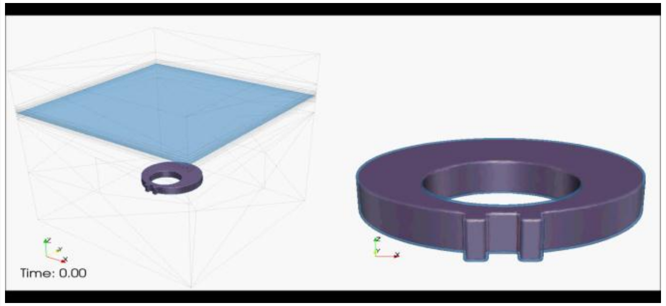


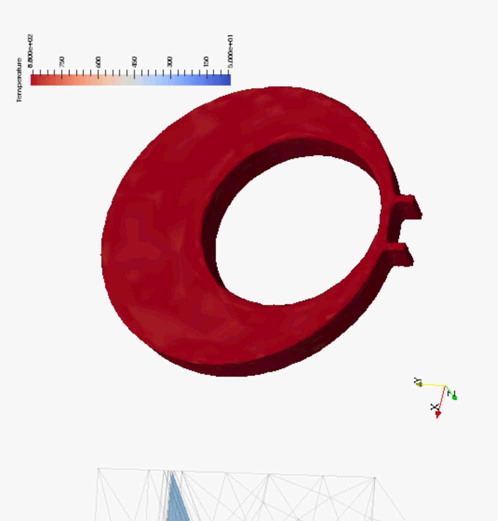
FSI and Industrial heat treatment











Industry 4.0: Industrial partnership

Industrial Chair: INFINITY

Title: « Multi-scale digital framework for safety design of indutrial quenching processes »

12 industrial partners

Period: 2018 – 2022

Site web: www.chaireinfinity.fr

Design of new high-quality material products

Convergence of HPC and Data Science

Extract useful information from large data set to perform efficiently large scale simulations

- Convergence between:
  - Models
  - High performance simulation algorithms and data analytics

HPC vs BigData/AI

Parallelism for scalability

HPC	Big Data
Performance comes first	Ease of programming comes first
Low level programming (MPI+X)	High level programming (Spark, TensorfFlow)
Thin Software stack	Thick software stack
Stable software libs	Quickly changing libs
HPC centers	Cloud platforms
Jobs runs few hours on thoursands of cores	Jobs runs few days on tens of nodes

Source: HPC-Big Data Lab - Inria

HPC and BigData/AI Convergence

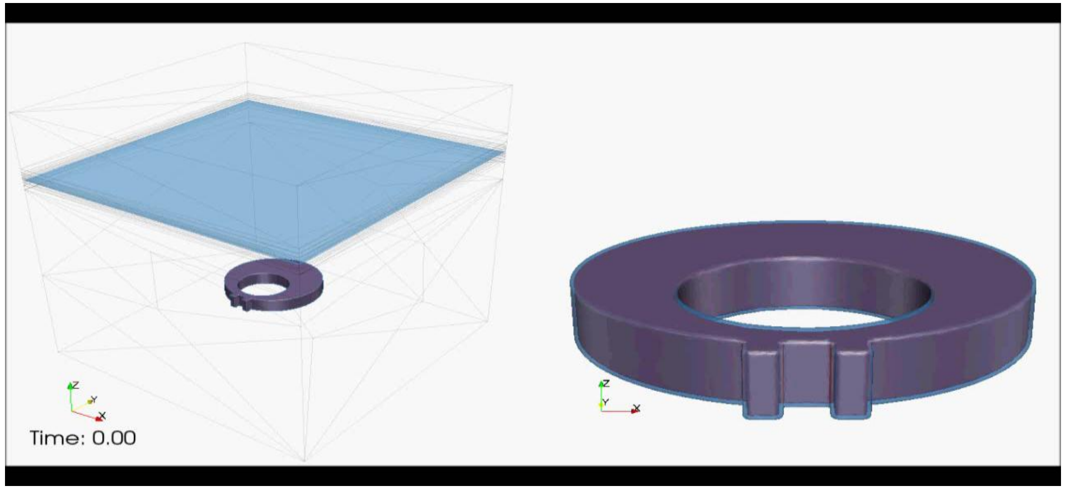
Two research directions

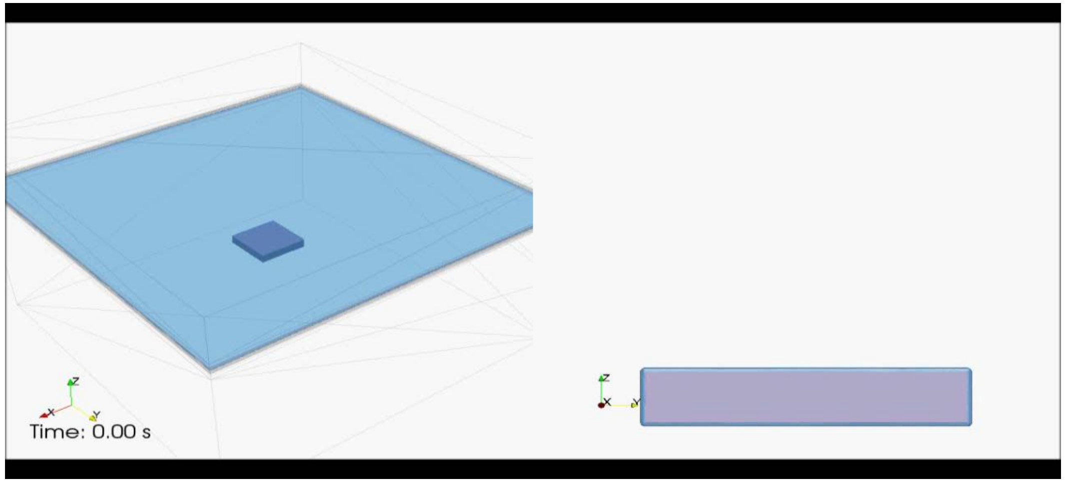
- Accelerating AI and Big Data with the help of HPC
  - Accelerating ML/DL with task-based programming (Dask, StarPU)

- AI/Big Data analytics for large scale scientific simulations
  - Large scale parallel deep reinforcement learning (Turbulence models, optimization,…)

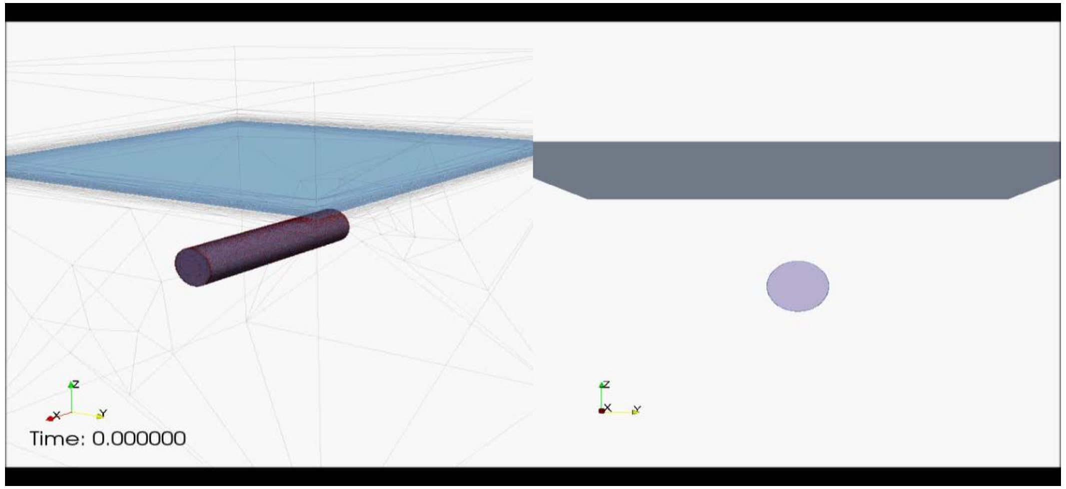
Fast decision-making Tool - deep reinforcement learning

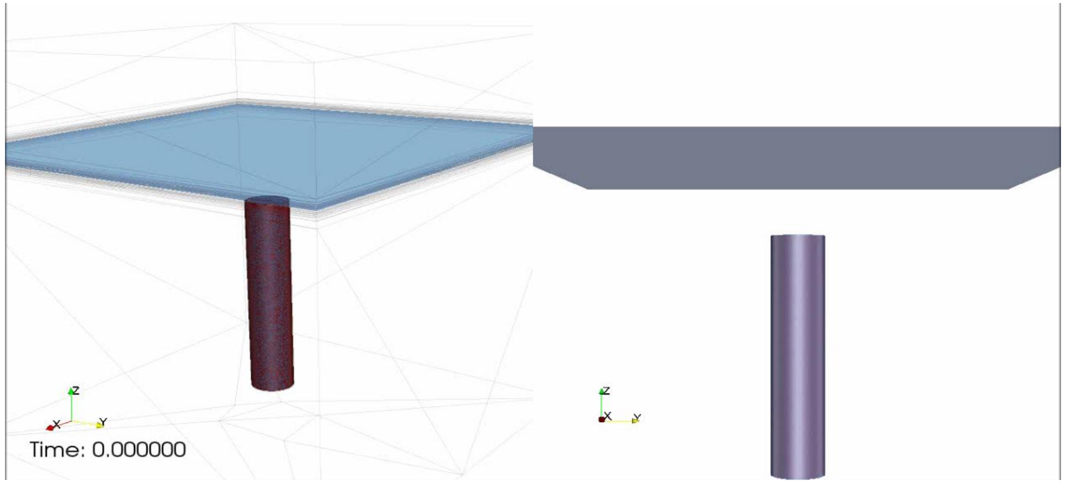
Geometry effect





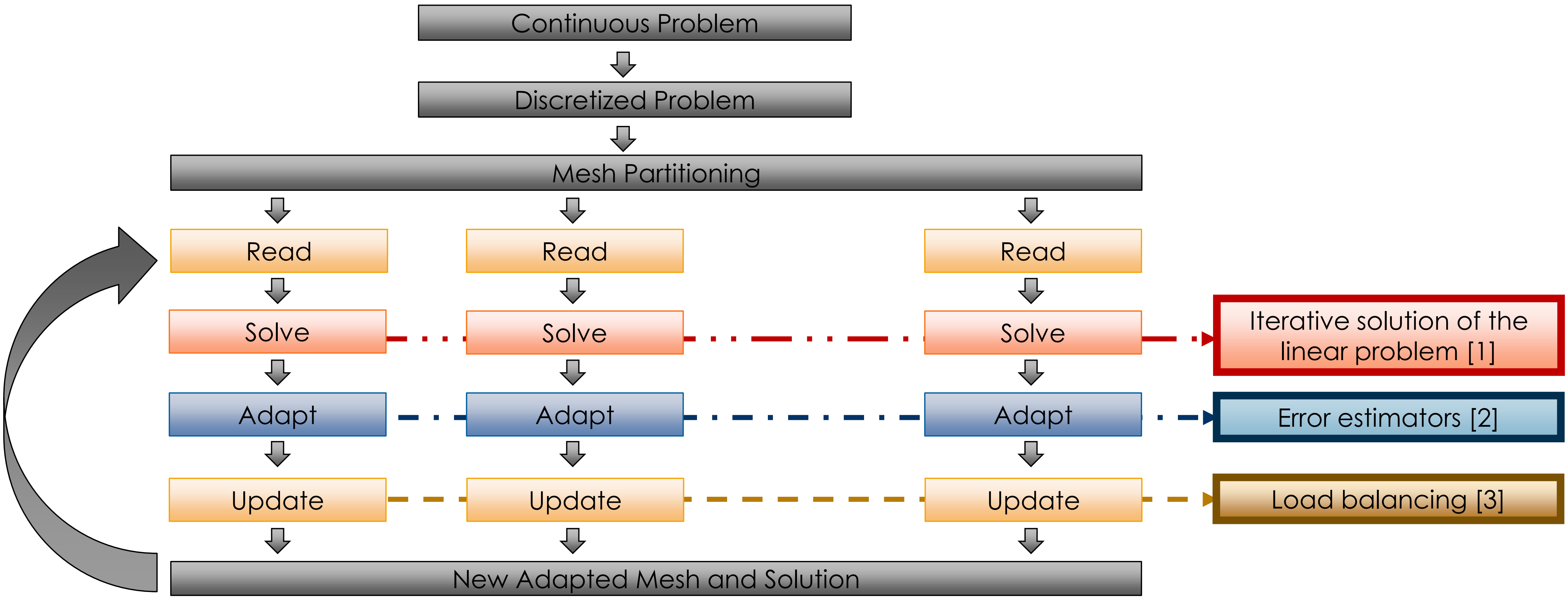
Orientation effect





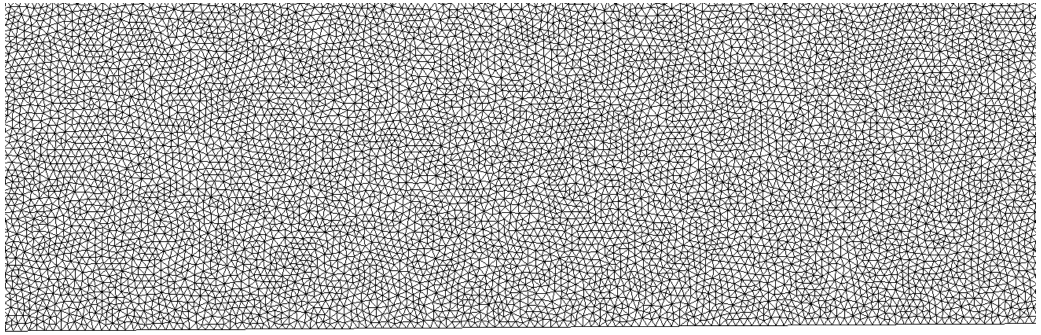
Numerical Simulation – PhD M. Khalloufi, Tom Bell Prize 2017

Parallel Adaptive Finite Element Framework



[1] G. Manzinali, E. Hachem, Y. Mesri, Adaptive stopping criterion for iterative linear solvers combined with anisotropic mesh adaptations, CMAME, 2018
[2] Bazile, A., Hachem, E., Larroya-Huguet, J. C., Mesri, Y. Variational Multiscale error estimator for anisotropic adaptive fluid mechanic simulations.CMAME, *331*, 94-115, 2018.
[3] Y. Mesri, Predictive load balancing for parallel anisotropic mesh adaptation applications, preprint, 2018

Anisotropic and dynamic mesh adaptation



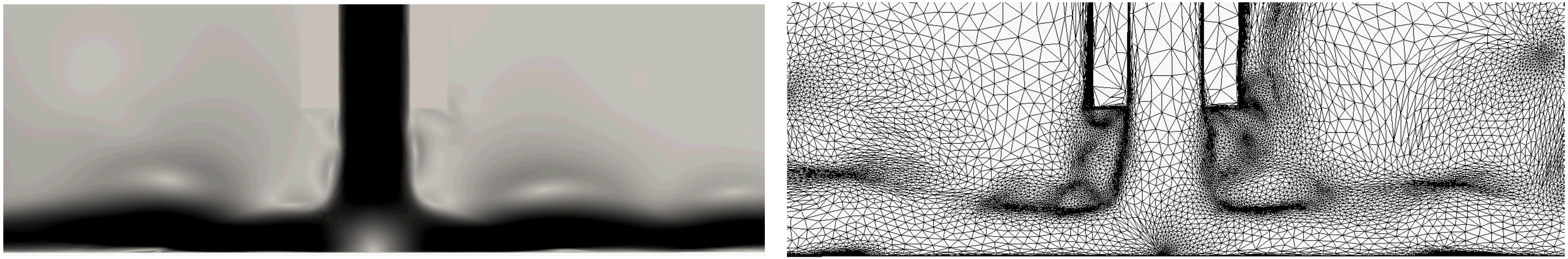
Modify the mesh anisotropically and dynamically according to a criteria.

add, remove, stretch cells.

along a certain direction

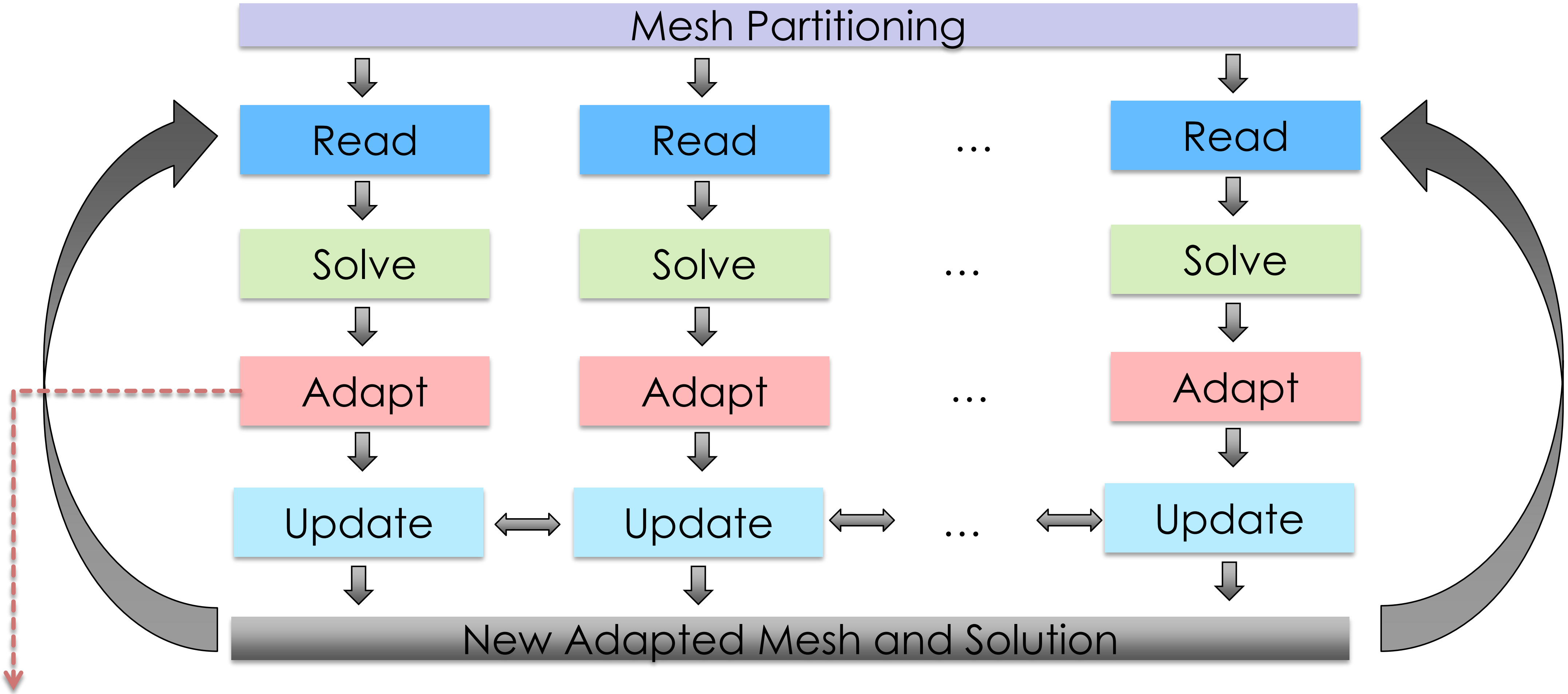
in the course of time

via an error estimation, or the solution's variation.



Dynamic anisotropic mesh according to velocity variations: velocity field (left), adapted mesh (right).

Dynamic and parallel mesh adaptation algorithm



<u>Adapt:</u>

1. **Estimate the error at each iteration step.**
2. Correlate the error with the mesh geometry.
3. Generate a new mesh in the metric space.

Error Indicator:

$$\eta_{\Omega_e} = d|\Omega_e|^{\frac{1}{p}}|\lambda_d(x_0)|h_d^2$$

Type 1: Interpolation error estimator

$$\|u - u_h\|_{L^p(\Omega)} \leq C\|u - \Pi u\|_{L^p(\Omega)}$$

Interpolation error estimator:

- d the dimension of the problem,
- λ_d the eigenvalue of the recovered Hessian matrix $H_R(u)(x)$,
- h_d the size of the element in the direction d.

$$\bar{\eta}_T^p = \int_T \left(\mathcal{H}(u_h(x_T))(x - x_T) \cdot (x - x_T)\right)^p dT$$

Anisotropic metric definition:

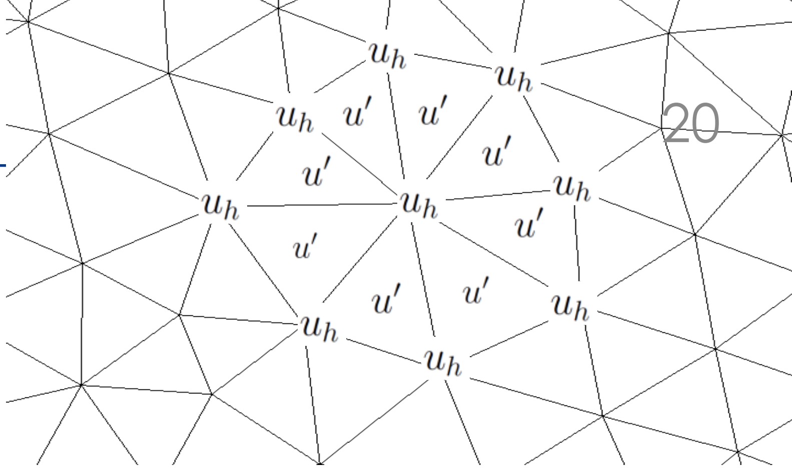
$$\mathcal{H} = \mathcal{R}\Lambda\mathcal{R}^T = |\lambda_1| e_1 \otimes e_1 + \ldots + |\lambda_d| e_d \otimes e_d$$

$$\begin{cases} \text{Find } h_T = \{h_{1T}, \cdots, h_{dT}\}, T \in \mathscr{T}_h \text{ that minimizes the cost function} \\ F(h_T) = \sum_{T \in \mathscr{T}_h} (\eta_T)^p \\ \text{under the constraint} \quad N_{\mathscr{T}_h'} = C_0^{-1} \sum_{T \in \mathscr{T}_h} \int_T \prod_{i=1}^{d} \frac{1}{h_{iT}} \, dT \end{cases}$$

➢Modification of the mesh according to this geometrical transformation.

Mesri, Y., Khalloufi, M., & Hachem, E. (2016). On optimal simplicial 3D meshes for minimizing the Hessian-based errors. *Applied Numerical Mathematics*, *109*, 235-249.

Type 2: A posteriori subscales error estimator



A posteriori subscales error estimator: $||u'||_{L^\infty(\Omega)}$

Variational Multi Scale framework: $u = u_h + u' \quad u_h \in \mathcal{S}_h, \quad u' \in \mathcal{S}'$

resolved part (coarse scales) ← u_h ; u' → unresolved part (subscales) $\approx Error$

Two contribution to the error: $u'(\mathbf{x}) = u'_{bub}(\mathbf{x}) + u'_{poll}(\mathbf{x})$

u'_{bub} → Internal residual error (inside the element)

u'_{poll} → Inter-element error (outside the element)

❖ *negligible for convective-dominated heat transfer*

We obtain an explicit expression of the error:

$$u'(\mathbf{x}) \approx u'_{bub}(\mathbf{x}) = \sum_{i=1}^{n_{bub}} c_i^b b_i(\mathbf{x})$$

Irisarri, D., & Hauke, G. (2017). Pointwise Error Estimation for the One-Dimensional Transport Equation Based on the Variational Multiscale Method. *IJCM*, *14*(04), 1750040.

➢ We compute explicitly the a posteriori subscales error estimator.

New isotropic & anisotropic metric mesh adaptation based on subscales

New isotropic metric based mesh adaptation:

We propose the following new isotropic metric tensor:

$$\mathcal{H}_{iso} = \mathcal{R}\Lambda\mathcal{R}^T = |\lambda| e_1 \otimes e_1 + \ldots + |\lambda| e_d \otimes e_d$$

With:

$$|\lambda| = \frac{1}{h_{new}^2} = \frac{\|u'\|_{L^\infty(\Omega_e)}}{u'_{TOL}} \times \frac{1}{h^2}$$

Where $\mathcal{R}$ is the orthogonal matrix built with eigenvectors $(e_i)_{\{i=1,..,d\}}$ of $H_R(u_h(x))$

New anisotropic metric based mesh adaptation:

We propose a new anisotropic local error indicator:

$$\eta_{\Omega_e,new} = d|\Omega_e|^{\frac{1}{p}} \times |\lambda_d(x_0)| \times \frac{\|u'\|_{L^\infty(\Omega_e)}}{u'_{TOL}} \times h_{d,new}^2$$

With

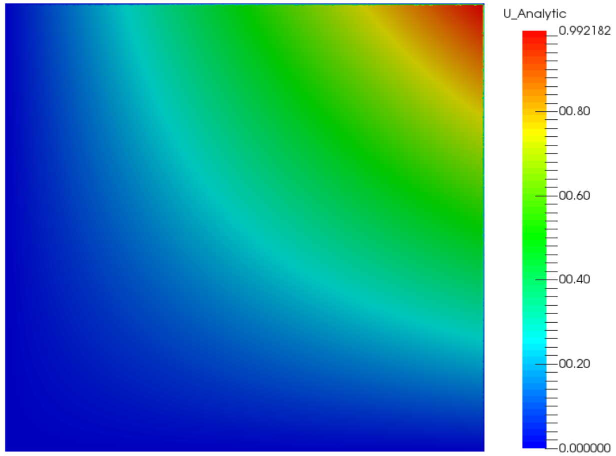
$$\mathcal{H}_{aniso}^{new} = \mathcal{R}\Lambda\mathcal{R}^T = \frac{\|u'\|_{L^\infty(\Omega_e)}}{u'_{TOL}} |\lambda_1| e_1 \otimes e_1 + \ldots + \frac{\|u'\|_{L^\infty(\Omega_e)}}{u'_{TOL}} |\lambda_d| e_d \otimes e_d$$

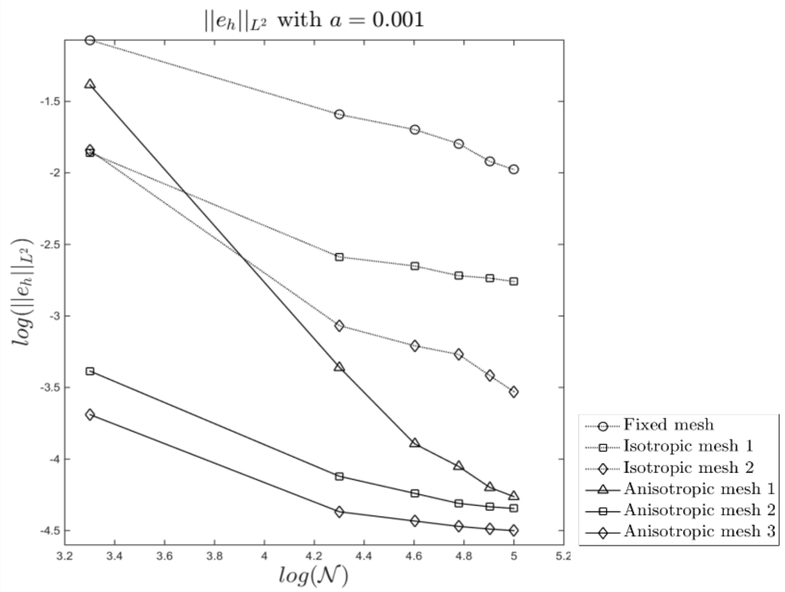
Where $\mathcal{R}$ is the orthogonal matrix built with eigenvectors $(e_i)_{\{i=1,..,d\}}$ of $H_R(u_h(x))$

Bazile, A., Hachem, E., Larroya-Huguet, J. C., & Mesri, Y. (2018). Variational Multiscale error estimator for anisotropic adaptive fluid mechanic simulations: Application to convection–diffusion problems. *CMAME*, *331*, 94-115.

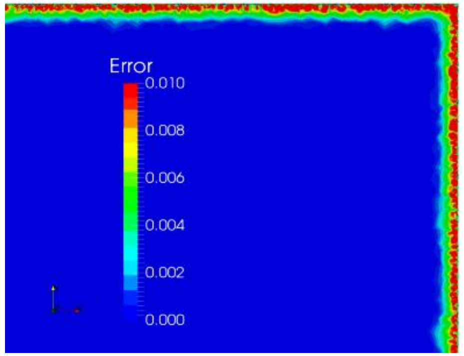
Numerical benchmarks with analytical solutions

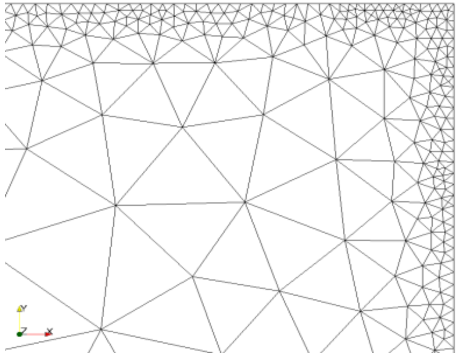
Numerical example in 2D:





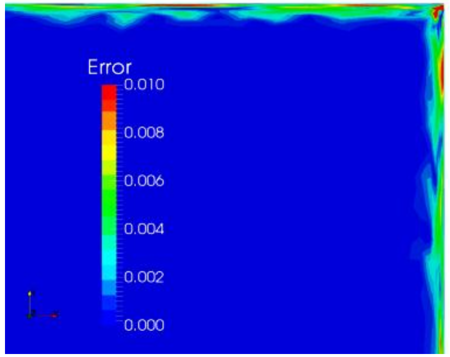
Isotropic mesh 2

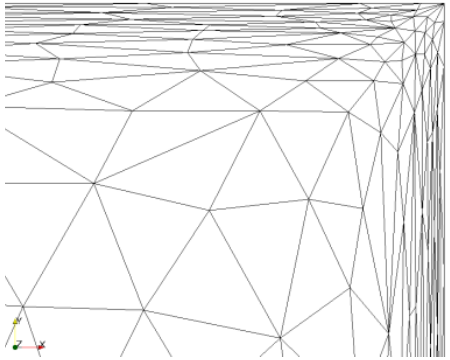




$N = 20\,798$

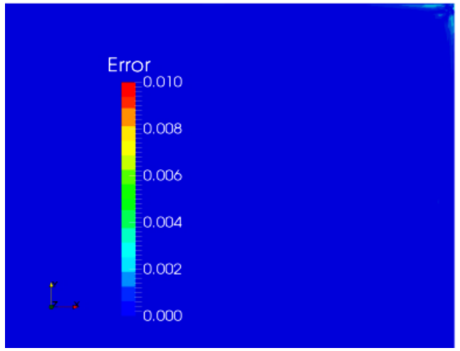
Anisotropic mesh 1

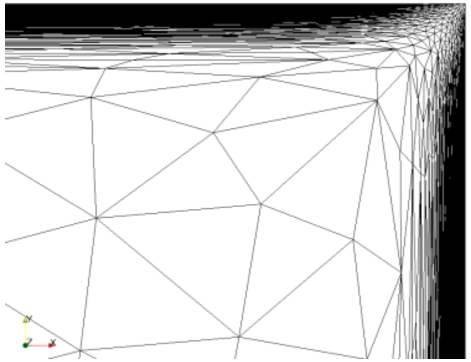




$N = 20\,932$

Anisotropic mesh 3



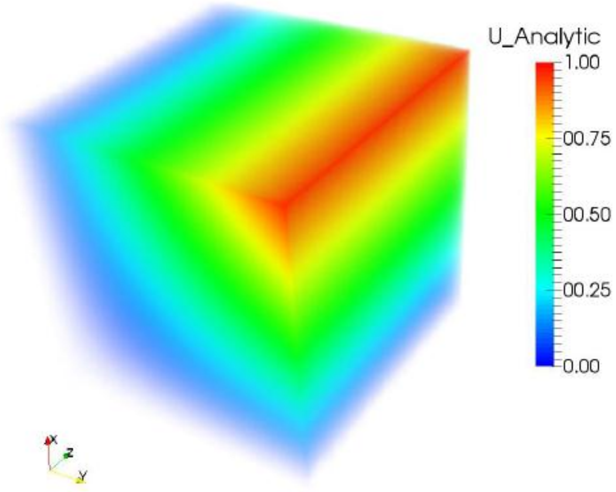


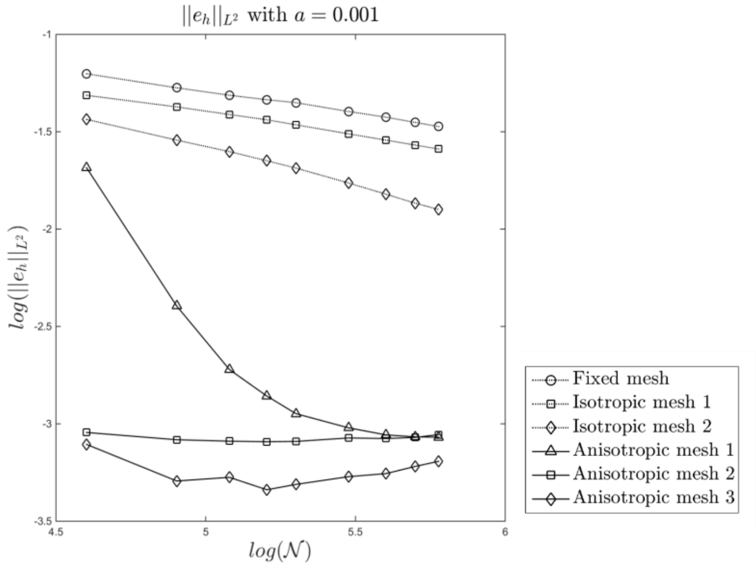
$N = 23\,182$

Bazile, A., Hachem, E., Larroya-Huguet, J. C., & Mesri, Y. (2018). Variational Multiscale error estimator for anisotropic adaptive fluid mechanic simulations: Application to convection–diffusion problems. *CMAME*, *331*, 94-115.

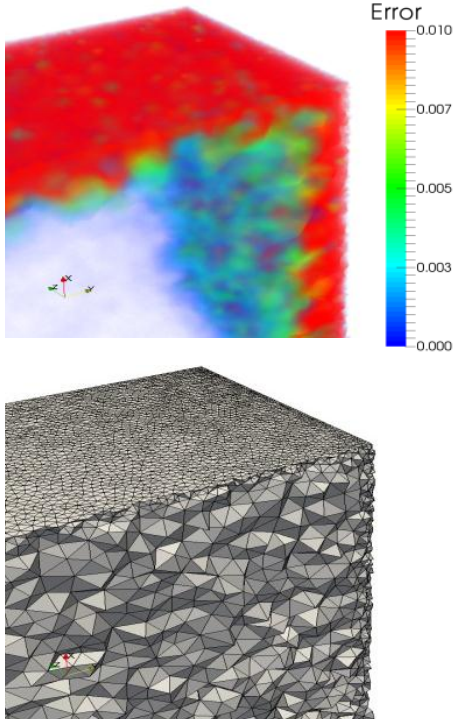
Numerical benchmarks with analytical solutions

Numerical example in 3D:



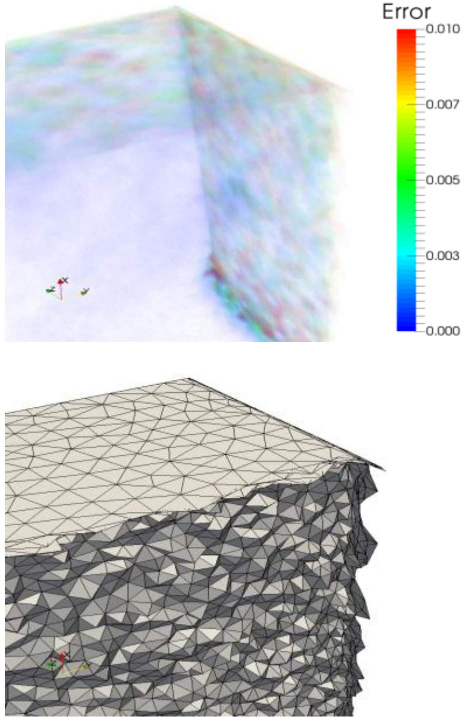


Isotropic mesh 2



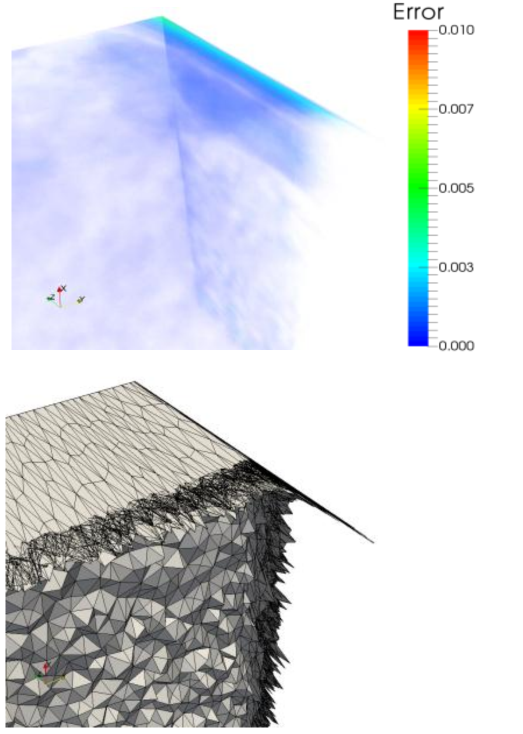
$N = 198\,946$

Anisotropic mesh 1



$N = 212\,242$

Anisotropic mesh 3

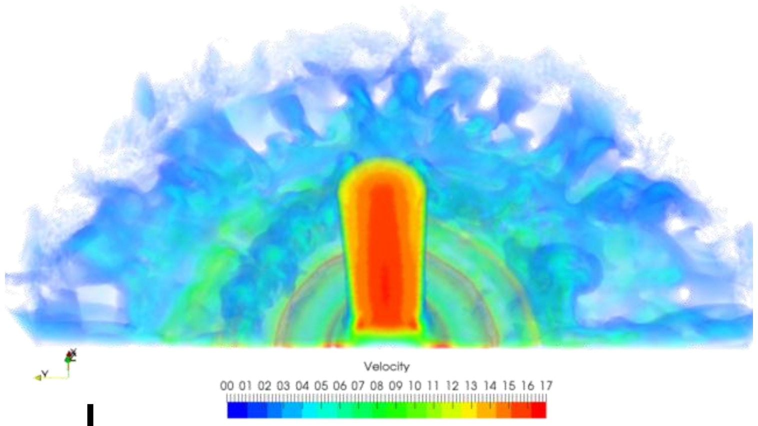


$N = 199\,659$

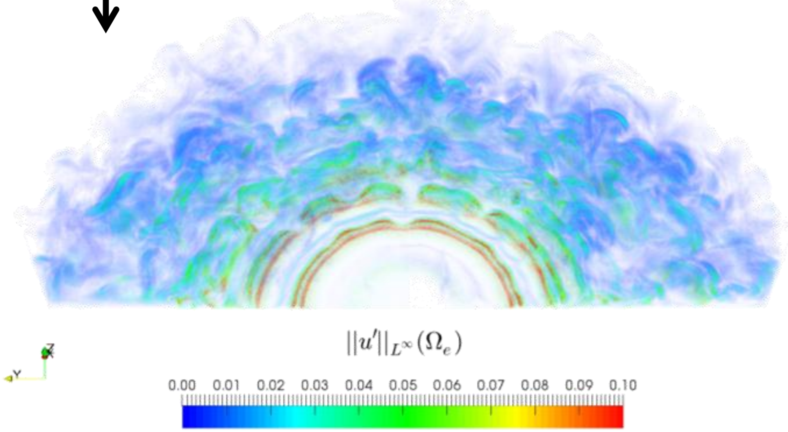
Bazile, A., Hachem, E., Larroya-Huguet, J. C., & Mesri, Y. (2018). Variational Multiscale error estimator for anisotropic adaptive fluid mechanic simulations: Application to convection–diffusion problems. *CMAME*, *331*, 94-115.

Application to the impingement jet cooling

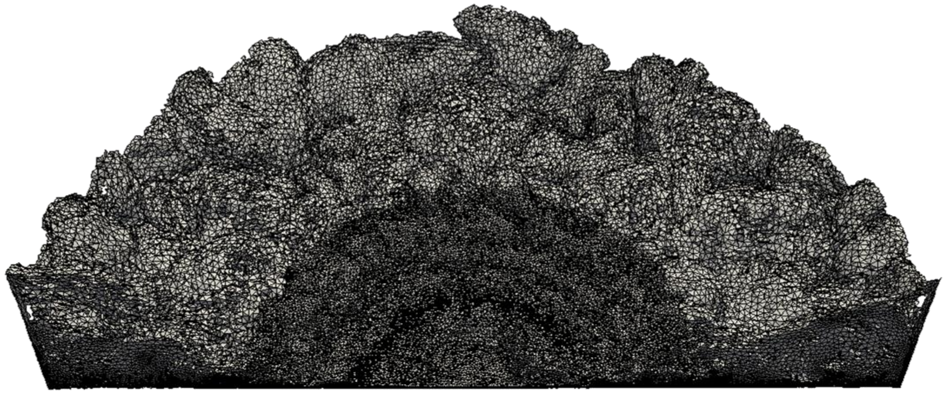
Capturing the thermal activity due to the secondary vortexes with the new anisotropic mesh adaptation based on the subscales error estimator.



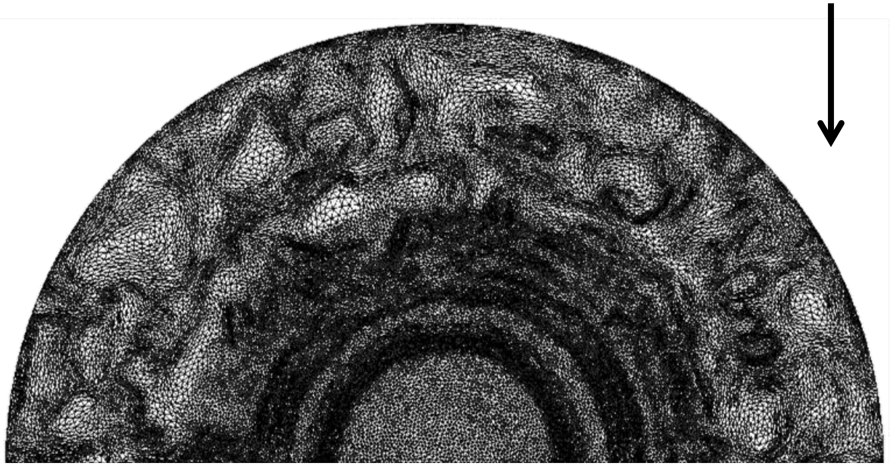
Velocity field:



A posteriori subscales error estimate:



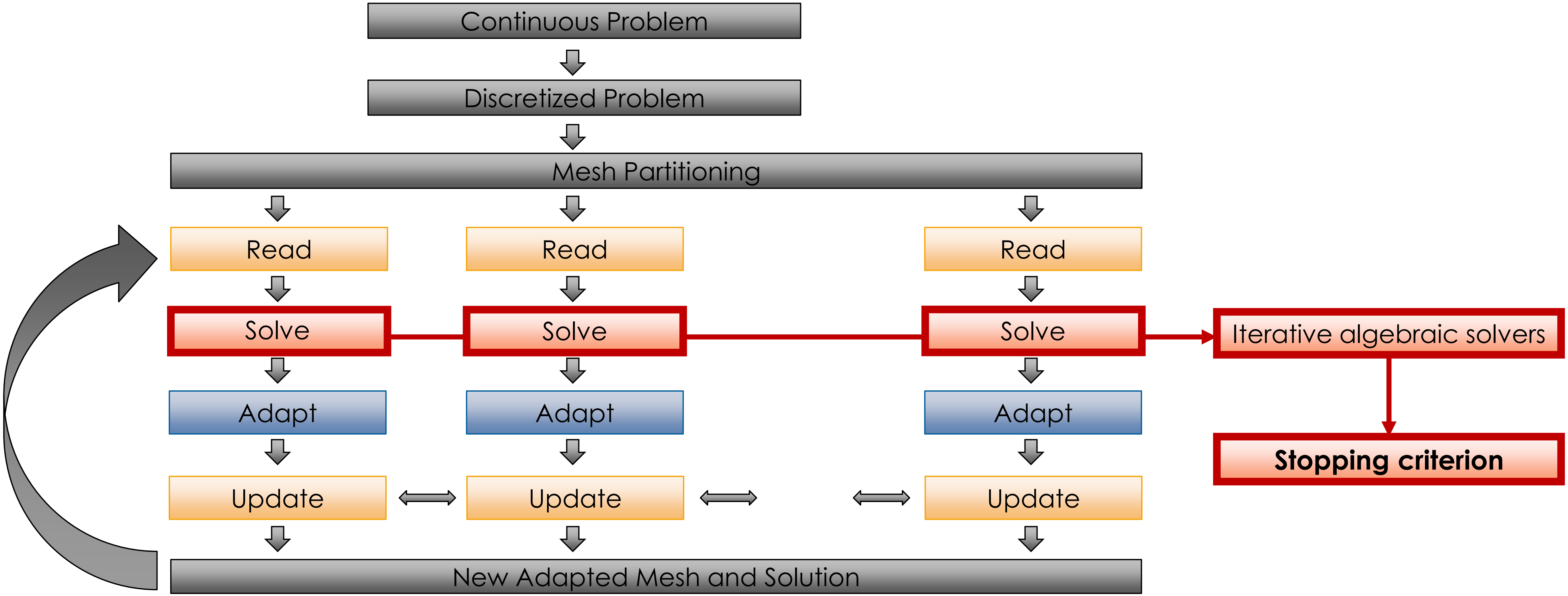
Multiscale adapted mesh (using $\eta_{\Omega_e,new}$):



Slice view on plan XY at z=0

Adaptive Stopping Criteria for Iterative Solvers

AFEM framework – Adaptive Stopping Criterion



G. Manzinali, E. Hachem, Y. Mesri, Adaptive stopping criterion for iterative linear solvers combined with anisotropic mesh adaptation, application to convection-dominated problems,

Adaptive stopping criterion – Error estimate based

The main idea is to **stop** the iterative resolution when the **algebraic error** is lower than to the **estimated error**

$$\|\nabla(u_h - u_h^n)\|_{L^2} \leqslant c \left(\sum_{T \in \mathcal{T}_h} \eta_T^2 \right)^{1/2}$$

Assumptions

- **Algebraic error** approximated to the **residual error**

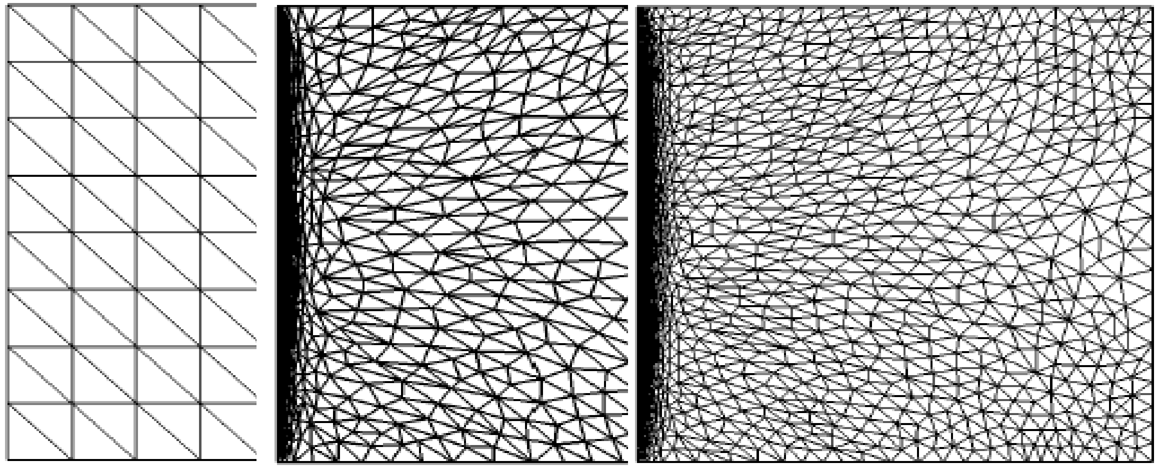
$$\|\nabla(u_h - u_h^n)\|_{L^2}^2 \approx \|\mathbf{r}^n\|_{L^2}^2 \longrightarrow \|\mathbf{r}^n\| \leq 0.01 \left(\sum_{T \in \mathscr{T}_h} \eta_T^2 \right)^{1/2}$$

- Scaling factor $c = 0.01$

[3] Marco Picasso. A stopping criterion for the conjugate gradient algorithm in the framework of anisotropic adaptive finite elements. International Journal for Numerical Methods in Biomedical Engineering, 25(4):339–355, 2009

Numerical results – Steady Problems

We start from a coarse uniform **mesh** and we perform 30 adaptation steps



SOLVER:

- Conjugate Gradient (Laplace)
- GMRES (Convection – diffusion)

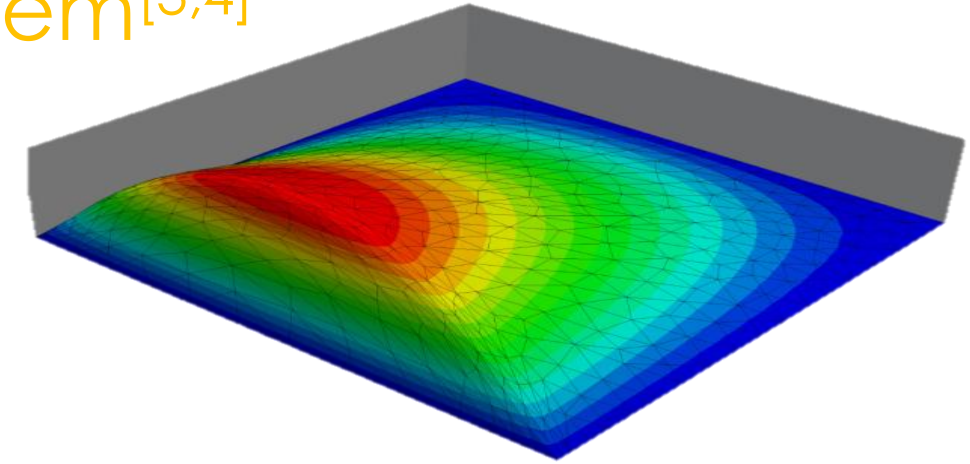
PRECONDITIONER:

- ILU(k)
- For GMRES right preconditioner implemented

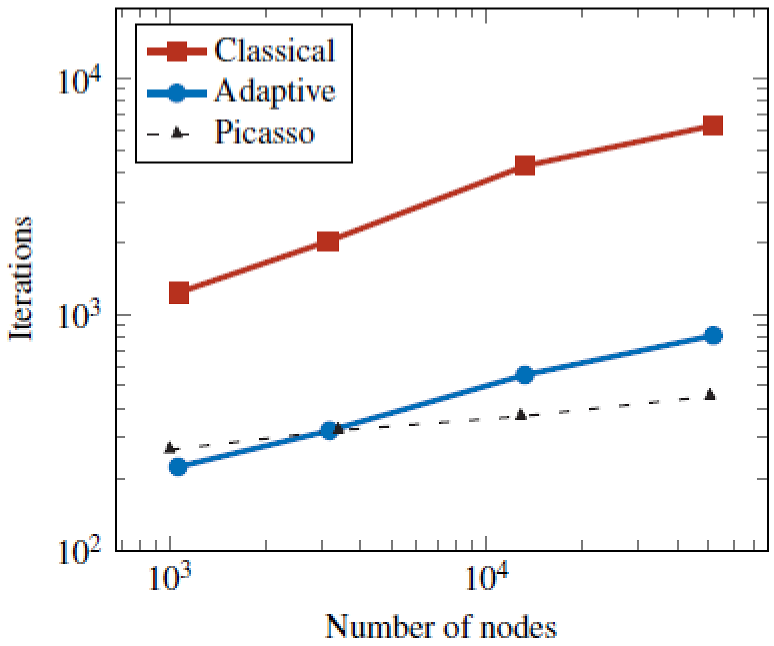
We compare the **total number** of linear **iterations** needed for the solution of all the steps

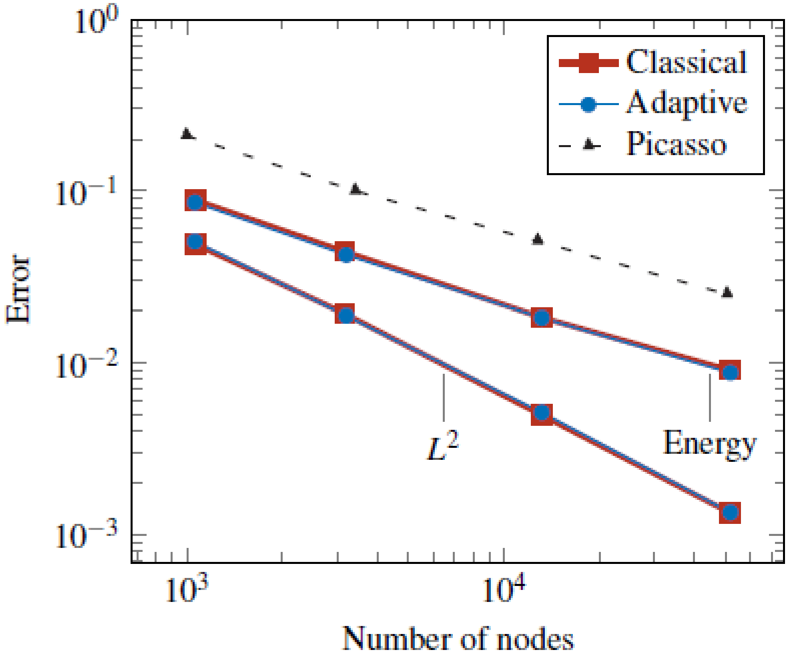
- Using a **classical** stopping criterion
- Using the proposed **adaptive** stopping criterion

Steady Problems – Laplace problem[3,4]



$$\begin{cases} -\Delta u = f, & \text{in } \Omega\,, \\ u = 0, & \text{in } \Gamma\,. \end{cases}$$

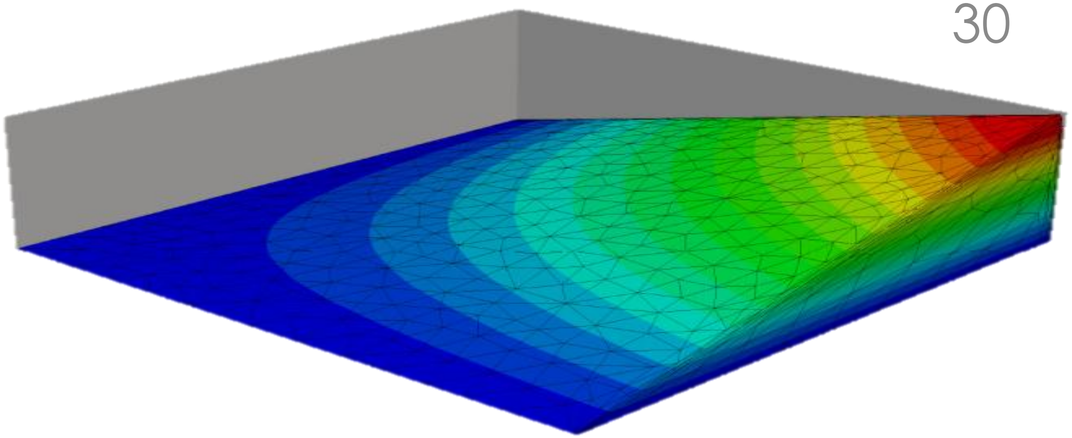




[3] Marco Picasso. A stopping criterion for the conjugate gradient algorithm in the framework of anisotropic adaptive finite elements. International Journal for Numerical Methods in Biomedical Engineering, 25(4):339–355, 2009

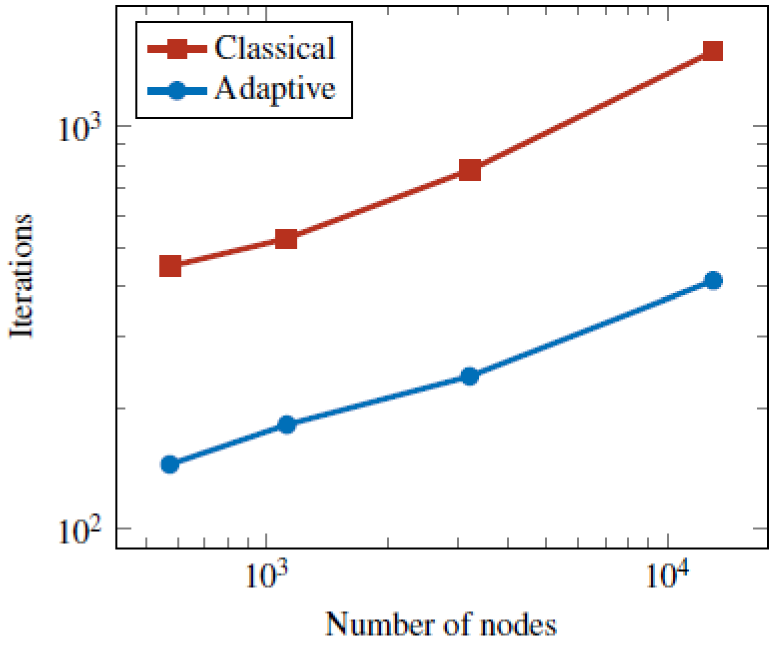
[4] Luca Formaggia and Simona Perotto. Anisotropic error estimates for elliptic problems. Numerische Mathematik, 94(1):67–92, 2003

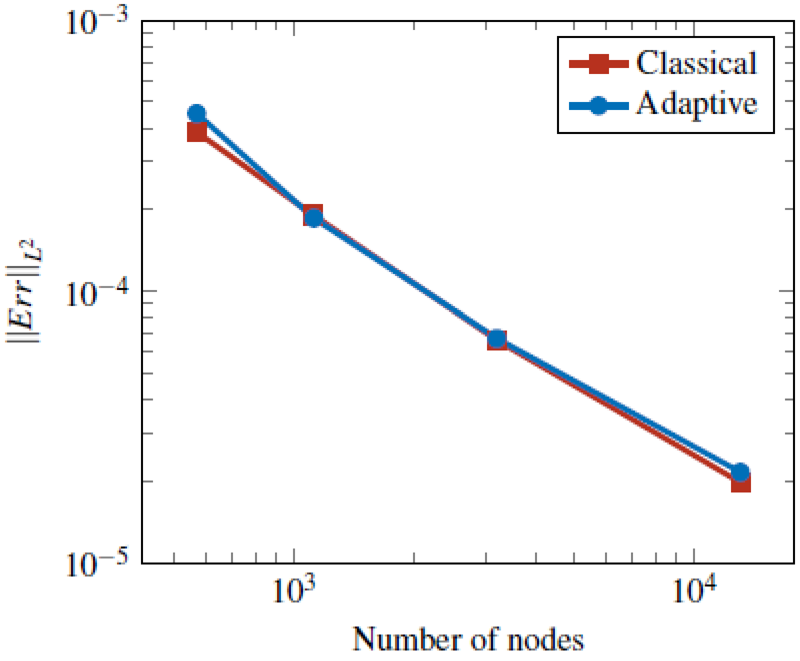
Steady Problems – Convection-diffusion



Problem equations

$$\begin{cases} -\nabla \cdot (a\nabla u) + \mathbf{v} \cdot \nabla u = f, & \text{in } \Omega \\ u = g, & \text{in } \Gamma \end{cases}$$



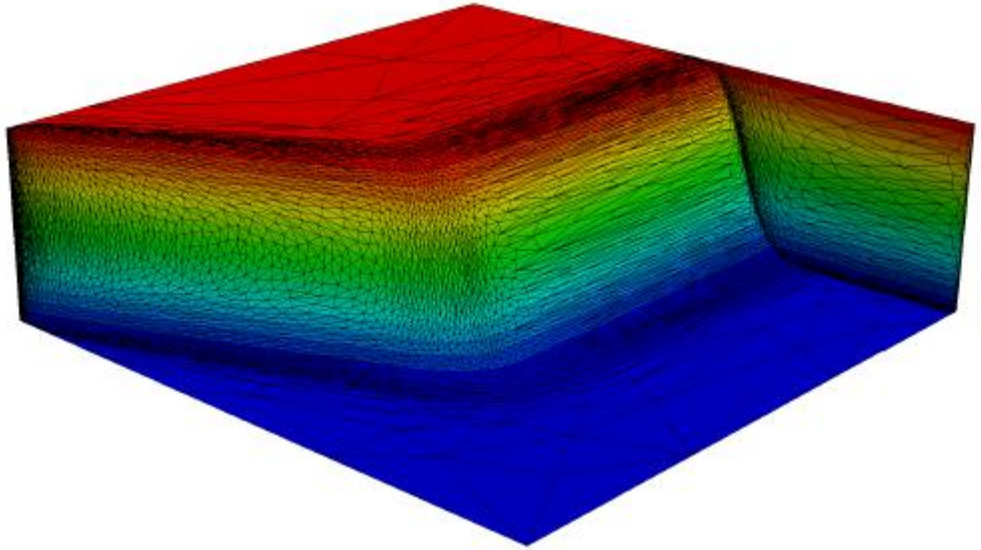


G. Manzinali, E. Hachem, Y. Mesri, Adaptive stopping criterion for iterative linear solvers combined with anisotropic mesh adaptation, application to convection-dominated problems,

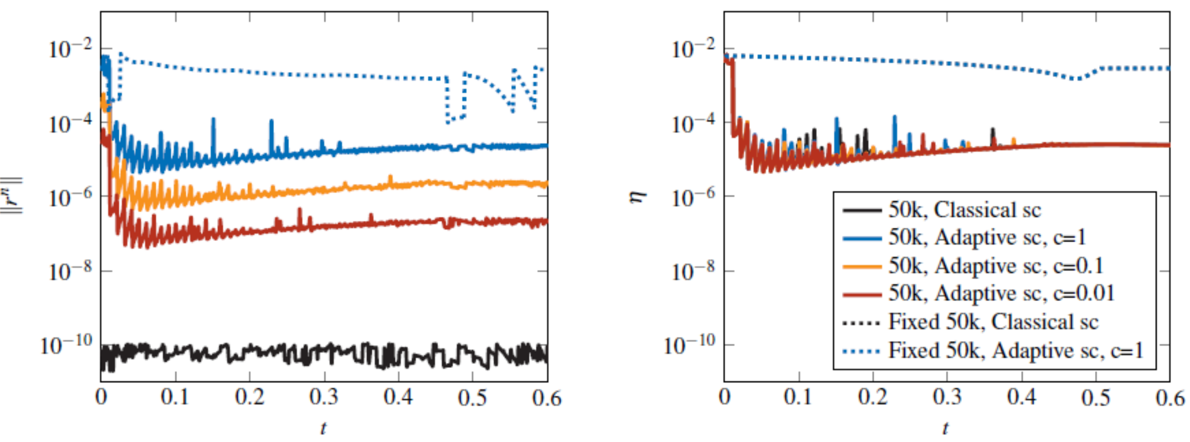
Unsteady Problems – Convection-diffusion

Problem equations

$$\begin{cases} \partial_t u(x,t) + \mathbf{v}\cdot\nabla u(x,t) - \nabla\cdot(k\nabla u(x,t)) = f & \text{in } \Omega \\ u(.,0) = u_0 & \text{in } \Omega \\ u(x,t) = g & \text{in } \Gamma \end{cases}$$



(a) Final residual. (b) Estimated error.



Stopping criterion	Total Iterations	Ratio	CPU time Solver	CPU time reduction Solver	CPU time simulation
ssical	13633	/	158s	/	803s
e "c=0.01"	7245	1.9	91s	-42%	710s
ve "c=0.1"	5615	2.4	70s	-56%	688s
ve "c=1"	3882	**3.5**	53s	**-66%**	680s

$$\|\nabla(u_h - u_h^n)\|_{L^2} \leqslant c \left(\sum_{T\in\mathcal{T}_h} \eta_T^2 \right)^{1/2}$$

G. Manzinali, E. Hachem, Y. Mesri, Adaptive stopping criterion for iterative linear solvers combined with anisotropic mesh adaptation, application to convection-dominated problems,

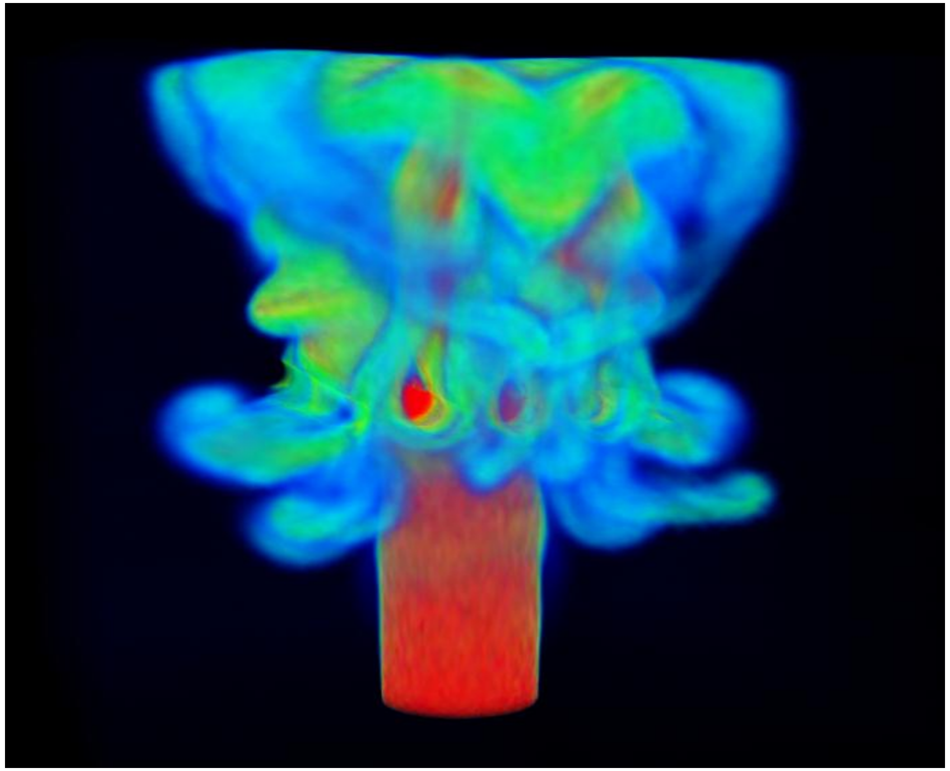
Unsteady Problems – Navier-Stokes + Thermal

$$\|\nabla(u_h - u_h^n)\|_{L^2} \leqslant c \left(\sum_{T \in \mathcal{T}_h} \eta_T^2 \right)^{1/2}$$

Problem equations

$$\begin{cases} \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \\ \rho(\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u}) - \nabla \cdot (2\mu\, \varepsilon(\mathbf{u}) - p\, \mathbf{I_d}) = \rho_0 \beta (T - T_0)\, \mathbf{g} & \text{in } \Omega \\ \rho C_p(\partial_t T + \mathbf{u} \cdot \nabla T) - \nabla \cdot (\lambda \nabla T) = f & \text{in } \Omega \end{cases}$$

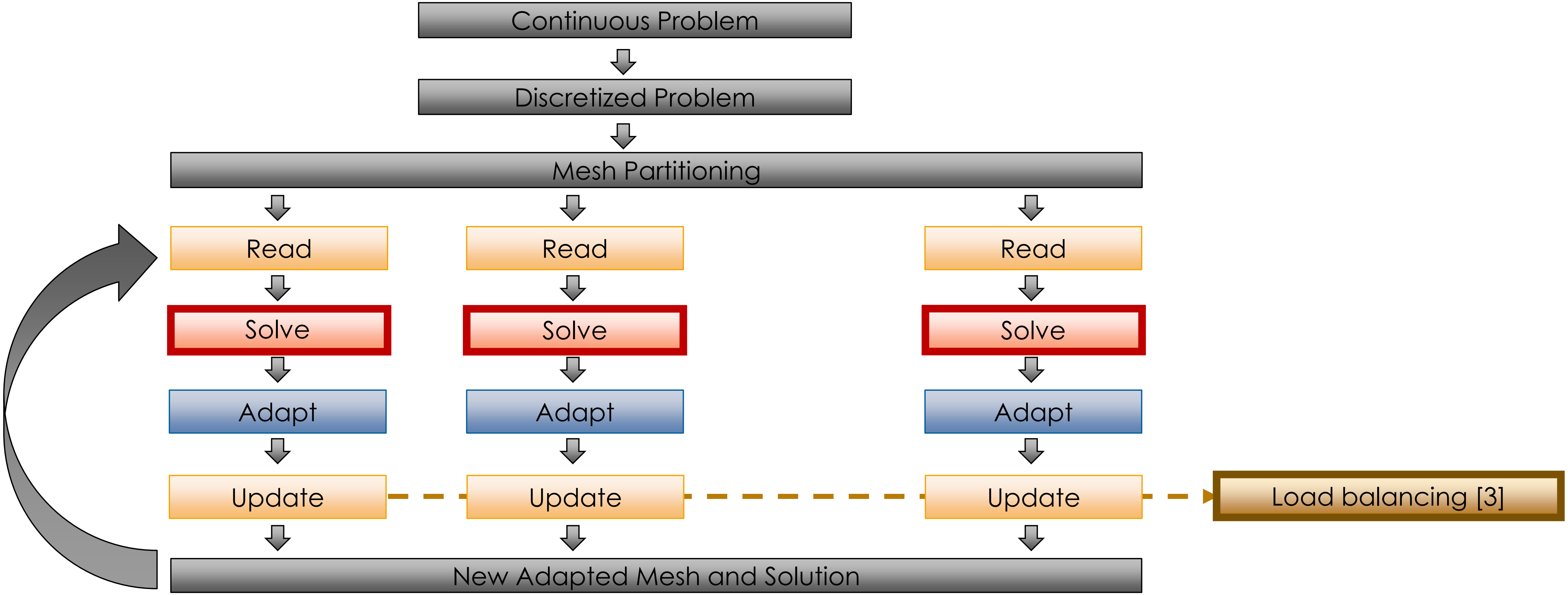
→ **~ 40% CPU time saving**



Stopping criterion	Total Iterations	Ratio	CPU time Solver	CPU time reduction Solver	CPU time simulation
Classical	609922	/	37h	/	41h
Adaptive "c=0.1"	107846	**5,7**	23h	**-38%**	28h

AFEM Framework – Load balancing

AFEM Framework – Load balancing



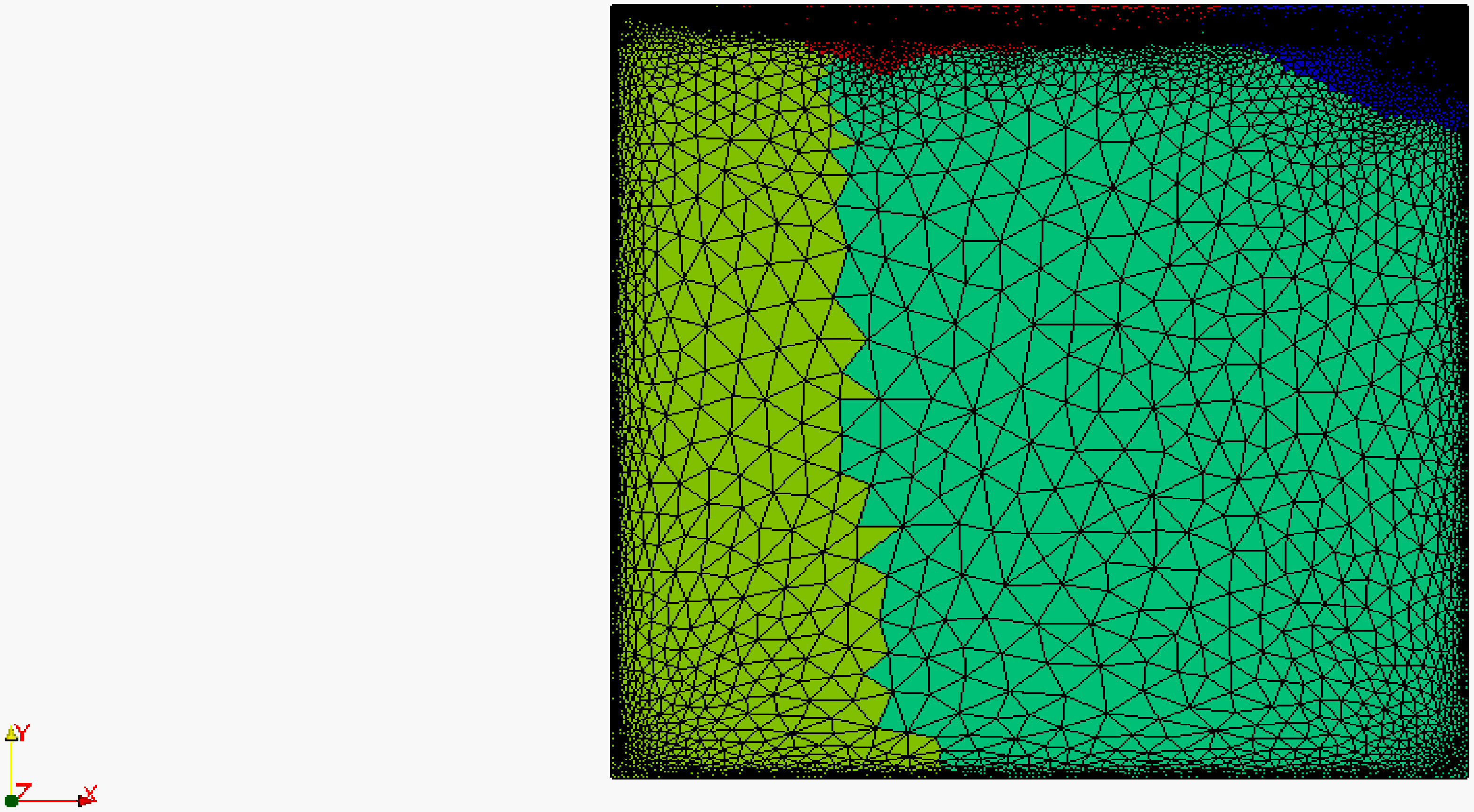
[1] G. Manzinali, E. Hachem, Y. Mesri, Adaptive stopping criterion for iterative linear solvers combined with anisotropic mesh adaptation, application to convection-dominated problems, CMAME, 2018

[2] Bazile, A., Hachem, E., Larroya-Huguet, J. C., Mesri, Y. Variational Multiscale error estimator for anisotropic adaptive fluid mechanic simulations: Application to convection–diffusion problems. CMAME, *331*, 94-115, 2018.

[3] Y. Mesri, Predictive load balancing for parallel anisotropic mesh adaptation applications, preprint, 2018

Dynamic load balancing (lid-driven cavity)

Load balancing based on local topological mesh repartitioning



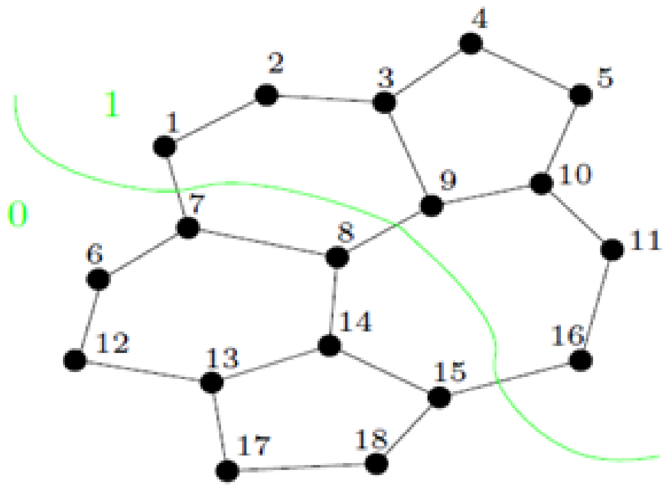
Estimate the expected dynamic workload

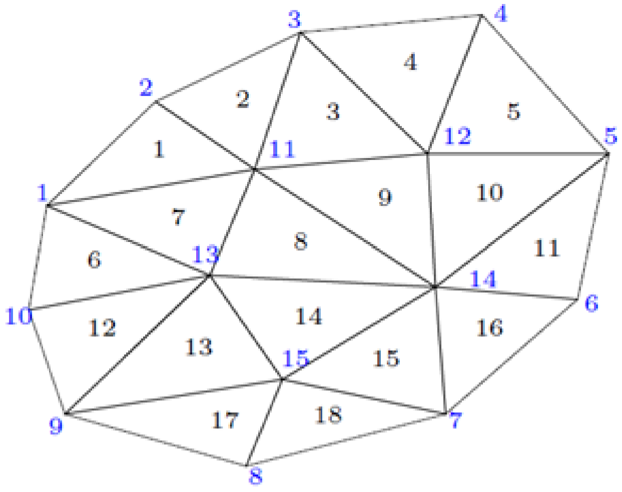
Error Estimator η_T → Metric $M(T)$ → Mesh Quality in Metric space

$$q(T) = C_0 \frac{(Volume)(T)_{M(T)}}{h^d_{M(T)}}$$

↓

Weights the mesh/graph entities for repartitioning





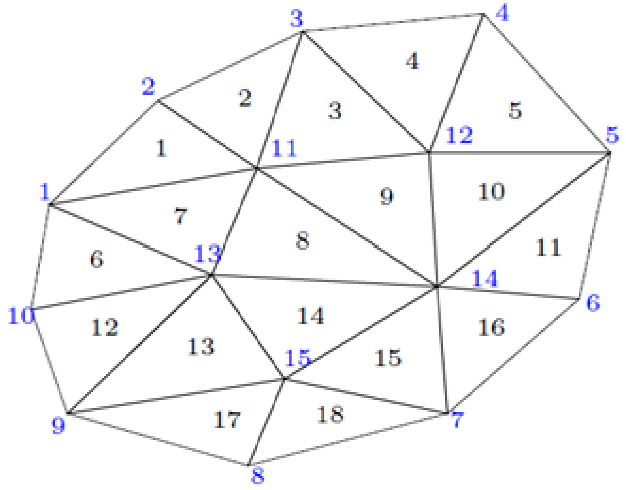
Predictive load balancing

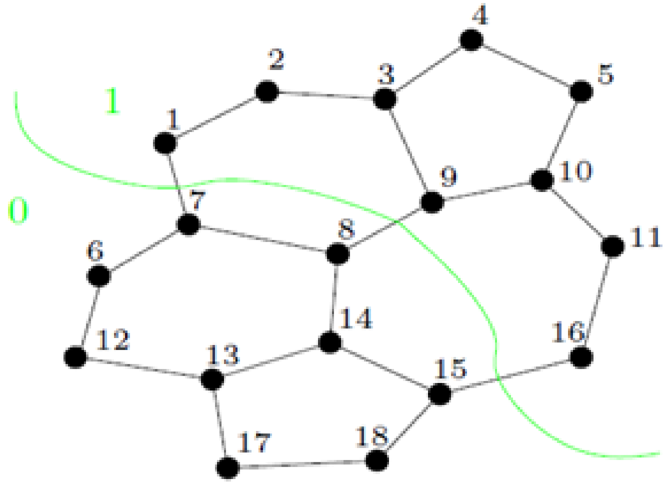
1. $$q(T) = C_0 \frac{(Volume)(T)_{M(T)}}{h^d_{M(T)}}$$

2. $$w(v) = \frac{1}{q(T)}$$

3. $$w(u,v) = \frac{1}{2q(S)} + \frac{1}{2q(T)}$$

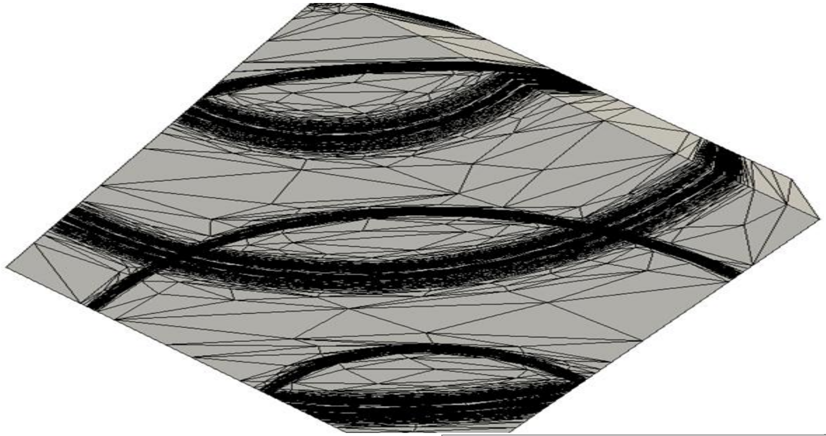
4. $$F(W, A, m) := max_{p \in V(A)}(t_p) + max_{p \in V(A))}(c_p)$$

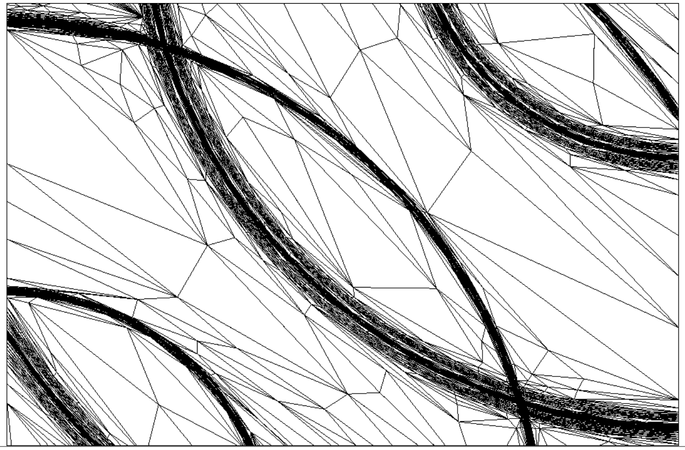


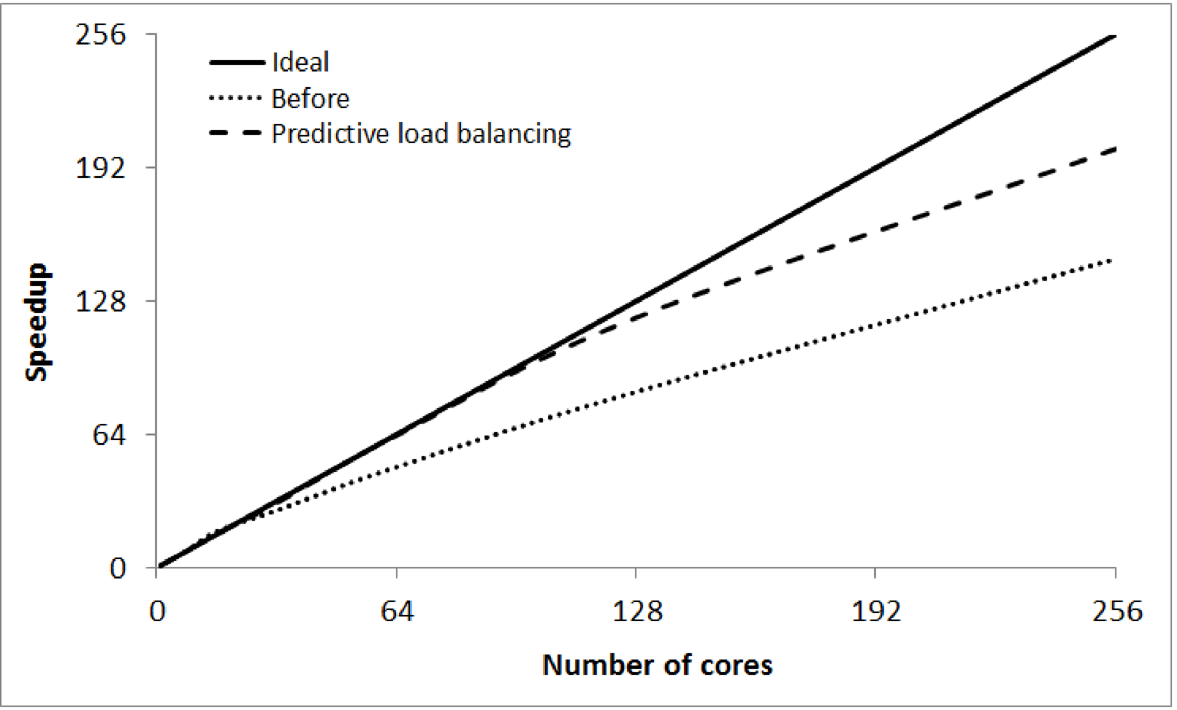


Predictive load balancing

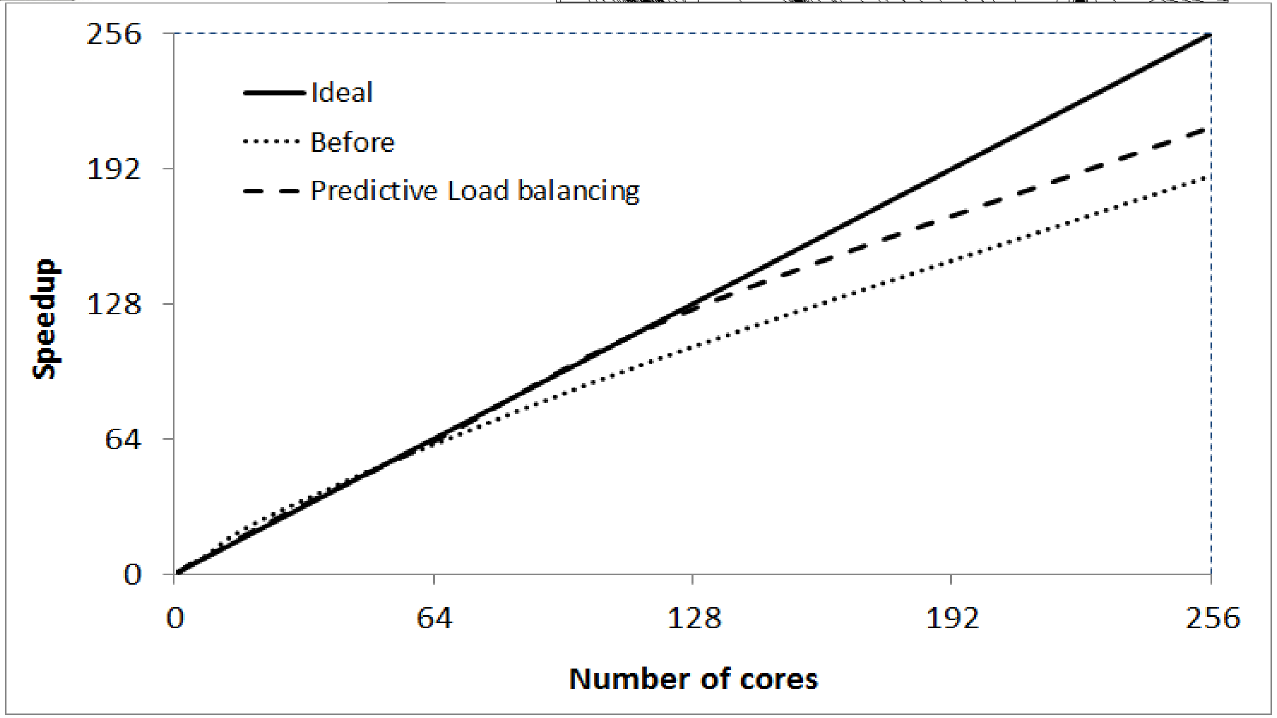
Y. Mesri, Parallel High-Reynolds Incompressible Flow with Adaptive Anistrotropic Meshing, submitted





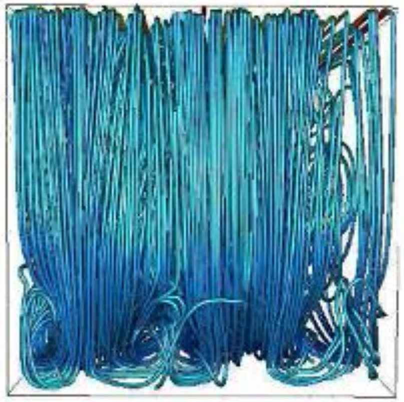


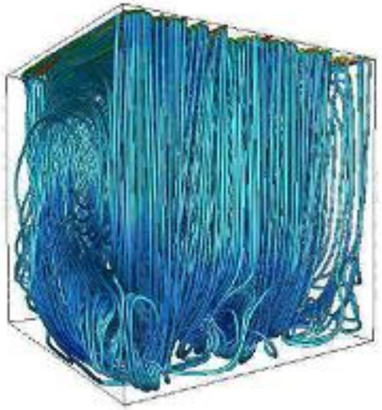
3D test case



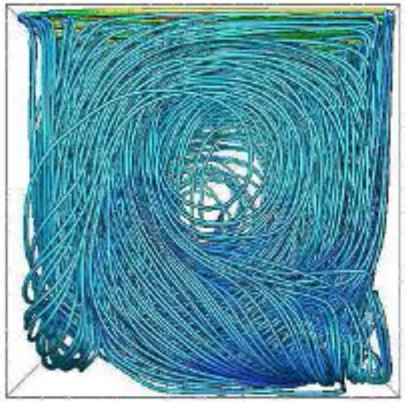
2D test case

Parallel NS with predictive load balancing

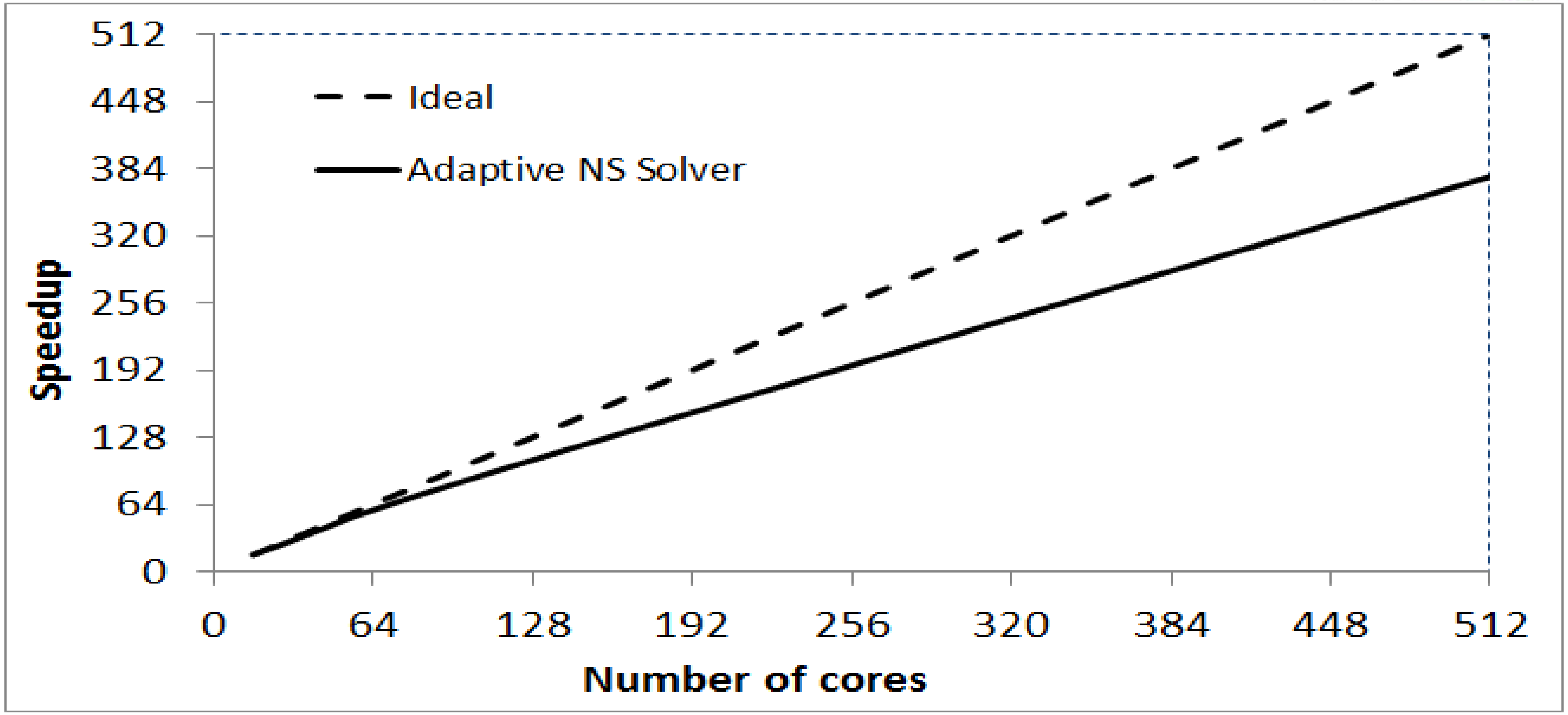




3D lid-driven cavity



30M elements
NS Incompressible
Re = 3200



Y. Mesri, Parallel High-Reynolds Incompressible Flow with Adaptive Anistrotropic Meshing, submitted

Potential Collaboration topics?

Solve over 100 billion unknowns system

Methods

- Numerical Methods and their implementation
- Experimental prototyping and calibration
- Data analytics

Technology

- Cloud computing
- HPC and big data processing
- Artificial Intelligence/DL/ML

Applications

- Muti-phase flows
- Fluid Structure Interaction
- Heat transfer

Targets new applications

- Digital Health
- Mobility
- …

- Système d'équations aux dérivées partielles

$$A(u) = f$$

- Méthode numérique

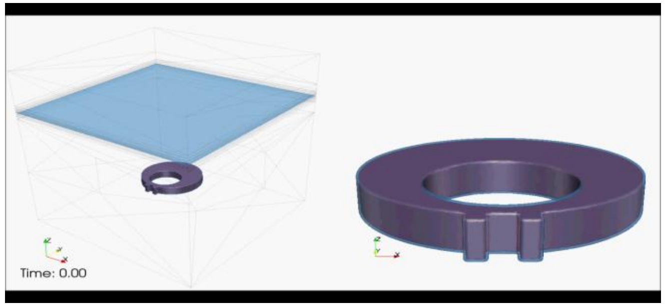
$$A_h(u_h) = f_h$$

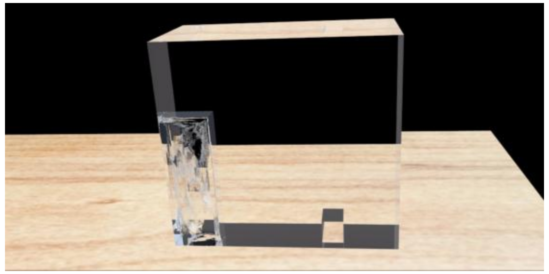
- Résolution itérative

$$A^{k-1}U^{k,i} = F^{k-1} - R^{k-1,i}, \quad (3)$$

le vecteur $U^{k,i}$ est une représentation algébrique de u_h à l'itération non-linéaire k et linéaire i

Over 262 144 JuQUEEN cores





THANK YOU FOR YOUR ATTENTION