Optimisation topologique de formes pour des problèmes multi-physiques avec maillage exact des formes et interfaces

Grégoire Allaire (CMAP, École Polytechnique)
Charles Dapogny (LJK, CNRS and Grenoble University)
Florian Feppon (KU Leuven)
Pierre Jolivet (IRIT, CNRS, Toulouse)

Séminaire SISMA, Bruyères-le-Châtel, 31 Mars 2023

Outline

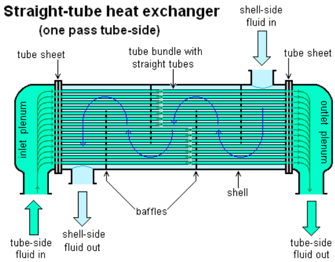
- I - Motivation and examples
- II - Fluid-to-fluid heat exchanger
- III - Algorithmic details
- IV - Numerical results
- V - Fluid-structure interaction
- VI - Conclusion and perspectives

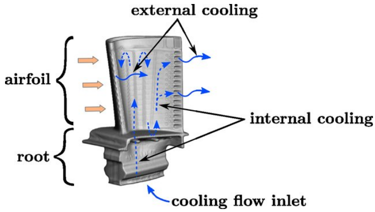
For details and other examples, see:

F. Feppon, G. Allaire, C. Dapogny, P. Jolivet, *Topology optimization of thermal fluid-structure systems using body-fitted meshes and parallel computing*, J. Comp. Phys., 417 (2020).

F. Feppon, G. Allaire, C. Dapogny, P. Jolivet, *Body-fitted topology optimization of 2D and 3D fluid-to-fluid heat exchangers*, CMAME, 376, 113638 (2021).

Heat exchangers and cooling systems.





heat exchanger

turbine blade with internal cooling

Motivation: multi-physics optimization

Two examples:

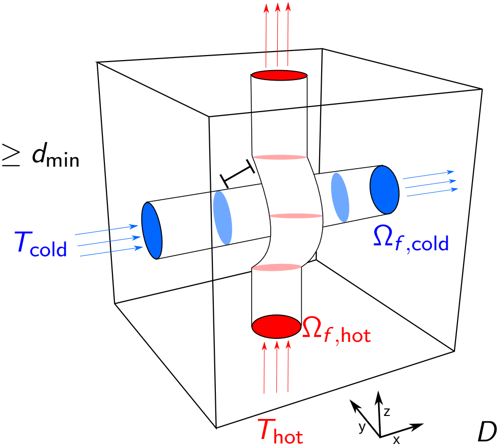
1. Fluid-to-fluid heat exchangers (coupling flow and heat equation).
2. Fluid-solid interaction.

(Other examples: fluid-thermal-solid, thermo-mechanics in additive manufacturing.)

Mathematical and numerical issues:

1. Interface optimization rather than boundary optimization.
2. Adjoint system and optimization algorithm.

$d(\Omega_{f,\mathrm{hot}}, \Omega_{f,\mathrm{cold}}) \geq d_{\min}$



A non-mixing constraint for the two fluids is imposed with a minimal distance $d_{\min}$.

Model and goal

Physical models:

- Incompressible Navier-Stokes equations for the two fluids.
- Steady-state heat equation in the fluid (including convection) and in the surrounding solid.

Objective functions and constraints:

- Maximize the heat exchange between the two fluids.
- Constraint on the pressure drop in the two fluid channels.
- Constraint on the volumes of the fluid channels.
- Minimal distance between the hot and cold fluids to guarantee non-mixing.
- Allow for topology changes.
- Track exactly the fluid-solid interface.

Navier-Stokes equations

Two fluids but one single set of equations (because of the non-mixing condition.

The fluid domain is $\Omega_f = \Omega_{f,\text{cold}} \cup \Omega_{f,\text{hot}}$, the velocity is $\boldsymbol{v}$ and pressure p. The inlet velocity is $\boldsymbol{v}_0$.

$$\begin{cases} -\operatorname{div}(\sigma_f(\boldsymbol{v}, p)) + \rho \boldsymbol{v} \cdot \nabla \boldsymbol{v} = 0 & \text{in } \Omega_f \\ \operatorname{div}(\boldsymbol{v}) = 0 & \text{in } \Omega_f \\ \boldsymbol{v} = \boldsymbol{v}_0 & \text{on } \partial\Omega_f^D \text{ (inlet)} \\ \sigma_f(\boldsymbol{v}, p)\boldsymbol{n} = 0 & \text{on } \partial\Omega_f^N \text{ (outlet)} \\ \boldsymbol{v} = 0 & \text{on } \Gamma \text{ (interface)} \end{cases}$$

with the viscous stress $\sigma_f(\boldsymbol{v}, p) = 2\nu e(\boldsymbol{v}) - pI$. There is a no-slip boundary condition on the channel walls Γ.

Fixed domain $D = \Omega_f \cup \Omega_s$ with interface $\Gamma = \partial\Omega_f \cap \partial\Omega_s$.

Heat equation

The temperature field T is T_s in the solid domain Ω_s and T_f in the fluid domain Ω_f

$$\begin{cases} -\operatorname{div}(k_f \nabla T_f) + \rho c_p \boldsymbol{v} \cdot \nabla T_f = 0 & \text{in } \Omega_f \\ -\operatorname{div}(k_s \nabla T_s) = 0 & \text{in } \Omega_s \\ T = 100 & \text{on } \partial\Omega_f^D \cap \partial\Omega_{f,\text{hot}} \\ T = 0 & \text{on } \partial\Omega_f^D \cap \partial\Omega_{f,\text{cold}} \\ -k_f \dfrac{\partial T_f}{\partial \boldsymbol{n}} = 0 & \text{on } \partial\Omega^N \cap \partial\Omega_f \\ -k_s \dfrac{\partial T_s}{\partial \boldsymbol{n}} = 0 & \text{on } \partial\Omega^N \cap \partial\Omega_s \\ T_f = T_s & \text{on } \Gamma \\ -k_f \dfrac{\partial T_f}{\partial \boldsymbol{n}} = -k_s \dfrac{\partial T_s}{\partial \boldsymbol{n}} & \text{on } \Gamma, \end{cases}$$

where Γ is the fluid-solid interface.

Optimization problem

Minimize the difference of outgoing heat fluxes for the two fluids

$$\min_{\Omega_f \subset D} J(\Omega_f) = -\left(\int_{\Omega_{f,\text{cold}}} \rho c_p \boldsymbol{v} \cdot \nabla T \mathrm{d}x - \int_{\Omega_{f,\text{hot}}} \rho c_p \boldsymbol{v} \cdot \nabla T \mathrm{d}x \right)$$

with the following constraints:

$$\mathtt{DP}(\Omega_{f,\text{hot}}) = \int_{\partial\Omega^D_{f,\text{hot}}} p\mathrm{d}s - \int_{\partial\Omega^N_{f,\text{hot}}} p\mathrm{d}s \leq \mathtt{DP}_0$$

$$\mathtt{DP}(\Omega_{f,\text{cold}}) = \int_{\partial\Omega^D_{f,\text{cold}}} p\mathrm{d}s - \int_{\partial\Omega^N_{f,\text{cold}}} p\mathrm{d}s \leq \mathtt{DP}_0$$

$$\mathrm{Vol}(\Omega_{f,\text{hot}}) \leq \mathtt{V}_0 \qquad \mathrm{Vol}(\Omega_{f,\text{cold}}) \leq \mathtt{V}_0$$

$$P_{\text{hot}\to\text{cold}}(\Omega_f) \geq d_{\min} \qquad P_{\text{cold}\to\text{hot}}(\Omega_f) \geq d_{\min}$$

with $P_{\text{cold}\to\text{hot}}(\Omega_f) = \left(\int_{\partial\Omega_{f,\text{cold}}} \frac{1}{|d_{\Omega_{f,\text{hot}}}|^4} \mathrm{d}s \right)^{-\frac{1}{4}}$ and $d_{\Omega_{f,\text{hot}}}$ the signed distance for the hot fluid.

Objective function

$$J(\Omega_f) = -\left(\int_{\Omega_{f,\text{cold}}} \rho c_p \boldsymbol{v} \cdot \nabla T \mathrm{d}x - \int_{\Omega_{f,\text{hot}}} \rho c_p \boldsymbol{v} \cdot \nabla T \mathrm{d}x\right)$$

By integration by parts, and since $\boldsymbol{v} = 0$ on the walls,

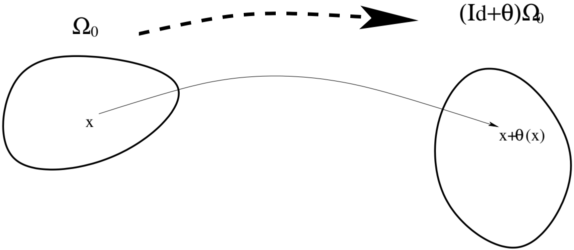
$$\int_{\Omega_{f,\text{cold}}} \rho c_p \boldsymbol{v} \cdot \nabla T \mathrm{d}x = \int_{\text{cold outlet}} \rho c_p \boldsymbol{v} \cdot n_{\text{out}} T \mathrm{d}s - \underbrace{\int_{\text{cold intlet}} \rho c_p \boldsymbol{v} \cdot n_{\text{in}} T \mathrm{d}s}_{\text{given}}$$

$$-\int_{\Omega_{f,\text{hot}}} \rho c_p \boldsymbol{v} \cdot \nabla T \mathrm{d}x = \underbrace{\int_{\text{hot inlet}} \rho c_p \boldsymbol{v} \cdot n_{\text{in}} T \mathrm{d}s}_{\text{given}} - \int_{\text{hot outlet}} \rho c_p \boldsymbol{v} \cdot n_{\text{out}} T \mathrm{d}s$$

Minimizing J amounts to maximize the heat extracted by the cold fluid plus the heat lost by the hot fluid (which are the same).

Shape derivative

To compute the gradient of $J(\Omega_f)$ and of the constraints, we rely on Hadamard's method.



Let Ω_0 be a reference domain. Shapes are parametrized by a **vector field** θ

$$\Omega = (\mathrm{Id} + \theta)\Omega_0 \quad \text{with} \quad \theta \in C^1(\mathbb{R}^d; \mathbb{R}^d).$$

Definition: the shape derivative of $J(\Omega)$ at Ω_0 is the Fréchet differential of $\theta \to J\big((\mathrm{Id} + \theta)\Omega_0\big)$ at 0 in $C^1(\mathbb{R}^d;\mathbb{R}^d)$.

Hadamard structure theorem: the shape derivative of $J(\Omega)$ can always be written (in a distributional sense)

$$J'(\Omega_0)(\theta) = \int_{\partial\Omega_0} \theta(x) \cdot n(x)\, j(x)\, ds$$

where $j(x)$ is an integrand depending on the state (solution of the pde's) and an adjoint.

Remark. The adjoint equation is a linear pde which is classical for single physics. It is more delicate for multi-physics: different coupling order, different interface transmission conditions...

Actually, it is an interface Γ, rather than an exterior boundary $\partial\Omega$, which is optimized.

- Additional mathematical difficulties !
- Transmission conditions at the interface must be properly differentiated.
- Many errors in the literature...
- Correct results for an interface between two phases:
  - heat equation: Hettlich-Rundell (98), Pantz (05),
  - elasticity system: A.-Jouve-Van Goethem (11).
- Adjoints are coupled in the reverse order of states. (Not standard in commercial codes.)

A simple example for the heat equation with $\boldsymbol{v} = 0$.

Lemma. The shape derivative with respect to the interface Γ of

$$J(\Omega_f) = \int_D j(T)dx$$

is $\quad J'(\Omega_f)(\theta) = \int_\Gamma \mathcal{D}(x)\,\theta \cdot n\,ds \quad$ with

$$\mathcal{D}(x) = (k_s - k_f)\nabla_t T \cdot \nabla_t T_{\text{adj}} - \left(\frac{1}{k_s} - \frac{1}{k_f}\right)\left(k\frac{\partial T}{\partial n}\right)\left(k\frac{\partial T_{\text{adj}}}{\partial n}\right)$$

where ∇_t is the tangential gradient and T_{adj} is the adjoint state

$$\begin{cases} -\operatorname{div}(k\nabla T_{\text{adj}}) = -j'(T) & \text{in} \quad D, \\ T_{\text{adj}} = 0 & \text{on} \quad \Gamma_D, \\ k\nabla T_{\text{adj}} \cdot n = 0 & \text{on} \quad \Gamma_N. \end{cases}$$

Shape derivatives

- We compute the shape derivatives of the objective function J and of each constraint.
- One adjoint for the objective and one for each pressure drop constraint.
- The adjoint is a system of linear equations.
- The coupling of the adjoint is reversed: first, solve for T_{adj}, second, solve for $\boldsymbol{v}_{\mathrm{adj}}$.
- Shape derivatives are carried by the interface, so is is convenient to mesh it exactly.
- Formulas are ugly... See the papers for details !

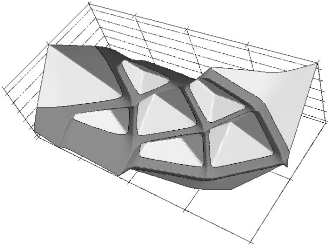
Remark. If one relies on a different topological optimization method (say SIMP or density-based algorithms), the adjoints are the same but the derivatives may be simpler.

III - Algorithmic details

- Finite element computations with the parallel version of `FreeFem++`:
  https://freefem.org/
- Topology optimization with the level set method.
- Body-fitted mesh at each iteration thanks to `Mmg3d`:
  https://www.mmgtools.org/
- Small isolated fluid components (because of topological changes) are detected and removed at each iteration.
- Optimization algorithm: null space gradient.
- Shape sensitivities: adjoint method in the Hadamard framework.
- Non-mixing constraint for the two fluids computed with signed distance functions.

Level set method (Osher and Sethian)

General method with many applications !



A shape $\Omega \subset D$ is parametrized by a **level set** function

$$\psi(x) < 0 \Leftrightarrow x \in \Omega, \ \ \psi(x) > 0 \Leftrightarrow x \in (D \setminus \Omega)$$

Assume that the shape $\Omega(t)$ evolves in time t with a normal velocity $V(t,x)$. Then its motion is governed by the following Hamilton Jacobi equation

$$\frac{\partial \psi}{\partial t} + V |\nabla_x \psi| = 0 \quad \text{ in } D.$$

Advection velocity = shape gradient

The normal velocity V is deduced from the shape gradient of the objective function

$$J'(\Omega_f)(\theta) = \int_{\partial\Omega_f} \theta(x) \cdot n(x)\, j(x)\, ds$$

such that $J'(\Omega_f)(\theta) \leq 0$. For example, $V = \theta \cdot n = -j$.

A better choice is to use an extension-regularization equation. Solve the variational formulation for $V \in H^1(D)$

$$\int_D \left(\varepsilon^2 \nabla V \cdot \nabla \varphi + V\varphi\right) dx = -\int_{\partial\Omega_f} \varphi\, j\, ds \quad \forall\, \varphi \in H^1(D),$$

where ε is of the order of a mesh cell. Then $\theta \cdot n = V$.

Remark. It works fine even when $J'(\Omega_f)(\theta)$ is not written as a surface integral.

Optimization algorithm

When there are N constraints $C(\Omega)$, one could rely on a Lagrangian

$$\mathcal{L}(\Omega, \lambda) = J(\Omega) + \lambda \cdot C(\Omega)$$

where $\lambda \in \mathbb{R}^N$ is a Lagrange multiplier and use an Uzawa-type algorithm. But convergence is slow...

Better optimization algorithm: null-space gradient.

- It is an implicit algorithm (for λ) based on a linearization of $J(\Omega)$ and $C(\Omega)$.
- It is a first-order gradient algorithm which provides exponential convergence to the admissible domain.

Null-space gradient algorithm

For simplicity, consider a Hilbert space $\mathcal{H}$ and assume only equality constraints $C : \mathcal{H} \mapsto \mathbb{R}^N$

$$\min_{x \in \mathcal{H}, s.t. C(x)=0} J(x)$$

Assume $\mathrm{rank} \nabla C(x) = N$. The algorithm reads

$$x^{n+1} = x^n - \delta_J \xi_J^n - \delta_C \xi_C^n$$

where $\delta_J, \delta_C > 0$ are descent steps and

$$\xi_J = \left(\mathrm{Id} - (\nabla C)^T \mathcal{M} \nabla C \right) \nabla J$$

$$\xi_C = (\nabla C)^T \mathcal{M} C$$

where $\mathcal{M} = \left((\nabla C)(\nabla C)^T \right)^{-1}$ is a $N \times N$ matrix and ξ_J is the orthogonal projection on the tangent hyperplane to $C(x) = 0$.

Null-space gradient algorithm (ctd.)

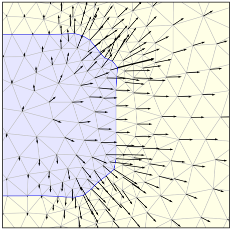
- For shape optimization, x becomes θ and $\mathcal{H}$ becomes $H^1(D;\mathbb{R}^d)$ (instead of $C^1(D;\mathbb{R}^d)$, slight cheating...).
- Well suited to the extension-regularization process.
- For inequality constraints, consider only active inequalities.
- Further refinement: among active inequalities, consider only those which will be violated by moving along ∇J (need a dual problem to find them).

F. Feppon, G. Allaire, C. Dapogny, *Null space gradient flows for constrained optimization with applications to shape optimization*, COCV, 26, (2020).

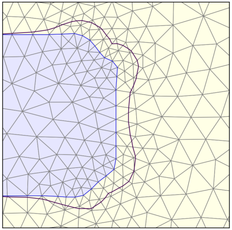
Numerical algorithm

1. Initialization of the level set function ψ_0 (including holes).
2. Iteration until convergence for $k \geq 1$:
    1. Compute the state $(\boldsymbol{v}^k, T^k)$ and adjoint $(\boldsymbol{v}^k_{\mathrm{adj}}, T^k_{\mathrm{adj}})$ for the shape ψ_k.
       Deduce the shape gradient $=$ normal velocity $= V_k$
    2. Advect the level set with V_k (solving the Hamilton Jacobi equation) to obtain a new shape ψ_{k+1}.
    3. Remesh the new shape.

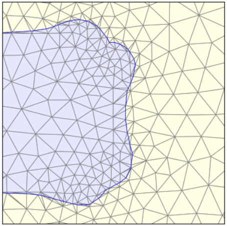
Exact remeshing with Mmg



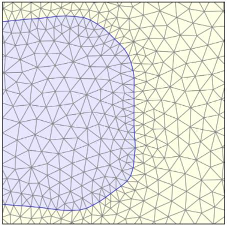
Initial interface



Zero-level set after advection



Cut mesh (bad quality)



Adapted mesh

https://www.mmgtools.org/

Charles Dapogny, Cécile Dobrzynski†, Pascal Frey, Algiane Froehly

Numerical values

Non-dimensional values.
Domain $D = (0,1)^3$, inlet radius $a = 0.1$

Re	$d_{\min}$	ρ	k_f	k_s	Pe	c_p	$\mathtt{DP}_0$	$\mathtt{V}_0$	T_{hot}	T_{cold}
100	0.1	10	1	10	2,000	200	3.72	0.15	100	0

Initial mesh: 3.8 million tetrahedra (2.3 10^6 in the fluid domain).
Final mesh: 1.7 million tetrahedra (686,000 in the fluid domain).
Optimization: 360 iterations.

Initialization

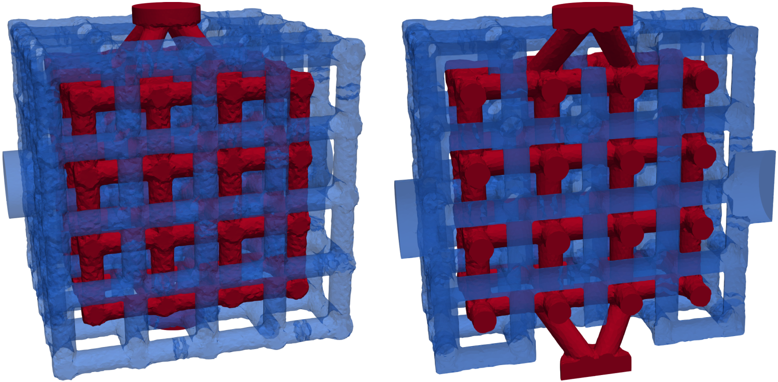


Figure: Initial distribution of fluid considered for the 3D heat exchanger. The hot and cold phase are depicted in red and blue respectively and are disjoint regions. Cut with respect to the z-axis on the right.

Optimal design (fluid parts)

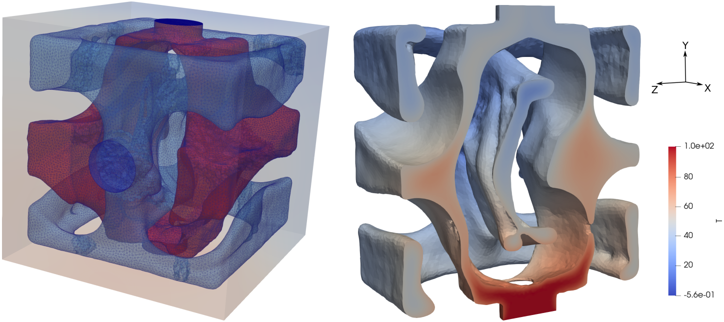


Figure: Optimized channels of the cold and hot fluids, respectively colored in blue and red (left), temperature distribution (right).

Optimal design (solid part)

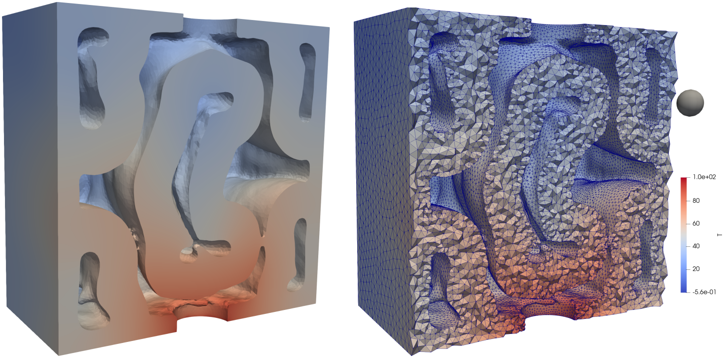


Figure: Temperature field in the solid, respecting the constraint on the minimum wall thickness (indicated by the small ball on the right).

Iteration history (including topology changes)

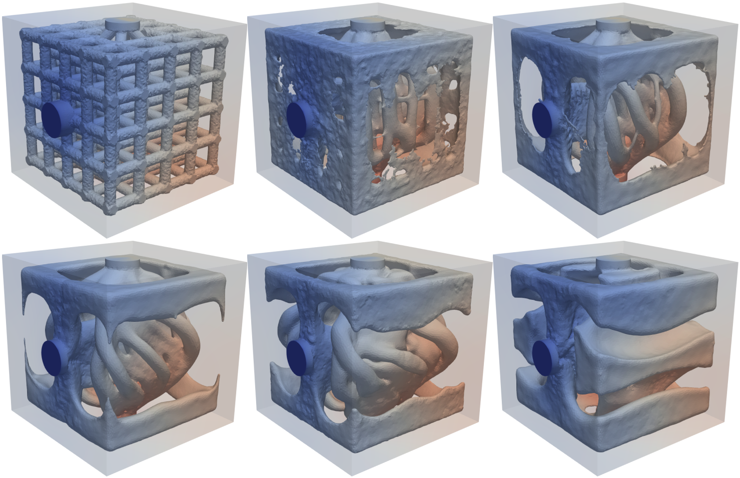


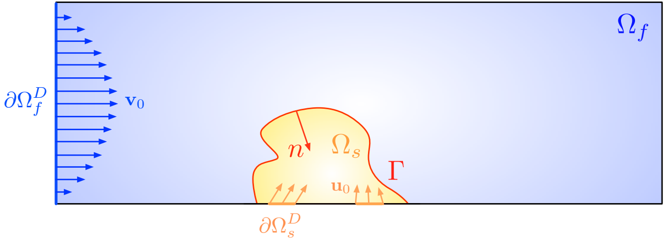
Figure: Iterations 0, 25, 50, 110, 180, and 360 of the optimization.

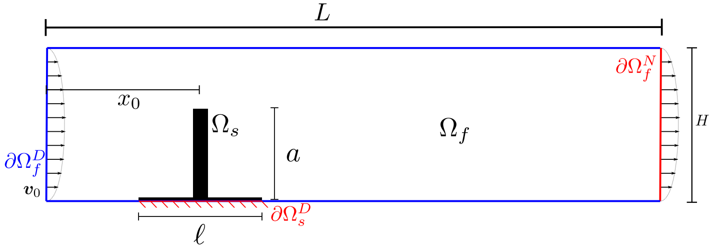
CPU time

Operation	Iteration 1	Iteration 20
Removal of isolated mesh fluid parts	41 sec	21 sec
Signed distance function to Ω_f	31 sec	21 sec
Signed distance function to $\Omega_{f,\text{hot}}$	14 sec	10 sec
Signed distance function to $\Omega_{f,\text{cold}}$	22 sec	17 sec
State equations (30 processes)	344 sec	253 sec
Shape sensitivities (30 processes)	340 sec	259 sec
Advection of the solid–fluid interface	11 sec	8 sec
Remeshing (still sequential)	696 sec	521 sec
Total	26 min	19 min

Table: Main computational times for iterations 1 and 20.

Total CPU time for 360 iterations: 8 days on 30 processing units of an Intel(R) Xeon(R) CPU E5-2407 @ 2.4 GHz





Re-inforcement of a vertical bar under an horizontal flow.

Fluid-structure optimization

- Navier-Stokes equations for the incompressible fluid.
- Linearized elasticity equations for the structure, subject to the fluid pressure load.
- Minimize the compliance of the solid structure.
- Weak coupling of the direct system: first solve Navier-Stokes, second solve elasticity.
- Reverse coupling for the adjoint: first solve the adjoint elasticity equations, second solve the adjoint Navier-Stokes equations.

F. Feppon, G. Allaire, F. Bordeu, J. Cortial, C. Dapogny, *Shape optimization of a coupled thermal fluid-structure problem in a level set mesh evolution framework*, SeMA Journal, 76(3), 413-458 (2019).

Model

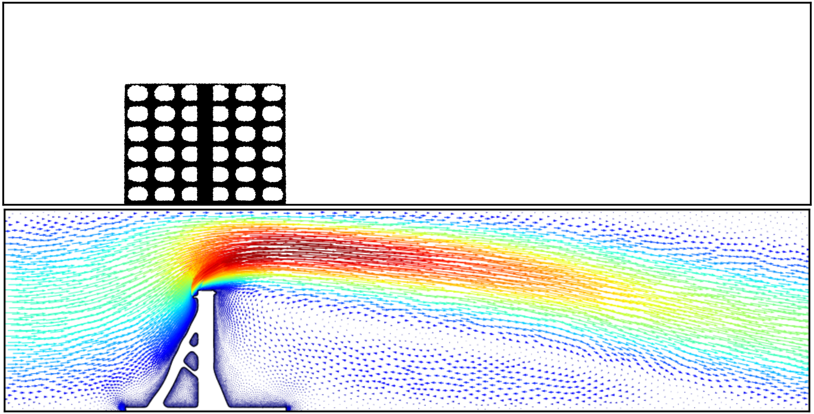
Navier-Stokes equations:

$$\begin{cases} -\operatorname{div}(\sigma_f(v,p)) + \rho v\cdot\nabla v = 0 & \text{in } \Omega_f \\ \operatorname{div}(v) = 0 & \text{in } \Omega_f \\ v = v_0 & \text{on } \partial\Omega_f^D \\ \sigma_f(v,p)\boldsymbol{n} = 0 & \text{on } \partial\Omega_f^N \\ v = 0 & \text{on } \Gamma, \end{cases}$$

where $\sigma_f(v,p) = 2\nu e(v) - pI$ with $e(v) = (\nabla v + \nabla v^T)/2$.
Fluid/solid interface Γ. Elasticity equations:

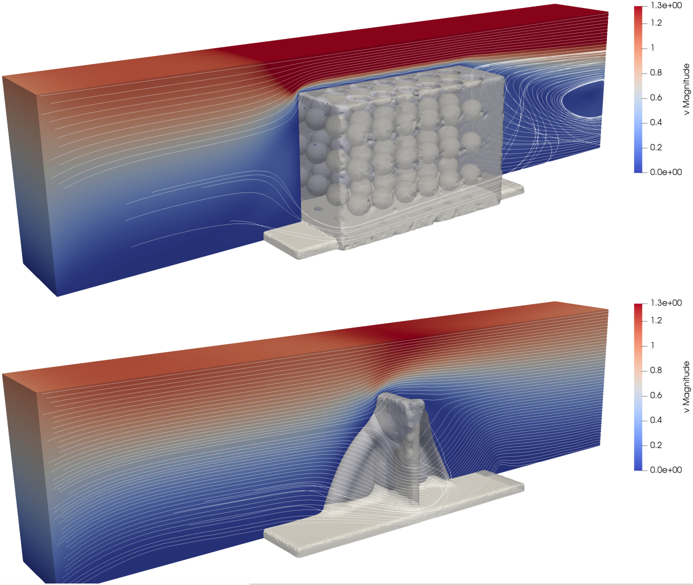
$$\begin{cases} -\operatorname{div}(\sigma_s(u)) = 0 & \text{in } \Omega_s \\ u = 0 & \text{on } \partial\Omega_s^D \\ \sigma_s(u)\cdot\boldsymbol{n} = 0 & \text{on } \partial\Omega_s^N \\ \sigma_s(u)\cdot\boldsymbol{n} = \sigma_f(v,p)\cdot\boldsymbol{n} & \text{on } \Gamma, \end{cases}$$

where $\sigma_s(u) = Ae(u) = 2\mu e(u) + \lambda \operatorname{tr}(e(u))I$.



Initialization and final design (Reynolds number 60).

Fluid-structure optimization in 3D



VI - Conclusions and perspectives

- Body-fitted meshing is crucial for accurate simulations.
- `Mmg` is very efficient and a parallel version, `ParMmg`, appeared in November 2021.
- Non-mixing constraint is easy in the level set framework.
- Other multi-physics problems can be tackled with our approach.
- `FreeFem++` and `Mmg` are open source softwares.