



Topology optimization of a thermo-fluid system and an eigenfrequency problem

Hao Li^{1,*}, Nari Nakayama^{1,*}, Pierre Jolivet², Naoyuki Ishida¹, Kousei Wano¹,
Tsuguo Kondoh¹, Kozo Furuta¹, Takayuki Yamada³, and Shinji Nishiwaki¹

¹Kyoto Univ., ²IRIT (France), ³UTokyo

*li.hao.48z@st.kyoto-u.ac.jp (H. Li)

*nakayama.nari.42z@st.kyoto-u.ac.jp (N. Nakayama)

December 11, 2021

Contents

1 Introduction

2 Key ingredients of the framework

- 2.1 Reaction–diffusion equation-based level-set method
- 2.2 Mesh adaptation techniques
- 2.3 Parallel computing
- 2.4 B-Reps conversion & simulation

3 Benchmark problems

- 3.1 Natural convection
- 3.2 Lift–drag
- 3.3 Multi-material design for a mean compliance problem
- 3.4 Multi-material design for an eigenfrequency problem

4 Final words

Table of Contents

1 Introduction

2 Key ingredients of the framework

- 2.1 Reaction–diffusion equation-based level-set method
- 2.2 Mesh adaptation techniques
- 2.3 Parallel computing
- 2.4 B-Reps conversion & simulation

3 Benchmark problems

- 3.1 Natural convection
- 3.2 Lift–drag
- 3.3 Multi-material design for a mean compliance problem
- 3.4 Multi-material design for an eigenfrequency problem

4 Final words

Introduction–Topology optimization (TO)

Topology optimization (TO) is a mathematical method that optimizes material layout within a given design space, for a given set of loads, boundary conditions and constraints with the goal of maximizing the performance of the system.

TO is different from shape optimization and size optimization in the sense that the design can **attain any shape within the design space**, instead of dealing with predefined configurations.

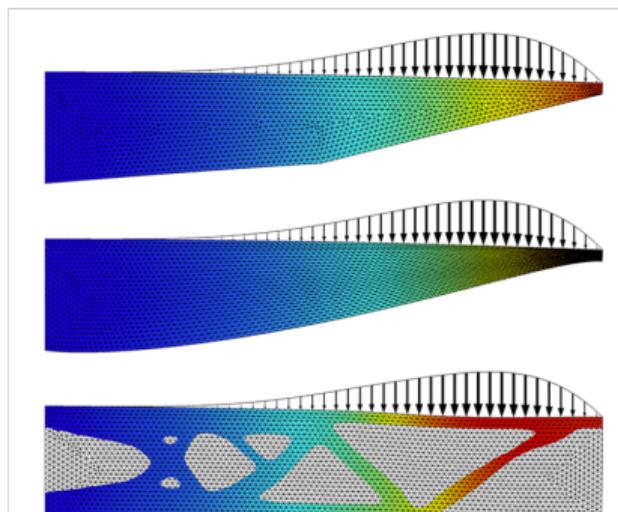


Fig. 1: Size, shape, and topology optimization
<https://www.comsol.jp/release/5.5/optimization-module>

Introduction–Mean compliance problem

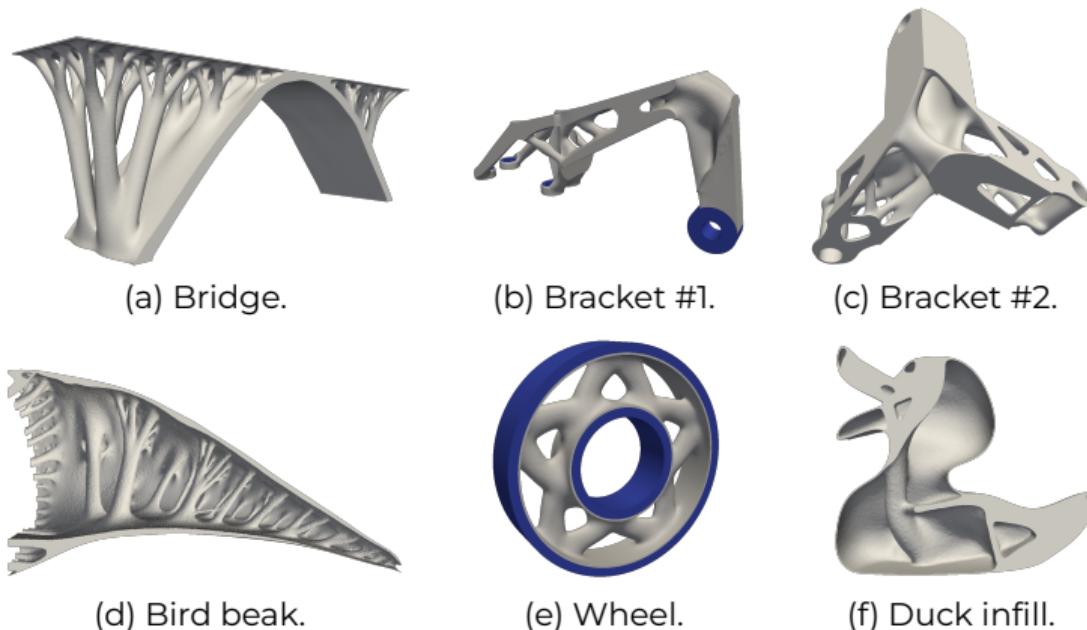
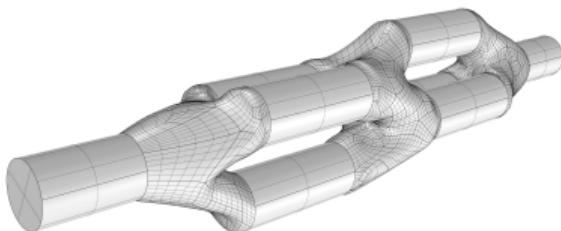


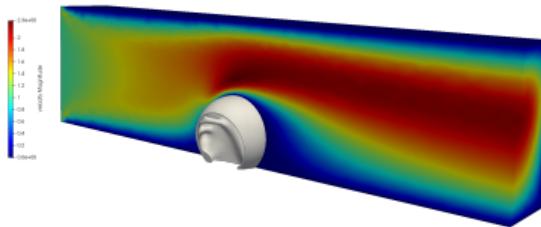
Fig. 2: Mean compliance problem¹

¹ Li, H., Yamada, T., Jolivet, P., Furuta, K., Kondoh, T., Izui, K., and Nishiwaki, S. Full-scale 3D structural topology optimization using adaptive mesh refinement based on the level-set method. *Finite Elements in Analysis and Design*. (2021) **194**: 103561. <https://doi.org/10.1016/j.finel.2021.103561>

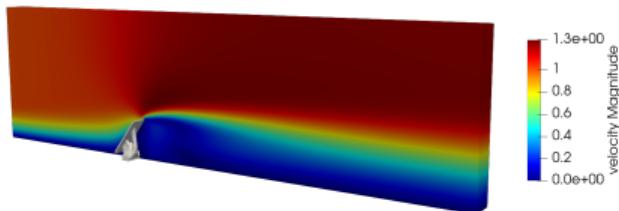
Introduction–Fluid–structure interaction problem



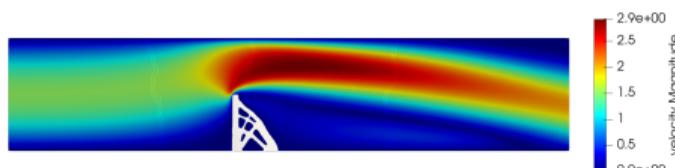
(a) Pipe connector.



(b) 3D “dry” FSI.



(c) 3D “wet” FSI.



(d) 2D “wet” FSI.

Fig. 3: Minimal power dissipation and FSI problems²

² Li, H., Kondoh, T., Jolivet, P., Furuta, K., Yamada, T., Zhu, B., Izui, K., and Nishiwaki, S. Three-dimensional topology optimization of a fluid–structure system using body-fitted mesh adaption based on the level-set method. *Applied Mathematical Modelling*. (2022) **101**: 276–308. <https://doi.org/10.1016/j.apm.2021.08.021>

Table of Contents

1 Introduction

2 Key ingredients of the framework

- 2.1 Reaction–diffusion equation-based level-set method
- 2.2 Mesh adaptation techniques
- 2.3 Parallel computing
- 2.4 B-Reps conversion & simulation

3 Benchmark problems

- 3.1 Natural convection
- 3.2 Lift–drag
- 3.3 Multi-material design for a mean compliance problem
- 3.4 Multi-material design for an eigenfrequency problem

4 Final words

Table of Contents

1 Introduction

2 Key ingredients of the framework

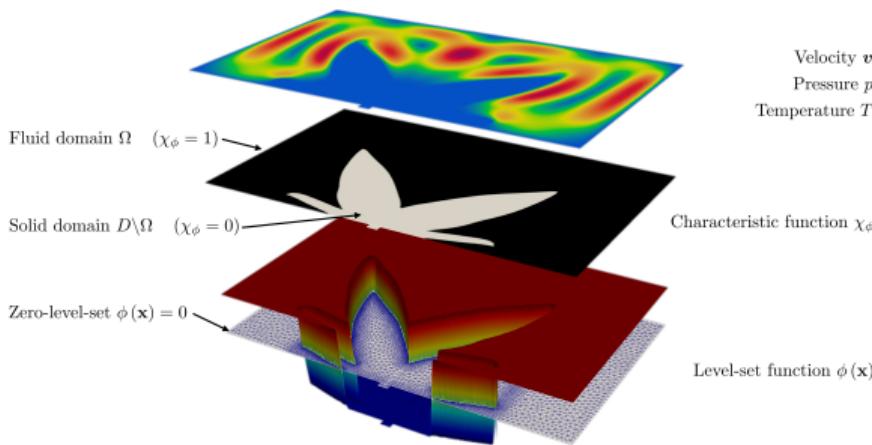
- 2.1 Reaction–diffusion equation-based level-set method
- 2.2 Mesh adaptation techniques
- 2.3 Parallel computing
- 2.4 B-Reps conversion & simulation

3 Benchmark problems

- 3.1 Natural convection
- 3.2 Lift–drag
- 3.3 Multi-material design for a mean compliance problem
- 3.4 Multi-material design for an eigenfrequency problem

4 Final words

Level-set boundary expression



The level-set function ϕ can be defined to have a piecewise constant profile as follows:

$$\begin{cases} 0 < \phi(\mathbf{x}) \leq 1 & \text{for } \mathbf{x} \in \Omega \\ \phi(\mathbf{x}) = 0 & \text{for } \mathbf{x} \in \partial\Omega \\ -1 \leq \phi(\mathbf{x}) < 0 & \text{for } \mathbf{x} \in D \setminus \Omega. \end{cases} \quad (1)$$

Next, a system can be modeled using a “1/0 binary structure” by mapping the level-set function $\phi(\mathbf{x})$ to the characteristic function χ_ϕ , as follows:

$$\chi_\phi = \begin{cases} 1 & \text{if } \phi \geq 0 \\ 0 & \text{if } \phi < 0. \end{cases} \quad (2)$$

Lastly, using **Ersatz material approach**, the material properties are interpolated, i.e.,

$$\begin{cases} \kappa(\chi_\phi) = 1 + (c_k - 1) \frac{q_\kappa(1 - \chi_\phi)}{q_\kappa + \chi_\phi} \\ \alpha(\chi_\phi) = \alpha_{\max} \frac{q_\alpha(1 - \chi_\phi)}{q_\alpha + \chi_\phi} \\ \dots \end{cases} \quad (3)$$

Level-set evolution equation

Level-set evolution equation (Yamada et al., 2010):

$$\begin{cases} \frac{\partial \phi}{\partial t} = - \left(\tilde{C} \bar{F}' - \tau \nabla^2 \phi \right) & \text{in } D \\ \phi = 0 & \text{on } \partial D, \end{cases} \quad (4)$$

Semi-discretize in fictitious time, Eq. (4) can be rewritten as:

$$\begin{cases} \frac{1}{\Delta t} (\phi_{n+1} - \phi_n) = - \left(\tilde{C} \bar{F}' - \tau \nabla^2 \phi_{n+1} \right) & \text{in } D \\ \phi = 0 & \text{on } \partial D. \end{cases} \quad (5)$$

Table of Contents

1 Introduction

2 Key ingredients of the framework

2.1 Reaction–diffusion equation-based level-set method

2.2 Mesh adaptation techniques

2.3 Parallel computing

2.4 B-Reps conversion & simulation

3 Benchmark problems

3.1 Natural convection

3.2 Lift–drag

3.3 Multi-material design for a mean compliance problem

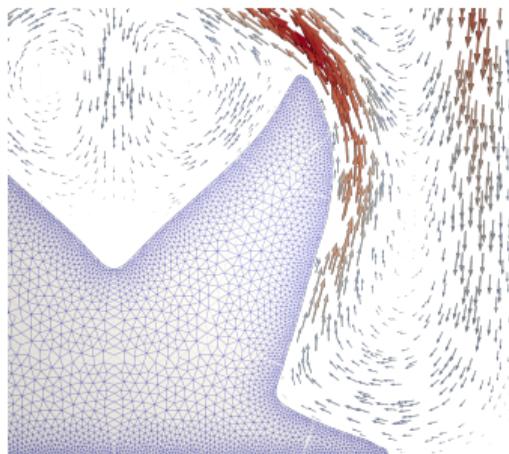
3.4 Multi-material design for an eigenfrequency problem

4 Final words

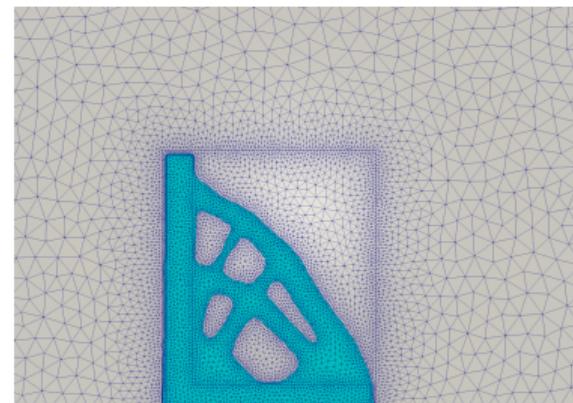
Body-fitted mesh

Highlights

- ① Modeling a fluid–solid system using separate equations;
- ② Convert into simulation model from what we see to what we get.



(a) Natural convection.



(b) FSI.

Body-fitted mesh

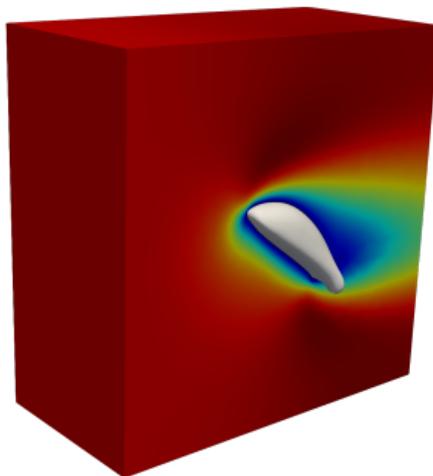
```
load "mmg"
ShtruncGlobal = mmg3d(ShtruncGlobal, metric=ophiGlobal[], iso=true, hmin=hmin, hmax=hmax,
                      hgrad = hgrad, hausd=hausd, localParameter = lp, requiredTriangle = rt);
```

- **iso**: if true, allows the implicit domain mesh (body-fitted) adaptation;
- **hmax, hmin**: maximum and minimum edge size, respectively;
- **hgrad**: imposes two adjacent edges e_1 and e_2 to satisfy $\frac{1}{h_{\text{grad}}} \leq \frac{l_{e_1}}{l_{e_2}} \leq h_{\text{grad}}$;
- **hausd**: imposes the maximal distance between the piecewise linear representation of the boundary and the reconstructed ideal boundary;
- **localParameter**: associates local Hausdorff, minimal edge size and maximal edge size to boundary references. The local parameters overwrite the global ones;
- **requiredTriangle**: associates the unchanged elements to the boundary reference.

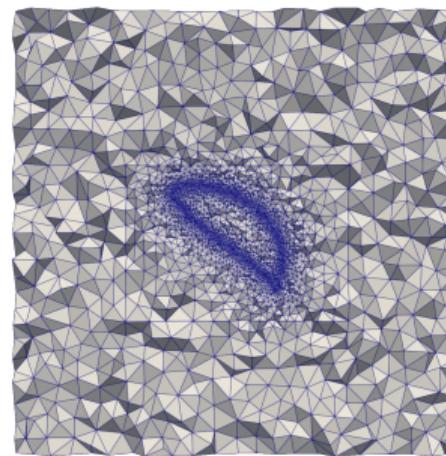
Anisotropic mesh

Highlights

- ① Capable to handle large meshes and complex geometries;



(a) Lift–drag problem.



(b) Slice view.

Fig. 5: Anisotropic mesh.

Anisotropic mesh

2D case:

```
ShGlobal = adaptmesh(ShGlobal, [uG,uGB], ratio = ratio, err = err, hmax = hmax, hmin = hmin);
```

3D case:

```
load "mshmet"
real[int] met = mshmet(ShGlobal, [uG,uGB,uGC], aniso = 1, hmin = hmin, hmax = hmax, err = err,
    nbregul = 1, normalization = 1);
load "parmmg"
ShGlobal = parmmg3d(ShGlobal, metric = met, hgrad = hgrad, comm=mpiCommWorld);
```

- **ratio**: ratio for a prescribed smoothing on the metric (1.5 by default);
- **err**: interpolation error level (0.01 by default);
- **metric**: array of 3 real arrays to set or get metric data information;
- **aniso**: if true, allows the adapted mesh to be anisotropic.

Table of Contents

1 Introduction

2 Key ingredients of the framework

2.1 Reaction–diffusion equation-based level-set method

2.2 Mesh adaptation techniques

2.3 Parallel computing

2.4 B-Reps conversion & simulation

3 Benchmark problems

3.1 Natural convection

3.2 Lift–drag

3.3 Multi-material design for a mean compliance problem

3.4 Multi-material design for an eigenfrequency problem

4 Final words

Numerical implementation

- **FreeFEM is used for the discretization of PDEs.** After decomposing an initial mesh using a graph partitioner package like METIS, the distributed assembly of the weak forms of Equations is performed by FreeFEM.
- **PETSc is used for the linear algebra backend.** The resulting discrete linear systems are passed over to PETSc. They are solved using multigrid preconditioners, GAMG for Lamé and hypre for RDE, modified augmented Lagrangian (mAL) for Navier-Stokes³.
- **SLEPc⁴ for a linear eigenvalue problem.**

These ensure that this part of the TO framework is scalable with respect to the problem size.

³Moulin, J., Jolivet, P., & Marquet, O. (2019). Augmented lagrangian preconditioner for large-scale hydrodynamic stability analysis. *Computer Methods in Applied Mechanics and Engineering*, 351, 718–743.

⁴EPSSolve: <https://slepc.upv.es/documentation/current/docs/manualpages/EPS/EPSSolve.html>

Table of Contents

1 Introduction

2 Key ingredients of the framework

2.1 Reaction–diffusion equation-based level-set method

2.2 Mesh adaptation techniques

2.3 Parallel computing

2.4 B-Reps conversion & simulation

3 Benchmark problems

3.1 Natural convection

3.2 Lift–drag

3.3 Multi-material design for a mean compliance problem

3.4 Multi-material design for an eigenfrequency problem

4 Final words

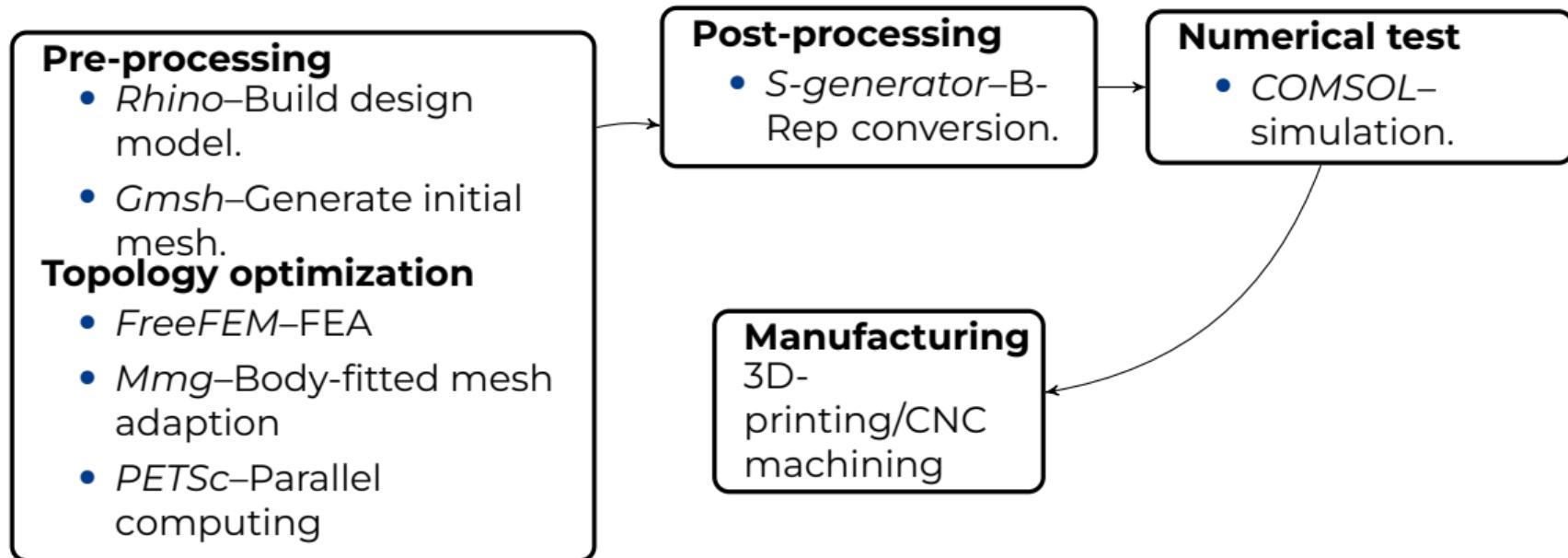


Fig. 6: The complete workflow of the product development based on topology optimization.

S-generator: <https://www.quint.co.jp/jp/pro/osg/index.htm>

B-Reps Conversion

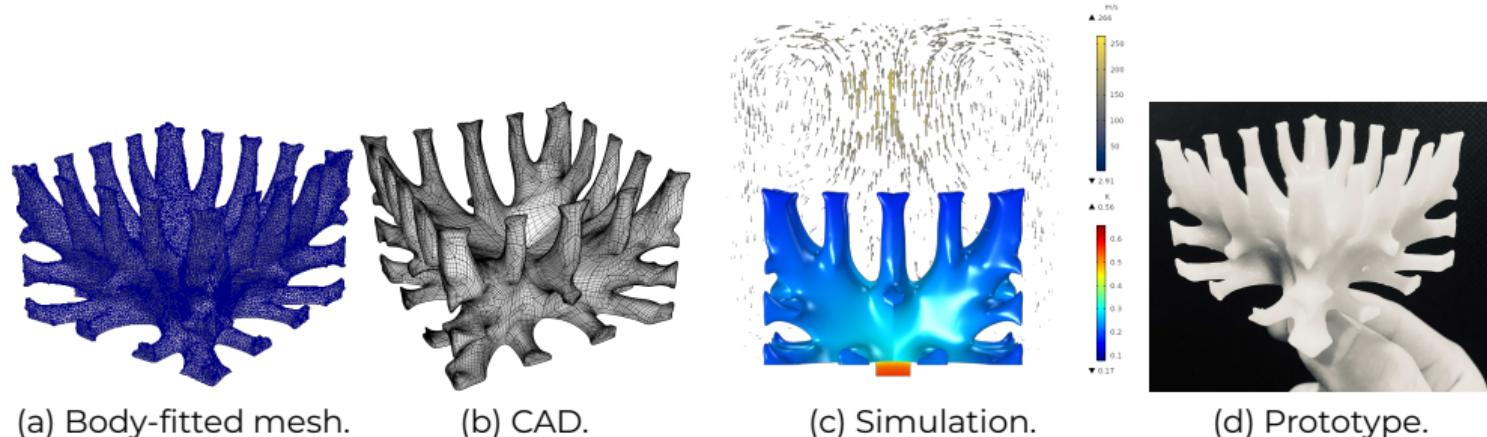


Fig. 7: A heat sink design case.

B-Reps conversion contributed by Kousei Wano
(<https://www.youtube.com/channel/UCeeZw9A4j33kzvDLMQHEgCg>)

Table of Contents

1 Introduction

2 Key ingredients of the framework

- 2.1 Reaction–diffusion equation-based level-set method
- 2.2 Mesh adaptation techniques
- 2.3 Parallel computing
- 2.4 B-Reps conversion & simulation

3 Benchmark problems

- 3.1 Natural convection
- 3.2 Lift–drag
- 3.3 Multi-material design for a mean compliance problem
- 3.4 Multi-material design for an eigenfrequency problem

4 Final words

Table of Contents

1 Introduction

2 Key ingredients of the framework

2.1 Reaction–diffusion equation-based level-set method

2.2 Mesh adaptation techniques

2.3 Parallel computing

2.4 B-Reps conversion & simulation

3 Benchmark problems

3.1 Natural convection

3.2 Lift–drag

3.3 Multi-material design for a mean compliance problem

3.4 Multi-material design for an eigenfrequency problem

4 Final words

Natural convection

Natural convection is a type of flow, of motion of a liquid such as water or a gas such as air, in which the fluid motion **is not generated by any external source** (like a pump, fan, suction device, etc.) but by some parts of the fluid being heavier than other parts. In most cases this leads to **natural circulation**, the ability of a fluid in a system to circulate continuously, with gravity and possible changes in heat energy.

In the field of thermal management, Natural convection is a **reliable, low noise, low vibration** form of air cooling, relying on buoyancy to drive air flow instead of a fan or pump. Like conduction, since natural convection doesn't rely on a mechanical system to drive flow, its size differs than that of forced air and liquid cold components, and can be more compact.

Natural convection cooling is used in situations where reliability, noise and size are critical. For instance in **cell phones, GPUs, semiconductors, telecommunication equipment and LEDs**.

Optimization model

$$\begin{aligned} \inf_{\chi_\phi \in \mathcal{X}} J(\Omega) &= \int_{\omega} QT d\Omega \\ \text{s.t. } G_1 &= \frac{\int_D 1 - \chi_\phi d\Omega}{\int_D d\Omega} - V_{\max} \leq 0, \end{aligned} \tag{6}$$

Governing equations:

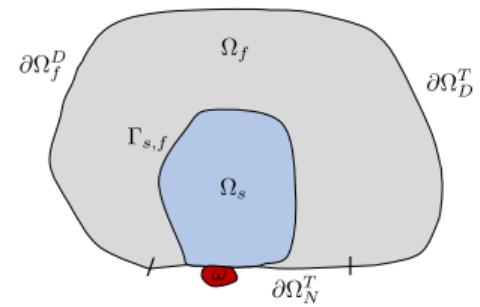
$$\begin{cases} (\mathbf{v} \cdot \nabla) \mathbf{v} - \Pr \nabla \cdot (\nabla \mathbf{v} + \nabla \mathbf{v}^T) + \nabla p - \mathbf{f}_{\text{Da}}(\mathbf{v}) - f_B(T) \mathbf{e}_3 = 0 \\ -\nabla \cdot \mathbf{v} = 0 \\ (\mathbf{v} \cdot \nabla T) - \nabla \cdot (\kappa \nabla T) - Q(\mathbf{x}) = 0, \end{cases} \tag{7}$$

where $Q(\mathbf{x})$ is the body heat source depending on the space \mathbf{x} :

$$Q(\mathbf{x}) = \begin{cases} Q_0 & \mathbf{x} \in \omega \\ 0 & \mathbf{x} \in \Omega \setminus \omega. \end{cases} \tag{8}$$

Boundary conditions:

$$\begin{cases} \mathbf{v} = 0 & \text{on } \partial\Omega \\ T = 0 & \text{on } \Gamma \\ \nabla T \cdot \mathbf{n} = 0 & \text{on } \partial\Omega \setminus \Gamma. \end{cases} \tag{9}$$



Sensitivity analysis I

Using Lagrange's method, the optimization problem given by Eq. (6) can be replaced with an unconstrained optimization problem. The Lagrangian $\mathcal{L}(\mathbf{v}, \mathbf{v}_A, p, p_A, T, T_A, \lambda_1, \Omega)$ is formulated as:

$$\begin{aligned} & \mathcal{L}(\mathbf{v}, \mathbf{v}_A, p, p_A, T, T_A, \lambda_1, \Omega) \\ &= \underbrace{\int_{\omega} QT d\Omega}_{\text{obj. function}} + \underbrace{\int_{\Omega} \mathbf{v}_A \cdot \left((\mathbf{v} \cdot \nabla) \mathbf{v} - \Pr \nabla \cdot (\nabla \mathbf{v} + \nabla \mathbf{v}^T) + \nabla p + \alpha \mathbf{v} - \text{Gr} \Pr^2 T \mathbf{e}_3 \right) d\Omega}_{\text{momentum conservation}} \\ & \quad - \underbrace{\int_{\Omega} p_A \underbrace{(\nabla \cdot \mathbf{v})}_{\text{continuity}} d\Omega}_{\text{continuity}} + \underbrace{\int_{\Omega} T_A \underbrace{((\mathbf{v} \cdot \nabla T) - \nabla \cdot (\kappa \nabla T) - Q)}_{\text{energy conservation}} d\Omega}_{\text{energy conservation}} + \lambda_1 \underbrace{\left(\frac{\int_D 1 - \chi_{\phi} d\Omega}{\int_D d\Omega} - V_{\max} \right)}_{\text{volume constraint}} \end{aligned} \quad (10)$$

Sensitivity analysis II

Using the definition of Gâteaux derivative, the partial derivative of the Lagrangian $\mathcal{L}(\mathbf{v}, \mathbf{v}_A, p, p_A, T, T_A, \lambda_1, \Omega)$ with respect to \mathbf{v} , p , and T in the direction ξ at the stationary point will be:

$$\begin{aligned} & \left\langle \frac{\partial \mathcal{L}}{\partial \mathbf{v}} (\mathbf{v}, \mathbf{v}_A, p, p_A, T, T_A, \lambda_1, \Omega), \xi \right\rangle \\ &= \int_{\Omega} -\xi \cdot (\mathbf{v} \cdot \nabla \mathbf{v}_A) + \xi \cdot (\mathbf{v}_A \cdot \nabla \mathbf{v}^T) - \xi \cdot (\text{Pr} \nabla \cdot (\nabla \mathbf{v}_A + \nabla \mathbf{v}_A^T)) + \xi \cdot \alpha \mathbf{v}_A + \xi \cdot \nabla p_A + T_A (\xi \cdot \nabla T) d\Omega \quad (11) \\ &+ \int_{\Gamma} (\mathbf{v}_A \cdot \xi) (\mathbf{v} \cdot \mathbf{n}) + (\xi \cdot \text{Pr} (\nabla \mathbf{v}_A + \nabla \mathbf{v}_A^T)) \cdot \mathbf{n} - (\mathbf{v}_A \cdot \text{Pr} (\nabla \xi + \nabla \xi^T)) \cdot \mathbf{n} - (p_A \xi) \cdot \mathbf{n} d\Gamma = 0. \end{aligned}$$

$$\begin{aligned} & \left\langle \frac{\partial \mathcal{L}}{\partial p} (\mathbf{v}, \mathbf{v}_A, p, p_A, T, T_A, \lambda_1, \Omega), \xi \right\rangle \quad (12) \\ &= - \int_{\Omega} \nabla \cdot \mathbf{v}_A \xi d\Omega + \int_{\Gamma} (\mathbf{v}_A \xi) \cdot \mathbf{n} d\Gamma = 0 \end{aligned}$$

$$\begin{aligned} & \left\langle \frac{\partial \mathcal{L}}{\partial T} (\mathbf{v}, \mathbf{v}_A, p, p_A, T, T_A, \lambda_1, \Omega), \xi \right\rangle \\ &= \int_{\omega} Q \xi d\Omega - \int_{\Omega} \mathbf{v}_A \cdot (\text{Gr} \text{Pr}^2 \xi \mathbf{e}_3) - (\nabla T_A \cdot \mathbf{v}) \xi - \nabla \cdot (\kappa \nabla T_A) \xi d\Omega \quad (13) \\ &+ \int_{\Gamma} (T_A \mathbf{v} \xi + \kappa \nabla T_A \xi - \kappa T_A \nabla \xi) \cdot \mathbf{n} d\Gamma = 0 \end{aligned}$$

Sensitivity analysis III

The above system of equations should be satisfied with any arbitrary ξ . Therefore, the adjoint equation can be obtained as follows:

$$\begin{cases} -\mathbf{v} \cdot \nabla \mathbf{v}_A + \mathbf{v}_A \cdot \nabla \mathbf{v}^T - \text{Pr} \nabla \cdot (\nabla \mathbf{v}_A + \nabla \mathbf{v}_A^T) + \alpha \mathbf{v}_A + \nabla p_A + T_A \nabla T = 0 \\ -\nabla \cdot \mathbf{v}_A = 0 \\ Q - \text{Gr} \text{Pr}^2 \mathbf{v}_A \cdot \mathbf{e}_3 - \nabla T_A \cdot \mathbf{v} - \nabla \cdot (\kappa \nabla T_A) = 0, \end{cases} \quad (14)$$

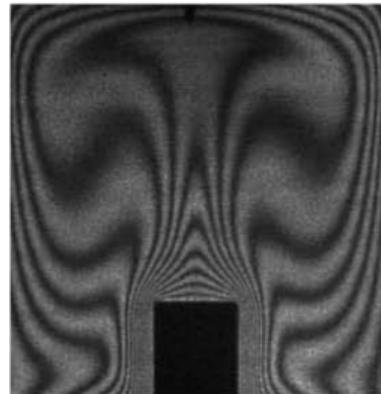
with the following adjoint boundary conditions:

$$\begin{cases} \mathbf{v}_A = 0 & \text{on } \partial\Omega \\ T_A = 0 & \text{on } \Gamma \\ (T_A \mathbf{v} + \kappa \nabla T_A) \cdot \mathbf{n} = 0 & \text{on } \partial\Omega \setminus \Gamma. \end{cases} \quad (15)$$

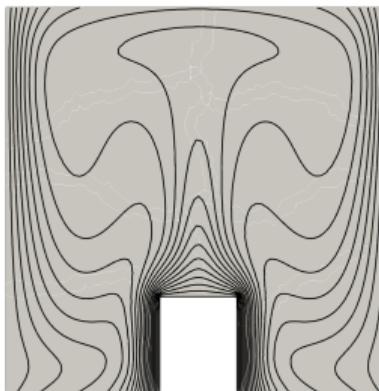
Finally, following the RAMP material interpolation scheme, the design sensitivity can be easily derived as follows:

$$\frac{d\mathcal{L}}{d\chi_\phi} = \int_{\Omega} -\frac{q_\alpha \alpha \max(q_\alpha + 1)}{(q_\alpha + \chi_\phi)^2} \mathbf{v} \cdot \mathbf{v}_A - \nabla T_A \cdot \nabla T \frac{q_\kappa (c_k - 1) (q_\kappa + 1)}{(q_\kappa + \chi_\phi)^2} - \lambda_1 d\Omega, \quad (16)$$

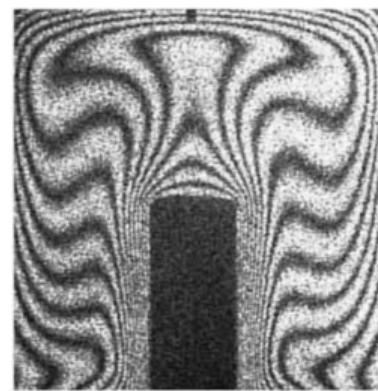
Experimental and numerical verification



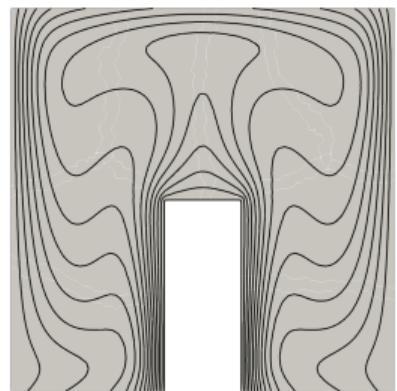
(a) Experimental.



(b) Numerical.



(c) Experimental.



(d) Numerical.

Fig. 8: Comparison amongst numerical (left) and experimental (right) results of Paroncini and Corvaro, 2009 for the isothermal contour at $\text{Ra} = 2.11 \cdot 10^5$ & $\xi = 0.25$, and $\text{Ra} = 2.25 \cdot 10^5$ & $\xi = 0.5$. [(a) and (c) reprinted from Paroncini and Corvaro, 2009 with permission from Elsevier.]

Fully-transient natural convection solver I

We introduce a dimensionless time by scaling the real time such that $t \rightarrow \frac{V_{\text{ref}}}{L}t$. Then, the unsteady-state natural convection problem can be formulated as follows:

$$\begin{cases} \frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{v} - \text{Pr} \nabla \cdot (\nabla \mathbf{v} + \nabla \mathbf{v}^T) + \nabla p + \alpha \mathbf{v} - \text{Gr} \text{Pr}^2 T \mathbf{e}_3 = 0 \\ -\nabla \cdot \mathbf{v} = 0 \\ \frac{\partial T}{\partial t} + (\mathbf{v} \cdot \nabla T) - \nabla \cdot (\kappa \nabla T) - Q = 0. \end{cases} \quad (17)$$

For the time integration, we use a second-order (BDF2) finite difference scheme to advance in time the velocity \mathbf{v} and temperature T . Then, the time-dependent terms can be semi-implicitly discretized as follows:

$$\frac{d\mathbf{v}}{dt} \simeq \frac{3\mathbf{v}_{n+1} - 4\mathbf{v}_n + \mathbf{v}_{n-1}}{2\delta t}, \quad \frac{dT}{dt} \simeq \frac{3T_{n+1} - 4T_n + T_{n-1}}{2\delta t}, \quad (18)$$

where δt is the time step. To advance the solution from time t_n to t_{n+1} , we give an initial guess (\mathbf{v}_n, p_n, T_n) .

Fully-transient natural convection solver II

Then, the weak formulation of the Jacobian and the residual at a given linearization point $(\mathbf{v}_{n+1}, p_{n+1}, T_{n+1})$, for an increment $(\delta\mathbf{v}_{n+1}, \delta p_{n+1}, \delta T_{n+1})$, are as follows:

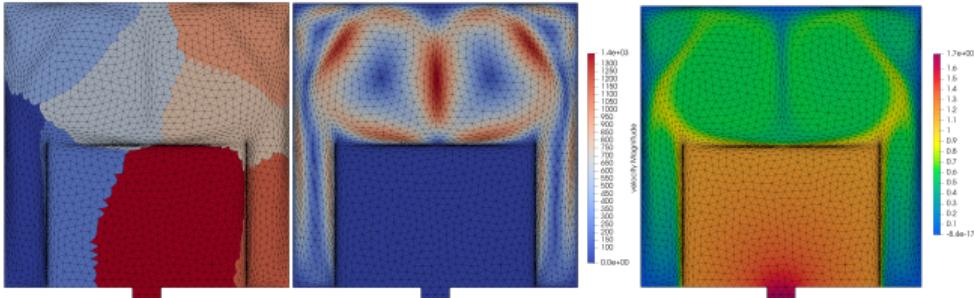
$$\begin{aligned} F(\mathbf{v}_{n+1}, p_{n+1}, T_{n+1}) = & \int_{\Omega} \left(\frac{3\mathbf{v}_{n+1} - 4\mathbf{v}_n + \mathbf{v}_{n-1}}{2\delta t} \right) \cdot \tilde{\mathbf{v}} + ((\mathbf{v}_{n+1} \cdot \nabla) \mathbf{v}_{n+1}) \cdot \tilde{\mathbf{v}} + \text{Pr} (\nabla \mathbf{v}_{n+1} + \nabla \mathbf{v}_{n+1}^T) : \nabla \tilde{\mathbf{v}} \\ & - p_{n+1} \nabla \cdot \tilde{\mathbf{v}} + \alpha \mathbf{v}_{n+1} \cdot \tilde{\mathbf{v}} - \text{Gr} \text{Pr}^2 T_{n+1} \mathbf{e}_3 \cdot \tilde{\mathbf{v}} - \tilde{p} \nabla \cdot \mathbf{v}_{n+1} \\ & + \left(\frac{3T_{n+1} - 4T_n + T_{n-1}}{2\delta t} \right) \tilde{T} + \tilde{T} (\mathbf{v}_{n+1} \cdot \nabla T_{n+1}) + (\kappa \nabla \tilde{T}) \cdot \nabla T_{n+1} d\Omega + \int_{\omega} Q \tilde{T} d\Omega \\ & \forall (\tilde{\mathbf{v}}, \tilde{p}, \tilde{T}) \in \tilde{\mathbf{V}} \times \tilde{P} \times \tilde{V} \end{aligned} \quad (19)$$

$$\begin{aligned} DF(\mathbf{v}_{n+1}, p_{n+1}, T_{n+1})(\delta\mathbf{v}_{n+1}, \delta p_{n+1}, \delta T_{n+1}) = & \int_{\Omega_f} \frac{3}{2\delta t} \delta\mathbf{v}_{n+1} \cdot \tilde{\mathbf{v}} + ((\delta\mathbf{v}_{n+1} \cdot \nabla) \mathbf{v}_{n+1}) \cdot \tilde{\mathbf{v}} + ((\mathbf{v}_{n+1} \cdot \nabla) \delta\mathbf{v}_{n+1}) \cdot \tilde{\mathbf{v}} + \text{Pr} (\nabla \tilde{\mathbf{v}} + \nabla \tilde{\mathbf{v}}^T) : \nabla \delta\mathbf{v}_{n+1} \\ & - \delta p_{n+1} \nabla \cdot \tilde{\mathbf{v}} + \alpha \delta\mathbf{v}_{n+1} \cdot \tilde{\mathbf{v}} - \text{Gr} \text{Pr}^2 \delta T_{n+1} \mathbf{e}_3 \cdot \tilde{\mathbf{v}} - \tilde{p} \nabla \cdot \delta\mathbf{v}_{n+1} \\ & + \frac{3}{2\delta t} \delta T_{n+1} \tilde{T} + \tilde{T} (\delta\mathbf{v}_{n+1} \cdot \nabla T) + \tilde{T} (\mathbf{v}_{n+1} \cdot \nabla \delta T_{n+1}) + \nabla \tilde{T} \cdot (\kappa \nabla \delta T_{n+1}) d\Omega \\ & \forall (\tilde{\mathbf{v}}, \tilde{p}, \tilde{T}) \in \tilde{\mathbf{V}} \times \tilde{P} \times \tilde{V}. \end{aligned} \quad (20)$$

Fully-transient natural convection solver III

Then, we construct the Newton sequence and solve Eq. (21) iteratively:

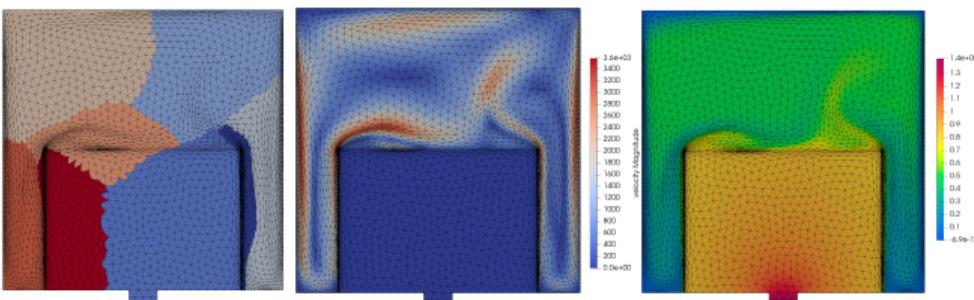
$$DF(\mathbf{v}_{n+1}, p_{n+1}, T_{n+1})(\delta\mathbf{v}_{n+1}, \delta p_{n+1}, \delta T_{n+1}) = F(\mathbf{v}_{n+1}, p_{n+1}, T_{n+1}). \quad (21)$$



(a) Domain decomposition.

(b) Velocity distribution.

(c) Temperature distribution.



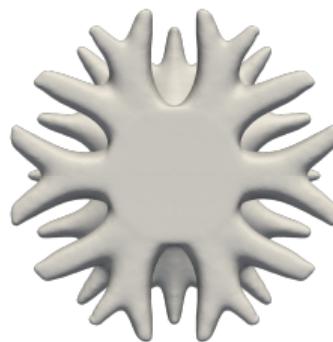
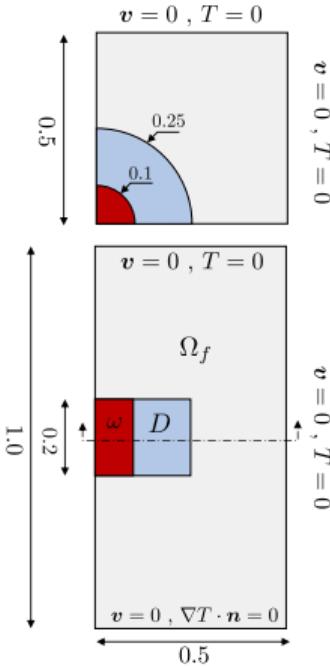
(d) Domain decomposition.

(e) Velocity distribution.

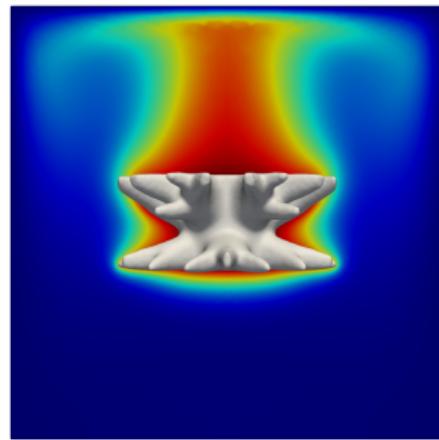
(f) Temperature distribution.

Fig. 9: 2D transient simulation result for different Grashof numbers: (a)–(c) $\text{Gr} = 10^7$ at time $t = 0.025$ and (d)–(f) $\text{Gr} = 10^8$ at time $t = 0.0994$. Anisotropic finite element mesh and subdomains for the domain decomposition and parallel computing using 8 MPI processes (left column). Velocity distribution (central column). Temperature distribution (right column).

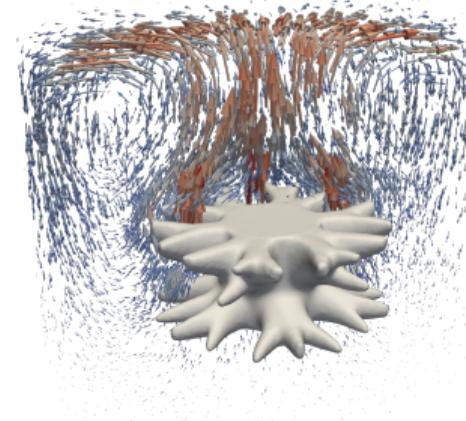
Design example



(a) Top view.



(b) Temperature.



(c) Velocity glyph.

Fig. 10: A suspended heated cylinder. ([See animation](#))

- Grashof number: $Gr = 10^6$;
- Last iteration: $3.56 \cdot 10^6$ unknowns;
- Total runtime (300 iterations): 41 h23 min.
- 32 cores, Intel(R) Xeon(R) Gold 6246R CPU @ 3.40GHz, Memory size is 128 GB;

Table of Contents

1 Introduction

2 Key ingredients of the framework

2.1 Reaction–diffusion equation-based level-set method

2.2 Mesh adaptation techniques

2.3 Parallel computing

2.4 B-Reps conversion & simulation

3 Benchmark problems

3.1 Natural convection

3.2 Lift–drag

3.3 Multi-material design for a mean compliance problem

3.4 Multi-material design for an eigenfrequency problem

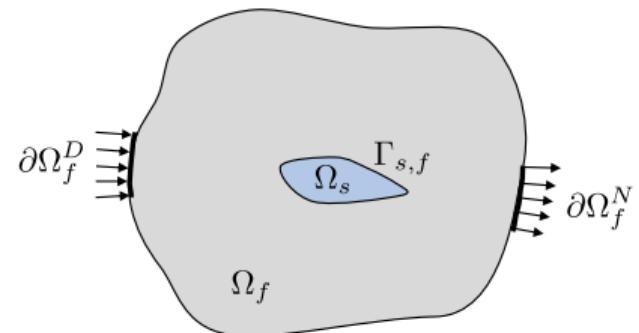
4 Final words

Optimization model I

$$\begin{aligned} \inf_{\chi_\phi \in \mathcal{X}} J(\Omega) &= -\text{Lift}(\Gamma, \mathbf{v}(\Gamma), p(\Gamma)) \\ \text{s.t. } &\begin{cases} G_1 = \text{Drag}(\Gamma, \mathbf{v}(\Gamma), p(\Gamma)) - C_D \text{Drag}_0 \leq 0 \\ G_2 = \frac{\int_D (1 - \chi_\phi) d\Omega}{\int_D d\Omega} - V_{\max} = 0 \\ G_3 = \frac{\int_D \mathbf{x} (1 - \chi_\phi) d\Omega}{\int_D (1 - \chi_\phi) d\Omega} - \mathbf{x}_0 = 0, \end{cases} \end{aligned} \quad (22)$$

Governing equations:

$$\left\{ \begin{array}{ll} -\operatorname{div}(\sigma_f(\mathbf{v}, p)) + (\mathbf{v} \cdot \nabla) \mathbf{v} + \mathbf{f}_{\text{Da}}(\mathbf{v}) = 0 & \text{in } \Omega_f \\ -\operatorname{div}(\mathbf{v}) = 0 & \text{in } \Omega_f \\ \mathbf{v} = \mathbf{v}_0 & \text{on } \partial\Omega_f^D \\ \sigma_f(\mathbf{v}, p) \cdot \mathbf{n}_f = 0 & \text{on } \partial\Omega_f^N \\ \mathbf{v} = 0 & \text{on } \Gamma_{s,f} \\ \mathbf{v} \cdot \mathbf{n} = 0 & \text{on } \Gamma, \end{array} \right. \quad (23)$$



where $\Gamma := \partial\Omega \setminus (\partial\Omega_f^D \cup \partial\Omega_f^N \cup \Gamma_{s,f})$ denotes the free-slip boundaries.

Optimization model II

The lift and drag force can be expressed by the boundary integral, as follows⁵:

$$\begin{aligned}\text{Drag} &= - \int_{\Gamma} (\mathbf{v} \cdot \mathbf{e}_x) (\mathbf{v} \cdot \mathbf{n}) + p \mathbf{n} \cdot \mathbf{e}_x d\Gamma \\ \text{Lift} &= - \int_{\Gamma} (\mathbf{v} \cdot \mathbf{e}_y) (\mathbf{v} \cdot \mathbf{n}) + p \mathbf{n} \cdot \mathbf{e}_y d\Gamma\end{aligned}\tag{24}$$

⁵Kondoh, T., Matsumori, T., & Kawamoto, A. (2012). Drag minimization and lift maximization in laminar flows via topology optimization employing simple objective function expressions based on body force integration. *Structural and Multidisciplinary Optimization*, 45(5), 693–701.

Sensitivity analysis I

The Lagrangian is formulated as follows:

$$\begin{aligned}\mathcal{L}(\boldsymbol{v}, \boldsymbol{v}_A, p, p_A, \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \Omega) \\ = J(\Omega) + \langle \boldsymbol{v}_A, R_1 \rangle + \langle p_A, R_2 \rangle + \sum_{i=1}^5 \langle \lambda_i, G_i \rangle,\end{aligned}\tag{25}$$

The design sensitivity can be derived as follows:

$$\frac{d\mathcal{L}}{d\chi_\phi} = \int_\Omega \frac{d\alpha(\chi_\phi)}{d\chi_\phi} \boldsymbol{v} \cdot \boldsymbol{v}_A - \lambda_2 - \lambda_3(x - x_0) - \lambda_4(y - y_0) - \lambda_5(z - z_0) d\Omega,\tag{26}$$

where \boldsymbol{v}_A and p_A can be obtained by solving the following adjoint equation:

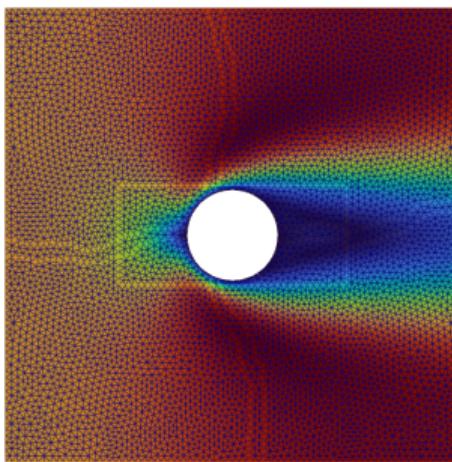
$$\begin{cases} -\boldsymbol{v} \cdot \nabla \boldsymbol{v}_A + \boldsymbol{v}_A \cdot \nabla \boldsymbol{v}^T - \frac{1}{Re} \nabla \cdot (\nabla \boldsymbol{v}_A + \nabla \boldsymbol{v}_A^T) + \alpha \boldsymbol{v}_A + \nabla p_A = 0 \\ -\nabla \cdot \boldsymbol{v}_A = 0, \end{cases}\tag{27}$$

Sensitivity analysis II

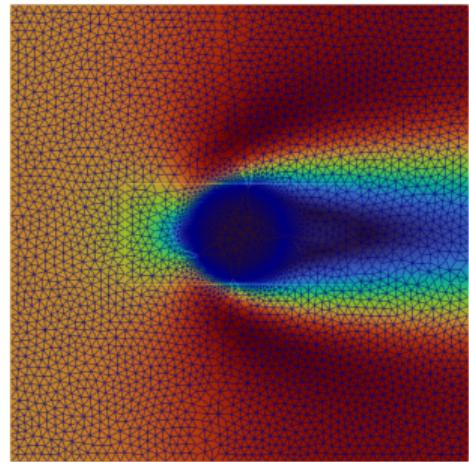
with the following adjoint boundary conditions:

$$\begin{cases} \mathbf{v}_A = \lambda_1 \mathbf{e}_x - \mathbf{e}_y & \text{on } \partial\Omega_f^D \\ (\mathbf{v} \cdot \mathbf{e}_y) \mathbf{n} - \lambda_1 ((\mathbf{v} \cdot \mathbf{e}_x) \mathbf{n} + \mathbf{e}_x (\mathbf{v} \cdot \mathbf{n})) + v_A (\mathbf{v} \cdot \mathbf{n}_f) \\ + \frac{1}{Re} (\nabla v_A + \nabla v_A^T) \cdot \mathbf{n}_f - p_A \mathbf{n}_f = 0 & \text{on } \partial\Omega_f^N \\ \mathbf{v}_A = 0 & \text{on } \Gamma_{s,f} \\ \mathbf{v}_A \cdot \mathbf{n} = 0 & \text{on } \Gamma. \end{cases} \quad (28)$$

Flow modeling strategies

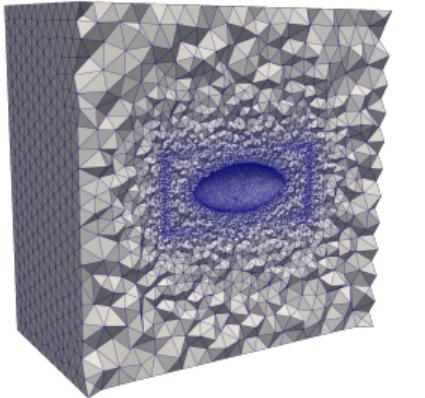


(a) Body-fitted mesh.

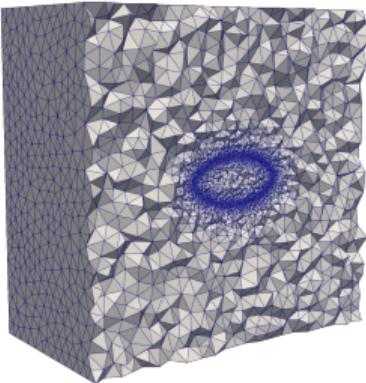


(b) Anisotropic mesh.

Fig. 11: “Separate” (left) and “hybrid” (right) flow modeling strategies.



(a) Body-fitted mesh.



(b) Anisotropic mesh.

Fig. 12: “Separate” (left) and “hybrid” (right) flow modeling strategies.

Actions	“Separate” modeling	“Hybrid” modeling
Solve governing equations	127.6	37.1
Solve adjoint equations	41.0	47.5
Compute sensitivity and Lagrange multiplier	0.16	0.12
Solve RDE	0.23	0.23
Visualize output	0.48	0.51
Centralize solution to process #0	0.93	0.87
Remesh	33.7	2.41
Partition updated mesh	5.19	4.11
Interpolate solution to updated mesh	1.72	1.39
Total	211.0	94.2

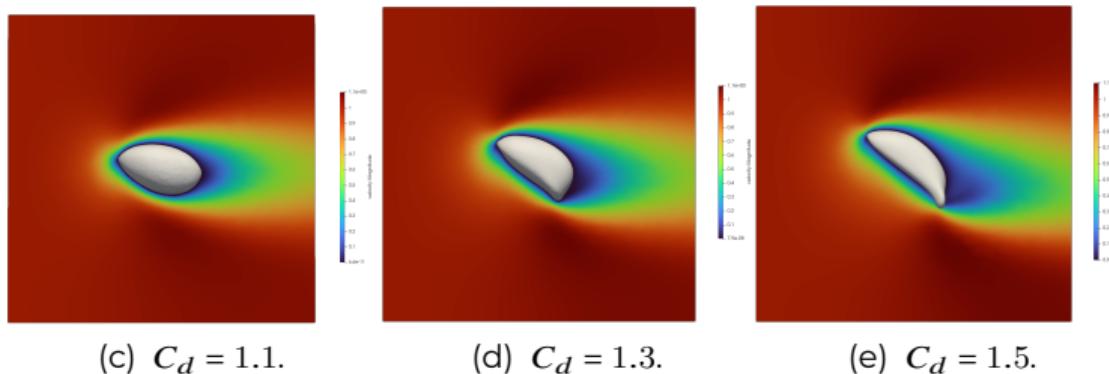
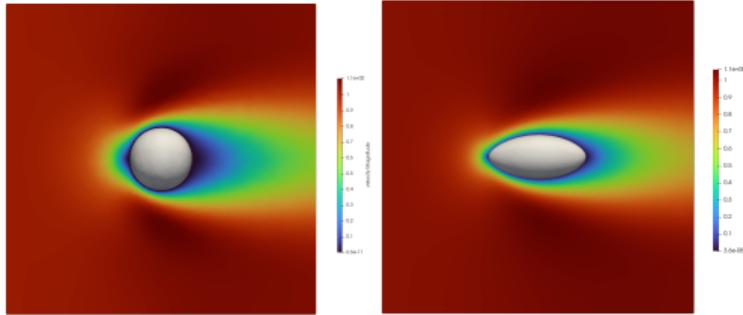


Fig. 13: Design results of lift–drag problem.

Table of Contents

1 Introduction

2 Key ingredients of the framework

- 2.1 Reaction–diffusion equation-based level-set method
- 2.2 Mesh adaptation techniques
- 2.3 Parallel computing
- 2.4 B-Reps conversion & simulation

3 Benchmark problems

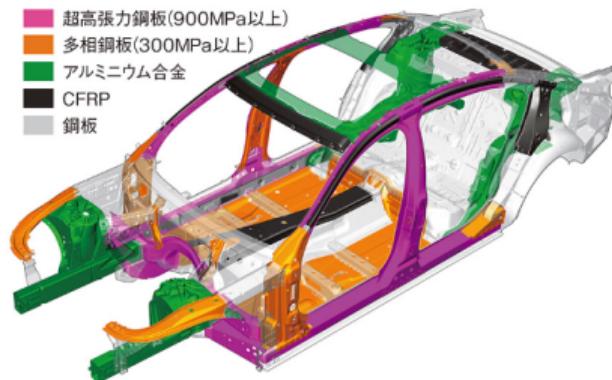
- 3.1 Natural convection
- 3.2 Lift–drag
- 3.3 Multi-material design for a mean compliance problem**
- 3.4 Multi-material design for an eigenfrequency problem

4 Final words

Multi-material structures

Introduction of multi-material structures, **using the right materials in the right parts**, is necessary and indispensable for reducing the weight of automobiles, aircraft and other transportation equipment.

Thus, comprehensive technology development, beginning with establishment of design techniques for **optimizing multi-material structures** and also including modeling of joints of dissimilar materials, is an urgent challenge. Because integration of development results for different types of materials and collaboration will be critical.



<http://techon.nikkeibp.co.jp/>

Multi-material level-set method

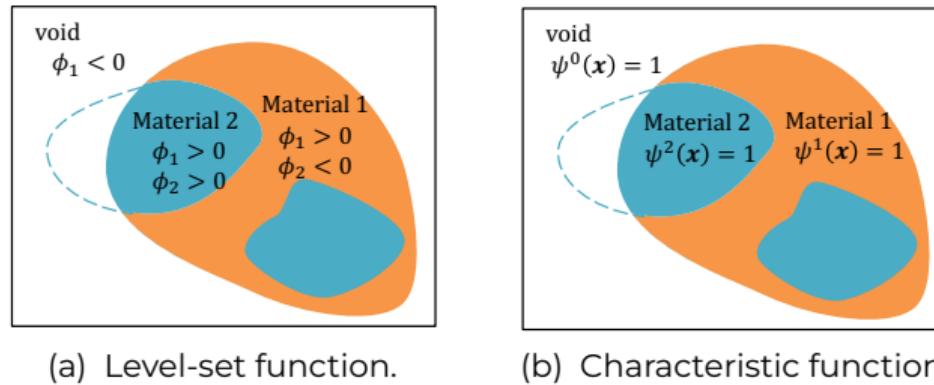


Fig. 14: Level-set boundary expression for multi-material structures.

$$\begin{cases} \psi^0(\mathbf{x}) = \{1 - \chi_{\phi^1}(\mathbf{x})\} \\ \psi^1(\mathbf{x}) = \chi_{\phi^1}(\mathbf{x}) \{1 - \chi_{\phi^2}(\mathbf{x})\} \\ \psi^2(\mathbf{x}) = \chi_{\phi^1}(\mathbf{x}) \chi_{\phi^2}(\mathbf{x}) \end{cases} \quad (29)$$

$$\begin{aligned} C_{ijkl}(\mathbf{x}) &= C_{ijkl}^2 \psi^2(\mathbf{x}) + C_{ijkl}^1 \psi^1(\mathbf{x}) + C_{ijkl}^0 \psi^0(\mathbf{x}) \\ \rho(\mathbf{x}) &= \rho^2 \psi^2(\mathbf{x}) + \rho^1 \psi^1(\mathbf{x}) + \rho^0 \psi^0(\mathbf{x}) \end{aligned} \quad (30)$$

Optimization model

$$\begin{aligned} \inf_{\chi_\phi \in \mathcal{X}} J(\Omega) &= \int_{\Gamma_t} \mathbf{g} \cdot \mathbf{u} d\Gamma \\ \text{s.t. } &\begin{cases} G_1 = \int_D \chi_{\phi_1} (1 - \chi_{\phi_2}) d\Omega - V_{\max}^1 \leq 0 \\ G_2 = \int_D \chi_{\phi_1} \chi_{\phi_2} d\Omega - V_{\max}^2 \leq 0 \end{cases} \end{aligned} \quad (31)$$

Governing equations:

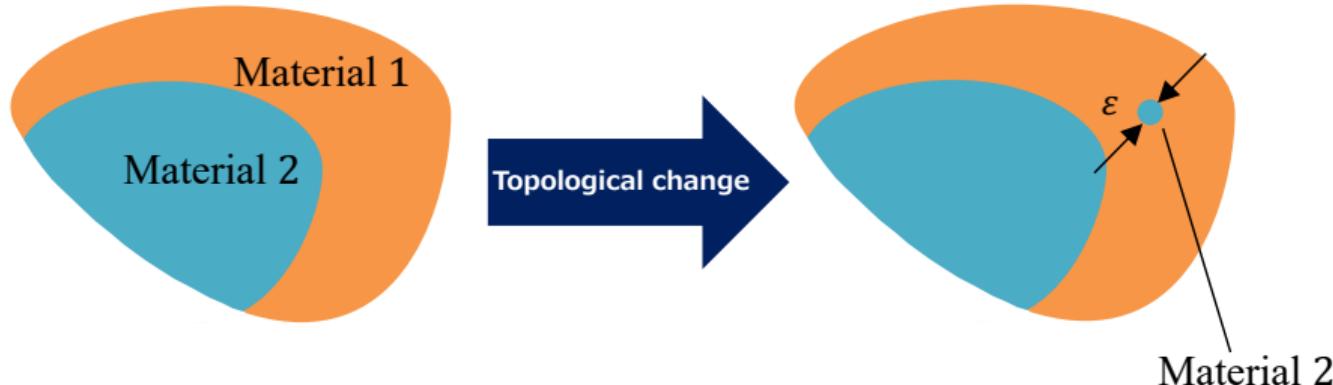
$$\begin{aligned} C_{ijkl}^1 u_{j,kl} &= 0 && \text{in } \Omega_1 \\ C_{ijkl}^2 u_{j,kl} &= 0 && \text{in } \Omega_2 \\ u_i^1 &= u_i^2 && \text{on } \Gamma_{12} \\ u_{j,k}^1 C_{ijkl}^1 n_l^1 + u_{j,k}^1 C_{ijkl}^2 n_l^2 &= 0 && \text{on } \Gamma_{12} \\ u_i &= 0 && \text{on } \Gamma_u \\ u_{j,k} C_{ijkl} n_l &= g_i && \text{on } \Gamma_t \\ u_{j,k} C_{ijkl} n_l &= 0 && \text{on } \Gamma_N \end{aligned} \quad (32)$$

Topological derivative

Definition

The topological derivative is, conceptually, a derivative of a shape functional with respect to infinitesimal changes in its topology, such as adding an infinitesimal hole.

$$D\mathcal{L}^{1 \rightarrow 2} = \lim_{\varepsilon \rightarrow 0} \frac{\delta F^{1 \rightarrow 2}}{V_\varepsilon} \quad (33)$$



Topological derivative

Theorem (Topological-shape sensitivity analysis)

Let $f(\epsilon)$ be a function chosen in order to $0 < |D_T^*(\hat{x})| < \infty$, then the limit with $\epsilon \rightarrow 0$ that appears in the definition of the topological derivative can be written as (Novotny et al., 2003):

$$D_T^*(\hat{x}) = D_T(\hat{x}) = \lim_{\epsilon \rightarrow 0} \frac{1}{f'(\epsilon) |V_n|} \left. \frac{d\psi(\Omega_\tau)}{d\tau} \right|_{\tau=0} \quad (34)$$

$$\begin{aligned} D_T J^{1 \rightarrow 2} = & \frac{2\mu^1(\kappa^1 + 1) \left\{ \mu^2(\kappa^1 - 1) - \mu^1(\kappa^2 - 1) \right\}}{(\mu^1(\kappa^2 - 1) + 2\mu^2)(\kappa^1 - 1)^2} \left\{ \frac{\partial u_x}{\partial x} + \frac{\partial u_y}{\partial y} \right\}^2 \\ & + \frac{\mu^1(\mu^2 - \mu^1)(\kappa^1 + 1)}{\mu^1 + \kappa^1 \mu^2} \left\{ \frac{\partial u_x}{\partial x} - \frac{\partial u_y}{\partial y} \right\}^2 \\ & + \frac{\mu^1(\mu^2 - \mu^1)(\kappa^1 + 1)}{\mu^1 + \kappa^1 \mu^2} \left\{ \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \right\}^2 \end{aligned} \quad (35)$$

κ is Kosolov's constant defined as follows:

$$\begin{aligned} \kappa &= \frac{3 - \nu}{1 + \nu} \quad (\text{for plane stress}) \\ \kappa &= 3 - 4\nu \quad (\text{for plane strain}) \end{aligned} \quad (36)$$

Similarly, we compute $D_T J^{1 \rightarrow 0}$, $D_T J^{2 \rightarrow 0}$, $D_T J^{0 \rightarrow 1}$, $D_T J^{0 \rightarrow 2}$, and $D_T J^{2 \rightarrow 1}$.

Topological derivative

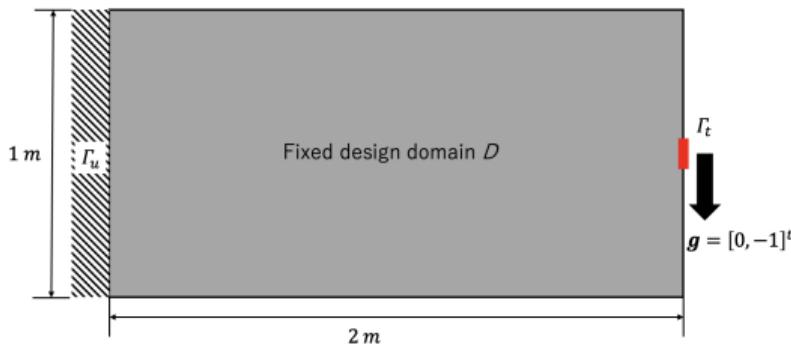
The design sensitivities are computed as:

$$\begin{cases} D\mathcal{L}^{(1)} = \chi_{\phi^1} (1 - \chi_{\phi^2}) D\mathcal{L}^{1 \rightarrow 0(\text{Void})} + \chi_{\phi^1} \chi_{\phi^2} D\mathcal{L}^{2 \rightarrow 0(\text{Void})} \\ \quad + (1 - \chi_{\phi^1}) \{(1 - \chi_{\phi^2}) D\mathcal{L}^{0(\text{Void}) \rightarrow 1} + \chi_{\phi^2} D\mathcal{L}^{0(\text{Void}) \rightarrow 2}\} \\ D\mathcal{L}^{(2)} = \chi_{\phi^1} (1 - \chi_{\phi^2}) D\mathcal{L}^{1 \rightarrow 2} + \chi_{\phi^1} \chi_{\phi^2} D\mathcal{L}^{2 \rightarrow 1} \end{cases} \quad (37)$$

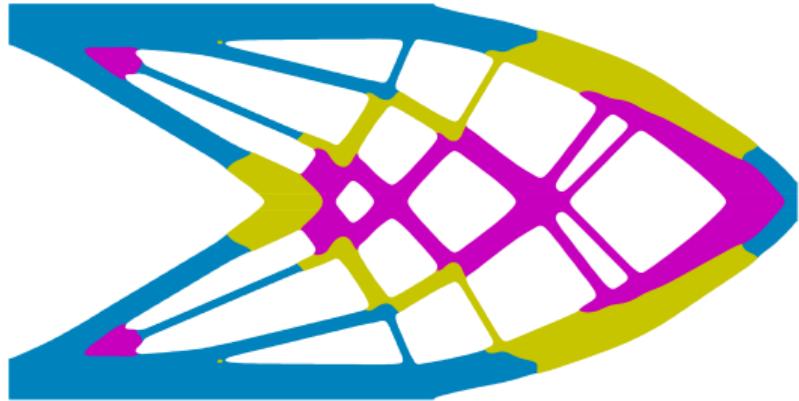
Reaction-diffusion equation for updating multiple level-set functions:

$$\begin{aligned} \frac{\partial \phi^k}{\partial t} &= -K \cdot F'_R \\ &= -K \left(D\mathcal{L}^{(k)} - \tau^k \nabla^2 \phi^k \right) \end{aligned} \quad (38)$$

Design example–2D case



(a) Design model.

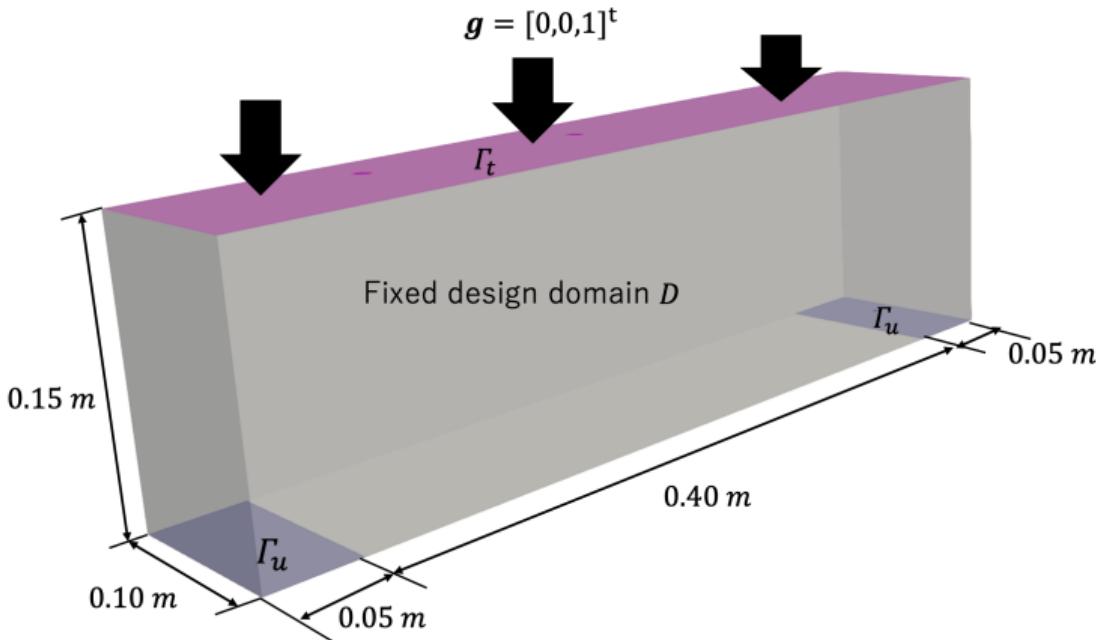


(b) Optimal solution.

Fig. 15: A multi-material design for a cantilever.

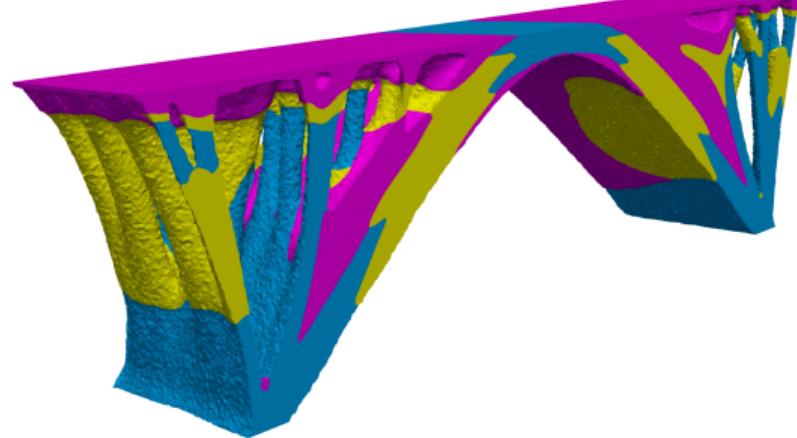
- Volume fraction
 - ▶ pink : 10%
 - ▶ yellow: 10%
 - ▶ blue : 20%
- Young's modulus
 - ▶ pink : 1.0
 - ▶ yellow: 2.0
 - ▶ blue : 3.0
- Poisson's ratio
 - ▶ pink :0.3
 - ▶ yellow:0.3
 - ▶ blue :0.3

Design example–3D case

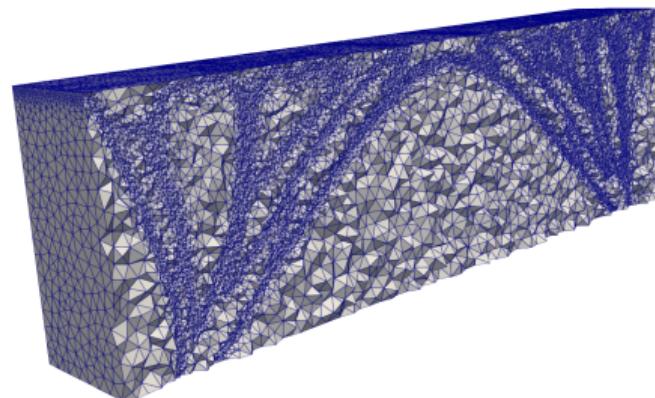


- Volume fraction
 - ▶ pink : 10%
 - ▶ yellow: 10%
 - ▶ blue : 10%
- Young's modulus
 - ▶ pink : 1.0
 - ▶ yellow: 2.0
 - ▶ blue : 3.0
- Poisson's ratio
 - ▶ pink : 0.3
 - ▶ yellow: 0.3
 - ▶ blue : 0.3

Design example–3D case



(a) Optimal solution.



(b) Cross section view of anisotropic mesh.

Fig. 16: A multi-material design for a bridge. [see the animation](#)

- $1.23 \cdot 10^6$ unknowns, $2.37 \cdot 10^6$ tetrahedrons;
- 8 cores, Intel(R) Xeon(R) W CPU @ 3.20GHz, Memory size is 32 GB.

Table of Contents

1 Introduction

2 Key ingredients of the framework

2.1 Reaction–diffusion equation-based level-set method

2.2 Mesh adaptation techniques

2.3 Parallel computing

2.4 B-Reps conversion & simulation

3 Benchmark problems

3.1 Natural convection

3.2 Lift–drag

3.3 Multi-material design for a mean compliance problem

3.4 Multi-material design for an eigenfrequency problem

4 Final words

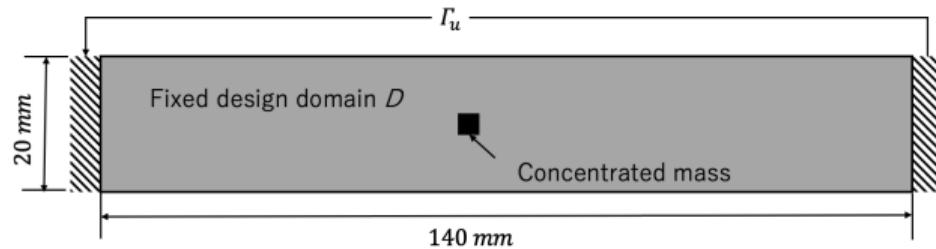
Optimization model

$$\begin{aligned} \inf_{\chi_\phi \in \mathcal{X}} J(\Omega) &= - \left(\sum_{m=1}^n \frac{1}{\lambda(m)} \right)^{-1} \\ \text{s.t. } &\begin{cases} G_1 = \int_D \chi_{\phi_1} (1 - \chi_{\phi_2}) d\Omega - V_{\max}^1 \leq 0 \\ G_2 = \int_D \chi_{\phi_1} \chi_{\phi_2} d\Omega - V_{\max}^2 \leq 0 \end{cases} \end{aligned} \quad (39)$$

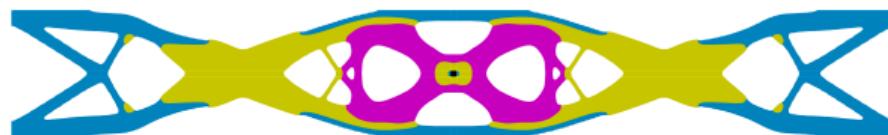
Governing equations:

$$\begin{aligned} C_{ijkl}^1 u_{j,kl}^{(m)} &= -\lambda^{(m)} \rho^1 u_i^{(m)} && \text{in } \Omega_1 \\ C_{ijkl}^2 u_{j,kl}^{(m)} &= -\lambda^{(m)} \rho^2 u_i^{(m)} && \text{in } \Omega_2 \\ u_i^{(m)1} &= u_i^{(m)2} && \text{on } \Gamma_{12} \\ u_{j,k}^{(m)1} C_{ijkl}^1 n_l^1 + u_{j,k}^{(m)1} C_{ijkl}^2 n_l^2 &= 0 && \text{on } \Gamma_{12} \\ u_i^{(m)} &= 0 && \text{on } \Gamma_D \\ u_{j,k}^{(m)} C_{ijkl} n_l &= 0 && \text{on } \Gamma_N \end{aligned} \quad (40)$$

Design example–2D case



(a) Design model.



(b) Optimal solution.

- Volume fraction
 - ▶ pink : 10%
 - ▶ yellow: 20%
 - ▶ blue : 20%
- Young's modulus
 - ▶ pink : 45 GPa
 - ▶ yellow: 68 GPa
 - ▶ blue : 210 GPa
- Poisson's ratio
 - ▶ pink : 0.35
 - ▶ yellow: 0.33
 - ▶ blue : 0.30
- Mass density
 - ▶ pink : 1740 kg m^{-3}
 - ▶ yellow : 2680 kg m^{-3}
 - ▶ blue : 7850 kg m^{-3}
- Concentrated mass
 - ▶ 4.0 kg m^{-1}

Fig. 17: A multi-material design for a beam.

Design example–2D case

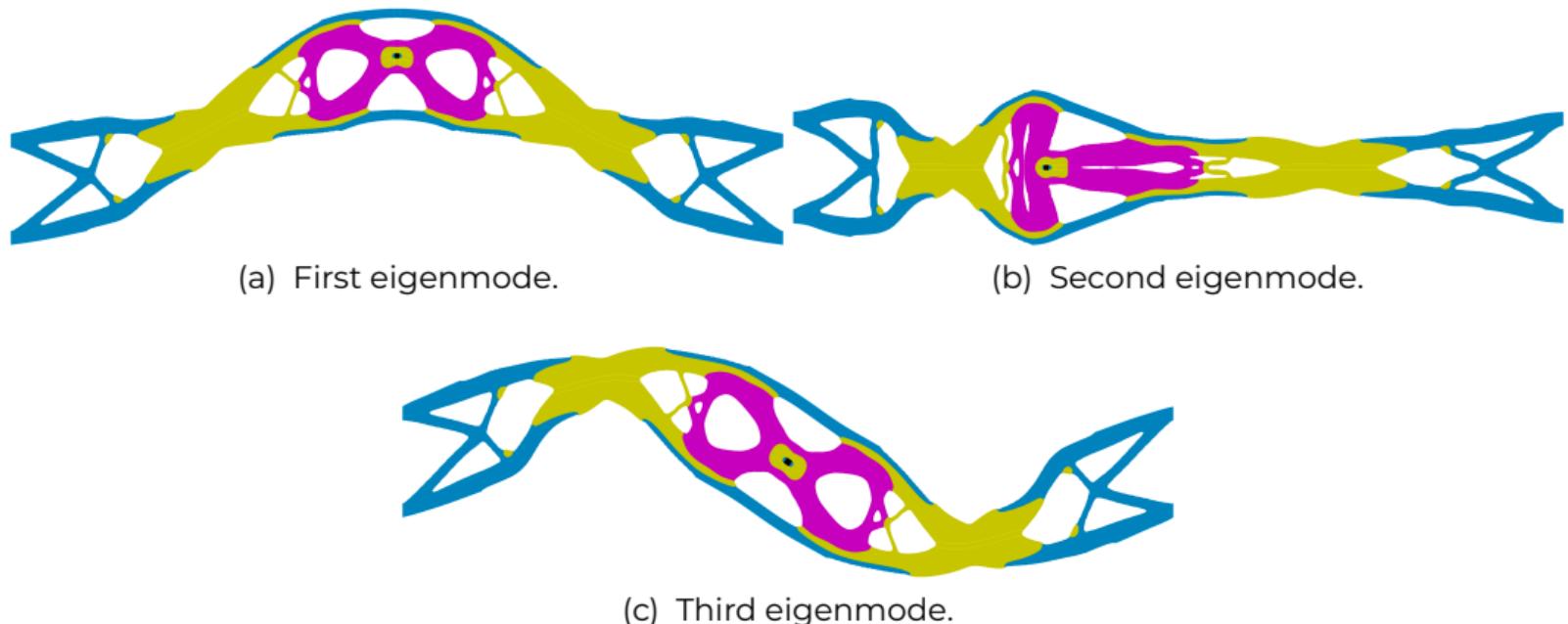
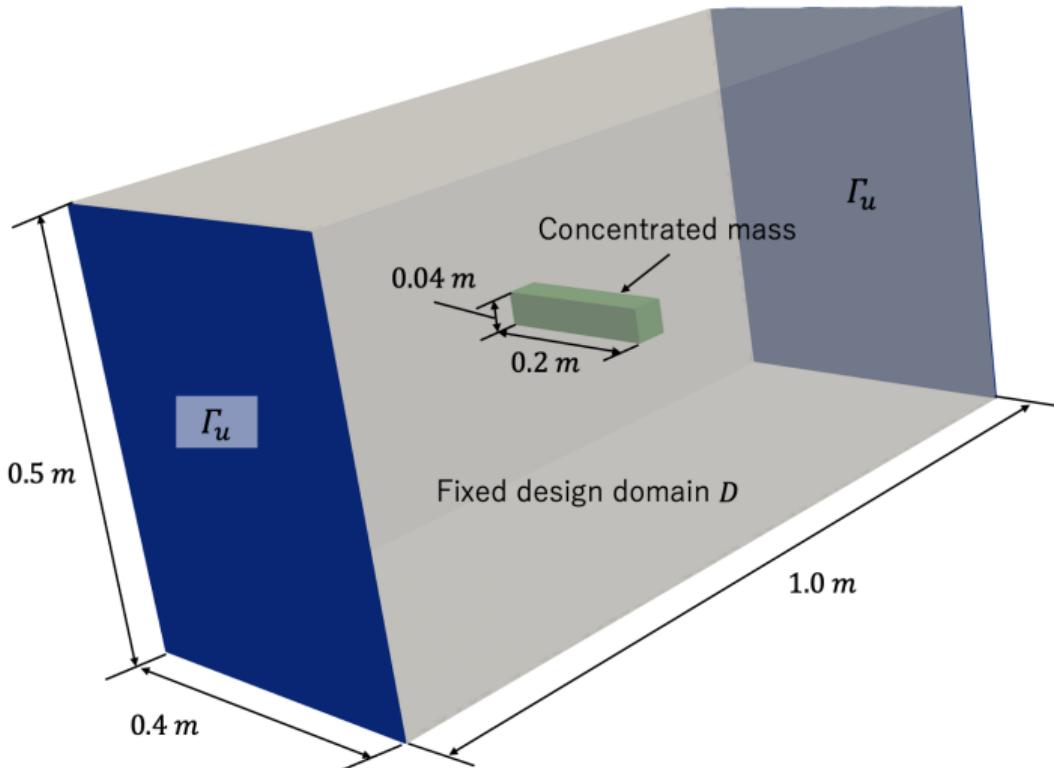


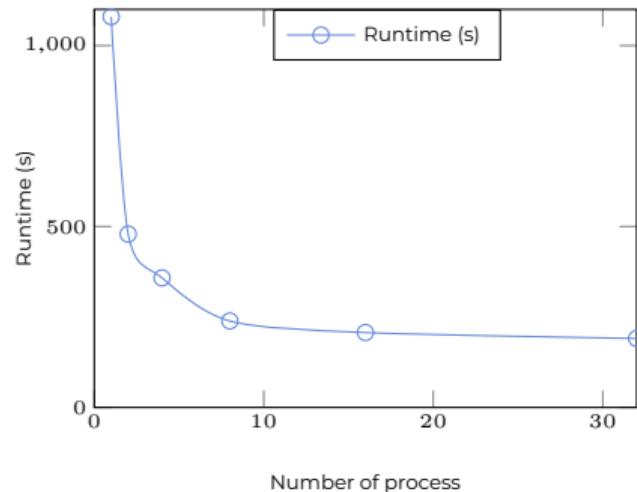
Fig. 18: Eigenmodes for the optimal solution.

Design example–3D case

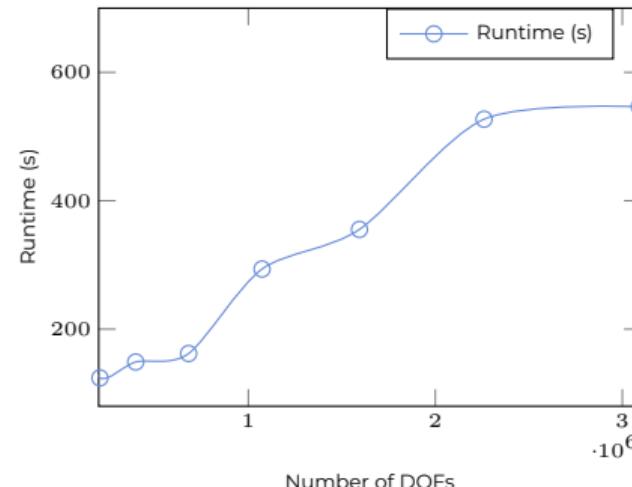


- Volume fraction
 - ▶ pink :0.15
 - ▶ yellow:0.10
 - ▶ blue :0.15
- Young's modulus
 - ▶ pink :45 GPa
 - ▶ yellow:68 GPa
 - ▶ blue :210 GPa
- Poisson's ratio
 - ▶ pink :0.35
 - ▶ yellow:0.33
 - ▶ blue :0.30
- Mass density
 - ▶ pink : 1740 kg mm^{-3}
 - ▶ yellow : 2680 kg mm^{-3}
 - ▶ blue : 7850 kg mm^{-3}
- Concentrated mass
 - ▶ 500 kg

Scalability analysis



(a) $6.81 \cdot 10^5$ unknowns.

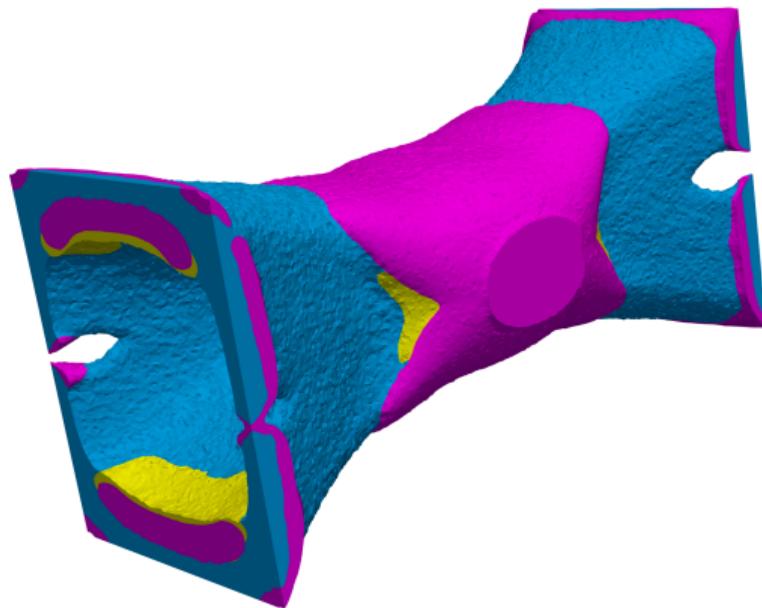


(b) 8 processes

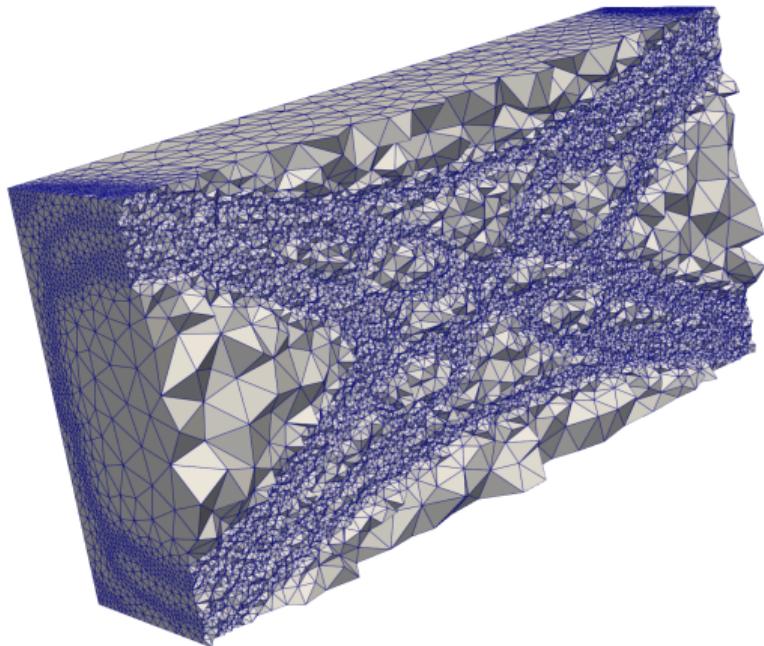
Fig. 19: Scalability analysis for an eigenfrequency solver.

- 40 cores, Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz, Memory size is 256 GB.

Design example–3D case



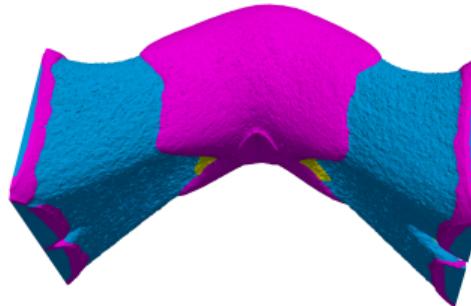
(a) Optimal solution.



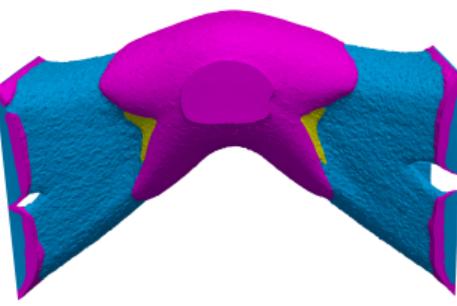
(b) Cross section view of mesh.

Fig. 20: A multi-material design for a beam. ([see animation](#))

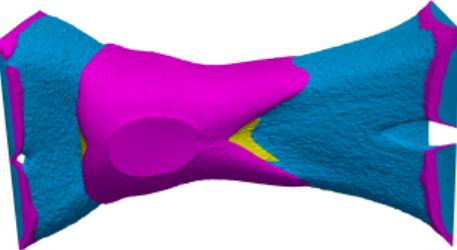
Design example–3D case



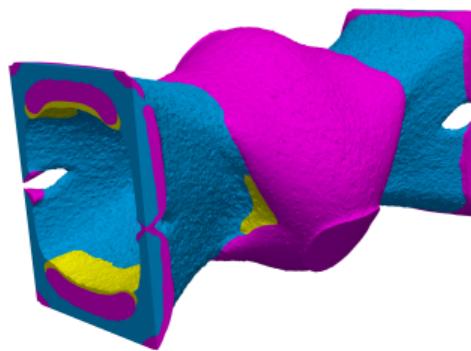
(a) First eigenmode.



(b) Second eigenmode.



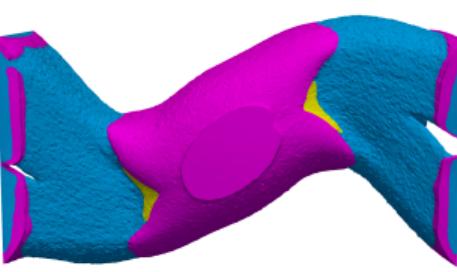
(c) Third eigenmode.



(d) Forth eigenmode.



(e) Fifth eigenmode.



(f) Sixth eigenmode.

Table of Contents

1 Introduction

2 Key ingredients of the framework

2.1 Reaction–diffusion equation-based level-set method

2.2 Mesh adaptation techniques

2.3 Parallel computing

2.4 B-Reps conversion & simulation

3 Benchmark problems

3.1 Natural convection

3.2 Lift–drag

3.3 Multi-material design for a mean compliance problem

3.4 Multi-material design for an eigenfrequency problem

4 Final words

① Summary

- ▶ Interesting applications in multiphysics problems
 - ▶ Natural convection;
 - ▶ Lift-drag;
 - ▶ mutimaterial design for mean compliance problem;
 - ▶ multimaterial design for an eigenvalue problem.
- ▶ Easy-to-use and highly scalable TO framework implemented in FreeFEM-PETSc/SLEPc-Mmg/ParMmg;

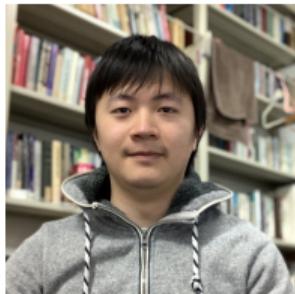
② Future work

- ▶ Forced convection problem;
- ▶ Fluid-to-fluid heat exchanger;
- ▶ Acoustic metamaterial.

③ Wish list

- ▶ Automatic differential toolbox;
- ▶ Turbulent flow solver.

Acknowledgment



N. Nakayama



K. Wano



N. Ishida



K. Furuta



T. Yamada



P. Jolivet



T. Kondoh



S. Nishiwaki



Thank you for your kind attention!

Q & A

APPENDIX

Augmented Lagrangian for inequality constraint problem

Lagrange multiplier λ_i associated with the i -th constraint G_i is updated as follows:

$$\lambda_i^{n+1} = \lambda_i^n + c_1 G_i^n + c_2 (G_i^{n-1} - G_i^n), \quad (41)$$

where c_1 and c_2 are the positive constant.

Body-fitted mesh

Body-fitted meshing algorithm generates polyhedrons and other types of elements, i.e., tetrahedrons in our case, at the **intersection of Cartesian cells with the geometrical boundary**.

Our numerical implementation of this mesh evolution algorithm relies on a sequential version of **Mmg2d** and **Mmg3d**⁶.

(<https://www.mmgtools.org/>)

⁶Dapogny, C., Dobrzynski, C., & Frey, P. (2014). Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *Journal of Computational Physics*, 262, 358–378.

Anisotropic mesh

Anisotropic meshes are those that introduce narrow mesh elements in the regions where the solutions are expected to change rapidly. The basic idea of this mesh evolution strategy is to use the **Delaunay algorithm** (George & Borouchaki, 1998) to generate a new triangular (2D) or tetrahedral (3D) mesh whose edges are close to the unit length in the metric $M = \frac{|\mathcal{H}|}{\varepsilon}$, where $|\mathcal{H}|$ is the Hessian of the variable (i.e., velocity components v_i) at point x and ε the interpolation error for the variables. The algorithm details can be found in Frey and George, 1999.

See also: Borouchaki et al., 1996; Castro-Diaz et al., 2000; George and Borouchaki, 1998; Hecht and Mohammadi, 1997; Mohammadi and Pironneau, 2010.

Our implementation relies on: 1) centralized version of **ParMmg**⁷; or 2) sequential version of **Mmg3d**.

⁷Cirrottola, L., & Froehly, A. (2021). Parallel Unstructured Mesh Adaptation Based on Iterative Remeshing and Repartitioning. <https://hal.inria.fr/hal-03208569>.

Topological derivative–eigenvalue problem

$$D_T J^{1 \rightarrow 2} = - \left(\sum_{m=1}^n \frac{1}{\lambda^{(m)}} \right)^{-2} \left(\sum_{m=1}^n \frac{D_T \lambda^{(m)1 \rightarrow 2}}{(\lambda^{(m)})^2} \right) \quad (42)$$

$$\begin{aligned} D_T \lambda^{(m)1 \rightarrow 2} &= \frac{2\mu^1(\kappa^1 + 1) \left\{ \mu^2(\kappa^1 - 1) - \mu^1(\kappa^2 - 1) \right\}}{(\mu^1(\kappa^2 - 1) + 2\mu^2)(\kappa^1 - 1)^2} \left\{ \frac{\partial u_x^{(m)}}{\partial x} + \frac{\partial u_y^{(m)}}{\partial y} \right\}^2 \\ &\quad + \frac{\mu^1(\mu^2 - \mu^1)(\kappa^1 + 1)}{\mu^1 + \kappa^1 \mu^2} \left\{ \frac{\partial u_x^{(m)}}{\partial x} - \frac{\partial u_y^{(m)}}{\partial y} \right\}^2 \\ &\quad + \frac{\mu^1(\mu^2 - \mu^1)(\kappa^1 + 1)}{\mu^1 + \kappa^1 \mu^2} \left\{ \frac{\partial u_x^{(m)}}{\partial y} + \frac{\partial u_y^{(m)}}{\partial x} \right\}^2 \\ &\quad - \lambda^{(m)} \left(\rho^2 - \rho^1 \right) |\mathbf{u}^{(m)}|^2 \end{aligned} \quad (43)$$

Runtime breakdown–3D eigenvalue problem

Actions	Runtim (s)
Solve governing equations	142.1
Compute sensitivity and Lagrange multiplier	304.7
Solve RDE	6.30
Centralize solution to process #0	0.13
Remesh	44.7
Partition updated mesh	11.5
Interpolate solution to updated mesh	1.28
Total	510.7

- $9.62 \cdot 10^5$ unknowns, $1.88 \cdot 10^6$ tetrahedrons;
- 8 cores, Intel(R) Xeon(R) W CPU @ 3.20GHz, Memory size is 32 GB.

References I

- Borouchaki, H., Castro-Diaz, M., George, P. L., Hecht, F., & Mohammadi, B. (1996). Anisotropic adaptive mesh generation in two dimensions for cfd. 197–206.
- Castro-Diaz, M., Hecht, F., & Mohammadi, B. (2000). Anisotropic grid adaptation for inviscid and viscous flows simulations. *International Journal of Computational Fluid Dynamics*, 25, 475–491.
- Cirrottola, L., & Froehly, A. (2021). Parallel Unstructured Mesh Adaptation Based on Iterative Remershing and Repartitioning.
<https://hal.inria.fr/hal-03208569>
- Dapogny, C., Dobrzynski, C., & Frey, P. (2014). Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *Journal of Computational Physics*, 262, 358–378.
- Frey, P. J., & George, P.-L. (1999). *Maillages: Applications aux éléments finis*. Hermès Science Publications.

References II

- George, P.-L., & Borouchaki, H. (1998). Delaunay triangulation and meshing. 1. vyd. paris. *Hermes*.
- Hecht, F., & Mohammadi, B. (1997). Mesh adaption by metric control for multi-scale phenomena and turbulence. 859.
- Kondoh, T., Matsumori, T., & Kawamoto, A. (2012). Drag minimization and lift maximization in laminar flows via topology optimization employing simple objective function expressions based on body force integration. *Structural and Multidisciplinary Optimization*, 45(5), 693–701.
- Mohammadi, B., & Pironneau, O. (2010). *Applied shape optimization for fluids*. Oxford university press.
- Moulin, J., Jolivet, P., & Marquet, O. (2019). Augmented lagrangian preconditioner for large-scale hydrodynamic stability analysis. *Computer Methods in Applied Mechanics and Engineering*, 351, 718–743.
- Novotny, A. A., Feijóo, R. A., Taroco, E., & Padra, C. (2003). Topological sensitivity analysis. *Computer methods in applied mechanics and engineering*, 192(7-8), 803–829.

References III

- Paroncini, M., & Corvaro, F. (2009). Natural convection in a square enclosure with a hot source. *International journal of thermal sciences*, 48(9), 1683–1695.
- Yamada, T., Izui, K., Nishiwaki, S., & Takezawa, A. (2010). A topology optimization method based on the level set method incorporating a fictitious interface energy. *Computer Methods in Applied Mechanics and Engineering*, 199(45-48), 2876–2891.

Paper list

- "Topology optimization for eigenfrequency maximization problems based on the multi-material level-set method", under preparation.
- "Topology optimization for lift-drag problems incorporated with anisotropic adaptive mesh", a short note, under preparation.
- Li, H.* Kondoh, T., Jolivet, P.* Furuta, K., Yamada, T., Zhu, B., Zhang, H., Izui, K., & Nishiwaki S*. "Optimum design and thermal modeling for 2D and 3D natural convection problems incorporating level set-based topology optimization with body-fitted mesh." Preprint submitted to International Journal for Numerical Methods in Engineering (Wiley).
- Li, H.* Kondoh, T., Jolivet, P.* Furuta, K., Yamada, T., Zhu, B., Izui, K., & Nishiwaki S*. "Three-dimensional topology optimization of fluid-structure system using body-fitted mesh adaption based on the level-set method." *Applied Mathematical Modelling* 101 (2022): 276–308. <https://doi.org/10.1016/j.apm.2021.08.021>
- Li, H.* Yamada, T., Jolivet, P.* Furuta, K., Kondoh, T., Izui, K., & Nishiwaki, S.* "Full-scale 3D structural topology optimization using adaptive mesh refinement based on the level-set method." *Finite Elements in Analysis and Design* 194 (2021): 103561. <https://doi.org/10.1016/j.finel.2021.103561>
- Li, H., Ding, X.* Jing D., Xiong, M., & Meng, F. "Experimental and numerical investigation of liquid-cooled heat sinks designed by topology optimization." *International Journal of Thermal Science* 146 (2019): 106065. <https://doi.org/10.1016/j.ijthermalsci.2019.106065>
- Li, H., Ding, X.* Meng, F., Jing D., & Xiong, M. "Optimal design and thermal modelling for liquid-cooled heat sink based on multi-objective topology optimization: An experimental and numerical study." *International Journal of Heat and Mass Transfer* 144 (2019): 118638. <https://doi.org/10.1016/j.ijheatmasstransfer.2019.118638>