

# YETI : YET another Iga code

A. Duval<sup>1</sup>, M. Guerder<sup>1</sup>, T. Hirschler<sup>2</sup>, T. Elguedj<sup>1</sup>

<sup>1</sup> Univ Lyon, CNRS, INSA Lyon, LaMCoS, UMR5259, 69621 Villeurbanne, France, [arnaud.duval@insa-lyon.fr](mailto:arnaud.duval@insa-lyon.fr)

<sup>2</sup> Institute of Mathematics, Chair of Numerical Modeling and Simulation, Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland, [thibaut.hirschler@epfl.ch](mailto:thibaut.hirschler@epfl.ch)

**Résumé** — YETI est une bibliothèque Python publiée sous licence LGPL et proposant une implémentation de l'analyse isogéométrique basée sur les NURBS et B-splines. Ses principales fonctionnalités sont dédiées à l'analyse mécanique et à l'optimisation de forme de structures solides ou coques. Une représentation immergée des structures permet de modéliser efficacement des géométries de type raidisseur, congé ou raccord dans l'optique d'une modification de forme.

**Mots clés** — analyse isogéométrique, optimisation, couplage, logiciel libre.

## 1 Introduction

L'analyse isogéométrique se base sur les primitives géométriques utilisées en conception assistée par ordinateur (NURBS, T-splines, etc.) pour décrire la solution d'un problème aux dérivées partielles. Elle permet une modélisation exacte de la géométrie et propose des méthodes de raffinement de maillage ne dégradant pas cette dernière [1]. Ce lien fort entre modélisation géométrique et analyse rend cette méthode intéressante pour traiter certaines classes de problèmes d'optimisation de forme.

YETI est une bibliothèque Python *open source* proposant une implémentation de cette méthode. Différentes fonctionnalités sont proposées telles que l'analyse de structures solides ou coques, l'optimisation de forme avec calcul analytique des sensibilités dans le cas de problèmes auto-adjoints, le couplage de patches *via* une méthode *mortar* ainsi que la modélisation immergée, facilitant la prise en compte et l'optimisation de raidisseurs ou congés.

## 2 Architecture

YETI se présente sous la forme d'une bibliothèque Python permettant à l'utilisateur de :

- lire des fichiers d'entrée correspondant à la paramétrisation NURBS d'un problème de mécanique (syntaxe inspirée des fichiers d'entrée Abaqus);
- raffiner la paramétrisation;
- coupler différents domaines;
- spécifier des conditions aux limites;
- calculer les matrices de masse, raideur et second membre du problème linéaire à résoudre;
- calculer les sensibilités de différentes grandeurs d'intérêt (compliance discrète, volume, fréquence propre, déplacement, etc.) utilisables dans une boucle d'optimisation;
- projeter la solution obtenue sur un maillage lisible par Paraview.

Les routines de plus bas niveau sont écrites en Fortran 95 et une interface en Python est réalisée à l'aide de l'utilitaire `f2py` [2]. L'utilisateur peut ensuite utiliser les bibliothèques Python `numpy`, `scipy`, `nlopt`, etc. pour la résolution de systèmes linéaires ou l'optimisation.

La configuration du code est faite avec CMake [3] et un jeu de cas tests a été mis en place pour valider la non-régression du code et le qualifier au regard des résultats de la littérature.

### 3 Fonctionnalités et applications

#### 3.1 Analyse

La fonctionnalité de base de YETI consiste à calculer les matrices de masse, de raideur et le second membre pour une structure à comportement élastique linéaire. Des éléments ont été implémentés afin de pouvoir traiter les cas de solides 2D ou 3D et les coques de Kirchhoff-Love. Les chargements peuvent être de type déplacement imposé, effort surfacique (uniforme ou non) et effort volumique. La figure 1 illustre le cas d'une pale de turboréacteur soumise à un effort centrifuge et à un champ de pression non uniforme issu d'un calcul de dynamique des fluides.

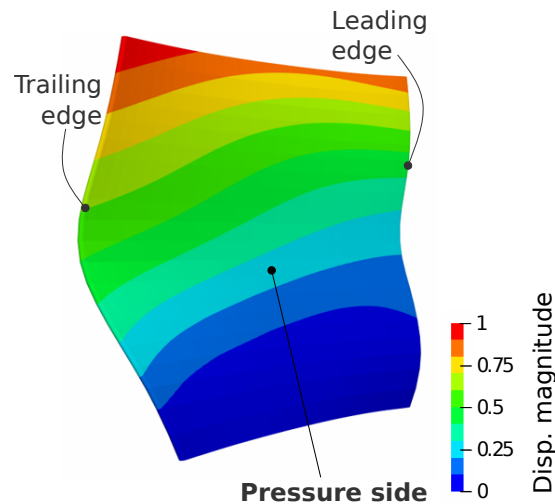


FIGURE 1 – Champ de déplacement dans une pale de turboréacteur soumise à un effort centrifuge et un champ de pression non uniforme (d'après [4]).

#### 3.2 Optimisation

YETI permet de traiter notamment les cas d'optimisation de forme pour les problèmes de mécanique auto-adjoints (élasticité linéaire avec des fonctionnelles d'optimisation faisant intervenir volume, compliance, déplacement, contrainte ou valeur propre et dont les variables de design sont reliées à la position des points de contrôle de la géométrie). L'optimisation est faite à partir de deux modèles :

- un modèle d'optimisation pour lequel les coordonnées des points de contrôle sont pilotées par les variables de design du problème d'optimisation ;
- un modèle d'analyse, généralement plus fin que le modèle d'optimisation afin d'obtenir une bonne qualité de solution.

Une boucle type d'un algorithme d'optimisation à gradient démarre du modèle d'optimisation qui est ensuite raffiné pour obtenir le modèle d'analyse. Sur ce dernier sont calculées la solution du problème linéaire et les sensibilités, qui sont ensuite agrégées sur le modèle d'optimisation. Le détail d'une telle boucle est montré à la figure 2. Les sensibilités sont calculées de façon exacte à partir d'une expression analytique permettant la décomposition en une partie propre à la représentation géométrique et une partie propre à la grandeur d'intérêt (détails de l'implémentation dans [5]). La figure 3 montre un exemple d'optimisation de forme d'une plaque trouée.

#### 3.3 Couplage

L'analyse isogéométrique basée sur les NURBS impose, de par sa structure en produit tensoriel, d'avoir des éléments géométriques élémentaires (les patches) homéomorphes à des carrés en 2D et des cubes en 3D. Ceci impose de coupler plusieurs patches ensemble pour représenter une géométrie complexe (exemple en figure 4). Dans le cas d'un couplage fort (les patches voisins partagent les mêmes points de contrôle). Ceci limite fortement le raffinement du maillage car, celui-ci étant structuré, le raffinement doit se propager de patch en patch pour assurer la compatibilité de maillage nécessaire à un couplage

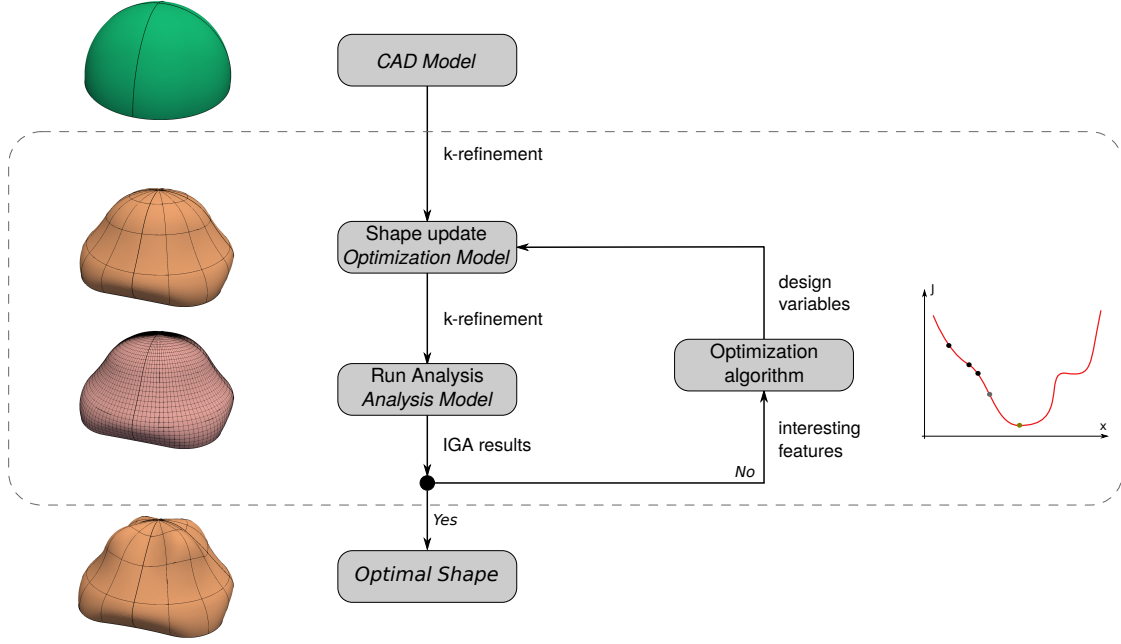


FIGURE 2 – Représentation schématique d’un processus d’optimisation de forme par analyse isogeométrique (d’après [7]).

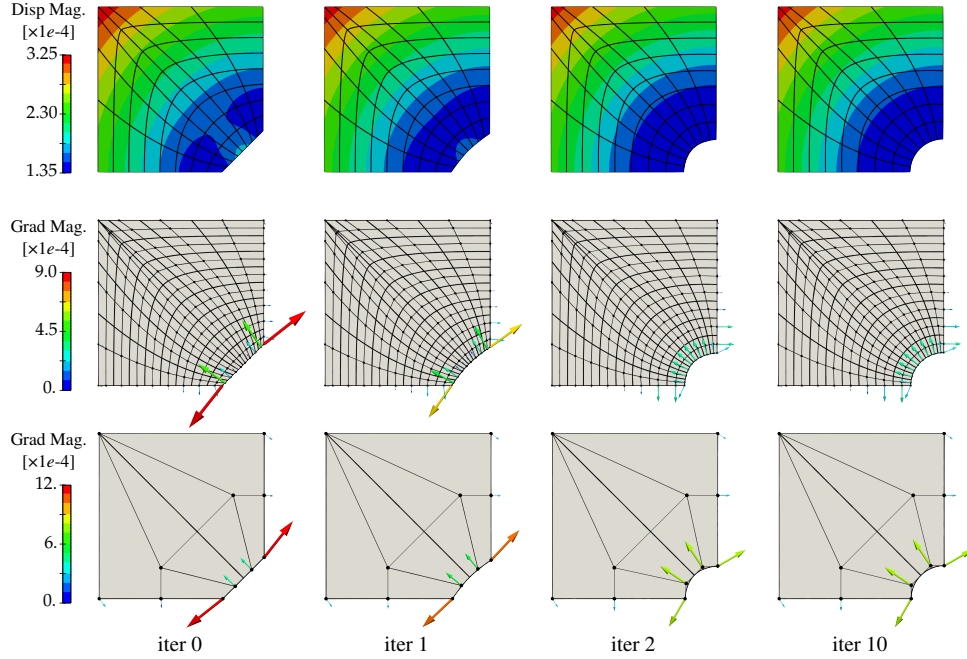


FIGURE 3 – Historique de l’optimisation de forme d’un quart de plaque trouée soumise à une traction bi-axiale. Le problème d’optimisation a pour objectif de minimiser la compliance à volume constant. Les variables de design sont reliées aux coordonnées des points de contrôle définissant la forme du trou central. (d’après [5]).

fort. Pour y remédier, YETI implémente un couplage faible de type *mortar*. Pour ce faire, la contrainte d’égalité des déplacements (et rotations dans le cas de coques) à l’interface  $\Gamma$  entre deux domaines  $\Omega_1$  et  $\Omega_2$  est assurée *via* l’introduction de multiplicateurs de Lagrange, résultant en un problème linéaire sous la forme matricielle suivante :

$$\begin{bmatrix} K^{(1)} & 0 & C^{(1)T} & Z^{(1)T} \\ 0 & K^{(2)} & C^{(1)T} & Z^{(1)T} \\ C^{(1)} & C^{(2)} & 0 & 0 \\ Z^{(1)} & Z^{(2)} & 0 & 0 \end{bmatrix} \begin{pmatrix} u^{(1)} \\ u^{(2)} \\ \tilde{\Lambda} \\ \hat{\Lambda} \end{pmatrix} = \begin{pmatrix} f^{(1)} \\ f^{(2)} \\ 0 \\ 0 \end{pmatrix}$$

où les  $K^i$  sont les matrices de raideur des domaines,  $C^{(i)}$  et  $Z^{(i)}$  les termes de couplage pour respectivement les déplacements et les rotations,  $u^{(i)}$  les degrés de liberté mécaniques,  $\tilde{\Lambda}$  et  $\hat{\Lambda}$  les multiplicateurs de Lagrange et  $f^{(i)}$  les efforts extérieurs. La structure de ce type de système permet l'utilisation d'un solveur par décomposition de domaine de type FETI [8]. La figure 5 montre un exemple de résolution d'un problème faisant intervenir plusieurs patches coque non compatibles.

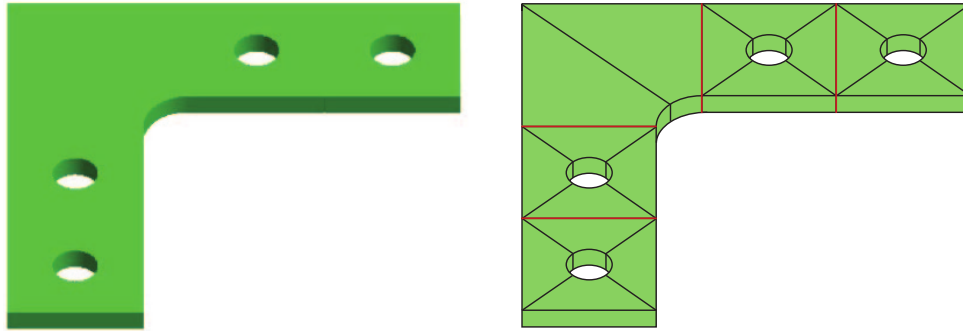


FIGURE 4 – Représentation d'une géométrie complexe à l'aide de 5 patches NURBS. Les frontières des patches sont représentées en rouge et celles des éléments en noir (d'après [6]).

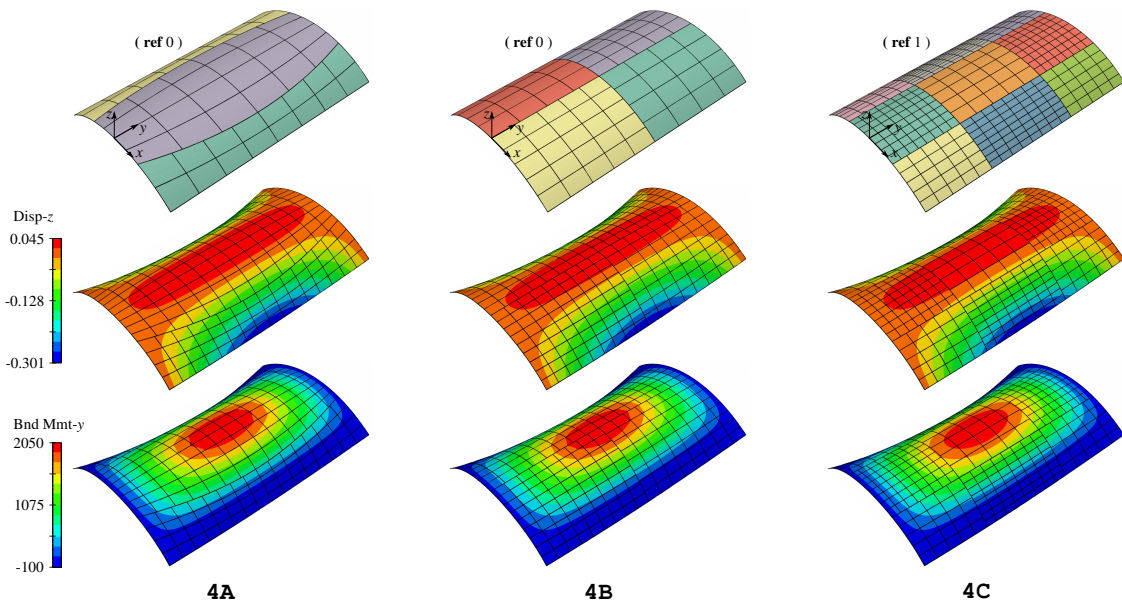


FIGURE 5 – Cas test du toit de Scordelis-Lo. Trois décompositions de domaine différentes présentant des patches non compatibles sont proposées. Les champs de déplacement et de moment de flexion sont affichés pour chacune des configurations (d'après [8]).

### 3.4 Modélisation immergée

Le couplage de domaines non compatibles présenté dans la section précédente peut être utilisé pour représenter des géométries telles que des raidisseurs ou des congés d'arête. Pour utiliser un tel modèle dans le cadre de l'optimisation de forme, il est nécessaire de pouvoir assurer la bonne connexion des différents patches lors d'une modification de forme. La figure 6 illustre ce type de problème dans le cas d'une portion de cylindre munie d'un raidisseur pour lequel on modifie soit la forme du cylindre, soit la forme du raidisseur. La solution proposée dans YETI pour ce cas consiste à immerger le raidisseur dans un volume NURBS, lui-même issu de la surface représentant le cylindre. Cette modélisation est illustrée à la figure 7. Ainsi, la représentation du raidisseur dans l'espace physique est issue de la composition de la paramétrisation du raidisseur et de la paramétrisation de son enveloppe. La modification de forme du raidisseur ou du cylindre se fait ainsi en garantissant une bonne connexion, nécessaire au couplage des

deux domaines. Cette fonctionnalité a été implémentée dans YETI pour les coques de Kirchhoff-Love et pour les solides.

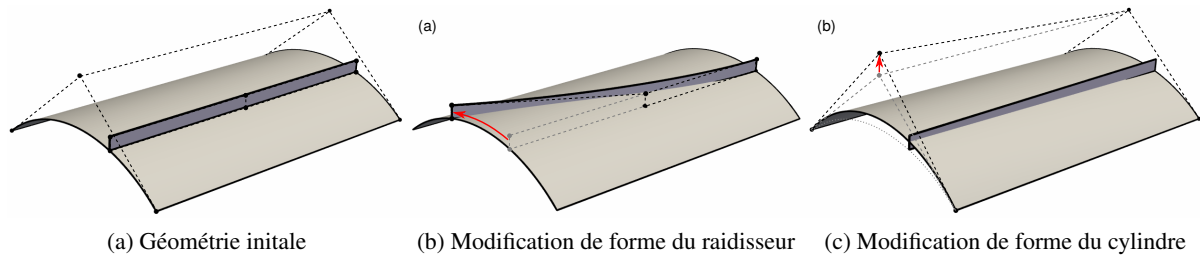


FIGURE 6 – Portion de cylindre munie d’un raidisseur. La modification de forme du raidisseur ou du cylindre entraîne la déconnexion des deux patches. (d’après [9]).

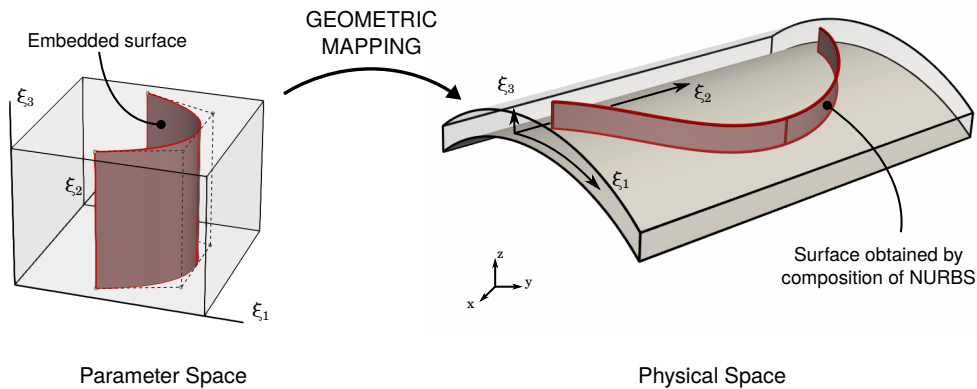


FIGURE 7 – Construction d’une structure raidie *via* une surface immergée. La surface dans l’espace physique est obtenue par composition d’une surface NURBS et d’un volume NURBS, assurant une connexion parfaite entre les deux patches (d’après [9]).

La figure 8 illustre cette fonctionnalité dans le cas d’une aile d’avion composée de coques constituant l’intrados et l’extrados et de coques immergées constituant des raidisseurs internes. Un tel modèle peut être utilisé dans le cadre d’un problème d’optimisation visant à trouver la forme des raidisseurs permettant de minimiser la flèche de l’aile soumise à un chargement en pression.

## 4 Évolutions futures

Différentes fonctionnalités sont en cours d’implémentation dans YETI.

### 4.1 Élasticité à gradient

L’utilisation des fonctions NURBS en analyse isogéométrique permet de modéliser des champs de solution de degré et de continuité élevés. Ceci permet l’implémentation de lois de comportement élastiques à gradient telles que celle proposée par Mindlin [11] faisant intervenir le second gradient de la déformation, soit la dérivée troisième des déplacements.

### 4.2 Quadrature pondérée et méthodes sans matrice

L’utilisation de fonction B-splines de degré élevée (jusqu’à 10) fait augmenter drastiquement le nombre de points d’intégration de la quadrature de Gauss-Legendre standard, augmentant de ce fait le temps de calcul. La quadrature pondérée [12] permet de réduire le nombre de points d’intégration jusqu’à deux par direction paramétrique et par élément au cœur du domaine. Cette dernière se construit en exploitant la répétition des motifs des fonction B-splines pour construire une règle de quadrature dont

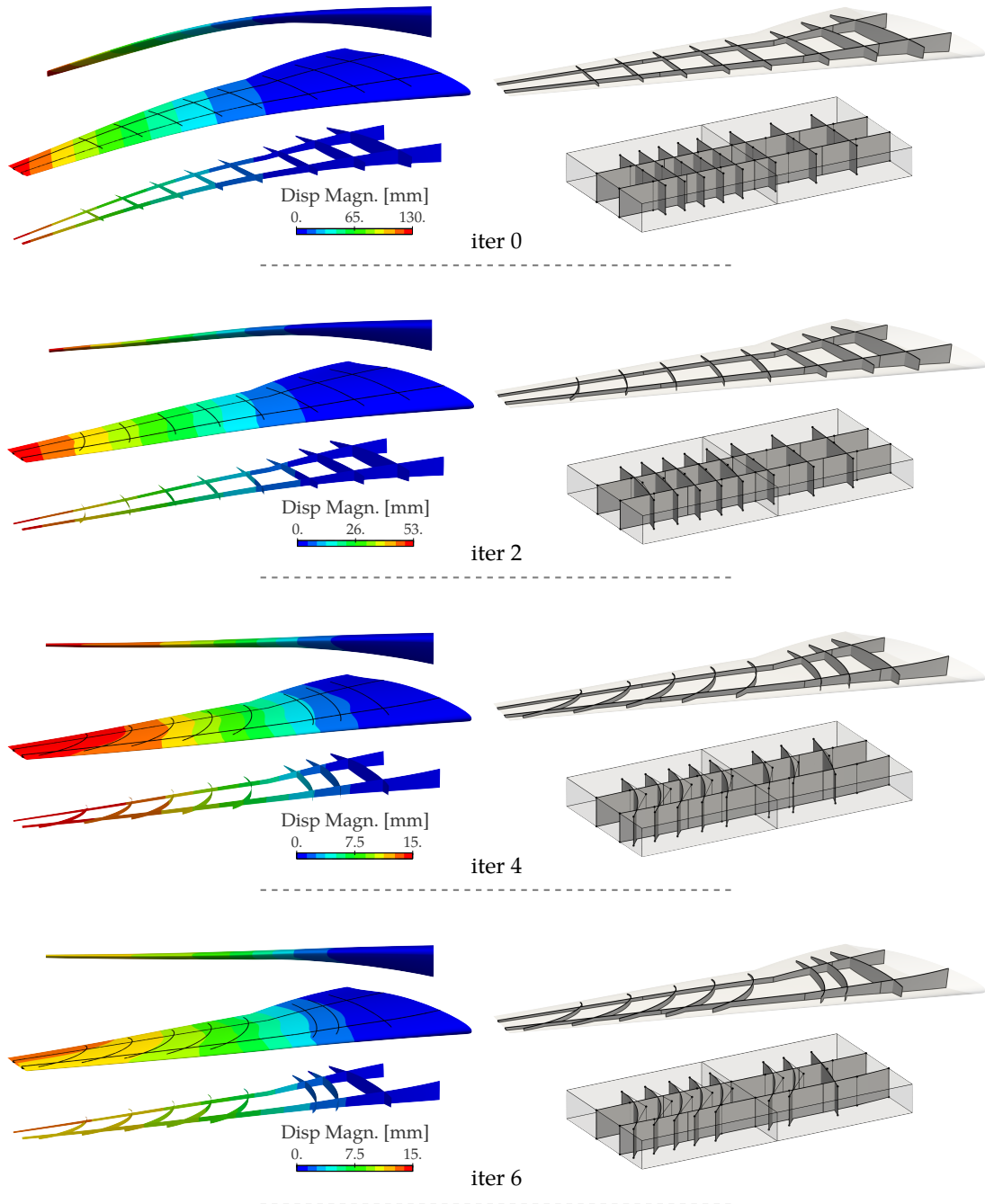


FIGURE 8 – Optimisation de forme d’une aile d’avion soumise à un chargement en pression. On cherche la forme des raidisseurs permettant de minimiser la flèche en bout d’aile. Les raidisseurs sont modélisés sous la forme de coques immergées (d’après [10]).

les poids sont pré-calculés en pré-traitement. Couplée à des méthodes sans matrice [13] pour lesquelles il n’est plus nécessaire de former les matrices du système à résoudre, ceci permet de réduire le temps de calcul jusqu’à trois ordres de grandeur comparativement aux éléments finis classiques.

## 5 Conclusion

Développé depuis 2015 au LaMCoS, YETI est publié sous licence LGPL à compter de 2022 [14]. L’objectif est d’en faire une plateforme de recherche ouverte à la communauté pour l’implémentation de méthodes basées sur l’analyse isogéométrique. Sa licence libre non contaminante permet également de l’utiliser comme composant d’outils métier dans un cadre industriel.

## Références

- [1] T.J.R. Hugues, J.A. Cottrell, Y. Bazilevs. *Isogeometric analysis : CAD, finite element, NURBS, exact geometry and mesh refinement*, Computer Methods in Applied Mechanics and Engineering 194, 4135-4195, 2005.
- [2] F2PY user guide and reference manual, <https://numpy.org/doc/stable/f2py/>
- [3] CMake Reference Documentation <https://cmake.org/documentation/>
- [4] M. Guerder, A. Duval, T. Elguedj, P. Feliot, J. Touzeau. *Isogeometric shape optimisation of volumetric blades for aircraft engines*, Structural and Multidisciplinary Optimization, accepted.
- [5] T. Hirschler, R. Bouclier, A. Duval, ; T. Elguedj, J. Morlier. *A New Lighting on Analytical Discrete Sensitivities in the Context of IsoGeometric Shape Optimization*, Archives of Computational Methods in Engineering 28, 2371-2408, 2021.
- [6] J.A. Cottrell, T.J.R. Hugues, Y. Bazilevs. *Isogeometric Analysis : Toward Integration of CAD and FEA*, Wiley, 2009.
- [7] T. Hirschler, R. Bouclier, A. Duval, T. Elguedj, J. Morlier. *Isogeometric sizing and shape optimization of thin structures with a solid-shell approach*, Structural and Multidisciplinary Optimization 59, 767-785, 2019.
- [8] T. Hirschler, R. Bouclier, D. Dureisseix, A. Duval, T. Elguedj, J. Morlier. *A dual domain decomposition algorithm for the analysis of non-conforming isogeometric Kirchhoff-Love shells*, Computer Methods in Applied Mechanics and Engineering 357, 2019.
- [9] T. Hirschler, R. Bouclier, A. Duval, T. Elguedj, J. Morlier. *The embedded isogeometric Kirchhoff-Love shell : from design to shape optimization of non-conforming stiffened multipatch structures*, Computer Methods in Applied Mechanics and Engineering 349, 774-797, 2019.
- [10] T. Hirschler. *Isogeometric modeling for the optimal design of aerostructures*, thèse de doctorat de l'université de Lyon, 2019
- [11] R.D. Mindlin. *Second gradient of strain and surface-tension in linear elasticity*, International Journal of Solids and Structures 4, 417-438, 1965.
- [12] F. Calabrò, G. Sangalli, M. Tani. *Fast formation of isogeometric Galerkin matrices by weighted quadrature*, Computer Methods in Applied Mechanics and Engineering 316, 606-622, 2017.
- [13] G. Sangalli, M. Tani. *Matrix-free weighted quadrature for a computationally efficient isogeometric k-method*, Computer Methods in Applied Mechanics and Engineering 338, 117-133, 2018.
- [14] YETI, <https://lamcosplm.insa-lyon.fr/projects/yeti>