



**HAL**  
open science

## **Méthode d'implémentation non-intrusive de l'analyse isogéométrique dans un code éléments finis**

Marie Tirvaudey, Robin Bouclier, Jean-Charles Passieux, Ludovic Chamoin, Cuong  
Ha Minh

### ► **To cite this version:**

Marie Tirvaudey, Robin Bouclier, Jean-Charles Passieux, Ludovic Chamoin, Cuong Ha Minh. Méthode d'implémentation non-intrusive de l'analyse isogéométrique dans un code éléments finis. 23e Congrès Français de Mécanique (CFM 2017), Aug 2017, Lille, France. <hal-01584833>

**HAL Id: hal-01584833**

**<https://hal.science/hal-01584833v1>**

Submitted on 4 Mar 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# Méthode d'implémentation non intrusive de l'analyse isogéométrique dans un code élément fini

**M. TIRVAUDEY**<sup>a,b</sup>, **R. BOUCLIER**<sup>c</sup>, **J-C. PASSIEUX**<sup>b</sup>, **L. CHAMOIN**<sup>a</sup>,  
**C. HA-MINH**<sup>a</sup>

a. LMT Cachan / ENS Paris-Saclay / CNRS / Université Paris-Saclay

{ tirvaudey,chamoin,haminh } @lmt.ens-cachan.fr

b. Institut Clément Ader (ICA), Université de Toulouse, CNRS - INSA - ISAE - Mines Albi - UPS,  
Toulouse, France

passieux@insa-toulouse.fr

c. Institut de Mathématiques de Toulouse (IMT), Université de Toulouse, CNRS - UPS - INSA,  
Toulouse, France bouclier@insa-toulouse.fr

## Résumé :

*La méthode isogéométrique présente des avantages quant à son lien avec la CAO. Afin de pouvoir utiliser des codes préexistants donc généralement basés sur une structure élémentaire avec cette méthode, nous présentons une "réinterprétation" de l'extraction de Bézier. En effet à partir du maillage NURBS il est possible d'en extraire un maillage à structure élémentaire sur lequel des calculs éléments finis peuvent être lancés, en modifiant uniquement les fonctions de forme utilisées. Nous avons ensuite mis en place une projection entre les bases éléments finis et les bases NURBS afin de bien récupérer les résultats isogéométrique. Un exemple numérique permet de valider la performance de cette projection.*

## Abstract :

*La même en anglais*

**Mots clefs : Analyse isogéométrique, couplage non-intrusif.**

## 1 Introduction

L'utilisation des matériaux composites est devenue un des enjeux technologiques majeurs dans l'industrie. Une des tendances actuelle vise à remplacer une partie des essais par des simulations numériques afin de réduire les coûts et temps de développement de nouveaux produits. Parmi les méthodes numériques les plus avancées, l'analyse isogéométrique (IGA) permet de faire un pas de plus dans cette voie en ce sens qu'elle permet d'utiliser les fonctions de la CAO directement pour faire le calcul. Initiée dans [1], cette méthode a fortement attiré l'attention de la communauté scientifique et l'industrie, et elle a fait l'objet de nombreux travaux sur des applications diverses en Calcul de Structures [2]. L'analyse

isogéométrique facilite donc la représentation de la géométrie, en permettant une description géométrique exacte ce qui pourrait avoir un intérêt pour la modélisation de l'endommagement des composites stratifiés.

En effet les modèles les plus prédictifs, comme le mésomodèle d'endommagement du LMT [3] actuellement implémenté dans le code de calcul éléments finis SAMCEF, nécessite un maillage fin pour pouvoir capter la complexité de la physique du matériau. L'utilisation de l'IGA pourrait permettre de réduire le coût numérique du au grand nombre de degrés de liberté tout en conservant un espace d'approximation riche.

Quelques études sur la propagation du délaminage dans les matériaux composites en IGA ont déjà été réalisées en utilisant par exemple des zones cohésives isogéométriques [4, 5, 6]. Cependant l'implémentation de l'IGA dans un tel code structuré autour des éléments finis n'est pas direct. Récemment, Borden a développé un outil [7] faisant le lien entre l'analyse isogéométrique et élément fini. Cet outil : l'extraction de Bézier permet de créer une structure élémentaire de l'analyse isogéométrique qui permet de s'intégrer dans des codes classiques en modifiant uniquement les fonctions de forme utilisées.

## 2 Courte introduction sur l'analyse isogéométrique

Afin de mieux comprendre la mise en place de l'extraction de Bézier, une courte introduction sur la construction des B-splines et des NURBS est faite dans cette partie. Pour plus de détails se référer à [1, 8].

### 2.1 Vecteur de nœuds et B-Splines

Les fonctions B-Splines sont définies à l'aide d'un vecteur noeud :  $\Xi = \{\xi_1, \xi_2, \dots, \xi_{n+p+1}\}$ . Il est constitué de coordonnées paramétriques  $\xi_i$  rangées dans l'ordre croissant,  $p$  étant le degré polynomial de la B-Spline et  $n$  le nombre de fonctions associées.

La  $i$ -ème fonction de base B-Spline de degré  $p$  est définie par  $N_{i,p}$  de façon récursive en utilisant la formule de Cox-de Boor [9] :

— pour  $p = 0$  :

$$N_{i,0}(\xi) = \begin{cases} 1 & \text{si } \xi_i \leq \xi < \xi_{i+1} \\ 0 & \text{sinon.} \end{cases}$$

— pour  $p = 1, 2 \dots n$  on construit :

$$N_{i,p} = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi)$$

Les cas étudiés seront principalement d'ordre  $p$  supérieur ou égal à 2 afin que les fonctions de forme se différencient de celles des éléments finis classiques. En général, une fonction d'ordre  $p$  aura  $p - 1$  dérivées continues. Par ailleurs, si un nœud  $\xi_i$  a une multiplicité  $m_i$ , alors le nombre de dérivées continues diminue de  $m_i$ .

Ces fonctions B-splines servent à définir les fonctions NURBS :

$$R_i^p(\xi) = \frac{N_{i,p}(\xi)w_i}{\sum_{j=1}^n N_{j,p}(\xi)w_j}$$

où les  $w_i$  sont des poids propres à chaque point de contrôle de la NURBS dans  $\mathbb{R}^d$ .

Les courbes B-Spline ou NURBS sont alors définies à l'aide d'un jeu de fonctions de forme  $N_{i,p}$ , respectivement  $R_i^p$ , associées à des points de contrôle  $P_i$  de la façon suivante :

$$C(\xi) = \sum_{i=0}^n N_{i,p}(\xi)P_i \quad \text{respectivement} \quad C(\xi) = \sum_{i=0}^n R_i^p(\xi)P_i$$

## 2.2 L'analyse isogéométrique en pratique

L'idée principale de l'analyse isogéométrique est de modéliser exactement la géométrie avec des fonctions qui vont servir à approximer la solution. La figure 1 montre la différence majeure entre les deux méthodes : pour la FEM (méthode des éléments finis) la base d'approximation va donner la géométrie tandis qu'avec la méthode IGA la base d'approximation découle directement de la géométrie.

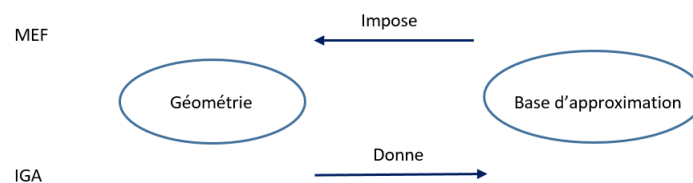


FIGURE 1 – Différence IGA/FEM

Même si la philosophie du calcul reste similaire à une étude éléments finis, de grosses modifications sont à prévoir dans la routine même du code en particulier car les points de contrôles analogues aux noeuds ne sont pas forcément interpolant et les fonctions de formes NURBS sont définies de façon globales.

## 3 Un lien entre l'IGA et la MEF : la transformation de Bézier

La construction d'éléments Bézier isogéométriques ainsi que celle de l'opérateur extraction de Bézier a été mise en place par Borden [7] afin de disposer d'une structure élémentaire pour l'analyse isogéométrique. Cette structure a pour but de pouvoir s'intégrer facilement dans des codes éléments finis classiques déjà existants.

### 3.1 Les courbes de Bézier et les polynômes de Bernstein

La transformation développée par Borden, repose sur une définition ancienne des courbes de Bézier [10]. Une courbe de Bézier d'ordre  $p$  est une combinaison linéaire de fonctions de Bernstein  $\mathbf{B}(\xi) = \{B_{a,p}(\xi)\}_{a=1}^{p+1}$  associées à des points de contrôle  $\mathbf{P} = \{P_{a,p}\}_{a=1}^{p+1}$ . La courbe est ainsi définie par :

$$C(\xi) = \sum_{a=1}^{p+1} P_a B_{a,p}(\xi) \quad \xi \in [0, 1]$$

Les polynômes de Bernstein sont quant à eux construits par récurrence pour  $\xi \in [0, 1]$  :

$$\begin{cases} B_{a,p}(\xi) = (1 - \xi)B_{a,p-1}(\xi) + \xi B_{a-1,p-1} \\ B_{1,0} \equiv 1 \\ B_{a,p}(\xi) \equiv 0 \quad \text{if } a < 1 \text{ or } a > p + 1 \end{cases}$$

### 3.2 La décomposition de Bézier

Afin de passer d'un maillage isogéométrique à un maillage constitué d'éléments de Bézier, la décomposition de Bézier est utilisée. Le principe de cette décomposition est de partir du vecteur noeud définissant la géométrie et de répéter, à l'aide d'un  $h$ -raffinement chacun des noeuds internes afin qu'ils atteignent une multiplicité au moins  $p$ .

En effet en ajoutant un par un ces noeuds, la continuité va diminuer afin de devenir  $C^0$  aux noeuds internes. Les points de contrôle sont par ailleurs recalculés à chaque fois qu'un noeud est inséré afin de ne pas modifier la continuité de la courbe. Cette décomposition est illustrée sur la figure 2.

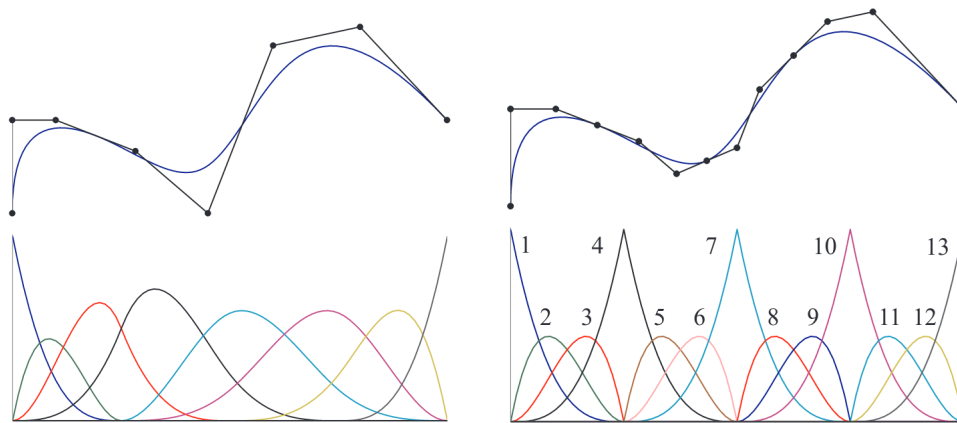


FIGURE 2 – D'une géométrie B-Spline (à gauche) à une géométrie Bézier (à droite) [7]

### 3.3 L'extraction de Bézier en pratique

Sur un élément de la géométrie on passe donc d'un ensemble de fonctions de forme B-Splines définies de façon globale à un ensemble de fonctions de Bernstein définies localement. La relation entre ces deux types de fonctions de forme est faite grâce à l'opérateur de Bézier  $\mathbf{C}$  de la façon suivante :

$$\mathbf{N}(\xi) = \mathbf{C}\mathbf{B}(\xi)$$

La construction de l'opérateur  $\mathbf{C}$  est détaillée dans l'article de Borden [7].

La géométrie Bézier précédemment créée ayant une structure élémentaire peut être intégrée dans des codes de calculs éléments finis existants en venant modifier les fonctions de formes afin d'utiliser les fonctions Bernstein. Il faut par ailleurs venir changer la table de connectivité entre les éléments afin de retrouver le maillage initial ce qui entraîne de lourdes modifications dans les fonctions d'assemblages des différents opérateurs  $\mathbf{K}$ ,  $F$ .

Afin de rendre cette extraction beaucoup plus non intrusive, le choix a été fait de faire la transformation de Bézier de façon globale en venant simplement modifier la sous-routine des fonctions de forme. Ainsi

un problème type :  $\mathbf{K}^B U^B = F^B$  peut être résolu sur la base des fonctions de Bernstein  $C^0$  dans un code élément fini classique.

Pour obtenir le résultat sur la base B-Spline il suffit de réaliser une projection sur cette base qui est incluse dans la base des fonctions de Bernstein avec l'opérateur de Bézier de la façon suivante :

$$\mathbf{K}^{IGA} = \mathbf{C}\mathbf{K}^B\mathbf{C}^T$$

$$F^{IGA} = \mathbf{C}F^B$$

$$U^{IGA} = \mathbf{C}U^B$$

Cette démarche peut s'étendre au cas des NURBS en faisant intervenir les poids :

$$\mathbf{R}^{IGA} = \mathbf{W}^{IGA} \frac{\mathbf{N}(\xi)}{W^{IGA}(\xi)}$$

où  $\mathbf{W}^{IGA}$  est une matrice diagonale des poids NURBS et  $W^{IGA}(\xi) = \sum_{i=1}^n w_i N_i = \mathbf{w}^T \mathbf{N}(\xi) = (\mathbf{C}^T \mathbf{w})^T \mathbf{B}(\xi) = W^B(\xi)$

Il en découle donc :

$$\mathbf{R}^{IGA} = \mathbf{W}^{IGA} \mathbf{C} (\mathbf{W}^B)^{-1} \mathbf{W}^B \frac{\mathbf{B}(\xi)}{W^B(\xi)} = \mathbf{W}^{IGA} \mathbf{C} (\mathbf{W}^B)^{-1} \mathbf{R}^B$$

En définissant un nouvel extracteur :  $\mathbf{C}_W = \mathbf{W}^{IGA} \mathbf{C} (\mathbf{W}^B)^{-1}$ , la résolution éléments finis peut se faire sur les fonctions de forme Bernstein pondérées puis être projetée sur l'espace NURBS :

$$\mathbf{K}^{IGA} = \mathbf{C}_W \mathbf{K}^B \mathbf{C}_W^T$$

## 4 Application numérique

Dans un premier temps, il a été choisi de réaliser un calcul en élasticité linéaire afin de valider les résultats pouvant être obtenus avec la transformation de Bézier par rapport à des références bibliographiques aussi bien en IGA qu'en FEM.

L'exemple choisi est une poutre circulaire en cisaillement. Les propriétés élastiques du matériau sont les suivantes :  $E = 10000$  et  $\nu = 0.25$ . Un déplacement constant  $u_0 = 0.01$  est appliqué sur le bord inférieur de la poutre comme illustré sur la partie gauche de la figure 3. Une solution analytique pour l'énergie de déformation se trouve dans la littérature [11].

Le maillage de contrôle de Bézier tracé sur la figure 3 à droite est une interpolation linéaire entre les points de contrôle de Bézier : les carrés. Ils sont bien plus nombreux que les points de contrôle NURBS, les étoiles, pour représenter la même géométrie.

L'évolution de l'énergie de déformation en fonction du nombre de degrés de liberté est tracée sur la figure 4. Les méthodes entièrement IGA et Bézier projetée sur IGA sont bien identiques. La courbe Bézier représente l'énergie calculée sur la base de Bézier sans la projeter sur la base NURBS ce qui explique une convergence similaire mais pour un nombre de degrés de liberté plus important. Le calcul a aussi été réalisé sur des bases de Lagrange. Avec des éléments Lagrange Q9, très similaires aux éléments de Bézier, l'erreur obtenue est légèrement plus grande pour le même nombre de degrés de

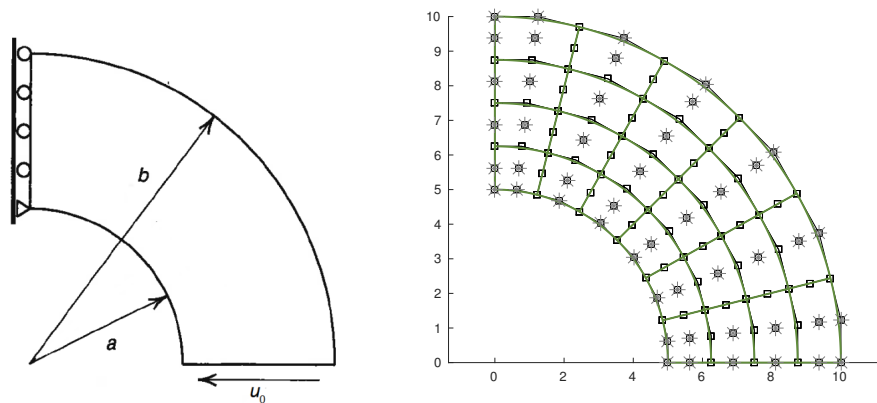


FIGURE 3 – Problème envisagé et un maillage type Bézier associé

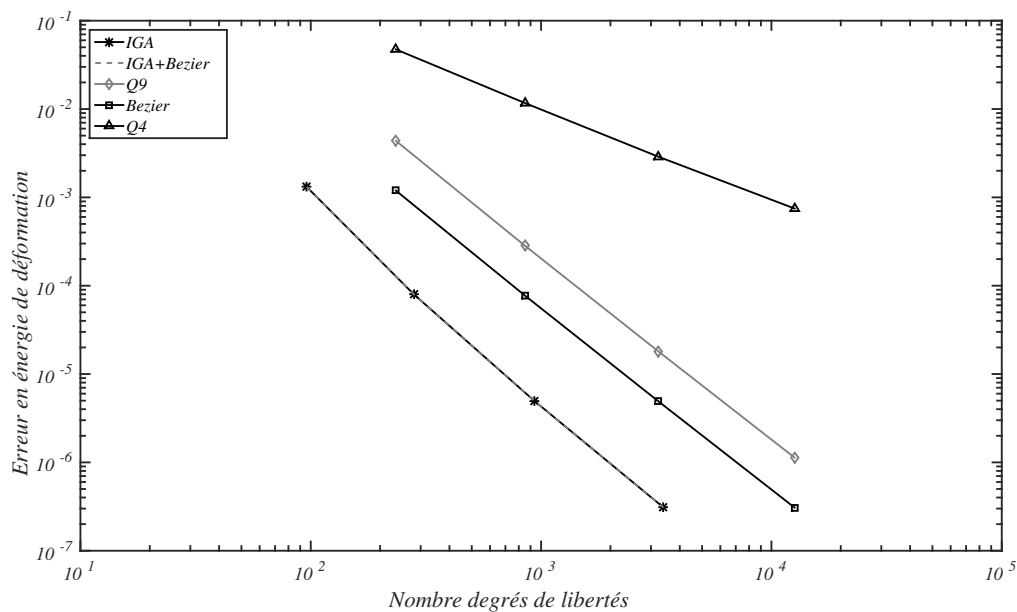


FIGURE 4 – Convergence de l'énergie de déformation en fonction du raffinement

liberté. En effet cette base  $n$ 'approxime pas exactement la géométrie comme le font les fonctions de Bernstein. Quant aux éléments Lagrange Q4, ils ne sont pas suffisants pour atteindre une convergence similaire à l'analyse isogéométrique car trop grossiers.

## 5 Conclusion

Ce cas d'étude a permis de mettre en place la méthode de l'extraction de Bézier de façon globale. Ainsi à partir d'un maillage NURBS initial, il est possible de créer une structure élémentaire permettant de s'intégrer dans des codes industriels préexistants. Les opérateurs créés dans un tel code éléments finis n'ont plus qu'à être projetés sur la base NURBS afin d'obtenir le résultat isogéométrique.

Cette méthode est en cours d'implémentation dans SAMCEF afin de pouvoir réaliser des calculs éléments finis de type propagation de fissure sur une géométrie NURBS sans venir modifier en profondeur le code déjà existant.

## Références

- [1] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis : CAD, finite elements, NURBS, exact geometry and mesh refinement, *Computer Methods in Applied Mechanics and Engineering*, 2005, pp.4135-4195.
- [2] V. Nguyen, C. Anitescu, S. Bordas, T. Rabczuk, Isogeometric analysis : an overview and computer implementation aspects, *Mathematics and Computers in Simulation*, 2015, V.117, pp.89-116.
- [3] P. Ladevèze, F. Daghia, E. Abisset, C. Mauff, A micromechanics-based interface mesomodel for virtual testing of laminated composites, *Advanced Modeling and Simulation in Engineering Sciences*, 2014, V.1, p.1-16.
- [4] V.Phu, P. Kerfriden, S.Bordas, Two- and three-dimensional isogeometric cohesive elements for composite delamination analysis, *Composites Part B*, 2014, V.60, pp.193-212.
- [5] S.Hosseini, J.Remmers, C.Verhoosel, R.Borst, Propagation of delamination in composite materials with isogeometric continuum shell elements, *International Journal for Numerical Methods in Engineering*, 2015, V.102, pp.159-179.
- [6] Y. Guo, M. Ruess, A layerwise isogeometric approach for NURBS-derived laminate composite shells, *Composite Structure*, 2015, V.124, pp.300-309.
- [7] M. Borden, M.Scott, J.Evan, T.J.R. Hughes, Isogeometric Finite Element Data Structures based on Bezier Extraction of NURBS by Isogeometric Finite Element Data Structures based on Bézier, ICES REPORT, 2008.
- [8] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs, *Isogeometric Analysis - Toward Integration of CAD and FEA*, 2009.
- [9] E. Cohen, T. Lyche, R. Riesenfeld, Discrete B-splines and subdivision techniques in computer-aided geometric design and computer graphics, *Computer Graphics and Image Processing*, 1980, V.14, pp.87-111.
- [10] P. Bézier, *Courbes & Surfaces*, Hermès, 1986.
- [11] O.C. Zienkewiez, R.L. Taylor, *The finite Element Mehtod - The Basis - Sixth Edition*, Butterworth-Heinemann, 2005, V.1.