

# Transformation entre représentations NURBS et Bézier

Transformation between NURBS and Bézier Representations

Khalfoun Rayan et Maxime Stauffert

**English Abstract**—NURBS are a popular choice used to model curves and surfaces with high precision. However, many visualization tools, like `ParaView`, work better with Bézier representations. In this work, we present a simple method to convert NURBS into standard Bézier segments using a matrix-based approach. A `Python` implementation is provided to enable accurate and efficient visualization of NURBS geometries

## 1 INTRODUCTION

Quand on modélise des courbes ou des surfaces sur ordinateur, surtout en 3D, on est rapidement confronté à la problématique principale de ce sujet qui est : comment représenter des formes complexes de façon à la fois précise, fluide, modifiable localement et tout en maintenant un coût de calcul raisonnable? Pour répondre à cela, les **B-splines**, plus précisément leur extension rationnelle, les **NURBS** (Non-Uniform Rational B-Splines), sont devenus les outils les plus utilisés dans l'industrie, notamment en **CAO** (Conception Assistée par Ordinateur), en animation. Le domaine de la simulation commence lui aussi à s'intéresser aux NURBS, on peut citer les travaux de **Régis Duvigneau** [1] qui a exploré l'utilisation des NURBS, notamment en les adaptant pour des méthodes de Galerkin discontinue.

Les NURBS permettent de représenter des formes très variées (courbes, arcs, surfaces lisses, coniques) à l'aide de points de contrôle, d'un vecteur de nœuds non uniforme, d'un degré et de poids rationnels. Mais dans certaines situations, notamment pour des besoins de visualisation ou d'analyse isogéométrique, il devient nécessaire de convertir une courbe ou une surface NURBS en une suite de segments de **Bézier**. Cela s'explique par le fait que les segments Bézier sont plus simples à manipuler : chaque segment est local, bien défini sur l'intervalle  $[0, 1]$ , et directement utilisable dans de nombreux outils numériques.

Plusieurs méthodes existent, mais dans ce sujet nous allons nous intéresser, en nous basant sur les travaux de **Lanlan Yan** [2], à une conversion qui repose sur une **matrice de changement de base** entre la base B-spline locale active sur un intervalle et la base de Bernstein. On appelle cette matrice **l'opérateur d'extraction de Bézier**. Jusqu'ici, pour

obtenir cet opérateur, il fallait souvent passer par des étapes lourdes, complexes et peu optimales.

Un exemple complet d'implémentation en `Python` est présenté, incluant la génération des matrices de conversion, la transformation des points de contrôle pondérés, et la visualisation des segments Bézier extraits d'une courbe NURBS.

L'objectif visé est de permettre une visualisation fidèle des NURBS dans `ParaView`, en s'appuyant sur une conversion en segments Bézier. Actuellement, cette visualisation est bien gérée dans `GLVis`, mais l'enjeu est d'étendre cette capacité à des outils plus puissants et standardisés comme `ParaView`.

## 2 CONCLUSION

Cette méthode de conversion NURBS vers Bézier offre une alternative efficace, locale et facile à mettre en œuvre pour représenter les formes complexes des NURBS tout en facilitant leur intégration dans les outils modernes de visualisation comme `ParaView`. Elle s'inscrit comme une solution prometteuse pour la visualisation d'ordre élevé.

## REFERENCES

- [1] R. Duvigneau. Isogeometric analysis for compressible flows using a discontinuous galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 333:443–461, 2018.
- [2] L. Yan. Conversion from nurbs to bezier representation. *Computer Aided Geometric Design*, 113:102380, 2024.

- *Khalfoun Rayan*: CEA, DAM, DIF, F-91297 Arpajon  
E-mail: [Rayan.Khalfoun@cea.fr](mailto:Rayan.Khalfoun@cea.fr)
- *Stauffert Maxime*: CEA, DAM, DIF, F-91297 Arpajon  
E-mail: [Maxime.Stauffert@cea.fr](mailto:Maxime.Stauffert@cea.fr)