Implementation of an isogeometric finite element toolbox in Diffpack

Md Naim Hossain, Frank Vogel, Daniel Alves Paladim, Vinh Phu Nguyen, Prof. Stephane P.A. Bordas

inuTech GmbH, Nuremberg, Germany
Cardiff School of Engineering, Institute of Mechanics and Advanced Materials, Cardiff University, Wales, UK

11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
July 20 – 25, 2014, Barcelona, Spain
Outline

- Objective
- Diffpack
- Isogeometric Analysis
- Bézier Extraction of NURBS and T-Splines
- Object-oriented approach of IGAFEM module in Diffpack
- Numerical example
- Conclusion
- Future work
- Acknowledgement
Objective
Objective

- Our main motivation is to provide a generic implementation of isogeometric analysis (IGA) within a flexible C++ environment, namely the Diffpack platform.

- Flexibility of using existing Diffpack FEM classes with some modification for IGA based on NURBS and T-Spline using Bezier extraction operator.

- How object-oriented programming is useful for the treatment of data structures and operations associated with IGA.

- Implementation of T-Spline module using Rhino3D Autodesk T-Spline plugin.

- The Bézier extraction operator decomposes the NURBS or T-spline basis functions to be represented over $C^0$ continuous Bézier elements.

- Verifying and validating 2D and 3D elasticity numerical examples for different spline bases and comparison with FEM solutions.
Diffpack
Diffpack

- Diffpack is a object-oriented software environment with main emphasis on numerical solutions of partial differential equations.

- Diffpack is a collection of C++ libraries with classes, functions and other utility programs.

- Diffpack supports a variety of numerical methods with distinct focus of FEM but has no inherent restrictions of the type of PDEs.

- The numerical functionality is embedded in an environment of software engineering tools supporting the management of Diffpack development projects.
Diffpack® is a Development Environment

• PDEs

\[ K(S) = \lambda_o(S) + \lambda_w(S), \]
\[ f(S) = \frac{\lambda_w(S)}{K(S)}, \]
\[ h(S) = -\lambda_o(S)f(S)P_c(S), \]
\[ \lambda_w = k_w(...), \]
\[ \lambda_o = k_o(...). \]

\[- \nabla \cdot [K(S)\nabla P] = q, \]
\[ S_i + \nabla \cdot [v f(S)] = \nabla \cdot (h(S)\nabla S), \]
\[ v = -K(S)\nabla P \]

Object-Oriented (C++) Tools for the numerical Modeling and Solution of Differential Equations
Diffpack® Summary

- is a problem-solving environment for simulation problems
- are numerical libraries for PDE solution (> 600 C++ Classes)
- simplifies the solver development process significantly
- nicely complements standard FEM-programs

Learn more about it from http://www.diffpack.com
Isogeometric Analysis (IGA)
Isogeometric Analysis (IGA)

• A new simulation methodology closing the gap between Computer Aided Design (CAD) and Finite Element Analysis (FEA) by using the same shape functions (first introduced in 2005 by T.J.R Hughes).
  - Spline basis functions which is used to describe the geometry of the object also used to describe unknown solution field during the analysis.
  - Replace traditional Lagrange FEM functions by CAD B-Splines, NURBS, T-Splines, PHT-Splines etc.

• IGA includes standard FEA but it provides other possibilities:
  - Efficient and precise geometric modeling.
  - Simplified mesh refinement (h,p,k,r-refinement).
  - Smooth basis function with higher continuity compared to FEM basis function.
  - Superior approximation properties.
Nurbs-NonUniform Rational B-Splines

- NURBS (NonUniform Rational B-Spline) can exactly represent elementary curves and surfaces (circle, ellipse, cylinder, cone..) which can not be represented by polynomial splines (B-Splines).

- NURBS are rational functions of B-Splines and inherit all their favorable geometrical properties.

- In addition, there exist efficient algorithms for their evaluation and refinement.

- Rational basis function and NURBS curve:

\[ R_i^p (\zeta) = \frac{N_{i,p}(\zeta)w_i}{\sum_{i=1}^{n}N_{i,p}(\zeta)w_i} \]

\[ C(\zeta) = \sum_{i=1}^{n} R_i^p (\zeta)B_i. \]

- Rational surfaces and solids:

\[ R_{i,j}^{p,q} (\zeta, \eta) = \frac{N_{i,p}(\zeta)M_{j,q}(\eta)w_{i,j}}{\sum_{i=1}^{n}\sum_{j=1}^{m}N_{i,p}(\zeta)M_{j,q}(\eta)w_{i,j}} , \]

\[ R_{i,j,k}^{p,q,r} (\zeta, \eta, \zeta) = \frac{N_{i,p}(\zeta)M_{j,q}(\eta)L_{k,r}(\zeta)w_{i,j,k}}{\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l}N_{i,p}(\zeta)M_{j,q}(\eta)L_{k,r}(\zeta)w_{i,j,k}} , \]
**T-spline fundamentals:**

- No tensor product restriction as for NURBS.
- Incomplete rows and columns of control points.
- Local knot vectors define the T-spline basis function.
- The local knot vectors to $s_1$ are $\Xi = \{\xi_1, \xi_2, \xi_3, \xi_4, \xi_5, \xi_6\}$ and $H = \{\eta_1, \eta_2, \eta_3, \eta_4, \eta_5\}$

**Example: A simple T-mesh**

- Each knot lines represents a knot value.
- Incomplete knot lines terminates in T-junctions.
Bézier Extraction of NURBS and T-Splines
Bézier extraction

- B-splines, NURBS and T-splines can be written in terms of Bernstein polynomials and the Bézier extraction operator \( C \).
- \( C \) is generated by knot insertions until the multiplicity at each internal knot is equal to the polynomial order \( p \).

B-splines
- \( P^b = C^T P \)
- \( N(\xi) = CB(\xi) \)

NURBS
- \( P^b = (W^b)^{-1} C^T WP \)
- \( R(\xi) = WC \frac{B(\xi)}{W^b(\xi)} \)
Bézier extraction of T-splines

- **Bezier extraction operator for T-splines:**
  - Same idea as Bézier extraction of NURBS.
  - Map T-spline basis functions to Bernstein polynomials.

- **Differences compared to NURBS Extraction Operator:**
  - Local knot vectors vs. global knot Vector
  - Introduce the extended knot vector $\Xi = \{0,0,0,0,1,2,3,3,3,3,3\}$
  - Local tensor product domain vs. global tensor product domain one row to each basis function in support.
  - The element extraction operator for the knot span $[0,1)$ becomes

\[
\begin{bmatrix}
N_1 \\
N_2 \\
N_3 \\
N_4
\end{bmatrix} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & \frac{1}{2} & \frac{1}{4} \\
0 & 0 & \frac{1}{2} & \frac{7}{12} \\
0 & 0 & 0 & \frac{1}{6}
\end{bmatrix}
\begin{bmatrix}
B_1 \\
B_2 \\
B_3 \\
B_4
\end{bmatrix}
\]
Object oriented approach of IGAFEM module in Diffpack
• Isogeometric analysis based on Bézier extraction of NURBS:

  - Changes confined to shape function routine.
  - Extraction operators and Bézier basis functions are pre-calculated.
  - Output: NURBS basis functions and derivatives.
Implementation of IGAFEM in Diffpack-2

- Isogeometric analysis based on Bézier extraction of T-Splines:
  - Performed by importing a T-mesh of Bézier extraction from Rhino3D using autodesk T-Spline Plugin.
  - **Input**: Control points, Bézier Extraction operators and BC of mesh side and nodes.

**Advantage:**
- Locally refine the mesh at necessary regions.
- Easily read the Bézier mesh, geometry, control points and extraction operator which are not generated by the program.
Numerical Examples
Cylinder subjected to internal pressure (NURBS)

\[ E = 3 \times 10^7, a = 0.3, b = 0.5 \]
\[ P = 3000 \text{N/m}^2, \]
Plane stress

- Meshes at different subsequent refinement stage
- Displacement \( U_x \) and \( U_y \)
- Stresses \( \sigma_x \) and \( \sigma_{xy} \)
Cylinder subjected to internal pressure (NURBS vs. FEM)

Energy norm

L2-norm

NURBS perform better than Lagrange
Plate Circular Hole (NURBS)

- A plane stress problem.
- Only one quarter of plate analyzed.

Meshes at different subsequent refinement stage

Displacement $U_x$ and $U_y$

Stresses $\sigma_x$ and $\sigma_{xy}$
Plate Circular Hole (NURBS vs. FEM)

NURBS perform better than Lagrange
Innovative Numerical Technologies

3D Pinched Cylinder (NURBS)

Meshes at different subsequent refinement stage

Pinched Cylinder

- $E = 3.0 \cdot 10^4$
- $\nu = 0.3$
- $P = 1.0$
- $R = 300$
- $L = 600$
- $t = 3$

InuTech

Innovative Numerical Technologies
Cylinder subjected to internal pressure (T-Spline)

- T-mesh at different local refinement:

- Displacement and stress contour plot:
Timoshenko Beam (NURBS vs. T-spline)

- NURBS mesh
- T-Spline mesh

Nurbs IGA displacement and stress
T-Spline IGA displacement and stress

Energy norm
T-Spline perform better than NURBS

Innovative Numerical Technologies
Conclusions
Conclusions

- Bézier extraction is significantly easing implementation of isogeometric analysis in an existing FEM Diffpack Kernel framework.

- NURBS avoid geometric error in discretization of the problem.

- NURBS based IGA has higher accuracy than Lagrange based FEM analysis.

- NURBS elements has the same convergency rate as Lagrange elements but with far fewer DOFs.

- A FE code capable of handling extraction operators can easily incorporate for both NURBS and T-splines.

- It allows an analyst to do T-splines FEA without understanding the details of T-splines using Bézier extraction of Rhino3D Autodesk T-spline plugin.
Future work
Future work

- Extension of the toolbox for 3D T-Spline IGA problem.
- PHT spline based IGAFEM within Diffpack platform.
Acknowledgement
Acknowledgement

- A very special thanks to Prof. Stéphane P.A. Bordas and Vinh Phu Nguyen from Cardiff University and Frank Vogel from inuTech GmbH for their valuable and insightful guidance.

- Financial support of the Framework Programme 7 Initial Training Network Funding under grant number 289361 "Integrating Numerical Simulation and Geometric Design Technology (INSIST)".