

ISOGEOMETRIC ANALYSIS FOR 2D MAGNETOSTATIC COMPUTATIONS WITH MULTI-LEVEL BÉZIER EXTRACTION FOR LOCAL REFINEMENT

Andreas Grendas¹, Michael Wiesheu², Sebastian Schöps² and Benjamin Marussig¹

¹ Graz University of Technology
Institute of Applied Mechanics
Technikerstraße 4/II, 8010 Graz, Austria
a.grendas@tugraz.at, marussig@tugraz.at
www.tugraz.at/institute/am-bm

² Computational Electromagnetics Group
Technical University of Darmstadt
Schloßgartenstr. 8, 64289 Darmstadt, Germany
michael.wiesheu@tu-darmstadt.de, sebastian.schoeps@tu-darmstadt.de
www.cem.tu-darmstadt.de/cem/start/index.en.jsp

Key words: Isogeometric analysis, Local refinement, Multi-level Bézier extraction, Magneto-static

Summary. Local refinement is vital for efficient numerical simulations. In the context of Isogeometric Analysis (IGA), hierarchical B-splines have gained prominence. The work applies the methodology of truncated hierarchical B-splines (THB-splines) as they keep additional properties. The framework is further enriched with Bézier extraction, resulting in the multi-level Bézier extraction method. We apply this discretization method to 2D magnetostatic problems. The implementation is based on an open-source Octave/MATLAB IGA code called GeoPDEs, which allows us to compare our routines with globally refined spline models as well as locally refined ones where the solver does not rely on Bézier extraction.

1 Introduction

In IGA [1], the Basis splines (B-spline) and Non-Uniform Rational B-splines (NURBS) functions used to represent the CAD geometry are also employed to approximate the field variables within the numerical model. This approach reduces the geometric discretization errors inherent in finite element models that use an approximated computational geometry.

A key topic is the generalization of spline constructions, enabling local refinement by breaking the global tensor product structure of multivariate splines. Several methods have been devised to address this limitation of tensor product structures, such as T-splines [2], locally refined B-spline (LR-splines) [3], and hierarchical B-splines (HB-splines) [4]. HB-splines operate on a multi-level structure with a sequence of different levels of detail and perform local refinement by activating and deactivating of B-splines on these levels. The HB-splines can be extended with a truncation mechanism, which leads to truncated HB-splines (THB-splines) [5]. Additional information about the THB-splines will be discussed in the corresponding section.

This work introduces an adaptive IGA framework based on THB-splines, which is extended with the discretization technique of classical Bézier extraction [6]. This approach could potentially implement the hierarchical B-spline shape functions into an existing finite element (FE) solver by utilizing Bézier element with a standard single-level basis (Bernstein polynomials), in other words, multi-level Bézier extraction [7, 8].

Our implementation relies on GeoPDEs, an open-source Octave/MATLAB IGA code [9], where we can provide proper comparisons as the tool already offers adaptivity based on THB-splines. For the numerical example, we solve a scalar 2D magnetostatic problem. In addition, a posteriori error estimator is introduced, which requires further investigation for more complex geometries, but it can promise refinement without an exact solution.

2 Background

This section summarizes the relevant concepts to offer a clear overview of how the adaptivity method is employed in IGA by utilizing multi-level Bézier extraction. Starting with B-splines, followed by the multi-level structure based on THB-splines, and finally applying the Bézier extraction technique.

2.1 B-splines

In IGA, B-splines serve as the basis functions for representing geometry and performing the simulation analysis [1, 10]. B-spline basis functions are defined by degree p , and a knot vector,

$$\Xi = \{\xi_1 \leq \dots \leq \xi_{n+p+1}\}, \quad (1)$$

where n is the number of the basis functions. The Cox-de Boor recursion formula defines the n functions, by starting from degree $p = 0$, up to the prescribed degree,

$$N_{i,0} = \begin{cases} 1 & \text{for } \xi_i \leq \xi < \xi_{i+1} \\ 0 & \text{else} \end{cases} \quad (2)$$

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

A B-spline curve can be constructed as the linear combination of the basis functions,

$$C(\xi) = \sum_{i=1}^n N_{i,p}(\xi) \mathbf{P}_i, \quad (3)$$

where $\mathbf{P}_i \in \mathbb{R}^d$ are the control points associated to the basis functions.

In general, B-spline functions are piece-wise polynomials, they are non-negative over their domain, they constitute a partition of unity, they have local support, and finally, their continuity depends on knot multiplicity k , C^{p-k} . In Fig. 1, we have a quadratic B-spline that represent all the aforementioned properties.

For higher dimensions, we need to construct multivariate B-splines as a tensor product of univariate B-splines, as indicated in Fig. 2. Considering \mathbb{R}^2 , the degrees p and q and knot vectors

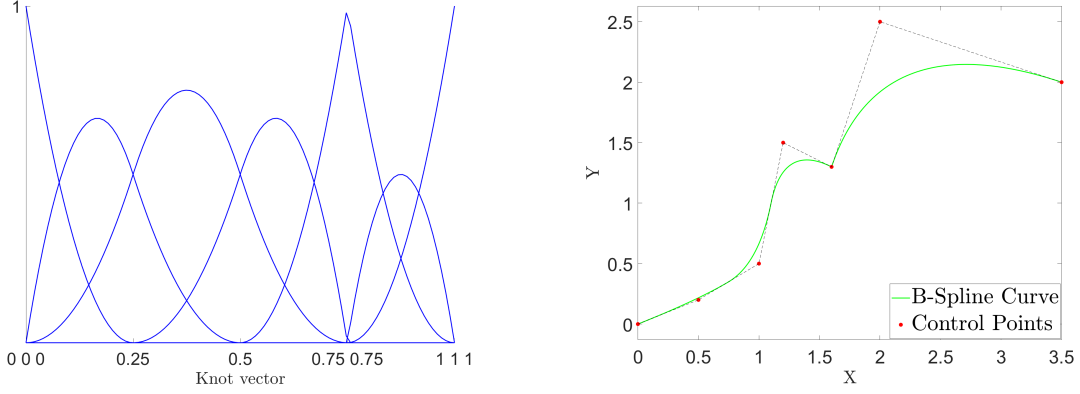


Figure 1: B-spline function with knot vector, $\Xi = [0, 0, 0, 0.25, 0.5, 0.75, 0.75, 1, 1, 1]$ and degree $p = 2$. Left: basis functions. Right: curve with control points.

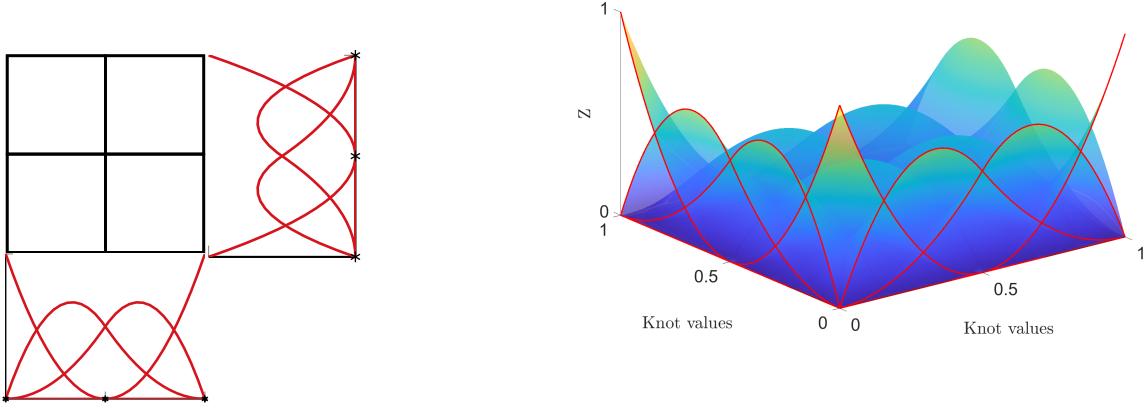


Figure 2: Reference domain with the basis function in 2D, as tensor product structure.

$\Xi = \{\xi_1 \leq \dots \leq \xi_{n+p+1}\}$ and $H = \{\eta_1 \leq \dots \leq \eta_{m+q+1}\}$, define each dimension of a surface. Thus, two sets of basis functions along with the control net $\mathbf{P}_{i,j}$ with $i = 1, 2, \dots, n$, $j = 1, 2, \dots, m$, describe the surface,

$$S(\xi, \eta) = \sum_{i=1}^n \sum_{j=1}^m N_{i,p}(\xi) M_{j,q}(\eta) \mathbf{P}_{i,j}. \quad (4)$$

2.2 Multi-level structure with THB-splines

A multi-level structure with hierarchical B-splines offers a state-of-the-art method for adaptive refinement, especially in representing complex geometries. Knot insertion algorithms construct the individual levels, ℓ . We follow the dyadic refinement technique with $2d \cdot n_{\ell}$ elements for each new level of refinement (d is the dimension and n_{ℓ} number of elements in $\ell - 1$), so

the levels preserve the same polynomial degree. This multi-level structure consists of sequence of nested spline spaces of univariate splines, $\mathbb{S}^{(0)}(\Omega) \subset \mathbb{S}^{(1)}(\Omega) \cdots \subset \mathbb{S}^{(\ell_{max})}(\Omega)$. In Fig. 3, a univariate quadratic B-spline with two additional levels of refinement is depicted. The geometry and the parametric representation do not change under the refinement process, but the relation between the levels occurs by an operator called subdivision matrix \mathbf{S} . The parent functions, $N(\xi)_{i,p}^{(\ell)}$ of the level ℓ are a linear combination of fine-scale functions $\hat{N}_{j,p}(\xi)^{(\ell+1)}$ (children functions) of level $\ell + 1$,

$$N(\xi)_{i,p}^{(\ell)} = \sum_{j=1}^{\hat{n}} \mathbf{S}_{j,i} \hat{N}_{j,p}(\xi)^{(\ell+1)}. \quad (5)$$

THB-splines [5] enhance traditional hierarchical B-splines by restoring the partitioned unity property and thereby reducing their overlap. As a result, THB-splines provide a more flexible and powerful tool for local refinement. The black basis functions in Fig. 3, are the overlapping ones, where we apply the truncation mechanism. The truncated basis functions are defined as,

$$truncN(\xi)_{i,p}^{(\ell)} = \sum_{j \in \eta_D^{\ell+1}} S_{ji}^{(\ell)} N(\xi)_{j,p}^{(\ell+1)}. \quad (6)$$

As previously stated, in a hierarchical structure, each basis function at level ℓ is a linear combination of basis functions at level $\ell + 1$, encoded in the subdivision matrix (5). At that point, the global multi-level extraction operator, \mathbf{M}_{ℓ}^{glob} , can be defined by applying a sequence of nested spline spaces, where the goal is to extract the hierarchical functions supporting each element at the specific level. For levels $\ell = 0, \dots, \ell_{max}$ the $\mathbf{M}_{\ell_{max}}^{glob}$ can be obtained by joining the rows, corresponding to the active basis, of the refinement operators $\mathbf{S}^{(\ell, \ell_{max})}$ [7]. For example, the multi-level operator for level 2, $\mathbf{M}^{(\ell_2)}$, is derived by the operators $\mathbf{S}^{(\ell_0, \ell_1)}$, $\mathbf{S}^{(\ell_1, \ell_2)}$, $\mathbf{S}^{(\ell_2, \ell_2)}$ and $\mathbf{M}^{(\ell_2)} \in \mathbb{R}^{n \times m}$, where n is the active basis functions of ℓ_0 plus the the active basis functions of ℓ_1 and ℓ_2 and m is the total number of basis functions of ℓ_2 .

2.3 Standard Bézier extraction

Similar to standard finite elements, there is a desire for an element-by-element assembly routine to increase the computational efficiency of the assembly procedure. One way to achieve this is Bézier extraction, which maps a piecewise Bernstein polynomial basis into a B-spline basis and vice versa [6]. This mapping allows using C^0 Bézier elements as the finite element representation of smooth splines.

We use Bézier decomposition, which is accomplished by repeating all interior knots of a knot vector until they have a multiplicity equal to the degree of the polynomial [6]. A representation of Bézier extraction for a univariate B-spline function is depicted in Fig. 4. An important feature is that the initial curve geometry remains unchanged by changing only the basis and the corresponding control points. The B-spline functions, \mathbf{N} , can be expressed as a linear combination of Bernstein polynomials \mathbf{B} with the Bézier extraction operator \mathbf{E} .

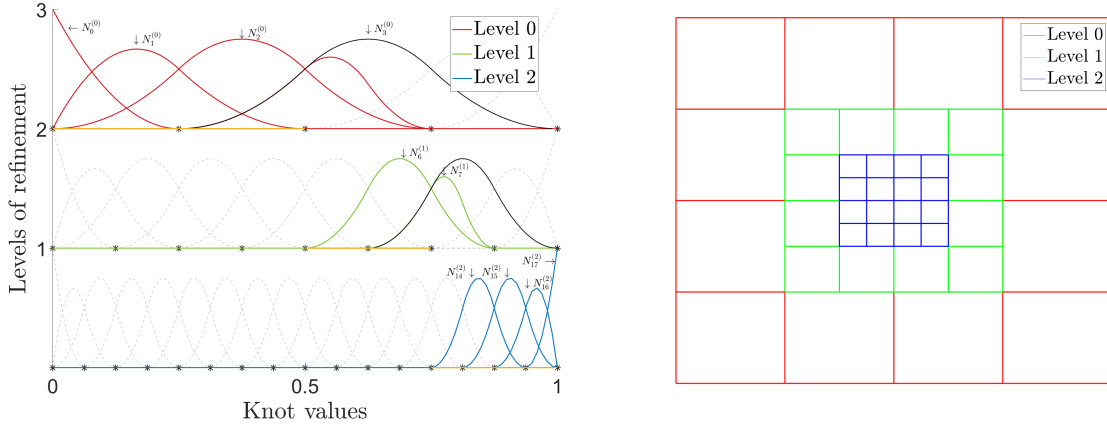


Figure 3: Left: a univariate quadratic THB-spline with two additional levels of refinement. Right: a 2D representation of a tensor product quadratic THB-splines.

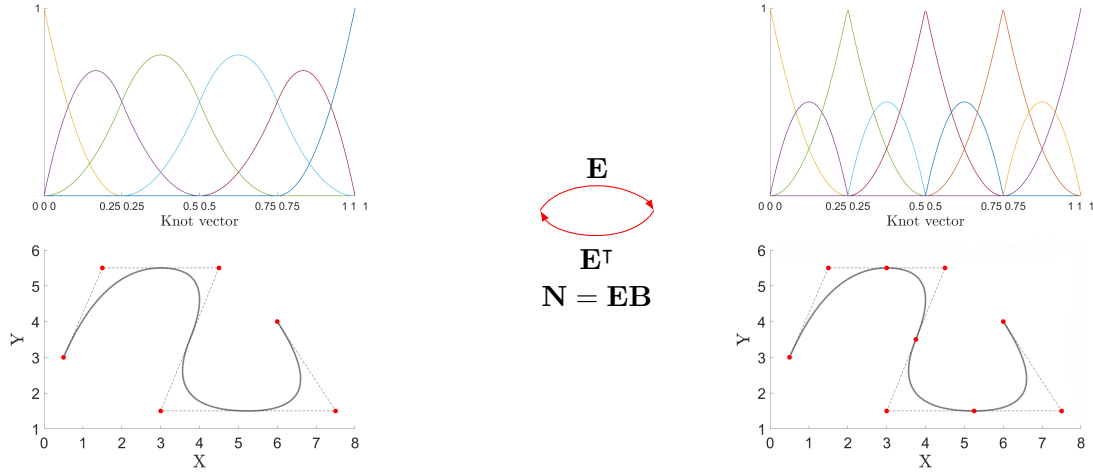


Figure 4: Left: B-spline basis and curve. Right: Bézier basis and curve.

3 Multi-level Bézier extraction

The combination of those two techniques of multi-level THB-splines and Bézier extraction results in multi-level Bézier extraction, which decompose B-spline basis functions into a series of simpler Bézier elements across multiple hierarchical levels.

To construct the multi-level Bézier extraction operator, \mathbf{C} , for the spline basis, we localize the global multi-level extraction operator \mathbf{M}_e^{loc} and the Bézier extraction operator \mathbf{E}_e^{loc} , for an element-by-element approach. This creates a direct map \mathbf{C}^e from a standard set of reference basis functions equal for each element (Bernstein basis, \mathbf{B}) to the multi-level local basis (hierarchical basis, \mathbf{H}^e),

$$\mathbf{H}^e = \mathbf{M}_e^{loc} \mathbf{E}_e^{loc} \mathbf{B} = \mathbf{C}^e \mathbf{B}. \quad (7)$$

In Fig. 5, there is a uni-variate B-spline element on the left side with basis functions from different levels of the refinement. This element can be transformed directly to a Bézier element

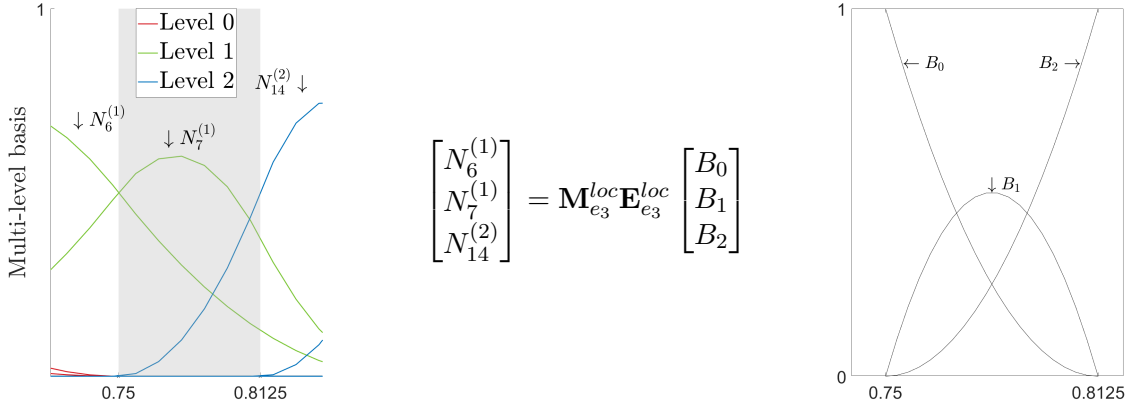


Figure 5: Translation from hierarchical spline space to local Bernstein polynomials.

with Bernstein basis functions only through the application of the operator $\mathbf{C}^{e_3} = \mathbf{M}_{e_3}^{loc} \mathbf{E}_{e_3}^{loc}$.

The same procedure can be extended into multipatch domains, where the computation domain Ω is divided into n_i non-overlapping subdomains Ω_i or patches. Therefore more complex geometries can be handled with greater detail. The geometrical mapping from the reference domain, Φ_i , corresponds to each patch to construct the physical geometry. Section 4 presents a numerical example of a multipatch domain for a scalar magnetostatic 2D problem.

3.1 Problem assembly for numerical simulation

The multi-level Bézier extraction operator can be used to assemble the system of matrices and vectors of the discretized problem. For instance, the calculation of the stiffness matrix of B-spline elements usually requires special types of algorithms, able to handle elements with a non-standard set of basis and hierarchical structure. In this work, GeoPDEs [9] is employed due to its implementation of local refinement techniques in IGA with THB-splines.

On the other hand, if Bézier elements are considered, allow for an element-wise assembly as in standard finite element solvers, we can compute a stiffness matrix of a classic Bézier element and then transform it to a B-spline element stiffness matrix. For this, a local element-by-element approach needs to be taken into account, and then with the proper transformation matrix the global stiffness matrix of the THB-spline basis is computed,

$$\begin{aligned} \mathbf{K}_{B-spline}^e &= \mathbf{C}^e \mathbf{K}_{Bezier}(\mathbf{C}^e)^T \\ \mathbf{K}_{global} &= \sum_{e=1}^{n_{el}} \mathcal{T}(\mathbf{K}_{B-spline}^e). \end{aligned} \quad (8)$$

The \mathbf{K}_{Bezier} is the stiffness matrix of a Bézier element on the corresponding level of refinement. Together with the local multi-level Bézier extraction operators, \mathbf{C}^e , the stiffness matrix of the THB-spline space element is calculated, $\mathbf{K}_{B-spline}^e$.

3.2 Posteriori error estimator

Let us now look at a posteriori error estimator based on the least-square formulation [11]. Here the refinement follows consecutive refinement levels to consider the areas of interest locally.

In brief, first, the initial discretized system needs to be solved,

$$\begin{bmatrix} A_h & B_h \\ B_h^T & 0 \end{bmatrix} \begin{bmatrix} p \\ u \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}. \quad (9)$$

Here, $A_h \in \mathbb{R}^{M_Y \times M_Y}$, is the stiffness matrix of the fine mesh, $B_h \in \mathbb{R}^{M_Y \times M_X}$, is the stiffness matrix of the coarse mesh, mapped onto the fine mesh and f is the source term. The solution vector consists of the error approximation, p , and the numerical solution u . With the error approximation, we evaluate the local error indicators into the coarse mesh,

$$\eta_k^2 := \int |\nabla p_h(x)|^2 dx. \quad (10)$$

Finally, a marking strategy determines the elements that need refinement. In this work, the Dörfler criterion is applied with a minimal set $M \subset N$ where N , is all elements in the coarse mesh and M , is the subset of elements marked for refinement based on the error indicators,

$$\sum_{k \in M} \eta_k^2 \geq \theta \sum_{k \in N} \eta_k^2, \quad (11)$$

where θ is a parameter in the range $0 < \theta < 1$.

As an example of this error estimator, a Poisson problem with non-homogeneous Dirichlet boundary conditions in a unit square domain with a peak value in the middle was computed. The results in Fig. 6, measure the true error between the numerical and the exact solution (blue line) and the behavior of the estimator (red line). The two lines converge similar and that indicates that the error estimator captures the behavior of the true error in the numerical solution. This gives the potential to extend the use of the error estimator to more complex problems with more detailed geometries.

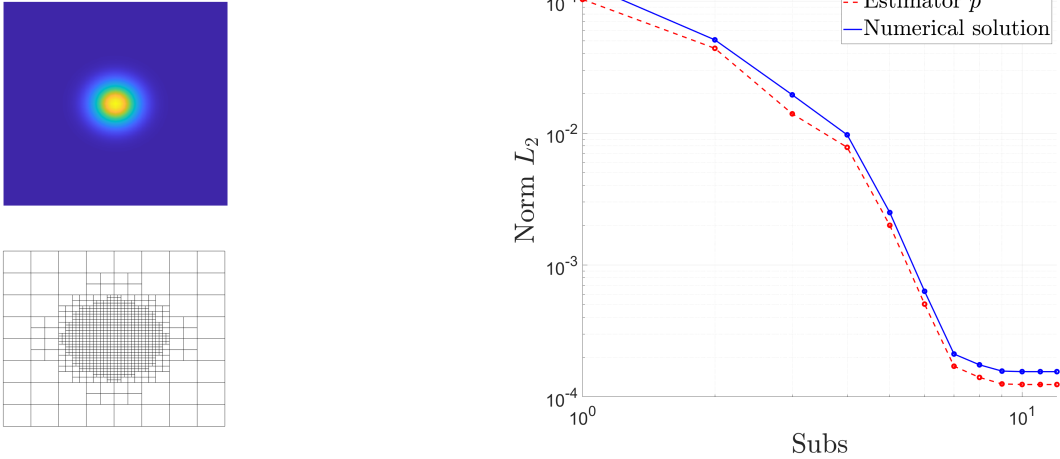


Figure 6: Estimator results for the Poisson problem.

4 Numerical example

GeoPDEs [9], an open-source FE solver for IGA, implemented in Matlab/Octave, is applied for the numerical example of this contribution. We have a 2D scalar magnetostatic problem with a horseshoe magnet and a metal sheet. This is a multipatch geometry with 30 patches and 3 different materials, air, iron, and neodymium magnet, as shown in Fig. 7.

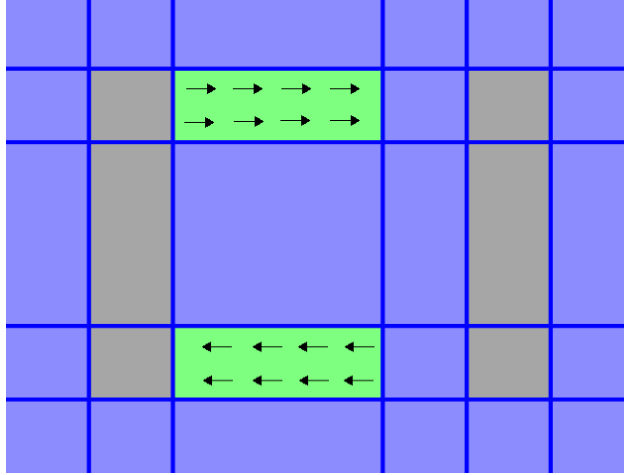


Figure 7: The thick lines indicate the boundaries of the patches. The variation in colors represents the different materials. The blue is the air, the grey is the iron and the light green is the neodymium magnet. The arrows indicate the direction of the magnetic field.

To solve the magnetostatic problem in magnetic vector potential form, the following derivation is applied. We start with the Ampère–Maxwell equation, given in terms of the magnetic field strength \mathbf{H} , where the displacement currents are disregarded, and the Gauss’s law for magnetism, \mathbf{B} is the magnetic flux density,

$$\begin{aligned}\nabla \times \mathbf{H} &= \mathbf{J}_f \\ \nabla \cdot \mathbf{B} &= 0.\end{aligned}\tag{12}$$

Then the definition of the magnetic vector potential and the material law (constitutive equation) are introduced.

$$\begin{aligned}\mathbf{B} &= \nabla \times \mathbf{A}, \implies \nabla \cdot \mathbf{B} = \nabla \cdot (\nabla \times \mathbf{A}), \\ \mathbf{B} &= \mu \mathbf{H} + \mathbf{B}_r \implies \mathbf{H} = \mu^{-1} (\mathbf{B} - \mathbf{B}_r), \text{ with } \mu = \mu_0 \mu_r,\end{aligned}\tag{13}$$

where μ_0 is the vacuum permeability and μ_r is the relative permeability. The 3D equations reduce to a 2D form by assuming that all variations along the z -axis are zero and the fields vary only over the xy -plane. By expressing the magnetic flux density, \mathbf{B} , in terms of the vector potential, \mathbf{A} , and substituting it into the constitutive relations and Ampère–Maxwell Law, we derive a scalar differential equation.

$$\nabla \times (\mu^{-1} (\nabla \times \mathbf{A} - \mathbf{B}_r)) = \mathbf{J},\tag{14}$$

and in the 2D case,

$$\frac{\partial}{\partial x} \left(\mu^{-1} \left(-\frac{\partial A_z}{\partial x} - B_{ry} \right) \right) - \frac{\partial}{\partial y} \left(\mu^{-1} \left(\frac{\partial A_z}{\partial y} - B_{rx} \right) \right) = J_z. \quad (15)$$

Note that in 3D edge elements (15) must be used to discretize (14) [12], in 2D nodal elements are sufficient.

To demonstrate the correctness of the Bézier approach, we do not apply the error estimator of Section 3.2 but use a numerical reference solution. The reference solution is obtained using a uniformly refined mesh with a high density of elements and many degrees of freedom to the specific problem (No. elements = 108000, DoFs = 112302). With that reference solution u , the true error approximation with a prescribed tolerance is applied. The quality of the approximate solution u_h is assessed by the L_2 -error,

$$\epsilon_{L_2} = \left(\int_{\Omega} |u - u_h|^2 dx \right)^{1/2}. \quad (16)$$

An element is marked for the refinement if,

$$\epsilon_{L_2,n} > tolerance = 10^{-8}. \quad (17)$$

The final locally refined mesh is in Fig. 8.

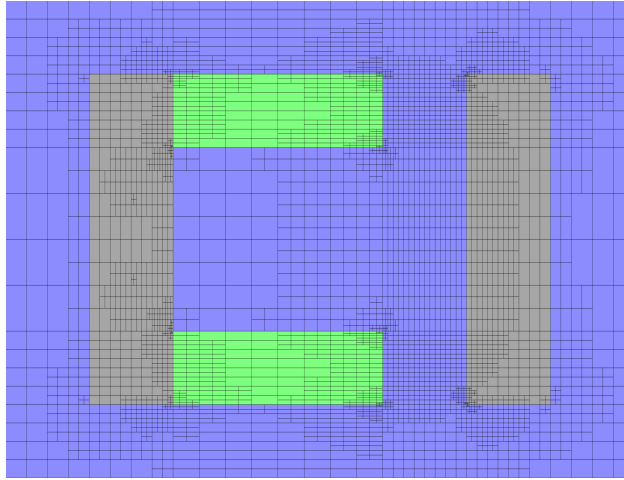


Figure 8: Local mesh refinement.

The boundary conditions are homogeneous Dirichlet conditions, i. e. , a flux wall, to ensure that no magnetic field exists outside our domain, see Fig. 9 left. Then, the magnetic flux density can be derived by the definition of the magnetic vector potential, as shown in Fig. 9 right.

Finally, the efficiency of the multi-level Bézier extraction for local refinement can be described by the relative L_2 -error approximation between uniform and local refinement, as shown in Fig. 10. As expected, the local refinement converges faster with fewer degrees of freedom, but at the same time, this verifies the effectiveness of multi-level Bézier extraction.

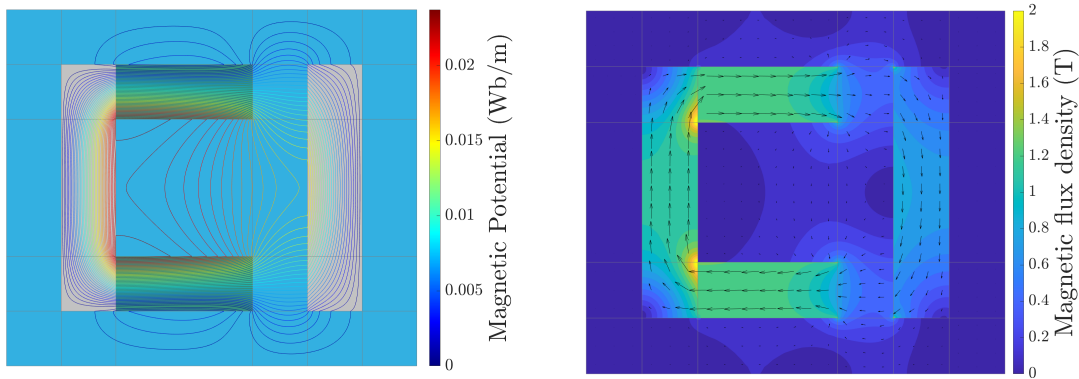


Figure 9: Left: magnetic potential lines. Right: magnetic field density.

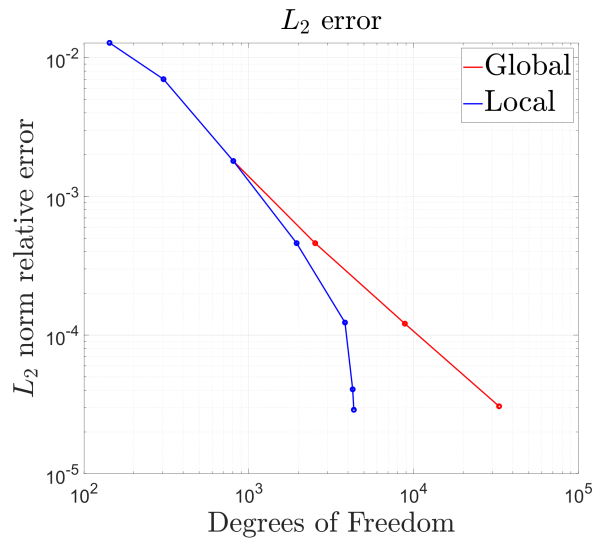


Figure 10: L_2 norm relative error, local and global refinement.

5 Conclusions

We discussed the extension of the Bézier extraction technique to truncated hierarchical B-splines (THB-splines). We demonstrated the behavior of this approach in the context of 2D magnetostatic simulation. The multi-level Bézier extraction can be compatible with other hierarchical spline spaces and is not restricted to THB-splines. The example of multipatch domains demonstrates that multi-level Bézier extraction applies to complex geometries, where the calculation of Bézier elements suffices to simulate over hierarchical spaces. Since the assembly is performed element-by-element with a subsequent multiplication of the extraction operators, integrating THB-splines into other Bézier element-based isogeometric analysis (IGA) codes is straightforward. Additionally, the use of standard FE codes is simplified significantly. The approximation error drives the refinement process. The error was computed using an error estimator based on consecutive levels of refinement in one example, while the other utilized a reference solution.

6 Acknowledgments

The work is supported by the joint DFG/FWF Collaborative Research Centre CREATOR (DFG: Project-ID 492661287/TRR 361; FWF: 10.55776/F90) at TU Darmstadt, TU Graz and JKU Linz.

REFERENCES

- [1] Hughes, T.J., Cottrell, J.A. and Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer methods in applied mechanics and engineering*, 194(39-41), pp.4135-4195.
- [2] Scott, M.A., Borden, M.J., Verhoosel, C.V., Sederberg, T.W. and Hughes, T.J., 2011. Isogeometric finite element data structures based on Bézier extraction of T-splines. *International Journal for Numerical Methods in Engineering*, 88(2), pp.126-156.
- [3] Dokken, T., Lyche, T. and Pettersen, K.F., 2013. Polynomial splines over locally refined box-partitions. *Computer Aided Geometric Design*, 30(3), pp.331-356.
- [4] Kraft, R., 1997. Adaptive and linearly independent multilevel B-splines. SFB 404, Geschäftsstelle.
- [5] Giannelli, C., Jüttler, B. and Speleers, H., 2012. THB-splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*, 29(7), pp.485-498.
- [6] Borden, M.J., Scott, M.A., Evans, J.A. and Hughes, T.J., 2011. Isogeometric finite element data structures based on Bézier extraction of NURBS. *International Journal for Numerical Methods in Engineering*, 87(1-5), pp.15-47.
- [7] D'Angella, D., Kollmannsberger, S., Rank, E. and Reali, A., 2018. Multi-level Bézier extraction for hierarchical local refinement of Isogeometric Analysis. *Computer Methods in Applied Mechanics and Engineering*, 328, pp.147-174.
- [8] Grendas, A. and Marussig, B., 2023. Multi-level Bézier extraction of truncated hierarchical B-splines for isogeometric analysis. *PAMM*, 23(4), p.e202300092.
- [9] Garau EM, Vázquez R. Algorithms for the implementation of adaptive isogeometric methods using hierarchical B-splines. *Applied Numerical Mathematics*. 2018 Jan 1;123:58-87.
- [10] Piegl, L. and Tiller, W., 2012. *The NURBS book*. Springer Science & Business Media.
- [11] Köthe, C., Löscher, R. and Steinbach, O., 2023. Adaptive least-squares space-time finite element methods. arXiv preprint arXiv:2309.14300.
- [12] Monk, P., 2003. *Finite element methods for Maxwell's equations*. Oxford university press.