VOLUMETRIC SPLINE GEOMETRIES FOR FINITE ELEMENTS IN SIMULATION AND OPTIMIZATION OF ELECTRIC MACHINES

L. Blumrich[*], Y. Burkhardt[*], C. Geuzaine[†], B. Marussig[‡], S. Schöps[§], M. Wiesheu [§]

[*]Electrical Drive Systems, Technical University of Darmstadt, 64289 Darmstadt, Germany
[†]Montefiore Institute, University of Liège, B-4000 Liège, Belgium
[‡]Institute of Applied Mechanics, Technical University of Graz, 8010 Graz, Austria
[§]Computational Electromagnetics Group, Technical University of Darmstadt, 64289 Darmstadt, Germany
leon.blumrich@tu-darmstadt.de

Keywords: Finite element analysis, splines, geometry.

Abstract

During design optimization, geometry modifications pose challenges for mesh generation. We propose a method that combines spline-based representations with an automated structured mesh generation. The approach provides a reduction in the computational complexity compared to conventional meshing techniques, while allowing for seamless integration of geometric changes into the mesh. The concept is implemented in Gmsh and GetDP.

1 Introduction

In the design process of electrical machines, their geometry is frequently modified, see for example [1]. The geometry is represented by Computer-aided design (CAD) kernels which rely on boundary representations (BREP) based on Non-Uniform Rational B-Splines (NURBS) [2]. However, simulators based on finite elements, need volumetric or surface meshes in 3d or 2d, respectively. Therefore, shape modifications come with several challenges. On the one hand, common freeform optimization that moves mesh nodes does not result in proper CAD descriptions. On the other hand, BREP-driven freeform optimization keeps the CAD representation but may require remeshing or mesh morphing algorithms. To this end, Hughes et al. [3] proposed isogeometric analysis (IGA). It was later adapted for simulation and optimization of electric machines, e.g. in [4]. The idea is to use the same spline basis functions for geometry and solution representation. However, this requires dedicated simulators.

Here, we suggest a compromise: the geometry is represented by volumetric (bivariate in 2d or trivariate in 3d) splines and they are used to automatically generate (polynomially curved) structured meshes by pushing nodes and their interconnections through the spline geometry mapping. We demonstrate this using Gmsh and GetDP [5].

2 Volumetric spline descriptions

As discussed above, CAD models are typically represented by NURBS, in particular, since they are well suited

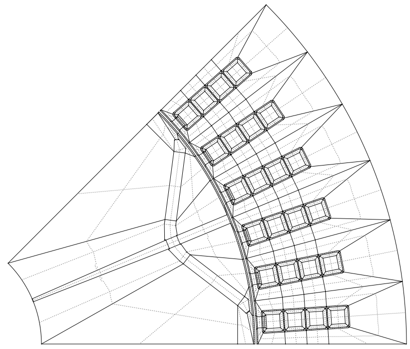

Fig. 1: Bivariate spline model of a machine geometry.

to precisely describe conic sections, offering local smoothness control and providing an intuitive framework for defining curves and surfaces [2]. Based on the standard spline theory described in [6], one constructs the basis of B-Splines from an open knot vector $\Xi = [\xi_1, \dots, \xi_{N_\xi}] \in [0,1]^{N_\xi}$. We denote by $B_i^p \in S^p(\Xi)$ the i-th B-Spline basis function of degree p. The NURBS basis is given as

$$N_i^p(u) = \frac{\omega_i B_i^p(u)}{\sum_j \omega_j B_j^p(u)} \tag{1}$$

with an additional weighting factor ω_i. In CAD, boundary representations are typically used, i.e., a two-dimensional machine model is represented by curves from the reference to the physical domain

$$\mathbf{F}_n(u) = \sum_i N_i^p(u)\mathbf{P}_{n,i} \tag{2}$$

where $\mathbf{P}_{n,i} \in \mathbb{R}^2$ are the control points of curve n. However, we discuss here the case of bivariate spline representations. We use the multi-patch decomposition $\mathbf{S} = \bigcup_{n=1}^{N_\mathrm{s}} \mathbf{S}_n$. The $\mathbf{S}_n$ are called patches and map from the unit square to the physical space

$$R_{i,j}^{p,q}(u,v) = \frac{B_i^p(u)B_j^q(v)w_{i,j}}{\sum_{k=0}^{n}\sum_{l=0}^{m} B_k^p(u)B_l^q(v)w_{k,l}} \tag{3}$$

with degrees p, q and weights $w_{n,i,j}$ such that

$$\mathbf{S}_n(u,v) = \sum_i \sum_j R_{i,j}^{p,q}(u,v)\mathbf{P}_{n,i,j} \tag{4}$$

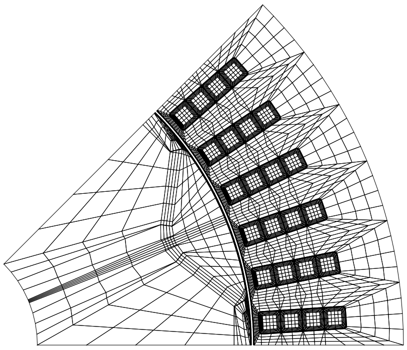

Fig. 2: Straight-sided mesh pushed through mapping.

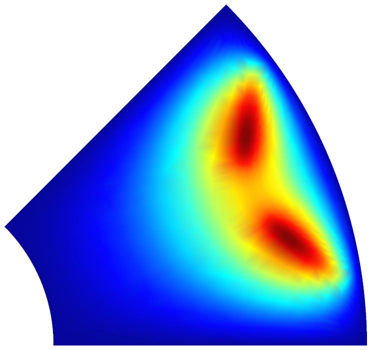

Fig. 3: GetDP result for Laplace problem on rotor domain.

is a surface given in terms of the control points $\mathbf{P}_{n,i,j} \in \mathbb{R}^2$. Fig. 1 shows an exemplary machine model based on bivariate spline patches.

3 Mesh creation and simulation

Let us discuss the non-curved case in two dimensions first: we define a set of equidistant points in the reference domain, that is, $u_n = l/N_\mathsf{u}$ with $l = 0, \ldots, N_\mathsf{u}$. This allows us to define their Cartesian product $[u_1, \cdots, u_{N_\mathsf{u}}] \times [u_1, \cdots, u_{N_\mathsf{u}}]$. These nodes, along with their interconnecting edges (e.g., from (u_i, u_j) to (u_i, u_{j+1})), form a mesh in the reference domain. The points can be easily pushed through the mapping to obtain the points

$$(x_{n,i}, y_{n,j}) = \mathbf{S}_n(u_i, u_j) \text{ for } 0 \leq i, j \leq N \tag{5}$$

in the physical domain. Now, using the same node connectivity as before, we create a straight-sided mesh in physical space, which contains, for example, the edge from $(x_{n,i}, y_{n,j})$ to $(x_{n,i}, y_{n,j+1})$. The only remaining task is to identify the interface nodes of neighboring patches. The mesh created from the geometry of Fig. 1 is shown in Fig. 2. Similarly, three-dimensional meshes and polynomially curved elements can be created. The latter are obtained by evaluating the derivatives at the points (u_i, u_j) or additional points. The second approach is the natural way to export in Gmsh, see [7, 10.2.2].

Fig. 3 shows the result of a Laplace problem computed by GetDP on the mesh from Fig. 2. The full paper will demonstrate the benefits in the context of shape optimization.

4 Discussion

The complexity of this procedure is very low, i.e., linear in the number of nodes $N_\mathsf{s} N_\mathsf{u}^2$. In addition, shape modifications of the spline geometry are automatically reflected in the mesh. On the other hand, the mesh is structured and depends on the geometry parametrization, i.e., even though the mesh is equidistant in the reference domain, the physical mesh can have arbitrarily high aspect ratios depending on (the derivatives of) the spline mappings. Although mappings can be optimized, e.g., based on the Winslow functional [8], spline geometries must be created with care, possibly by hand. However, this shall pay off, for example, in large-scale optimization runs.

Acknowledgment

This work is supported by the Collaborative Research Center CREATOR (DFG: 492661287/TRR 361; FWF: 10.55776/F90). Further support is provided by the Graduate School CE at TU Darmstadt.

References

[1] P. Di Barba, *Multiobjective Shape Design in Electricity and Magnetism*, ser. Lecture Notes in Electrical Engineering. Springer, 2010.

[2] E. Cohen, R. F. Riesenfeld, and G. Elber, *Geometric Modeling with Splines: An Introduction*. CRC Press, 2001.

[3] J. A. Cottrell, T. J. R. Hughes, and Y. Bazilevs, *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley, 2009.

[4] M. Merkel, P. Gangl, and S. Schöps, “Shape optimization of rotating electric machines using isogeometric analysis,” *IEEE Trans. Energ. Convers.*, vol. 36, no. 4, 02 2021, arxiv:1908.06009.

[5] C. Geuzaine and J.-F. Remacle, “Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities,” *Int. J. Numer. Meth. Eng.*, vol. 79, pp. 1309–1331, 2009.

[6] L. Beirão da Veiga, A. Buffa, G. Sangalli, and R. Vázquez, “Mathematical analysis of variational isogeometric methods,” *Acta. Num.*, vol. 23, pp. 157–287, 05 2014.

[7] C. Geuzaine and J.-F. Remacle, *Gmsh Reference Manual 4.14.0*.

[8] J. Gravesen, A. Evgrafov, D.-M. Nguyen, and P. Nørtoft, “Planar parametrization in isogeometric analysis,” in *Mathematical Methods for Curves and Surfaces*. Springer, vol. 8177, pp. 189–212.