Isogeometric Analysis with Trimmed CAD Models

Benjamin Marussig

1 Graz Center of Computational Engineering (GCCE), Graz University of Technology, Krenngasse 37/I, 8010 Graz, Austria

Trimmed CAD models are the most popular geometry representation in engineering design, but their direct application to an isogeometric analysis can lead to ill-conditioned system matrices. An approach is presented that effectively resolves this issue by combining so-called extended B-splines with a local refinement procedure.

1 Introduction

Trimming is an omnipresent procedure in current computer aided design (CAD) systems. It provides the basis for Boolean operations which are the most important CAD functions in creating objects [1]. During these operations, intersections of tensor product surfaces are computed and trimming enables the visualization of the resulting objects. This is done by defining trimming curves which represent the intersection in each parameter space of the surfaces involved. These curves allow the partitioning of a surface into a visible area \( A^v \) and its “hidden” complement, where only the former is considered for graphical display.

On the one hand, the trimming concept is a very convenient and effective way to define arbitrary shapes, but on the other hand it causes several problems: First of all, the actual definitions of intersecting surfaces are never adjusted to reflect the object obtained by a Boolean operation. In essence trimmed CAD models cheat the eye of the user [2], which works very well in computer graphics, but is not sufficient for downstream applications that rely on the underlying mathematical representation of a model, e.g. additive manufacturing or mesh generation for simulations. Secondly, trimmed models yield severe robustness issues for solid modeling and the exchange of data between different systems [3]. The key source of the problem is the fact that intersections (and trimming curves) cannot be computed exactly; at least not in a fashion practical and efficient enough for CAD systems. As a result, the connectivity of the surfaces of a trimmed model is not explicitly defined and the resulting inaccuracies manifest themselves as tiny gaps and overlaps between adjacent surfaces. These model flaws may seem benign at first glance, yet the existence of these imperfections has been classified as the single most pressing unresolved problem in the field of CAD [4]. Several approaches (usually based on tolerances) have been developed to mitigate the robustness issues of trimmed models, but there is not a canonical solution to this problem. In other words, the procedures for validating trimmed CAD models may change from one CAD system to another software, making the exchange of model data a delicate issue.

Various aspects of trimmed CAD models have to be taken into account when they are directly applied to an isogeometric analysis. The specific tasks required can be summarized as: (i) identification of regular elements within \( A^v \) and elements cut by a trimming curve, (ii) tailored integration for cut elements, (iii) proper coupling between trimmed elements, and (iv) stabilization. For details on all these points the interested reader is referred to [3] and the references cited therein. In this work, it is focused on the last aspect, namely the stability which may be compromised by basis functions that are cut by a trimming curve such that only a very small portion of their support is within the domain of interest \( A^v \). The following sections present an elegant procedure to eliminate such basis functions, and how it can be improved by applying local refinement along trimming curves if needed.

2 Conditioning problems due to trimmed basis functions

A linearly independent set of univariate B-spline basis functions \( \{B_{i,p}\}_{i=0}^n \) is defined by a knot vector \( \Xi \) and a polynomial degree \( p \). \( \Xi \) specifies a non-decreasing sequence of parametric coordinates \( u_j \leq u_{j+1} \) which not only defines the overall parametric domain \( [a,b] \), but divides the parameter space into a set of knot spans \( [u_{i}, u_{i+1}] \) – the regular elements used for analysis. Each B-spline \( B_{i,p} \) is a piecewise polynomial function and the knot spans given by the knots \( \{u_i, \ldots, u_{i+p+1}\} \) define its local support \( \text{supp}(B_{i,p}) \). Furthermore, the multiplicity of the knot values determines the continuity between adjacent segments, providing control over the smoothness of the overall basis. The space of splines on \([a,b]\) is given by

\[
S_{\Xi,p}([a,b]) := \left\{ \sum_{i=0}^{n} B_{i,p}(u)c_i \mid u \in [a,b], \ c_i \in \mathbb{R}, \ i = 0, \ldots, n \right\}. \quad (1)
\]

with \( c_i \) denoting the control coefficients of the basis function.
Multivariate basis functions of any dimension $\delta$ are obtained by computing the tensor product of univariate B-splines, which are defined by separate knot vectors $\Xi_j$ with $j = 1, \ldots, \delta$:

$$B_{i,p}(u) = \prod_{j=1}^{\delta} B_{ij,pj}(u_j)$$  \hspace{1cm} (2)

with multi-indices $i = \{i_1, \ldots, i_\delta\}$ and $p = \{p_1, \ldots, p_\delta\}$, and the related univariate B-spline in the $j$-direction denoted by $B_{ij,pj}$. The former multi-index defines the position of the basis function in the tensor product structure and the latter represents the degree in each parametric direction. In general, the support of $B_{i,p}$ is given by the tensor product of the supports of its univariate components. In case of trimmed parameter spaces, however, this support may be arbitrarily cut and thus, its size within the domain of interest, i.e., $\text{supp} \{B_{i,p}\} \cap A^t$, can become arbitrarily small. B-splines with small support are indeed troublesome, since they can lead to ill-conditioned system matrices when a trimmed basis is used in an analysis [5]. In the following these basis functions are labeled as degenerate B-splines. A simply way to identify such basis functions is to use the Greville abscissa – if this characteristic point is within $A^t$ the related B-spline is stable; otherwise it is classified as degenerate.

3 Elimination of degenerate B-splines

The following extended B-spline concept has been introduced by Höllig and coworkers [6, 7] in the context of a B-spline based fictitious domain method. The key idea of the stabilization procedure is to substitute degenerate B-splines by extensions of B-splines with a sufficiently large support. In general, every B-spline $B_{i,p}$ consists of polynomial segments $B^s_i$ – one for each knot span $s$ of its support. Moreover, each knot span $s$ contains $p + 1$ non-zero B-splines represented by $B^s_i$ with $i = s - p, \ldots, s$. These functions locally span the full space of degree $p$ polynomials

$$\mathcal{P}^p([u_s, u_{s+1}]) \equiv \text{span} \{B_{s-p,p}(u), \ldots, B_{s,p}(u)\} \equiv \text{span} \{B^s_{i-p}(u), \ldots, B^s_i(u)\}. \hspace{1cm} (3)$$

In case of a trimmed knot span $t$, the contributions of this locally spanned space can lead to instabilities. This can be resolved by replacing the polynomial segments $B^s_i$ by extensions of $B^s_i$ where $s$ denotes the closest knot span which does not contain any degenerate B-splines. Since all polynomial segments are within $S_{\Xi,p}$, the extended polynomial segments $B^s_i$ of the non-trimmed knot span $s$ can be expressed exactly by B-splines of the trimmed knot span $t$ such that

$$B^s_i(u) = \sum_{j=1-t}^t B_{ij,p}(u) \alpha_{i,j} \hspace{2cm} u \in [u_t, u_{t+1}). \hspace{1cm} (4)$$

It is emphasized that this does not change $B^s_i$, but provides a means of representing the segments beyond their associated knot span $s$. The main task is to determine the values of the extrapolation weights $\alpha_{i,j}$. In principal this is done by solving an interpolation problem, but due to the properties of the functions involved some weights are trivial: If the index $j$ corresponds to a stable B-spline, the weights are given by

$$e_{i,1} = 1 \iff B^s_i(u) \equiv B_{i,p}(u), \hspace{0.5cm} u \in [u_s, u_{s+1}], \hspace{0.5cm} \forall i \in \{s - p, \ldots, s\}, \hspace{1cm} (5)$$

$$e_{i,j} = 0 \iff B^s_i(u) \not\equiv B_{j,p}(u), \hspace{0.5cm} u \in [u_s, u_{s+1}], \hspace{0.5cm} \forall j \in \{s - p, \ldots, s\} \setminus i. \hspace{1cm} (6)$$

The remaining extrapolation weights related to degenerate B-splines can be determined by

$$e_{i,j} = \frac{1}{p!} \sum_{k=0}^p (-1)^k (p - k)! \beta_{p-k} k! \alpha_k \hspace{1cm} (7)$$

with the coefficients $\alpha$ and $\beta$ referring to the constants of the polynomials

$$B^s_i(u) = \sum_{k=0}^p \alpha_k u^k \hspace{1cm} \text{and} \hspace{1cm} \psi_{j,p}(u) = \prod_{m=1}^p (u - u_{j+m}) = \sum_{k=0}^p \beta_k u^k. \hspace{1cm} (8)$$

The former is the power basis form of the extended polynomial segment, which may be derived exactly by Taylor expansion, and the Newton polynomials $\psi_{j,p}$ result from a quasi interpolation procedure, i.e., the de Boor–Fix functional [8]. A detailed discussion on the conversion of the polynomials to power basis form and the role of quasi interpolation is given in [5]. By taking the trivial extrapolations weights (5) and (6) into account, the final extended B-spline is defined as

$$B^e_{i,p} = B_{i,p} + \sum_{j \in S_t} B_{j,p} \epsilon_{i,j} \hspace{1cm} (9)$$

© 2018 The Authors. PAMM published by Wiley-VCH Verlag GmbH & Co. KGaA Weinheim. www.gamm-proceedings.com
where \( B_{i,p} \) is the stable B-spline from which the extension originates, and \( J_i \) is the index set of all degenerate B-splines related to \( B_{i,p}^0 \). Note that \( J_i \) may contain basis functions from various trimmed knot spans. Definition (9) also applies to multivariate basis functions. The corresponding extrapolation weights of a multivariate extended B-spline are determined by the tensor product of its univariate weights computed for each parametric direction.

The effect of extended B-splines is demonstrated by an example of a bivariate basis due to uniform open knot vectors \( \Xi_1 = \Xi_2 = \{0, 0.25, 0.5, 0.75, 1, 1.25, 1.5, 1.75, 2, 2.2\} \) and \( p_1 = p_2 = 2 \). This basis is trimmed by a parameter \( t \) such that the domain of interest is \( A^v = [0, t]^2 \). The condition numbers of the related spline interpolation matrix \( A \) and mass matrix \( M \) are computed. These matrices are generally defined by

\[
A[j,i] = B_i(u_j) \quad \text{and} \quad M[j,i] = \int_{A^v} B_i(u) B_j(u) \, dA^v
\]

with \( u \) denoting Greville abscissae which may have to be shifted into \( A^v \). Furthermore, the same investigations are performed with extended B-splines. The results for various positions of \( t \in (1.5, 2.0) \) are summarized in Fig. 1.

Note the peaks of the graphs corresponding to conventional B-splines that occur when \( t \) approaches a knot, e.g., the value 1.75. Based on the condition numbers reported on the left of Fig. 1, it becomes apparent that the proper substitution of degenerate B-splines by extended polynomial segments resolves this stability issues of a trimmed basis. For more information regarding the properties of extended B-splines, the interested reader is referred to [5, 7].

### 4 Enhanced control by local refinement

The extended B-spline concept can be improved by combining it with a local refinement scheme. This addition does not affect the stability properties per se, but facilitates the control over the extrapolation length \( d_e \) of extended B-splines, that is, the minimal distance of the center of the related stable knot span \( s \) to the trimming curve, in order to improve the approximation quality of these functions in the vicinity of trimming curves.

A scheme that suits extended B-splines very well are multilevel hierarchical B-splines. They provide local refinement through a sequence of \( \ell_{\max} + 1 \) nested spline spaces

\[
\mathcal{S}^{(0)}(\Omega) \subset \mathcal{S}^{(1)}(\Omega) \subset \mathcal{S}^{(2)}(\Omega) \subset \cdots \subset \mathcal{S}^{(\ell_{\max})}(\Omega)
\]

where \( \mathcal{S}^{(\ell)} := \operatorname{span} \left\{ B_{i}^{(\ell)} \right\}_{i=0}^{n(\ell)} \) and \( \Omega \) denotes the correspond parametric domain. In particular, the use of truncated hierarchical B-splines (THB-splines) [9] is proposed. B-splines that overlap with B-splines on a finer level are truncated so that a partition of unity is achieved, leading to a reduction of the overlap. This improves the conditioning of a hierarchical basis and its stability properties. In fact, a THB-basis is strongly stable with respect to the infinity norm [10]. That is, the constants to be considered in the stability analysis of the basis do not depend on the number of levels.

Using THB-splines the extrapolation length can be controlled by an admissibility criterion

\[
d_e < c_e \cdot h_u
\]

where \( h_u \) refers to the average knot span size of the initial level 0 of the basis and \( c_e \) is a user-defined constant. According to [11], a good choice for this constant is \( c_e = \frac{p}{2} \). Local refinement along trimming curves is performed by adding hierarchical
levels until the related extrapolation length $d^{(e)}_c$ complies with condition (11). For a detailed discussion the interested reader is referred to [11].

In order to assess the performance of the proposed extended THB-spline concept the following Laplace problem is considered: A unit cube defines the boundary $\Gamma$ of the domain of interest $\Omega = \mathbb{R}^3 \setminus \Omega^-$ where $\Omega^-$ refers to the void. The prescribed Neumann data along $\Gamma$ is given by

$$ t(y) = T(\bar{x}, y) $$

where $T(\bar{x}, y)$ is the fundamental solution for the flux and $\bar{x}$ is a source point located in the center of the cube. The cube is discretized by trimmed surfaces and the local refinement is driven by the admissibility criterion (11) using $c_{r} = b/2$. For comparison, a model that consists of regular surfaces only is used as well. The problem is analyzed by a direct isogeometric boundary element formulation using a collocation approach, for details see e.g., [12]. Fig. 2 shows the resulting convergence plots. Table 1 provides the number of hierarchical levels of the extended THB-basis used for each refinement step $r$; when the value is equal to 1, no refinement has been applied.

These results demonstrate that (i) optimal convergence rates are obtained and that (ii) only in few cases local refinement is actually required. Nevertheless, the ability of applying local refinement is crucial; without it the convergence graphs would have clear outliers with high relative errors at the refinement steps marked with a 2 in Table 1 (these results have been omitted in Fig. 2 for the sake of clarity) and thus, it would not be guaranteed that results improve when the mesh size $h$ is reduced.

5 Conclusions and outlook

Extended THB-splines are presented as a means to enable a stable isogeometric analysis with trimmed CAD models. Since this approach improves the property of the spline basis, it is independent of the numerical method used and can be easily applied in diverse contexts. To sum up, the stability issue of trimmed CAD models can be controlled in an effective and general manner. However, direct analysis of trimmed CAD models involves a multitude of challenges which have to be addressed (some of them are highlighted in the introduction) and they will be in the focus of future research activities.

Acknowledgements This research was supported by the Austrian Science Fund (FWF): J3884-N32.

References

[1] J. Corney and T. Lim, 3D Modeling with ACIS (Saxe-Coburg, 2001).
[2] L. A. Piegl, Computer-Aided Design 37(4), 461–470 (2005).
[3] B. Marussig and T. J. R. Hughes, Archives of Computational Methods in Engineering pp. 1–69 (2017).
[4] T. W. Sederberg, X. Li, H. W. Lin, H. Ipson, and G. T. Finnigan, ACM Transactions On Graphics 27(3), 79:1–79:8 (2008).
[5] B. Marussig, J. Zechner, G. Beer, and T. P. Fries, Computer Methods in Applied Mechanics and Engineering 316, 497–521 (2016).
[6] K. Höllig and U. Reif, Computer Aided Geometric Design 20(5), 277–294 (2003).
[7] K. Höllig, Finite Element Methods with B-Splines (Society for Industrial and Applied Mathematics, 2003).
[8] C. de Boor, A practical guide to splines (Springer, New York, NY [u.a.], 2001).
[9] C. Giannelli, B. Jüttler, and H. Speleers, Computer Aided Geometric Design 29(7), 485–498 (2012).
[10] C. Giannelli, B. Jüttler, and H. Speleers, Advances in Computational Mathematics 40, 459–490 (2014).
[11] B. Marussig, R. Hiemstra, and T. J. R. Hughes, Computer Methods in Applied Mechanics and Engineering 334, 79–110 (2018).
[12] B. Marussig, Seamless Integration of Design and Analysis through Boundary Integral Equations, Monographic Series TU Graz: Structural Analysis (Verlag der Technischen Universität Graz, 2016).