Planar domain parameterization with THB-splines

Antonella Falini[a], Jaka Špeh[a,*], Bert Jüttler[a]

[a]*Institute of Applied Geometry, Johannes Kepler University of Linz, Altenberger Str. 69, 4040 Linz, Austria*

Abstract

Isogeometric analysis uses spline parameterizations to describe the geometry of the physical domain. If such a parameterization is not available, it has to be generated from the domain boundaries. The construction of a good parameterization is crucial since it strongly influences the accuracy of the subsequent analysis. It is of interest to use adaptive techniques such as hierarchical splines for the parameterization, since this facilitates an accurate representation of detailed geometries and potentially improves the efficiency of the subsequent numerical simulation. We use truncated hierarchical (TH) B-splines to generate hierarchical spline spaces, since these functions possess many useful properties, such as linear independence and the fact that they form a non-negative partition of unity. In order to address the trade-off between computational effort and level of difficulty of a specific instance of the problem, we propose three levels of domain parameterization techniques that are based on THB-splines.

Keywords: parameterization, hierarchical splines, isogeometric analysis

1. Introduction

In this section we provide a motivation for this work and describe the related work on domain parameterization in isogeometric analysis. We then specify the problem addressed in the paper in detail and give an outline of the paper.

1.1. Motivation

The numerical simulation framework of isogeometric analysis (IGA), which has been proposed recently by Hughes et al. (2005), uses the mathematical technology of multivariate spline functions to describe both the computational domain and the functions that represent the unknown physical quantities. It has been introduced in order to address the challenging issue of data exchange between the CAD model and the numerical simulation software, which is greatly facilitated by using the same exact representation of the geometry.

However, IGA typically requires parameterizations of the *interior* of an object (a computational domain), while CAD systems only provide information about its boundary, cf. Cohen et al. (2010). Thus, methods for constructing a domain parameterization from given boundary data are required in order to make CAD models usable for IGA. In this paper we consider the case of single-patch parameterizations in the plane, where we need to assume that the computational domain is topologically equivalent to the unit square. Segmentation techniques, which split more complex domains into simpler ones that admit a single-patch parameterizations, are well-studied in the rich literature on block-structured quadrilateral mesh generation. For instance, an approach based on the medial axis has been proposed by Tam and Armstrong (1991).

A tensor-product spline parameterization of the computational domain is determined by control points, knot vectors and the degrees of the tensor-product B-splines. Typically, the knot vectors and degrees of

---

*Corresponding author, Tel. num. +43732 2468 4086

Email addresses: `antonella.falini@jku.at` (Antonella Falini), `jaka.speh@jku.at` (Jaka Špeh), `bert.juettler@jku.at` (Bert Jüttler)

the computational domain are specified by the given boundary representation. Clearly, the use of tensor-product splines imposes severe restrictions on the distribution of the degrees of freedom, due to the non-local influence of the knot lines. Using multivariate splines that provide the possibility of adaptive refinement can eliminate these restrictions. In the present paper, this is achieved by using hierarchical splines, which are represented by their basis consisting of truncated hierarchical (TH)B-splines (Giannelli et al., 2012). See Fig. 1 for a first example, where the use of adaptive refinement allows for an accurate representation of the domain boundaries while keeping the total number of degrees of freedom fairly small.
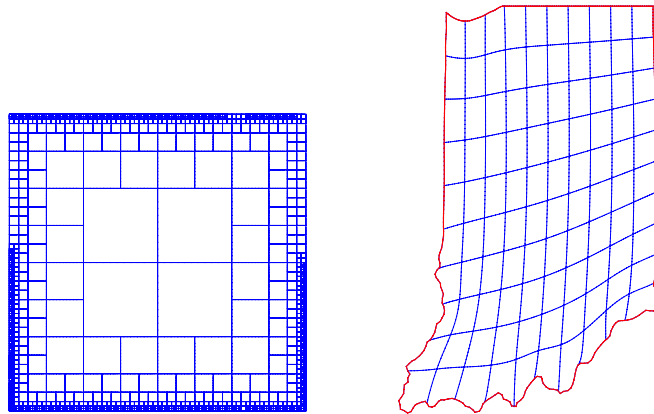

Figure 1: The hierarchical mesh in the parameter domain (left) and the parameterization of the U.S. state of Indiana (right) obtained by using hierarchical splines defined on it.

The quality of the parameterization of the computational domain is then determined by the positions of inner control points. Hence, finding a good placement of the inner control points inside the computational domain is a key issue. It is important to produce a good parameterization because this affects the accuracy of the isogeometric analysis. As a basic requirement, we need a parameterization that is free from self-intersections, so that it provides an injective mapping from the parameter domain to the computational domain. In addition, several quality measures for parameterizations have been described in the literature.

We present three levels of domain parameterization techniques that are based on THB-splines, with varying level of computational effort. The first level is formed by the linear direct method, where the parameterization is computed by solving a single linear system. The second level, which consists of the nonlinear direct method, also employs various quality measures in order to improve the properties of the parameterization. In these two levels, we exploit the adaptivity in order to obtain highly accurate representations along the boundary, see Fig. 2, left. In the third level, we use a harmonic mapping, which is computed with the help of the boundary element method (BEM), in order to construct a parameterization where the regularity is guaranteed. In this case, the use of THB-splines is needed to obtain an accurate representation of this mapping not only along the boundary, but also in the interior of the domain, see Fig. 2, right. We obtain parameterizations that are suitable for IGA, as shown in Fig. 3.

In this paper, we restrict ourselves to simple methods in the sense that they can be implemented without using algorithms for solving (possibly constrained) non-linear optimization problems. In all three levels, we use linear systems to compute the solutions. Only in the second level, we use a simple Gauss–Newton type technique for minimizing a non-linear objective function (which is a sum of squares and therefore admits this approach). Consequently, computing the solution is relatively fast and makes it possible to perform adaptive refinement, based on an iterative approach.

1.2. Related work

Parameterization methods possess numerous applications in geometric design, image processing and computer graphics, and have continuously attracted the attention of many researchers. In particular, various techniques for mesh and surface parameterization exist (Floater and Hormann, 2005; Hormann et al., 2008; Sheffer et al., 2006, and the references cited therein).
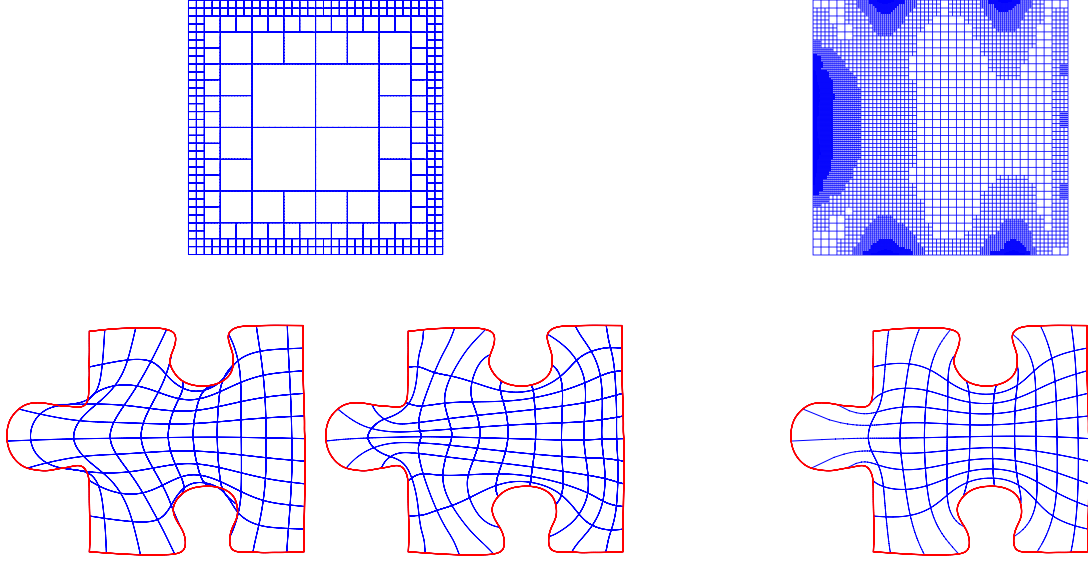
Figure 2: Left: Parameterizations of the jigsaw puzzle boundary piece obtained by using the linear (bottom left) and nonlinear (bottom right) direct methods. The same hierarchical mesh (top) is used for both parameterizations. It has been refined along the boundary in order to represent the geometry with sufficient accuracy. Right: A parameterization of the same piece obtained by using the indirect method, which is based on a harmonic mapping (bottom). In this case, the hierarchical mesh (top) is refined not only along the boundary but also in the interior of the domain, in order to represent the inverse of a harmonic mapping.
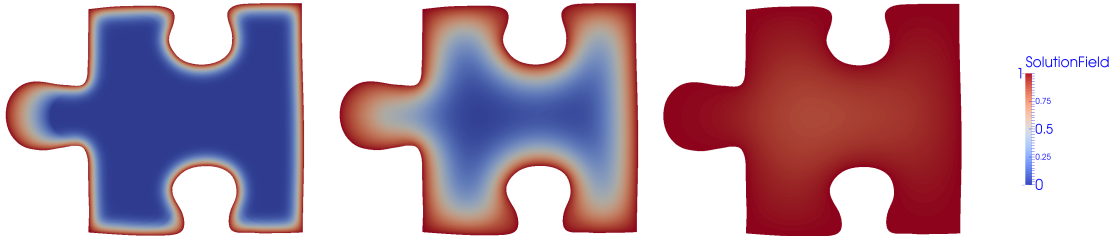

Figure 3: The solution of the heat equation on the jigsaw puzzle. The temperature at the boundary is set to 1, while the interior is initialized with temperature 0. The pictures are captured after 10, 100 and 1000 time steps (starting from left to right). See section 6 for more information.

Most approaches assume that a representation of the interior of the computational domain (e.g., a triangulation of a surface) is available. In contrast, we are interested in methods that do not make this assumption and which are therefore based on the given boundary representation.

A simple method – which still gives good results in many cases – is the use of discrete Coons patches introduced by Farin and Hansford (1999). It is explicit and computationally inexpensive. However, the resulting map needs not to be injective, and it may not be suitable for singular cases and complex shapes.

Another approach is based on the spring model (Gravesen et al., 2014). This technique is slightly more expensive than the discrete Coons patch, but the quality of the resulting parameterizations is better. Other linear methods, such as methods based on mean value coordinates (Hormann and Floater, 2006), are discussed by Gravesen et al. (2014) too. There it is also observed that while linear methods do not guarantee the injectivity of the resulting parameterization, they can be used to provide initial solutions for non-linear optimization.

If an initial parameterization is available, it can be optimized with respect to some quality functionals

(e.g., length of parameter lines, uniformity, orthogonality, skewness, eccentricity and others) in order to relocate the inner control points to a better position. A two-step method has been described by Xu et al. (2011). Firstly, an initial mapping from the parametric domain (usually the unit square) to the computational domain is constructed. Its injectivity is guaranteed by imposing certain inequality constraints on the coordinates of inner control points. Secondly, the quality of the mapping is improved by minimizing some quality functionals while preserving the injectivity of the mapping. The extension to the trivariate multi-block case has been presented by Xu et al. (2013). Reparameterization methods have been used by Xu et al. (2014).

In the context of T-splines, adaptive splines have been used for domain parameterization by Y. Zhang in several publications, (e.g. Zhang et al., 2013).

1.3. Problem statement and outline

We consider the following problem: A planar domain $\Omega \subseteq \mathbb{R}^2$, which admits a single-patch parameterization (i.e., it is diffeomorphic to the unit square) is given. We also assume that a parameterization of the boundary

$$\mathbf{b} : \partial[0,1]^2 \to \partial\Omega$$

is available. It is not necessary that this parameterization is given by spline curves; in fact the given parameterization will always be approximated by hierarchical splines.

The aim of the work is to find a mapping $\mathbf{s} : [0,1]^2 \to \Omega$, which is represented by THB-splines, such that

$$\mathbf{s}|_{\partial[0,1]^2} \approx \mathbf{b},$$

see Fig. 4. The next section summarizes the properties of THB splines and explains why this basis is simultaneously well suited for geometric modeling and for numerical simulation via IGA.
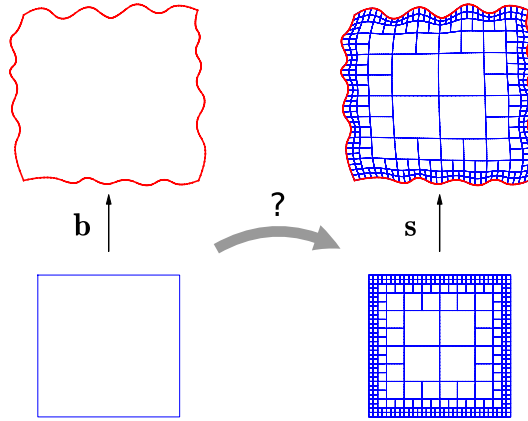

Figure 4: Left: boundary parameterization $\mathbf{b}$ of the domain. Right: adaptively refined domain parameterization $\mathbf{s}$, represented by THB-splines.

Clearly, in order to be useful for IGA, the domain parameterization $\mathbf{s}$ needs to be bijective and regular at every interior point. The shape of the boundary precludes the existence of such a parameterization if non-convex vertices are present (see Gravesen et al. (2014) for details), therefore, we will assume that this is not the case. More precisely, the boundary should consist of at least four smooth curve segments which

meet in convex vertices, and four of them should correspond to the four vertices of the parameter domain. The use of THB splines allows to represent additional vertices approximately using the adaptive refinement. (It is even possible to represent them exactly by locally inserting knots with higher multiplicity, but this has not yet been exploited in our implementation.)

In order to obtain a parameterization with these properties, we present three levels of domain parameterization techniques that are based on THB-splines, with varying level of computational effort. These techniques are described in Sections 3, 4 and 5 of the paper. They are based on a combination of several existing approaches, such as the use of quality measures, non-linear optimization, and harmonic mappings. In the latter case, in order to avoid the use of traditional mesh generation we use the boundary element method (BEM) for computing the harmonic mappings.

We discuss the performance of the different methods in Section 6. Finally we conclude the paper.

2. Preliminaries: THB-splines

Tensor-product B-splines are well established in CAD. They have many useful properties, but also a major disadvantage: They do not provide the possibility of local refinement. Indeed, the insertion of a knot into one of the knot vectors refines an entire row or column of cells in the mesh.

Several generalizations of tensor-product splines were introduced to address this problem, including T-splines, PHT-splines, LR splines and hierarchical splines (Deng et al., 2008; Dokken et al., 2013; Forsey and Bartels, 1988; Kraft, 1997; Sederberg et al., 2003). In this paper we focus on the latter approach, and we use the recently introduced basis of truncated hierarchical (TH) B-splines for the corresponding adaptively refined spline spaces (Giannelli et al., 2012). Fig. 5 shows an example of THB-splines over a hierarchical mesh with three levels of refinement.

The THB-splines possess several useful properties, which make them well-suited for applications in isogeometric analysis:

- THB-splines form a non-negative *partition of unity*. This is particularly important in geometric design, since it guarantees affine invariance and the convex hull property.

- Compared to the standard hierarchical basis of the same space introduced by Kraft (1997), the number of THB-splines that possess non-zero values at a given point is smaller. Consequently, when using hierarchical spline spaces for numerical simulation or least-squares approximation, the coefficient matrices of the linear systems obtained are *more sparse* for THB-splines than for the standard basis.

- Under certain simple conditions on the hierarchical refinement, the THB-splines are *complete*: they span the full space of piecewise polynomial functions which are available on the hierarchical grid (Mokriš et al., 2014).

- THB-splines are *strongly stable* basis with respect to the maximum norm if the knots of the different levels satisfy a quasi-uniformity condition (Giannelli et al., 2014). More precisely, the maximum norm of the coefficient vector and the L^∞ norm of the associated spline function are equivalent, and the constants in the corresponding inequalities are independent of the number of levels and on the specific choice of the knot vectors.

By using adaptive refinement of an initial tensor-product mesh we will obtain a parameterization

$$\mathbf{s}(u,v) = \sum_{\ell=1}^{L} \sum_{j=1}^{J^\ell} \mathbf{c}_j^\ell \tau_j^\ell(u,v), \quad (u,v) \in [0,1]^2, \tag{1}$$

of the domain $\Omega \subset \mathbb{R}^2$, which is represented as a linear combination of THB-splines $\tau_j^\ell(u,v)$ and coefficients $\mathbf{c}_j^\ell \in \mathbb{R}^2$. The coefficients admit an interpretation as control points since the THB splines form a non-negative partition of unity. In this equation, the upper index ℓ identifies the level, L is the total number of levels of

refinement and J^ℓ is the number of THB-splines of level ℓ. In order to simplify the notation we define the matrix of coefficients

$$\mathbf{c} = (\ldots, \mathbf{c}_j^\ell, \ldots) \in \mathbb{R}^{2,n},$$

where $n = \sum_{\ell=1}^{L} J^\ell$ is the number of THB-splines from all levels.

Two issues need to be addressed in order to find such a parameterization. Firstly we have to identify a suitable hierarchical spline space. This is done by applying adaptive refinement to an initial tensor-product space. Secondly, the unknown coefficients (control points) $\mathbf{c}$ need to be found, and this will be done by minimizing a suitable objective function. Both steps are intertwined, since the adaptive refinement is guided by the result of the second step. The following three sections present three approaches, which require an increasing amount of computational effort.

3. Linear direct methods

The simplest class of methods is based on quadratic optimization problems, which lead to systems of linear equations.

3.1. The optimization problem

The first method for obtaining the coefficients of parameterization (1) is to minimize the objective function

$$\int_{\partial[0,1]^2} \|\mathbf{s} - \mathbf{b}\|^2 + \lambda Q(\mathbf{c}) \to \min_{\mathbf{c}} \tag{2}$$

where λ is a parameter and Q is a quadratic functional. Candidates for the functional Q are

- the *parametric length functional*, which penalizes long parameter lines

$$\mathcal{Q}_\ell(\mathbf{c}) = \int_{[0,1]^2} (\|\mathbf{s}_u\|^2 + \|\mathbf{s}_v\|^2) \mathrm{d}u \, \mathrm{d}v,$$

- and the *uniformity functional*

$$\mathcal{Q}_u(\mathbf{c}) = \int_{[0,1]^2} (\|\mathbf{s}_{uu}\|^2 + 2\|\mathbf{s}_{uv}\|^2 + \|\mathbf{s}_{vv}\|^2) \mathrm{d}u \, \mathrm{d}v,$$

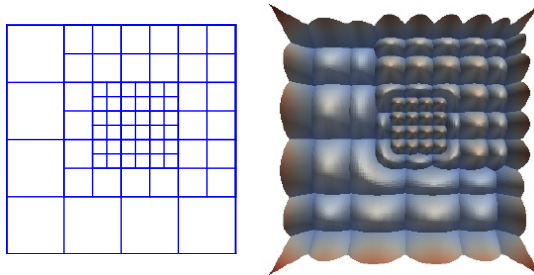which measures the thin plate spline energy or the uniformity of the parameterization.

Figure 5: Left: The hierarchical mesh is obtained by dyadic refinement (cross insertion). Right: The THB-splines of degree $(2,2)$ over the mesh. They are obtained by selecting tensor-product B-splines from all levels and then subtracting the contributions of functions selected at higher levels, see Giannelli et al. (2012) for details.
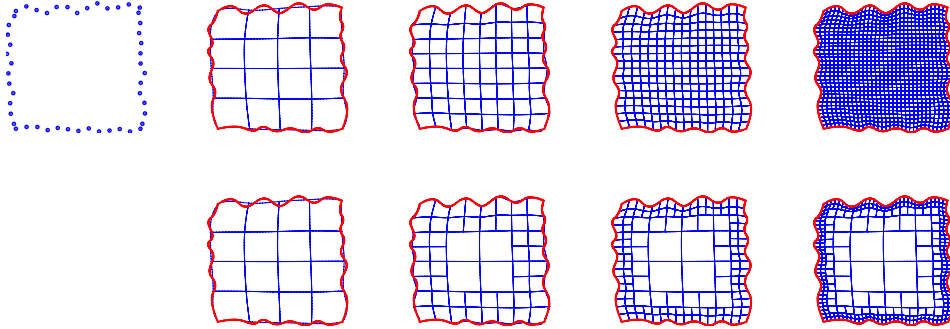
Figure 6: Uniform vs. adaptive refinement of a domain parameterization with tensor-product B-splines (top row) and THB-splines (bottom row).

The first term in Eq. (2) expresses the fact that the solution should match the boundary with high precision. Using only this term would lead to a singular system of equations, since it does not impose any constraints for the interior of the domain. Therefore, the second term is introduced. It guarantees the regularity of the system, which is obtained from the objective function.

The choice for the parameter λ is based on a trade-off between accuracy and parameterization quality. On the one hand, this value should be large enough to guarantee a real influence of the functional over the parameterization. On the other hand, since every functional penalizes the entire parameter domain, λ should be small enough to achieve a good parameterization of the boundary.

In order to find a solution to our minimization problem (2) we need to solve a linear system. Clearly, this is only possible when using quadratic functionals, since functionals of higher order would lead to non-linear systems.

Remark 1. If an exact representation of the boundary is available and compatible with the hierarchical spline space, then the first part of the objective function can be replaced by linear constraints on the THB coefficients along the boundary, or these constraints can even be used to eliminate these coefficients as unknowns from the problem. This approach would allow to preserve a given exact representation of the domain boundary.

3.2. The procedure

The input for a domain parameterization problem is a parameterization of the boundary or simply a sequence of points (with associated parameter values, possibly estimated from the data, e.g. via chord-length parameterization). See for example Fig. 6, where the input is given by a set of points. Then, the coefficients of initial B-spline surface are computed, by minimizing the objective function. The procedure marks the boundary of $\mathbf{s}$, where the error $\|\mathbf{s} - \mathbf{b}\|^2$ is above a prescribed threshold. The marked area is then refined. Note that the use of THB splines allows local refinement (Fig. 6) while tensor-product B-splines do not.

Only the boundary elements are refined by this procedure. In order to keep the number of degrees of freedom as small as possible, we restrict the refinement to the vicinity of the boundary cells.

After that we construct the new basis of THB-splines and recompute the coefficients, again by minimizing (2). This initiates a new step of the iteration. The algorithm terminates either when a maximum number of iterations is reached or when the error on the boundary is below a prescribed threshold.

Table 1 reports the difference between numbers of degrees of freedom of B-splines and THB-splines for the example in Fig. 6. We obtain the same error in the boundary (below 10^{-6} with respect to the size of the bounding box) with fewer control points.

3.3. Examples

In the example shown in Fig. 1 we have obtained a parameterization of a domain in the shape of the U.S. state of Indiana. The input points were sampled from a map.

Levels	B-splines	THB-splines	Ratio
1	49	49	100%
2	121	97	80%
3	361	193	53%
4	1225	385	31%

Table 1: Numbers of degrees of freedom of B-splines and THB-splines for the example in Fig. 6.

The domain is approximated by using THB-splines of bidegree $(3, 3)$ with 7 refinement levels. The error on the boundary is under 10^{-6}. The parameter λ for the quadratic functional $\mathcal{Q}_u$ is 10^{-12}.

The THB-splines approximate the boundary with high accuracy. Some parts of the boundary are not very complicated and there is no need for high level of refinement.

We also applied the linear direct method to the jigsaw puzzle boundary piece in Fig. 2 (bottom left). In this case, the resulting parameterization is singular. For larger values of the regularization parameter λ, the parameterization tends to become regular, but this simultaneously increases the error along the boundary. Also, adaptive refinement does not improve this situation. Thus, many domains require more sophisticated techniques for parameterization.

4. Nonlinear direct methods

Since the simple quadratic functionals from the previous section do not always lead to satisfactory results, we now propose to extend the catalog of functionals.

4.1. The optimization problem

Quality measures are functionals which measure certain properties of the parameterization. Relocation of the control points can improve the measure of the parameterization. By moving only the inner control points of the THB-spline, the boundary does not change, because it is determined by the boundary control points in the case of the clamped knot vectors.

The quadratic functionals $\mathcal{Q}_\ell$ and $\mathcal{Q}_u$ are instances of quality measures. Now we consider non-quadratic functionals also, which are taken from the rich literature on mesh generation. We restrict ourselves to functionals which are integrals of squares of some function which depends on the first and second derivatives of the parameterization,

$$\int_{[0,1]^2} F(\mathbf{s}_u, \mathbf{s}_v, \mathbf{s}_{uu}, \mathbf{s}_{uv}, \mathbf{s}_{vv})^2 \mathrm{d}u \, \mathrm{d}v.$$

Discretizing such functionals via numerical quadrature gives a *sum of squares*, which can be minimized efficiently by *Gauss-Newton-type methods*. Also, we do not consider higher order derivatives since the parameterizations are generally only C^2 smooth.

More precisely, we consider

- the *orthogonality functional*

$$\mathcal{Q}_o(\mathbf{c}) = \int_{[0,1]^2} (\mathbf{s}_u \mathbf{s}_v)^2 \mathrm{d}u \, \mathrm{d}v,$$

    which measures the orthogonality of parameter lines,

- the *skewness functional*,

$$\mathcal{Q}_s(\mathbf{c}) = \int_{[0,1]^2} \left(\frac{(\mathbf{s}_u \mathbf{s}_v)^2}{\mathbf{s}_u \mathbf{s}_u \cdot \mathbf{s}_v \mathbf{s}_v} \right)^2 \mathrm{d}u \, \mathrm{d}v,$$

    which measures the cosine of the angle between tangent vectors, and

- the *eccentricity functional*,

$$\mathcal{Q}_e(\mathbf{c}) = \int_{[0,1]^2} \left(\frac{\mathbf{s}_u \mathbf{s}_{uu}}{\mathbf{s}_u \mathbf{s}_u}\right)^2 + \left(\frac{\mathbf{s}_v \mathbf{s}_{vv}}{\mathbf{s}_v \mathbf{s}_v}\right)^2 \mathrm{d}u\,\mathrm{d}v,$$

which measures the acceleration of the parameterization projected onto the tangents along the parameter lines.

Typically, improving just one quality measure does not give good results. Instead we use a weighted linear combination

$$\mathcal{Q}(\mathbf{c}) = \omega_u \mathcal{Q}_u(\mathbf{c}) + \omega_\ell \mathcal{Q}_\ell(\mathbf{c}) + \omega_o \mathcal{Q}_o(\mathbf{c}) + \omega_s \mathcal{Q}_s(\mathbf{c}) + \omega_e \mathcal{Q}_e(\mathbf{c}) \to \min_{\mathbf{c}^*}, \tag{3}$$

where $\omega_u, \omega_\ell, \omega_o, \omega_s, \omega_e$ are positive weights and $\mathbf{c}^*$ denotes the inner control points. As mentioned before, discretization transforms all functionals into sums of squares, hence the minimum is found iteratively using Gauss-Newton method.

This technique requires an initial solution. We use the result of the linear direct method as the initial solution. The method stops if we obtain a regular parameterization or if the maximum number of iterations is reached. The regularity is checked by evaluating the sign of the Jacobian at points sampled from a fine grid.

4.2. Examples

Consider again the parameterization of the jigsaw puzzle boundary piece in Fig. 2. In this situation, we were able to obtain a regular parameterization (left column, bottom right) from an initial non-regular one (left column, bottom left). More precisely, we used 90 iterations minimizing $\mathcal{Q}_s$ with weight 1 and $\mathcal{Q}_e$ with weight 0.001, setting all the other weights to zero to find this result.

Clearly, the main problem of this approach is the selection of appropriate weights. These are found by trial and error, i.e., by testing a large catalog of possible combinations. It is not known whether a choice of weights giving a regular solution exists at all. In another slightly more complicated example, which is shown in Fig. 7, we did not succeed to obtain a regular parameterization, even after experimenting with many different linear combinations.

Another disadvantage is that we do not know whether minimizing Eq. 3 yields a regular solution for some choice of the weights at all. For example in Fig. 7, we were not able to find such parameters that the minimization would produce a regular result. The next section presents a different approach, where no user-guided selection of parameters is needed.
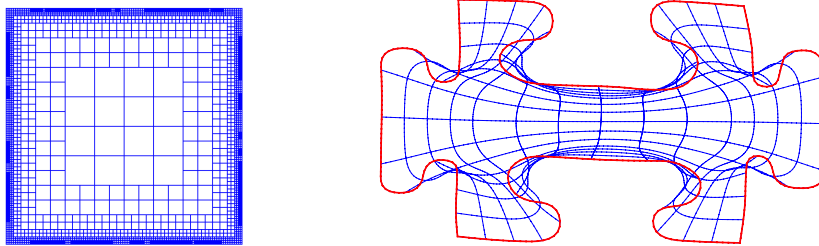

Figure 7: Left: Adaptively refined parameter domain. Right: Non-regular parameterization of a jigsaw puzzle piece (blue) and input boundary (red).

5. Indirect method

We present a different procedure for constructing the parameterization **s** of our domain Ω. We call it the "indirect" method since it is based on a mapping in the other direction: from Ω to the unit square, see Fig. 8. More precisely, we solve the Laplace equation on the computational domain using a boundary element method (BEM). This will produce a bijective harmonic map $\mathbf{h} : \Omega \to [0,1]^2$. The desired parameterization **s** will be found by approximating the inverse map $\mathbf{h}^{-1}$ using THB splines. As a major advantage, this method facilitates adaptive refinement also in the interior of the domain, since the approximation error that the inverse map provides can be used to guide the refinement.

Clearly, the Laplace equation could also be solved by a finite element method, possibly even based on isogeometric analysis. However, this would require a computational mesh or a parameterization of the domain, which is not yet available. Thus, using FEM for this step would lead to a "vicious circle", which can be avoided by the boundary element method.

A related approach has been used in Nguyen and Jüttler (2012). In that paper, the harmonic mapping was computed by a web-spline-based approach (Höllig, 2003), which requires a careful choice of the parameter that controls the influence of the boundary conditions. Also, the earlier paper was based solely on tensor-product splines. In the present paper we use a boundary element method (BEM) instead of web splines, which gives more accurate results and eliminates the need to choose a free parameter. In addition, we extend the mentioned works by introducing the possibility of adaptive refinement based on the use of THB-splines.

A valid alternative to harmonic mappings has been presented by Schneider et al. (2013). There the authors propose the novel concept of composite barycentric mappings. However, the theoretical guarantees for the bijectivity of such a mapping are currently restricted to convex polygons.

Before going into details of our method we think it is worth presenting some theoretical background concerning harmonic mappings. The remainder of this section is organized as follows: firstly we summarize several well-known results regarding harmonic mappings. This is followed by the description of our method in the BEM framework. Finally we describe the steps needed to compute the desired parameterization **s**.
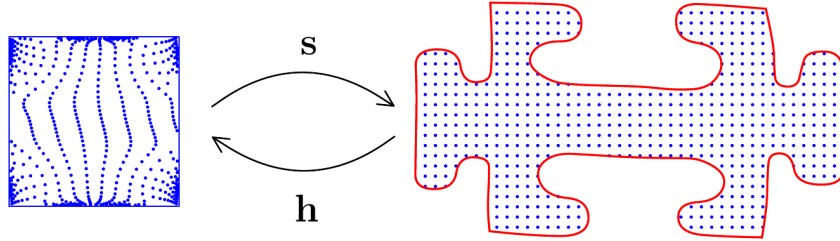

Figure 8: Parameterization by the indirect method. The harmonic map **h** maps sample points from the inside of input domain to parametric domain. Using these pairs of points, the solution **s** is constructed by least-squares approximation.

5.1. Theoretical background

Given a simply connected domain $\Omega \subset \mathbb{R}^2$, a convex target $T \subset \mathbb{R}^2$ and a boundary correspondence from $\partial\Omega$ to ∂T, there exists a univalent harmonic mapping $\mathbf{h} : \Delta\mathbf{h} = 0$. The proof is sketched in Fig. 9. Firstly, the Riemann mapping theorem ensures the existence of a unique conformal mapping r between any simply connected domain Ω and the unit disk $\mathcal{D}$. Secondly, the Radó-Kneser-Choquet theorem asserts that the unit disk $\mathcal{D}$ can always be harmonically mapped onto a convex domain T, bounded by a Jordan curve, provided a homeomorphic boundary correspondence $\varphi : \partial\mathcal{D} \to \partial T$. Indeed its harmonic extension $\hat{\varphi}$ will provide a univalent harmonic map from the unit disk to the target T. Combining this two results the existence of a univalent harmonic mapping $\mathbf{h} = r \circ \hat{\varphi}$ is guaranteed from any simply connected Jordan domain Ω onto a bounded convex domain T, since the composition $r \circ \hat{\varphi}$ is also harmonic. The required parameterization **s** will be obtained by approximating the inverse of the harmonic map **h**.
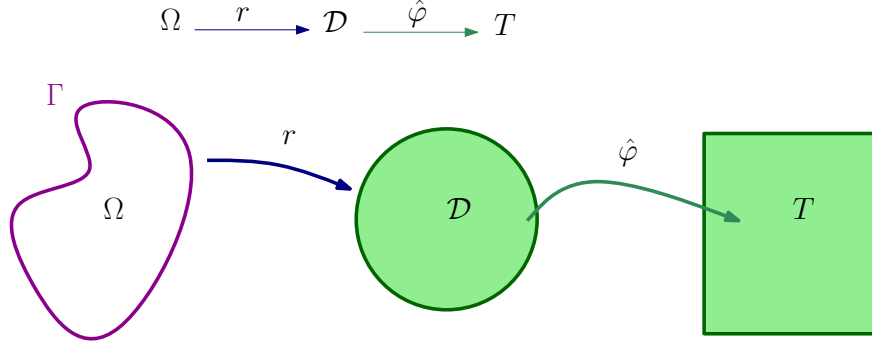
Figure 9: The Riemann mapping theorem provides a conformal map $r : \Omega \to \mathcal{D}$. The Radó-Kneser-Choquet theorem provides the harmonic extension $\hat{\varphi}$ to the unit disk of the homeomorphism $\varphi : \partial \mathcal{D} \to \partial T$.

5.2. Computation of the harmonic univalent map: A glance at BEM

The univalent harmonic map $\mathbf{h} : \Omega \to T$ consisting of two components $\mathbf{h}(\mathbf{x}) = (h_1(\mathbf{x}), h_2(\mathbf{x}))$, where $\mathbf{x} = (x_1, x_2)$, will be computed solving two Laplace problems (one for each component) with suitable Dirichlet boundary conditions. Given a map $\beta : \partial\Omega \to \partial\hat{\Omega}$, the problem is compactly formulated as follows: Find $\mathbf{h}$ such that

$$\begin{cases} \Delta \mathbf{h} = 0 & \text{in } \Omega, \\ \mathbf{h}(\mathbf{x}) = \beta(\mathbf{x}) & \text{on } \partial\Omega. \end{cases} \tag{4}$$

We compute a numerical solution using an isogeometric boundary element method (BEM). BEM is very well described in the literature (Brebbia and Dominguez, 1996; Gaul et al., 2003; Sauter and Schwab, 2011). We will limit ourselves to the discussion of the Laplace problem (4).

The idea behind BEM consists in transforming the partial differential equation into a boundary integral equation (BIE) and representing the solution of the problem by means of boundary potentials (*representation formula*). More precisely, following Rjasanow and Steinbach (2007), the starting point is the equation

$$\alpha \mathbf{h}(\mathbf{y}) = \int_{\partial\Omega} G(\mathbf{x}, \mathbf{y}) \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{n}} \, d\partial\Omega - \int_{\partial\Omega} \mathbf{h}(\mathbf{x}) \frac{\partial G(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}} \, d\partial\Omega \tag{5}$$

where

$$\alpha = \begin{cases} 1 & \text{if } y \in \Omega, \\ \frac{1}{2} & \text{if } y \in \partial\Omega, \\ 0 & \text{otherwise}. \end{cases}$$

The symbol $G(\mathbf{x}, \mathbf{y})$ denotes the Green's function

$$G(\mathbf{x}, \mathbf{y}) = -\frac{1}{2\pi} \ln(|\mathbf{x} - \mathbf{y}|) \tag{6}$$

and $\partial\mathbf{h}(\mathbf{x})/\partial\mathbf{n}$ is the directional derivative with respect to the outward-pointing normal vector of the domain boundary. This derivative will be called the *flux* of the solution. The derivative $\partial G(\mathbf{x}, \mathbf{y})/\partial\mathbf{n}$ refers to the normal derivative with respect to the first argument, i.e.,

$$\frac{\partial G(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}} = \frac{\mathrm{d}}{\mathrm{d}\tau} G(\mathrm{x} + \tau \mathrm{n_x}, \mathbf{y})|_{\tau=0},$$

where $\mathrm{x} \in \partial\Omega$ and $\mathrm{n_x}$ is the associated outward-pointing normal vector. In Eq. (5), it is assumed that the domain Ω has a C^1-smooth boundary, but the equation can be extended to domains with non-smooth boundaries also.

11

This equation has two aspects. Firstly, if both $\mathbf{h}$ and its normal derivative along the boundary are known, then it can be used to evaluate the solution at any other point. Secondly, this equation can be used to compute the flux $\partial \mathbf{h}(\mathbf{x})/\partial \mathbf{n}$, simply by restricting it to the domain boundary.

In order to perform these computations in practice, we define a 1-periodic boundary parameterization $\boldsymbol{\rho}(u)$, $u \in \mathbb{R}$ such that:

- corners of domain and target correspond to each other,

- the target boundary is also parameterized by, i.e.,

$$\mathbf{f}(u) = (f_1(u), f_2(u)) = \mathbf{h} \circ \boldsymbol{\rho}(u)$$

- the flux $\partial \mathbf{h}(\mathbf{x})/\partial \mathbf{n}$ of the solution $\mathbf{h}$ is represented as a function of u using the parameterization $\boldsymbol{\rho}$,

$$\mathbf{g}(u) = \frac{\partial \mathbf{h}}{\partial \mathbf{n}} \circ \boldsymbol{\rho}(u).$$

We use these notations and we rewrite equation (5) by substituting $\boldsymbol{\rho}(u)$ in place of $\mathbf{x}$:

$$\alpha \mathbf{h}(\mathbf{y}) = \int_0^1 G(\boldsymbol{\rho}(u), \mathbf{y}) \, \mathbf{g}(u) \, \|\dot{\boldsymbol{\rho}}(u)\| \, \mathrm{d}u - \int_0^1 \mathbf{f}(u) \frac{\partial G(\boldsymbol{\rho}(u), \mathbf{y})}{\partial \mathbf{n}} \, \|\dot{\boldsymbol{\rho}}(u)\| \, \mathrm{d}u. \tag{7}$$

Before we can use this equation to evaluate $\mathbf{h}$ at arbitrary points $\mathbf{y} \in \Omega$, we need to find the flux $\mathbf{g}$ by solving the BIE

$$\frac{1}{2}\mathbf{f}(v) = \int_0^1 G(\boldsymbol{\rho}(u), \boldsymbol{\rho}(v))\mathbf{g}(u) \, \|\dot{\boldsymbol{\rho}}(u)\| \, \mathrm{d}u - \int_0^1 \mathbf{f}(u)\frac{\partial G(\boldsymbol{\rho}(u), \boldsymbol{\rho}(v))}{\partial \mathbf{n}} \, \|\dot{\boldsymbol{\rho}}(u)\| \, \mathrm{d}u \tag{8}$$

which is derived from (7).

5.3. Discretization and collocation method

Keeping in mind the problem (4), we solve the BIE (8) for the unknown flux $\mathbf{g}$ of the solution $\mathbf{h}$, represented as a spline function

$$\mathbf{g}(u) = \sum_{j=1}^n \mathbf{c}_j N_j(u), \ u \in [0, 1]. \tag{9}$$

The B-splines $N_j(u)$ are defined over a suitable 1-periodic knot vector. The given functions are represented in the same form,

$$\begin{cases} \boldsymbol{\rho}(u) & = \sum_{j=1}^n \mathbf{d}_j N_j(u) \\ \mathbf{f}(u) & = \sum_{j=1}^n \hat{\mathbf{d}}_j N_j(u) \end{cases}$$

with known control points $\mathbf{d}_j$, $\hat{\mathbf{d}}_j$. The unknown coefficients $\mathbf{c}_j$ can be computed with three different methods: collocation method, Galerkin method and least squares method (Aimi et al., 1997; Simpson et al., 2012). We chose a collocation scheme with collocation points in the parameter domain given by Greville abscissas[1] $(v_k)_{k=1,\dots,n}$ (Johnson, 2005). This leads to a linear system with n equations and two vectors on the right-hand side,

$$A\mathbf{c} = B \tag{10}$$

where the entries of matrix A are

$$A_{kj} = \int_0^1 G(\boldsymbol{\rho}(u), \boldsymbol{\rho}(v_k))N_j(u) \, \|\dot{\boldsymbol{\rho}}(u)\| \, \mathrm{d}u, \quad (k = 1, \dots, n; \quad j = 1, \dots, n).$$

---

[1]Greville points at corners require a different value for α in the Eq. (5), since the factor $1/2$ holds only under the assumption of a C^1-smooth boundary.

The k-th row of the right hand side B takes the form

$$B_k = \frac{1}{2}\mathbf{f}(v_k) + \int_0^1 \mathbf{f}(v_k) \cdot \frac{\partial G(\boldsymbol{\rho}(u), \boldsymbol{\rho}(v_k))}{\partial \mathbf{n}} \|\dot{\boldsymbol{\rho}}(u)\| \, \mathrm{d}u, \qquad k = 1, \ldots, n$$

while the $N \times 2$-matrix $\mathbf{c}$ stores the unknown coefficients c_j. The linear system (10) has two right-hand-sides and is fully populated. It should be noted that some of the integrals in the final discretized BIE contain singular kernels due to the definition of the Green function G, thus special methods are needed to evaluate them. Several techniques to address this problem are discussed in the literature (Huang and Cruse, 1993; Jun et al., 1985; Sauter and Schwab, 2011).

5.4. THB-spline projection of the inverse mapping

The final step for computing the unknown parameterization $\mathbf{s}$ consists in approximating the inverse map $\mathbf{h}^{-1}$. This is done by projecting $\mathbf{h}^{-1}$ into the THB-spline space, which is performed by executing the following steps:

1. Sample points $\mathbf{x}_j \in \Omega, j = 0, 1, \ldots, J$.
2. Map sampled points into unit square by using $\mathbf{h}$. This produces pairs $(\mathbf{h}(\mathbf{x}_j), \mathbf{x}_j)$.
3. Use fitting to construct parameterization $\mathbf{s}$. More precisely, we solve the problem

$$\left(\sum_{j=0}^{M} \mathbf{s}(\mathbf{h}(\mathbf{x}_j)) - \mathbf{x}_j \right)^2 + \lambda Q(\mathbf{c}) \to \min_{\mathbf{c}}$$

where $Q(\mathbf{c})$ is a quadratic functional from the Eq. (2) and $\lambda \ll 1$ is a positive parameter. The quadratic functional makes the system non-singular if the number of sampled points M is too small[2].
4. If the accuracy is not sufficient, then increase the number of degrees of freedom by adaptive refinement of the mesh and continue with the previous step.

5.5. Examples

Fig. 10 shows a regular parameterization of the jigsaw puzzle piece, which was obtained using the indirect method. The direct methods that were described in the previous two sections were not able to produce a regular result. By using the possibility of adaptive refinement, which is provided by the THB-splines, we could keep a low number of degrees of freedom compared to the tensor-product B-spline case. The order of the error on the boundary is around 10^{-5}.

As a major difference to the approaches presented in the previous two sections, there are no user-selected weights. The parameterization method will give a regular results whenever the number of degrees of freedom (for solving the boundary integral integration and for the THB-spline projection of the inverse harmonic mapping) is sufficiently large. Clearly, this also means that the user does not have the possibility to fine-tune the solution by adjusting free parameters. The method has also been applied to the jigsaw puzzle boundary piece in Fig. 2 (bottom right) where it gives a different result than the non-linear direct method with user-guided selection of free parameters. Due to the uniqueness of the harmonic mapping, the result of the indirect parameterization procedure is fully determined by the boundary correspondence between the domain and the parameter domain.

---

[2]Since this method approximates the inverse harmonic map, the sampling of additional points is another possibility to eliminate problems with singular matrices. If one adopts this approach, the functional $Q(\mathbf{c})$ can be omitted.
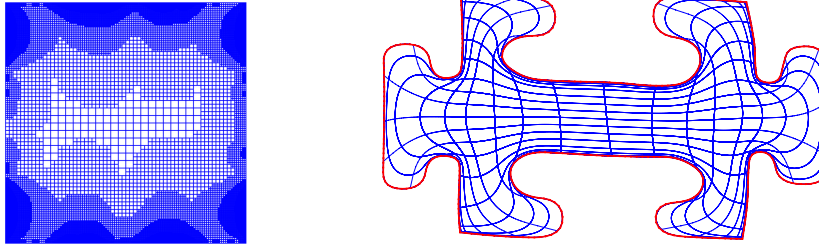
Figure 10: A parameterization of the jigsaw puzzle piece by THB-splines over an adaptively refined domain (left), which was obtained using the indirect method. The result is represented by parameter lines (right).

6. Discussion

Along with the methods of the three different levels we also presented several examples within this paper. The simplest examples, such as the U.S. state of Indiana (Fig. 1) or the square with wiggly boundary (Fig. 6) do not pose any problems and can be successfully dealt with by all three approaches.

The next class of examples contains the jigsaw puzzle boundary piece (Fig. 2). In this situation, the linear direct methods failed. Using the nonlinear direct method changed the non-regular parameterization into a regular one. This approach, however, required additional work to identify the correct choice of the free parameters (the weights in the objective function). Moreover, the result does not appear very natural since the shape of some of the parameter lines is more complicated than the shape of the domain boundaries. In this case, the indirect method gives a better looking result without the need to play around with free parameters.

Finally there are examples which can be dealt with only by using the indirect method, based on harmonic mappings, such as the jigsaw puzzle piece in Figs. 7 and 10. This example is particularly difficult since the geometry is highly distorted. In a real application one would probably prefer to perform a segmentation into smaller subdomains and to parameterize them separately. However we consider this example as a stress test for indirect method, which has the potential to deal with any domain.

Finally we use the three methods to create a parameterization of a domain in the shape of Austria (Fig. 11). The bidegree of the solution is $(3, 3)$ and the desired accuracy is 10^{-5}. The number of iterations of the direct method is equal to 4. Here both direct methods return a singular parameterization, but the nonlinear method improves the initial result of linear one and reduces the area of negative Jacobian determinant from 4% to 2% (in the parameter domain). The weights used by the nonlinear method are 1 for $\mathcal{Q}_S$ and 0.001 for $\mathcal{Q}_e$. The number of iterations is 25. Finally, the indirect finds a regular parameterization. The adaptive fitting process requires 8 iterations for the computation of a solution. The number is higher than for the direct method, but the direct method did not succeed in this example.

The resulting parameterizations can be successfully used for IGA-based simulations. As a simple example we consider the heat equation

$$\begin{aligned} u_t &= \triangle u + f & &\text{on } \Omega \times [0, T] \text{ and} \\ u &= g & &\text{on } \partial\Omega \times [0, T], \end{aligned}$$

with the unknown field u, the source f, the boundary condition g and time T. In Fig. 3, we solve it with $f = 0$ and $g = 1$. The result shows how the puzzle piece gets warmer inside.

The second example is displayed in Fig. 12. In this case the source is set to 0 everywhere except on a small circular region around Linz, where it is set to 10. The boundary conditions are set to 0. We can see how heat propagates from Linz towards the remainder of Austria.

More information about isogeometric simulations with THB-splines will be provided in a forthcoming paper (Giannelli et al., 2015).
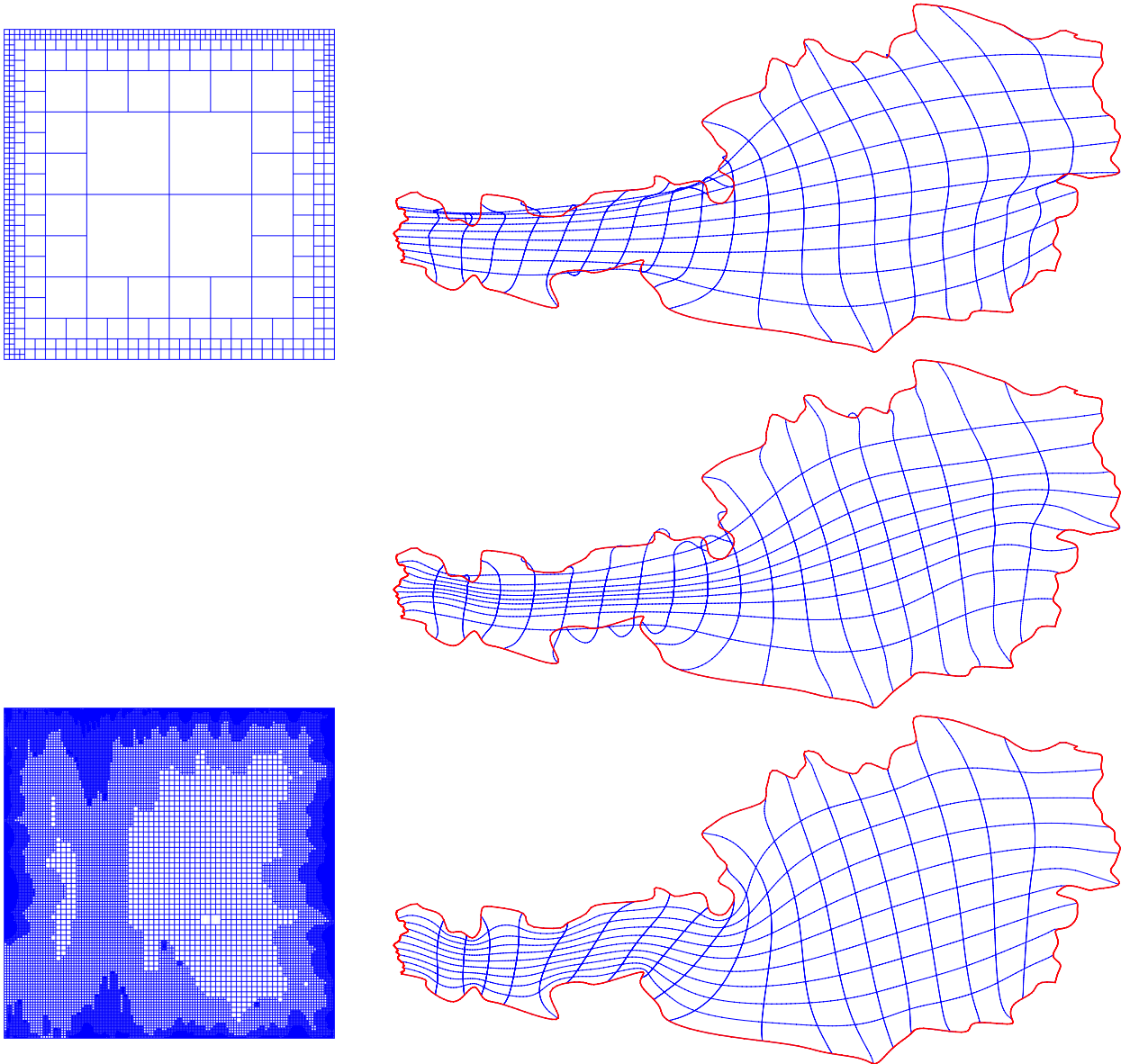
Figure 11: Right column, from top: Parameterization of Austria computed with direct linear, direct nonlinear and indirect method. Left column: The hierarchical meshes used for the first two methods (top) and for the indirect one (bottom).

7. Conclusion

We proposed several techniques for solving the planar domain parameterization problem in isogeometric analysis. These techniques, which allow adaptive refinement to be performed, are based on the mathematical technology of THB-splines. Consequently we were able to obtain parameterizations with a lower number of degrees of freedom than traditional tensor-product approaches. THB-splines possess good mathematical properties, such as the fact that they form a non-negative partition of unity, which makes them well suited for applications in geometric design and numerical simulation.

The parameterization techniques that we described in this paper were assigned to three levels with increasing computational effort. The simpler methods work for relatively simple domains, while more complicated shapes require additional efforts. The most sophisticated method, which is based on the approximation

15

Figure 12: The solution of the heat equation for the parameterization of Austria. The boundary conditions are set to 0. The source function is also set to 0 everywhere, except on a small circle with origin in Linz where is set to 10. The pictures show how heat propagates from Linz after 10 and 1000 time steps.

of the inverse of a harmonic mapping, has two main advantages: Firstly, it does not need any user-selected parameters, except for the polynomial degrees and for the initial (tensor-product) knot configuration. Secondly, it provides error estimators that can guide the adaptive refinement of the THB-splines in the entire domain. In contrast to this, the simpler techniques do not provide such estimators and lead to an adaptive refinement only along the domain boundary.

Other advanced techniques for planar domain parameterization in isogeometric analysis, such as the methods described in Gravesen et al. (2014) that are (among others) based on the minimization of the Winslow functional, rely on minimization of the (generally non-quadratic) functionals subject to (possibly non-linear) constraints. However, optimization tools typically do not guarantee to find the global minimum, finding a local one instead. This makes it difficult to provide theoretical guarantees for these approaches.

In contrast, the indirect method described in this paper is based only on relatively simple mathematical techniques which provide such guarantees. In fact, if the number of degrees of freedom is sufficiently large (and this can be guaranteed by uniform or adaptive refinement), then the boundary element method finds highly accurate approximation of the harmonic mapping (which is guaranteed to be injective), and the approximation of its inverse will also succeed.

It should be noted that, when applied to a given domain boundary, our methods produce an approximation thereof by functions in the hierarchical spline space. In many situations, there is no geometric description available that satisfies the needs of geometric design and numerical simulation, e.g., when using data generated by measurements such as geographic data. Using the IGA framework then requires only one approximation step, and all subsequent processes are based on the same representation of the geometry. This representation is then considered as "exact" even though it is only an approximation of the exact geometry.

In cases where an exact representation of the boundary is available and compatible with the hierarchical spline spaces, we can modify our methods to preserve this geometrical input, as outlined in Remark 1.

Future work will be devoted to planar domains which are not simply connected and to the three-dimensional situation. In the first case we plan to use results on harmonic mappings between multiply connected domains from Duren and Hengartner (1997). In the second case, since the theoretical results for harmonic mappings between three-dimensional domains are less strong than in the bivariate case, we may also try to use other mappings, similar to the approach used in Nguyen and Jüttler (2012).

References

Aimi, A., Diligenti, M., Monegato, G., 1997. New numerical integration schemes for applications of Galerkin BEM to 2-D problems. Int. J. Num. Meth. Engrg. 40 (11), 1977–1999.

Brebbia, C. A., Dominguez, J., 1996. Boundary elements: an introductory course. WIT press.

Cohen, E., Martin, T., Kirby, R., Lyche, T., Riesenfeld, R., 2010. Analysis-aware modeling: Understanding quality considerations in modeling for isogeometric analysis. Comp. Math. Appl. Mech. Engrg. 199 (5), 334–356.

Deng, J., Chen, F., Li, X., Hu, C., Tong, W., Yang, Z., Feng, Y., 2008. Polynomial splines over hierarchical T-meshes. Graphical Models 70 (4), 76–86.

Dokken, T., Lyche, T., Pettersen, K., 2013. Polynomial splines over locally refined box-partitions. Comp. Aided Geom. Des. 30 (3), 331–356.

Duren, P., Hengartner, W., 1997. Harmonic mappings of multiply connected domains 180, 201–220.

Farin, G., Hansford, D., 1999. Discrete Coons patches. Comp. Aided Geom. Des. 16 (7), 691–700.

Floater, M. S., Hormann, K., 2005. Surface parameterization: a tutorial and survey. In: Dodgson, N. A., Floater, M. S., Sabin, M. A. (Eds.), Advances in Multiresolution for Geometric Modelling. Mathematics and Visualization. Springer, Berlin, Heidelberg, pp. 157–186.

Forsey, D. R., Bartels, R. H., 1988. Hierarchical B-spline refinement. Computer Graphics (ACM) 22 (4), 205–212.

Gaul, L., Kögl, M., Wagner, M., 2003. Boundary Element Methods for Engineers and Scientists. Springer.

Giannelli, C., Jüttler, B., Kleiss, S., Mantzaflaris, A., Simeon, B., Špeh, J., 2015. THB-splines: An effective mathematical technology for adaptive refinement in geometric design and isogeometric analysis, in preparation.

Giannelli, C., Jüttler, B., Speleers, H., 2012. THB-splines: The truncated basis for hierarchical splines. Comp. Aided Geom. Des. 29 (7), 485 – 498.

Giannelli, C., Jüttler, B., Speleers, H., 2014. Strongly stable bases for adaptively refined multilevel spline spaces. Adv. Comp. Math. 40, 459–490.

Gravesen, J., Evgrafov, A., Nguyen, D.-M., Nørtoft, P., 2014. Planar Parametrization in Isogeometric Analysis. In: Mathematical Methods for Curves and Surfaces. Springer, pp. 189–212.

Höllig, K., 2003. Finite element methods with B-splines. Vol. 26 of Frontiers in Applied Mathematics. SIAM, Philadelphia.

Hormann, K., Floater, M. S., 2006. Mean value coordinates for arbitrary planar polygons. ACM Trans. on Graphics 25 (4), 1424–1441.

Hormann, K., Polthier, K., Sheffer, A., Dec. 2008. Mesh parameterization: Theory and practice. In: SIGGRAPH Asia 2008 Course Notes. No. 11. ACM Press, Singapore, pp. v+81.

Huang, Q., Cruse, T., 1993. Some notes on singular integral techniques in boundary element analysis. Int. J. Num. Meth. Engrg. 36 (15), 2643–2659.

Hughes, T. J., Cottrell, J. A., Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comp. Meth. Appl. Mech. Engrg. 194 (39), 4135–4195.

Johnson, R. W., 2005. Higher order B-spline collocation at the Greville abscissae. Applied Numerical Mathematics 52 (1), 63–75.

Jun, L., Beer, G., Meek, J., 1985. Efficient evaluation of integrals of order $1/r, 1/r^2, 1/r^3$ using Gauss quadrature. Engineering analysis 2 (3), 118–123.

Kraft, R., 1997. Adaptive and linearly independent multilevel B-splines. In: Surface fitting and multiresolution methods. Vanderbilt University Press, Nashville, TN, pp. 209–218.

Mokriš, D., Jüttler, B., Giannelli, C., 2014. On the completeness of hierarchical tensor-product B-splines. J. Comp. Appl. Math. 271, 53–70.

Nguyen, T., Jüttler, B., 2012. Parameterization of contractible domains using sequences of harmonic maps. In: Boissonnat, J.-D., et al. (Eds.), Curves and Surfaces. Vol. 6920 of Lecture Notes in Computer Science. Springer, pp. 501–514.

Rjasanow, S., Steinbach, O., 2007. The Fast Solution of Boundary Integral Equations. Springer New York.

Sauter, S., Schwab, C., 2011. Boundary Element Methods. Springer.

Schneider, T., Hormann, K., Floater, M. S., 2013. Bijective composite mean value mappings. Computer Graphics Forum 32 (5), 137–146.

Sederberg, T., Zheng, J., Bakenov, A., Nasri, A., 2003. T-splines and T-NURCCs. ACM Trans. on Graphics 22 (3), 477–484.

Sheffer, A., Praun, E., Rose, K., 2006. Mesh parameterization methods and their applications. Foundations and Trends in Computer Graphics and Vision 2 (2), 105–171.

Simpson, R., Bordas, S., Trevelyan, J., Rabczuk, T., 2012. A two-dimensional isogeometric boundary element method for elastostatic analysis. Comp. Math. Appl. Mech. Engrg. 209, 87–100.

Tam, T., Armstrong, C., 1991. 2D finite element mesh generation by medial axis subdivision. Adv. Engrg. Softw. Workst. 13, 313–324.

Xu, G., Mourrain, B., Duvigneau, R., Galligo, A., 2011. Parameterization of computational domain in isogeometric analysis: methods and comparison. Comp. Math. Appl. Mech. Engrg. 200 (23), 2021–2031.

Xu, G., Mourrain, B., Duvigneau, R., Galligo, A., 2013. Analysis-suitable volume parameterization of multi-block computational domain in isogeometric applications. Computer-Aided Design 45 (2), 395 – 404.

Xu, G., Mourrain, B., Galligo, A., Rabczuk, T., 2014. High-quality construction of analysis-suitable trivariate NURBS solids by reparameterization methods. Computational Mechanics 54, 1303–1313.

Zhang, Y., Wang, W., Hughes, T., 2013. Conformal solid T-spline construction from boundary T-spline representations. Computational Mechanics 51 (6), 1051–1059.