




## A Genetic Algorithm Approach for Boundary Optimization in Converting Trimmed NURBS to Untrimmed T-spline

Xiaoxiao Du<sup>1</sup> , Gang Zhao<sup>2</sup>, Wei Wang<sup>3</sup>

<sup>1</sup>Beihang University, [duxiaoxiao@buaa.edu.cn](mailto:duxiaoxiao@buaa.edu.cn)

<sup>2</sup>Beihang University, [zhaog@buaa.edu.cn](mailto:zhaog@buaa.edu.cn)

<sup>3</sup>Beihang University, [jrrt@buaa.edu.cn](mailto:jrrt@buaa.edu.cn)

Corresponding author: Wei Wang, [jrrt@buaa.edu.cn](mailto:jrrt@buaa.edu.cn)

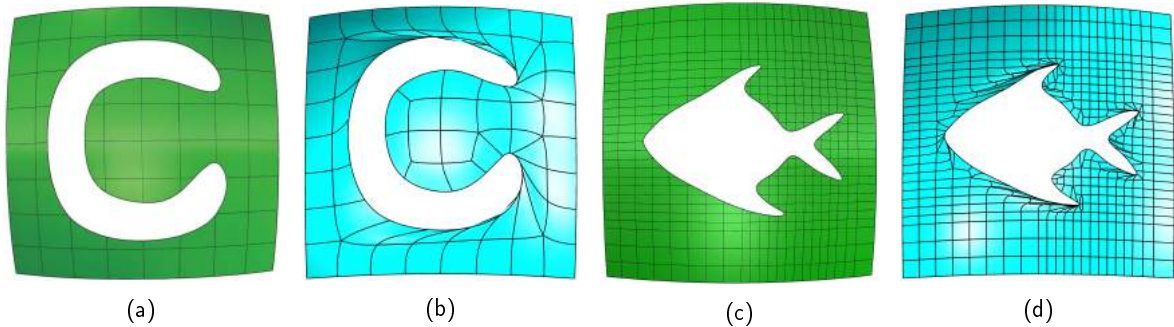
**Abstract.** For fully exploiting the elegant properties of T-splines like the local refinement and the capacity of representing complicated geometries in industry, this paper intends to solve the problem of the regional distortion and intersection that arise when converting trimmed NURBS into untrimmed T-spline surface, which is usually unstructured T-spline surface including multiple extraordinary points. A genetic algorithm is employed to optimize the fitting error from the trimming curve and the distance between the control polygon of the fitting curve and the trimming curve's predefined enclosed polygon. Bézier extraction operation for unstructured T-spline is implemented for visualization and isogeometric analysis where Gaussian quadrature rule is used for all integral terms. Several examples are investigated to verify the validity of the proposed method.

**Keywords:** T-splines, trimmed NURBS, genetic algorithm, boundary optimization, isogeometric analysis

**DOI:** <https://doi.org/10.14733/cadaps.2020.348-361>

### 1 INTRODUCTION

As unifying the mathematical expressions of conic curves/surfaces and free-form curves/surfaces, NURBS has been widely used in various fields in the past several decades (e.g., CAGD, CAE, CAM, and Computer Graphics), and also becomes the central technology to describe free-form curves and surfaces in some international standards for geometry data exchange and graphic design (e.g., IGES, STEP, and PHIGS) [13, 25]. The tensor-product topological structure of NURBS derives a lot of fast, efficient, and stable algorithms. Nevertheless, this structure limits the implementation of local refinement while superfluous control points will be propagated by using knot insertion algorithm on the NURBS surface. Moreover, it is well-known that the defects like gap and overlap are commonplace in complex engineering models consisting of trimmed NURBS surfaces. A laborious work may have to be done to make these initial 'dirty' models available for engineering applications.



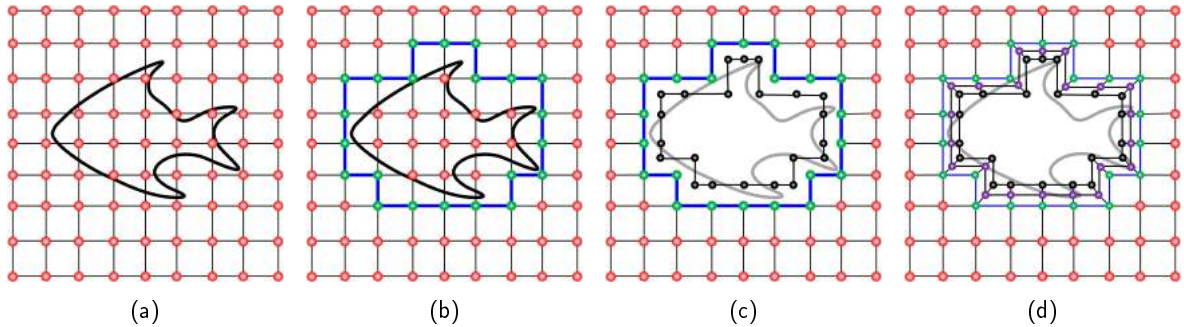
**Figure 1:** Two examples of trimmed NURBS surfaces and converted T-spline surfaces without optimization: C-trimmed and fish-trimmed surfaces. (a) and (c) Trimmed NURBS, (b) and (d) converted T-splines.

For example, due to these model problems in CAD, a lot of extra workloads, like interfaces conforming [12, 41] and trimmed elements mapping [17, 18], have to be investigated before using these models for isogeometric analysis, which was proposed to seamlessly combine CAD and FEA by directly using the spline basis functions in CAD as shape functions in FEA [2, 8].

T-splines, regarded as the generalization of NURBS, was initially proposed to enable true local refinement by allowing the existence of T-junctions [32]. Compared with NURBS, a large number of non-essential control points can be eliminated by using local refinement strategies in T-splines [30]. In addition, T-splines provides a suitable method to remove the gap and overlap on the interfaces between different patches [31]. These attractive advantages including local refinement and watertightness play important roles in the isogeometric analysis. To explore the applications of T-splines in the isogeometric analysis, on the one hand, the linear independence of T-splines blending functions was first investigated to satisfy the requirement of isogeometric analysis for shape functions [5, 20, 36]; On the other hand, different local refinement strategies have been proposed to impel the development of T-splines in the isogeometric analysis [11, 28, 40]. T-splines based isogeometric analysis, so far, has been successfully applied to contact [9, 10], crack and fracture [14, 33], shape optimization [19, 21], thin structure mechanics [24, 35], and electromagnetics [6].

With the development of the T-splines, extraordinary points are introduced to extend its ability for the representation of complex geometries. The existence of extraordinary points provides a way to convert a trimmed NURBS surface to an untrimmed T-spline surface [31], which will bring much convenience for surface representation and editing, as well as isogeometric analysis scenarios. Moreover, a so-called weighted T-splines has been proposed to reparameterize the trimmed NURBS surface without using extraordinary points in [23]. Nevertheless, watertightness and other properties are still open and need further investigations. In the process of converting a trimmed NURBS to an untrimmed T-spline, B-spline curves with the prescribed number of control points are expected to approximate the trimming curves defined in the trimmed NURBS. Then the obtained control points of the fitting curve and the vertices of the predefined enclosed polygon which encompasses the trimmed part of the original NURBS surface are connected by a middle layer of control points. However, the edges of T-mesh generated in this step are likely to be distorted and intersect with each other in the narrow neighborhood of the trimming curve, which will result in bad parameterization as shown in Fig. 1 and affect the simulation results in the context of isogeometric analysis [38].

Converting trimmed NURBS to watertight T-splines has been successfully performed in [31]. However, the phenomenon of distortion or intersection has not been discussed there. Usually based on the original model's characteristics (mainly in terms of tolerance and parameterization), the converting results tend to have Bézier elements' distortions. While poor-quality Bézier elements will further affect the accuracy of simulation results in isogeometric analysis. The reason for this phenomenon is the disconnection between the trimming curve



**Figure 2:** The process of converting a trimmed NURBS surface to an untrimmed T-spline surface. (a) A fish-shaped trimming curve, (b) the smallest axis-aligned blue polygon, (c) removing control points in polygon A and approximating the trimming curve and (d) inserting a middle layer of purple control points.

approximation and the predefined enclosed polygon. This could be improved by fitting the trimming curve by several new curves as well as optimizing the fitting algorithm. Therefore, this paper is to solve the problems of the surface's regional distortion and intersection that arise when converting trimmed NURBS into T-spline surface, which is usually unstructured T-spline surface including multiple extraordinary points. We employ a genetic algorithm to optimize the distance between the control polygon of the fitting curve and the predefined enclosed polygon thus construct certain connections between these two polygons, thus eliminate the regional distortions on the resulting surface, which are adverse to isogeometric analysis. The obtained unstructured T-spline surface will be tested for isogeometric analysis, which is a promising application for introducing T-spline into real industrial arena [29].

The outline of this paper is organized as follows. A detailed conversion of trimmed NURBS surface to T-spline surface is described in Sec. 2. Section 3 presents the multi-objective optimization problem extracted from the T-spline conversion and is solved by using a genetic algorithm. Section 4 gives the formulae about Bézier extraction from the unstructured T-mesh for isogeometric analysis. Several examples are investigated to validate the proposed method in Sec. 5. In the last, the conclusions and future works are summarized in Sec. 6.

## 2 TRIMMED-NURBS TO UNTRIMMED T-SPLINES

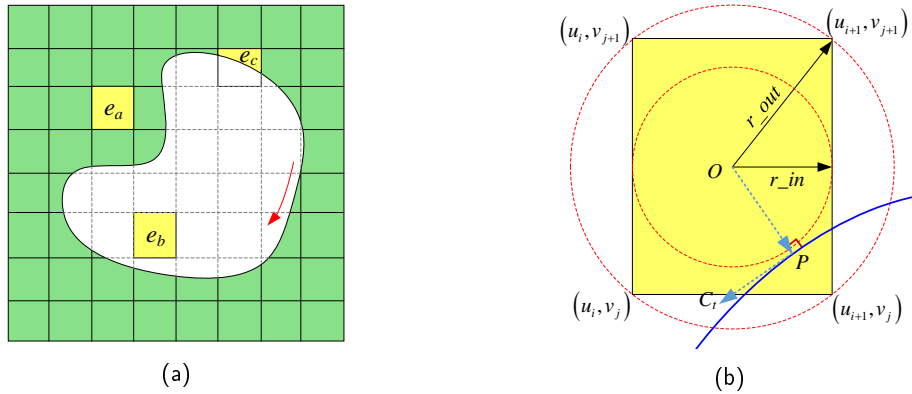
In this section, the general process of converting trimmed NURBS surface to untrimmed T-spline surface is described in Fig. 2. Subdivision method could be used to narrow the approximated error. Both the trimmed NURBS surface and untrimmed T-spline surface are bi-cubic in this paper. The whole process can be described into four steps:

**Step 1:** Inverse computation of the trimming curve.

According to the relationship between the trimming curve and NURBS surface in the physical domain, it is easy to find the corresponding trimming curve  $C$  in the NURBS surface's parametric domain through inverse computation as presented in Fig. 2a. The trimming curves in the parametric domain corresponding to the physical trimming curves usually exist in IGES files but not in STEP files.

**Step 2:** Finding an axis-aligned polygon.

Finding the smallest enclosed-polygon  $A$  (denoted by the blue polygon) of the trimming curve in the parametric grid as presented in Fig. 2b. The edges of the polygon  $A$  are axis-aligned and coincide with the iso-parametric lines of the NURBS surface. Knot refinement algorithm can be used to refine the NURBS parametric grid to obtain a closer polygon to the trimming curve and narrow the approximation error. Another thread way is to convert the untrimmed NURBS surface to T-spline firstly and then execute the local refinement



**Figure 3:** Finding the axis-aligned polygon. (a) Elements in initial NURBS mesh are classified into three types:  $e_a$ -valid element,  $e_b$ -invalid element,  $e_c$ -trimmed element; (b) trimmed element detection. The outer boundary of the mesh consisting of trimmed elements is the axis-aligned polygon.

of the T-spline, which was implemented in [31]. Nevertheless, how to efficiently realize the local refinement of T-splines is still an open question full of challenges [11, 28].

**Step 3:** Removing the invalid control points and fitting the trimming curve.

As shown in Fig. 2c, removing the control points located in the enclosed polygon  $\mathbf{A}$  and approximating the trimming curve in physical domain with a B-spline curve whose control polygon is denoted by  $\mathbf{B}$ . The number of control points in  $\mathbf{B}$  is equal to the number of vertices of polygon  $\mathbf{A}$ . Assuming that the polygon  $\mathbf{D}$  denotes the corresponding polygon of  $\mathbf{A}$  in physical space, the ideal control polygon  $\mathbf{B}$  should possess a 'similar' shape of the polygon  $\mathbf{D}$  to obtain a good parameterization. In consequence, we have to coordinate the curve approximation and the shape consistency, which is a nonlinear optimization problem. A genetic algorithm is introduced to minimize the Euclidean distance between the polygon  $\mathbf{D}$  and the control polygon  $\mathbf{B}$ .

**Step 4:** Completing the T-mesh.

Inserting a middle layer of purple control points between polygon  $\mathbf{D}$  and the control polygon  $\mathbf{B}$  as depicted in Fig. 2d. The choice of the purple control points should minimize the orthogonal distance between the trimmed NURBS surface and the converted T-spline surface. The knot interval between the purple control points and the corresponding black control points are set to be zero to satisfy the Bézier boundary conditions. It should be noticed that the choice of purple control points is also an optimization problem and will not be discussed in this paper.

An unstructured T-mesh can be built based on the above four steps. Here we discuss in detail how to find an axis-aligned polygon as presented in the second step. As depicted in Fig. 3a, the elements in the initial NURBS mesh are classified into three types:  $e_a$ -valid elements,  $e_b$ -invalid elements, and  $e_c$ -trimmed elements. For each element  $e = [u_i, u_{i+1}] \otimes [v_j, v_{j+1}]$  as presented in Fig. 3b, it is easy to find the center point  $O = [\frac{u_i+u_{i+1}}{2}, \frac{v_j+v_{j+1}}{2}]$ . Let  $r_{in}$  and  $r_{out}$  denote the radius of inscribed circle and circumcircle, we have  $r_{in} = \min(\frac{u_{i+1}-u_i}{2}, \frac{v_{j+1}-v_j}{2})$ ,  $r_{out} = \sqrt{(u_{i+1}-u_i)^2 + (v_{j+1}-v_j)^2}/2$ . Finding the projection of the center point  $O$  on the trimming curve by a point  $P$  with distance  $d = |\mathbf{OP}|$  and calculate the first derivative  $\mathbf{C}_t$  at the point  $P$ , then we firstly detect the elements by,

$$\begin{cases} \text{Case 1: } d < r_{in}, & \text{trimmed elements,} \\ \text{Case 2: } d > r_{out}, & \text{untrimmed elements,} \\ \text{Case 3: } r_{in} < d < r_{out}, & \text{further detection.} \end{cases}$$

For case 2, the untrimmed elements can be further divided into valid elements and invalid elements by calculating the cross product  $\mathbf{H} = \mathbf{OP} \times \mathbf{C}_i$ . The element is considered to be a valid element when the vector  $\mathbf{H}$  points out of the page. Otherwise, it is an invalid element. For case 3, we should further calculate the projection of the elements' four corner points on the trimming curve denoted by  $\mathbf{P}_i (i = 1, 2, 3, 4)$ . If there exists at least one point  $\mathbf{P}_i$  in the element, then the element is considered to be a trimmed element. Otherwise it is an untrimmed element and can be further detected by using the strategy for case 2. After finding all of the trimmed elements, one can acquire the axis-aligned polygon by detecting the boundary edges of the mesh consisting of trimmed elements. It is noted that half-edge data structure is an efficient and powerful structure for mesh manipulation, and therefore can be used for the procedure's implementation. Some open-sources, e.g., OPENMESH [4], CGAL [7], can be employed for a quick implementation.

### 3 OPTIMIZATION WITH GENETIC ALGORITHM

The problem of finding a 'similar' shape between the control polygon of the fitting curve  $\mathbf{B}$  and the predefined enclosed polygon  $\mathbf{D}$  is simplified to the problem of minimizing the distance between  $\mathbf{B}$  and  $\mathbf{D}$  as illustrated in Fig. 2. Meanwhile, it is expected that the fitting error is as small as possible, which is also an optimization problem. Therefore, it is a multi-objective optimization problem and can be solved by using a genetic algorithm which has been successfully applied in numerous fields including curve fitting [39].

#### 3.1 Initial Individuals and Population

According to the definition of T-mesh, the parametric length of the opposite edges in each element in T-mesh should be equal. Therefore, the knot vector of the fitting curve with control polygon  $\mathbf{B}$  can be defined according to the parametric length (knot intervals) of the enclosed polygon  $\mathbf{D}$ . For simple trimming shape, we can use a single curve for fitting while several fitting curves should be used for complicated trimming shape. Knot vectors, number of control points, degree, and parameters for data points are important factors for B-spline curve fitting. Note that knot vectors, number of control points, and degree depend on the enclosed polygon and initial NURBS surface. Therefore, the parameters for data points are the only variables for optimization.

Assuming that the number of data points is  $m$  and knot values belong to  $[a, b]$ , the parameters for the first and last data points are chosen as  $a$ , and  $b$ , the parameters corresponding to the rest of the data points denoted by  $\bar{\xi}_i, (i = 2, \dots, m - 1)$  are considered to be genes for optimization. These  $m - 2$  genes for each individual are generated randomly in  $[a, b]$  and sorted with size. The initial population consists of  $N$  individuals and is denoted by  $\bar{\xi}_N$ .

#### 3.2 Fitness Function

Two objective functions are considered in this paper: one is the distance between the polygon  $\mathbf{D}$  and the polygon  $\mathbf{B}$ , the other is the fitting error. Assume that the total number of control points in the polygon  $\mathbf{D}$  and  $\mathbf{B}$  is  $n$ . The vertices are denoted by  $\mathbf{D}_i$  and  $\mathbf{B}_i$  with  $i = 1, 2, \dots, n$ . Then the first objective function  $d_1(\bar{\xi})$  can be written as

$$d_1(\bar{\xi}) = \sum_{i=1}^n \|\mathbf{D}_i - \mathbf{B}_i(\bar{\xi})\|, \quad (1)$$

where  $\mathbf{B}_i$  is control point. The second objective function  $d_2(\bar{\xi})$  is used to minimize the fitting error and can be expressed as

$$d_2(\bar{\xi}) = \sum_{i=1}^m \|\mathbf{Q}_i - \mathbf{C}(\bar{\xi}_i)\|. \quad (2)$$

This multi-objective optimization problem can be written as

$$\min_{s.t. \bar{\xi} \in \bar{\xi}_N} \{d_1(\bar{\xi}), d_2(\bar{\xi})\}. \quad (3)$$

A linear scalarization method is used to formulate this two-objective optimization into a single-objective optimization problem by assigning a weight to each objective function. Then this modified objective function is given as

$$d(\bar{\xi}) = w_1 d_1(\bar{\xi}) + w_2 d_2(\bar{\xi}), \quad (4)$$

where  $w_1$  and  $w_2$  denote the weights and are chosen as 0.5 in this paper. It should be noticed that the objective functions  $d_1(\bar{\xi})$  and  $d_2(\bar{\xi})$  may have different magnitudes and therefore may affect the weights in the  $d(\bar{\xi})$ . So we should unify the magnitude of each objective function firstly before using linear scalarization. Let  $d_1^i(\bar{\xi})$  and  $d_2^i(\bar{\xi})$  denote the objective functions for the  $i$ -th individual, then  $d_1^i(\bar{\xi})$  are normalized as

$$d_1^i(\bar{\xi}) = \frac{d_1^i(\bar{\xi}) - \min_{j=1,2,\dots,N} \{d_1^j(\bar{\xi})\}}{\max_{j=1,2,\dots,N} \{d_1^j(\bar{\xi})\} - \min_{j=1,2,\dots,N} \{d_1^j(\bar{\xi})\}}, \quad i = 1, 2, \dots, N, \quad (5)$$

in which  $\min$  and  $\max$  mean finding the minimum and maximum value, respectively. The second objective functions  $d_2^i(\bar{\xi})$  are also normalized in the same way.

The problem is evolved into finding the minimization of the objective functions  $d(\bar{\xi})$ . For better procedure implementation, this minimization problem is transformed into a maximization problem by introducing a series of fitness functions defined as

$$f_i(\bar{\xi}) = \max_{j=1,2,\dots,N} \{d^j(\bar{\xi})\} - d^i(\bar{\xi}), \quad i = 1, 2, \dots, N, \quad (6)$$

where  $f_i(\bar{\xi})$  is the fitness value for the  $i$ -th individual and is abbreviated as  $f_i$  in the following expressions. A larger fitness value means a larger possibility of the individual to be chosen in the selection stage.

### 3.3 Selection Method

The roulette wheel method is employed to select special individuals for reproduction. Let  $f_i$  be the fitness values of individuals at a generation, the possibility of an individual being selected in the generation is computed as:

$$p_i = \frac{f_i}{\sum_{j=1}^N f_j}, \quad i = 1, 2, \dots, N, \quad (7)$$

For procedure implementation, the possibility  $p_i$  is modified by

$$p_i = \sum_{j=1}^i p_j, \quad i = 1, 2, \dots, N. \quad (8)$$

Then the  $i$ -th individual is chosen when a random value belongs to  $[p_{i-1}, p_i]$  where  $p_0 = 0, p_N = 1$ . Given  $N$  random values belonging to  $[0, 1]$ , we can build a new selected generation consisting of all chosen individuals. Note that one individual may be chosen several times.

### 3.4 Crossover Method

Crossover is to produce a "child" which inherits a lot of characteristics from its parents. A possibility value  $p_c$  is given to determine which individuals are chosen to produce "children". The selected generation obtained

after using the selection method discussed in the above subsection is divided into  $N/2$  pairs. Iterate each pair and input a random value of  $r$ , if  $r < p_c$ , this pair of individuals are chosen to produce two "children" to replace the parent individuals in the next generation. Otherwise, this pair of individuals have remained in the next generation. Let  $\mathbf{G}_1$  and  $\mathbf{G}_2$  be the selected pair of parent individuals, two children individuals  $\mathbf{G}'_1$  and  $\mathbf{G}'_2$  can be produced by:

$$\begin{aligned}\mathbf{G}'_1 &= \alpha \mathbf{G}_1 + (1 - \alpha) \mathbf{G}_2 \\ \mathbf{G}'_2 &= \alpha \mathbf{G}_2 + (1 - \alpha) \mathbf{G}_1\end{aligned}\quad (9)$$

where  $\alpha$  is a random number between 0 and 1. A so-called simulated binary crossover (SBX) operator is also an appropriate choice for real-coded crossover [1].

### 3.5 Mutation Method

Mutation, analogous to biological mutation, is to maintain genetic diversity in the reproduction by altering one or more gene values in each individual. A predefined value  $p_m$  is given to determine the possibility of mutation. For each individual, if a random value  $r < p_m$ , a random gene value in this individual is replaced by a random value. The gene values should be reordered after mutation for curve fitting. Note that the possibility of mutation ( $p_m$ ) and crossover ( $p_c$ ) can be decided adaptively [34].

## 4 BÉZIER EXTRACTION

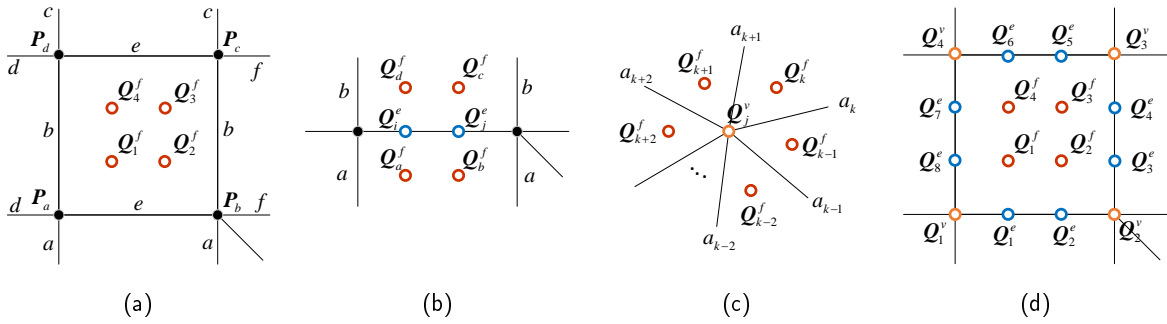
Bézier extraction operators have been widely used for linearly mapping local Bernstein basis to global splines basis like NURBS [3], T-spline [27], THB spline [15]. The transformed Bézier elements are similar to classical FEA elements and suitable for isogeometric analysis by using FEA framework without considerable modification. To make it suitable for isogeometric analysis, the unstructured T-splines generated in this paper are transformed into Bézier patches by using a simple method proposed in [29]. For each element in T-mesh, we can generate four face points, eight edge points and four vertex points, which consist of the control points of a Bézier element. As demonstrated in Fig. 4,  $\mathbf{P}_a, \mathbf{P}_b, \mathbf{P}_c, \mathbf{P}_d$  denote four corner (control) points of an element in T-mesh. Four face points are extracted from each element and two edge points can be computed from two adjacent face points for each edge. The vertex points are calculated from its one-ring neighboring face points. Superscripts  $f, e, v$  denote the face point, edge point and vertex point, respectively. The lower case letters  $a, b, c, d, e, f$  denote the knot intervals.

With control points  $\mathbf{P}_a, \mathbf{P}_b, \mathbf{P}_c, \mathbf{P}_d$ , four face points  $\mathbf{Q}_1^f, \mathbf{Q}_2^f, \mathbf{Q}_3^f, \mathbf{Q}_4^f$  given in Fig. 4a, can be computed by

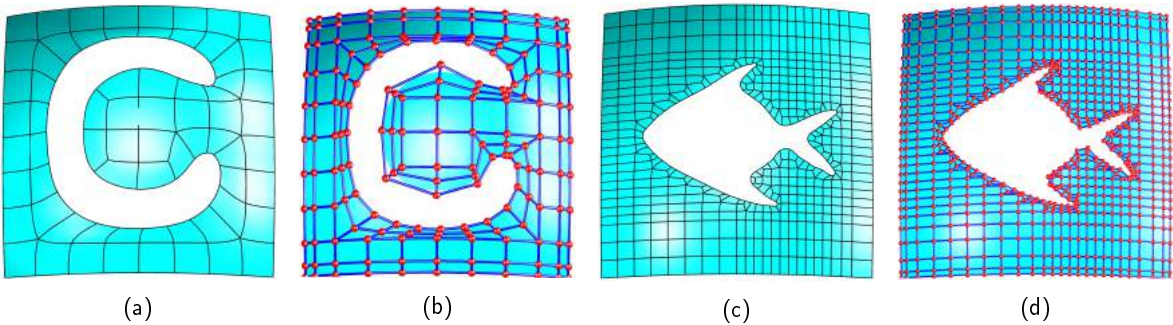
$$\begin{aligned}\mathbf{Q}_1^f &= \left(\frac{b+c}{a+b+c}\right) \left(\frac{e+f}{d+e+f}\right) \mathbf{P}_a + \left(\frac{b+c}{a+b+c}\right) \left(\frac{d}{d+e+f}\right) \mathbf{P}_b \\ &+ \left(\frac{a}{a+b+c}\right) \left(\frac{d}{d+e+f}\right) \mathbf{P}_c + \left(\frac{a}{a+b+c}\right) \left(\frac{e+f}{d+e+f}\right) \mathbf{P}_d,\end{aligned}\quad (10)$$

$$\begin{aligned}\mathbf{Q}_2^f &= \left(\frac{b+c}{a+b+c}\right) \left(\frac{f}{d+e+f}\right) \mathbf{P}_a + \left(\frac{b+c}{a+b+c}\right) \left(\frac{d+e}{d+e+f}\right) \mathbf{P}_b \\ &+ \left(\frac{a}{a+b+c}\right) \left(\frac{d+e}{d+e+f}\right) \mathbf{P}_c + \left(\frac{a}{a+b+c}\right) \left(\frac{f}{d+e+f}\right) \mathbf{P}_d,\end{aligned}\quad (11)$$

$$\begin{aligned}\mathbf{Q}_3^f &= \left(\frac{c}{a+b+c}\right) \left(\frac{f}{d+e+f}\right) \mathbf{P}_a + \left(\frac{c}{a+b+c}\right) \left(\frac{d+e}{d+e+f}\right) \mathbf{P}_b \\ &+ \left(\frac{a+b}{a+b+c}\right) \left(\frac{d+e}{d+e+f}\right) \mathbf{P}_c + \left(\frac{a+b}{a+b+c}\right) \left(\frac{f}{d+e+f}\right) \mathbf{P}_d,\end{aligned}\quad (12)$$



**Figure 4:** Bézier extraction for a bi-cubic unstructured T-spline. (a) Four face points extracted from four control points; (b) two edge points calculated from the adjacent face points; (c) vertex point computed from the 1-ring neighboring face points; (d) Sixteen control points of a bi-cubic Bézier element consisting of four face points, eight edge points, and four vertex points.



**Figure 5:** The optimized T-spline topology structure converted from the C-trimmed and fish-trimmed NURBS surfaces. (a) and (c) The optimized T-spline parametric grids, (b) and (d) the corresponding T-meshes.

$$\begin{aligned}
 Q_4^f &= \left( \frac{c}{a+b+c} \right) \left( \frac{e+f}{d+e+f} \right) P_a + \left( \frac{c}{a+b+c} \right) \left( \frac{d}{d+e+f} \right) P_b \\
 &+ \left( \frac{a+b}{a+b+c} \right) \left( \frac{d}{d+e+f} \right) P_c + \left( \frac{a+b}{a+b+c} \right) \left( \frac{e+f}{d+e+f} \right) P_d.
 \end{aligned}
 \tag{13}$$

The edge points  $Q_i^e, Q_j^e$  given in Fig. 4b, for each edge can be calculated from the adjacent face points as

$$Q_i^e = \frac{b}{a+b} Q_a^f + \frac{a}{a+b} Q_d^f,
 \tag{14}$$

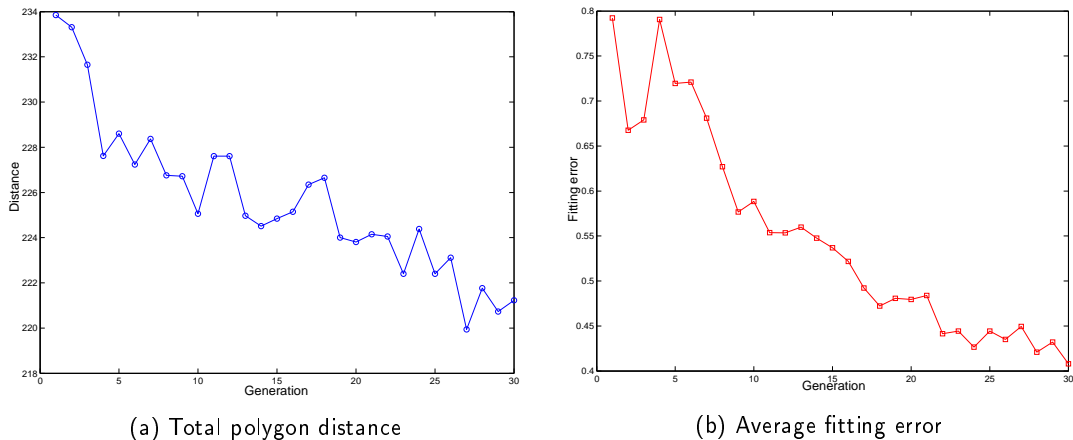
$$Q_j^e = \frac{b}{a+b} Q_b^f + \frac{a}{a+b} Q_c^f.
 \tag{15}$$

The vertex point  $Q_j^v$  shown in Fig. 4c is computed from its one-ring neighboring face points by

$$Q_j^v = \sum_{k=1}^M \left( \frac{a^{k-1}}{a^{k-1} + a^{k+1}} \right) \left( \frac{a^{k+2}}{a^k + a^{k+2}} \right) Q_k^f.
 \tag{16}$$

Based on the above Eqns. 10-16, sixteen Bézier control points can be obtained by ordering the face points, edge points, and vertex points in each T-mesh element as shown in Fig. 4d.





**Figure 6:** The variation of the polygon distance and fitting error with respect to generations in the optimization of C-trimmed model. (a) Total polygon distance and (b) average fitting error.

## 5 EXAMPLES

### 5.1 Model Conversion

Several conversion examples will be given in this subsection. Firstly, as shown in Fig. 5, we present the results of the optimized T-splines converted from trimmed NURBS given in Fig. 1. Both the distance between the enclosed polygon  $D$  and control polygon  $B$  (see Sec. 2), and the fitting error are optimized by using a genetic algorithm. It can be found that the parameterization within the narrow neighborhood of the trimming curve has been significantly improved when compared with the one shown in Fig. 1. To obtain a better-optimized result, trimming curve could be divided into several curves for approximation. The segment points can be chosen as with the extreme values of the trimming curve's curvature. In addition, to better observing the variation of the objective functions, we plot the values of distance and fitting error with respect to the generation in the C-trimmed model's conversion as given in Fig. 6. Maximum generation is 30. Ignoring the slight fluctuation, both values of distance as plotted in Fig. 6a and fitting error as plotted in Fig. 6b gradually decrease with the evolution of each generation. Apart from these two examples, we present other three examples to validate the implemented approach. As illustrated in Fig. 7, letters A-, E-, and H-trimmed NURBS surfaces are converted into untrimmed T-splines. It is observed that defects like distortion and intersection are greatly alleviated by using a genetic algorithm for optimization. The conversion time is mainly related to trimming curve fitting. Average time for each iteration is  $t = 1.1867s$  with 50 individuals and 200 data points for fitting. Note that the optimization procedure is implemented on a MATLAB platform and the computation efficiency may be improved by using C/C++ procedure.

### 5.2 Isogeometric Analysis

In this subsection, we will implement isogeometric analysis on our converted T-spline model. A benchmark problem, free vibration of an annular plate, is investigated as an example. As shown in Fig. 8, The annular plate model with outer radius  $R = 100$  and inner radius  $r = 30$  is initially built with a trimmed NURBS patch in the commercial software Rhino. Thickness  $h = 10$ , Young's modulus  $E = 200 \times 10^9$ , Poisson ratio  $\nu = 0.3$  and density  $\rho = 8000$ . The outer edge is simply supported and the inner edge is free. There are 144 Bézier elements which have been extracted from the converted T-spline model. Figure 9 presents the first ten mode shapes of the annular plate calculated by using isogeometric analysis. A dimensionless frequency parameter  $\varpi$



**Figure 7:** Three examples of converting the trimmed Letters A, E, and H into untrimmed T-splines. Left column: trimmed NURBS representation in Rhino; the second column: converted T-splines without optimization; the third and fourth columns: converted T-spline surfaces and T-meshes after optimization.

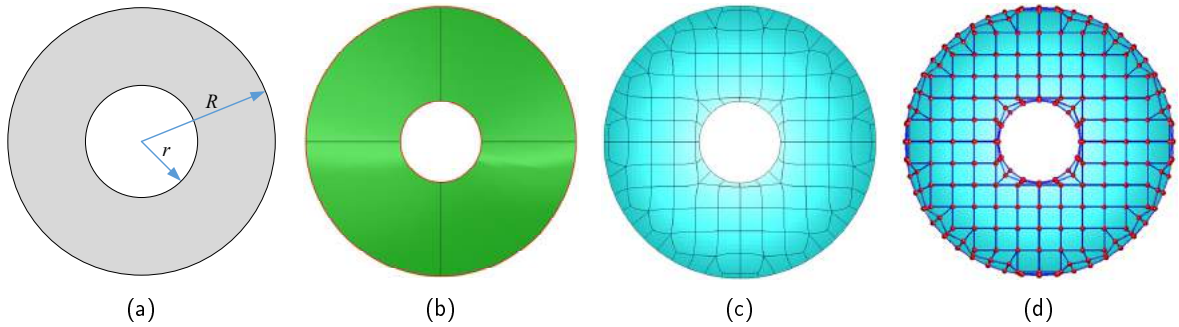
is employed for the convenience of comparison, which is defined as

$$\varpi = \omega R^2 \sqrt{\rho h / D_0}, \quad (17)$$

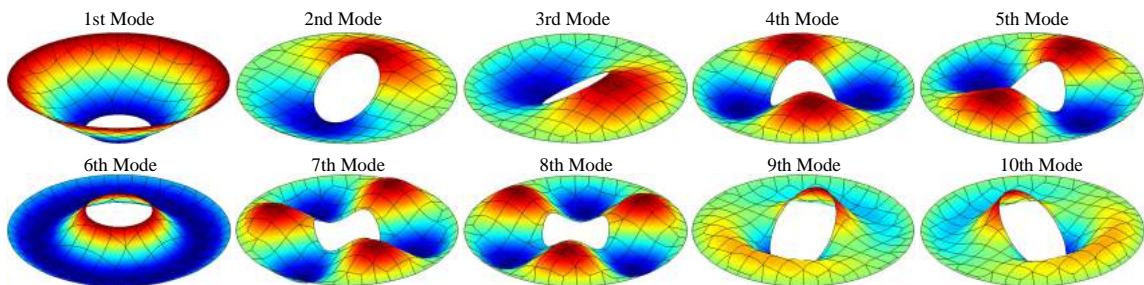
where  $\omega$  denotes the natural frequency of the annular plate. Parameter  $D_0$  can be calculated by  $D_0 = \frac{Eh^3}{12(1-\nu^2)}$ . The obtained results have been compared with standard IGA results and those from existing literature [16, 22] as listed in Table 1. It can be found that our results by using converted T-spline agree very well with existing results and that from standard IGA.

## 6 CONCLUSIONS

In this paper, the phenomenon of distortion or intersection introduced in converting trimmed NURBS to watertight T-splines is eliminated or greatly alleviated by using a genetic algorithm. In the conversion process, we ought to strike a balance between the approximation error and the good parameterization. A multi-objective optimization problem is constructed to minimize the fitting error, as well as the Euclidean distance between the fitting curve of the trimming curve and the predefined enclosed polygon. A genetic algorithm is employed for optimization. The converted T-spline surfaces show that the proposed method could maintain a good parameterization and greatly alleviate the problem of intersection and distortion. In addition, we extract the



**Figure 8:** Conversion of trimmed annular plate into untrimmed T-spline for vibration analysis. (a) Dimensions with  $r = 30, R = 100$ ; (b) Trimmed model built in the commercial software Rhino; (c) converted untrimmed T-spline and (d) T-mesh.



**Figure 9:** The first ten mode shapes of the unstructured T-spline based annular plate with simply supported outer and free inner edges,  $h/R = 0.1, r/R = 0.3$ .

Bézier elements from the converted T-spline surface model for free vibration analysis of plate by using the isogeometric approach. Future works will focus on more complicated trimming cases and the conversion of multiple trimmed NURBS patches into a single T-spline patch.

## ACKNOWLEDGEMENTS

This research was supported by National Natural Science Foundation of China (Project No. 61572056).

## ORCID

Xiaoxiao Du, <http://orcid.org/0000-0002-2324-1005>

## REFERENCES

- [1] Agrawal, R.B.; Deb, K.; Agrawal, R.B.: Simulated binary crossover for continuous search space. *Complex Systems*, 9(2), 115–148, 1995.
- [2] Bazilevs, Y.; Calo, V.M.; Cottrell, J.A.; Evans, J.A.; Hughes, T.J.R.; Lipton, S.; Scott, M.A.; Sederberg, T.W.: Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8), 229–263, 2010. <http://doi.org/10.1016/j.cma.2009.02.036>.

**Table 1:** Comparison of frequency parameters,  $\varpi = \omega R^2 \sqrt{\rho h / D_0}$ , for thick annular plates with simply supported outer and free inner edge,  $h/R = 0.1$  and  $r/R = 0.3$ .

Method	Mode types					
	(0,1)	(0,2)	(1,1)	(1,2)	(2,1)	(2,2)
Irie [16]	4.63	34.92	12.19	41.45	23.07	57.18
Liew [22]	4.6329	35.002	12.208	41.582	23.116	57.445
NURBS based IGA	4.6300	34.4074	12.1697	41.7415	22.9207	57.0244
T-spline based IGA	4.6302	34.9389	12.1478	41.4491	22.9030	57.1159

- [3] Borden, M.J.; Scott, M.A.; Evans, J.A.; Hughes, T.J.: Isogeometric finite element data structures based on Bézier extraction of NURBS. *International Journal for Numerical Methods in Engineering*, 87(1-5), 15–47, 2011. <http://doi.org/10.1002/nme.2968>.
- [4] Botsch, M.; Steinberg, S.; Bischoff, S.; Kobbelt, L.: Openmesh-a generic and efficient polygon mesh data structure. In *Proceedings of the OpenSG symposium*, 1–5, 2002.
- [5] Buffa, A.; Cho, D.; Sangalli, G.: Linear independence of the T-spline blending functions associated with some particular T-meshes. *Computer Methods in Applied Mechanics and Engineering*, 199(23-24), 1437–1445, 2010. <http://doi.org/10.1016/j.cma.2009.12.004>.
- [6] Buffa, A.; Sangalli, G.; Vázquez, R.: Isogeometric methods for computational electromagnetics: B-spline and T-spline discretizations. *Journal of Computational Physics*, 257, 1291–1320, 2014. <http://doi.org/10.1016/j.jcp.2013.08.015>.
- [7] CGAL: The Computational Geometry Algorithms Library. <https://www.cgal.org/>.
- [8] Cottrell, J.A.; Hughes, T.J.; Bazilevs, Y.: *Isogeometric analysis: toward integration of CAD and FEA*. John Wiley & Sons, 2009. <http://doi.org/10.1002/9780470749081>.
- [9] Dimitri, R.; De Lorenzis, L.; Scott, M.; Wriggers, P.; Taylor, R.; Zavarise, G.: Isogeometric large deformation frictionless contact using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 269, 394–414, 2014. <http://doi.org/10.1016/j.cma.2013.11.002>.
- [10] Dimitri, R.; Zavarise, G.: T-splines discretizations for large deformation contact problems. *Proceedings in Applied Mathematics and Mechanics*, 15(1), 183–184, 2015. <http://doi.org/10.1002/pamm.201510082>.
- [11] Dörfel, M.R.; Jüttler, B.; Simeon, B.: Adaptive isogeometric analysis by local h-refinement with T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8), 264–275, 2010. <http://doi.org/10.1016/j.cma.2008.07.012>.
- [12] Du, X.; Zhao, G.; Wang, W.: Nitsche method for isogeometric analysis of Reissner–Mindlin plate with non-conforming multi-patches. *Computer Aided Geometric Design*, 35, 121–136, 2015. <http://doi.org/10.1016/j.cagd.2015.03.005>.
- [13] Farin, G.: *Curves and surfaces for CAGD: a practical guide*. Morgan Kaufmann, 2001. <http://doi.org/10.1016/B978-1-55860-737-8.X5000-5>.
- [14] Ghorashi, S.S.; Valizadeh, N.; Mohammadi, S.; Rabczuk, T.: T-spline based XIGA for fracture analysis of orthotropic media. *Computers & Structures*, 147, 138–146, 2015. <http://doi.org/10.1016/j.compstruc.2014.09.017>.
- [15] Hennig, P.; Müller, S.; Kästner, M.: Bézier extraction and adaptive refinement of truncated hierarchical

- NURBS. *Computer Methods in Applied Mechanics and Engineering*, 305, 316–339, 2016. <http://doi.org/10.1016/j.cma.2016.03.009>.
- [16] Irie, T.; Yamada, G.; Takagi, K.: Natural frequencies of thick annular plates. *Journal of Applied Mechanics*, 49(3), 633–638, 1982. <http://doi.org/10.1115/1.3162539>.
- [17] Kim, H.J.; Seo, Y.D.; Youn, S.K.: Isogeometric analysis for trimmed CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 198(37-40), 2982–2995, 2009. <http://doi.org/10.1016/j.cma.2009.05.004>.
- [18] Kim, H.J.; Seo, Y.D.; Youn, S.K.: Isogeometric analysis with trimming technique for problems of arbitrary complex topology. *Computer Methods in Applied Mechanics and Engineering*, 199(45-48), 2796–2812, 2010. <http://doi.org/10.1016/j.cma.2010.04.015>.
- [19] Kostas, K.; Ginnis, A.; Politis, C.; Kaklis, P.: Ship-hull shape optimization with a T-spline based BEM–isogeometric solver. *Computer Methods in Applied Mechanics and Engineering*, 284, 611–622, 2015. <http://doi.org/10.1016/j.cma.2014.10.030>.
- [20] Li, X.; Zheng, J.; Sederberg, T.W.; Hughes, T.J.; Scott, M.A.: On linear independence of T-spline blending functions. *Computer Aided Geometric Design*, 29(1), 63–76, 2012. <http://doi.org/10.1016/j.cagd.2011.08.005>.
- [21] Lian, H.; Kerfriden, P.; Bordas, S.: Shape optimization directly from CAD: An isogeometric boundary element approach using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 317, 1–41, 2017. <http://doi.org/10.1016/j.cma.2016.11.012>.
- [22] Liew, K.; Yang, B.: Elasticity solutions for free vibrations of annular plates from three-dimensional analysis. *International Journal of Solids and Structures*, 37(52), 7689–7702, 2000. [http://doi.org/10.1016/S0020-7683\(99\)00306-6](http://doi.org/10.1016/S0020-7683(99)00306-6).
- [23] Liu, L.; Zhang, Y.J.; Wei, X.: Weighted T-splines with application in reparameterizing trimmed NURBS surfaces. *Computer Methods in Applied Mechanics and Engineering*, 295, 108–126, 2015. <http://doi.org/10.1016/j.cma.2015.06.020>.
- [24] May, S.; Vignollet, J.; Borst, R.d.: Powell–Sabin B-splines and unstructured standard T-splines for the solution of the Kirchhoff–Love plate theory exploiting Bézier extraction. *International Journal for Numerical Methods in Engineering*, 107(3), 205–233, 2016. <http://doi.org/10.1002/nme.5163>.
- [25] Piegl, L.; Tiller, W.: *The NURBS book*, 2nd ed. Springer, Berlin, Heidelberg, 1997. <http://doi.org/10.1007/978-3-642-97385-7>.
- [26] Renner, G.; Weiss, V.: Exact and approximate computation of B-spline curves on surfaces. *Computer-Aided Design*, 36(4), 351–362, 2004. [http://doi.org/10.1016/S0010-4485\(03\)00100-3](http://doi.org/10.1016/S0010-4485(03)00100-3).
- [27] Scott, M.A.; Borden, M.J.; Verhoosel, C.V.; Sederberg, T.W.; Hughes, T.J.: Isogeometric finite element data structures based on Bézier extraction of T-splines. *International Journal for Numerical Methods in Engineering*, 88(2), 126–156, 2011. <http://doi.org/10.1002/nme.3167>.
- [28] Scott, M.A.; Li, X.; Sederberg, T.W.; Hughes, T.J.: Local refinement of analysis-suitable T-splines. *Computer Methods in Applied Mechanics and Engineering*, 213, 206–222, 2012. <http://doi.org/10.1016/j.cma.2011.11.022>.
- [29] Scott, M.A.; Simpson, R.N.; Evans, J.A.; Lipton, S.; Bordas, S.P.; Hughes, T.J.; Sederberg, T.W.: Isogeometric boundary element analysis using unstructured T-splines. *Computer Methods in Applied Mechanics and Engineering*, 254, 197–221, 2013. <http://doi.org/10.1016/j.cma.2012.11.001>.
- [30] Sederberg, T.W.; Cardon, D.L.; Finnigan, G.T.; North, N.S.; Zheng, J.; Lyche, T.: T-spline simplification and local refinement. *ACM Transactions on Graphics*, 23(3), 276–283, 2004. <http://doi.org/10.1145/1015706.1015715>.

- [31] Sederberg, T.W.; Finnigan, G.T.; Li, X.; Lin, H.; Ipson, H.: Watertight trimmed NURBS. *ACM Transactions on Graphics*, 27(3), 79, 2008. <http://doi.org/10.1145/1399504.1360678>.
- [32] Sederberg, T.W.; Zheng, J.; Bakenov, A.; Nasri, A.: T-splines and T-NURCCs. *ACM Transactions on Graphics*, 22(3), 477–484, 2003. <http://doi.org/10.1145/1201775.882295>.
- [33] Singh, S.; Singh, I.; Mishra, B.; Bhardwaj, G.; Bui, T.: A simple, efficient and accurate Bézier extraction based T-spline XIGA for crack simulations. *Theoretical and Applied Fracture Mechanics*, 88, 74–96, 2017. <http://doi.org/doi.org/10.1016/j.tafmec.2016.12.002>.
- [34] Srinivas, M.; Patnaik, L.M.: Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man, and Cybernetics*, 24(4), 656–667, 1994. <http://doi.org/10.1109/21.286385>.
- [35] Uhm, T.K.; Youn, S.K.: T-spline finite element method for the analysis of shell structures. *International Journal for Numerical Methods in Engineering*, 80(4), 507–536, 2009. <http://doi.org/10.1002/nme.2648>.
- [36] Wang, A.; Zhao, G.; Li, Y.D.: Linear independence of the blending functions of T-splines without multiple knots. *Expert Systems with Applications*, 41(8), 3634–3639, 2014. <http://doi.org/10.1016/j.eswa.2013.12.012>.
- [37] Wang, W.; Pottmann, H.; Liu, Y.: Fitting B-spline curves to point clouds by curvature-based squared distance minimization. *ACM Transactions on Graphics*, 25(2), 214–238, 2006. <http://doi.org/10.1145/1138450.1138453>.
- [38] Xu, G.; Mourrain, B.; Duvigneau, R.; Galligo, A.: Parameterization of computational domain in isogeometric analysis: methods and comparison. *Computer Methods in Applied Mechanics and Engineering*, 200(23-24), 2021–2031, 2011. <http://doi.org/10.1016/j.cma.2011.03.005>.
- [39] Yoshimoto, F.; Harada, T.; Yoshimoto, Y.: Data fitting with a spline using a real-coded genetic algorithm. *Computer-Aided Design*, 35(8), 751–760, 2003. [http://doi.org/10.1016/S0010-4485\(03\)00006-X](http://doi.org/10.1016/S0010-4485(03)00006-X).
- [40] Zhang, J.; Li, X.: Local refinement for analysis-suitable++ T-splines. *Computer Methods in Applied Mechanics and Engineering*, 342, 32–45, 2018. <http://doi.org/10.1016/j.cma.2018.07.024>.
- [41] Zhao, G.; Du, X.; Wang, W.; Liu, B.; Fang, H.: Application of isogeometric method to free vibration of Reissner–Mindlin plates with non-conforming multi-patch. *Computer-Aided Design*, 82, 127–139, 2017. <http://doi.org/10.1016/j.cad.2016.04.006>.