Elsevier Editorial System(tm) for Computer-Aided Design
Manuscript Draft

Manuscript Number: CAD-D-13-00315

Title: An integrated design-analysis framework for three dimensional composite panels

Article Type: Research Paper

Keywords: isogeometric analysis, B-spline, NURBS, finite elements, CAD, composite panels, offset curves/surfaces

Corresponding Author: Prof. Stéphane Pierre Alain BORDAS, Ph.D.

Corresponding Author's Institution: Cardiff School of Engineering

First Author: Vinh Phu Nguyen, Dr.

Order of Authors: Vinh Phu Nguyen, Dr.; Pierre Kerfriden, Dr.; Stéphane Pierre Alain BORDAS, Ph.D.; Timon Rabczuk, Prof.

Abstract: We present an integrated design-analysis framework for three dimensional composite panels. The main components of the proposed framework consist of (1) a new curve/surface offset algorithm and (2) the isogeometric concept recently emerged in the computational mechanics community. Using the presented approach, finite element analysis of composite panels can be performed with the only input being the geometry representation of the composite surface. In this paper, non-uniform rational B-splines (NURBS) are used to represent the panel surfaces. A stress analysis of curved composite panel with stiffeners is provided to demonstrate the proposed framework.

We present an integrated design-analysis framework for three dimensional composite panels. The main components of the proposed framework consist of (1) a new curve/surface offset algorithm and (2) the isogeometric concept recently emerged in the computational mechanics community. Using the presented approach, finite element analysis of composite panels can be performed with the only input being the geometry representation of the composite surface. In this paper, non-uniform rational B-splines (NURBS) are used to represent the panel surfaces. A stress analysis of curved composite panel with stiffeners is provided to demonstrate the proposed framework.

cover.pdf

Dear Profs. Editors-in-Chief,

Please find enclosed a copy of the paper entitled ”An integrated design-analysis framework for three dimensional composite panels”, co-authored with VP Nguyen, P Kerfriden and T Rabczuk, which we would like to submit to Computer Aided Design.
This is an original paper which has neither previously, nor simultaneously, in whole or in part been submitted anywhere else. We look forward to hearing if it is considered of interest for CAD.

Best regards,

Stephane Bordas

hightlights.pdf

1. An integrated CAD-FEA framework for composite panels is described
2. Algorithms to compute offsets of NURBS curves/surfaces are presented that allow to build analysis-suitable trivariate NURBS representation of CAD surfaces

An integrated design-analysis framework for three dimensional composite panels

Vinh Phu Nguyen[a,1], Pierre Kerfriden[a,2], Stéphane P.A. Bordas[a,3,*], Timon Rabczuk[b,4]

[a] *School of Engineering, Institute of Mechanics and Advanced Materials, Cardiff University, Queen's Buildings, The Parade, Cardiff CF24 3AA*
[b] *Institute of Structural Mechanics, Bauhaus-Universität Weimar, Marienstraße 15 99423 Weimar*

Abstract

We present an integrated design-analysis framework for three dimensional composite panels. The main components of the proposed framework consist of (1) a new curve/surface offset algorithm and (2) the isogeometric concept recently emerged in the computational mechanics community. Using the presented approach, finite element analysis of composite panels can be performed with the only input being the geometry representation of the composite surface. In this paper, non-uniform rational B-splines (NURBS) are used to represent the panel surfaces. A stress analysis of curved composite panel with stiffeners is provided to demonstrate the proposed framework.

Keywords: isogeometric analysis, B-spline, NURBS, finite elements, CAD, composite panels, offset curves/surfaces

1. Introduction

Two important industries in engineering are Computer Aided Design (CAD) and Finite Element Analysis (FEA). FEA was developed to improve analysis in engineering and CAD was developed to improve the design process. The FEA evolution started in the 1940s whereas the CAD became a mature field much later in the 1970s. That explains why different mathematical models have been employed to represent the same object. In FEA, trivariate polynomials of low order (usually one or two) are used to approximate the solid object while in CAD, the same object is represented by NURBS (Non Uniform Rational B-splines). Due to the difference in the geometrical representation, the transfer from a CAD model to a FEA model requires another technology–the so-called mesh generators that transform the CAD model in to a finite element (FE) mesh that is suitable for a FE computation. Meshing complex structures is, however, a very time-consuming process. Furthermore there is no way to go from FEA back to CAD.

It is probably that the first work that attempted to link CAD and FEA was the work of Kagan and his co-workers [1, 2]. In the referred works, B-splines were utilized to represent the solid geometry in the FE model. Therefore, both CAD and FEA models employ the same technology–B-splines to construct the object of interest. Along this line of research, another notable contribution was made by Cirak et al [3, 4] in which subdivision surface, which is a CAD technology extensively used in animation [5], was used in a finite element thin shell model. It was not until 2005 that the idea was generalized and a new field was emerged–the isogeometric analysis (IGA) by Hughes and his co-workers in the seminal paper [6] where NURBS were adopted in FE solid/structural/fluid mechanics models. Since

*Corresponding author
[1] nguyenpv@cardiff.ac.uk, ORCID: 0000-0003-1212-8311
[2] pierre@cardiff.ac.uk
[3] stephane.bordas@alum.northwestern.edu, ORCID: 0000-0001-7622-2193
[4] timon.rabczuk@uni-weimar.de

this seminal paper, a monograph has been published entirely on the subject [7] and applications have been found in several fields including structural mechanics, solid mechanics, fluid mechanics, biomechanics and contact mechanics. It has also gained popularity in shape optimization community, see e.g., [8, 9] and references therein. We refer to [10, 11] for recent works on IGA with industrial applications and [12] for an overview of IGA, its recent developments and its computer implementation aspects. Not only IGA reduces the gap between CAD and FEA, but also it has triggered a new drive in spline research after a quiet period, see for instance the locally refined splines reported in [13], the polynomial splines over hierarchical T-meshes (PHT) in [14, 15]. There is an increasing communication between CAD and FEA researchers, see e.g., [16, 17, 18, 19, 20]. Particularly, in [16], a new concept coined analysis-aware-modeling was proposed in which CAD model parameters are selected to facilitate isogeometric analysis. T-splines–a generalization of NURBS developed in [21] were also used in a FE context, see e.g., [19].

In this paper we are going to present an integrated CAD-FEA framework for design and analysis of composite panels which have been extensively used in automotive and aircraft industries due to their high strength and low weight. As a CAD object, the composite panel is described by a NURBS surface which can be directly imported into our isogeometric finite element code. Since our ultimate goal is to perform a failure analysis of the composite panel due to delamination at the interfaces between the plies of which promising preliminary results are reported in [22, 23], a trivariate representation of the composite is required. It should be emphasized that if a shell model is sufficient, then the bivariate NURBS surfaces can be directly used in a FE package without any further complication see e.g., [24, 25]. To this end, we develop a simple algorithm that for a given NURBS surface and a thickness of the panel, a trivariate NURBS can be built. The algorithm is based on the computation of offsets of a NURBS surface which is a topic of extensive research in the CAD community, see e.g., [26, 27, 28, 29, 30] and references therein. However existing offset algorithms cannot be used directly for our FE analyses, we therefore devise a new algorithm to compute offsets of NURBS curves and surfaces. Our contribution certainly enlarges the application field of IGA to composite structures.

Traditional methods used to approximate offset curves (surfaces) fall into two categories, those that use the geometry and topology of the original curve (surface) to manipulate the control points to produce an offset approximation, and those that use sampling points from the exact offset curve (surface) as input for an approximation method that fits a curve (surface) to the offset. Our offsetting algorithm falls into the latter group and is able to generate exact offsets for circles and lines. It also produces non self-intersecting offset curves. We use an optimization algorithm (specially the gradient descent method) to iteratively move the sought-for offset curve/surface from a starting position to the "exact" offset curve/surface. We then use this algorithm to generate analysis-suitable trivariate NURBS solids which model a class of objects of important applications–curved composite panels.

The remainder of the paper is organized as follows. Section 2 presents a new algorithm for computing of offset of NURBS curves and surfaces. Section 3 is devoted to a discussion on isogeometric finite elements followed by some numerical examples given in Section 4. Section 5 ends the paper with some concluding remarks.

2. Computing offset of NURBS curves/surfaces

2.1. NURBS curves and surfaces

We briefly discuss the B-splines/NURBS curves and surfaces here, for details we refer to the standard textbook [31]. NURBS basis functions are defined as

$$R_{i,p}(\xi) = \frac{N_{i,p}(\xi)w_i}{W(\xi)} = \frac{N_{i,p}(\xi)w_i}{\sum_{j=1}^{n} N_{j,p}(\xi)w_j} \tag{1}$$

where $N_{i,p}(\xi)$ denotes the ith B-spline basis function of order p and w_i are a set of n positive weights. Selecting appropriate values for the w_i permits the description of many different types of curves including polynomials and

circular arcs. For the special case in which $w_i = c, i = 1, 2, \ldots, n$ the NURBS basis reduces to the B-spline basis of which definition is described in what follows.

Given a knot vector $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$, the B-spline basis functions are defined recursively starting with the zeroth order basis function ($p = 0$) given by

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

and for a polynomial order $p \geq 1$

$$N_{i,p}(\xi) = \frac{\xi - \xi_i}{\xi_{i+p} - \xi_i} N_{i,p-1}(\xi) + \frac{\xi_{i+p+1} - \xi}{\xi_{i+p+1} - \xi_{i+1}} N_{i+1,p-1}(\xi) \tag{3}$$

This is referred to as the Cox-de Boor recursion formula. Note that the first and last knots of Ξ have $p+1$ multiplicity so that the NURBS basis are interpolatory there.

A NURBS curve is given by

$$\mathbf{C}(\xi) = \sum_{I=1}^{n} R_{I,p}(\xi) \mathbf{P}_I \tag{4}$$

where n denotes the number of basis functions–also the number of control points and $\mathbf{P}_I \in \mathbb{R}^d$ (d is the number of spatial directions) are the control points.

Given two knot vectors (one for each direction) $\Xi = \{\xi_1, \xi_2, \ldots, \xi_{n+p+1}\}$ and $\mathscr{H} = \{\eta_1, \eta_2, \ldots, \eta_{m+q+1}\}$ and a control net $\mathbf{P}_{i,j} \in \mathbb{R}^d$, a tensor-product NURBS surface is defined as

$$\mathbf{S}(\xi, \eta) = \sum_{i=1}^{n} \sum_{j=1}^{m} R_{i,j}^{p,q}(\xi, \eta) \mathbf{P}_{i,j} \tag{5}$$

where $R_{i,j}^{p,q}$ are given by

$$R_{i,j}^{p,q}(\xi, \eta) = \frac{N_i(\xi) M_j(\eta) w_{i,j}}{\sum_{\hat{i}=1}^{n} \sum_{\hat{j}=1}^{m} N_{\hat{i}}(\xi) M_{\hat{j}}(\eta) w_{\hat{i},\hat{j}}} \tag{6}$$

Defining a global index as

$$I = n(j-1) + i \tag{7}$$

Eq. (5) can be rewritten in a more compact form as

$$\mathbf{S}(\boldsymbol{\xi}) = \sum_{I=1}^{n \times m} R_I^{p,q}(\boldsymbol{\xi}) \mathbf{P}_I \tag{8}$$

in which $R_I^{p,q}$ is a bivariate NURBS basis function defined as $R_I^{p,q}(\boldsymbol{\xi}) = R_{i,j}^{p,q}(\xi, \eta)$. This compact form will be used extensively in an analysis context. A definition of a trivariate tensor-product NURBS solid is similar and hence not presented here.

2.2. NURBS curve offsets

The offset curve of $\mathbf{C}(\xi) = (x(\xi), y(\xi))^{\mathrm{T}}$, denoted by $\mathbf{C}^o(\xi)$ is defined by

$$\mathbf{C}^o(\xi) = \mathbf{C}(\xi) + t\mathbf{n}(\xi) \tag{9}$$

in which t is the offset distance which is a constant and $\mathbf{n}(\xi)$ is the unit normal defined by for planar curves

$$\mathbf{n}(\xi) = \frac{(-y'(\xi), x'(\xi))}{\sqrt{x'(\xi)^2 + y'(\xi)^2}} \tag{10}$$

Due to the square root appeared in the unit normals, offset curves cannot be exactly represented as polynomial or rational curves. That is the reason one has to resort to approximations. Note that the notation $x'(\xi)$ refers to the first derivative of x with respect to ξ.

The n control points of the base curve are represented by $\mathbf{P}_0$. Our goal is to generate an offset curve with offset distance denoted by t from the given (so-called progenitor) B-spline curve subjected to the requirement that the offset curve must have the same parametrization as the original curve. That is, the offset curve is of p degree and has the same knot vector Ξ as its progenitor. The control points of the offset curve are denoted by $\mathbf{P}$ and are the unknowns[5] to be computed. The requirement that the offset curve is of the same format as its progenitor is due to the fact that our goal is to build a trivariate NURBS from the original surface and its offset surface. This restriction comes from the tensor-product nature of NURBS surfaces/solids.

In the first step, a set of points on the offset curve is generated. To this end, we divide the knots Ξ into a number of equal intervals. For a knot $\xi_i (i = 1, m)$, an offset point is given by

$$\begin{aligned} \mathbf{x}_{0i} &= \sum_{I}^{n} R_I(\xi_i)\mathbf{P}_{0I} \\ \mathbf{x}_i &= \mathbf{x}_{0i} + t\mathbf{n}_i \end{aligned} \tag{11}$$

where $\mathbf{n}_i$ represents the unit normal vector to the progenitor curve at point $\mathbf{x}_{0i}$. Once computed, the set of offset points $\{\mathbf{x}_i\}_{i=1}^m$ is fixed.

Since the offset curve goes through the first and last control points with open knots vectors, we have $\mathbf{P}_1 = \mathbf{x}_1$ and $\mathbf{P}_n = \mathbf{x}_m$. Hence, there are $(n-2)$ remaining control points to be determined. As the first guess for the gradient descent method, we may assume that the offset curve is initially a line i.e., its control points locate on the line connecting $\mathbf{P}_1$ and $\mathbf{P}_n$. Another option is to choose the control points coincides with the control points of the base curve $\mathbf{P}_i^{(0)} = \mathbf{P}_{0i},\ (i = 2, \ldots, n-1)$. The superscript (k) is used to indicate quantities at k iteration in the gradient decent algorithm– k is the iteration index. On this initial offset curve, a set of sampling points $\{\mathbf{x}_i^{(k)}\}_{i=1}^m$ $(k = 0)$ is generated which has the same number of points as $\{\mathbf{x}_i\}$. We then define a system of springs in which each spring connects one point in $\{\mathbf{x}_i\}$ and one point in $\{\mathbf{x}_i^{(0)}\}$. The energy of this system of springs is given by

$$E(\mathbf{P}) = \frac{1}{2}\sum_{i}^{m} k_s u_i(\mathbf{P})^2 \tag{12}$$

[5]In our current formulation only the coordinates of the control points are the unknowns to be determined. The weights are inherited from the weights of the base curve. It is certainly possible to consider the weights as unknowns. However this is not yet studied in the scope of this contribution.

in which k_s represents the spring stiffness. In our implementation, $k_s = 1$ is used. u_i is computed as the distance between two points defining the ith spring . As E is approaching its minimum value, the sought-for offset curve is moving towards the "exact" offset curve defined by the offset points $\{\mathbf{x}_i\}$. Note that the energy is a function of the unknown control points $\mathbf{P}$ that defines the sought-for offset curve.

In order to find the minimum of E we use the gradient descent method combined with a line search. The control points of the first iteration $\mathbf{P}^{(1)}$ are then computed according to

$$\begin{aligned} \mathbf{P}^{(1)} &= \mathbf{P}^{(0)} - \gamma^{(0)} \nabla E^{(0)} \\ \gamma^{(0)} &= \arg\min_{\gamma^{(0)}} E(\mathbf{P}^{(0)} - \gamma^{(0)} \nabla E^{(0)}) \end{aligned} \tag{13}$$

where γ denotes the so-called step length. The second equation of Eq. (13) denotes a one dimensional minimization problem which is therein solved using a backtracking line search method [32]. The proposed algorithm is graphically illustrated in Fig. (1) and given in Algorithm 1. The iterative process proceeds until the energy is smaller than a predefined tolerance ϵ.

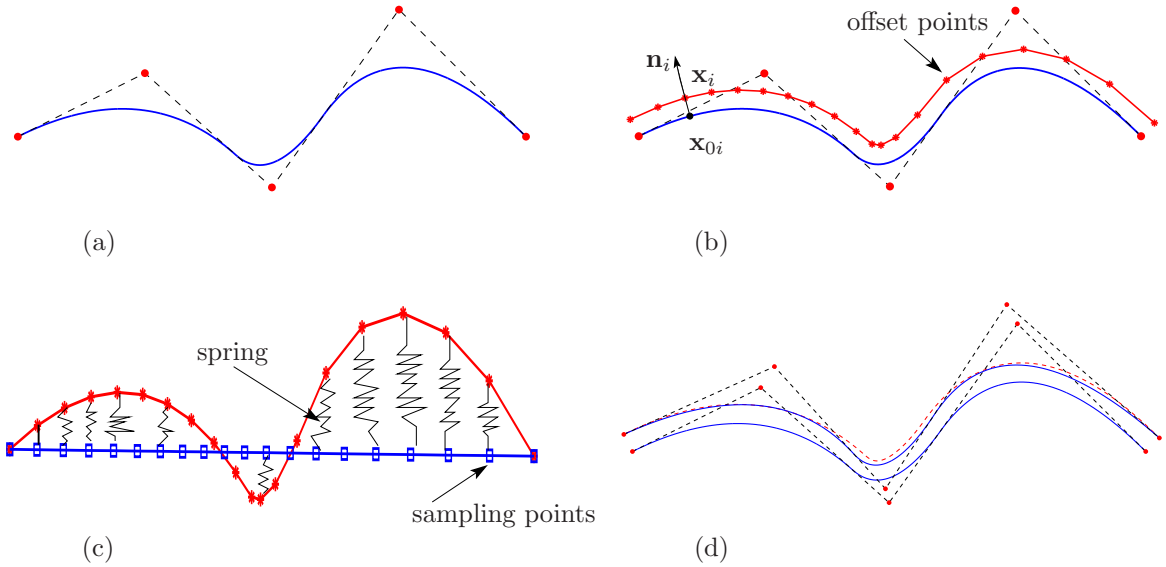


Figure 1: The proposed algorithm to create an offset curve of a given B-spline curve: (a) the original curve, (b) a set of offset points is generated, (c) sampling points on the initial guess offset curve and a system of springs connecting those sampling points and the offset points and (d) the final offset curve is obtained when the energy of the spring is minimum.

For an analytical expression for the energy E is not available, we resort to a numerical differentiation to compute

the energy gradient

$$\nabla E^{(k)} = \frac{1}{h}\begin{bmatrix} \vdots \\ E(\mathbf{P}_I^{(k)} + [h \;\; h]) - E(\mathbf{P}_I^{(k)}) \\ \vdots \end{bmatrix} \tag{14}$$

in which $\nabla E^{(k)}$ is a matrix of dimension $(n-2)\times 2$. In words, to compute the energy gradient along the x-direction at a specific control point, we move it along the x-direction a very small distance h (in our implementation, a value of 10^{-8} was used for h) and compute the change in the energy. Note that when a control point is changed, the sought-for NURBS curve is changed and so are the sampling points $\{\mathbf{x}_i^{(k)}\}$.

Algorithm 1 Curve/surface offset algorithm.

compute the offset points $\{\mathbf{x}_i\}$ (Eq. (11));
compute initial control points $\mathbf{P}^{(0)}$;
$k = 0$, $e = 100$;
while $e \geq \epsilon$ **do**
    compute sampling points on the offset curve;
    compute the energy e using Eq. (12);
    compute the gradient $\nabla E^{(k)}$ using Eq. (14);
    compute the step length $\gamma^{(k)}$;
    update control points $\mathbf{P}^{(k+1)} = \mathbf{P}^{(k)} - \gamma^{(k)}\nabla E^{(k)}$;
    compute sampling points on the offset curve;
    compute the energy e using Eq. (12);
    $k \leftarrow k + 1$
end while

As the first example to test our algorithm, we consider a circle as shown in Fig. (2). We are going to offset it inwards. The circle is exactly represented by a quadratic ($p = 2$) NURBS curve with 9 control points. We use 20 offset points (star points in the right of the referred figure). The initial control points of the offset curve are denoted by green solid circles which are, except the first and last points, the control points of the progenitor circle. The blue squares denote the 20 sampling points on the initial offset curve. For an exact offset of a circle exists, the proposed algorithm converges to a very high accuracy (we used a tolerance of 10^{-4}).

As a second example, let us build the offset of a cubic Bézier curve (having 4 control points) as given in Fig. (3). It is obvious that the error is large (Fig. (3a)) since only 4 control points were used. If a higher accuracy is desired, it is simply to use more control points for defining the offset curve. As shown in Fig. (3)b the offset was sufficiently accurate with 10 control points. This example illustrates that our curve offset algorithm can be well applied for cases in which the requirement of having the same parametrization is relaxed (for CAD applications for example). For those cases, an improvement of the proposed algorithm can be made by using an adaptive optimization scheme. We start with a curve of the same format as its base and perform the optimization given in Algorithm 1. If the error is found to be larger than a predefined tolerance, new control points are added and Algorithm 1 is used again. However we do not further follow this path so as to focus on our analysis target.

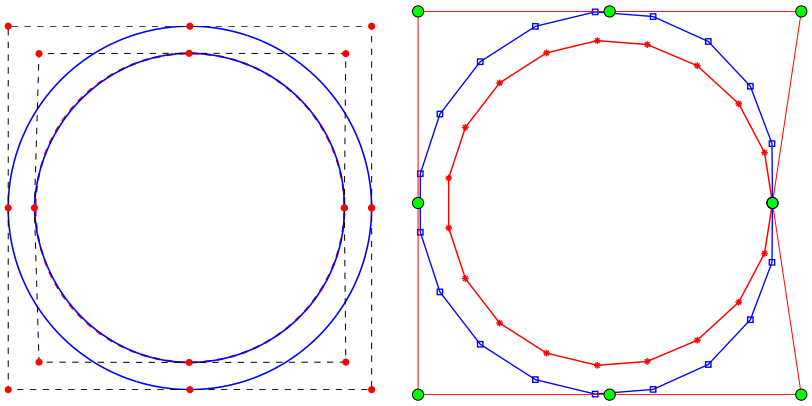
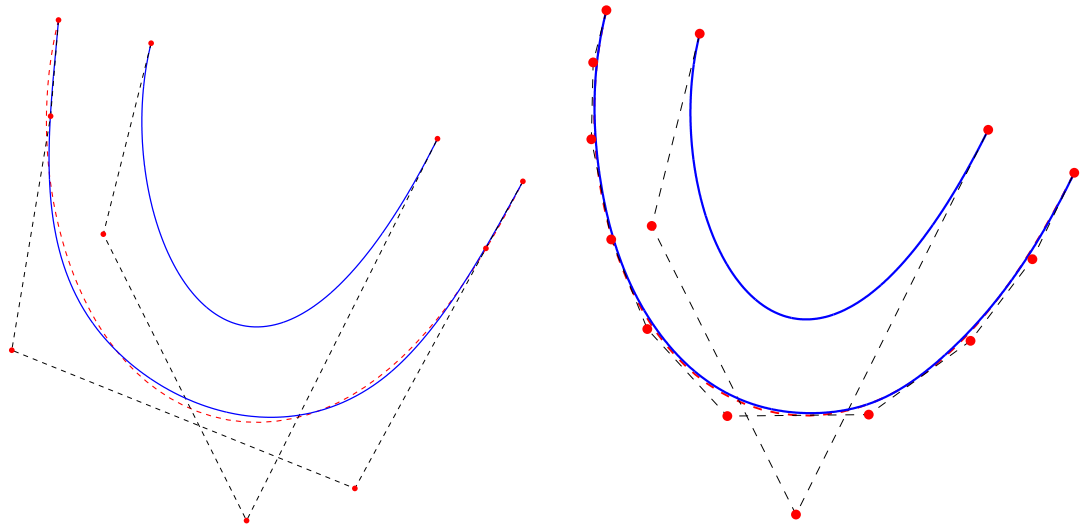


Figure 2: Offset of a circle using the proposed algorithm: 20 sampling points ($m = 20$) are used. Stars denote offset points whereas the green squares are the sampling points on the initial offset curve of which the control points are the green filled circles.



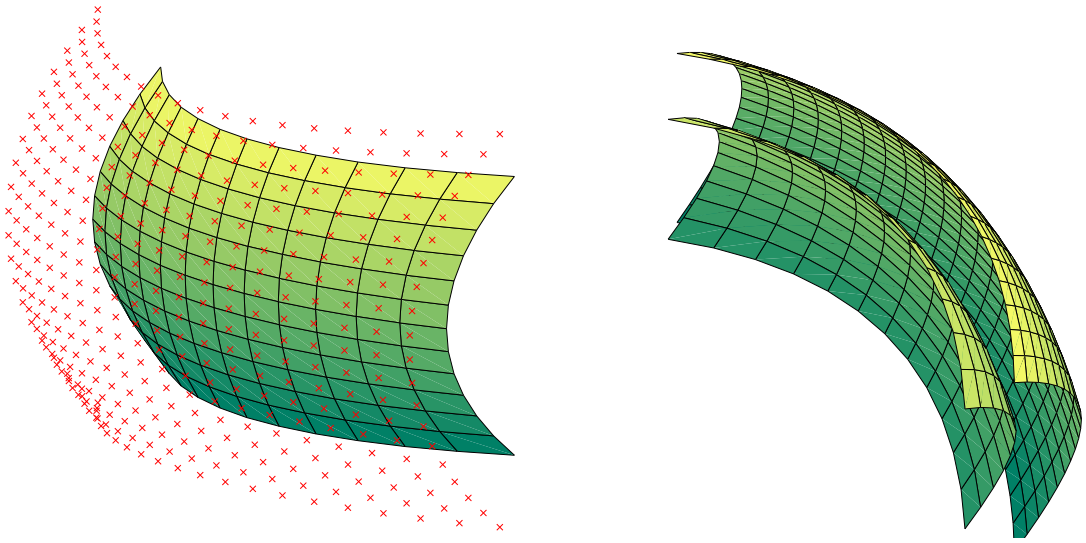
(a) (b)

Figure 3: Cubic Bézier curve example: (a) offset curve has the same number of CPs as the original curve and (b) offset curve has 10 CPs. The dotted lines denote the "exact" offset curve that joins the offset points.

2.3. NURBS surface offset

The proposed algorithm for curves can be straightforwardly extended to B-splines surfaces. We refer to Fig. (4) for such an application of offsetting a bi-quadratic Bézier surface. The offset surface has the same parametrization as its base. The method converged in 4 iterations for a tolerance $\epsilon = 0.1$. In this context, it should be emphasized that the gradient descent algorithm converges linearly (and yet requires only first derivatives of the objective function E).

Therefore, more involved Newton-like methods should be adopted if real-time generation of offset surfaces is required. It should be emphasized that our aim was not a real-time curve/surface offsetting algorithm.



(a) (b)

Figure 4: Generation of offset surface: (a) original surface and a grid of 20×20 offset points and (b) the resulting offset surface.

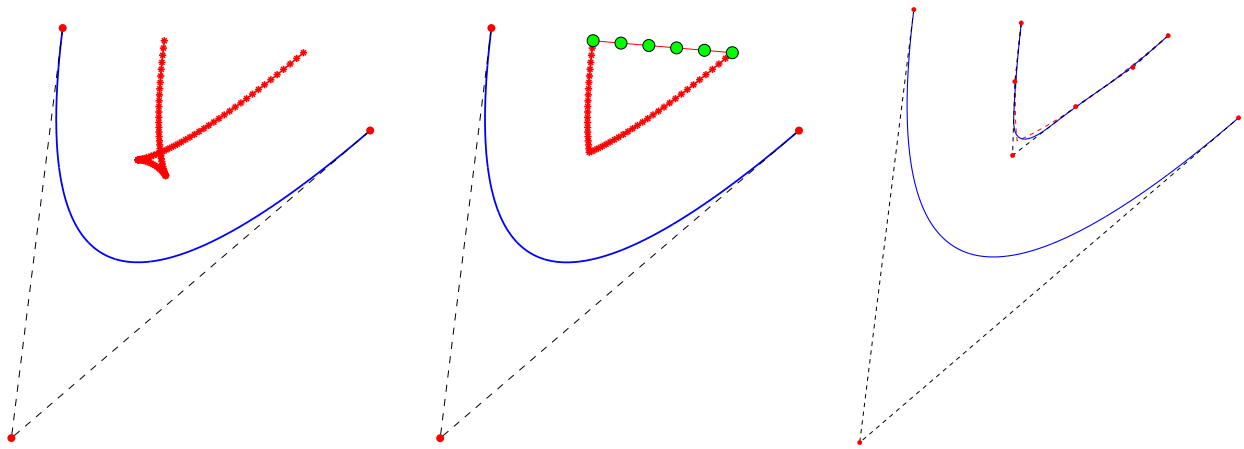
2.4. Non self-intersecting offset curves

When the instantaneous radius of curvature of the curve is less than the offset distance there will be self intersection of the offset curve, see Fig. (5a). After removing the offset points in trouble, one obtains the "exact" offset curve as shown in Fig. (5b). In order to get a good approximation to the offset curve, we used a cubic Bézier with 6 control points and the result is presented in Fig. (5c). As can be seen, the proposed offset algorithm was able to build non self-intersecting offset curves. We, however, do not go further for this special case keeping in mind that our target–composite panels do not have high curvatures so that self intersection would occur.

2.5. Solid NURBS construction from boundary representations of a composite panel

In a CAD environment, a curved 3D composite panel is usually represented by a NURBS surface. This is often not sufficient from an analysis point of view if a detailed modeling of the physical behavior through the thickness of the panel is to be performed. In this case, a trivariate solid representation of the panel surface is needed. In this section we are going to show that the offset algorithm described in Section 2 can be used for this purpose.

For ease of demonstration, let us first consider the case of generating a bivariate surface from a NURBS curve as shown in Fig. (6). In the left of the referred figure, a given B-spline curve and its offset that has the same parametrization are given. Thanks to the same format of these curves, a tensor-product surface bounded by these curves can be easily obtained. The knot vector along the offset distance is given simply as $\mathscr{H} = \{0, 0, 1, 1\}$. The control points of the surface are the control points of the base curve and its offset. The right of the figure shows a refined model which is suitable for FE calculations. The idea is straightforwardly extended to NURBS surfaces as illustrated in Fig. (7).



(a) (b) (c)

Figure 5: A cubic Bézier curve with radius of curvature less than the offset distance : (a) offset points with local intersection (b) after removing and (c) offset curve.

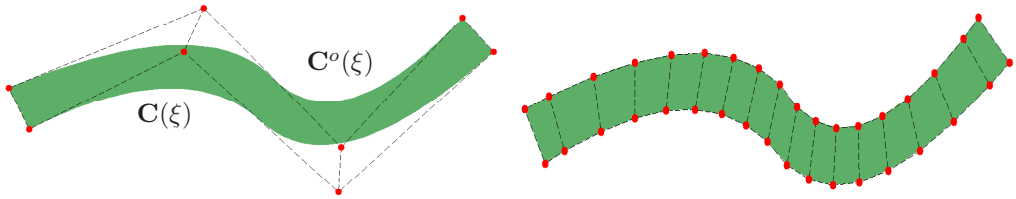


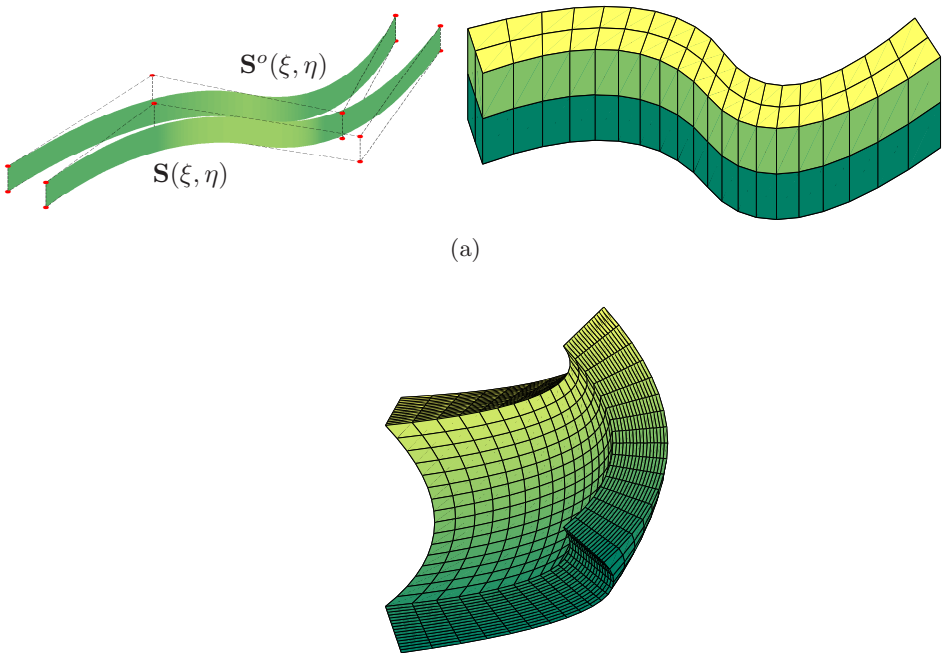
Figure 6: Generation of surface using offset curves (left) and refined model suitable for a FE analysis (right).

As a 3D example with more complex geometry, we consider a singly curved thick-walled laminate which was studied in [33]. Air-intakes of formula race cars and strongly curved regions of ship hulls provide examples for such thick-walled curved laminates designs. The geometry of the sample is given in Fig. (8). Since the geometry representation of the object of interest is the same in both CAD and FEA environment, it is very straightforward and fast to get an analysis-suitable model when changes are made to the CAD model, for instance changing the thickness t. This is in sharp contrary to Lagrange finite elements which uses a different geometry representation.

The geometry of the singly curved thick-walled laminates can be built by first creating a NURBS curve as shown in Fig. (9). Next, an offset of this curve with offset distance t is created. Having these two curves, a NURBS surface can be constructed. Finally, the cross section is extruded along the width direction. The corresponding NURBS meshes to be readily used in a FEM environment are given in Fig. (10). FE analysis of this structure can be found in Ref.[23].

Remark 2.1. The computational aspects of our algorithm are listed as follows

- How many sampling points are enough;
- How to distribute those points?;
- How the initial guess of the curve effects the result?



(a)

(b)

Figure 7: Generation of solids: (a) extruded surface and (b) free form surface.

About the first question the algorithm presented in Ref. [26] can be used. As far as the second question is concerned, it all depends on the curvature of the base curve. Usually more sampling points should be placed in the place where the curvature is high. However, we simply used a uniform distribution because in our application, the curve is quite smooth. From our experiences, the initial guess of the offset curve has a slight impact on the convergence rate.

3. Isogeometric analysis

The isogeometric concept refers to the utilization of the basis functions used to represent the geometry (usually they are NURBS but other CAD technologies such as subdivision surfaces can be equally employed) to approximate the field variables. This concept can be used in a finite element context, a boundary element context [34, 11] or even in a meshfree framework. In this paper, we use an isogeometric finite element method (IGAFEM) which is suitable for nonlinear analyses. In what follows, we briefly present the main ideas of IGAFEM for two dimensional linear elasticity problems, extension to three dimensional case is straightforward. We refer to [6, 7] and [12] for details.

Consider a domain Ω, bounded by Γ. The boundary is partitioned into two sets: Γ_u and Γ_t with displacements prescribed on Γ_u and tractions $\overline{\mathbf{t}}$ prescribed on Γ_t: $\Gamma = \overline{\Gamma_t \cup \Gamma_u}$, $\Gamma_t \cap \Gamma_u = \emptyset$. The weak form of a linear elastostatics

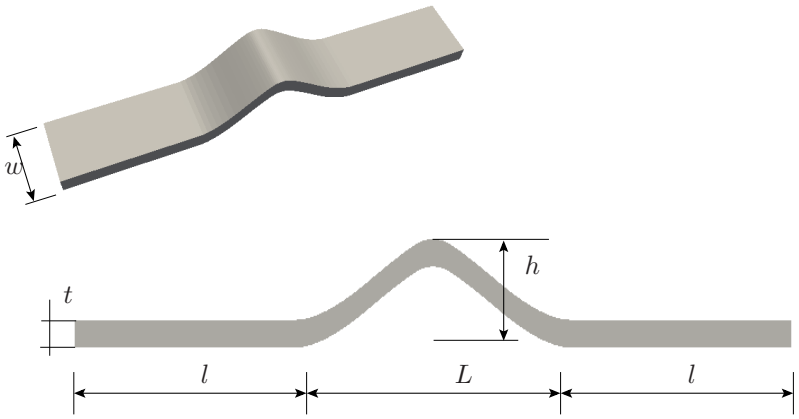


Figure 8: Singly curved thick-walled laminates: geometry configuration. The thickness t is constant. Dimensions can be found in Ref.[23].

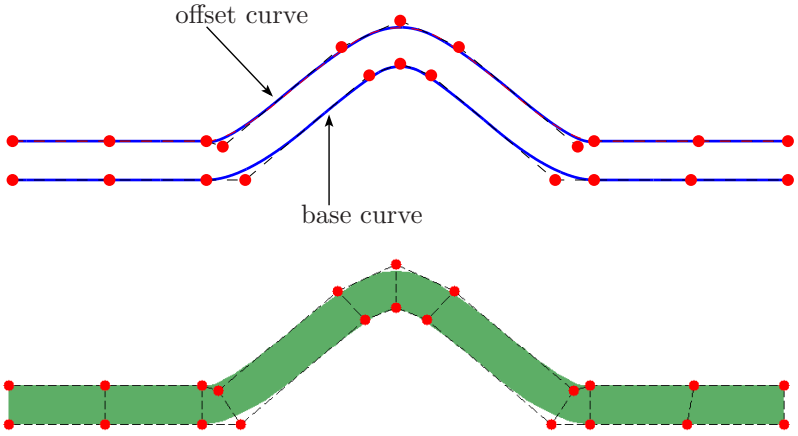


Figure 9: Singly curved thick-walled laminates: building the cross section as a B-spline surface made of the base curve and its offset. The red points denote the control points. It should be emphasized that the offset curve cannot be built simply using an extrusion operation on the base curve.

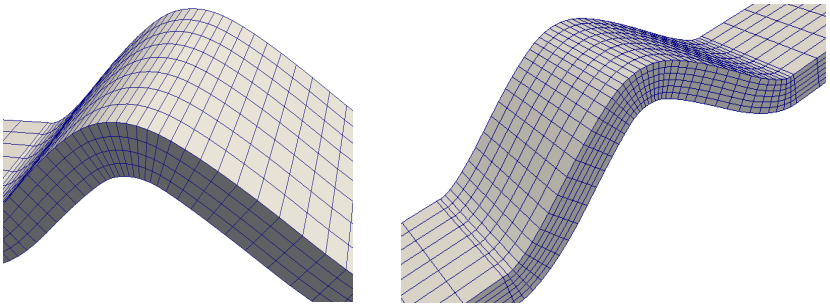


Figure 10: Singly curved thick-walled laminates: NURBS meshes.

problem is to find the displacement field $\mathbf{u}$ in the trial space [6], such that for all test functions $\delta\mathbf{u}$ in the test space [7],

$$\int_\Omega \boldsymbol{\varepsilon}(\mathbf{u}) : \mathbf{D} : \boldsymbol{\varepsilon}(\delta\mathbf{u})\mathrm{d}\Omega = \int_{\Gamma_t} \overline{\mathbf{t}} \cdot \delta\mathbf{u}\mathrm{d}\Gamma + \int_\Omega \mathbf{b} \cdot \delta\mathbf{u}\mathrm{d}\Omega \tag{15}$$

where the elasticity matrix is denoted by $\mathbf{D}$, $\mathbf{b}$ refers to a body force, $\boldsymbol{\varepsilon}$ represents the strain field which is taken as the symmetric part of the displacement gradient $\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla\mathbf{u} + \nabla^{\mathrm{T}}\mathbf{u})$. Using the Bubnov-Galerkin method where the same shape functions R_I–the NURBS basis functions– are used for $\mathbf{u}$ and $\delta\mathbf{u}$ we can write

$$\mathbf{u}(\mathbf{x}) = \sum_I^{nn} R_I(\boldsymbol{\xi})\mathbf{u}_I, \quad \delta\mathbf{u}(\mathbf{x}) = \sum_I^{nn} R_I(\boldsymbol{\xi})\delta\mathbf{u}_I \tag{16}$$

where $\mathbf{u}_I = [u_{xI}, u_{yI}]^{\mathrm{T}}$ is the nodal unknown vector, $\delta\mathbf{u}_I$ denote the nodal displacement variations and nn is the number of control points. Note the similarity of the above equations with Eq. (8). This is the well known isoparametric concept in FEM.

Substitution of these approximations into Eq. (15) and using the arbitrariness of the nodal displacement variations $\delta\mathbf{u}_I$ gives the discrete equations

$$\mathbf{K}\ \mathbf{u} = \mathbf{f} \tag{17}$$

with

$$\mathbf{K}_{IJ} = \int_\Omega \mathbf{B}_I^{\mathrm{T}}\mathbf{D}\mathbf{B}_J\mathrm{d}\Omega, \quad \mathbf{f}_I = \int_{\Gamma_t} R_I\overline{\mathbf{t}}\mathrm{d}\Gamma + \int_\Omega R_I\mathbf{b}\mathrm{d}\Omega \tag{18}$$

In two dimensions, the strain-displacement $\mathbf{B}_I$ matrix is given by

$$\mathbf{B}_I = \begin{bmatrix} R_{I,x} & 0 \\ 0 & R_{I,y} \\ R_{I,y} & R_{I,x} \end{bmatrix} \tag{19}$$

Solving the system of linear equations in Eq. (17) gives the nodal values $\mathbf{u}$ based on which derived quantities such

[6] contains C^0 functions
[7] contains C^0 functions but vanishes on Γ_u

as strains and stresses can be determined at any point. For details on FEM, we refer to standard textbooks on the subject e.g., [35, 36]. Note that $R_{I,x}$ denotes the first order partial derivative of R_I with respect to x.

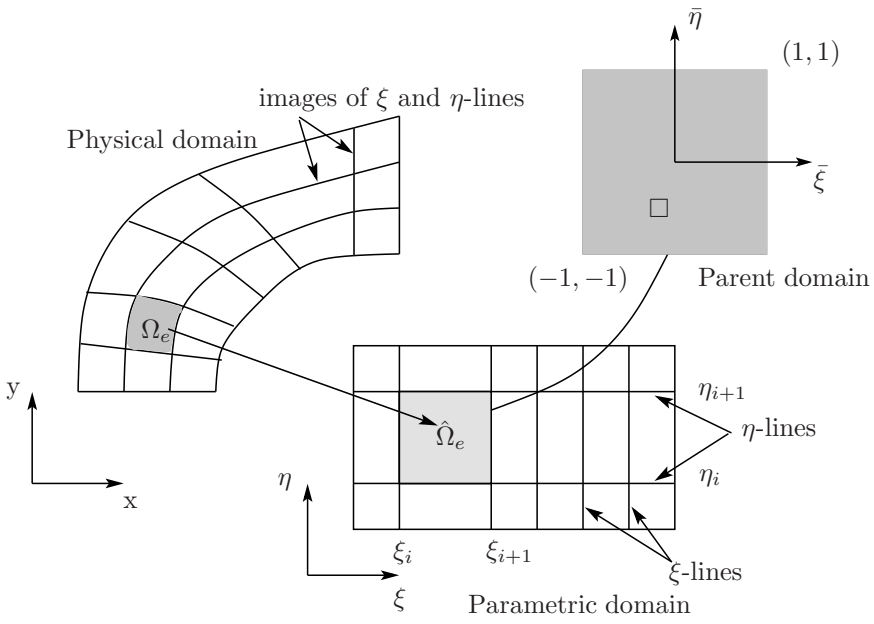


Figure 11: Definition of domains used for integration in isogeometric analysis.

Domain integrals as appeared in Eq. (18) are numerically evaluated as in standard FEM. However, there is one difference because the NURBS basis are defined in the parameter space or parametric domain whereas the quadrature rule are defined in the so-called bi-unit parent domain, see Fig. (11). For a function $f(x, y)$, one writes

$$\int_{\Omega} f(x,y)\mathrm{d}\Omega = \bigcup_{e=1}^{nel} \int_{\Omega_e} f(x,y)\mathrm{d}\Omega_e = \bigcup_{e=1}^{nel} \int_{\hat{\Omega}_e} f(x(\xi),y(\eta))|J_{\boldsymbol{\xi}}|\mathrm{d}\hat{\Omega}_e = \bigcup_{e=1}^{nel} \int_{\square} f(\bar{\xi},\bar{\eta})|J_{\boldsymbol{\xi}}||J_{\bar{\boldsymbol{\xi}}}|\mathrm{d}\square \tag{20}$$

where $\bigcup$ represents the standard assembly operator, *nel* denotes the number of finite elements which are defined as non-zero knot spans, $|J_{\boldsymbol{\xi}}|$ denotes the determinant of the Jacobian of the transformation from the parametric domain to the physical domain and $|J_{\bar{\boldsymbol{\xi}}}|$ represents the determinant of the Jacobian of the transformation from the parent domain to the parametric domain. This transformation, which defines a map from a square to a rectangle, is trivial and hence not discussed here. The final integral can be performed using standard Gauss-Legendre quadrature. Specially, a $(p+1)\times(q+1)$ Gaussian quadrature is adopted for two dimensional elements with p and q denoting the orders of the NURBS basis in the ξ and η directions, respectively.

There exists some open-source IGA packages written in Matlab, for example GeoPDEs reported in [37], ISOGAT [38] and MIGFEM, written by the first author, hosted at `https://sourceforge.net/projects/cmcodes/` and described in [12]. Our offset algorithm is implemented in MIGFEM.

4. Examples

In this section we present a stress analysis of a curved composite panel with stiffeners as given in Fig. (12). The panel is made from an orthotropic elastic material with material constants E_{11} and E_{22} being the Youngs moduli of the ply

in fiber direction and transverse direction, respectively, ν_{21} and ν_{23} are the longitudinal and transverse Poissons ratios, and G_{12} is the longitudinal shear modulus. We used $E_{11} = 115$ GPa, $E_{22} = E_{33} = 8.5$ GPa, $\nu_{12} = \nu_{23} = \nu_{31} = 0.29$, $G_{12} = G_{31} = 4.5$ GPa. Fig. (13) presents the geometry modeling procedure. We start with the skin and stiffener curves. Using the proposed offset algorithm yields the offset curves and from that bivariate B-spline surfaces can be defined. In the next step, extrusion was made to create trivariate B-spline solids which are analysis-suitable. Finally, the correct number of plies are built using knot insertion along the thickness direction exploiting the C^{p-m} property of NURBS where m denotes the knot multiplicity. Each ply interface represents a surface where the displacement field should be C^0 so that the strain field is discontinuous. Note that along the thickness direction a linear basis was used, the knots to be inserted can be easily determined. For example, assuming that there are 8 plies of the same thickness, hence the knots to be inserted are $[1/8, 2/8, 3/8, 4/8, 5/8, 6/8, 7/8]$. Note that the original knot vector along the thickness direction was $[0, 0, 1, 1]$. For the purpose of analysis, a more refined model is required which is obtained by adopting the so-called p-refinement (order elevated in CAD terminology) followed by a h-refinement (knot insertion in CAD terminology). This is the so-called k-refinement [6] which is a unique feature of IGA compared to standard FEM.

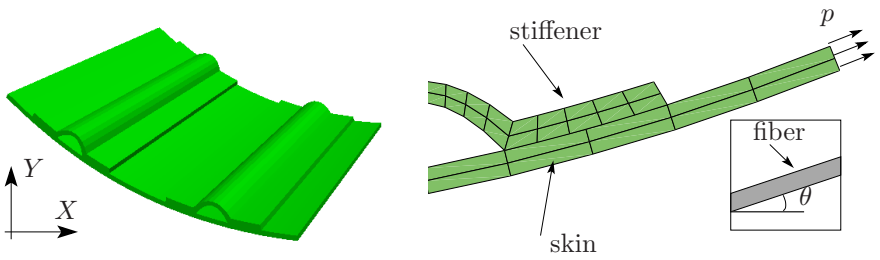
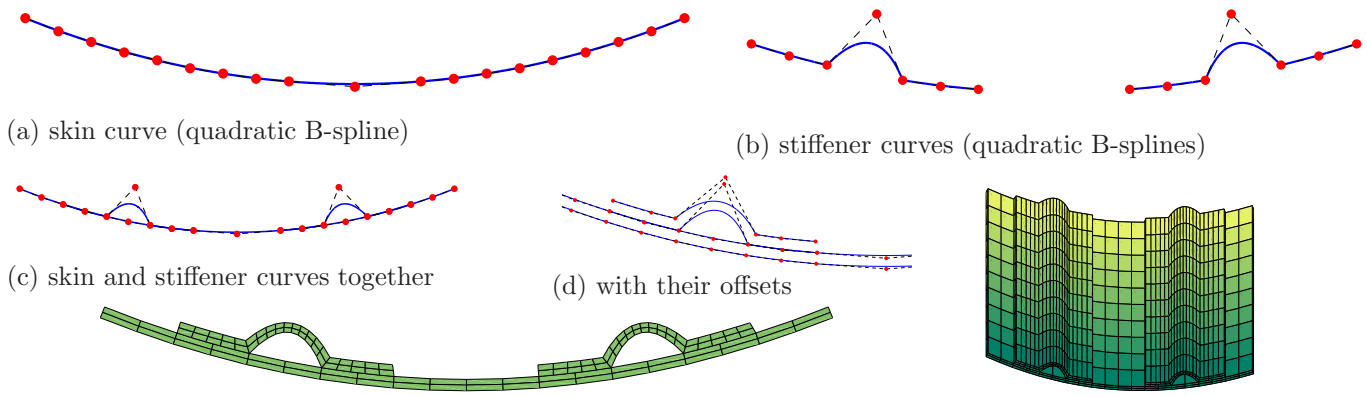


Figure 12: A curved composite panel with two curved stiffeners: the length of the panel is 50 mm and the width is 50 mm. The skin/stiffeners consist of 8 plies (with ply thickness of 0.1 mm) with stacking $[0/90/0/90]_s$. The fiber direction is denoted by θ.



(a) skin curve (quadratic B-spline)

(b) stiffener curves (quadratic B-splines)

(c) skin and stiffener curves together

(d) with their offsets

(e) bivariate B-spline surfaces

(f) trivariate B-spline solid

Figure 13: A curved composite panel with two curve stiffeners: geometry modeling procedure.

For analysis we used a quadratic-quadratic-linear (X direction, Y direction (thickness) and Z direction) NURBS representation. One layer of elements is used for each ply. This NURBS can be directly used in a conforming multi-patch IGA code (we used our open source MIGFEM [12] for this analysis). Figure 14 shows the contour plot of σ_{XX}. Visualization was performed using Paraview [39]. Note that we have restricted to conforming multiple NURBS patch discretization where at the patch interface the control points must match each other. Methods to relax this restriction was recently presented in [40] which provides more flexibility.

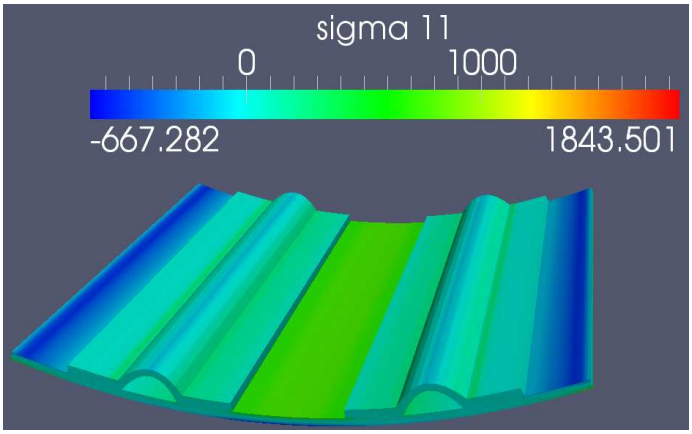


Figure 14: A curved composite panel with two curve stiffeners: stress contour plot.

5. Conclusion

We presented an integrated CAD-FEA framework for designing and modeling three dimensional curved composite panels. To this end, a new algorithm was presented to compute offsets of NURBS curves and surfaces which allows a trivariate representation of NURBS surface to be constructed. The geometry object can be directly used in a FE analysis using the isogeometric analysis concept. Stress analyses of a three dimensional curved composite panel with curved stiffeners demonstrated the effectiveness of the proposed integrated CAD-FEA framework. It facilitates the design of composite panels by allowing (1) CAD data to be directly used in a FEA package, (2) parameter studies in which the number of plies, the ply thickness can be easily varied and (3) an automatic generation of cohesive interface elements for delamination analysis [23]. Preliminary proof-of-concept examples were provided to demonstrate the capabilities of the offset algorithm and isogeometric analysis in the context of design and analysis of composite panels. In depth studies and more complex analyses are under the way. For knowledge sharing, our work will be released as a part in the open source IGA code, MIGFEM hosted at `https://sourceforge.net/projects/cmcodes/`, developed by the same authors.

Acknowledgements

The authors would like to acknowledge the partial financial support of the Framework Programme 7 Initial Training Network Funding under grant number 289361 “Integrating Numerical Simulation and Geometric Design Technology”. Stéphane Bordas also thanks partial funding for his time provided by 1) the EPSRC under grant EP/G042705/1 Increased Reliability for Industrially Relevant Automatic Crack Growth Simulation with the eXtended Finite Element

Method and 2) the European Research Council Starting Independent Research Grant (ERC Stg grant agreement No. 279578) entitled "Towards real time multiscale simulation of cutting in non-linear materials with applications to surgical simulation and computer guided surgery".

References

[1] P. Kagan, A. Fischer, and P. Z. Bar-Yoseph. New B-Spline Finite Element approach for geometrical design and mechanical analysis. *International Journal for Numerical Methods in Engineering*, 41(3):435–458, 1998.

[2] P. Kagan and A. Fischer. Integrated mechanically based CAE system using B-Spline finite elements. *Computer-Aided Design*, 32(8–9):539 – 552, 2000.

[3] F. Cirak, M. Ortiz, and P. Schröder. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering*, 47(12):2039–2072, 2000.

[4] F. Cirak, M. J. Scott, E. K. Antonsson, M. Ortiz, and P. Schoröder. Integrated modeling, finite-element analysis, and engineering design for thin-shell structures using subdivision. *Computer-Aided Design*, 34(2):137 – 148, 2002.

[5] E. Catmull and J. Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design*, 10(6):350 – 355, 1978.

[6] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39-41):4135–4195, 2005.

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

[8] W. A. Wall, M. A. Frenzel, and C. Cyron. Isogeometric structural shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 197(33-40):2976–2988, 2008.

[9] K. Li and X. Qian. Isogeometric analysis and shape optimization via boundary integral. *Computer-Aided Design*, 43(11):1427 – 1437, 2011.

[10] D. Großmann, B. Jüttler, H. Schlusnus, J. Barner, and Anh-Vu Vuong. Isogeometric simulation of turbine blades for aircraft engines. *Computer Aided Geometric Design*, 29(7):519 – 531, 2012.

[11] M.A. Scott, R.N. Simpson, J.A. Evans, S. Lipton, S.P.A. Bordas, T.J.R. Hughes, and T.W. Sederberg. Isogeometric boundary element analysis using unstructured T-splines. *Computer Methods in Applied Mechanics and Engineering*, 254:197 – 221, 2013.

[12] V. P. Nguyen, S.P.A. Bordas, and T. Rabczuk. Isogeometric analysis: An overview and computer implementation aspects. *Computer Aided Geometric Design*, pages –, 2013. submitted.

[13] T. Dokken and V. Skytt. Locally refined splines. In *Proceedings of IV European Conference On Computational Mechanics. Solids, Structures and Coupled Problems in Engineering*, Paris, France, 16-21 May 2010.

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

[15] N. Nguyen-Thanh, H. Nguyen-Xuan, S.P.A. Bordas, and T. Rabczuk. Isogeometric analysis using polynomial splines over hierarchical T-meshes for two-dimensional elastic solids. *Computer Methods in Applied Mechanics and Engineering*, 200(21-22):1892–1908, 2011.

[16] E. Cohen, T. Martin, R.M. Kirby, T. Lyche, and R.F. Riesenfeld. Analysis-aware modeling: Understanding quality considerations in modeling for isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):334–356, 2010.

[17] G. Xu, B. Mourrain, R. Duvigneau, and A. Galligo. Optimal analysis-aware parameterization of computational domain in 3D isogeometric analysis. *Computer-Aided Design*, 45(4):812 – 821, 2013.

[18] D. Burkhart, B. Hamann, and G. Umlauf. Iso-geometric Finite Element Analysis Based on Catmull-Clark : subdivision Solids. *Computer Graphics Forum*, 29(5):1575–1584, 2010.

[19] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J.A. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, and T.W. Sederberg. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):229–263, 2010.

[20] W. Wang, Y. Zhang, L. Liu, and T. J.R. Hughes. Trivariate solid T-spline construction from boundary triangulations with arbitrary genus topology. *Computer-Aided Design*, 45(2):351 – 360, 2013.

[21] T. W. Sederberg, J. Zheng, A. Bakenov, and A. Nasri. T-splines and T-NURCCs. *ACM Transactions on Graphics*, 22:477–484, 2003.

[22] V. P. Nguyen and H. Nguyen-Xuan. High-order B-splines based finite elements for delamination analysis of laminated composites. *Composite Structures*, 102:261–275, 2013.

[23] V. P. Nguyen, P. Kerfriden, and S. Bordas. Isogeometric cohesive elements for two and three dimensional composite delamination analysis. *Composites Science and Technology*, 2013. http://arxiv.org/abs/1305.2738.

[24] D.J. Benson, Y. Bazilevs, M.-C. Hsu, and T.J.R. Hughes. A large deformation, rotation-free, isogeometric shell. *Computer Methods in Applied Mechanics and Engineering*, 200(13-16):1367–1378, 2011.

[25] J. Kiendl, Y. Bazilevs, M.-C. Hsu, R. Wüchner, and K.-U. Bletzinger. The bending strip method for isogeometric analysis of Kirchhoff-Love shell structures comprised of multiple patches. *Computer Methods in Applied Mechanics and Engineering*, 199(37-40):2403–2416, 2010.

[26] L. A. Piegl and W. Tiller. Computing offsets of NURBS curves and surfaces. *Computer-Aided Design*, 31(2):147 – 156, 1999.

[27] G.V.V. Ravi Kumar, K.G. Shastry, and B.G. Prakash. Computing offsets of trimmed NURBS surfaces. *Computer-Aided Design*, 35(5):411 – 420, 2003.

[28] S.-H.F. Chuang and J.-L. Shih. A novel approach for computing C2-continuous offset of NURBS curves. *The International Journal of Advanced Manufacturing Technology*, 29(1-2):151–158, 2006.

[29] J.-L. Shih and S.-H. Frank Chuang. One-sided offset approximation of freeform curves for interference-free NURBS machining. *Computer-Aided Design*, 40(9):931 – 937, 2008.

[30] G. Elber, I. Lee, and M. Kim. Comparing offset curve approximation methods. *IEEE Computer Graphics and Applications*, 17:62–71, 1997.

[31] L. A. Piegl and W. Tiller. *The NURBS Book*. Springer, 1996.

[32] J. Nocedal and S.J. Wright. *Numerical optimization*. New York, NY: Springer Verlag, 1999.

[33] G. Kress, R. Roos, M. Barbezat, C. Dransfeld, and P. Ermanni. Model for interlaminar normal stress in singly curved laminates. *Composite Structures*, 69(4):458 – 469, 2005.

[34] R.N. Simpson, S.P.A. Bordas, J. Trevelyan, and T. Rabczuk. A two-dimensional isogeometric boundary element method for elastostatic analysis. *Computer Methods in Applied Mechanics and Engineering*, 209–212:87–100, 2012.

[35] T.J.R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, Mineola, NY, 2000.

[36] J. Fish and T. Belytschko. *A First Course in Finite Elements*. Wiley, 2007.

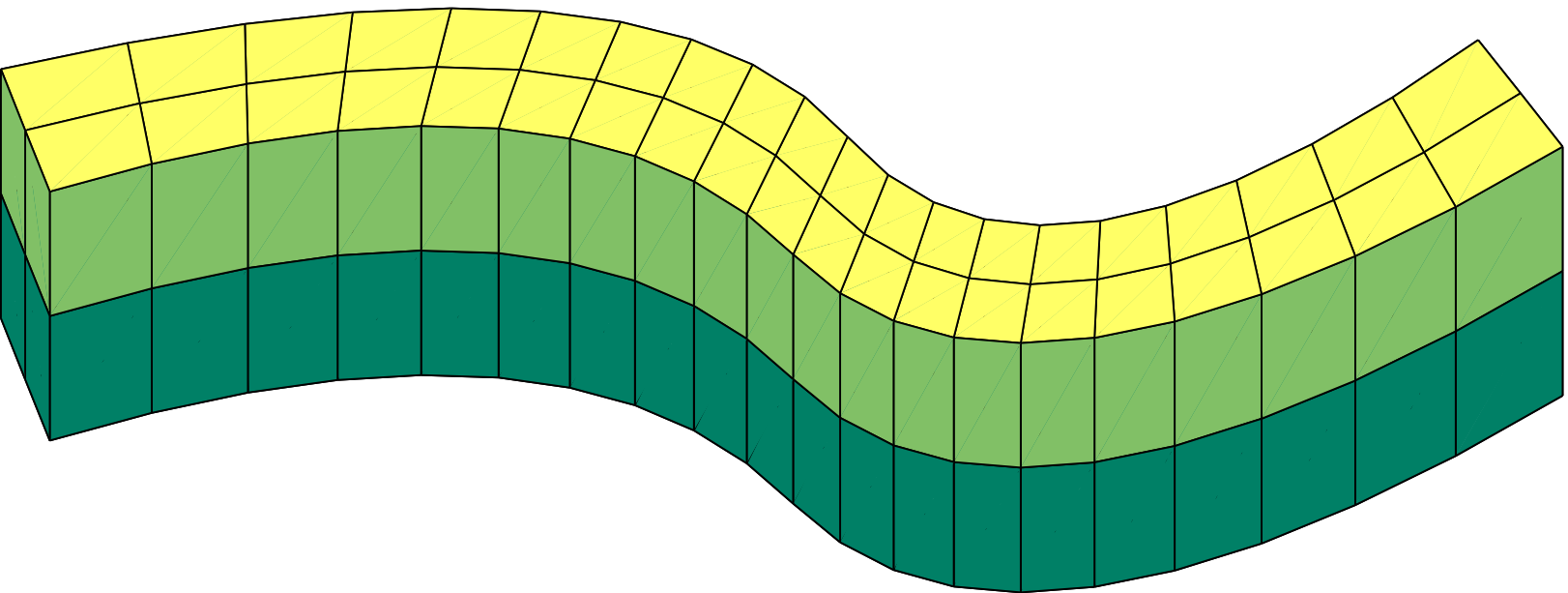
[37] C. de Falco, A. Reali, and R. Vázquez. GeoPDEs: a research tool for Isogeometric Analysis of PDEs. *Advances in Engineering Software*, 42(12):1020–1034, 2011.

[38] A.-V. Vuong, Ch. Heinrich, and B. Simeon. ISOGAT: a 2D tutorial MATLAB code for Isogeometric Analysis. *Computer Aided Geometric Design*, 27(8):644–655, 2010.

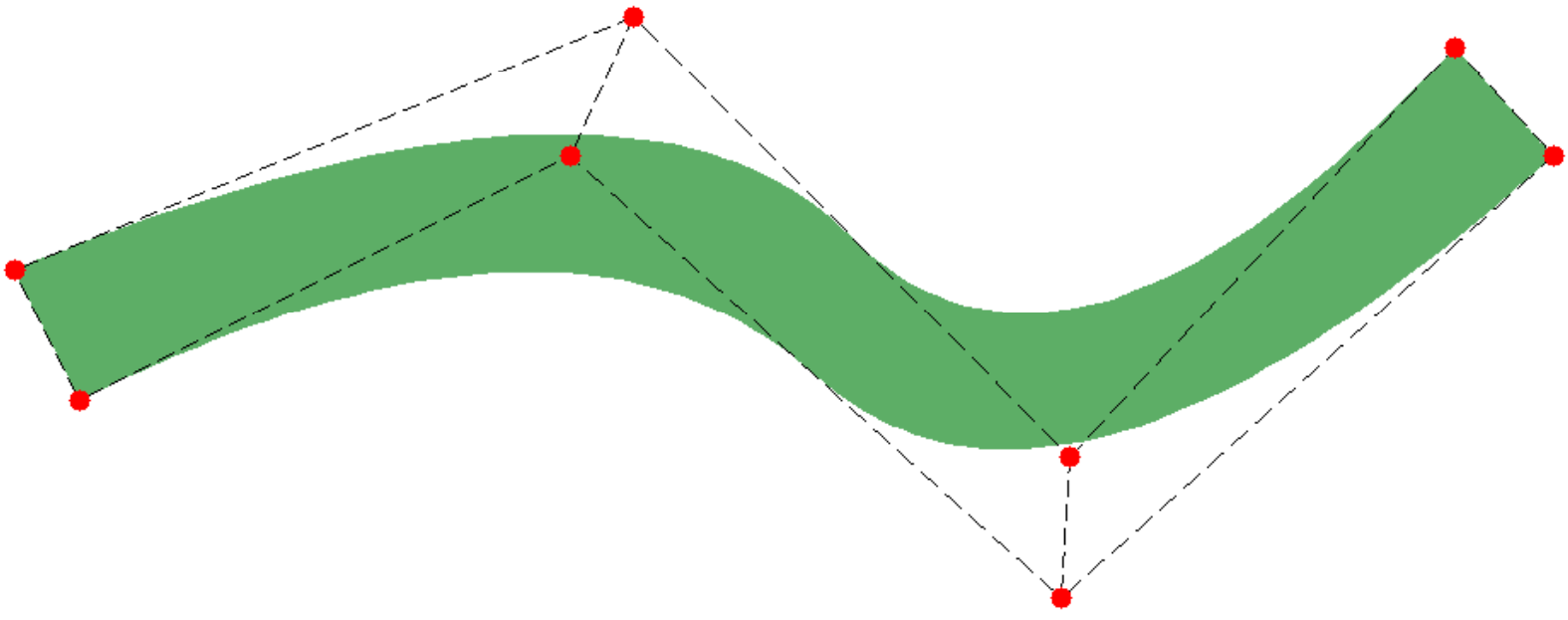
[39] A. Henderson. *ParaView Guide, A Parallel Visualization Application*. Kitware Inc., 2007.

[40] V. P. Nguyen, P. Kerfriden, M. Brino, S. Bordas, and E. Bonisoli. Nitsche's method for two and three dimensional NURBS patch coupling. *Computational Mechanics*, 2013. http://arxiv.org/abs/1308.0802.

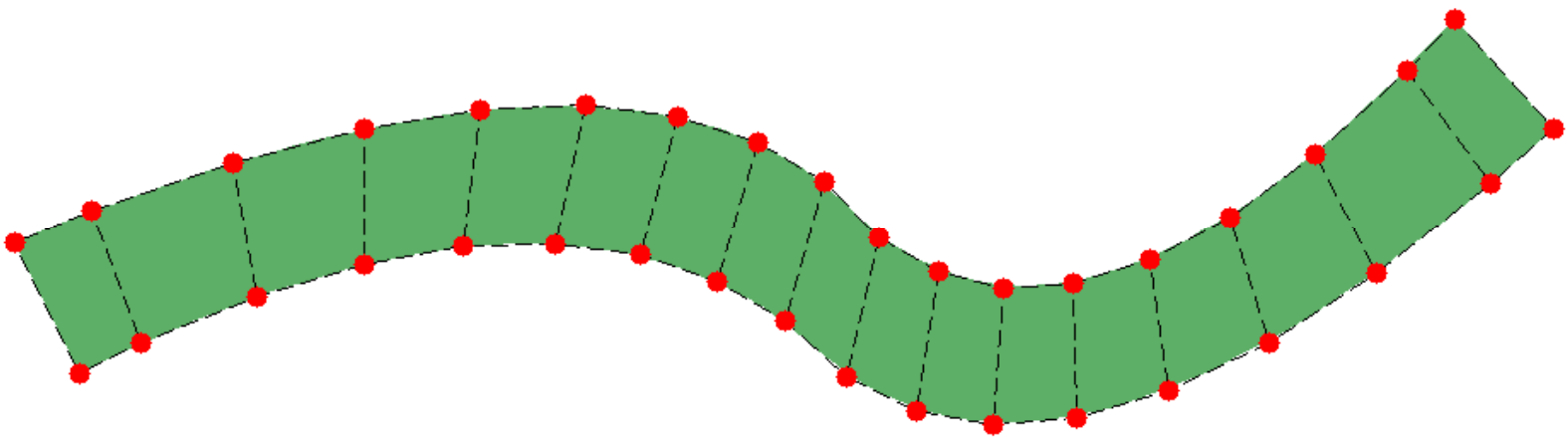
solid.eps



surf1.eps



surf2.eps



surf3.eps