INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 0000; **00**:1–8
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme

A non-iterative method for robustly computing the intersections between a line and a curve or surface

Xiao Xiao[1], Laurent Busé[2] and Fehmi Cirak[1]*

[1] *Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK*
[2] *Université Côte d'Azur, Inria, 2004 route des Lucioles, 06902 Sophia Antipolis, France*

SUMMARY

The need to compute the intersections between a line and a high-order curve or surface arises in a large number of finite element applications. Such intersection problems are easy to formulate but hard to solve robustly. We introduce a non-iterative method for computing intersections by solving a matrix singular value decomposition (SVD) and an eigenvalue problem. That is, all intersection points and their parametric coordinates are determined in one-shot using only standard linear algebra techniques available in most software libraries. As a result, the introduced technique is far more robust than the widely used Newton-Raphson iteration or its variants. The maximum size of the considered matrices depends on the polynomial degree q of the shape functions and is $2q \times 3q$ for curves and $6q^2 \times 8q^2$ for surfaces. The method has its origin in algebraic geometry and has here been considerably simplified with a view to widely used high-order finite elements. In addition, the method is derived from a purely linear algebra perspective without resorting to algebraic geometry terminology. A complete implementation is available from http://bitbucket.org/nitro-project/. Copyright © 0000 John Wiley & Sons, Ltd.

Received . . .

KEY WORDS: high-order finite elements; curved meshes; interrogation; implicitisation; algebraic geometry

1. INTRODUCTION

There has recently been an increased academic and industrial interest in high-order finite elements due to their efficiency advantages over classical low-order elements, see e.g.[1, 2]. To achieve their full potential high-order methods require the curved domain boundaries to be approximated with non-planar elements. The intersection between the curved elements and a given line is required in a wide range of applications, including contact [3], mesh generation [4, 5] and immersed finite elements [6]. As known, intersection computations lead to an easy to formulate, but hard to solve nonlinear root-finding problem. The prevalent technique in computational mechanics for solving such problems is Newton-Raphson iteration, which is in general not very robust, especially when no good starting points are available or multiple intersections are present.

In computer-aided geometric design (CAD) and manufacturing (CAM) intersection computation is a recurring task and, to this end, a number of ingenious methods have been developed [7]. Especially promising are the non-iterative methods with an origin in algebraic geometry, which are for the most part unknown in computational mechanics. Algebraic geometry deals with systems of polynomial equations and geometric objects defined by them and provides the most rigorous framework for intersection computations [8]. The specific technique considered in this paper is the

*Correspondence to: Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK. E-mail: f.cirak@eng.cam.ac.uk implicitisation technique proposed by Busé [9], which shares some commonalities with the method of moving lines/planes introduced by Sederberg and Chen [10]. Different from the original work, in this paper we derive the method from a purely linear algebra viewpoint. To follow the presented derivations it is sufficient to only know the notion of the algebraic degree of a curve or surface. The algebraic degree of a curve or surface is defined as its number of intersections with a line. Counting all intersections (real, complex, multiple ones and ones at infinity), the algebraic degree of a polynomial curve of degree q and the corresponding tensor-product surface are q and $2q^2$, respectively. As will become clear, adopting a linear algebra viewpoint has the added benefit that many linear algebra techniques, like preconditioning and various matrix decompositions, become readily available for intersection computations.

In the following, we first discuss the intersection between a planar Lagrange curve and a line and provide an easy to follow illustrative example. This simple case is sufficient to introduce and discuss the key aspects of the proposed non-iterative technique. Its extension to the surface case is straightforward and is discussed last.

2. INTERSECTION OF LINES WITH CURVES

2.1. Moving lines and intersections

Let $\boldsymbol{x}(\theta) = \left(x^1(\theta),\, x^2(\theta)\right)^{\mathsf{T}}$ be a planar parametric curve, with $\boldsymbol{x}(\theta) \in \mathbb{R}^2$, of degree q_x given either in Lagrange basis $L_i(\theta)$ or power (monomial) basis $P_j(\theta)$ with

$$\boldsymbol{x}(\theta) = \sum_{i=1}^{q_x+1} L_i(\theta)\boldsymbol{x}_i = \sum_{j=1}^{q_x+1} P_j(\theta)\boldsymbol{\alpha}_j \,, \tag{1}$$

where $\boldsymbol{x}_i \in \mathbb{R}^2$ are the nodal coordinates and $\boldsymbol{\alpha}_j \in \mathbb{R}^2$ are the coefficients in the power basis. As usual, the power basis $P_j(\theta)$ contains the consecutive powers of θ from 0 up to q_x. The two basis are related by

$$P_j(\theta) = \sum_{i=1}^{q_x+1} L_i(\theta) P_j(\theta_i) \,, \tag{2}$$

where $P_j(\theta_i)$ is the Vandermonde matrix and θ_i is the parametric coordinate of the i-th Lagrange node. Following a similar approach a curve given in any other polynomial basis can be re-expressed in the power basis.

To define a point $\boldsymbol{x}(\theta)$ on the curve as the intersection of several moving lines, or pencils of lines, consider the line

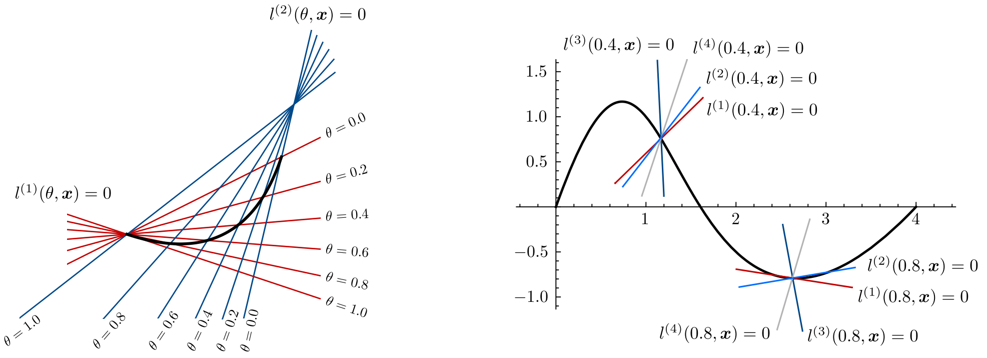
$$l(\theta,\, \boldsymbol{x}) = \begin{pmatrix} \boldsymbol{x} \\ 1 \end{pmatrix} \cdot \boldsymbol{g}(\theta) = x^1 g^1(\theta) + x^2 g^2(\theta) + g^3(\theta) = 0 \,, \tag{3}$$

where $\boldsymbol{x} \in \mathbb{R}^2$ and $\boldsymbol{g}(\theta) = \left(g^1(\theta),\, g^2(\theta),\, g^3(\theta)\right)^{\mathsf{T}}$ is an auxiliary vector collecting the parameters of the line. For a fixed θ equation (3) describes a line and the line moves with the parameter θ, see Figure 1a . The three parameters of the line are assumed to be polynomial functions given by

$$\boldsymbol{g}(\theta) = \sum_{l=1}^{q_g+1} \widetilde{P}_l(\theta)\boldsymbol{g}_l \,. \tag{4}$$

The degree q_g of the power basis $\widetilde{P}(\theta)$ has to be chosen sufficiently high in order to be able to compute all the intersection points (real, complex, multiple ones and ones at infinity). The number of intersection points is equivalent to the algebraic degree of the curve $\boldsymbol{x}(\theta)$. A curve $\boldsymbol{x}(\theta)$ of degree q_x has q_x intersection points with a line. As will become clear, the number of intersections implies a constraint on the minimum possible degree q_g for $\widetilde{P}(\theta)$.

DOI: 10.1002/nme



(a) Two linear moving lines ($q_g = 1$) defining a quadratic curve ($q_x = 2$)

(b) Four of the five cubic moving lines ($q_g = 3$) defining a cubic curve ($q_x = 3$) at $\theta \in \{0.4,\ 0.8\}$

Figure 1. Moving lines and their intersections.

Next, the aim is to find several moving lines of the form (3), or more specifically their coefficients $\boldsymbol{g}_l$ in (4), with a common intersection point which is a point on the curve $\boldsymbol{x}(\theta)$. It is required that each line satisfies at the common intersection point

$$l(\theta,\,\boldsymbol{x}(\theta)) = \begin{pmatrix}\boldsymbol{x}(\theta)\\ 1\end{pmatrix}\cdot\boldsymbol{g}(\theta) = \begin{pmatrix}\boldsymbol{x}(\theta)\\ 1\end{pmatrix}\cdot\left(\sum_{l=1}^{q_g+1}\widetilde{P}_l(\theta)\boldsymbol{g}_l\right) = 0\,. \tag{5}$$

After introducing the definition of the curve (1) this yields

$$\sum_{l=1}^{q_g+1}\left(\sum_{j=1}^{q_x+1}\begin{pmatrix}\boldsymbol{\alpha}_j\\ 1\end{pmatrix}P_j(\theta)\widetilde{P}_l(\theta)\right)\boldsymbol{g}_l = 0\,. \tag{6}$$

The bracketed term can be expressed in a new power basis $\widehat{P}_k$ of dimension $q_x + q_g + 1$ with

$$\sum_{k=1}^{q_x+q_q+1}\widehat{P}_k(\theta)C_{kl} = \sum_{j=1}^{q_x+1}\begin{pmatrix}\boldsymbol{\alpha}_j\\ 1\end{pmatrix}P_j(\theta)\widetilde{P}_l(\theta)\,, \tag{7}$$

where the matrix components C_{kl} contain the known coefficients $\boldsymbol{\alpha}_j$. Equation (6) can now be rewritten as

$$\sum_{l=1}^{3(q_g+1)}\left(\sum_{k=1}^{q_x+q_g+1}\widehat{P}_k(\theta)C_{kl}\right)h_l = 0\,, \tag{8}$$

where the array $\boldsymbol{h}$ contains the components of the yet unknown vectors $\boldsymbol{g}_l$ sorted (by choice) in the following way

$$\boldsymbol{h} = \begin{pmatrix}g_1^1 & g_2^1 & \dots & g_{q_g+1}^1 & g_1^2 & g_2^2 & \dots & g_{q_g+1}^2 & g_1^3 & g_2^3 & \dots & g_{q_g+1}^3\end{pmatrix}^{\mathsf{T}}\,. \tag{9}$$

It is required that (8) is always satisfied irrespective of θ, which is the case for the right null vectors of the matrix C_{kl}. The right null vectors are determined with a SVD, see e.g. [11], yielding the set of null vectors $\boldsymbol{g}_l^{(i)}$, where the index (i) denotes the number of the null vector.

The number of null vectors of C_{kl} depends on the degrees q_x and q_g of the basis $P_j(\theta)$ and $\widetilde{P}_l(\theta)$, and the coefficients $\boldsymbol{\alpha}_j$ of the specific curve considered. For subsequent computations the number of null vectors must be more than the number of intersections of the curve with a line (or its algebraic degree). The non-square matrix C_{kl} has $q_x + q_g + 1$ rows and $3(q_g + 1)$ columns. Hence, its number of right null vectors must be equal or greater than $3(q_g + 1) - (q_x + q_g + 1) = 2q_g - q_x + 2$.[†] In

[†] The number of right null vectors is larger then, for instance, a quadratic curve is described with a cubic polynomial (i.e. $q_x = 3$ but $\alpha_4 = 0$). More precisely, if n is the largest integer such that $\boldsymbol{\alpha}_n \neq 0$ then the number of right null vectors of C_{kl} is exactly $2q_g - n + 3$.

order to obtain the q_x intersections it is necessary to have

$$2q_g - q_x + 2 \geq q_x \quad \Rightarrow \quad q_g \geq q_x - 1\,. \tag{10}$$

The set of null vectors denoted with $\boldsymbol{g}^{(i)}$ introduced in (3) yields a set of moving lines

$$l^{(i)}(\theta,\, \boldsymbol{x}) = \begin{pmatrix} \boldsymbol{x} \\ 1 \end{pmatrix} \cdot \left(\sum_{l=1}^{q_g+1} \widetilde{P}_l(\theta) \boldsymbol{g}_l^{(i)} \right) = 0 \tag{11}$$

with a common intersection point on the curve $\boldsymbol{x}(\theta)$. As an example, in Figure 1a the description of a quadratic curve with $q_x = 2$ by two moving lines $l^{(1)}(\theta,\, \boldsymbol{x})$ and $l^{(2)}(\theta,\, \boldsymbol{x})$ with $q_g = 1$ is shown.

Next, the intersection of a given parametric line

$$\boldsymbol{r}(\xi) = \begin{pmatrix} r^1(\xi) \\ r^2(\xi) \end{pmatrix} = \begin{pmatrix} c_1^1 \\ c_1^2 \end{pmatrix} \xi + \begin{pmatrix} c_0^1 \\ c_0^2 \end{pmatrix} \tag{12}$$

with the curve $\boldsymbol{x}(\theta)$ is considered, where $\boldsymbol{c}_1 = (c_1^1,\, c_1^2)^\mathsf{T}$ and $\boldsymbol{c}_0 = (c_0^1,\, c_0^2)^\mathsf{T}$ are two given vectors. We require that the determined set moving lines $\{l^{(i)}(\theta,\, \boldsymbol{x}) = 0\}$ and $\boldsymbol{r}(\xi)$ intersect at the same point to be yet determined. According to (5) at the common intersection point it is required that

$$l^{(i)}(\theta,\, \boldsymbol{r}(\xi)) = \begin{pmatrix} \boldsymbol{r}(\xi) \\ 1 \end{pmatrix} \cdot \left(\sum_{l=1}^{q_g+1} \widetilde{P}_l(\theta) \boldsymbol{g}_l^{(i)} \right) = 0\,. \tag{13}$$

These equations describe the intersection of each moving line $l^{(i)}(\theta,\, \boldsymbol{x}) = 0$ with the given line $\boldsymbol{r}(\xi)$ and can be rewritten as

$$\sum_{l=1}^{q_g+1} \widetilde{P}_l(\theta) \left[\begin{pmatrix} \boldsymbol{r}(\xi) \\ 1 \end{pmatrix} \cdot \left(\boldsymbol{g}_l^{(i)} \right) \right] = 0\,. \tag{14}$$

As discussed the number of moving lines satisfies $\max(i) \geq q_x$ and all of them can be combined in one homogenous equation system

$$\sum_{l=1}^{q_g+1} \widetilde{P}_l(\theta) \left[A_{li} - \xi B_{li} \right] = 0\,. \tag{15}$$

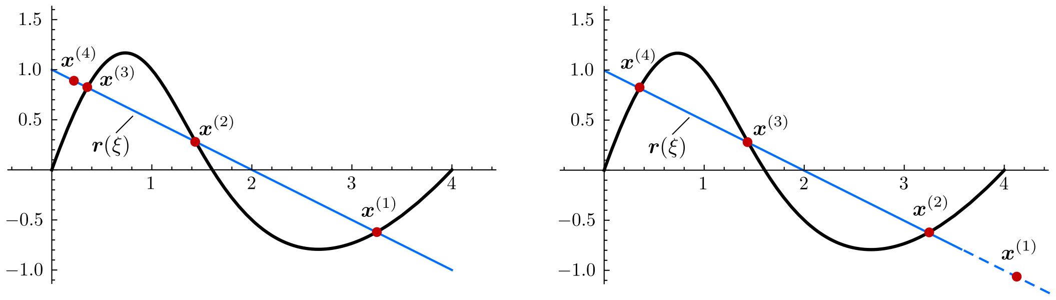
with A_{li} and B_{li} representing the components of two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Both matrices have $q_g + 1$ rows and $2q_g - q_x + 2$ or more columns. To obtain $\widetilde{P}_l(\theta)$ and ξ that satisfy (15) the following generalised eigenvalue problem is considered

$$\boldsymbol{\phi} \left(\boldsymbol{A} - \xi \boldsymbol{B} \right) = \boldsymbol{0}\,. \tag{16}$$

That is, the eigenvalues $\xi^{(j)}$ are the parametric coordinates of the intersection points on the line $\boldsymbol{r}(\xi)$ and the eigenvectors are (up to a multiplicative constant) the basis functions $\widetilde{P}_l(\theta^{(j)})$ evaluated at the intersection points $\theta^{(j)}$. Unfortunately, the matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are not always square and computing the values $\xi^{(j)}$ that satisfy this equation requires non-standard linear algebra techniques. However, as will be discussed in Section 2.2, for the purposes of intersection computation it is sufficient to consider a square eigenvalue problem obtained from (16) by taking only some of its columns. The non-complex eigenvalues of this square eigenvalue problem contain all the intersection points between the given line and the curve. Some of these non-complex eigenvalues may not be actual intersection points, but they can easily be identified.

A non-complex eigenvalue $\xi^{(j)}$ of (16), or its corresponding square eigenvalue problem, gives the potential intersection point $\boldsymbol{r}(\xi^{(j)})$. The respective unknown parameter value $\theta^{(j)}$ on the curve satisfies the equation $\boldsymbol{x}(\theta^{(j)}) = \boldsymbol{r}(\xi^{(j)})$, which is a hard to solve nonlinear problem. According to (15) and (16), however, the left null vector $\boldsymbol{\phi}^{(j)}$ corresponding $\xi^{(j)}$ is proportional to the vector $\widetilde{P}_l(\theta^{(j)})$, which is exploited to determine $\theta^{(j)}$. More precisely, if there is a single parameter

DOI: 10.1002/nme



(a) Intersection points given by (26)

(b) Intersection points given by (27)

Figure 2. Intersection points between a cubic Lagrange curve and a line.

value $\theta^{(j)}$ the ratio of any two consecutive components yields

$$\theta^{(j)} = \frac{\widetilde{P}_{i+1}}{\widetilde{P}_i} = \frac{\phi_{i+1}^{(j)}}{\phi_i^{(j)}} \,. \tag{17}$$

It is assumed here that the monomials in $\widetilde{P}_i$ are sorted in increasing order so that the ratio of two consecutive entries is simply $\theta^{(j)}$. If there are p parameter values $(\theta^{(j,1)}, \ldots, \theta^{(j,p)})$ corresponding to the eigenvalue $\xi^{(j)}$, then the corresponding left null vector space is given by a matrix $\boldsymbol{K}$ with p rows. Keeping the previous assumption on the ordering of the monomials in $\widetilde{P}_i$, we define the $p \times p$ matrices $\boldsymbol{\Delta}_i$ by taking the columns i to $p+i$ of $\boldsymbol{K}$. Then, the parameter values $(\theta^{(j,1)}, \ldots, \theta^{(j,p)})$ are obtained by solving a generalised eigenvalue problem

$$(\boldsymbol{\Delta}_{i+1} - \theta \boldsymbol{\Delta}_i)\boldsymbol{\psi} = \boldsymbol{0}. \tag{18}$$

2.2. *Illustrative example*

The intersection of a cubic Lagrange curve with a line depicted in Figure 2 is considered next. The cubic curve $\boldsymbol{x}(\theta)$ interpolating the points $\boldsymbol{x}_1 = (0,\, 0)^\mathsf{T}$, $\boldsymbol{x}_2 = (1,\, 1)^\mathsf{T}$, $\boldsymbol{x}_3 = (2,\, -0.5)^\mathsf{T}$ and $\boldsymbol{x}_4 = (4,\, 0)^\mathsf{T}$ is expressed in power basis with

$$\boldsymbol{x}(\theta) = \sum_{j=1}^{4} P_j(\theta)\boldsymbol{\alpha}_j = \begin{pmatrix} 0 & 4 & -4.5 & 4.5 \\ 0 & 11.25 & -31.5 & 20.25 \end{pmatrix} \begin{pmatrix} P_1(\theta) \\ P_2(\theta) \\ P_3(\theta) \\ P_4(\theta) \end{pmatrix} \,. \tag{19}$$

The intersections between the given line

$$\boldsymbol{r}(\xi) = \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \xi \begin{pmatrix} 4 \\ -2 \end{pmatrix} = \begin{pmatrix} 4\xi \\ 1 - 2\xi \end{pmatrix} \tag{20}$$

and the curve $\boldsymbol{x}(\theta)$ are sought.

As discussed, the degree q_g of the auxiliary polynomial needs to satisfy $q_g \geq q_x - 1 = 2$. In this example choosing $q_g = 2$ yields matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ that have the dimensions 3×3 and it is straightforward to compute the eigenvalues ξ of the generalised eigenvalue problem (16). To illustrate the more challenging case with non-square matrices $\boldsymbol{A}$ and $\boldsymbol{B}$, which turns out to be inevitable in the case of surfaces, we choose here $q_g = 3$ such that

$$\boldsymbol{g}(\theta) = \sum_{l=1}^{4} \widetilde{P}_l(\theta)\boldsymbol{g}_l \,. \tag{21}$$

The right null vectors $\boldsymbol{g}_l^{(i)}$ that define the moving lines are obtained from (8) with $\boldsymbol{C}$ and the components

$$\boldsymbol{C} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 4 & 0 & 0 & 0 & 11.25 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -4.5 & 4 & 0 & 0 & -31.5 & 11.25 & 0 & 0 & 0 & 0 & 1 & 0 \\ 4.5 & -4.5 & 4 & 0 & 20.25 & -31.5 & 11.25 & 0 & 0 & 0 & 0 & 1 \\ 0 & 4.5 & -4.5 & 4 & 0 & 20.25 & -31.5 & 11.25 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4.5 & -4.5 & 0 & 0 & 20.25 & -31.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4.5 & 0 & 0 & 0 & 20.25 & 0 & 0 & 0 & 0 \end{pmatrix} . \tag{22}$$

This matrix has the dimensions 7×12, its rank is seven and the dimension of its right null space is five. Each of the corresponding five independent null vectors $\boldsymbol{g}_l^{(i)}$ defines one moving line. Figure 1b depicts the first four moving lines $l^{(1)}(\theta, \boldsymbol{x}) = 0$, $l^{(2)}(\theta, \boldsymbol{x}) = 0$, $l^{(3)}(\theta, \boldsymbol{x}) = 0$ and $l^{(4)}(\theta, \boldsymbol{x}) = 0$ at the parameter values $\theta = 0.4$ and $\theta = 0.8$.

Substituting the line (20) as in (15) gives the two non-square matrices (with four significant digits)

$$\boldsymbol{A} = \begin{pmatrix} 0.08710 & 0.08740 & 0.04298 & 0.05767 & 0.01708 \\ -0.2222 & -0.02814 & 0.8999 & -0.01814 & 0.03355 \\ 0.04655 & -0.07068 & -0.03786 & 0.9989 & 0.01874 \\ -0.03786 & 0.1289 & -0.01752 & -0.001102 & 1.010 \end{pmatrix} \tag{23}$$

and

$$\boldsymbol{B} = \begin{pmatrix} 1.046 & 1.101 & 1.499 & 0.7322 & 0.2201 \\ -3.351 & -1.347 & -0.1816 & 0.04373 & 0.1881 \\ 2.390 & -2.402 & -0.1053 & 0.2525 & 0.3777 \\ -0.1302 & 2.987 & -0.3424 & 0.1006 & 0.3987 \end{pmatrix} . \tag{24}$$

One approach to obtaining the generalised eigenvalues of matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ is to use pencil reduction, see [12, 13, 14], which is not a widely used linear algebra operation and may introduce additional numerical issues because of several numerical rank estimations. Alternatively, the eigenvalue problems defined by square submatrices $\boldsymbol{A}^{\square}$ and $\boldsymbol{B}^{\square}$ of the largest size, e.g. the first four columns of $\boldsymbol{A}$ and $\boldsymbol{B}$, can be considered

$$\boldsymbol{\phi}^{\square} \left(\boldsymbol{A}^{\square} - \xi \boldsymbol{B}^{\square} \right) = \boldsymbol{0} . \tag{25}$$

Although there can only be three intersection points for a cubic curve this problem has four eigenvalues and eigenvectors. Notice that each column i of $\boldsymbol{A} - \xi \boldsymbol{B}$, i.e $\sum_l \phi_l (A_{li} - \xi B_{li}) = 0$, represents the intersection between the line $\boldsymbol{r}(\xi)$ and the moving line $l^{(i)}(\theta, \boldsymbol{x}) = 0$. Taking a different couple of submatrices $\boldsymbol{A}^{\square}$ and $\boldsymbol{B}^{\square}$ of largest size, like the last four columns of $\boldsymbol{A}$ and $\boldsymbol{B}$, yields a different set of intersection points. Both sets of intersection points contain the three true intersection points in addition to one fictitious intersection point.

To demonstrate this observation, the intersection points with two different pairs of largest square submatrices $\boldsymbol{A}^{\square}$ and $\boldsymbol{B}^{\square}$ are computed. The eigenvalues for the problem defined by the first four columns of $\boldsymbol{A}$ and $\boldsymbol{B}$ are

$$\xi^{(1)} = 0.8112 , \quad \xi^{(2)} = 0.3594 , \quad \xi^{(3)} = 0.08875 , \quad \xi^{(4)} = 0.05326 , \tag{26}$$

and the eigenvalues for the problem defined by the last four columns $\boldsymbol{A}$ and $\boldsymbol{B}$ are

$$\xi^{(1)} = 28.05 , \quad \xi^{(2)} = 0.8112 , \quad \xi^{(3)} = 0.3594 , \quad \xi^{(4)} = 0.08875 . \tag{27}$$

It is evident that $\xi^{(4)}$ for the first problem and $\xi^{(1)}$ for the second problem correspond to fictitious intersection points while the other three eigenvalues correspond to the true intersection points, see Figure 2. The coordinates of the three true intersection points are computed by introducing $\xi^{(j)}$ in the line equation (20) yielding

$$\boldsymbol{x}^{(1)} = (3.245,\, -0.6225)^{\mathsf{T}}\,, \quad \boldsymbol{x}^{(2)} = (1.438,\, 0.2813)^{\mathsf{T}}\,, \quad \boldsymbol{x}^{(3)} = (0.3550,\, 0.8225)^{\mathsf{T}}\,. \tag{28}$$

The fictitious points can also be detected without computing several eigenvalue problems and comparing their eigenvalues, that is different from above only one single eigenvalue problem is computed. This is accomplished by determining the parametric coordinates of the intersection points $\theta^{(j)}$ on the curve $\boldsymbol{x}(\theta)$. As indicated in (17) the parametric coordinates are computed using the eigenvectors. As an example, consider the eigenvalue problem (25) defined by first four columns of $\boldsymbol{A}$ and $\boldsymbol{B}$. Its eigenvalues are given on (26) and the coordinates of the intersection points are

$$\begin{aligned} &\boldsymbol{x}^{(1)} = (3.245,\, -0.6225)^{\mathsf{T}}, \quad \boldsymbol{x}^{(2)} = (1.438,\, 0.2813)^{\mathsf{T}}, \quad \boldsymbol{x}^{(3)} = (0.3550,\, 0.8225)^{\mathsf{T}}, \\ &\boldsymbol{x}^{(4)} = (0.2130,\, 0.8935)^{\mathsf{T}}\,. \end{aligned} \tag{29}$$

The respective parameters $\theta^{(j)}$ of the intersection points are

$$\theta^{(1)} = 0.9014\,, \quad \theta^{(2)} = 0.5\,, \quad \theta^{(3)} = 0.09861\,, \quad \theta^{(4)} = 0.03932\,. \tag{30}$$

A point is a true intersection point if and only if $\boldsymbol{x}(\theta^{(j)}) = \boldsymbol{x}^{(j)}$; otherwise, it is a fictitious point. It can easily be found $\boldsymbol{x}(\theta^{(4)}) = (0.1506,\, 0.3949) \neq \boldsymbol{x}^{(4)}$ such that the fourth point is not an intersection point. Thus, in practice the intersection points are computed from a single couple of square matrices $\boldsymbol{A}^{\square}$ and $\boldsymbol{B}^{\square}$.

3. INTERSECTION OF LINES WITH SURFACES

The extension of the introduced method to surfaces is straightforward. Let $\boldsymbol{x}(\boldsymbol{\theta}) = (x^1(\boldsymbol{\theta}),\, x^2(\boldsymbol{\theta}),\, x^3(\boldsymbol{\theta}))^{\mathsf{T}}$ be a parametric surface, with $\boldsymbol{x}(\boldsymbol{\theta}) \in \mathbb{R}^3$, of bi-degree $(q_x^1,\, q_x^2)$ given either in Lagrange basis $L_{\boldsymbol{i}}(\theta^1,\, \theta^2)$ or power basis $P_{\boldsymbol{j}}(\theta^1,\, \theta^2)$ with

$$\boldsymbol{x}(\boldsymbol{\theta}) = \sum_{\boldsymbol{i}} L_{\boldsymbol{i}}(\boldsymbol{\theta})\boldsymbol{x}_{\boldsymbol{i}} = \sum_{\boldsymbol{j}} P_{\boldsymbol{j}}(\boldsymbol{\theta})\boldsymbol{\alpha}_{\boldsymbol{j}}\,, \tag{31}$$

where $\boldsymbol{i} = (i^1,\, i^2)$ and $\boldsymbol{j} = (j^1,\, j^2)$ are multi-indices, $\boldsymbol{\theta} = (\theta^1,\, \theta^2)$ are the parametric surface coordinates, and $\boldsymbol{x}_{\boldsymbol{i}} \in \mathbb{R}^3$ and $\boldsymbol{\alpha}_{\boldsymbol{j}} \in \mathbb{R}^3$ are the coefficients in the two basis. Usually, in finite element applications the degrees q_x^1 and q_x^2 of the surface $\boldsymbol{x}(\boldsymbol{\theta})$ are the same.

In line with the curve case, a point $\boldsymbol{x}(\boldsymbol{\theta})$ on the surface is defined as the intersection of several moving planes of the form

$$l(\boldsymbol{\theta},\, \boldsymbol{x}) = \begin{pmatrix} \boldsymbol{x} \\ 1 \end{pmatrix} \cdot \boldsymbol{g}(\boldsymbol{\theta}) = x^1 g^1(\boldsymbol{\theta}) + x^2 g^2(\boldsymbol{\theta}) + x^3 g^3(\boldsymbol{\theta}) + g^4(\boldsymbol{\theta}) = 0\,, \tag{32}$$

where $\boldsymbol{g}(\boldsymbol{\theta}) = \left(g^1(\boldsymbol{\theta}),\, g^2(\boldsymbol{\theta}),\, g^3(\boldsymbol{\theta}),\, g^4(\boldsymbol{\theta})\right)^{\mathsf{T}}$ is an auxiliary vector collecting the parameters of the plane. Although $l(\boldsymbol{\theta},\, \boldsymbol{x}) = 0$ is now a plane instead of a line, it is still denoted with the same symbol to keep the notation simple. The parameters $\boldsymbol{g}(\boldsymbol{\theta})$ are assumed to be of the following form

$$\boldsymbol{g}(\boldsymbol{\theta}) = \sum_{\boldsymbol{l}} \widetilde{P}_{\boldsymbol{l}}(\boldsymbol{\theta})\boldsymbol{g}_{\boldsymbol{l}}\,. \tag{33}$$

The bi-degree $(q_g^1,\, q_g^2)$ of the power basis $\widetilde{P}(\boldsymbol{\theta})$ has to be sufficiently high to describe all the intersection points of the surface $\boldsymbol{x}(\boldsymbol{\theta})$ with a line.

The planes describing the surface $\boldsymbol{x}(\boldsymbol{\theta})$ have to satisfy

$$l(\boldsymbol{\theta},\, \boldsymbol{x}(\boldsymbol{\theta})) = \begin{pmatrix} \boldsymbol{x}(\boldsymbol{\theta}) \\ 1 \end{pmatrix} \cdot \boldsymbol{g}(\boldsymbol{\theta}) = \left[\sum_{\boldsymbol{j}} P_{\boldsymbol{j}}(\boldsymbol{\theta}) \begin{pmatrix} \boldsymbol{\alpha}_{\boldsymbol{j}} \\ 1 \end{pmatrix}\right] \cdot \left(\sum_{\boldsymbol{l}} \widetilde{P}_{\boldsymbol{l}}(\boldsymbol{\theta})\boldsymbol{g}_{\boldsymbol{l}}\right) = 0\,, \tag{34}$$

DOI: 10.1002/nme

which can be rearranged to

$$\sum_{\boldsymbol{l}} \sum_{\boldsymbol{k}} \widehat{P}_{\boldsymbol{k}}(\boldsymbol{\theta}) C_{\boldsymbol{kl}} h_{\boldsymbol{l}} = 0 \tag{35}$$

with a new power basis $\widehat{P}_{\boldsymbol{k}}$ of bi-degree $(q_x^1 + q_g^1 + 1,\, q_x^2 + q_g^2 + 1)$ and the array $\boldsymbol{h}$ containing the sorted components of the vectors $\boldsymbol{g}_{\boldsymbol{l}}$. The matrix $C_{\boldsymbol{kl}}$ has $(q_x^1 + q_g^1 + 1)(q_x^2 + q_g^2 + 1)$ rows and $4(q_g^1 + 1)(q_g^2 + 1)$ columns. It has at least $4(q_g^1 + 1)(q_g^2 + 1) - (q_x^1 + q_g^1 + 1)(q_x^2 + q_g^2 + 1)$ right null vectors, i.e. the difference between the number of columns and rows. The surface $\boldsymbol{x}(\boldsymbol{\theta})$ has the algebraic degree $2q_x^1 q_x^2$, which is equal to its number of intersections with a line. In order to obtain all the intersections the following condition has to be satisfied

$$4(q_g^1 + 1)(q_g^2 + 1) - (q_x^1 + q_g^1 + 1)(q_x^2 + q_g^2 + 1) \geq 2q_x^1 q_x^2\,. \tag{36}$$

As in the curve case, the degree of $\widetilde{P}(\boldsymbol{\theta})$ along the θ^2 direction can be chosen as $q_g^2 \geq q_x^2 - 1$ yielding

$$q_g^1 \geq 2q_x^1 - 1. \tag{37}$$

By symmetry, it is also valid to choose $q_g^1 \geq q_x^1 - 1$ and $q_g^2 \geq 2q_x^2 - 1$. After the right null vectors of $C_{\boldsymbol{kl}}$ are computed the subsequent steps in computing the intersections are identical to the curve case.

Finally, for the sake of completeness, we mention that the introduced intersection algorithm also applies to triangular finite elements. If $\boldsymbol{x}(\boldsymbol{\theta})$ is a triangular parametric surface of degree q_x then the degree q_g of the auxiliary vector $\boldsymbol{g}(\boldsymbol{\theta})$ has to be chosen to satisfy $q_g \geq 2(q_x - 1)$; see also [9, §3].

4. CONCLUSIONS

The introduced method is able to determine in one-shot, simultaneously without iterating, all the intersection points between a line and a curve or surface. It can be applied to curves or surfaces given in any polynomial basis, like the Lagrange or Bernstein, after a straightforward conversion to the power basis. Aspects requiring further research include cases for which the eigenvalue problem (18) is degenerate and the preconditioning of the eigenvalue problem (25). In the accompanying implementation, several choices have been considered for both.

REFERENCES

1. Hughes TJR, Cottrell JA, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 2005; **194**:4135–4195.
2. Karniadakis G, Sherwin S. *Spectral/hp element methods for computational fluid dynamics*. Oxford University Press, 2013.
3. Wriggers P, Zavarise G. Computational contact mechanics. *Encyclopedia of computational mechanics* 2004; .
4. Xie ZQ, Sevilla R, Hassan O, Morgan K. The generation of arbitrary order curved meshes for 3d finite element analysis. *Computational Mechanics* 2013; **51**:361–374.
5. Turner M, Peiró J, Moxey D. Curvilinear mesh generation using a variational framework. *Computer-Aided Design* 2018; **103**:73–91.
6. Rüberg T, Cirak F. Subdivision-stabilised immersed b-spline finite elements for moving boundary flows. *Computer Methods in Applied Mechanics and Engineering* 2012; **209–212**:266–283.
7. Patrikalakis NM, Maekawa T. *Shape interrogation for computer aided design and manufacturing*. Springer, 2009.
8. Sederberg TW. *Computer Aided Geometric Design*. Computer Aided Geometric Design Course Notes, Brigham Young University, 2012.
9. Busé L. Implicit matrix representations of rational Bézier curves and surfaces. *Computer-Aided Design* 2014; **46**:14–24.
10. Sederberg TW, Chen F. Implicitization using moving curves and surfaces. *SIGGRAPH 1995 Conference Proceedings*, ACM, 1995; 301–308.
11. Strang G. *Introduction to Linear Algebra*. Fifth edn., Wellesley-Cambridge Press, 2016.
12. Xiao X, Sabin M, Cirak F. Interrogation of spline surfaces with application to isogeometric design and analysis of lattice-skin structures. *Submitted for publication.* 2018; .
13. Busé L, Luu Ba T. Matrix-based Implicit Representations of Rational Algebraic Curves and Applications. *Computer Aided Geometric Design* 2010; **27**(9):681–699.
14. Beelen T, Van Dooren P. An improved algorithm for the computation of Kronecker's canonical form of a singular pencil. *Linear Algebra and its Applications* 1988; **105**:9–65.