Finite Elements in Analysis and Design 47 (2011) 1209–1220

Contents lists available at ScienceDirect

Finite Elements in Analysis and Design

journal homepage: www.elsevier.com/locate/finel

Numerical integration over 2D NURBS-shaped domains with applications to NURBS-enhanced FEM

Ruben Sevilla, Sonia Fernández-Méndez *

Laboratori de Càlcul Numèric (www-lacan.upc.edu), Departament de Matemàtica Aplicada III, E.T.S. de Ingenieros de Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Jordi Girona 1, E-08034 Barcelona, Spain

ARTICLE INFO

Article history:
Received 29 June 2007
Received in revised form
26 October 2010
Accepted 25 May 2011
Available online 17 June 2011

Keywords:
Numerical integration
NURBS
NURBS-enhanced finite element method
Gauss–Legendre
Composite rule

ABSTRACT

This paper focuses on the numerical integration of polynomial functions along non-uniform rational B-splines (NURBS) curves and over 2D NURBS-shaped domains, i.e. domains with NURBS boundaries. The integration of the constant function $f=1$ is of special interest in computer aided design software and the integration of very high-order polynomials is a key aspect in the recently proposed NURBS-enhanced finite element method (NEFEM). Several well-known numerical quadratures are compared for the integration of polynomials along NURBS curves, and two transformations for the definition of numerical quadratures in triangles with one edge defined by a trimmed NURBS are proposed, analyzed and compared. When exact integration is feasible, explicit formulas for the selection of the number of integration points are deduced. Numerical examples show the influence of the number of integration points in NEFEM computations.

1. Introduction

Non-uniform rational B-splines (NURBS) [1] are widely used in computer aided design (CAD). Some basic tools of CAD software are the computation of the length of a NURBS curve, the subdivision of a NURBS curve in equally spaced pieces and the computation of the area of a domain with NURBS boundaries, to name a few. These basic operations require the numerical integration of the constant function $f=1$ along NURBS curves and over domains with NURBS boundaries.

On the other hand, CAD models are usually employed by the finite element (FE) community in the preprocess stage, in order to build a spatial discretization of the computational domain. Once the discretization is generated, the exact boundary representation is replaced by a piecewise polynomial approximation. However, in the last decade many authors have pointed out the importance of the geometrical model in FE simulations, see for instance [2–5]. This fact has motivated novel numerical methodologies considering exact CAD descriptions of the computational domain. For instance, NURBS-enhanced finite element method (NEFEM) considers an exact representation of the geometry while maintaining the standard polynomial approximation of the solution. With the NEFEM approach standard FE interpolation and numerical integration is used in the large majority of the domain (i.e., in the interior, for elements not intersecting the boundary) preserving the computational efficiency of classical FE techniques. Specifically designed piecewise polynomial interpolation and numerical integration is required for those FEs along the NURBS boundary.

This paper is devoted to the study of the numerical integration of low- and high-order polynomial functions along trimmed NURBS curves and the integration over curved triangular elements with one edge defined by a trimmed NURBS. Particular emphasis is placed in the numerical integration of high-order polynomials, with applications to NEFEM. Several numerical quadratures are proposed and compared through numerical examples. The generalization to 3D domains is conceptually easy but it requires some extra attention to geometrical aspects and it is presented in [6].

Sections 2 and 3 recall the basic concepts on NURBS and NEFEM in two dimensions. Section 4 is devoted to the integration along NURBS curves. Some well-known 1D numerical quadratures are tested for the numerical integration of low- and high-order polynomials. The integration over triangular elements with one edge defined by a trimmed NURBS is addressed in Section 5. Two transformations for the definition of a numerical quadrature over a curved triangle are considered. The first one is a transformation from a straight-sided triangle in order to test the performance of triangle quadratures. The second one is a transformation from a rectangle to the curved triangle. When exact integration is feasible, explicit formulas for the selection of the number of integration points are deduced. Finally, numerical examples in

* Corresponding author. Tel.: +34 934011625; fax: +34 934011825.
E-mail addresses: ruben.sevilla@upc.edu (R. Sevilla), sonia.fernandez@upc.edu (S. Fernández-Méndez).

0168-874X/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.finel.2011.05.011

Section 6 show the influence of the number of integration points in NEFEM computation.

2. Basic concepts on NURBS curves

This section presents some basic notions of NURBS curves in order to introduce the notation and the concepts employed in the following sections. For a detailed presentation see for instance [1].

A q-th degree NURBS curve is a piecewise rational function defined in parametric form as

$$\boldsymbol{C}(\lambda)=\left(\sum_{i=0}^{\mathrm{n_{cp}}} v_i \boldsymbol{B}_i C_{i,q}(\lambda)\right) \Bigg/ \left(\sum_{i=0}^{\mathrm{n_{cp}}} v_i C_{i,q}(\lambda)\right), \quad \lambda \in [\lambda_a,\lambda_b], \tag{1}$$

where $\{\boldsymbol{B}_i\}$ are the coordinates of the *control points* (determining the *control polygon*), $\{v_i\}$ are their control weights, $\{C_{i,q}(\lambda)\}$ are the B-spline basis functions of degree q, and the interval $[\lambda_a,\lambda_b]$ is called the *parametric space*. The B-spline basis functions are defined recursively from the so-called *knot vector* $\Lambda=\{\lambda_0,\ldots,\lambda_{\mathrm{n_k}}\}=\{\underbrace{\lambda_a,\ldots,\lambda_a}_{q+1},\lambda_{q+1},\ldots,\lambda_{\mathrm{n_k}-q-1},\underbrace{\lambda_b,\ldots,\lambda_b}_{q+1}\}$ by

$$C_{i,0}(\lambda)=\begin{cases} 1, & \lambda \in [\lambda_i,\lambda_{i+1}), \\ 0, & \lambda \notin [\lambda_i,\lambda_{i+1}), \end{cases} \tag{2}$$

$$C_{i,k}(\lambda)=\frac{\lambda-\lambda_i}{\lambda_{i+k}-\lambda_i}C_{i,k-1}(\lambda)+\frac{\lambda_{i+k+1}-\lambda}{\lambda_{i+k+1}-\lambda_{i+1}}C_{i+1,k-1}(\lambda), \tag{3}$$

for $k=1\ldots q$, where λ_i, for $i=0,\ldots,\mathrm{n_k}$, are the *knots* or *breakpoints*. Note that the first and final knots must coincide with the endpoints of the parametrization interval and their multiplicity is always $q+1$. The multiplicity of the remaining knots, when it is larger than one, determines the decrease in the number of continuous derivatives. The number of control points, $\mathrm{n_{cp}}+1$, and knots, $\mathrm{n_k}+1$, are related to the degree of the parametrization, q, by the relation $\mathrm{n_k}=\mathrm{n_{cp}}+q+1$, see [1] for more details.

It is worth remarking that $\sum_{i=0}^{\mathrm{n_{cp}}} C_{i,q}(\lambda)=1$. Thus, Eq. (1) reduces to a (polynomial) B-spline curve when all the control weights are equal.

To summarize, a NURBS is just a piecewise rational function whose definition changes at breakpoints. Fig. 1 shows a NURBS curve and its control polygon.

In practice, CAD manipulators work with *trimmed* NURBS, which are defined as the initial parametrization restricted to a subspace of the parametric space. Fig. 2 shows the NURBS curve represented in Fig. 1 trimmed to the subinterval [0.05,0.75].

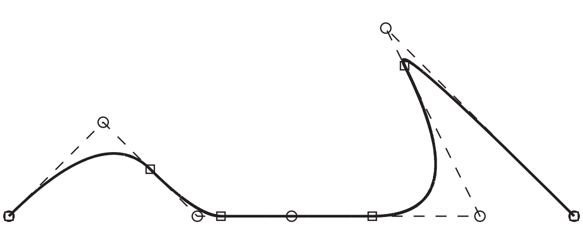

Fig. 1. NURBS curve (solid line), control points (○), control polygon (dashed line) and image of the breakpoints (□).

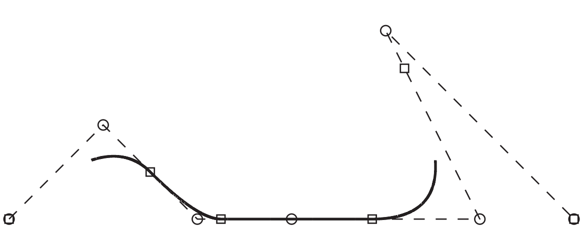

Fig. 2. Trimmed NURBS curve for $\lambda \in [0.05,0.75]$.

3. NEFEM fundamentals

Let $\Omega \subset \mathbb{R}^2$ be an open bounded domain whose boundary $\partial\Omega$ or a portion of it, is curved. A regular partition of the domain $\overline{\Omega}=\bigcup_e \overline{\Omega}_e$ in subdomains, triangles in this work, is assumed, such that $\Omega_i \bigcap \Omega_j=\emptyset$, for $i \neq j$. It is important to remark that, in the following, Ω_e denotes the element with an exact description of the curved boundary. For instance, Fig. 3 shows a domain with part of the boundary described by a NURBS curve corresponding to an airfoil profile, and a triangulation of the domain with curved FEs with an exact boundary representation, i.e. curved *NEFEM elements*.

As usual in FE mesh generation codes, it is assumed that every curved boundary edge belongs to a unique NURBS. That is, one element edge cannot be defined by portions of two (or more) different NURBS curves. But on the contrary, it is important to note that breakpoints, which characterize the piecewise nature of NURBS, are independent of the mesh discretization. Thus, the NURBS parametrization can change its definition inside one edge, that is breakpoints may belong to element edges and do not need to coincide with FE nodes.

Every *interior* element (i.e. elements not having an edge that coincides with the NURBS boundary) can be defined and treated as standard FEs. Therefore, in the vast majority of the domain, interpolation and numerical integration are standard. For elements with at least one edge on the NURBS boundary a specifically designed interpolation and numerical integration is considered.

The polynomial approximation is defined with the Cartesian coordinates $\boldsymbol{x}$,

$$u(\boldsymbol{x}) \simeq u^h(\boldsymbol{x})=\sum_{i=1}^{\mathrm{n_{en}}} u_i N_i(\boldsymbol{x}), \tag{4}$$

where u_i are nodal values, N_i are polynomial shape functions of order p in $\boldsymbol{x}$, and $\mathrm{n_{en}}$ is the number of element nodes. Therefore, the approximation considered in NEFEM ensures reproducibility of polynomials in the physical space for any order of approximation p. See [5] for information about efficient computation of the polynomial base for any degree of interpolation and for any nodal distribution in Ω_e. The exact description of the boundary is used to perform the numerical integration on the physical subdomain Ω_e. Thus, special numerical strategies are required for every element Ω_e.

4. Numerical integration along NURBS curves

This section is devoted to the numerical integration of polynomial functions along NURBS curves. As pointed out in the introduction, the numerical integration of the constant function $f=1$ is of particular interest in CAD. It allows to compute the

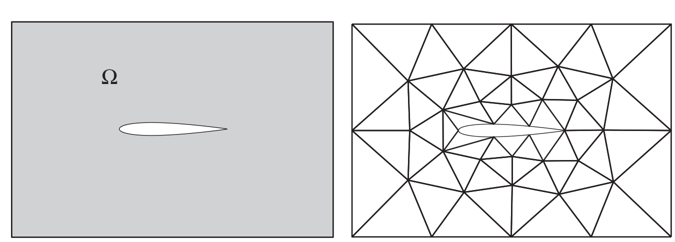

Fig. 3. Physical domain with part of the boundary defined by a NURBS curve (left) and a valid triangulation for NEFEM (right).