Computers and Structures 182 (2017) 55–73

Contents lists available at ScienceDirect

Computers and Structures

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

CAD-based collocation eigenanalysis of 2-D elastic structures

C.G. Provatidis

National Technical University of Athens, School of Mechanical Engineering, Mechanical Design and Control Systems Division, Heroon Polytechniou 9, 15780 Zografou, Greece

ARTICLE INFO

Article history:
Received 16 November 2015
Accepted 17 November 2016

Keywords:
Coons-Gordon
Bézier
B-splines
CAD/CAE
Global collocation
Eigenvalues

ABSTRACT

This paper investigates the performance of the global collocation method for the numerical eigenfrequency extraction of 2-D elastic structures. The method is applied to CAD-based macroelements, starting from the older blending function Coons-Gordon interpolation (based on Lagrange polynomials) and extending to tensor product Bézier and B-splines. Numerical findings show equivalence between Lagrangian and Bézierian macroelements, while a mass lumping procedure is proposed for the former ones. Concerning B-splines, the influence of multiplicity of inner knots and the position of collocation points is thoroughly investigated. The theory is supported by 2-D numerical examples on rectangular and curvilinear structures of simple shape under plane stress conditions, in which the approximate solution rapidly converges towards the exact solution faster than that of the conventional finite element of similar mesh density.

1. Introduction

The use of CAD-based global approximation for the numerical solution of partial differential equations (CAE: computer-aided-engineering) is as old as the theory of computer-aided-design (CAD) itself. It is well known that an industrial team (at General Motors) under Gordon's leadership, in early 1970s, used blending function methods, based on the ideas put forward in [1], to produce some interesting element families [2,3]. Although this team presented the mathematical background for the common description between the geometric model and the unknown variable (CAD/CAE integration), unknown reasons (perhaps the high computational cost) prevented further dissemination of this excellent idea.

One decade later, E1-Zafrany and Cookson [4,5] also used Coons' and Barnhill's ideas for quadrilateral and triangular patches, respectively, whereas Zhaobei and Zhiqiang proposed the use of Coons' interpolation for the analysis of plates and shells [6].

Nevertheless computational results concerning CAD-based isoparametric macroelements (occupying a Coons patch ABCD) were presented for the first time by Kanarachos, Deriziotis and Provatidis [7,8] in 2D potential and elasticity (static and dynamic) problems, where the so-called "C"-elements were successfully compared with conventional finite elements and boundary elements of similar mesh density. For a detailed review (of over 160 references) on the use of CAD-based macroelements the reader is referred to [9].

Summarizing some of the most important previous findings concerning macroelements that occupy a 2D quadrilateral patch ABCD or a 3D hexahedral block ABCDEFGH, in chronological correspondence with the progress in CAD-theory (Coons, Gordon, Bézier, B-splines and NURBS) (see, for instance, [10]), it has been reported that:

(i) (Boundary-only) *Coons* interpolation is capable of creating a broad family of arbitrary-noded elements that may be equivalent to that of Serendipity type. For example, the conventional 4- up to 8-noded 2D elements, as well as the 8- and 20-noded 3D elements can be directly derived applying Coons interpolation [11–14].
(ii) *Gordon-Coons* (transfinite blending function) interpolation when applied to a *structured* macroelement of which the boundary and internal nodal points lay at the same normalized (ξ, η) positions, degenerates to the classical Lagrangian type finite element [14].
(iii) Coons-Gordon interpolation allows for dealing with a (relatively) *unstructured* mesh of internal nodes. Using a *single* quadrilateral macroelement, not only simple shapes such as a rectangular or a circle can be treated, but also it was possible to perform analysis until the complexity of a pi-shaped domain (see, [14–16], among others). For more complex shapes, domain decomposition using large macroelements becomes necessary.

E-mail address: cprovat@central.ntua.gr
URL: http://users.ntua.gr/cprovat

http://dx.doi.org/10.1016/j.compstruc.2016.11.007
0045-7949/

(iv) For a given number of subdivisions per direction (x and y, or ξ and η) along the sides of the patch ABCD, Coons-Gordon interpolation allows for the construction of a large number of alternative macroelements. The two main influencing factors are:
  a. The choice of blending functions and
  b. The univariate interpolation along the four external sides of the macroelement as well as along its so-called "inter-boundaries" [17]. For the completeness of the paper, some basics are summarized in Appendices A and B.

(v) *Bézier* tensor product representation of surface geometry allows also for a global interpolation of the variable, u, within a patch. Since Bézier (Bernstein) polynomials share the same functional space with Lagrange polynomials, i.e. the classes $\{\xi^n\}$ and $\{\eta^n\}$, a CAD-based isogeometric solution is identical with an isoparametric solution based on Lagrangian type polynomials [18].

(vi) *B-splines* tensor product representation of surfaces is an "analogue" to the Lagrange tensor product interpolation, in the sense that univariate Lagrange polynomials are merely substituted by B-splines in both directions. Its applicability in conjunction with Galerkin-Ritz formulation has been summarized by Höllig [19]. An extension of this theory is the dominating NURBS-based isogeometric analysis (shortly IGA) [20].

(vii) *Barnhill* interpolation within triangular patches is capable of producing relevant macroelements. For example, conventional three - and six-noded triangular elements can be easily derived applying Barnhill interpolation (see, [9]), whereas this interpolation has been applied in conjunction with Bézier patches [21].

As a general remark, despite the elegant formulation of the above CAD-based interpolations, even in B-splines (which are characterized by compact support) the computer effort is sometimes comparable or even higher than what the conventional FEA requires. For this reason, in 2005 it was proposed to preserve the CAD-based approximation but use a global collocation method instead of the time consuming Galerkin-Ritz formulation [22, p. 6704]. Relevant works in 2D structures concern Poisson equation [23,24], elastostatics [25], and acoustics [26]. Early studies in 1-D elastodynamics are [27,28]. In the framework of Lagrange polynomials applied to the aforementioned 1-D problems, it was shown that the lumped mass can be easily set equal to the identity matrix when the nodal points are put at the non-uniform position of Gauss points [28].

Within this context, this paper continues extending previously published ideas, from 1-D to 2-D elastodynamics. A preliminary report in 2010 by Filippatos [29] concerning circular and rectangular structures under plane stress conditions, for both Dirichlet and traction-free boundary conditions, shows that, global collocation in conjunction with Lagrange polynomials leads to acceptable results. This paper substantially extends the latter report and is structured as follows:

First, uniform interpolation based on Lagrange-polynomial tensor products is used to interpolate either of displacement components (u_x or u_y) as well as the geometry [$x(\xi,\eta)$ or $y(\xi,\eta)$]. Using Dirichlet and Neumann (traction-free) boundary conditions, several versions of global collocation (*nodal* collocation, *orthogonal* collocation as well as images of Demko's and Greville's abscissae) are thoroughly tested. Orthogonal collocation refers to collocation points located at several positions such as those used in Gaussian quadrature, also the roots of Chebyshev polynomials (of first and second kind). Furthermore, a previous technique for mass lumping is successfully transferred from 1-D [28] to 2-D Lagrange-based collocation problems.

Second, a Bézier tensor product is introduced for both the geometry and displacement representations. Investigation is performed to determine whether the numerical solution using Bézier formulation is identical with that which is obtained using a uniform mesh of the abovementioned Lagrange-polynomial tensor product interpolation.

Third, B-splines tensor products are tested. Particular attention is paid on the handling of corners. The role of multiplicity of inner knots in conjunction with several alternative sets of collocation points is investigated.

Fourth, the above results based on the global collocation are compared with the conventional bilinear FEM as well as with the Galerkin-Ritz formulation based on the same global shape functions with those used in the global collocation schemes, for the same multiplicity of inner knots.

2. One-dimensional shape functions

This section refers to either of the Cartesian (x- and y-) directions, or even the curvilinear ξ- and η-directions (along the sides AB and DA, respectively, of a patch ABCD). Below, n corresponds to either of the subdivisions n_ξ or n_η, whereas the corresponding normalized domain is $[0, 1]$.

Given the breakpoints $x_0, \ldots, x_n$, where $x_0 \equiv 0$ and $x_n \equiv 1$, among several alternative interpolations, in this paper we focus on the following.

I. *Lagrange polynomials*

$$L_{j,n}(x) = \frac{(x - x_0)\ldots(x - x_{j-1})(x - x_{j+1})\ldots(x - x_n)}{(x_j - x_0)\ldots(x_j - x_{j-1})(x_j - x_{j+1})\ldots(x_j - x_n)} \tag{1}$$

II. *Bezier* (Bernstein) polynomials

$$B_{i,n}(x) = \binom{n}{i} x^i (1-x)^{n-i} = \frac{n!}{i!(n-i)!} x^i (1-x)^{n-i} \tag{2}$$

III. *B-Splines*

We start with the abovementioned breakpoints,

$$\{\mathbf{x}_b\} = [x_0, \ldots, x_n], \tag{3}$$

and then we introduce the *knot vector* $\{\mathbf{V}\}$:

$$\{\mathbf{V}\} = [v_0, \ldots, v_m], \tag{4}$$

which highly depends on the chosen *multiplicity* λ of internal knots (usually single, double, etc.), as follows:

- Multiplicity $\lambda = 1$:

$$\{\mathbf{V}\}_{\lambda=1} = \left[\underbrace{x_0, \ldots, x_0}_{p+1}, x_1, x_2, \ldots, x_{n-1}, \underbrace{x_n, \ldots, x_n}_{p+1}\right] \tag{5}$$

- Multiplicity $\lambda = 2$:

$$\{\mathbf{V}\}_{\lambda=2} = \left[\underbrace{x_0, \ldots, x_0}_{p+1}, \underbrace{x_1, x_1}_{2}, \underbrace{x_2, x_2}_{2}, \ldots, \underbrace{x_{n-1}, x_{n-1}}_{2}, \underbrace{x_n, \ldots, x_n}_{p+1}\right], \tag{6}$$

and so on.

It is noted that:

- the maximum allowable multiplicity is $\lambda = p - 1$.
- In all cases, $C^{p-\lambda}$ -continuity is ensured.

Therefore, Eqs. (5) and (6) and so on, lead to the unified relationship:

$$m = 2(p+1) + \lambda(n-1) - 1, \quad \lambda = 1, 2, \ldots \tag{7}$$