A Review on Attempts towards CAD/CAE Integration Using Macroelements

Christopher Provatidis

National Technical University of Athens, Greece
*Corresponding Author: cprovat@gmail.com

Copyright © 2013 Horizon Research Publishing All rights reserved.

Abstract In this paper we review several aspects of older and contemporary attempts to integrate computer-aided design (CAD: geometric model) and computer-aided engineering (CAE: finite elements, boundary elements, etc.). After a short review on formulas for the description of CAD surfaces, a systematic mechanism for creating several types of corresponding isoparametric macroelements is presented. Gordon-Coons is initially applied in conjunction with piecewise linear, Lagrange polynomials and natural B-splines. Then, it is extended to more basis and blending functions. In addition to the well-known ‘Lagrangian’-type elements, equivalent ‘Bézierian’-type elements are introduced. Tensor product B-splines and aspects of NURBS isogeometric formulation are given. In addition to quadrilaterals, triangular macroelements based on Barnhill’s interpolation are presented for the first time. The review covers applications of CAD-based macroelements in conjunction with the Galerkin-Ritz formulation, the Boundary Element Method, as well as recent Global Collocation procedures. A numerical example on a vibrating membrane elucidates the performance of the CAD-based global interpolation and depicts its superiority over the conventional finite element method.

Keywords CAD, CAE, Global Interpolation, Finite Element, Boundary Element, Collocation Techniques

1. Introduction

Computer-Aided Design (CAD) is the use of computer systems to assist in the creation, modification, analysis, or optimization of a design. One of the earliest personalities who have contributed in this area, and had a vision of interactive computer graphics as a powerful design tool, were *Steven Anson Coons* (1912 – 1979), a professor in the Mechanical Engineering Department at Massachusetts Institute of Technology (MIT) during the 1950s and 1960s. During World War II, he worked on the design of aircraft surfaces,

developing the mathematics to describe generalized “surface patches”. At MIT’s Electronic Systems Laboratory he investigated the mathematical formulation for these patches, and in 1967 published one of the most significant contributions to the area of geometric design, a treatise which has become known as “The Little Red Book” [1]. His “Coons Patch” was a formulation that presented the notation, mathematical foundation, and intuitive interpretation of an idea that would ultimately become the foundation for other surface descriptions that are commonly used today, such as B-spline surfaces, NURB surfaces, etc. His technique for describing a surface was to construct it out of collections of adjacent patches, which had continuity constraints that would allow surfaces to have curvature which was expected by the designer. Each patch was defined by four boundary curves, and a set of “blending functions” that defined how the interior was constructed out of interpolated values of the boundaries. The interested reader may find more details in [2].

Computer-Aided Engineering (CAE) is the broad usage of computer software to aid in engineering tasks. It includes computer-aided design (CAD), computer-aided analysis (CAA), computer-integrated manufacturing (CIM), computer-aided manufacturing (CAM), material requirements planning (MRP), and computer-aided planning (CAP). In a more strict sense, CAE areas include:

- Stress analysis on components and assemblies using FEA (Finite Element Analysis);
- Thermal and fluid flow analysis Computational fluid dynamics (CFD);
- Multibody dynamics (MBD) & Kinematics;
- Analysis tools for process (manufacturing) simulation for operations such as casting, molding, and die press forming.
- Optimization of the product or process.
- Safety analysis of postulate loss-of-coolant accident in nuclear reactor using realistic thermal-hydraulics code.
- In general, when omitting the abovementioned planning tasks, there are three phases into which any CAE procedure is divided:

- Pre-processing – defining the model and environmental factors to be applied to it (typically a finite element model, but facet, voxel and thin sheet methods are also used).
- Analysis solver (usually performed on high powered computers).
- Post-processing of results (using visualization tools).

This cycle is iterated, often many times, either manually or using built-in procedures, or finally by linking the entire procedure into a commercial optimization software such as [3, 4].

Henceforth, CAE will restrict to the finite element (FEM/FEA) and the boundary element (BEM) analysis, whereas a very recent Global Collocation Method (GCM) will be reviewed and commented. It is reminded that FEM decomposes the model of the structure in small areas or volumes, called finite elements [5, 6], whereas BEM deals mostly with the boundary of the structure [7].

It is accepted that the pioneers in the FEA area are John HadjiArgyris (1913–2004), Olgierd Cecil Zienkiewicz (1921–2009) and Ray Clough (1920–today). Some personal views are [8–12].

The decade of 1960s was very important for designers, as since then they have long used computers for their calculations. Initial developments were carried out in the 1960s within the aircraft and automotive industries in the area of 3D surface construction and NC programming, most of it independent of one another and often not publicly published until much later. Some of the mathematical description work on curves was developed in the early 1940s by Isaac Jacob Schoenberg, Apalatequi (Douglas Aircraft) and Roy Liming (North American Aircraft), however probably the most important work on polynomial curves and sculptured surface was done by Pierre Bézier (Renault), Paul de Casteljau (Citroen), Steven Anson Coons (MIT, Ford), James Ferguson (Boeing), Carl de Boor (GM), Birkhoff (GM) and Garabedian (GM) in the 1960s and W. Gordon (GM).

It is argued that a turning point was the development of SKETCHPAD system in MIT in 1963 by Ivan Sutherland [13], who was a student of S.A. Coons; however his PhD thesis was supervised by C.E. Shannon (1916–2001), the "father of information theory". The distinctive feature of SKETCHPAD was that it allowed the designer to interact with his computer graphically: the design can be fed into the computer by drawing on a CRT monitor with a light pen. Effectively, it was a prototype of graphical user interface (GUI), an indispensable feature of modern CAD.

First commercial applications of CAD were in large companies in the automotive and aerospace industries, as well as in electronics. Only large corporations could afford the computers capable of performing the calculations. Notable company projects were at GM (Dr. Patrick J. Hanratty) with DAC-1 (Design Augmented by Computer) 1964; Lockheed projects; Bell GRAPHIC 1 and at Renault (Bézier) UNISURF 1971 car body design and tooling. One of the most influential events in the development of CAD was the founding of MCS (Manufacturing and Consulting

Services Inc.) in 1971 by Dr. P. J. Hanratty[14], who wrote the system ADAM (Automated Drafting And Machining) but more importantly supplied code to companies such as McDonnell Douglas (Unigraphics), Computer vision (CADDs), Calma, Gerber, Autotrol and Control Data. As computers became more affordable, the application areas have gradually expanded. The development of *CAD software* for personal desk-top computers was the impetus for almost universal application in all areas of construction. Other key points in the 1960s and 1970s would be the foundation of CAD systems United Computing, Intergraph, IBM, Intergraph IGDS in 1974 (which led to Bentley MicroStation in 1984) CAD implementations have evolved dramatically since then. Initially, with 2D in the 1970s, it was typically limited to producing drawings similar to hand-drafted drawings. Advances in programming and computer hardware, notably solid modeling in the 1980s, have allowed more versatile applications of computers in design activities.

Key products for 1981 were the solid modelling packages -Romulus (ShapeData) and Uni-Solid (Unigraphics) based on PADL-2 and the release of the surface modeler CATIA (Dassault Systemes). Autodesk was founded 1982 by John Walker, which led to the 2D system AutoCAD. The next milestone was the release of Pro/ENGINEER in 1988, which heralded greater usage of feature-based modeling methods and parametric linking of the parameters of features. Also of importance to the development of CAD was the development of the B-rep solid modeling kernels (engines for manipulating geometrically and topologically consistent 3D objects) Parasolid (ShapeData) and ACIS (Spatial Technology Inc.) at the end of the 1980s and beginning of the 1990s, both inspired by the work of Ian Braid. This led to the release of mid-range packages such as SolidWorks in 1995, SolidEdge (Intergraph) in 1996, and IronCAD in 1998. Nowadays CAD is one of the main tools used in designing products.

In order to analyze mechanical components, which are usually characterized by complex shapes, the computer-aided-design (CAD) model is usually exported in the form of a neutral file (IGES, DXF, STEP) that is further imported by a selective finite element (FEM) or boundary element (BEM) code. This procedure induces some difficulties such as the appearance of double points (or double lines) or loss of at least a few geometrical data. Of course, specific CAD converters such as PATRAN can export many types of FEM data that can be immediately feed commercial FEM codes. Specialized CAD/CAM/CAE integrated systems such as PRO/ENGINEER, IDEAS and Dassault Systèmes (SOLIDWORKS, PLM: Product Lifecycle Solutions) exist for the last 20 or 25 years. However, in all these tools the computational mesh generation is a time-consuming task that has to follow the creation of the CAD model. A survey of thirty-four finite element systems until 1981 was reported by Brebbia [15].

The term "*CAD/CAE integration*" may have multiple meanings, which will be better understood through an example. Suppose that a bicycle designer assigns a tentative

tube size for the skeleton of the structure (CAD); then, an integrated CAD/CAE system has to automatically understand not only the diameter and the thickness but also the second moment of inertia of this tube, without the engineer having to perform the relevant numerical calculations. Closely related, the engineer can generate the mesh without exiting the CAD system and then entering the CAE system; in other words, both systems are integrated in one, sharing the same database as is possible.

A second point of view is the usually reported fact that the CAD/CAE integration and scientific visualization is related to both the reduction of errors in the information transfer from system to system as well as the reduction of memory resources and overall computational time [16, 17]. The ‘vehicle’ for CAD/CAE integration is to use a common platform to describe the geometry and the unknown variable that satisfies a partial differential equation (PDE) of a multi-physics boundary-value-problem (e.g. heat transfer, elastostatics, thermo-elasticity, fluid mechanics, electromagnetic, etc). A third point of CAD/CAE integration is to replace the usual finite elements (of small-size) with others of larger size applying the same interpolation for both the geometry and the variable. Relevant macro-elements are usually of “isoparametric” type whereas “isogeometric” ones have appeared [18].

The lack of a complete, synthetic and critical report on the abovementioned third point of CAD/CAE integration, which is directly related to the development and use of macroelements, has motivated the author to write this review article.

The paper is structured as follows. *Section 2* summarizes the one-dimensional (univariate) interpolation as well as the five basic CAD formulations. *Section 3* reviews the usual applications in which Coons’ interpolation has been used, as well as the derivation of global shape functions. *Section 4* provides details about the isoparametric or isogeometric-like approximation that is used for the interpolation of both the geometry and the variable or the unknown coefficients. *Section 5* reviews the three possible computational methods to be applied in conjunction with the global interpolation within subregions, and provides the numerical procedure for the estimation of mass and stiffness matrices of the structure. *Section 6* presents original numerical results concerning the extraction of natural frequencies for a fixed rectangular membrane. *Section 7* and *Section 8* are the “Discussion” and the “Conclusions”, respectively.

2. CAD Surface Interpolations

2.1. General

Historically, there are rather five main interpolation formulations in CAD theory [19]: (i) Coons interpolation, (ii) Gordon-Coons interpolation, (iii) Bézier interpolation, (iv) B-splines interpolation, and (v) NURBS. The first two of

them directly refer to the coordinates of nodal points that *belong* to the boundary or the interior of a patch, whereas the last three refer to control points (at the edges of polygonal generators that generally do *not* belong to the curvilinear boundary). In all five cases the interpolation of geometry refers to the coordinates of either nodal points or control points multiplied by proper basis functions. For the sake of completeness, we start with one-dimensional and then extend to two- and three-dimensional interpolation.

2.2. One-Dimensional Interpolation

Let us consider the domain $a \leq x \leq b$ in which we wish to interpolate the function $f(x)$.

Among others, the most usual interpolations of the function $f(x)$ are: (i) Power series, (ii) Lagrange polynomials, (iii) Legendre polynomials (p-method), (iv) Chebyshev polynomials, (v) Bernstein polynomials (Bézier curve), (vi) B-splines, and (vii) NURBS. All of them will be briefly described below.

2.2.1. Power Series

The function is approximated by:

$$f(x) = a_0 + a_1x + \dots + a_nx^n \equiv \sum_{j=0}^n a_jx^j \quad (1)$$

2.2.2. Lagrange Polynomials

The function is approximated by:

$$f(x) = \sum_{j=0}^n L_j(x) f(x_j) \equiv \sum_{j=0}^n L_j(x) f_j \quad (2)$$

where $L_j(x)$ denotes the well-known Lagrange polynomial (see, for example, [20]).

2.2.3. P-Method

The relevant series expansion includes the nodal values at the ends, $f(a)$ and $f(b)$ for which the classical linear hat functions are considered, i.e.: $N_1(x) = 1 - x/L$ and $N_2(x) = x/L$ where $L = b - a$. In addition, we define a certain polynomial degree ‘ p ’, which leads to a number of functions ϕ_j are defined in terms of the *Legendre polynomial* P_{j-1} :

$$\phi_j(x) = \sqrt{\frac{2j-1}{2}} \int_{-1}^x P_{j-1}(t) dt, \quad j = 2, 3, \dots \quad (3)$$

It is reminded that the Legendre polynomials are given by

$$P_m(x) = \frac{1}{2^m m!} \frac{d^m}{dx^m} (x^2 - 1)^m, \quad (4)$$

and they are solutions of the ordinary differential operation:

$$(1 - x^2) y'' - 2xy' + m(m+1)y = 0 \quad (5)$$

It is also reminded that the well-known Gauss points are roots of the above Legendre polynomials.

The basis functions N_1, N_2 are called *nodal shape functions* or *external shape functions* or *external modes*. The basis functions $N_i = \phi_{i-1}, i = 3, 4, \dots, p+1$ are called internal shape functions or internal modes, or, sometimes, “bubble modes”. Therefore:

$$f(x) = N_1(x)f_1 + N_2(x)f_2 + \sum_{j=3}^{p+1} N_j(x)a_j \quad (6)$$

The advantage of these functions is that they are orthogonal thus leading to banded mass and stiffness matrices [21, pp. 37-47].

2.2.4. Chebyshev Polynomials

Although these polynomials do not directly appear in CAD formulations, it is instructive to mention that they can approximate a function with the same accuracy with the power series. They are categorized in polynomials of first and second kind. For example, the Chebyshev polynomials of the second kind are defined as:

$$\begin{aligned} U_m(x) &= \frac{\sin\{(m+1)\cos^{-1}x\}}{\sin(\cos^{-1}x)} \\ &= \binom{m+1}{1}x^m - \binom{m+1}{3}x^{m-2}(1-x^2) \\ &\quad + \binom{m+1}{5}x^{m-4}(1-x^2)^2 - \dots \end{aligned} \quad (7)$$

and their roots are given by

$$\hat{\xi}_i = \cos \frac{i\pi}{(m+1)}, \quad i = 1, \dots, m. \quad (8)$$

As we will see later, the above roots can be used as collocation points in a global collocation method, which is an alternative to the Galerkin-Ritz procedure [22, 23].

2.2.5. Bernstein Polynomials (Bézier curve)

2.2.5.1. Nonrational Bézier Curve

A curve of n -th degree is defined as

$$\mathbf{C}(u) = \sum_{i=0}^n B_{i,n}(u)\mathbf{P}_i \quad 0 \leq u \leq 1 \quad (9)$$

where $\mathbf{C}(u) = [x(u) \ y(u)]^T$ are the Cartesian coordinates on the xy -plane.

The basis functions, $B_{i,n}(u)$, are the classical Bernstein polynomials that are defined as

$$B_{i,n}(u) = \binom{n}{i} u^i (1-u)^{n-i} = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i} \quad (10)$$

The geometrical coefficients, $\{\mathbf{P}_i\}$, are called *control points*. Equation (10) denotes that for a curve that is determined by $(n+1)$ control points, the term of highest degree is u^n . This implies that the polynomial degree of Bézier curve is determined by the number of control points. The basic properties of a Bézier curve are many and can be found elsewhere [24-27]. In brief:

- The curve passes through the first ($\mathbf{P}_0$) and the last ($\mathbf{P}_n$) control points and its tangent at these ends has the direction of the first straight segment ($\mathbf{P}_0\mathbf{P}_1$) and the last one ($\mathbf{P}_{n-1}\mathbf{P}_n$), respectively, of the control polygon ($\mathbf{P}_0\mathbf{P}_1 \dots \mathbf{P}_{n-1}\mathbf{P}_n$).
- The Bernstein polynomials are nonnegative: $B_{i,n}(u) \geq 0$
- They have the rigid-body property (partition of unity): $\sum B_{i,n}(u) = 1, \forall u \in [0, 1]$.
- They are smaller than unit except of those associated to the ends where they equal to unity, and appear a maximum in the interval $[0, 1]$ at $u_{\tan=0} = i/n$.
- They are symmetric with respect to the position $u = 1/2, \forall n$.
- They are rapidly calculated by the following recursive formula:

$$\begin{aligned} B_{i,n}(u) &= (1-u)B_{i,n-1}(u) + uB_{i-1,n-1}(u), \\ B_{i,n}(u) &\equiv 0 \ (i < 0, i > n) \end{aligned} \quad (11)$$

- Their derivative is also calculated by the following recursive formula:

$$B'_{i,n}(u) = \frac{dB_{i,n}(u)}{du} = n(B_{i-1,n-1}(u) - B_{i,n-1}(u)), \quad (12)$$

$$B_{-1,n-1}(u) \equiv B_{n,n-1}(u) \equiv 0$$

It is also remarkable that:

- The time-consuming computation of the i -combinations $\binom{n}{i}$ is simplified by using *Casteljau's* algorithm.
- Also, the interpolation of Bézier curve is mathematically equivalent with the power series (1), for the same value of n .
- Similarly, both the p -method and the Bernstein polynomials are equivalent with the Lagrange polynomials [see, (2)], of course for the same value of n .

2.2.5.2. Rational Bézier Curve

Since the conical surfaces, i.e. circles, ellipses, hyperbolas, cylinders, cones, spheres, et cetera, are not described precisely by means of polynomials, but require the use of rational functions, the form:

$$x(u) = \frac{X(u)}{W(u)}, \quad y(u) = \frac{Y(u)}{W(u)}, \quad (13)$$

it became necessary to introduce the n -th rational Bézier curve, which is defined as:

$$C(u) = \{x(u), y(u)\} = \frac{\sum_{i=0}^n B_{i,n}(u) \cdot w_i \cdot P_i\{x_i, y_i\}}{\sum_{i=0}^n B_{i,n}(u) \cdot w_i}, \quad 0 \leq u \leq 1 \quad (14)$$

or

$$C(u) = \sum_{i=0}^n R_{i,n}(u) \cdot P_i, \quad 0 \leq u \leq 1 \quad (15)$$

where

$$R_{i,n}(u) = \frac{B_{i,n}(u) \cdot w_i}{\sum_{j=0}^n B_{j,n}(u) \cdot w_j}, \quad 0 \leq u \leq 1 \quad (16)$$

2.2.6. B-Splines

2.2.6.1. Older Definitions

The meaning of splines was published in 1946 and later by Schoenberg [28]. It refers to the points $(x_0, f_0), (x_1, f_1), \dots, (x_n, f_n)$ with $x_0 < x_1 < \dots < x_n$, which we wish to interpolate through a multiply-defined function $f(x)$. The points $x_0, x_1, \dots, x_n$ are called “*breakpoints*”. For the sake of brevity and without loss of generality, we reduce to (cubic) polynomials of third degree. The desired properties are as follows:

- In each interval $x_{i-1} \leq x \leq x_i$, with $i = 1, 2, \dots, n$, the function $f(x)$ is a cubic polynomial.
- The function $f(x)$, as well as the first and second derivatives, are continuous at the above points.

Introducing the truncated power as:

$$\langle x - x_i \rangle_+^m = \begin{cases} 0, & x \leq x_i \\ (x - x_i)^m, & x > x_i \end{cases}, \quad (17)$$

which has $C^{(m-1)}$ -continuity, the original expression consists of a power series in the form [28]:

$$f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \sum_{j=1}^{n-1} b_j \langle x - x_j \rangle_+^3 \quad (18)$$

It is apparent that (18) includes $(n+3)$ terms and ensures C^2 -continuity, because for simplicity we considered $m = 3$. Alternatively, (18) can be modified so as to include additional truncated polynomials of second degree, i.e.:

$$f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \sum_{j=1}^{n-1} b_j \langle x - x_j \rangle_+^2 + \sum_{j=1}^{n-1} c_j \langle x - x_j \rangle_+^3, \quad (19)$$

Obviously, (19) includes $2(n+1)$ terms and ensures C^1 -continuity.

2.2.6.2. Contemporary Procedures

The breakthrough was made in 1972, independently by Cox [29] and DeBoor [30], who both achieved the B-splines and their derivatives to be rapidly computed through recursive formulas [including (11) and (12)]. Charles DeBoor [26] continued his research and his initial FORTRAN codes exist today in MATLAB™ (spline toolbox) under the name ‘*spcol*’, among others [31].

In brief, for a nondecreasing set of $(n+1)$ *breakpoints* $(x_0, x_1, \dots, x_{n-1}, x_n)$, and for a concrete polynomial degree ‘ p ’, we define the *knot* vector ‘ U ’, which strongly depends on the multiplicity of the internal nodes, as follows:

$$U = \begin{cases} \left\{ \underbrace{x_0, \dots, x_0}_{(p+1) \text{ terms}}, x_1, \dots, x_{n-1}, \underbrace{x_n, \dots, x_n}_{(p+1) \text{ terms}} \right\}, & \text{multiplicity} = 1 \\ \left\{ \underbrace{x_0, \dots, x_0}_{(p+1) \text{ terms}}, \underbrace{x_1, x_1}_{2 \text{ terms}}, \dots, \underbrace{x_{n-1}, x_{n-1}}_{2 \text{ terms}}, \underbrace{x_n, \dots, x_n}_{(p+1) \text{ terms}} \right\}, & \text{multiplicity} = 2 \end{cases} \quad (20)$$

Based on either of the knot vectors of (20), $(m+1)$ control points $(P_0, P_1, \dots, P_{m-1}, P_m)$ are constructed, so as the following relationship always holds:

$$m = n + p + 1 \quad (21)$$

Let us consider a nondecreasing sequence $U = \{u_0, \dots, u_m\}$ of real numbers, for example, $u_i \leq u_{i+1}$, $i = 0, \dots, m-1$. The elements u_i are called *knots*, and U is the knot-vector. The i -th B-splines basis functions of degree p (order $p+1$), denoted by $N_{i,p}(u)$, is defined as:

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (22)$$

2.2.6.3. Contemporary Versus Older Definitions

Some young readers may believe that (22) is different than (18) or (19). However, for example in case of $p = 3$ (cubic splines), it is trivial to prove that:

- When the multiplicity equals to *one*, the $(n+3)$ basis functions of (22) are *equivalent* (not identical) to those basis functions given by (18).
- When the multiplicity equals to *two*, the $2(n+1)$ basis functions of (22) are *equivalent* (not identical) to those basis functions given by (19).
- When there are no internal points ($n = 1$), B-splines coincides with p -th Bézier.

- Therefore, a lot of research conducted in 1960s using the older framework ((18) and (19)) may be repeated on the new framework of (22).

2.2.7. NonUniform Rational B-Spline (NURBS) Curve

Similarly to the rational Bézier curve [see, (16)], non uniform rational B-splines (NURBS) is an extension of the B-splines in the following form:

$$C(u) = \frac{\sum_{i=0}^n N_{i,p}(u) w_i P_i}{\sum_{i=0}^n N_{i,p}(u) w_i} \quad (23)$$

where $\{w_i\}$ are the weights. Usually, $a = 0$, $b = 1$ and $w_i > 0$ for all i . Taking the rational basis functions as,

$$R_{i,p}(u) = \frac{N_{i,p}(u) w_i}{\sum_{j=0}^n N_{j,p}(u) w_j}, \quad (24)$$

the curve is expressed as:

$$C(u) = \sum_{i=0}^n R_{i,p}(u) w_i P_i \quad (25)$$

and they are piecewise rational functions in the interval $u \in [0, 1]$.

2.2.8. Comparison between Different Sets Of Basis Functions

It is well known that power series is identical with *Lagrange polynomials* [24, pp. 304-308]; the only difference is that the first includes arbitrary coefficients, whereas the second cardinal shape functions (of [1,0]-type) that are multiplied by the associated nodal values.

Chebyshev polynomials are also multiplied by arbitrary coefficients and they may have some advantages when collocating at their roots [22, 23].

The use of *Legendre* polynomials is somehow different. The relevant series expansion includes the nodal values at the ends, $f(a)$ and $f(b)$ for which the classical linear hat functions are considered, i.e.: $N_1(x) = 1 - x/L$ and $N_2(x) = 1 - x/L$ where $L = b - a$. In addition, a certain desired number of “bubble” functions, in the form of differences between Legendre polynomials, are usually used. The advantage of these functions is that they are orthogonal thus leading to banded matrices [21, pp. 39-46].

Bernstein polynomials have been used in the construction of the well-known Bézier curves, and can be found in many textbooks such as [25-27].

B-splines is nothing new but the usual spline interpolation, which can be found in every textbook of numerical analysis such as [20], whereas detailed information is given elsewhere [27-30]. The superiority of B-splines is that it can handle every high number of data, in contrast to Lagrange polynomials that cannot handle more than ten to twelve

points at maximum.

Finally, *NURBS* is an extension of B-splines as it uses weights for every control point.

As one can notice in Fig.1, while Lagrange polynomials may exceed the unity and be characterized by high values particularly in the neighbourhood of the ends, Bernstein polynomials (or even general B-splines, not shown) are always less than or equal to unity (at the end nodes). It is worth-mentioning that, in both cases, the sum of the shape functions is equal to unity (rigid-body property, partition of unity). Similarly, in the p -method the sum of the first and last basis functions (N_1 and N_n , associated to the ends of the domain) is equal to unity, whereas the intermediate “bubble functions”, ϕ_j , vanish at the ends but their sum is different than zero at every internal point (see, [21]).

A side ascertainment on the equivalency between the three functional sets in power series, i.e. Lagrange polynomials, Bernstein polynomials and the p -method is the numerical finding that the calculated eigenvalues are identical [32, 33].

2.3. Two-Dimensional (2D) Interpolation

In brief, surfaces are categorized into quadrilateral and triangular patches, i.e. with four and three sides, respectively. The interpolations can treat either straight or curvilinear sides of the patch. The oldest way to create a 2D patch is Coons interpolation [1]. Later, tensor products of Bézier, B-splines and NURBS were applied [19, 27].

2.3.1. Quadrilateral Patches

We distinguish two categories. The first deals with data along the four boundaries of the quadrilateral; in other words, it consists the “*boundary-only*” formulation with nodal points along the boundary only. The second formulation uses the aforementioned boundary data as well as additional data in the interior of the quadrilateral; therefore it includes internal nodes as well, and is called “*transfinite*” interpolation.

Figure 1. Comparison between three equivalent sets of basis functions for seven uniform subdivisions using: (i) Bernstein (Bézier) polynomials, (ii)

Lagrange polynomials, and (iii) p -method

2.3.1.1. Coons Interpolation

A four-sided region ABCD, as shown on the left of Fig.2, can easily be mapped to a unit square in the $\xi\eta$ parametric domain shown on the right of the same figure by the method of Coons' patch [1]. For purposes of generalization, the relevant theory is given below using suitable projections.

Figure 2. Definition of the Coons-patch macroelement ABCD

First, the concept of the *lofting projector* 'P' is introduced. This projector is any idempotent linear operator, which maps a true surface to an approximate surface, subject to certain interpolatory constraints.

Let us assume that the Cartesian co-ordinates $\mathbf{x}(\xi, \eta) = \{x, y, z\}^T$ in A , with ξ and η denoting normalized co-ordinates, are known at the boundaries ($\xi = 0, 1$; $\eta = 0, 1$) of a curvilinear patch of area A . Let us also define the well-known cardinal blending functions:

$$E_0(\xi) = 1 - \xi, E_1(\xi) = \xi, E_0(\eta) = 1 - \eta, E_1(\eta) = \eta \quad (26)$$

Now, the following unidirectional, or lofting, operators $P_\xi\{\mathbf{x}\}$ and $P_\eta\{\mathbf{x}\}$ may be constructed (summation over repeated indices is understood):

$$\begin{aligned} P_\xi\{\mathbf{x}\} &= \mathbf{x}(\xi_i, \eta) \cdot E_i(\xi), \\ P_\eta\{\mathbf{x}\} &= \mathbf{x}(\xi, \eta_j) \cdot E_j(\eta). \end{aligned} \quad (27)$$

The above lofting operators form the basis for the definition of more complex operators with blending interpolation properties in more than one direction. So, the two-dimensional lofting operator:

$$P_{\xi\eta}\{\mathbf{x}\} = P_\xi P_\eta\{\mathbf{x}\} = \mathbf{x}(\xi_i, \eta_j) \cdot E_i(\xi) \cdot E_j(\eta) \quad (28)$$

can be constructed with the aid of the unidirectional operators $P_\xi\{\mathbf{x}\}$ and $P_\eta\{\mathbf{x}\}$.

Finally, the co-ordinates of any point in the interior of the curvilinear patch is approximated as:

$$\mathbf{x}(\xi, \eta) = (P_\xi + P_\eta - P_{\xi\eta})\{\mathbf{x}\} \quad (29)$$

or, using conventional notation, as:

$$\begin{aligned} \mathbf{x}(\xi, \eta) &= (1 - \xi)\mathbf{x}(0, \eta) + \xi\mathbf{x}(1, \eta) \\ &+ (1 - \eta)\mathbf{x}(\xi, 0) + \eta\mathbf{x}(\xi, 1) \\ &- (1 - \xi)(1 - \eta)\mathbf{x}(0, 0) - \xi(1 - \eta)\mathbf{x}(1, 0) \\ &- \xi\eta\mathbf{x}(1, 1) - (1 - \xi)\eta\mathbf{x}(0, 1) \end{aligned} \quad (30)$$

2.3.1.2. Gordon Interpolation

According to Gordon's interpolation formula [34], the Cartesian coordinates $\mathbf{x}(\xi, \eta) = [x(\xi, \eta), y(\xi, \eta)]^T$ of each point P of in the patch can be approximated by its boundaries ($\mathbf{x}(\xi, 0), \mathbf{x}(\xi, 1), \mathbf{x}(0, \eta), \mathbf{x}(1, \eta)$) as well as by its inter-boundaries (if any) as follows:

$$\mathbf{x}(\xi, \eta) = P_\xi(\mathbf{x}) + P_\eta(\mathbf{x}) - P_{\xi\eta}(\mathbf{x}) \quad (31)$$

and

$$\mathbf{u}(\xi, \eta) = P_\xi(\mathbf{u}) + P_\eta(\mathbf{u}) - P_{\xi\eta}(\mathbf{u}) \quad (32)$$

For example, based on Gordon's interpolation, it is easy to derive the shape functions for a five-node element having one internal node at its centre (in addition to the four ones at the corners) [35, 36].

Remark: It has been found that when the nodal points on the opposite sides of a patch possess identical normalized positions with those nodes in the interior, then Gordon-Coons interpolation degenerates to the cross product $P_{\xi\eta}$ [37, p.328]. Under the latter conditions, when Lagrange polynomials have been used, Coons interpolation coincides with the well-known interpolation in finite elements of Lagrange type (or Lagrange family) [5, p.153]. In conclusion, in patches where structured meshes of nodal points are used, all three basic CAD interpolations (Coons, Bézier, and B-splines) are expressed by a tensor product of 1-D ξ - and η -interpolations. Details are given below.

Figure 3. Definition of the Gordon-Coons macroelement ABCD

2.3.1.3. Tensor Products

By defining the breakpoints along each side of a

curvilinear quadrilateral ABCD or a box-like volume ABCDEFGH, for a certain polynomial degree (p) and for a certain multiplicity (usually one or two), the basis functions $N_{i,p}(u)$ are determined. It is noted that here u is either of the normalized coordinates ξ , η , or ζ , whereas the number of basis functions equal to the $(m+1)$ control points in the particular direction (m_ξ , m_η , and m_ζ , in the ξ -, η -, or ζ -direction, respectively). Then, the aforementioned one-dimensional control points produce $n = (1 + m_\xi) \times (1 + m_\eta)$ or $n = (1 + m_\xi) \times (1 + m_\eta) \times (1 + m_\zeta)$ control points for 2D and 3D problems, respectively. Each of the aforementioned control point is associated with one global shape function given by:

$$N_I = \begin{cases} N_{ip}(\xi) \cdot N_{jp}(\eta), & \text{for 2D problems} \\ N_{ip}(\xi) \cdot N_{jp}(\eta) \cdot N_{kp}(\zeta), & \text{for 3D problems} \end{cases} \quad (33)$$

Therefore, the geometry is interpolated by the expressions:

$$\begin{aligned} x(\xi, \eta) &= \sum_{I=1}^n N_I(\xi, \eta) x_I, \\ y(\xi, \eta) &= \sum_{I=1}^n N_I(\xi, \eta) y_I, \\ z(\xi, \eta) &= \sum_{I=1}^n N_I(\xi, \eta) z_I. \end{aligned} \quad (34)$$

2.3.1. Triangular Patches

We will present two alternatives as follows.

2.3.1.1. Side Degeneration

One possibility is to use the formulas previously used in quadrilateral patches, by degenerating one of the four sides into one point [1, p.3]. For example, when degenerating the entire side AD into a unique point A, the quadrilateral degenerates into a triangle ABC, (30) becomes [38, p.325]:

$$\begin{aligned} u(\xi, \eta) &= u_{BC}(\eta) \cdot \xi + u_{AB}(\xi) \cdot (1 - \eta) \\ &+ u_{AC}(\xi) \cdot \eta - \xi [(1 - \eta)u_B + \eta u_C] \end{aligned} \quad (35)$$

2.3.1.2. Barnhill's Formula

The mathematical description of triangular patches has been published by Barnhill [39-41]. In those works a "trilinearly blended interpolant" formulation is introduced, which is quite different than Gordon-Coons; today this formula can be found even in CAD textbooks such as [42, p.244]. The triangular patch shown in Fig. 4a can be mapped into the parametric patch of Fig. 4b through the relationship:

$$\begin{aligned} P(u, v, w) &= \frac{1}{2} \left[\frac{ug(v)}{1-v} + \frac{wh(1-v)}{1-v} + \frac{vh(w)}{1-w} \right. \\ &+ \frac{uf(1-w)}{1-w} + \frac{wf(u)}{1-u} + \frac{vg(1-u)}{1-u} \\ &\left. - wf(0) - ug(0) - vh(0) \right] \end{aligned} \quad (36)$$

The parametric patch of (36) is expressed as:

$$u + v + w = 1 \quad 0 \leq u \leq 1, \quad 0 \leq v \leq 1, \quad 0 \leq w \leq 1 \quad (37)$$

In this case, the parametric patch can be intersected with increasing values of the variables u, v that vary between 0 and 1, and then calculating the corresponding values of w from each set of u and v .

Apart from CAD applications, no application of Barnhill's formula to engineering analysis has been reported so far. For the first time, it is shown that, (36) inherently includes the conventional three- and six-noded triangular finite elements. Details are given in Appendix A and Appendix B, respectively. Of course, (36) allows for the derivation of novel arbitrary-noded triangular macroelements, that have been successfully applied to stress analysis problems [90].

Figure 4. Definition of the triangular parametric Barnhill's patch

2.4. Three-Dimensional (3D) Interpolation

Volume blocks are categorized into *hexahedral* (eight vertices, and twelve edges) and *tetrahedral* (four vertices, and six edges) blocks. The interpolations can treat either straight or curvilinear edges of the volume.

2.4.1. General

A boxlike region ABCDEFGH, shown in Fig. 5a, can easily be mapped to a unit cube in the rst parametric domain ($0 \leq r, s, t \leq 1$) shown in Fig. 5b. The relevant formula may be found in [1, p.41] and, as previously mentioned, is has been applied by Cook [43] for finite element mesh generation purposes. For more details the interested reader may also consult a textbook [44]. The only difference with a quadrilateral patch is that, here, six functions (associated to the six faces) are blended, instead of four functions (associated to the four boundary edges of the 2D patch). Below, the same formula is written in terms of projections.

Besides the above-mentioned $P_r\{\mathbf{x}\}$, $P_s\{\mathbf{x}\}$ and $P_{rs}\{\mathbf{x}\}$ used in the 2D problem, now, one-, two- and three-dimensional operators are further introduced as

follows:

$$\begin{aligned}
 P_t \{ \mathbf{x} \} &= \mathbf{x}(r, s, t_k) \cdot E_k(t) \\
 P_{st} \{ \mathbf{x} \} &= P_s P_t \{ \mathbf{x} \} = \mathbf{x}(r, s_j, t_k) \cdot E_j(s) \cdot E_k(t) \\
 P_{rt} \{ \mathbf{x} \} &= P_r P_t \{ \mathbf{x} \} = \mathbf{x}(r_i, s, t_k) \cdot E_i(r) \cdot E_k(t) \\
 P_{rst} \{ \mathbf{x} \} &= P_r P_s P_t \{ \mathbf{x} \} = \mathbf{x}(r_i, s_j, t_k) \cdot E_i(r) \cdot E_j(s) \cdot E_k(t)
 \end{aligned} \quad (38)$$

Again, summation over repeated indices is understood.

Having introduced the one-, two- and three-dimensional operators, then the following formula describes the interpolation of the co-ordinate vector [1,p.41]:

Figure 5. Three-dimensional superbrick macroelement analyzed using "boundary-only" Coons formulation

$$\begin{aligned}
 \mathbf{x}(r, s, t) &= -(-1)^1 \cdot (P_r + P_s + P_t) \{ \mathbf{x} \} \\
 &\quad - (-1)^2 \cdot (P_{rs} + P_{st} + P_{rt}) \{ \mathbf{x} \} \\
 &\quad - (-1)^3 \cdot P_{rst} \{ \mathbf{x} \}.
 \end{aligned} \quad (39)$$

2.4.2. Equivalent Expressions of Trivariate Coons' Formula

2.4.2.1. General Remarks

Equation (39) is generally applicable but it can be further simplified and be written in more readable expressions. So, in the case of a generalized curvilinear hexahedral /paralleloids (boxlike region), the geometry includes:

- Six surfaces, S
- Twelve edges, E , and
- Eight corners, C

Obviously, (39) includes all three quantities: Surfaces (S), edges (E) and corners (C). For convenience, the projections related to the S , E and C are denoted as follows:

$$\begin{aligned}
 S \{ \mathbf{x} \} &= (P_r + P_s + P_t) \{ \mathbf{x} \} \\
 E \{ \mathbf{x} \} &= (P_{rs} + P_{st} + P_{rt}) \{ \mathbf{x} \} \\
 C \{ \mathbf{x} \} &= P_{rst} \{ \mathbf{x} \}.
 \end{aligned} \quad (40)$$

However, in the case of adequately smooth and regular surfaces, the edges E can sufficiently describe S . In fact, applying (31) on the six surfaces of the superbrick, one can easily derive the following relationship:

$$S \{ \mathbf{x} \} + 3C \{ \mathbf{x} \} = 2E \{ \mathbf{x} \} \quad (41)$$

Then, substituting (40) and (41) into (39), one can finally derive three *equivalent* expressions of the three-dimensional Coons' interpolation formula, as follows:

$$\mathbf{x}(r, s, t) = S \{ \mathbf{x} \} - E \{ \mathbf{x} \} + C \{ \mathbf{x} \} \quad (42)$$

$$\mathbf{x}(r, s, t) = 1/2 S \{ \mathbf{x} \} - 1/2 C \{ \mathbf{x} \} \quad (43)$$

$$\mathbf{x}(r, s, t) = E \{ \mathbf{x} \} - 2C \{ \mathbf{x} \} \quad (44)$$

2.4.2.2. Comments on 3D Equivalent Expressions

Obviously, (42) is the most general expression because it includes any type of the surrounding surfaces S (edge-only noded and/or internally noded). Moreover, (43) is obtained by eliminating the edges (E) and it is based on the surface S , the last being corrected by the co-ordinates of the corners C . Finally, (44) includes only the twelve edges E and eight corners C , or in other words, the absolutely necessary data for the construction of a Coons' block made of Coons' surfaces; it is evident that (44) uses nodal points arranged along the twelve edges only.

Figure 6. Overall applicability of Coons-interpolation

In conclusion, in cases where the superbrick is sufficiently regular, (44) is the most advantageous. Using conventional notation, (44) becomes:

$$\begin{aligned}
 \mathbf{x}(r, s, t) &= (1-s)(1-t) \mathbf{x}(r, 0, 0) + r(1-s) \mathbf{x}(1, 0, t) \\
 &\quad + (1-s)t \mathbf{x}(r, 0, 1) + (1-r)(1-s) \mathbf{x}(0, 0, t) \\
 &\quad + s(1-t) \mathbf{x}(r, 1, 0) + rs \mathbf{x}(1, 1, t) \\
 &\quad + st \mathbf{x}(r, 1, 1) + (1-r)s \mathbf{x}(0, 1, t) \\
 &\quad + (1-r)(1-t) \mathbf{x}(0, s, 0) + (1-r)t \mathbf{x}(0, s, 1) \\
 &\quad + r(1-t) \mathbf{x}(1, s, 0) + rt \mathbf{x}(1, s, 1) \\
 &\quad - 2[(1-r)(1-s)(1-t) \mathbf{x}(0, 0, 0) + (1-r)(1-s)t \mathbf{x}(0, 0, 1) \\
 &\quad + (1-r)s(1-t) \mathbf{x}(0, 1, 0) + (1-r)st \mathbf{x}(0, 1, 1) \\
 &\quad + r(1-s)(1-t) \mathbf{x}(1, 0, 0) + r(1-s)t \mathbf{x}(1, 0, 1) \\
 &\quad + r(1-s)(1-t) \mathbf{x}(1, 1, 0) + rst \mathbf{x}(1, 1, 1)]
 \end{aligned} \quad (45)$$

2.4.3. The Three Alternative Uses of Coons Interpolation

In more detail, bivariate (2D) Coons interpolation on curvilinear surfaces is useful for (i) mesh generation for two different purposes (2D FEM or 3D BEM applications), and (ii) to derive 2D finite macroelements.

Also, trivariate (3D) Coons interpolation on curvilinear volumes is useful for (i) mesh generation (3D FEM applications), and (ii) to derive 3D finite macroelements, as shown in Fig.6.

It has been previously reported that, the mesh produced by 2D-Coons interpolation can be easily smoothed by taking the average of the *eight* neighbouring nodes. Also, the mesh produced by 3D-Coons interpolation can be easily smoothened by taking the average of *the twenty-six* neighbouring nodes [45-47].

3. Use of Coons' Interpolation

3.1. Older Findings

From the analysis sited in Section 2, it becomes clear that in 1967 Coons' interpolation was the first achievement in CAD surface theory, primarily aiming at describing surfaces or volumes [1]. In early 1970s, FEM analysts understood that the same interpolation might have a second usage, as a generator of *structured* finite element meshes [43, 48-54]. In more details, first a mesh generation is constructed on the parent unit square or cube for 2D and D problems, respectively. Second, nodal points are constructed along the boundaries of the virtual structure. Third, (30), or (39) and its variations, are applied and thus establish a mapping between the aforementioned nodes in the parent element and the virtual structure. As a result, the desired finite element mesh is generated.

Moreover, it was early understood that Coons interpolation could be used for engineering analysis purposes as well [55, 56].

3.2. Contemporary Findings

3.2.1. Contribution on Conventional Finite Element Meshes

In subsection 2.4.3 we mentioned that, due to the fact that the mesh generated using (30) or (39) for 2D and 3D problems, respectively, is usually of low quality, smoothing schemes have been successfully applied [57, 45-47]. The convergence leads to almost rectangular elements.

3.2.2. Construction of Gordon-Coons Macroelements for Engineering Analysis

3.2.2.1. Literature Survey

Early thoughts on the construction of large elements were made by the pioneer Bruce Irons [58]. Expressions of lower order large elements based on Coons-Gordon ideas are [55, 59-61]. Numerical examples on 2D potential problems were presented in 1989 by Kanarachos and Deriziotis [62] at the National Technical University of Athens (NTUA); as a member of NTUA's FEM/BEM group, the author can confirm that relevant attempts started therein in 1982. This work was extended to 2D elasticity [63]; in the relevant conference the author announced that the same methodology is not applicable only to Coons but also in conjunction with all other CAD formulations such as Gordon-Coons, Bézier, B-splines and NURBS.

Boundary-only Coons interpolation was applied to 2D

potential problems [37, 64-68], axisymmetric elastostatics [69], and 2D elastodynamics [70-73]. In the process of the research it was found that boundary-only Coons interpolation corresponds to a series expansion of the solution with those terms appearing in Pascal's triangle with a surplus of two (Serendipity family), a fact that was later found as a note in [74].

In some examples, the boundary-only Coons formulation was more accurate not only than the conventional FEM, with the same mesh density, but also than BEM [75].

It was later understood that, it is not always possible to accurately solve all engineering problems unless a sufficient number of internal nodes are used. The latter is easily achieved using Gordon-Coons (transfinite) interpolation [35-37, 72, 76]. Boundary-only Coons interpolation was successfully applied to 3D potential and elasticity problems [77-79]. The theme definitely closes with [80], which compares twelve alternative models one another; although it focuses on 2D elastostatics, the conclusions are of general importance.

Concerning BEM and solid modeling, initial attempts for CAD/CAE integration were made by Casale [81-83]. Application of Coons interpolation in 3D BEM problems was presented during the years 2001-2003 [84-87]; in these conferences the general applicability in conjunction with Bézier, B-splines and NURBS was repeated.

The overall conclusion is that boundary-only formulation sometimes cannot converge to the accurate solution, whereas the Gordon-Coons [88] always achieves it [89].

Concerning structures of triangular shape, the work of Kanarachos and Dimitriou [38, 89] has shown reliable results when a side of the quadrilateral is degenerated. In addition to the latter works, Provatidis and Antoniou [90] have found encouraging results when Barnhill's interpolation [39-41] is used for the engineering analysis.

As one can see inspecting [91-109], Pierre Bézier was an employee of Renault (in France) and started publishing since 1966, which is almost the same period in which Steven Coons was working at MIT (in USA). Bézier's theory is related to the discovery of the "control points", which are vertices of a polygon (generator) that controls a *curve*. The same theory is also extended to curved *surfaces*, using a set of generators [19, 27]. Extension to *volumes* is possible [110]. In the context of engineering analysis, it has been found that Bézier formulation is equivalent to the use of monomials or Lagrange polynomials [111], a matter that has been also treated in 1D problems [32, 33, 112].

Later, rational Bézier curves [113-115], which have several advantages over polynomial Bézier curves, appeared. Clearly, rational Bézier curves provide more control over the shape of a curve than does a polynomial Bézier curve. In addition, a perspective drawing of a 3D Bézier curve (polynomial or rational) is a rational Bézier curve, not a polynomial Bézier curve. Also, rational Bézier curves are needed to exactly express all conic sections. A degree two polynomial Bézier curve can only represent a *parabola*, in a similar way with the Lagrangian polynomials. Exact

representation of *circles* requires rational degree two Bézier curves. Rational Bézier curves permit some additional shape control: by changing the weights, you change the shape of the curve. It is possible to reparametrize the curve by simply changing the weights in a specific manner.

Finally, in the process of CAD evolution, the use of B-Splines was proposed in 1973-1974 [116,117]. It is reminded that B-splines is nothing different than usual splines but they are replaced by non-cardinal basis functions, $N_{i,p}(\xi)$ of polynomial degree p , which are calculated for every normalized coordinate, ξ , through fast iterative formulas developed independently by Cox [29] and DeBoor [30] in 1972, as mentioned by the latter researcher in his monograph [26]. B-splines appear a local support and include, as a special case, the abovementioned Bézier curves. Despite the local support of B-splines, a better control on regional changes is achieved when using the non-uniform rational B-splines (NURBS) that were developed in mid-seventies till early eighties [118,119]. The interested reader may also consult a later survey [120] and a useful monograph of 646 pages [27].

According to the abovementioned sequence in CAD evolution, i.e. Gordon-Coons $\rightarrow$ Bézier $\rightarrow$ B-splines $\rightarrow$ NURBS, a relevant contribution in structural (generally in engineering) analysis is anticipated, as follows. The common characteristic of all relevant works in engineering analysis is that, they all aim at decomposing the domain into CAD-controlled subregions [121-124].

Concerning *Gordon-Coons* (transfinite) interpolation, relevant research on graphics continued at least until 2004 [125-128], whereas engineering analysis was made in electrical engineering [129] and soft-tissue biomechanical [130] applications. Initial works are [55, 59-61]. As regards the FEM/BEM group at NTUA, *isoparametric* Gordon-Coons macroelements were published in the period 1989-2011 [38, 57, 62-73, 75-80, 84-90].

In this paper we will use the term “*isogeometric*” when referring to interpolations related to “control points”, i.e. Bézier, B-splines and NURBS.

Concerning Bézier representation, it is reminded that it is equivalent with the use of Lagrange polynomials [111]. In other words, for any selected degree of the polynomials involved, despite the fact that the control points sweep the entire domain, in fact it is “*as if nodal points along only the sides*” were used (2D: 4 sides of the quadrilateral, 3D: 12 edges of the hexahedron).

Concerning *B-splines* in engineering analysis, a forgotten but (in author’s opinion) pioneering work is due to Aristodemo [131] and later a lot of papers appeared [132-141]. Among these works, [137-141] refer to the boundary element method (BEM).

Concerning the evolution in NURBS, the interested reader may consult [142], among others. One of the first attempts to incorporate NURBS in structural analysis, and particularly in shape optimization, is due to Schramm and Pilkey [143], in 1993, followed by other investigators such as [144-146]. In 2005, the group of Prof. T. J. R. Hughes introduced the term

“*isogeometric*” elements [147-149], and later they applied an efficient quadrature to reduce the CPU-time [150].

The author [68] had timely discovered that, despite its elegance, the global interpolation is generally slower than the common finite element method (Galerkin-Ritz), of course for the same mesh density. In 2005, he proposed to preserve the CAD-based global interpolation and replace the domain integration through a global collocation scheme [73, p. 6704].

Concerning the Global Collocation Method (GCM), an initial contribution is due to Blerk and Botha [151]. The problem was systematically tackled starting from one-dimensional problems, first starting with Lagrange polynomials [22, 23, 152] and then continuing with B-splines [153, 154]. Then, the same idea was extended to 2D potential problems [155, 156], elastostatics [157], as well as eigenvalue problems in acoustics and elastodynamics [158, 159]. It is worth-mentioning that the application of *isogeometric* collocation methods was later proposed by others [160].

3.2.2.2. Global Shape Functions

Following [80], let us assume that the sides AB, BC, CD and DA of the patch ABCD shown in Fig. 3 include q_1, q_2, q_3 and q_4 nodes, respectively. In this case, the corresponding number of subdivisions per side are $n_1=q_1-1$, $n_2=q_2-1$, $n_3=q_3-1$ and $n_4=q_4-1$, respectively. Then, the number of nodes along the boundary of the patch becomes:

$$q_b = n_1 + n_2 + n_3 + n_4 = q_1 + q_2 + q_3 + q_4 - 4 \quad (46)$$

while considering the additional n_i internal nodes, the total number of nodes becomes:

$$q_e = q_b + n_i \quad (47)$$

In the sequence, the boundary values of the displacement vector, i.e. $\mathbf{u}(\xi, 0)$, $\mathbf{u}(1, \eta)$, $\mathbf{u}(\xi, 1)$ and $\mathbf{u}(0, \eta)$ in (30) or finally in (32), are interpolated by any set of basis functions $B_j(\hat{\xi})$ like those presented in Section 2 ($\hat{\xi}$ is either ξ or η ; the upper index in B_j below corresponds to the relevant side of the Coons patch):

$$\begin{aligned} \text{Side AB: } \mathbf{u}(\xi, 0) &= \sum_{j=1}^{q_1} B_j^{\text{AB}}(\xi) \mathbf{u}(\xi_j, 0) \\ \text{Side BC: } \mathbf{u}(1, \eta) &= \sum_{j=1}^{q_2} B_j^{\text{BC}}(\eta) \mathbf{u}(1, \eta_j) \\ \text{Side CD: } \mathbf{u}(\xi, 1) &= \sum_{j=1}^{q_3} B_j^{\text{CD}}(\xi) \mathbf{u}(\xi_j, 1) \\ \text{Side DA: } \mathbf{u}(0, \eta) &= \sum_{j=1}^{q_4} B_j^{\text{DA}}(\eta) \mathbf{u}(0, \eta_j) \end{aligned} \quad (48)$$

Typical boundary approximation in the above basis functions $B_j(\hat{\xi})$ might consist of (i) piecewise linear interpolation (kodei = 1), (ii) B-spline interpolation (kodei = 2), (iii) Lagrange polynomial (kodei = 3), (iv) Bernstein

(Bézier) polynomials (kodei = 4), and (v) NURBS (kodei = 5). The meaning of associate code “i” will be explained in subsection 3.2.2.3.

Similar interpolations are also considered along the inter-boundaries (if any). Finally, if all these interpolations are substituted into (48), and subsequently into (30) or (32), the solution $\mathbf{u}(\xi, \eta)$ is approximated by global cardinal shape functions $N_i(\xi, \eta)$ within the Coons-patch, as follows:

$$\mathbf{u}(\xi, \eta) = \sum_{j=1}^{q_i} N_j(\xi, \eta) \mathbf{u}_j, \quad (49)$$

with $\mathbf{u}_j$ denoting the displacement vector at nodal point ‘j’, appearing at the boundaries or/and the interior of the Coons macro-element. In case of Bézier B-splines or NURBS, $\mathbf{u}_j$ in (49) should be replaced by $\mathbf{a}_j$.

For example, the global shape function associated with the corner node A is given by:

$$N_A(\xi, \eta) = E_0(\xi) B_{q_A}^{DA}(\eta) + E_0(\eta) B_1^{AB}(\xi) - E_0(\xi) E_0(\eta). \quad (50)$$

Also, the global shape function associated with an intermediate node along the side AB is given by:

$$N_i(\xi, \eta) = E_0(\eta) B_i^{AB}(\xi), \quad i = 2, \dots, q_1 - 1. \quad (51)$$

Finally, the global shape function associated with an internal node is given by:

$$N_j(\xi, \eta) = E_k(\xi) E_l(\eta), \quad (52)$$

$$k = 1, \dots, n_\xi - 1; l = 1, \dots, n_\eta - 1$$

with the subscript ‘j’ denoting the jth node having normalized coordinates (ξ_k, η_l) .

3.2.2.3. The Role of Blending Functions

In addition to the index “i” that determines the basis functions along each of the four sides of a patch ABCD, now we deal with the type of the blending functions.

When the blending functions are linear functions influencing the entire side: $E_0(\xi) = 1 - \xi$ and $E_1(\xi) = \xi$, we assign them the kodej = 0. This case corresponds to the “boundary-only” Coons macroelement, under the notation “Ci0”. Again, “Ci0” means that all four sides are interpolated with the same set of basis functions, i.e. $i = 1$ for piecewise linear, $i = 2$ for B-splines, $i = 3$ for Lagrange polynomials, $i = 4$ for Bernstein (Bézier) polynomials, and $i = 5$ for NURBS.

In addition to the above linear $E_0(\xi)$ and $E_1(\xi)$, the blending functions $E_k(\xi)$ and $E_l(\eta)$, $k = 1, \dots, n_\xi - 1$; $l = 1, \dots, n_\eta - 1$, can be any reasonable cardinal functions that obey the delta-Kronecker condition: $E_i(\xi_j) = \delta_{ij}$. In more details, they have to vanish along the opposite boundary as well as along those interboundaries to which they are not associated. Typical blending functions are (i) linear functions for the entire side: $E_0(\xi) = 1 - \xi$ and $E_1(\xi) = \xi$ (kodej = 0), (ii) the hat (piecewise linear)

functions with compact support between two adjacent nodes (kodej = 1), (iii) cardinal cubic B-splines (kodej = 2), and (iv) Lagrange polynomials (kodej = 3).

Based on the above definitions and code pairs (i, j), we can construct $5 \times 4 = 20$ macroelement models, denoted by “Cij”, $i = 1, 2, 3, 4$, and 5, whereas $j = 0, 1, 2, 3$, and 4 (C stands for “Coons”). Again, the first index, i, refers to the type of the basis functions, whereas the second index, j, refers to the type of blending functions.

It is easy to understand that the number of possible models Cij can be highly increased when some of the sides have a different set of basis functions than what another side has.

Moreover, if opposite sides have the same number of nodal points ($n_1 = n_3 = n_\xi$, $n_2 = n_4 = n_\eta$) and all nodes (boundary and internal ones) are distributed along lines of constant ξ - and η -values. Then:

- When using *piecewise-linear* interpolation for both the basis ($B_j(\xi)$) and the blending functions ($E_i(\xi)$), (50) up to (52) produce a mesh of $n_\xi \times n_\eta$ conventional four-node bilinear finite elements. In fact, considering the compact support of the piecewise-linear functions, the proof is quite trivial.
- When using *Lagrange polynomials* for both the basis and the blending functions, (31) and (32) degenerate to $P_{\xi\eta}(\mathbf{x})$ and $P_{\xi\eta}(\mathbf{u})$, respectively. In other words, the shape functions become the well-known tensor products of Lagrange polynomials. For a proof we refer to Reference [37].

4. Isoparametric and “Isogeometric-like” Approximations

Gordon-Coons and Lagrangian type elements are based on the following series expansion:

$$\mathbf{x}(\xi, \eta) = \sum_{j=0}^n N_j(\xi, \eta) \cdot \mathbf{x}_j, \quad (53)$$

$$U(\xi, \eta) = \sum_{j=0}^n N_j(\xi, \eta) \cdot U_j$$

which characterizes all “isoparametric” elements.

In contrast, “isogeometric-like” approximations (i.e., tensor products of Bézier, B-splines and NURBS) are based on the following series expansion:

$$\mathbf{x}(\xi, \eta) = \sum_{j=0}^n N_j(\xi, \eta) \cdot \mathbf{x}_j, \quad (54)$$

$$U(\xi, \eta) = \sum_{j=0}^n N_j(\xi, \eta) \cdot a_j$$

where a_j are *not* generally the nodal values at the control points but generalized coefficients. Only at the very ends of the boundary the extreme control points ($\mathbf{P}_0$ and $\mathbf{P}_n$) coincide

with the corresponding ends of generator curves and, therefore, the coefficients a_0 and a_n coincide with the values U_0 and U_n , respectively.

It is remarkable that in all “isogeometric-like” approximations the key problem is to impose the boundary conditions. This difficulty is not apparent in 1D problem, but it appears in 2D and 3D ones. For example, if the entire side AB of the quadrilateral ABCD is given homogeneous Dirichlet conditions ($U = 0$), this implies that all coefficients a_j in (54) must vanish. In contrast, when half of the side AB is given the zero value whereas the rest is given for example by $U = \xi - 1/2$, we cannot ‘a-priori’ calculated these coefficients, unless we fulfill the condition at so many points as the number of control points along AB, and then invert the produced matrix. In this context, according to Natekar et al. [124], bivariate NURBS representation is applied to derive shape functions $N_I(\xi, \eta)$ that are based on the set of I -th control points defining the system geometry. The same shape functions are also used to approximate the displacement field within the domain. It is remarkable that at any point within the domain the sum of these shape functions equals to the unity but the value of $N_I(\xi, \eta)$ at the I -th control point is not unity to that node. Therefore, “even if the control point were to be coincident with the location of the boundary condition, direct application of the boundary condition is not possible since the specified field value will be distributed to control points influencing the point under consideration” [124].

The abovementioned difficulties were the reason that the author preferred to work mainly with Coons interpolation (although very early he suggested the application of NURBS [45, p.1 and pp. 32-33], while other research teams directly followed the tendency of the modern CAD systems and worked with NURBS [18], among others.

5. Computational Procedure

5.1. General

The vibration of an elastic solid structure is described through the stress equilibrium equation [5]:

$$\mathbf{L}^T \boldsymbol{\sigma} + \mathbf{f} = \rho \ddot{\mathbf{u}} \quad (55)$$

while for acoustical cavities (or membranes) it holds:

$$\left(\frac{1}{c^2} \right) \cdot \frac{\partial^2 u}{\partial t^2} - \nabla^2 u = 0 \quad (56)$$

5.2. Numerical Procedure

In this work we consider three numerical procedures: (i) *Galerkin/Ritz*, (ii) *Collocation*, and (iii) *Boundary Element Method*, as follows.

5.2.1. Galerkin/Ritz

Applying the Galerkin method to (55) or (56), one obtains the well-known matrix formulation [5]:

$$[\mathbf{M}] \{\ddot{\mathbf{a}}(t)\} + [\mathbf{K}] \{\mathbf{a}(t)\} = \{\mathbf{f}(t)\} \quad (57)$$

where $[\mathbf{M}]$ and $[\mathbf{K}]$ are the mass and stiffness matrices, respectively, which are given by:

$$m_{ij} = \begin{cases} \int_{\Omega} \mathbf{N}^T \rho \mathbf{N} d\Omega, & \text{in elasticity} \\ \frac{1}{c^2} \int_{\Omega} N_i N_j d\Omega, & \text{in acoustics} \end{cases} \quad (58)$$

$$k_{ij} = \begin{cases} \int_{\Omega} (\mathbf{L} \mathbf{N})^T \mathbf{E} (\mathbf{L} \mathbf{N}) d\Omega, & \text{in elasticity} \\ \int_{\Omega} \nabla N_i \nabla N_j d\Omega, & \text{in acoustics} \end{cases}$$

where N_i denote the shape functions by which the solution (displacement component or acoustic pressure) is interpolated as follows:

$$u(\mathbf{x}, t) = \sum_{i=1}^{n_e} N_i(\mathbf{x}) \cdot a_i(t), \quad (59)$$

where n_e is the number of DOF per element.

Also, $\{\mathbf{f}(t)\}$ is the force vector and $\{\mathbf{a}(t)\}$ denotes the vector of degrees of freedom, i.e. it consists of either the displacement components, u , or the generalized coefficients in which the solution is expanded.

5.2.2. Collocation

Using the same global shape function as above (section 4.1), and fulfilling the PDE (55) or (56), one obtains a similar matrix formulation with (57) with the difference that now *no domain integral appears*, that is:

$$m_{ij} = \begin{cases} \rho N_j(x_i), & \text{in elasticity} \\ \left(\frac{1}{c^2} \right) N_j(x_i), & \text{in acoustics} \end{cases} \quad (60)$$

$$k_{ij} = \begin{cases} -\mathbf{L}^T N_j(x_i) \cdot \mathbf{E} \cdot \mathbf{L} N_j(x_i), & \text{in elasticity} \\ -\nabla^2 N_j(x_i), & \text{in acoustics} \end{cases}$$

where ‘ i ’ corresponds to the collocation points and ‘ j ’ to the control points.

5.2.3. Boundary Element Method (BEM)

In the case of elasticity problems, the co-ordinate vector within the l -th patch, possessing q_l nodes, is interpolated on the basis of the boundaries of the patch as follows:

$$\mathbf{x}(r, s) = \sum_{j=1}^{q_l} N_j(r, s) \mathbf{x}_i^j = \mathbf{N} \mathbf{x} \quad (61)$$

Since each patch is considered as an isoparametric element it

holds that:

$$\begin{aligned} \mathbf{u}(r, s) &= \sum_{j=1}^{q_l} N_j(r, s) \mathbf{u}_i^j = \mathbf{N} \mathbf{u} \\ \mathbf{p}(r, s) &= \sum_{j=1}^{q_l} N_j(r, s) \mathbf{p}_i^j = \mathbf{N} \mathbf{p} \end{aligned} \quad (62)$$

By substituting (62) in the usual integral equation (see for example [7]) and summarizing over the N_p patches in which the boundary is divided, one obtains:

$$\begin{aligned} c^i \mathbf{u}^i + \sum_{ip=1}^{N_p} \left\{ \oint_{\Gamma_{ip}} \mathbf{p}_{lk}^* \mathbf{N}_k d\Gamma \right\} \mathbf{u}_{ip} \\ = \sum_{ip=1}^{N_p} \left\{ \oint_{\Gamma_{ip}} \mathbf{u}_{lk}^* \mathbf{N}_k d\Gamma \right\} \mathbf{p}_{ip} \end{aligned} \quad (63)$$

In (63) the infinitesimal area $d\Gamma$ is given by:

$$d\Gamma = |G(r, s)| dr ds \quad (64)$$

where the Jacobian is calculated as:

$$\begin{aligned} |G(r, s)| &= (g_1^2 + g_2^2 + g_3^2)^{1/2} \\ \text{where} \\ g_1 &= \frac{\partial x_2}{\partial r} \frac{\partial x_3}{\partial s} - \frac{\partial x_2}{\partial s} \frac{\partial x_3}{\partial r}, \\ g_2 &= \frac{\partial x_1}{\partial r} \frac{\partial x_3}{\partial s} - \frac{\partial x_1}{\partial s} \frac{\partial x_3}{\partial r}, \\ g_3 &= \frac{\partial x_1}{\partial r} \frac{\partial x_2}{\partial s} - \frac{\partial x_1}{\partial s} \frac{\partial x_2}{\partial r} \end{aligned} \quad (65)$$

Now, for the purposes of the numerical integration only, the patch is divided into $N_r \times N_s$ cells where a second set of normalized co-ordinates $(-1 \leq r', s' \leq 1)$ is introduced [84]. Therefore, the term $|G(r, s)| dr ds$ in (64) is replaced by $|G(r, s) \cdot G(r', s')| dr' ds'$, which requires a trivial (e.g., 2×2 , 3×3 , 4×4) Gaussian quadrature. A selective integration scheme has been recently developed.

Therefore, the final algebraic system obtains the form:

$$\mathbf{C} \mathbf{U} + \sum_{ip=1}^{N_p} \hat{\mathbf{H}}^{ip} \mathbf{U}^{ip} = \sum_{ip=1}^{N_p} \mathbf{G}^{ip} \mathbf{P}^{ip} \quad (66)$$

where $\mathbf{U}$ is the displacement vector of all nodes on the boundary of the structure (along the patch edges), $\mathbf{U}^{ip}$ and $\mathbf{P}^{ip}$ are displacement and traction vectors referring to the ip -th patch as well as $\mathbf{H}^{ip}$ and $\mathbf{G}^{ip}$ are the nonsymmetric influence matrices.

By properly assembling the submatrices in (66) we obtain:

$$\mathbf{H} \mathbf{U} = \mathbf{G} \mathbf{P} \quad (67)$$

If q is the total number of nodes along all edges of the patches on the boundary of the structure, the dimensions of

the vectors and matrices in (67) are as follows:

- $\mathbf{U}$: displacement vector ($3q \times 1$)
- $\mathbf{P}$: traction vector ($3\bar{q} \times 1$)
- $\mathbf{H}$: total displacement-influence matrix ($3q \times 3q$)
- $\mathbf{G}$: total traction-influence matrix ($3q \times 3\bar{q}$)

The above symbol $\bar{q}$ is larger than q ($\bar{q} = q + \Delta q$) with Δq depending on the number of sharp corners and their multiplicity in traction discontinuity [7].

The above static analysis described by (67) can be also extended to the solution of dynamic problems too. Briefly, using a set of radial basis functions a mass matrix is constructed for the entire structure and it is combined with the static matrices $\mathbf{H}$ and $\mathbf{G}$ shown in (67) [86, 87].

6. Numerical Results

In addition to previous results, in this section we present additional results that elucidate the computational performance of the proposed techniques.

Example: Eigenvalues of a fixed rectangular membrane

The dimensions of the membrane in the x - and y -directions are $a \times b = 2.5 \times 1.1$ m, the wave velocity is taken as $c = 1$ m/s, whereas the entire boundary is considered to be fixed. The free vibrations are governed by (56) and the exact eigenvalues are given by:

$$\omega_{ij}^2 = \pi^2 c^2 \left(\frac{i^2}{a^2} + \frac{j^2}{b^2} \right), \quad i, j = 1, 2, \dots, \quad (68)$$

The rectangular domain, 2.5×1.1 , is uniformly divided into $n_x \times n_y$ subdivisions, preserving the same ratio, i.e. $n_x \times n_y = 4 \times 2, 6 \times 3, 8 \times 4, 10 \times 5, 12 \times 6, 14 \times 7$, and 16×8 , thus leading to $(n_x+1)(n_y+1) = 15, 28, 45, 66, 91, 120$, and 153 nodal points, respectively. In all methods the number of unknowns is the same, whereas $2(n_x+n_y) = 12, 18, 24, 30, 36, 42$, and 48 nodal points belong to the boundary and have to be eliminated. Thus, in the particular case of *Dirichlet* problem, the number of unknowns is common for all the three methods as follows:

- Conventional finite element (**FEM**) using four-node bilinear elements.
- (Gordon)-Coons Patch Macroelement (**CPM**), in conjunction with the Galerkin-Ritz formulation (domain integration), according to (58).
- The abovementioned Gordon-Coons approximation, in conjunction with the Global Collocation Method (**GCM**), that is without domain integration, according to (60).

The computational error was determined by the formula $\left(\left| \omega_{calculated}^2 - \omega_{exact}^2 \right| / \omega_{exact}^2 \right) \times 100$, in percent (%), and the quality of the numerical solution is illustrated in Fig.7, where one can notice that the proposed CPM and GCM methods are far superior to the conventional FEM. Although CPM is the overall best it needs high computer time, whereas the GCM is much more efficient because it needs no domain integration at all.

Figure 7. Convergence quality of the proposed Coons Patch Macroelement (CPM) and Global Collocation Method (GCM) versus conventional finite element method (FEM), for the first four eigenvalues (ω^2) of a rectangular membrane $a \times b = 2.5\text{m} \times 1.1\text{m}$, $c = 1 \text{ m/s}$.

Figure 8(a) and (b). Conventional finite element meshes versus (b) Lagrange type family and (d) transfinite Coons-Patch macroelement.

7. Discussion

It was shown that all five CAD interpolations, i.e. Coons, Gordon, Bézier, B-splines and NURBS are capable of producing global shape functions that influence the entire patch of volume, thus constructing reliable macroelements that can deal with a large portion (subregion) of a structure.

The great amount of relevant work performed by the FEM/BEM group at NTUA started in 1982 but it was published for the first time in 1989 [62]. It was initially based on the nodal point-based variable u , using *natural* cardinal (of [1,0]-type) cubic B-splines of Schoenberg's style without compact support (see, for example, [66] and a graph in [69, p.541]). This choice was rather in contrast to the mainstream

investigations by other researchers, who used to deal with generalized coefficients, a . Here is the proper point to clarify that, the motivation of using Schoenberg's formula is the fact that the latter deals directly with the nodal values u themselves, and not with arbitrary coefficients α related to the control points. However, in the particular case of cubic interpolation, the *natural* B-splines (used in [62-68]) lack the cubic monomial x^3 , a fact that can be technically recovered when using the two rotations at the ends of every of the four boundary sides [80, p. 958].

Later, it was decided to leave cardinal B-splines and work in conjunction with *Lagrange polynomial* interpolation along each of the four sides of the Coons' patch [89, 37, 72, 73], among others. Lagrange polynomials have the advantage that they are directly associated to the nodal values. In this case, the results were extremely accurate and the convergence was fine, particularly when the exact solution was a polynomial. As shown elsewhere [66, 77], all conventional finite elements of the Lagrange family can be produced by the Gordon-Coons interpolation, whereas all conventional finite elements of the Serendipity family are produced by the "boundary-only" Coons interpolation.

Moreover, it was found that the use of Lagrange polynomials, again directly referring to the nodal values u , is equivalent with the use of Bézier (Bernstein) polynomials [111]. Of course, since modern B-splines based on DeBoor's formula are by-definition equivalent with those of Schoenberg [28], all work done so far by the NTUA's group could be repeated in terms of univariate DeBoor's B-splines, in which the matrix inversion is bypassed and also local support is achieved. We stress reader's attention to the fact that in this paragraph we referred to univariate B-splines interpolation along the four boundaries of the quadrilateral patch ABCD, as well as any interboundaries if present.

The advantage of using Gordon-Coons macroelements with internal nodes with respect to conventional Lagrangian type elements, is related to the capability of dealing with patches having a *different* discretization to opposite sides of the patch. An illustrative example is shown in Fig. 8d (note that the opposite sides AB and CD consist of 8 and 16 subdivisions, respectively, whereas the internal nodes form their mean average, equal to 12). Therefore, it is not necessary to uniformly divide each side whereas the internal nodes are not obliged all to possess the same position in horizontal and vertical lines. For other examples we refer to [37, 72, 80]. A second advantage is the capability of Gordon-Coons interpolation to deal with dissimilar patches, which can be easily merged [37, pp. 330-331; 88]. Obviously, three-dimensional problems are analogous with two-dimensional ones.

In this review paper we also extended the Gordon-Coons models previously presented in [80]. We have revealed that "playing" with both the basis and the blending functions we can develop a large number of alternative Coons-elements, some of which (not all of them) have been tested in [80]. It is quite interesting that even the classical four-node (bilinear) element can be produced using hat functions for both the

basis and the blending functions.

Independently of the model choice, global interpolation causes a high CPU-time, sometimes higher than the usual finite element solution [68, 150]. The latter has kept researchers busy for a long time. Provatidis [73, p. 6704] was the first who realized the need for replacing the domain integration of the Galerkin-Ritz by a Global Collocation scheme where the partial differential equation is satisfied at Gauss points, taken for the entire macroelement. In this way the overall CPU-time is highly reduced.

While Galerkin-Ritz can work to complex regions, such as L-shaped or even Pi-shaped, in conjunction with piecewise linear interpolation, the latter is not possible for the GCM (it works but leads to very poor results). Lagrange polynomials and B-splines interpolation do not work along an L-shaped line because they lead to an interpolant that extremely deviates from the actual line. Therefore, the domain decomposition becomes quite necessary.

Remaining by the Global Collocation Method (GCM), our numerical experience is that the domain decomposition and the necessary coupling between adjacent macroelements do not immediately lead to acceptable results, unless an adequate number of collocation per direction is considered. A general characteristic of the Global Collocation Method is that the degrees of freedom possessing a Neumann boundary condition, or is at the interface between two adjacent macroelements, are finally eliminated and thus decrease the dimension of the system that has to be solved.

There is no doubt that the most efficient way to accurately approximate the geometry variations of an object or a mechanical component is NURBS. For example, in order to approximate the surface of an ideal sphere, it becomes necessary to divide it into at least six equal Coons patches. Similar difficulties have forced some companies to abandon Coons interpolation and replace it by NURBS [161].

8. Conclusions

It was shown that all the well-known interpolations that are used for the analytical description of CAD surfaces or volumes, can become the "vehicle" to derive global shape or basis functions and thus to perform engineering analysis. The only restriction is that the subregions in which the entire structure has to be subdivided should be of regular shape that can be mapped to a square, a triangle, a cube, and so on. Particularly in dynamic analysis, since the approximation of the variable is continuous in a large area or volume, the produced modes and the relevant natural frequencies are of excellent quality. The advantage of the CAD-based approach is that the degrees of freedom are associated to points that are quite necessary to determine the geometry, a fact that enables shape optimization to be efficiently performed. Concluding, all CAD-based global approximations can produce the basis functions that can be involved in any usual computational method, such as the finite element method, the boundary element method and several global collocation schemes.

Acknowledgements

This work was partially supported by the Committee of Basic Research at NTUA, Greece, through the Project: Protagoras: ΔΙΔΥΜΟ (ΔΙΕΡΕΥΝΗΣΗ ΤΗΣ ΔΥΝΑΤΟΤΗΤΑΣ ΣΥΖΕΥΞΗΣ ΤΩΝ ΜΕΘΟΔΟΛΟΓΙΩΝ CAD/CAE), EDEIL #65/1388 (Investigation on CAD/CAE integration, 1 June 2004 – 28 February 2007). The proposal was submitted in May 2003. In addition, I acknowledge the PhD students who tried to deal with this matter since early 1980s at NTUA, as well as the undergraduate students who conducting their Diploma Work or MSc Thesis on this topic under my supervision. I have to point out that, primarily due to insufficient funding, and secondarily because this subject requires full concentration and dedication as well as excellent programming skills, some doctoral students discontinued and therefore the progress was slow.

Appendix A

Barnhill's interpolation to derive the linear shape functions of triangular element

Figure 8. Barnhill's formula applied to derive conventional triangular elements for (a) linear and (b) quadratic interpolation.

Let us consider the triangular element $ABC \equiv 123$, as shown in Fig. 8a. Assuming the variation of the variable, F , along every side to be *linear*, the boundary terms shown in Fig. 4 can be written as:

$$f(u) \triangleq F_{AB} = (1-u)F_A + uF_B \quad (A-1)$$

$$g(v) \triangleq F_{BC} = (1-v)F_B + vF_C \quad (A-2)$$

$$h(w) \triangleq F_{CA} = (1-w)F_C + wF_A \quad (A-3)$$

Substituting (A-1), (A-2) and (A-3) into (36), one obtains:

$$\begin{aligned} F(u, v, w) &= \frac{1}{2} \left[\frac{u[(1-v)F_B + vF_C]}{1-v} + \frac{w[(1-(1-v))F_C + (1-v)F_A]}{1-v} + \frac{v[(1-w)F_C + wF_A]}{1-w} \right. \\ &\quad \left. + \frac{u[(1-(1-w))F_A + (1-w)F_B]}{1-w} + \frac{w[(1-u)F_A + uF_B]}{1-u} + \frac{v[(1-(1-u))F_B + (1-u)F_C]}{1-u} \right. \\ &\quad \left. - wF_A - uF_B - vF_C \right] \\ &= \frac{1}{2} \left[\frac{u[(1-v)F_B + vF_C]}{1-v} + \frac{w[vF_C + (1-v)F_A]}{1-v} + \frac{v[(1-w)F_C + wF_A]}{1-w} \right. \\ &\quad \left. + \frac{u[wF_A + (1-w)F_B]}{1-w} + \frac{w[(1-u)F_A + uF_B]}{1-u} + \frac{v[uF_B + (1-u)F_C]}{1-u} \right. \\ &\quad \left. - wF_A - uF_B - vF_C \right] \\ &= \frac{1}{2} \left[uF_B + uv/(1-v)F_C + wv/(1-v)F_C + wF_A + vF_C + wv/(1-w)F_A \right. \\ &\quad \left. + uw/(1-w)F_A + uF_B + wF_A + wu/(1-u)F_B + vu/(1-u)F_B + vF_C \right. \\ &\quad \left. - wF_A - uF_B - vF_C \right] \end{aligned} \quad (A-4)$$

After reduction, and taking into consideration (37), i.e. $u + v + w = 1$, (A-4) leads to the well-known formula:

$$F(u, v, w) = wF_A + uF_B + vF_C \quad (\text{A-5})$$

Appendix B

Barnhill's interpolation to derive the quadratic shape functions of triangular element

Let us consider the triangular element ABC $\equiv$ 123456, as shown in Fig. 8b. Since the variation of the variable, F , along every side, S ($= AB, BC, CA$), is *quadratic*, we introduce the Lagrange polynomials:

$$\begin{aligned} L_S^1(\xi) &= (2\xi - 1)(\xi - 1), \quad L_S^2(\xi) = \\ &4\xi(1 - \xi), \quad L_S^3(\xi) = \xi(2\xi - 1) \end{aligned} \quad (\text{B-1})$$

Based on (B-1) the boundary terms shown in Fig. 4 can be written in terms of Lagrange polynomials as:

$$f(u) \triangleq F_{AB} = L_{AB}^1(u)F_1 + L_{AB}^2(u)F_4 + L_{AB}^3(u)F_2 \quad (\text{B-2})$$

$$g(v) \triangleq F_{BC} = L_{BC}^1(v)F_2 + L_{BC}^2(v)F_5 + L_{BC}^3(v)F_3 \quad (\text{B-3})$$

$$h(w) \triangleq F_{CA} = L_{CA}^1(w)F_3 + L_{CA}^2(w)F_6 + L_{CA}^3(w)F_1 \quad (\text{B-4})$$

Substituting (B-2), (B-3) and (B-4) into (36), one obtains:

$$\begin{aligned} F(u, v, w) &= \frac{1}{2} \left\{ \frac{u[L_{BC}^1(v)F_2 + L_{BC}^2(v)F_5 + L_{BC}^3(v)F_3]}{1 - v} \right. \\ &+ \frac{w[L_{CA}^1(1 - v)F_3 + L_{CA}^2(1 - v)F_6 + L_{CA}^3(1 - v)F_1]}{1 - v} \\ &+ \frac{v[L_{CA}^1(w)F_3 + L_{CA}^2(w)F_6 + L_{CA}^3(w)F_1]}{1 - w} \\ &+ \frac{u[L_{AB}^1(1 - w)F_1 + L_{AB}^2(1 - w)F_4 + L_{AB}^3(1 - w)F_2]}{1 - w} \\ &+ \frac{w[L_{AB}^1(u)F_1 + L_{AB}^2(u)F_4 + L_{AB}^3(u)F_2]}{1 - u} \\ &+ \frac{v[L_{BC}^1(1 - u)F_2 + L_{BC}^2(1 - u)F_5 + L_{BC}^3(1 - u)F_3]}{1 - u} \\ &\left. - wF_1 - uF_2 - vF_3 \right\} \end{aligned} \quad (\text{B-5})$$

Extracting common factors, (B-5) becomes:

$$\begin{aligned} F(u, v, w) &= \frac{1}{2} \left\{ F_1 \cdot \left[\frac{wL_{CA}^3(1 - v)}{1 - v} + \frac{vL_{CA}^3(w)}{1 - w} + \frac{uL_{AB}^1(1 - w)}{1 - w} + \frac{wL_{AB}^1(u)}{1 - u} - w \right] \right. \\ &+ F_2 \cdot \left[\frac{uL_{BC}^1(v)}{1 - v} + \frac{uL_{AB}^3(1 - w)}{1 - w} + \frac{wL_{AB}^3(u)}{1 - u} + \frac{vL_{BC}^1(1 - u)}{1 - u} - u \right] \\ &+ F_3 \cdot \left[\frac{uL_{BC}^3(v)}{1 - v} + \frac{wL_{CA}^1(1 - v)}{1 - v} + \frac{vL_{CA}^1(w)}{1 - w} + \frac{vL_{BC}^3(1 - u)}{1 - u} - v \right] \\ &+ F_4 \cdot \left[\frac{uL_{AB}^2(1 - w)}{1 - w} + \frac{wL_{AB}^2(u)}{1 - u} \right] \\ &+ F_5 \cdot \left[\frac{uL_{BC}^2(v)}{1 - v} + \frac{vL_{BC}^2(1 - u)}{1 - u} \right] \\ &\left. + F_6 \cdot \left[\frac{vL_{CA}^2(w)}{1 - w} + \frac{wL_{CA}^2(1 - v)}{1 - v} \right] \right\} \end{aligned} \quad (\text{B-6})$$

It is evident that $\frac{1}{2}$ of the functions that multiply the nodal values $F_1, F_2, F_3, F_4, F_5, F_6$ in (B-6) are the shape functions of the six-node finite element.

Substituting (B-1) in (B-6), the shape function $N_1(u, v, w)$ becomes:

$$\begin{aligned}
 N_1(u, v, w) &= \frac{1}{2} \left[\frac{w(1-v)(2(1-v)-1)}{1-v} + \frac{vw(2w-1)}{1-w} + \frac{u(2(1-w)-1)((1-w)-1)}{1-w} + \frac{w(2u-1)(u-1)}{1-u} - w \right] \\
 &= \frac{1}{2} \left[w(1-2v) + \frac{vw(2w-1) + uw(2w-1)}{1-w} - w(2u-1) - w \right] \\
 &= \frac{1}{2} \left[w(1-2v) + \frac{w(2w-1)(u+v)}{1-w} - w(2u-1) - w \right] \\
 &= \frac{1}{2} [w(1-2v) + w(2w-1) - w(2u-1) - w] \\
 &= \frac{1}{2} [w - 2wv + 2w^2 - w - 2wu] \\
 &= \frac{1}{2} [2w(-v + w - u)] \\
 &= 2w(w - 1/2)
 \end{aligned}$$

Working in a similar way, all the shape functions are given by:

$$N_1(u, v, w) = 2w(w - 1/2) \quad (\text{B-7})$$

$$N_2(u, v, w) = 2(1-v)((1-v) - 1/2) \quad (\text{B-8})$$

$$N_3(u, v, w) = 2(1-w)((1-w) - 1/2) \quad (\text{B-9})$$

$$N_4(u, v, w) = 4uw \quad (\text{B-10})$$

$$N_5(u, v, w) = 4uv \quad (\text{B-11})$$

$$N_6(u, v, w) = 4vw \quad (\text{B-12})$$

Moreover, taking into consideration that the triangular coordinates, (S_1, S_2, S_3) , are related with Barnhill's coordinates (u, v, w) by:

$$S_1 = u, \quad S_2 = 1 - v, \quad S_3 = 1 - w, \quad (\text{B-13})$$

it is immediately concluded that (B-7) until (B-12) coincide with those mentioned in finite element textbooks, for example [5,6].

Remark: Barnhill's formula allows for the construction of arbitrary-noded triangular macroelements. In such as case, Equation (B-6) must be properly extended and then is easily programmed.

REFERENCES

- [1] S. A. Coons. Surfaces for computer aided design of space form, Project MAC, MIT (1964), revised for MAC-TR-41 (1967), Springfield, VA 22161, USA: Available as AD 663 504 from the National Technical Information Service (CFSTI), Sills Building, 5285 Port Royal Road.
- [2] Online: <http://design.osu.edu/carlson/history/lesson4.html>
- [3] ModeFrontier, <http://www.esteco.com/modefrontier>
- [4] Matlab, <http://www.mathworks.com/>
- [5] O. C. Zienkiewicz. The finite element method, 3rd ed., McGraw-Hill, London, 1977.
- [6] K. J. Bathe. Finite Element Procedures in Engineering Analysis, Prentice-Hall, New Jersey, 1982.
- [7] C. A. Brebbia, J. Dominguez. Boundary Elements: an introductory course, Computational Mechanics Publications, Southampton, McGraw-Hill Book Company, 1992.
- [8] O. C. Zienkiewicz. The birth of the finite element method, Int J Numer Meth Engng, Vol. 60, 3-10, 2004.
- [9] A. Samuelsson, O. C. Zienkiewicz. History of the stiffness method, Int J Numer Meth Engng, Vol. 67(2), 149-157, 2006.
- [10] O. C. Zienkiewicz. Origins, Milestones and Directions of the Finite Element Method – A Personal View, Archives of Computational Methods in Engineering, Vol. 2(1), 1-48,

- 1995.
- [11] R. W. Clough. The finite element method after twenty-five years: A personal view, *Computers & Structures*, Vol. 12(4), 361-370, 1980.
- [12] R. W. Clough. The finite element method – a personal view at its original formulation. In: K. Bell (ed.), *From Finite Element to the Troll Platform*, Tapi, Trondheim, 1994.
- [13] I. I. Sutherland. SKETCHPAD: a man-machine graphical communication system. In: *Proc. AFIPS Spring Joint Computer Conference*, Detroit, Michigan, May 21-23, 1963, pp. 329-346 (Spartan Books, Baltimore, Md.) .
- [14] "MCS Founder: Patrick J. Hanratty, Ph.D., President and Founder". *mcsaz.com*.
- [15] C. A. Brebbia. *Finite Element Systems: A Handbook*, 2nd edn. Springer-Verlag, Berlin, 1982.
- [16] S. Park, K. Lee. High-performance trivariate NURBS representation for analyzing and visualizing fluid flow data. *Computers and Graphics*, Vol. 21, 473-482, 1997.
- [17] W. Martin, E. Cohen. Representation and extraction of volumetric attributes using trivariate splines: A mathematical framework, *Proceedings of the Solid Modelling Symposium*, 2001 (<http://www.cs.utah.edu>).
- [18] J. A. Cottrell, T. J. R. Hughes, Y. Bazilevs. *Isogeometric Analysis: Towards integration of CAD and FEA*, John Wiley, Chichester, 2009.
- [19] G. Farin, J. Hoschek, M.-S. Kim. *Handbook of Computer Aided Geometric Design*, North-Holland / Elsevier, 2002.
- [20] W. H. Press, B.P. Flannery, S. A. Teukolsky, W. T. Vetterling. *Numerical Recipes: The art of scientific computing*, Cambridge University Press, Cambridge, 1986.
- [21] B. Szabó, I. Babuška. *Finite Element Analysis*, John Wiley & Sons, Inc., New York, 1991.
- [22] C. G. Provatidis. Free vibration analysis of elastic rods using global collocation, *Archive of Applied Mechanics*, Vol. 78(4), 241-250, 2008.
- [23] C. G. Provatidis. Time- and frequency-domain analysis using lumped mass global collocation, *Archive of Applied Mechanics*, Vol. 78 (11), 909-920, 2008.
- [24] I. D. Faux, M. J. Pratt, *Computational geometry for design and manufacture*, Ellis Horwood, Chichester, 1979.
- [25] G. Farin. *Curves and surfaces for computer aided geometric design: a practical guide*, Academic Press, Boston, 1990.
- [26] C. DeBoor, *A Practical Guide to Splines*, Revised Edition, Springer-Verlag, New York, 2001
- [27] L. Piegl, W. Tiller, *The NURBS Book*, Springer-Verlag, Berlin, 1995.
- [28] I. J. Schoenberg. Contributions to the problem of approximation of equidistant data by analytic functions, *Quart. Appl. Math.*, Vol. 4, 45-99, 1946; also: I. J. Schoenberg, A. Whitney. On Polya Frequency Functions III: The positivity of translation determinants with an application to the interpolation problem by splines curves. *Trans Amer Math Soc*, Vol. 74, 246-259, 1953.
- [29] M. G. Cox, The numerical evaluation of B-splines, *Jour. Inst. Math. Applic.*, Vol. 10, 134-149, 1972.
- [30] C. De Boor. On calculating with B-splines, *J. Approx. Theory*, Vol. 6, 50-62, 1972.
- [31] C. DeBoor. Private Communication, 2007.
- [32] C. G. Provatidis. Higher order Galerkin/Ritz approximations in 1-D eigenvalue problems. In: D. Tsahalis (ed.) *Proceedings 3rd International Conference on Experiments /Process /System Modeling /Simulation & Optimization (3rd IC-EpsMsO)*, Athens, 8-11 July, 2009.
- [33] C. G. Provatidis. Equivalent finite element formulations for the calculation of eigenvalues using higher-order polynomials, *Applied Mathematics*, Vol. 1(1), 13-23, 2011.
- [34] W. J. Gordon. Blending functions methods of bivariate and multivariate interpolation and approximation, *SIAM J. Numer. Anal.*, Vol. 8, 158-177, 1971.
- [35] C. G. Provatidis. Solution of two-dimensional Poisson problems in quadrilateral domains using transfinite Coons interpolation, *Communications in Numerical Methods in Engineering*, Vol. 20(7), 521-533, 2004.
- [36] C. G. Provatidis, Performance of large Gordon-Coons finite elements in 2-D potential problems, *Proceedings GRACM 05*, Limassol, Cyprus, 29 June – 1 July, 2005.
- [37] C. G. Provatidis. Coons-patch macroelements in two-dimensional parabolic problems," *Applied Mathematical Modelling*, Vol. 30, 319-351, 2006.
- [38] A. E. Kanarachos, V. M. Dimitriou. Adaptive FEM for plane elasticity problems using "pre-constructed" large finite elements. In: D. Tsahalis (Ed.) *Proceedings 4th GRACM Congress on Computational Mechanics*, Patras, 27-29 June, 2002, pp. 324-332.
- [39] E. Barnhill, G. Birkhoff, M. J. Gordon. Smooth interpolation on triangles, *J. Approx. Theory*, Vol. 8, 114-128, 1973.
- [40] E. Barnhill, L. Mansfield, Error bounds for smooth interpolation on triangles, *J. Approx. Theory*, Vol. 11, 306-318, 1974.
- [41] R. E. Barnhill, J. A. Gregory. Compatible smooth interpolation on triangles, *J. Approx. Theory*, Vol. 15, 214-225, 1975.
- [42] K. Lee. *Principles of CAD/CAM/CAE Systems*, Addison-Wesley, Reading, Massachusetts, 1999.
- [43] W. A. Cook. Body oriented (natural) co-ordinates for generating three-dimensional meshes, *Int. J. Numer. Meth Eng*, Vol. 8, 27-43, 1974.
- [44] G. Beer, J. O. Watson. *Introduction to Finite and Boundary Element Methods for Engineers*, John Wiley & Sons, Chichester, 2002 (Chapter 11, pp. 357-377).
- [45] C. G. Provatidis. CAD-FEA integration using Coons interpolation, Technical Report MD&CS 01-2002, National Technical University of Athens, Mechanical Engineering Department; also submitted to "Engineering with Computers" (EWC03-019, Sept 12, 2003).
- [46] C. G. Provatidis, C. G. Vossou and E. G. Theodorou, On the CAD/CAE integration using Coons interpolation, *Proceedings 2nd International Conference "From Scientific Computing to Computational Engineering"*, Athens, Greece,

5-8 July, 2006.

- [47] C. G. Provatidis, Final Report for the Project: PROTAGORAS (DIDYMO: Investigation on CAD/CAE integration), EDEIL #65/1388, February, 2007.
- [48] W. J. Gordon, C. A. Hall, Construction of curvilinear co-ordinate systems and application to mesh generation, *Int J Numer Meth Eng*, Vol. 7, 461-477, 1973.
- [49] O. Röper. Ein Geometrie-Prozessor für die rechnerunterstützte Auslegung von Maschinenbauteilen mit Hilfe der Methode der Finite Elemente, Dissertation, Ruhr-Universität Bochum, 1978.
- [50] A. Kanarachos, O. Röper Rechnerunterstützte Netzgenerierung mit Hilfe der Coonsschen Abbildung, *VDI-Z*, Vol. 121, 297-303, 1979.
- [51] S. C. Wu, J. F. Abel. Representation and discretization of arbitrary surfaces for finite element shell analysis, *Int. J. Numerical Methods in Engineering*, Vol. 14, 813-836, 1979.
- [52] R. B. Haber, M. S. Shephard, J. F. Abel, R. H. Gallagher, D. P. Greensberg. A general three-dimensional graphical finite element preprocessor utilizing discrete transfinite mappings, *Int. J. Numerical Methods in Engineering*, Vol. 17, 1015-1044, 1981.
- [53] R. B. Haber, J. F. Abel. Discrete transfinite mappings for the description and meshing of three-dimensional surfaces using interactive computer graphics, *Int. J. Numerical Methods in Engineering*, Vol. 18, 41-66, 1982.
- [54] Y. B. Yildir, A. Wexler MANDAP – A FEM/BEM preparation package, *IEEE Trans Magnetics*, Vol. 19, 2562-2565, 1983.
- [55] W. J. Gordon, C. A. Hall. Transfinite element methods blending function interpolation over arbitrary curved element domains, *Numerische Mathematik*, Vol. 21, 109-112, 1973.
- [56] J. C. Cavendish, W. J. Gordon, C. A. Hall. Ritz-Galerkin approximations in blending function spaces. *Numerische Mathematik*, Vol. 26, 155-178, 1976.
- [57] C. Provatidis, A. Kanarachos. On the use of Coons' interpolation in CAD/CAE systems, In: N. Mastorakis (ed.), *Systems and Control: Theory and applications*, World Scientific and Engineering Society Press, pp. 343-348, 2000.
- [58] B. M. Irons. Engineering application of numerical integration in stiffness method, *AIAA J.*, Vol. 14(11), 2035-2037, 1966.
- [59] A. El-Zafrany A., R. A. Cookson, Derivation of Lagrangian and Hermitian shape functions for quadrilateral elements. *International Journal for Numerical Methods in Engineering*, Vol. 23 (10), 1939-1958, 1986.
- [60] A. El-Zafrany, R. A. Cookson, Derivation of Lagrangian and Hermitian shape functions for triangular elements, *International Journal for Numerical Methods in Engineering*, Vol. 23 (2), 275-285, 1986.
- [61] Z. Zhaobei, X. Zhiqiang. Coons' surface method for formulation of finite element of plates and shells. *Computers & Structures*; Vol. 27(1), 79-88, 1987.
- [62] A. Kanarachos, D. Deriziotis. On the solution of Laplace and wave propagation problems using "C-elements", *Finite Elements in Analysis and Design*, Vol. 5, 97-109, 1989.
- [63] A. Kanarachos, C. Provatidis, D. Deriziotis, N. Foteas. A new approach of the FEM analysis of two-dimensional elastic structures using global (Coons's) interpolation functions, in: Wunderlich, J. (Ed.), *CD Proceedings First European Conference on Computational Mechanics*, München-Germany, August-September, 1999.
- [64] C. Provatidis. A global approximation technique in noise-control, In: D.T. Tsahalis (ed.), *Proceedings 4th European Conference on Noise Control (EuroNoise-2001)*, 14-17 January 2001, Patras, Greece, Vol. I, pp. 23-31, 2001.
- [65] C. Provatidis. Acoustic analysis of two-dimensional mufflers using large finite elements derived from Coons' interpolation, In: P.A. Drakatos (ed.), *CD Proceedings of ASME – Greek Section*, September 17-20, 2001, Patras, Greece (Paper ANG1/P099, pages 1-6).
- [66] C. Provatidis, A. Kanarachos. Performance of a macro-FEM approach using global interpolation (Coons') functions in axisymmetric potential problems, *Computers & Structures*, Vol. 79(19), 1769-1779, 2001.
- [67] C. Provatidis. Coons-patch macroelements in potential Robin problems, *Forschung im Ingenieurwesen*, Vol. 67, 19-26, 2002.
- [68] C. G. Provatidis, Coons-patch macroelements in two-dimensional eigenvalue and scalar wave propagation problems, *Computers & Structures*, Vol. 82 (4-5), 383-395, 2004.
- [69] C. Provatidis. Analysis of axisymmetric structures using Coons' interpolation, *Finite Elements in Analysis and Design*, Vol. 39, 535-558, 2003.
- [70] C. Provatidis. Frequency analysis and transient response of two-dimensional structures using Coons-patch macroelements, In: M.J. Brennan, M.A. Ferman, B.A.T. Petersson, S.A. Rizzi and K. Wentz (eds.), *Proceedings of the VIII International Conference on Recent Advances in Structural Dynamics*, 14-16 July 2003, Southampton, UK (paper No.24).
- [71] C. Provatidis. Free vibrations of two-dimensional structures using Coons-patch macroelements, FEM and BEM, In: S.N. Atluri, D.E. Beskos and D. Polyzos (eds.), *Proceedings from ICCES 2003, Advances in Computational & Experimental Engineering & Sciences*, 24-29 July 2003, Corfu, Greece, Chapter 5, Paper No.249.
- [72] C. G. Provatidis. Free vibration analysis of two-dimensional structures using Coons-patch macroelements, *Finite Elements in Analysis and Design*, Vol. 42(6), 18-531, 2006.
- [73] C. G. Provatidis. Transient elastodynamic analysis of two-dimensional structures using Coons-patch macroelements, *International Journal of Solids and Structures*, Vol. 43 (22-23), 6688-6706, 2006.
- [74] M. Bercovier, E. Shilat. Enhancement of Gordon-Coons interpolations by "bubble functions", *Computer-Aided Design*, Vol. 10, 253-265, 1993.
- [75] C. Provatidis. A comparative study between Coons-patch macroelements and boundary elements in two-dimensional potential problems, In: D.T. Tsahalis (ed.), *Proceedings 4th GRACM Congress on Computational Mechanics*, 27-29 June, 2002, Patras, Greece, pp.43-50.
- [76] C. G. Provatidis. Eigenanalysis of two-dimensional acoustic cavities using transfinite interpolation, *Journal of Algorithms*

- & Computational Technology, Vol. 3(4), 477-502, 2009.
- [77] C. G. Provatidis. Three-dimensional Coons macroelements in Laplace and acoustic problems, *Computers and Structures*, Vol. 83, 1572-1583, 2005.
- [78] C. G. Provatidis. Analysis of box-like structures using 3-D Coons' interpolation, *Communications in Numerical Methods in Engineering*, Vol. 21, 443-456, 2005.
- [79] C. G. Provatidis. Three-dimensional Coons macroelements: Application to eigenvalue and scalar wave propagation problems, *International Journal for Numerical Methods in Engineering*, Vol. 65, 111-134, 2006.
- [80] C. G. Provatidis. Two-dimensional elastostatic analysis using Coons-Gordon interpolation, *Meccanica*, Vol. 47(4), 951-967, 2012.
- [81] M. S. Casale. Integration of geometric analysis and structural analysis using trimmed patches. PhD Thesis, University of California, Irvine, 1989.
- [82] M. S. Casale, J. E. Bobrow. The analysis of solids without mesh generation using trimmed patch boundary elements, *Engineering with Computers*, Vol. 5, 249-257, 1989.
- [83] M.S. Casale, J. E. Bobrow, R. Underwood. Trimmed-patch boundary elements: bridging the gap between solid modeling and engineering analysis, *Computer-Aided Design*, Vol. 24, 193-199, 1992.
- [84] C. Provatidis Stress analysis of 3D solid structures using large boundary elements derived from 2D-Coons' interpolation, In: P.A. Drakatos (ed.), *CD Proceedings of ASME – Greek Section*, September 17-20, 2001, Patras, Greece (Paper ANGI/P129, pages 1-6).
- [85] C. Provatidis. Analysis of three-dimensional sound radiation problems using trimmed patch boundary elements, In: D.T. Sahalis (ed.), *Proceedings 4th GRACM Congress on Computational Mechanics*, 27-29 June, 2002, Patras, Greece, pp.402-409.
- [86] C. Provatidis, N. Zafiroopoulos. Free-vibration analysis of three-dimensional solids using Coons-patch Boundary Superelements, In: E. Aifantis (ed.), *Proceedings 5th European Solids Mechanics Conference*, Thessaloniki, 17-22 August 2003.
- [87] C. Provatidis, N. Zafiroopoulos. Determination of eigenfrequencies in three-dimensional acoustic cavities using Coons-patch Boundary Superelements, In: E. Aifantis (ed.), *Proceedings 5th European Solids Mechanics Conference*, Thessaloniki, 17-22 August 2003.
- [88] A. Kanarachos, D. Grekas, C. Provatidis. Generalized Formulation of Coons' Interpolation. In: P.Kaklis&N.Sapidis (eds.) *Computer Aided Geometric Design From Theory to Practice*, NTUA Publications, Athens, Chapter 7, pp. 65-76, 1995.
- [89] V. Dimitriou, Adaptive finite elements and related meshes, Ph.D. dissertation (advisor: Prof. Andreas E. Kanarachos), National Technical University of Athens, School of Mechanical Engineering, Athens, August, 2004.
- [90] D. Antoniou. Barnhill's macroelements in plane elasticity problems. MSc Thesis, National Technical University of Athens, Interdepartmental Course in Computational Mechanics (supervisor: Prof. C.G. Provatidis), 2010.
- [91] P.Bézier. Définition numérique des courbes et surfaces I, *Automatisme*, Vol. 11, 625-632, 1966.
- [92] P.Bézier. Définition numérique des courbes et surfaces II. *Automatisme*, Vol. 12, 17-21, 1967.
- [93] P.Bézier. Procédé de définition numérique des courbes et surfaces non mathématiques. *Automatisme*, Vol. 13(5), 189-196, 1968.
- [94] P. Bézier. How Renault uses Numerical Control for Car Body Design and Tooling. *SAE Paper* 680010, 1968.
- [95] P. E. Bézier, Example of an existing system in motor industry: The UNISURF system, *Proc R Soc Lon Ser-A*, Vol. 321, 207-218, 1971.
- [96] P. Bézier. *Numerical Control: Mathematics and Applications*, Wiley, 1972 (translated by R. Forrest).
- [97] P. Bézier. UNISURF System : Principles, Program, Language, *Proc. 1973 PROLAMAT Conference*, Budapest (J. Harvany, ed.), North Holland Publ. Co., Amsterdam, 1973.
- [98] P. Bézier. Mathematical and practical possibilities of UNISURF. In: R. Barnhill and R. Riesenfeld (Eds.), *Computer Aided Geometric Design*, Academic Press, 1974.
- [99] P.Bézier. *Essay de définition numérique des courbes et des surfaces expérimentales*. PhDthesis, University of Paris VI, 1977.
- [100] P. Bézier. General distortion of an ensemble of biparametric surfaces, *Comput Aided Design*, Vol. 10 (2), 116-120, 1978.
- [101] P. E. Bézier. A view of CAD-CAM, *Comput Aided Design*, Vol. 13 (4), 207-209, 1981.
- [102] P. E. Bézier, S. Sioussiou, Semi-automatic system for defining free-form curves and surfaces, *Comput Aided Design*, Vol.15(2), 65-72, 1983.
- [103] P. E. Bézier, UNISURF, From styling to tool-shop, *COMPUT IND*, Vol. 4 (2), 115-126, 1983.
- [104] P. E. Bézier, CAD/CAM - past, requirements, trends, *Comput Aided Design*, Vol. 16 (2), 102-102, 1984.
- [105] P. Bézier. The mathematical basis of the UNISURF CAD System, Butterworths, London, 1986.
- [106] P. Bézier, 1st Steps of CAD, *Comput Aided Design*, Vol. 21(5), 259-261, 1989.
- [107] P. Bézier. Style, mathematics and NC, *Comput Aided Design*, Vol.22 (9), 524-526, 1990.
- [108] P. Bézier. A view of the CAD/CAM development period, *IEEE Ann HistComput*, Vol. 20 (2), 37-40, 1998.
- [109] C. Rabut. On Pierre Bézier's life and motivations. *Computer-Aided Design*, Vol. 34(7), 493-510, 2002.
- [110] D. Lasser. Bernstein-Bézier representation of volumes, *Computer-Aided Design*, Vol. 2, 145-149, 1985.
- [111] C. G. Provatidis. Bézier versus Lagrange polynomials-based finite element analysis of 2-D potential problems, *Advances in Engineering Software* (under revision).
- [112] C. G. Provatidis. Global versus local interpolation in the FEM free vibration analysis of prismatic bars, *Proceedings 7th GRACM International Congress in Mechanics*, Athens, 30

June-2 July, 2011.

- [113] A. R. Forrest, Curves and Surfaces for Computer-Aided Design, Ph.D dissertation, Cambridge University, Cambridge, UK, 1968.
- [114] G. E. Farin, Algorithms for rational Bézier curves, Computer Aided Design, Vol. 15(2), 73-77, 1983.
- [115] L. Piegl. A geometric investigation of the rational Bézier scheme of Computer Aided Design, Computers in Industry, Vol. 7, 401-410, 1986.
- [116] W. J., Gordon, R.F. Riesenfeld. B-spline curves and surfaces, In: Computer Aided Geometric Design, R. E. Barnhill and R.F. Riesenfeld, (Eds.), Academic Press, New York, 1974.
- [117] R. F. Riesenfeld, Applications of B-spline Approximation to Geometric Problems of Computer-Aided Design, Ph.D. dissertation, Syracuse University, USA, 1973.
- [118] K. J. Versprille., Computer-Aided Design Applications of the Rational B-spline Approximation Form, Ph.D. dissertation, Syracuse University, USA, 1975.
- [119] W. Tiller. Rational B-splines for curve and surface representation, IEEE Comput. Graph. and Appl., Vol. 3 (6), 61-69, 1983.
- [120] L. Piegl. On NURBS: A Survey, IEEE Comput. Graph. and Appl., Vol. 10 (1), 55-71, 1991.
- [121] W. W. Charlesworth, J. J. Cox, D. C. Anderson. The domain decomposition method applied to Poisson's equation in two dimensions, International Journal of Solids and Structures, Vol. 37(18), 3093-3115, 1994.
- [122] B. W. Clark, D. C. Anderson, The penalty boundary method for combining meshes and solid models in finite element analysis, Engineering Computations, Vol. 20(4), 344-365, 2003.
- [123] B. W. Clark, D. C. Anderson. The penalty boundary method, Finite Elements in Analysis and Design, Vol. 39, 387-401, 2003.
- [124] D. Natekar, X. Zhang, G. Subbarayan. Constructive solid analysis: a hierarchical, geometry-based meshless analysis procedure for integrated design and analysis, Computer-Aided Design, Vol. 36(5), 473-486, 2004.
- [125] F. Q. Lin, W. T. Hewitt, Expressing Coons-Gordon surfaces as NURBS, Computer-Aided Design, Vol. 26(2), 145-155, 1994.
- [126] L. Z. Wang, X. X. Zhu, Z. S. Tang. Coons type blended B-spline (CNSBS) surface and its conversion to NURBS surface, Computers & Graphics, Vol. 21(3), 297-303, 1997.
- [127] V. Zatko, Z. Mederlyová, On the Coons bodies, Proc. 15th Spring Conf. on Computer Graphics (SCCG 1999), Budmerice, Slovakia, May 1999, 247-253.
- [128] R. Ďurikovič, S. Czanner. Modelling with three types of Coons bodies, International Journal of Modelling and Simulation, IASTED, Vol. 24(2), 97-101, 2004.
- [129] P. De Doncker. The use of the transfinite interpolation in the method of moments applied to electromagnetic scattering by dielectric cylinders, IEEE Transactions on Antennas and Propagation, Vol. 48(5), 661-670, 2000.
- [130] A. Samani, J. Bishop, M. J. Yaffe, D. B. Plewes. Biomechanical 3-D Finite Element Modeling of the Human Breast Using MRI Data, IEEE Transactions on Medical Imaging, Vol. 20(4), 271-279, 2001.
- [131] M. Aristodemo. A high-continuity finite element model for two-dimensional elastic problems, Computers & Structures, Vol. 21 (5), 987-993, 1985.
- [132] J. M. Brown, M. I. G. Bloor, M. S. Bloor, M. J. Wilson. The accuracy of B-spline finite element approximations to PDE, Comput Methods ApplMechEng, Vol. 158, 221-234, 1998.
- [133] E. Turco, M. Aristodemo. A three-dimensional B-spline boundary element, Comput. Methods Appl. Mech. Engrg., Vol. 155, 119-128, 1998.
- [134] P. Kagan, A. Fischer. Integrated mechanical based CAE systems using B-spline based finite elements, Comput Aid Des, Vol. 32 (8-9), 539-552, 2000.
- [135] K. Höllig. Finite Element Approximation with Splines. In: Farin G, Hoschek J, Kim MS (Eds.), *Handbook of Computer Aided Geometric Design*, North-Holland, Amsterdam, 2002, Chapter 11, pp.283-307.
- [136] K. Höllig. Finite Element Methods with B-Splines, SIAM, Philadelphia, 2003.
- [137] U. Schramm, W. W. Pilkey. Higher order boundary elements for shape optimization using rational B-splines, Engineering Analysis with Boundary Elements, Vol. 14 (3), 255-266, 1994.
- [138] J. J. S. P. Cabral, L. C. Wrobel, C. A. Brebbia, A BEM formulation using B-splines: I-uniform blending functions, Engineering Analysis with Boundary Elements, Vol. 7 (3), 136-144, 1990.
- [139] J. J. S. P. Cabral, L. C. Wrobel, C. A. Brebbia. A BEM formulation using B-splines: II-multiple knots and non-uniform blending functions, Engineering Analysis with Boundary Elements, Vol. 8 (1), 51-55, 1991.
- [140] R. Ushatov, H. Power, J. J. Rêgo Silva. Uniform bicubic B-splines applied to boundary element formulation for 3-D scalar problems, Engineering Analysis with Boundary Elements, Vol. 13(4), 371-381, 1994.
- [141] M. Cerrolaza, W. Annicchiarico, M. Martinez. Optimization of 2D boundary element models using B-splines and genetic algorithms, Engineering Analysis with Boundary Elements, Vol. 24(5), 427-440, 2000.
- [142] E. Dimas, D. Briassoulis. 3D geometric modelling based on NURBS: a review, Advances in Engineering Software, Vol. 30, 741-751, 1999.
- [143] U. Schramm, W. D. Pilkey. The coupling of geometric descriptions and finite element using NURBS – A study in shape optimization, Finite Elements in Analysis and Design, Vol. 15, 11-34, 1993.
- [144] F. P. Renken, G. Subbarayan. NURBS-based solution to inverse boundary problems in droplet shape prediction, Comput Meth ApplMechEngng, Vol. 190 (11), 1391-1406, 2000.
- [145] K. Inoue, Y. Kikuchi, T. Masuyama. A NURBS finite element method for product shape design, Journal of Engineering Design, Vol. 16(2), 157-174, 2005.
- [146] X. Zhou, J. Lu. NURBS-Based Galerkin Method and

- Application to Skeletal Muscle Modeling, Association for Computing Machinery Inc., 71-78, 2005.
- [147] T. J. R. Hughes, J. A. Cottrell, Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Comput Methods Appl Mech Eng*, Vol. 194, 4135-4195, 2005.
- [148] J. A. Cottrell, A. Reali, Y. Bazilevs, T. J. R. Hughes. Isogeometric analysis of structural vibrations, *Comput Methods Appl Mech Eng*, Vol. 195, 5257-5296, 2006.
- [149] T. J. R. Hughes, J. A. Cottrell, Y. Bazilevs, A. Reali. Computational Geometry as a Basis for Computational Structures Technology: A Look into the Future. In: B.H.V. Topping, G. Montero and R. Montenegro (Eds.), *Innovation in Computational Structures Technology*, Saxe-Coburg Publications, Stirlingshire, 2006, pp.1-22 (ISBN-10 1-874672-27-X).
- [150] T. J. R. Hughes, A. Reali, G. Sangalli. Efficient quadrature for NURBS-based isogeometric analysis, *Comput Methods Appl Mech Eng*, Vol. 199, 301-313, 2010.
- [151] J. J. Van Blerk, J. F. Botha. Numerical solution of partial differential equations on curved domains by collocation, *Numerical Methods for Partial Differential Equations*, Vol. 9, 357-371, 1993.
- [152] C. G. Provatidis. Performance of a Lagrange based global finite element collocation method for eigenvalue structural analysis, *Proceedings 8th HSTAM International Congress on Mechanics*, Patras, 12 – 14 July, 2007.
- [153] C. G. Provatidis, S. Isidorou, Comparison of advanced collocation methods for the solution of ordinary differential equations. In: D. Tsahalis (ed.) *Proceedings 3rd International Conference on Experiments/Process/System Modeling/Simulation & Optimization (3rd IC-EpsMsO)*, Athens, 8-11 July, 2009.
- [154] C. G. Provatidis, S. K. Isidorou. B-splines collocation eigenvalue analysis of 1-D problems, *Proceedings 7th GRACM International Congress in Mechanics*, Athens, 30 June-2 July, 2011.
- [155] C. G. Provatidis, Global collocation method for 2-D rectangular domains, *Journal of Mechanics of Materials and Structures*, Vol. 3 (1), 185-194, 2008.
- [156] C. G. Provatidis. Integration-free Coons macroelements for the solution of 2-D Poisson problems, *International Journal for Numerical Methods in Engineering*, Vol. 77, 536-557, 2009.
- [157] C. G. Provatidis, K. S. Ioannou, Static analysis of two-dimensional elastic structures using global collocation, *Archive of Applied Mechanics*, Vol. 80 (4), 389-400, 2010.
- [158] A. Filippatos, Extraction of eigenfrequencies in acoustic cavities and elastic structures using the Global Collocation Method, Diploma Work, National Technical University of Athens, School of Mechanical Engineering, Athens, Greece, October, 2010 (supervisor: Prof. C. G. Provatidis).
- [159] C. G. Provatidis. Some issues on CAD/CAE integration: Global interpolation using isoparametric and isogeometric techniques, *Proceedings 7th GRACM International Congress in Mechanics*, Athens, 30 June-2 July, 2011.
- [160] F. Auricchio, L. Beirão da Veiga, T. J. R. Hughes, A. Reali, G. Sangalli. Isogeometric collocation methods, *Mathematical Models and Methods in Applied Sciences*, Vol. 20, 2075-2107, 2010.
- [161] INTEGRATED Engineering Software, *Advances of NURBS in CAE Modeling*. Online at: <http://www.integratedsoft.com/Papers/research/NURBS.pdf>