



# Enhanced parametric shape descriptions in PGD-based space separated representations

Mohammad Javad Kazemzadeh-Parsi, Amine Ammar, Jean Louis Duval,  
Francisco Chinesta

## ► To cite this version:

Mohammad Javad Kazemzadeh-Parsi, Amine Ammar, Jean Louis Duval, Francisco Chinesta. Enhanced parametric shape descriptions in PGD-based space separated representations. Advanced Modeling and Simulation in Engineering Sciences, 2021, 8 (1), 10.1186/s40323-021-00208-2. hal-03424406

**HAL Id: hal-03424406**

**<https://hal.science/hal-03424406>**

Submitted on 10 Nov 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

RESEARCH ARTICLE

Open Access



# Enhanced parametric shape descriptions in PGD-based space separated representations

Mohammad Javad Kazemzadeh-Parsi<sup>2,1\*</sup>, Amine Ammar<sup>2</sup>, Jean Louis Duval<sup>3</sup> and Francisco Chinesta<sup>3,4</sup>

\*Correspondence:

mohammadjavad.kazemzadehparsi@ensam.eu;

kazemzadeh@iaushiraz.ac.ir

<sup>1</sup> Department of Mechanical Engineering, Shiraz Branch, Islamic Azad University, Shiraz, Iran

Full list of author information is available at the end of the article

## Abstract

Space separation within the Proper Generalized Decomposition—PGD—rationale allows solving high dimensional problems as a sequence of lower dimensional ones. In our former works, different geometrical transformations were proposed for addressing complex shapes and spatially non-separable domains. Efficient implementation of separated representations needs expressing the domain as a product of characteristic functions involving the different space coordinates. In the case of complex shapes, more sophisticated geometrical transformations are needed to map the complex physical domain into a regular one where computations are performed. This paper aims at proposing a very efficient route for accomplishing such space separation. A NURBS-based geometry representation, usual in computer aided design—CAD—, is retained and combined with a fully separated representation for allying efficiency (ensured by the fully separated representations) and generality (by addressing complex geometries). Some numerical examples are considered to prove the potential of the proposed methodology.

**Keywords:** Space separated representation, Complex geometries, PGD, NURBS

## Introduction

A generic problem in physics consists of a differential operator acting on the so-called unknown field. In what follows for the sake of clarity we will assume a scalar unknown field depending on space  $\mathbf{x}$  and time  $t$ , as well as on a series of parameters, grouped in vector  $\boldsymbol{\mu}$ , related to the considered physics, the loading or the domain in which the problem is defined. Thus, the problem reads

$$\mathcal{L}(u(\mathbf{x}, t; \boldsymbol{\mu})) = f(\mathbf{x}, t; \boldsymbol{\mu}), \quad (1)$$

where  $\mathbf{x} \in \Omega_{\mathbf{x}}(\boldsymbol{\mu}) \subset \mathbb{R}^3$ ,  $t \in \Omega_t \subset \mathbb{R}$  and the parameteres  $\boldsymbol{\mu} \in \Omega_{\boldsymbol{\mu}} \subset \mathbb{R}^P$ .

For approximating the unknown function, and in absence of *a priori* knowledge, any multi-purpose polynomial approximation basis can be used. The Finite Element Method—FEM—uses simple approximation bases with interpolative properties and compact support. To approximate very general functions the approximation basis must be rich enough, with the computational efficiency impact, in particular when addressing nonlinear tran-

sient models, where the computational complexity scales with the space and time mesh resolution.

For alleviating such complexity, the first route consists in reducing the number of degrees of freedom. Thus, the so-called Proper Orthogonal Decomposition—POD—[10], extract from some offline computations the most significant modes  $\phi_i(\mathbf{x})$  for representing those solutions.

A generalization consists of expressing  $u(\mathbf{x}, t)$  from the finite sum

$$u(\mathbf{x}, t) \approx \sum_{i=1}^M T_i(t) X_i(\mathbf{x}), \quad (2)$$

with both functions  $T_i(t)$  and  $X_i(\mathbf{x})$  assumed unknown and calculated on the fly by using an appropriate linearization [25]. The interested reader can refer to [3, 9, 26, 29, 32, 33] and the references therein for additional references.

Equation (2) expresses that the solution of  $M$  problems in space for calculating functions  $X_i(\mathbf{x})$  and the same number of one-dimensional problems defined in time for computing function  $T_i(t)$  suffices for computing it, in fact few more because of the nonlinearity of the solution procedure.

PGD has also been used with the attempt of alleviating the complexity of the problems involving the space coordinates. For that purpose, when the spatial domain  $\Omega_{\mathbf{x}}$  can be fully or partially decomposed, i.e.  $\Omega_{\mathbf{x}} = \Omega_x \times \Omega_y \times \Omega_z$  or  $\Omega_{\mathbf{x}} = \Omega_{xy} \times \Omega_z$  respectively, the separated representation based approximations read respectively

$$u(x, y, z) \approx \sum_{i=1}^M X_i(x) Y_i(y) Z_i(z), \quad (3)$$

that allows calculating the 3D solution from a sequence of 1D problems, and

$$u(x, y, z) \approx \sum_{i=1}^M X_i(x, y) Z_i(z). \quad (4)$$

The former is specially suitable when the problem is defined in hexahedral domains, whereas the later applies in degenerated domains like plates [5], shells [6] or extruded domains (e.g. beams) [7, 27]. In both cases the computing time savings could be impressive, with a complexity scaling with standard 1D discretization for the former and 2D for the last, while computing a fully 3D solution [9]. The same approach was extensively considered in structural plate and shell models in [31, 36–38], squeeze flows [13] and electromagnetism problems [35] defined in laminates.

When the domain is not intrinsically separable, fully separated representations can be performed by immersing of the non-separable domain onto a fully separable one and use penalty formulations [15, 18]. Another valuable route consists in using appropriate geometrical mappings [2, 15–17].

A step forward in the use of separated representations consisted in assuming model parameters as extra-coordinates

$$u(\mathbf{x}, t, \mu_1, \dots, \mu_P) \approx \sum_{i=1}^M X_i(\mathbf{x}) T_i(t) \prod_{j=1}^P M_i^j(\mu_j), \quad (5)$$

leading to the definition of multi-parametric solutions of parametrized models, that once calculated offline, enable simulation, optimization, inverse analysis, uncertainty propagation and simulation-based control, all them under the real-time constraint [8, 10].

An appealing use of those separated representations concern their use in shape optimization as reported in [1]. However, several difficulties were noticed, among them the necessity of easily parametrizing complex geometries in a simple and compact form when addressing CAD geometries. There are techniques in CAD enabling accurate geometry descriptions based on the employ of NURBS. Recently such geometrical descriptors were combined with a discretization operating on the same framework, leading to the so-called Isogeometric Analysis (IGA) [12, 19]. Thus, the IGA unifies the NURBS-based geometry description and a NURBS-based field approximation. Some advantages of the IGA are [19]: elimination of mesh generation process, exact (or very accurate) geometric representation, easy formulation of high order element and superior numerical accuracy than standard FEM. Several works have shown superior accuracy and robustness of IGA and excellent behavior has been proven in a number of publications [4, 11, 14, 19–22, 28]. The NURBS can be also used in conjunction with usual discretization. Thus, they were also applied for enhancing standard finite element discretization [34]. Beside the above-mentioned methods, hyper reduction [24] and reduced basis isogeometric analysis [23] were also considered for geometry parametrization within the model order reduction setting.

The present paper aims at exploring the use of NURBS-based geometrical descriptions combined with the PGD-based fully 3D space separation. Moreover, considering the control points as model extra-coordinates within the PGD rationale we will address parametric geometries, that could be enriched by considering other parameters as extra-coordinates, like material parameters.

All these points will be addressed in the next sections, before illustrating the solution capabilities from the solution and analysis of few selected problems. For the sake of simplicity, and without loss of accuracy, the different capabilities will be illustrated on the heat equation.

## Geometrical description enabling fully separated representations

### Problem statement

As stated before, the main goal of the present work is to develop a procedure for solving boundary value problems (BVP) in non-separable geometries using the PGD. Therefore, to describe the proposed method and also to evaluate it, we are considering the nonlinear heat equation with temperature dependent thermal conductivity. The heat equation defined in the domain  $\Omega_x$  reads

$$\frac{\partial}{\partial x} \left( K(T) \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( K(T) \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( K(T) \frac{\partial T}{\partial z} \right) + f = 0, \quad (6)$$

where  $T$  is the temperature field,  $K(T)$  is the thermal conductivity assumed depending on the temperature and here assumed isotropic, and  $f$  is the volumetric heat source.

Appropriate boundary conditions must be enforced on the domain boundary  $\Gamma \equiv \partial\Omega_x$

$$T = \bar{T} \quad \text{on} \quad \Gamma_D \quad (7)$$

and

$$q = \bar{q} \quad \text{on} \quad \Gamma_N \quad (8)$$

with the heat flux  $q = -K \nabla T \cdot \mathbf{n}$  (with  $\nabla$  the gradient operator and  $\mathbf{n}$  the unit outwards vector defined on the domain boundary), and  $\bar{T}$  and  $\bar{q}$  the prescribed temperature and heat flux on the Dirichlet  $\Gamma_D$  and Neumann  $\Gamma_N$  boundaries respectively.



The Galerkin weak formulation of the heat equation reads

$$\begin{aligned} \int_{\Omega_x} \left[ \frac{\partial T^*}{\partial x} \left( K \frac{\partial T}{\partial x} \right) + \frac{\partial T^*}{\partial y} \left( K \frac{\partial T}{\partial y} \right) + \frac{\partial T^*}{\partial z} \left( K \frac{\partial T}{\partial z} \right) \right] d\mathbf{x} \\ = \int_{\Omega_x} T^* f d\mathbf{x} + \int_{\Gamma_N} T^* \bar{q} d\mathbf{x}, \end{aligned} \quad (9)$$

where  $T^*$  represents the test function, assumed regular enough and vanishing in  $\Gamma_D$ .

#### NURBS-based geometrical approximation

Using the NURBS basis functions (introduced in Appendix A), a single variate curve  $C$ , a bivariate surface  $S$  and a trivariate volume  $V$  can be described according to

$$\mathbf{x} = \sum_{i=1}^n R_i^p(\xi) \mathbf{P}_i, \quad \mathbf{x} \in C, \quad (10)$$

$$\mathbf{x} = \sum_{i=1}^n \sum_{j=1}^m R_{ij}^{pq}(\xi) \mathbf{P}_{ij}, \quad \mathbf{x} \in S, \quad (11)$$

and

$$\mathbf{x} = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^l R_{ijk}^{pqr}(\xi) \mathbf{P}_{ijk}, \quad \mathbf{x} \in V, \quad (12)$$

respectively, where,  $\mathbf{P}$  refers to the control points (the vertices of the so-called control net) in the physical domain (curve, surface or volume).

Equations (10)–(12) represent the application that maps any point  $\xi$  into the computational domain  $\Omega_\xi$  ( $[0, 1]$  in 1D,  $[0, 1]^2$  in 2D and  $[0, 1]^3$  in 3D) into a point  $\mathbf{x}$  in physical domain  $\Omega_x$ . The form of the NURBS-based approximation functions involved in these expressions is given in Appendix A.

These expressions allow computing the different partial derivatives involved in the differential operators as well as in the transformation Jacobian. The main drawback of such a mapping is that those derivatives have not an affine structure. For that purpose we apply a separated approximate representation —SAR— described in the next section.

#### Separated approximate representation (SAR)

Consider the generic function  $g(\mathbf{x})$  defined in the physical domain  $\Omega_x$  and  $g(\xi)$ , its counterpart in the computational domain  $\Omega_\xi$ . A separated approximate representation of  $g(\xi)$  in the computational domain reads

$$g(\xi) \approx \sum_{i=1}^{NG} \prod_{j=1}^{ND} M_j^T(\xi_j) \mathbf{G}_{ji}, \quad (13)$$

where  $M_j(\xi_j)$  is the vector of approximation functions (such as usual piecewise linear function like the ones employed in FEM) in term of  $j$ -th coordinate direction ( $\xi_1, \xi_2, \xi_3 \equiv \xi, \eta, \zeta$ ) and  $\mathbf{G}$  is the vector of coefficients.  $ND$  is the number of coordinates (2 in 2D and 3 in 3D) and  $NG$  is the number of modes used to express function  $g(\xi)$ .

The procedure for calculating the unknown coefficients  $\mathbf{G}_{ji}$  was addressed in [9], here revisited in Appendix B.

The SAR is used for performing the affine decomposition of the terms involved in the geometrical mapping

$$x(\xi) = \sum_{i=1}^{NX} \prod_{j=1}^{ND} M_j^T X_{ji}, \quad (14)$$

$$y(\xi) = \sum_{i=1}^{NY} \prod_{j=1}^{ND} M_j^T Y_{ji} \quad (15)$$

and

$$z(\xi) = \sum_{i=1}^{NZ} \prod_{j=1}^{ND} M_j^T Z_{ji} \quad (16)$$

where,  $NX$ ,  $NY$  and  $NZ$  are the number of modes required to reach the desired level of accuracy in the SAR of  $x(\xi)$ ,  $y(\xi)$  and  $z(\xi)$  respectively. The associated Jacobian expression is derived in Appendix .

#### Proper generalized decomposition constructor

The separated representation of the temperature field  $T(\xi)$  in the computational domain  $\Omega_\xi$  is constructed by progressive enrichment. Thus, with  $(n-1)$  modes already calculated, the  $n$  term approximate reads

$$T(\xi) \approx \sum_{i=1}^{n-1} \prod_{j=1}^{ND} M_j^T T_{ji} + \prod_{j=1}^{ND} M_j^T T_{jn}. \quad (17)$$

Finding the coefficient vectors  $T_{jn}$ , consists of using a fixed-point iteration [9]. At each iteration of the fixed-point algorithm, one vector among the  $ND$  unknown vectors  $T_{jn}$ , is considered unknown and the others are considered known.

Therefore, the test field involved in the weighted residual formulation, reads

$$T^{*d}(\xi) = M_d^T T^{*d} \prod_{\substack{j=1 \\ j \neq d}}^{ND} M_j^T T_{jn} \quad (18)$$

where  $d$  represents the considered direction, for calculating  $T_{dn}$ .

The other source of nonlinearity is the thermal conductivity that is assumed depending on the temperature. The simplest approach to overcome this issue, among many other alternatives, consists in evaluating it at the previous temperature approximate, and then apply the SAR according to

$$K(T(\xi)) = K(\xi) = \sum_{i=1}^{NK} \prod_{j=1}^{ND} M_j^T K_{ji}. \quad (19)$$

When considering three dimensions, i.e.  $ND = 3$ , the problem weak form reads

$$\begin{aligned} & \int_{\Omega_x} \frac{\partial T^{*d}}{\partial x} \frac{\partial T}{\partial x} K dx + \int_{\Omega_x} \frac{\partial T^{*d}}{\partial y} \frac{\partial T}{\partial y} K dx + \int_{\Omega_x} \frac{\partial T^{*d}}{\partial z} \frac{\partial T}{\partial z} K dx \\ & = \int_{\Omega_x} T^{*d} f dx + \int_{\Gamma_N} T^{*d} \bar{q} dx. \end{aligned} \quad (20)$$

The weak form is then mapped into the computational domain  $\Omega_\xi$  where the separated representation of the temperature field will apply, as well as the one of the conductivity.

For this purpose we will use the SAR of the transformation derivatives and the Jacobian determinant.

Thus, for example the first term in Eq. (20) leads to

$$\int_{\Omega_{\xi}} \left( h_{11} \frac{\partial T^{*d}}{\partial \xi} + h_{12} \frac{\partial T^{*d}}{\partial \eta} + h_{13} \frac{\partial T^{*d}}{\partial \zeta} \right) \left( h_{11} \frac{\partial T}{\partial \xi} + h_{12} \frac{\partial T}{\partial \eta} + h_{13} \frac{\partial T}{\partial \zeta} \right) K(\xi) |J| d\xi, \quad (21)$$

with  $K(\xi)$  given by the SAR (19), and where  $h_{11}$ ,  $h_{12}$  and  $h_{13}$  are the coefficients relating the derivatives in the physical and computational domains, with  $|J|$  the determinant of the Jacobian of the transformation (the expression of all them are given in Appendix ).

Equation (21) results at its turn in nine terms, where the first reads

$$\int_{\Omega_{\xi}} \frac{\partial T^{*d}}{\partial \xi} \frac{\partial T}{\partial \xi} h_{11} h_{11} K(\xi) |J| d\xi. \quad (22)$$

The 27 resulting terms involved in the left-hand side of Eq. (20) have the same structure, expressed by

$$\int_{\Omega_{\xi}} \frac{\partial T^{*d}}{\partial \xi_a} \frac{\partial T}{\partial \xi_b} h_{ca} h_{cb} K(\xi) |J| d\xi, \quad (23)$$

where indices  $a, b, c$  take values in the set  $\{1, 2, 3\}$ . The solution procedure is detailed in Appendix D.

## Numerical results

In this section, to evaluate the applicability of the proposed method, some numerical examples are solved and the results are compared with exact solutions or the ones obtained by using the FEM that will be considered as reference.

To evaluate the accuracy the following relative error norm is used

$$e = \left( \frac{\int_{\Omega_x} (T^{ref} - T^{PGD})^2 d\mathbf{x}}{\int_{\Omega_x} (T^{ref})^2 d\mathbf{x}} \right)^{1/2}, \quad (24)$$

where  $T^{ref}$  refers the reference solution and  $T^{PGD}$  to the approximate numerical solution obtained using the proposed method.

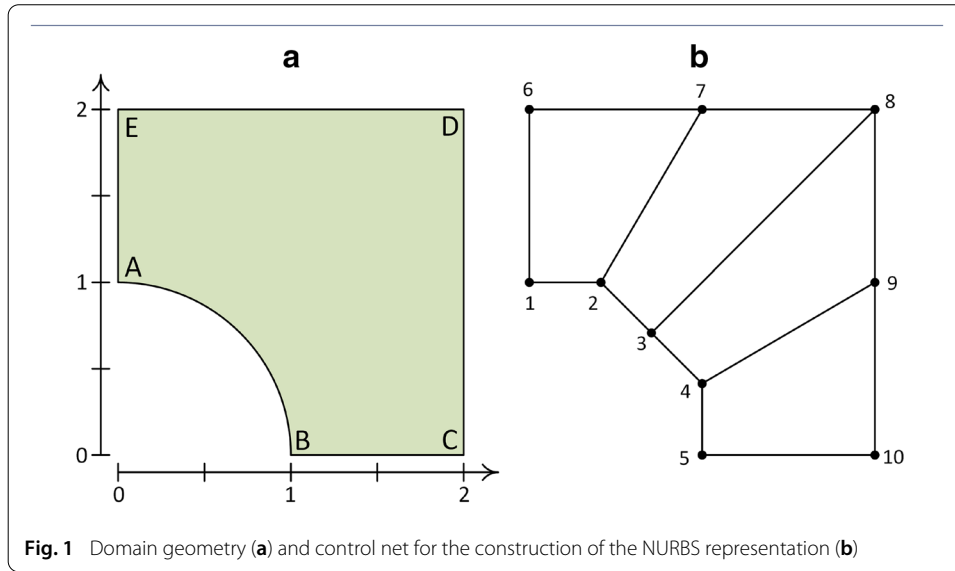
## 2D geometry

The domain geometry and control net associated to a NURBS surface are shown in Fig. 1. The order of the B-splines considered is  $p = 2$  in  $\xi$  and  $q = 1$  in  $\eta$ , associated to the knots (the interested reader can find the main concepts and notation related to the NURBS in Appendix A)

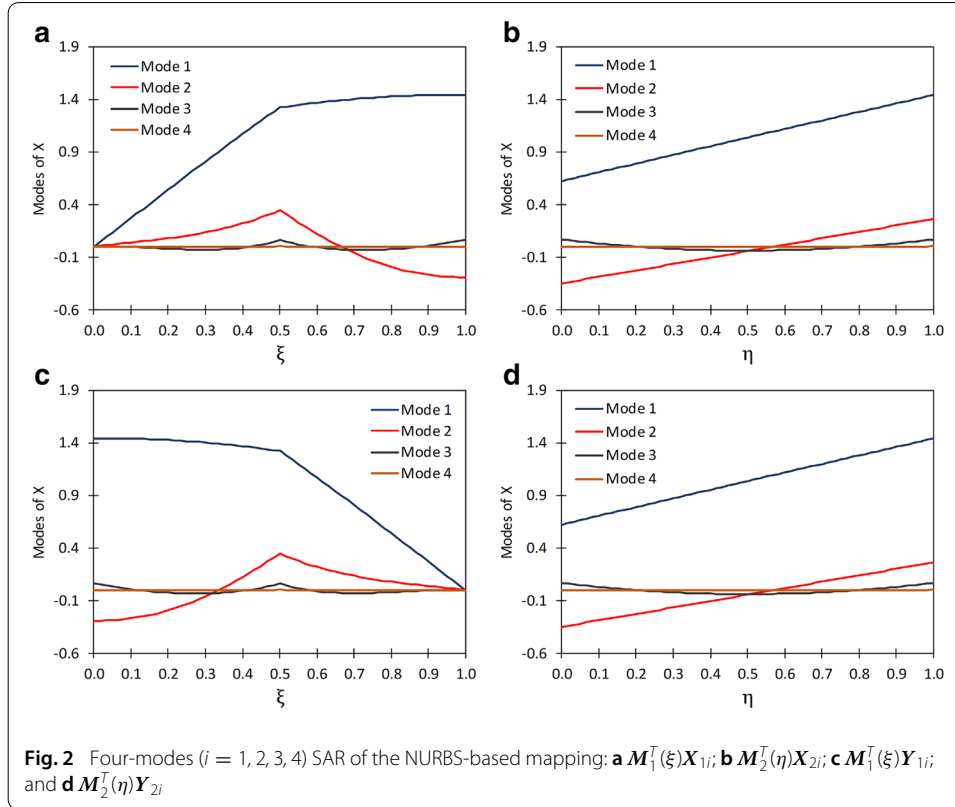
$$\begin{cases} \kappa_{\xi} = [0, 0, 0, 0.5, 0.5, 1, 1, 1]^T \\ \kappa_{\eta} = [0, 0, 1, 1]^T \end{cases}. \quad (25)$$

A repeated knot is added in the knot vector  $\kappa_{\xi}$  to enforce the sharp corner at the control point 8 in Fig. 1b. To exactly represent the circular part of the boundary we consider the weighting coefficients [30]

$$w = [1, \cos(\pi/8), 1, \cos(\pi/8), 1, 1, 1, 1, 1]^T. \quad (26)$$



**Fig. 1** Domain geometry (a) and control net for the construction of the NURBS representation (b)

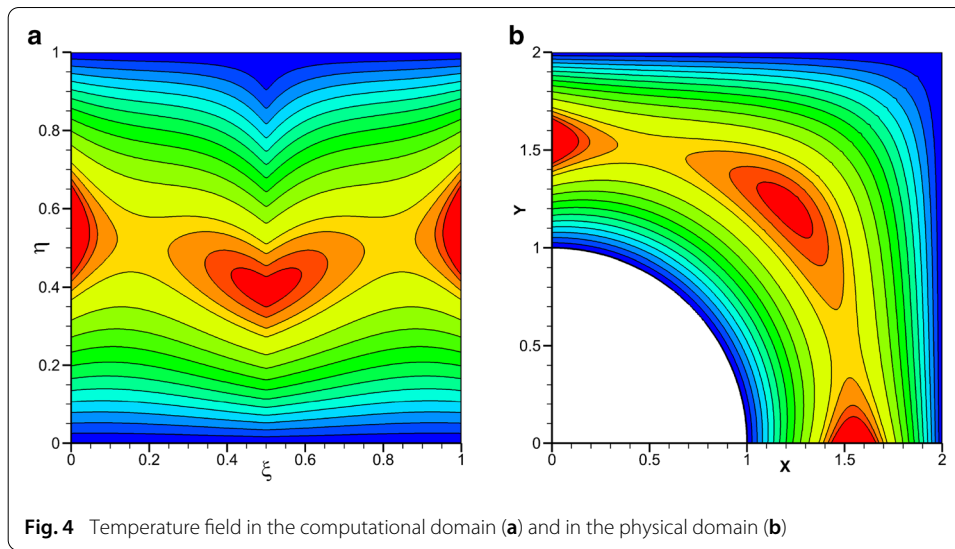
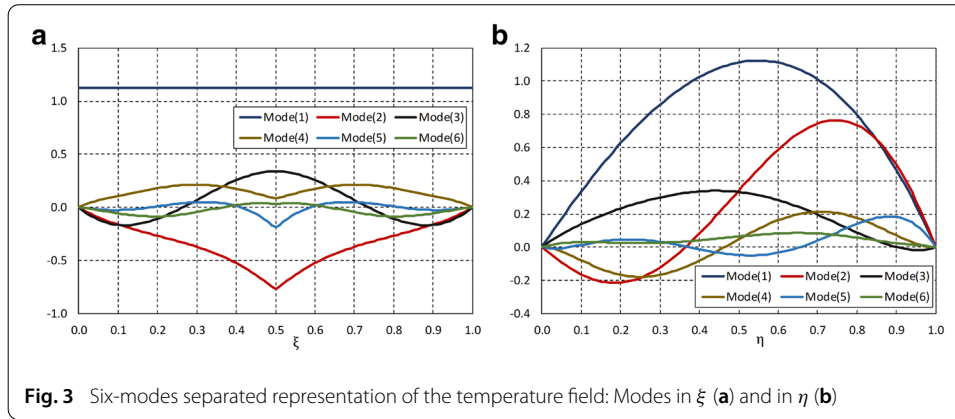


**Fig. 2** Four-modes ( $i = 1, 2, 3, 4$ ) SAR of the NURBS-based mapping: **a**  $M_1^T(\xi)X_{1i}$ ; **b**  $M_2^T(\eta)X_{2i}$ ; **c**  $M_1^T(\xi)Y_{1i}$ ; and **d**  $M_2^T(\eta)Y_{2i}$

A four-modes SAR of the domain geometry is shown in Fig. 2.

Then, by considering the boundary conditions on the different boundary segments reported in Fig. 1a

$$T(x, y) = \begin{cases} 0 & \text{on } AB \\ (4 - 2x)(x^2 - 1) & \text{on } BC \\ 0 & \text{on } CDE \\ (4 - 2y)(y^2 - 1) & \text{on } AE \end{cases}, \quad (27)$$



and the temperature dependency of the thermal conductivity expressed by

$$K(T) = 1 + T, \quad (28)$$

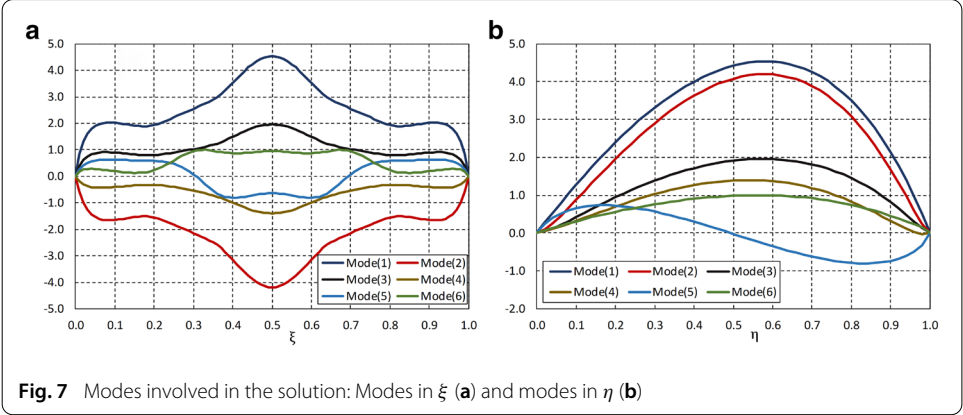
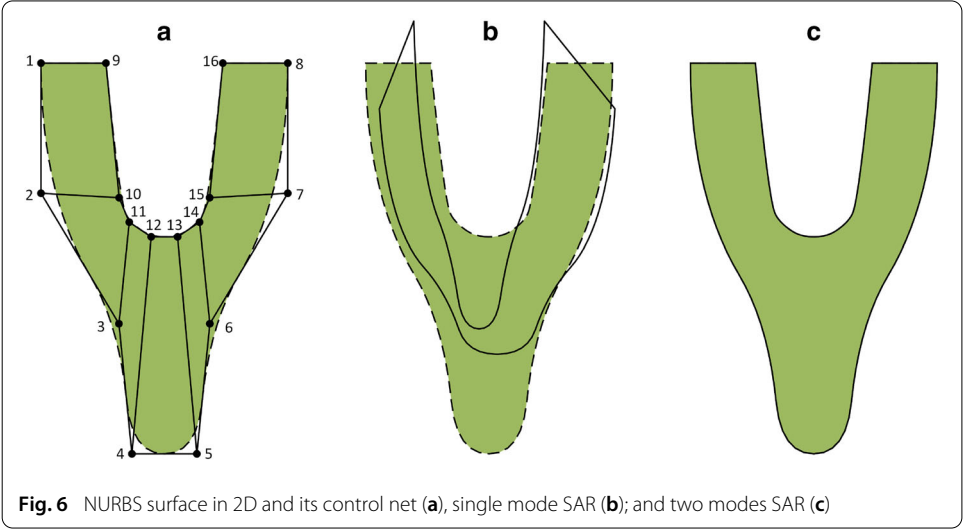
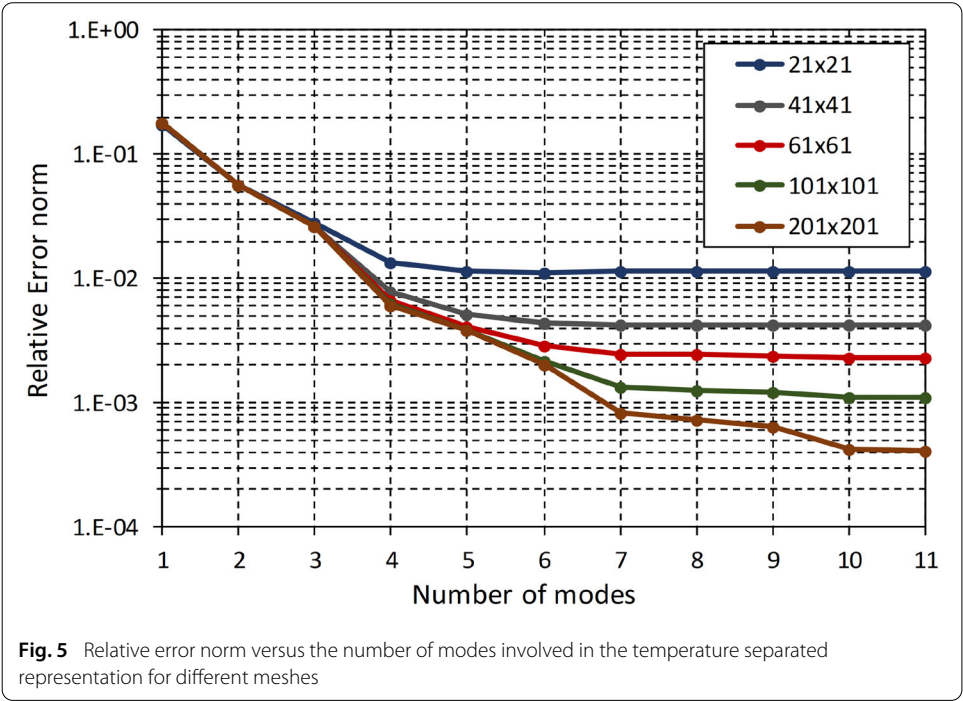
the exact solution reads

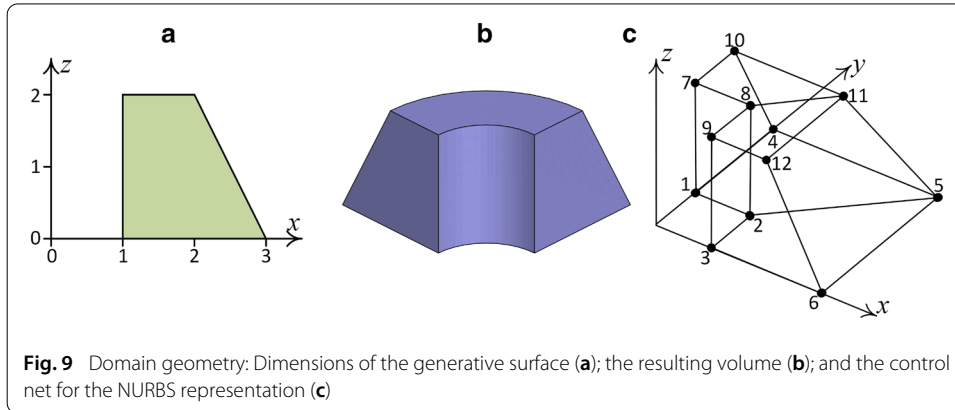
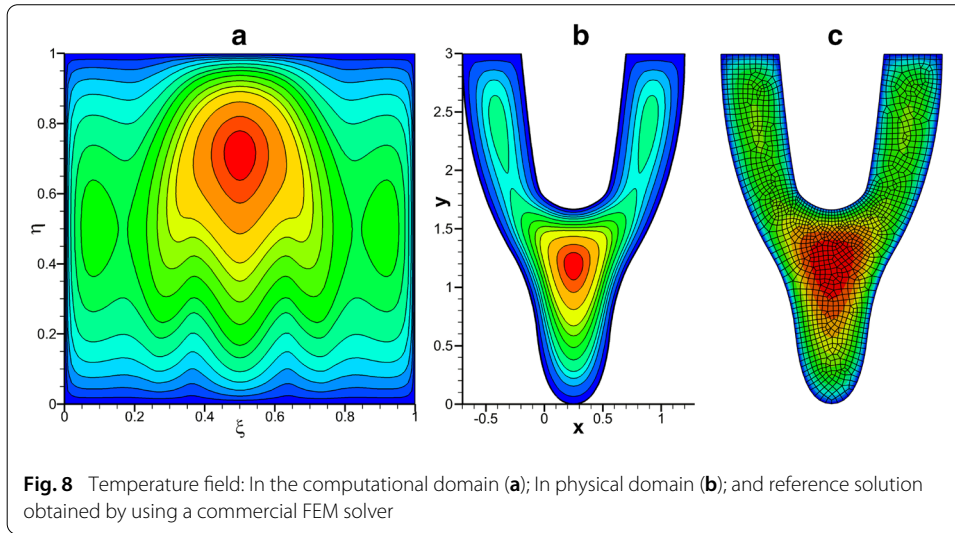
$$T(x, y) = (x - 2)(y - 2)(x^2 + y^2 - 1). \quad (29)$$

The forcing function  $f(x, y)$  can be obtained by substitution of the exact solution given by Eq. (29) into the problem model expressed by Eq. (6). The problem is then solved by using the proposed methodology whose first 6 modes are shown in Fig. 3.

The temperature distribution in the computational and physical domains is shown in Fig. 4. To evaluate the accuracy of the method and the effect the number of modes the problem is solved for different number of nodes and in each case the error norm with respect to the exact solution is computed using Eq. (24) and it is reported in Fig. 5.

The results show that the method converges very rapidly and the solution only requires few modes. As expected the solution accuracy increases by considering finer meshes, with no impact on the computational performances because of the use of separated representation that only solves one-dimensional problems.





## 2D complex geometry

In the second example, we consider the geometry given in Fig. 6, whose left image represents a 2D NURBS surface and its control net. When considering second order B-splines along the direction  $\xi$  ( $p = 2$ ) and one along direction  $\eta$  ( $q = 1$ ), and using the knots

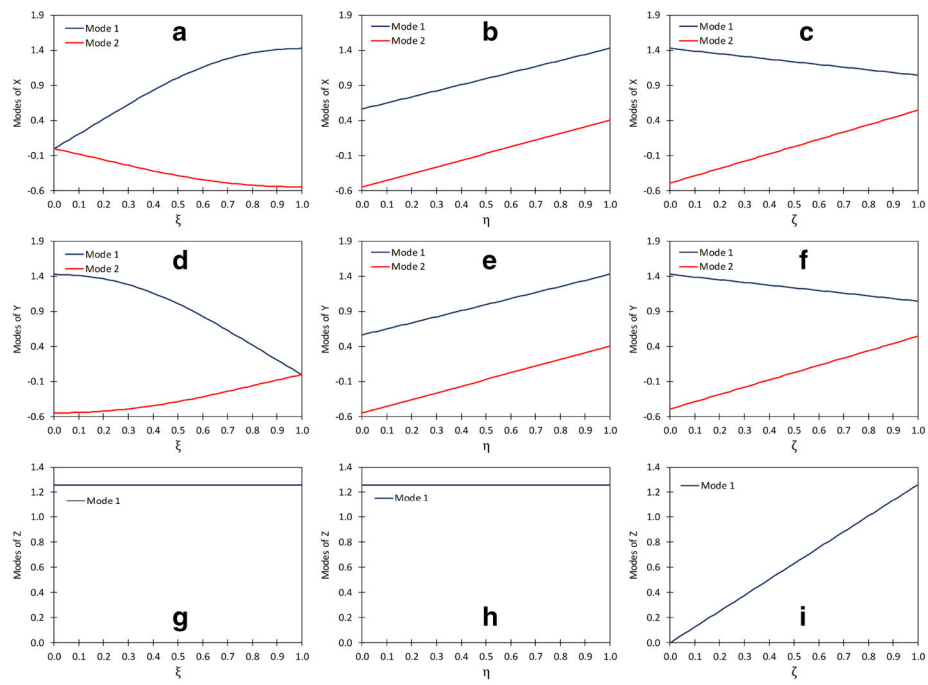
$$\kappa_{\xi} = [0, 0, 0, 0.25, 0.4, 0.5, 0.6, 0.75, 1, 1, 1]^T, \quad (30)$$

and

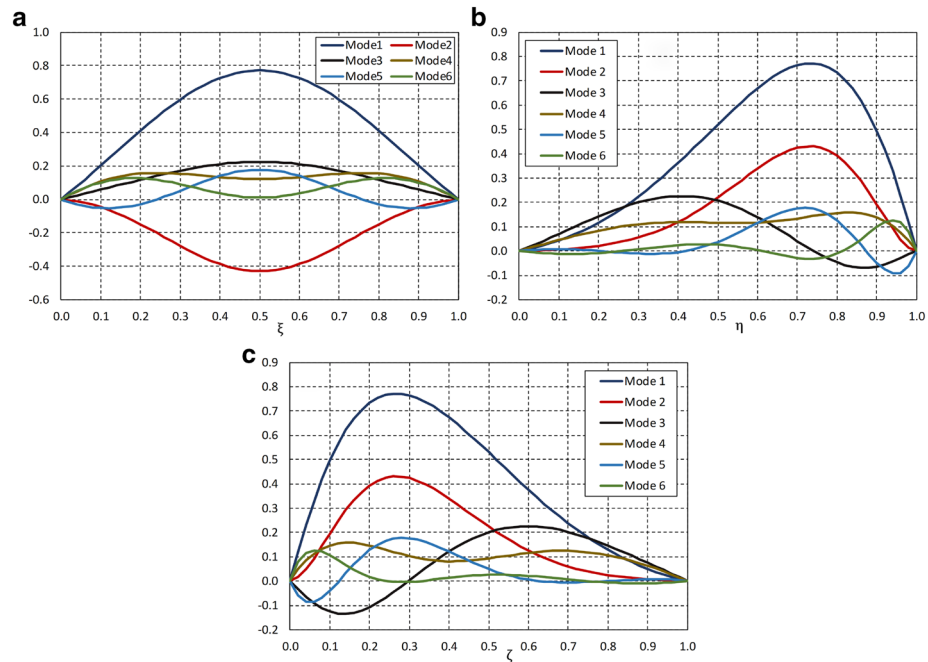
$$\kappa_{\eta} = [0, 0, 1, 1]^T. \quad (31)$$

the SAR of coordinate  $x(\xi)$  and  $y(\xi)$  related to a single mode, i.e.  $NX = NY = 1$ , results in the reconstructed geometry depicted in Fig. 6(centre). As it can be noticed in Fig. 6(right) two SAR modes,  $NX = NY = 2$ , suffice for generating an almost perfect representation of the mapping.

The nonlinear thermal problem is solved in the domain shown in Fig. 6 for proving the ability to perform separated representations in geometrically complex domains. Homogeneous Dirichlet boundary conditions are enforced on the domain boundary and the temperature dependent conductivity is taken the same that in the previous numerical test. A volumetric thermal source applies in the domain,  $f = 300$ . A mesh containing 61 nodes in  $\xi$  and 31 in  $\eta$  where considered. Even if a standard mesh-based discretization

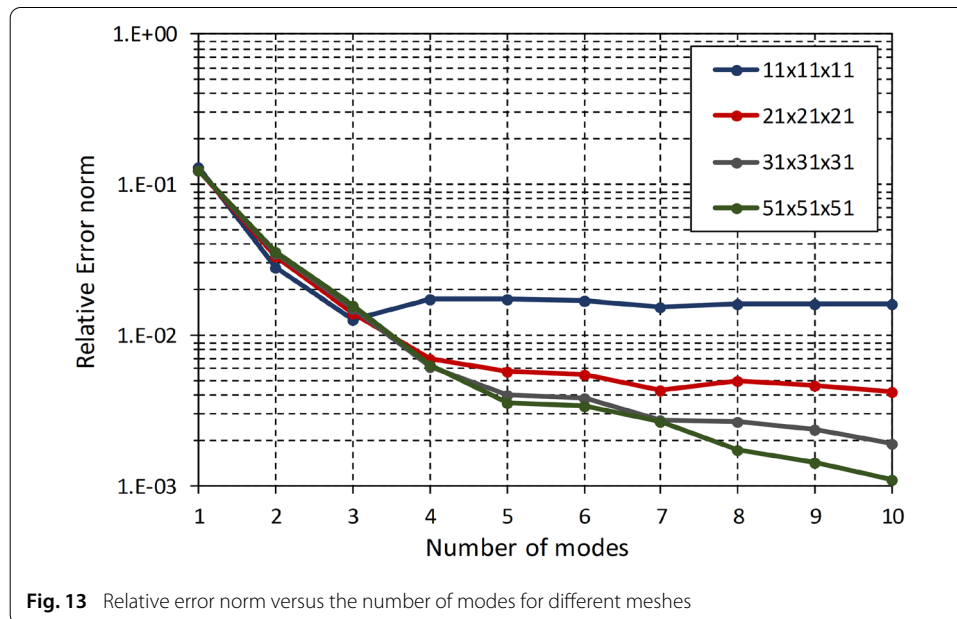
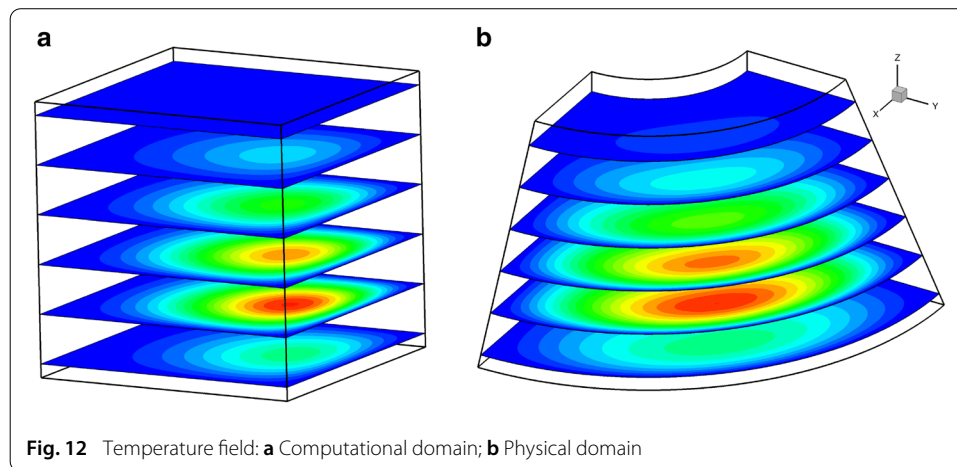


**Fig. 10** SAR of the NURBS-based mapping: **a**  $M_1^T(\xi)X_{1i}$ ; **b**  $M_2^T(\eta)X_{2i}$ ; **c**  $M_3^T(\zeta)X_{3i}$ ; **d**  $M_1^T(\xi)Y_{1i}$ ; **e**  $M_2^T(\eta)Y_{2i}$ ; **f**  $M_3^T(\zeta)Y_{3i}$ ; **g**  $M_1^T(\xi)Z_{1i}$ ; **h**  $M_2^T(\eta)Z_{2i}$ ; **i**  $M_3^T(\zeta)Z_{3i}$



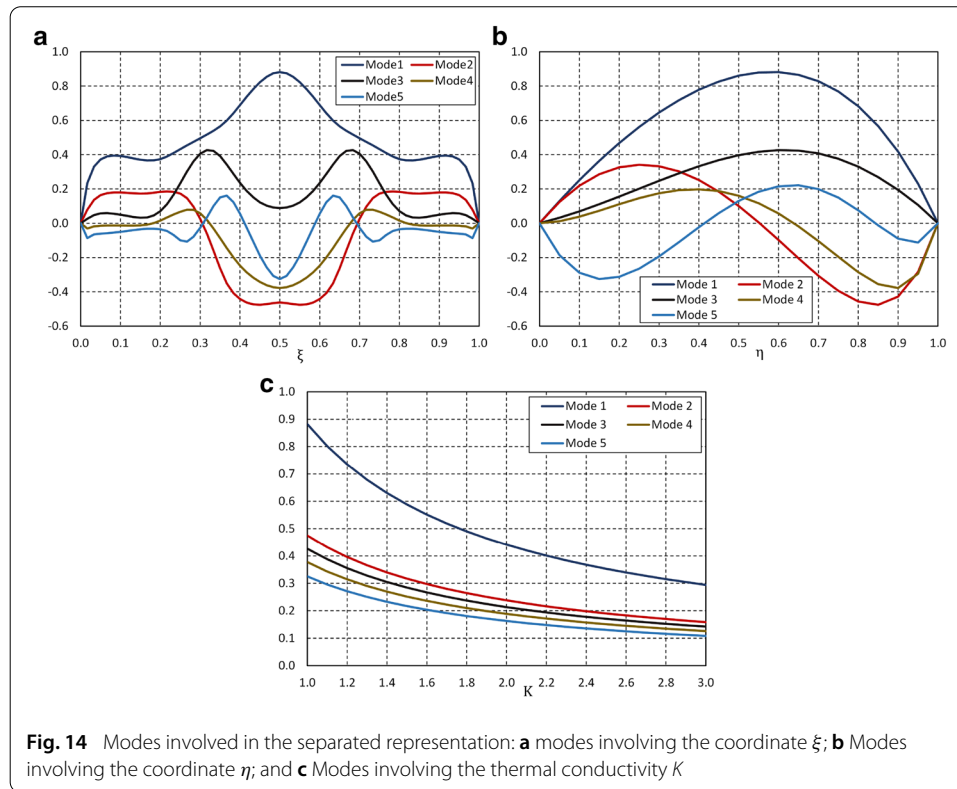
**Fig. 11** Six modes involved in the solution: **a** Modes in  $\xi$ ; **b** modes in  $\eta$ ; and **c** modes in  $\zeta$





would imply  $61 \times 31$  nodes (even more if operating directly in the physical domain for accurately adapting to the domain geometry), the separated representation only needs solving a sequence of 1D problems with 61 degrees of freedom or 31 depending on the direction considered,  $\xi$  and  $\eta$  respectively.

Again 6 modes (depicted in Fig. 7) were enough for representing accurately the solution, when compared with a standard FEM solution on a mesh consisting of quadratic elements and involving 4767 nodes. The converged temperature field is shown in Fig. 8 in the computational and physical domains, with the FEM reference solution. The maximum relative error concerning the maximum temperature is of 0.24% that can be considered excellent, and it could be reduced even more by considering either finer meshes or richer separated approximations.



### 3D geometry

In the present case, the thermal problem is solved in the 3D domain generated by rotating the 2D surface depicted in Fig. 9a around the  $z$  axis. The generated 3D volume is shown in Fig. 9b. That volume can be easily represented by using NURBS with 12 control points. The control net is shown in Fig. 9c. The order of the B-splines are  $p = 2$  in  $\xi$ ,  $q = 1$  in  $\eta$  and  $r = 1$  in  $\zeta$  with the knot vectors given by

$$\begin{cases} \kappa_{\xi} = [0, 0, 0, 1, 1, 1]^T \\ \kappa_{\eta} = [0, 0, 1, 1]^T \\ \kappa_{\zeta} = [0, 0, 1, 1]^T \end{cases} \quad (32)$$

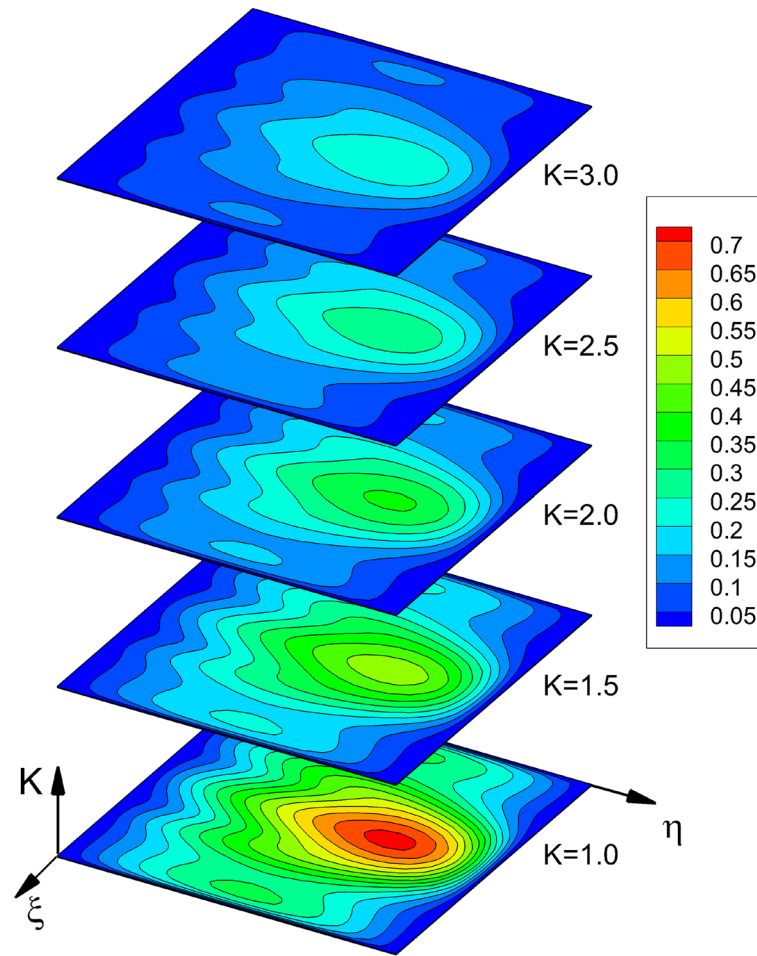
The weighting coefficients corresponding to control points 2, 5, 8 and 11 are  $\cos(\frac{\pi}{4})$  and the weighting coefficients of the rest of control points are unit values. This selection of weights ensures the exact representation of the conic surface [30].

The modes of the separated representation of the physical coordinates  $x$ ,  $y$  and  $z$  in terms of the computational coordinates  $\xi$ ,  $\eta$  and  $\zeta$  are shown in Fig. 10. To represent the domain geometry, only two modes for  $x$  and  $y$  and one mode for  $z$  suffice.

Homogeneous Dirichlet boundary conditions are applied on the domain boundary. The thermal conductivity is again the same than the one considered in the previous examples. The exact temperature distribution was enforced to be

$$T(x, y, z) = \frac{1}{50}xyz(z-2)(x^2+y^2-1)\left(x^2+y^2-\left(3-\frac{1}{2}z\right)^2\right), \quad (33)$$

needing for a volumetric thermal source that results from introducing the temperature field (33) into the problem strong form (6).

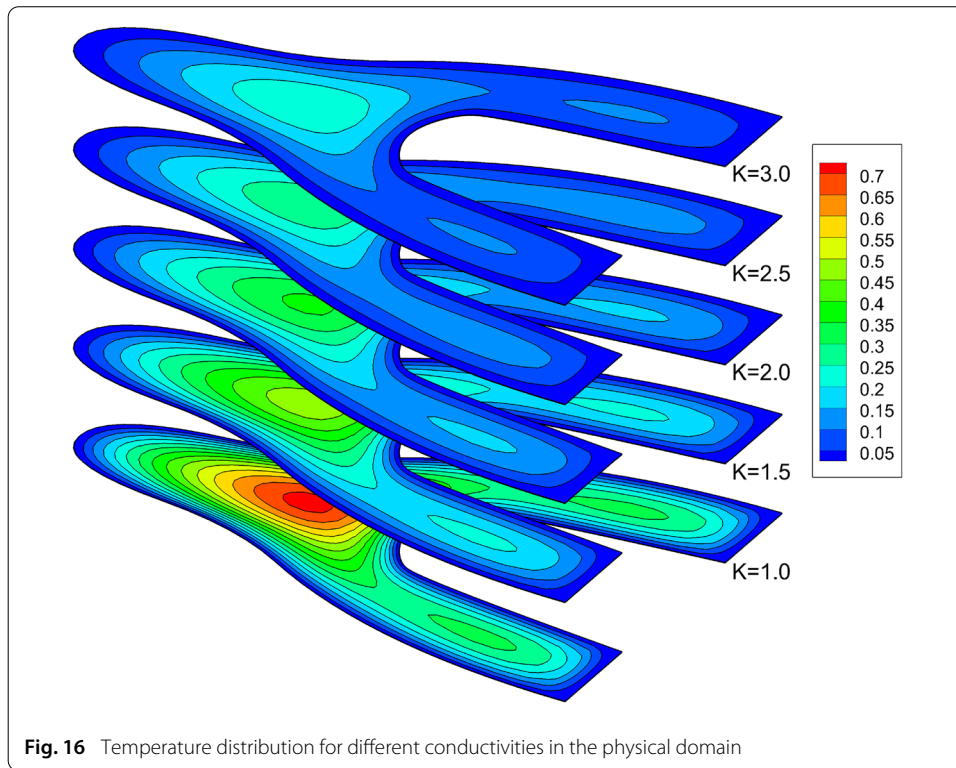


**Fig. 15** Temperature distribution for different conductivities in the computational domain

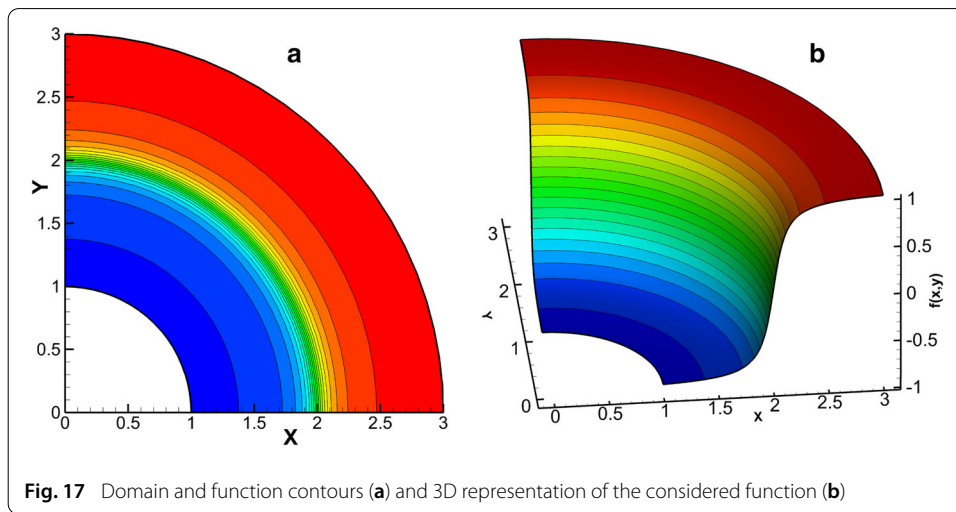
The solution involves the six modes shown in Fig. 11. The solution is depicted on some domain cross-sections of the computational domain in Fig. 12a whereas the corresponding solution in the physical domain is shown in Fig. 12b. The problem is solved for different number of modes and nodes, with the relative error norm, obtained by using Eq. (24), reported in Fig. 13.

#### Parametric solutions

The present section addresses the case in which the PGD operates first, as previously discussed, for separating the space coordinates in the computational domain, and then for including the material conductivity as an extra-coordinate. For that purpose we consider the problem considered in Section with now the conductivity as extra-coordinate. Thus, we are interested in computing the temperature field at each point and for any value of the thermal conductivity (linear, isotropic and uniformly distributed in the physical domain), taking values in the interval  $K \in [1, 3]$ .



**Fig. 16** Temperature distribution for different conductivities in the physical domain



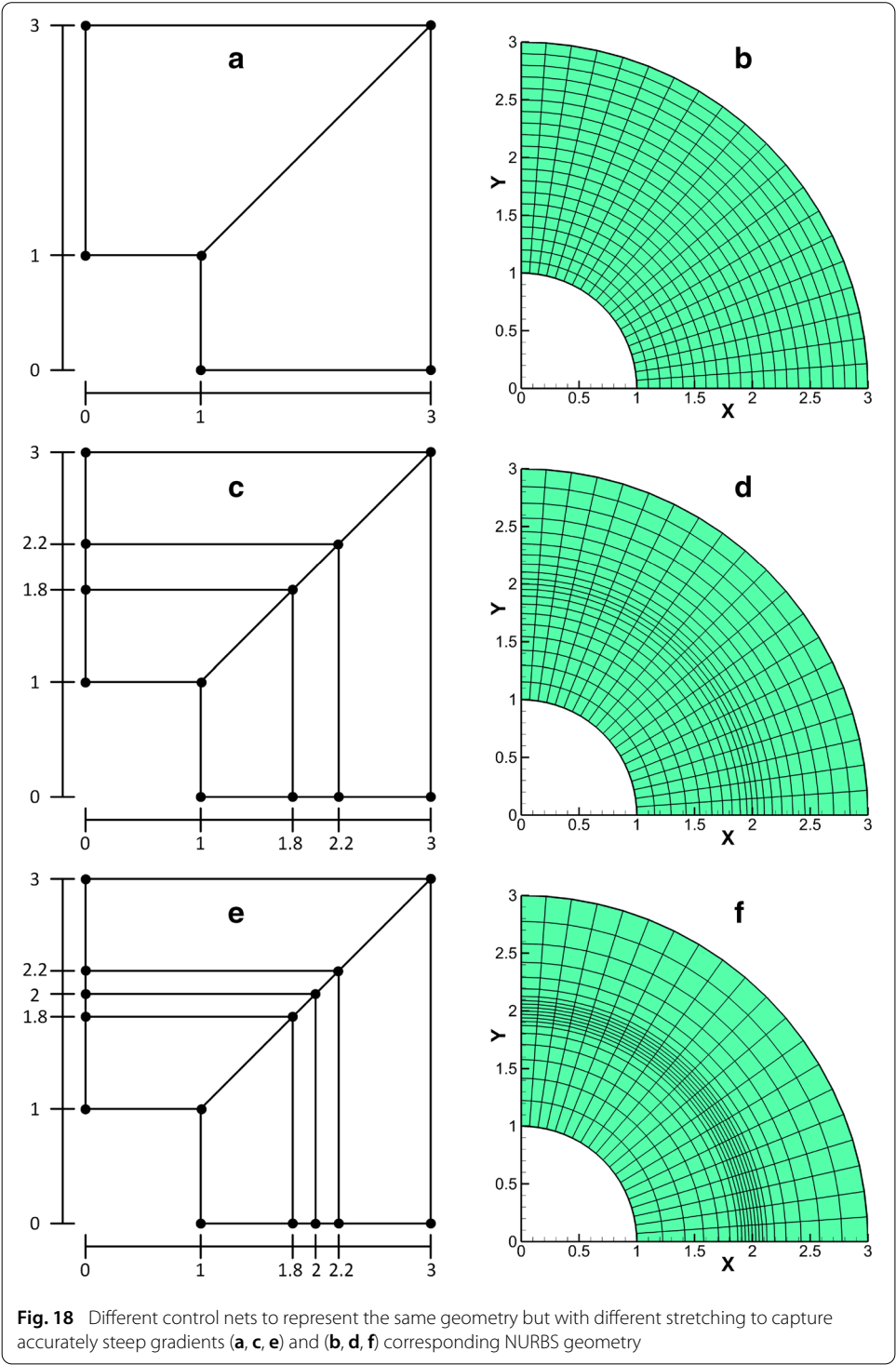
**Fig. 17** Domain and function contours (a) and 3D representation of the considered function (b)

Figure 14 represents the different modes involved in the solution separated representation, whereas Figs. 15 and 16 show the solution in both, the computational and the physical domains respectively.

#### Describing a solution exhibiting a steep gradient

To emphasize the performances of the proposed approach for describing at very low cost complex geometries in a very accurate way, in this section we consider the domain depicted in Fig. 17a, where the solution in each point is described by function  $f(x, y)$

$$f(x, y) = \frac{2}{\pi} \operatorname{atan} (3(x^2 + y^2 - 4)), \quad (34)$$



**Fig. 18** Different control nets to represent the same geometry but with different stretching to capture accurately steep gradients (a, c, e) and (b, d, f) corresponding NURBS geometry

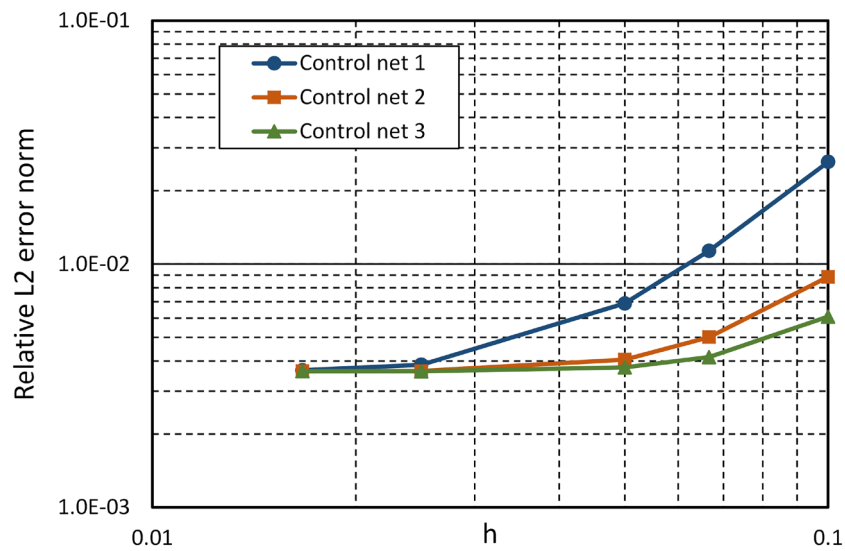


Fig. 19 Convergence analysis

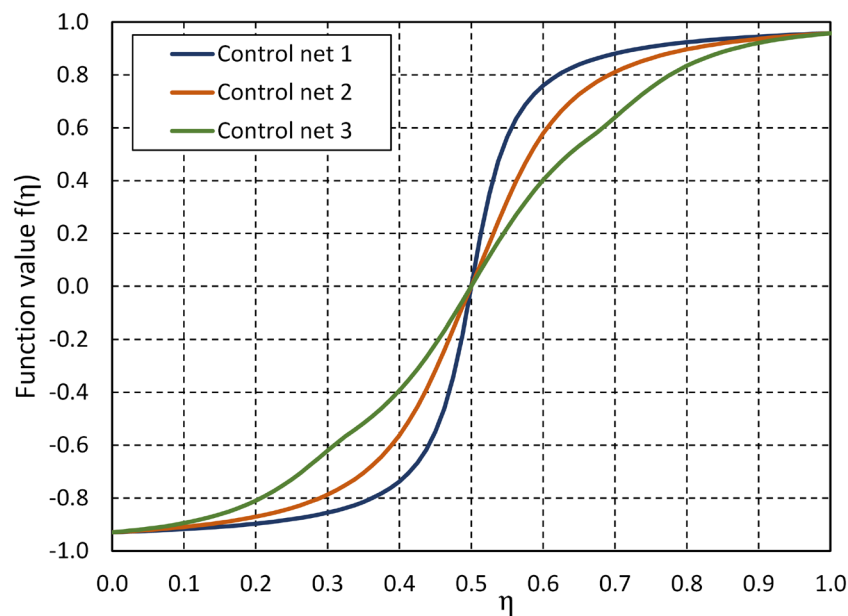
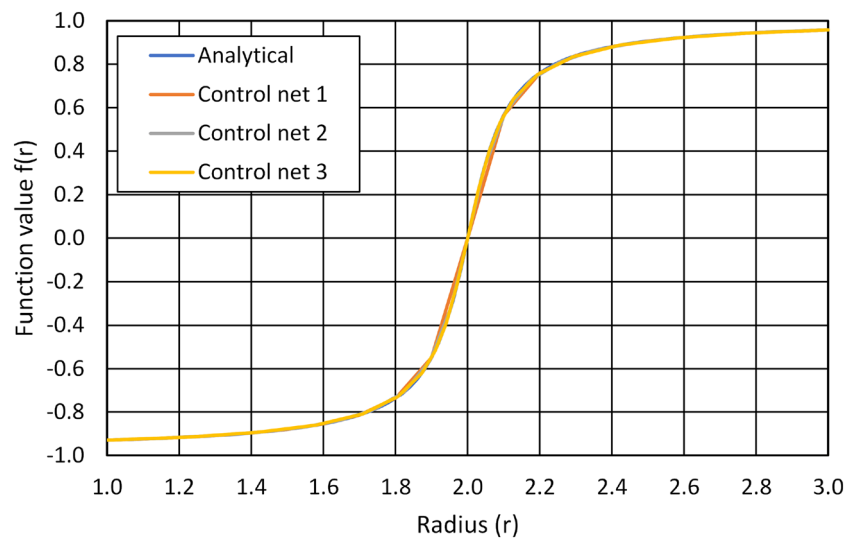


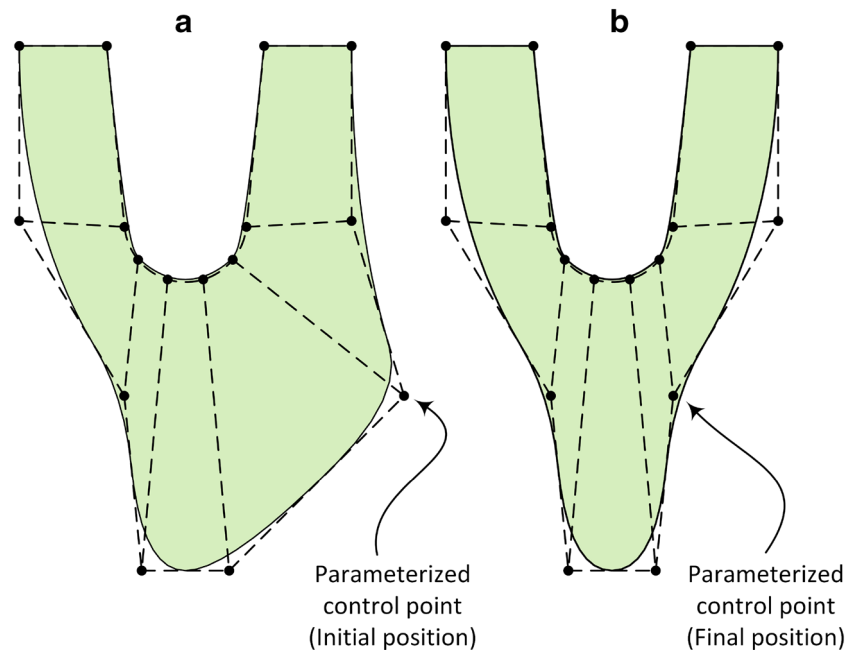
Fig. 20 Function distribution along the coordinate  $\eta$

depicted in Fig. 17b. As it can be noticed the solution exhibits a high gradient when the radius approaches  $r = 2$ .

Figure 18 depicts different geometrical descriptions enabling finer meshes in the region where the solution exhibits higher gradients ( $r = 2$ ). Figure 19 depicts the convergence in the solution representation when decreasing the mesh size in the three considered control nets. Finally Figs. 20 and 21 compare the solutions along the  $\eta$  coordinate (in the computational domain) and the radial one respectively, proving an excellent accuracy.



**Fig. 21** Solution  $f(r)$ , with  $r$  the radius

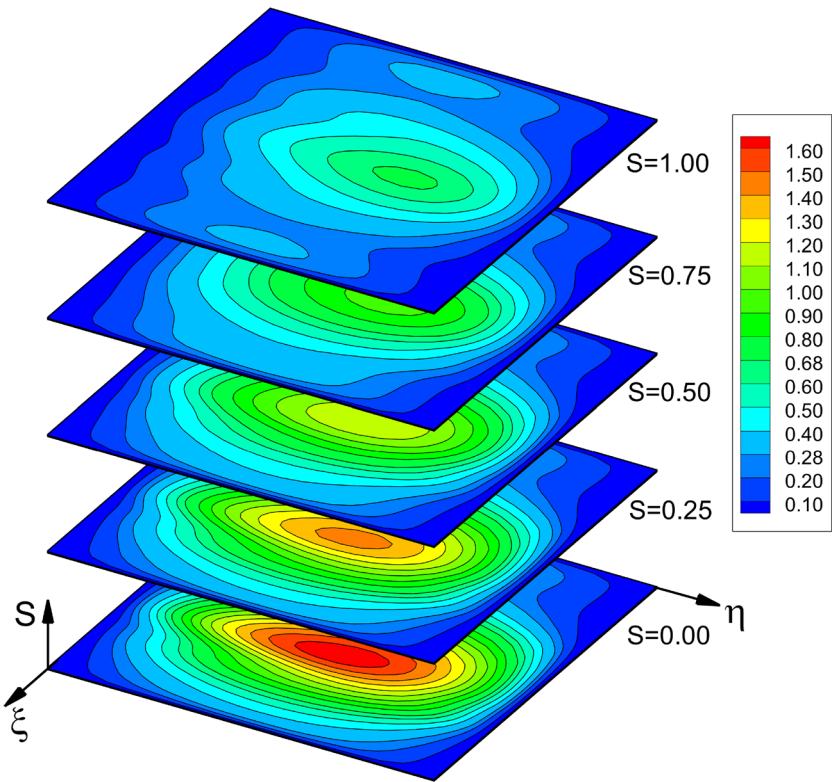


**Fig. 22** Geometry parameterization of a complex Y shaped domain: extreme positions of the moving control point

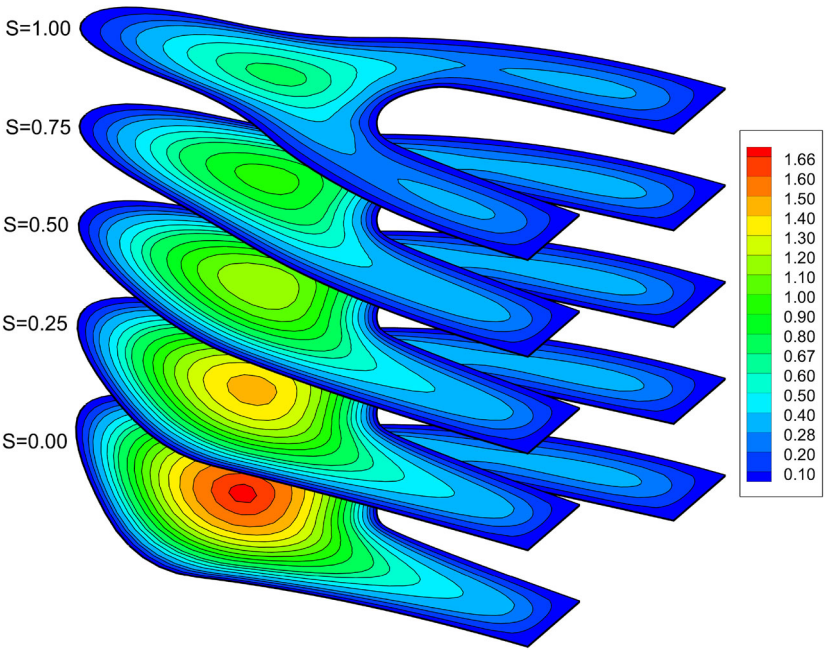
### Parametrized shapes

In this section instead of using as extra-coordinate a parameter related to the physics itself (the conductivity in Section ), here we consider the control point positions as parameters, and then, within the PGD rationale, as extra-coordinates, for in this manner computing from one shot the solution for any possible geometry (shape) according to the considered parameterization.



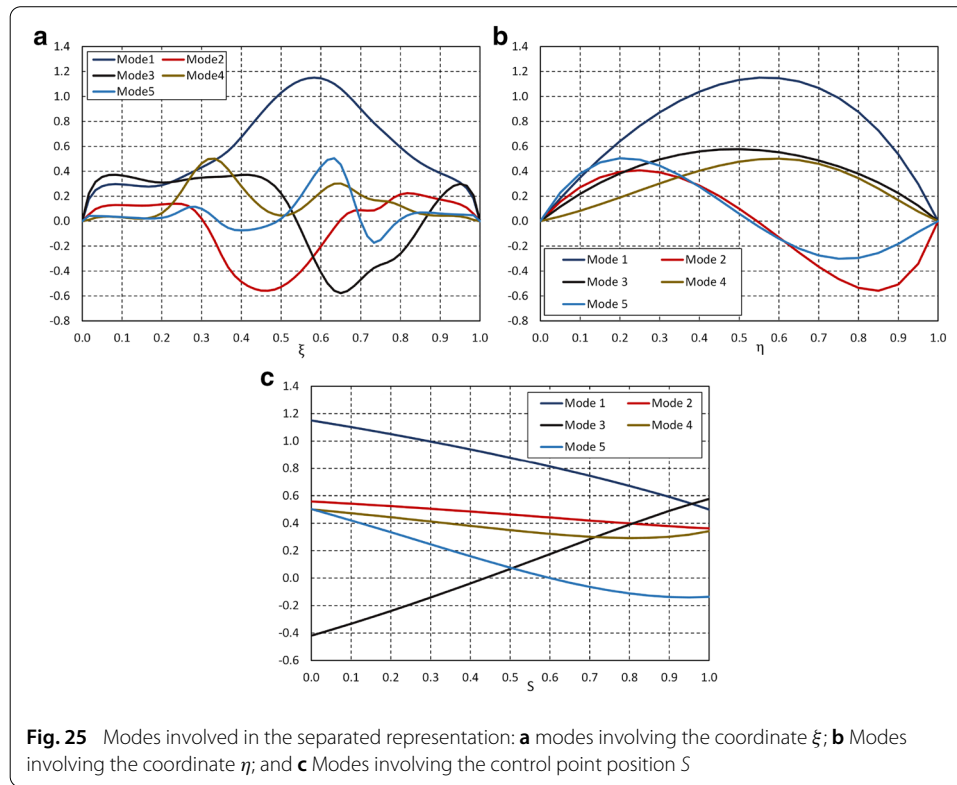


**Fig. 23** Temperature field in the computational domains particularized for different positions  $S$  of the control point indicated in Fig. 22, that is, for different shapes of the physical domain



**Fig. 24** Solutions in the physical domain related to those shown in Fig. 23





For that purpose, we consider the Y-shaped domain already introduced in section , where the location of one control point is considered as a parameter, that is, as a model extra-coordinate within the PGD rationale. The extreme positions of that control point and the corresponding geometries, are shown in Fig. 22. The temperature distribution in the computational and physical domains are given respectively in Figs. 23 and 24 on different slices corresponding to different values of the coordinate  $S$  (representing the control point position). Finally, the first 5 modes of the separated representation, concerning the different coordinates, are depicted in Fig. 25.

## Conclusions

PGD makes possible advanced parametric analyses, by considering parameters as model extra-coordinates, while circumventing the resulting curse of dimensionality by using separated representations. When applying separated representation to the space coordinates, the solution of 3D problems becomes a sequence of one-dimensional problems as soon as the 3D domain can be expressed in a separated form (hexahedra). This fact limits significantly the domain of applicability of space separated representations.

The present work enlarges the applicability of the PGD rationale to general non-regular geometries, expressed by using NURBS-based representations, very usual in CAD technologies. A separated representation is then proposed for the geometrical transformation, enabling the transformation of a 3D problem in a complex geometry into a sequence of one-dimensional problems solution. Many examples served to prove the capabilities and potential of the proposed methodology.

More complex geometries need multiple NURBS for approximating them. In particular, these complex geometries can be decomposed in simpler subdomains, each one approximated as described in the present paper, with some constraints applied on the NURBS, for taking into account the different transmission conditions on the interfaces. This topic is addressed in the second part of the present paper, that is being finalized.

#### Acknowledgements

Authors acknowledge the support of the ESI Group through its research chair at ENSAM ParisTech.

#### Authors' contributions

All the authors participated in the definition of techniques and algorithms. All authors read and approved the final manuscript.

#### Availability of data and materials

Interested reader can contact authors.

#### Declarations

##### Competing interests

The authors declare that they have no competing interests.

##### Author details

<sup>1</sup>Department of Mechanical Engineering, Shiraz Branch, Islamic Azad University, Shiraz, Iran, <sup>2</sup>LAMPA, Arts et Metiers Institute of Technology, Angers, France, <sup>3</sup>Innovation Department, ESI Group, Rungis, France, <sup>4</sup>PIMM Lab & ESI Chair, Arts et Metiers Institute of Technology, Paris, France.

## Appendix A: NURBS fundamentals

NURBS enable exactly representing the geometry of most curves, surfaces and volumes encountered in engineering applications, being nowadays a usual technology employed in CAD.

For the sake of completeness this section revisits the main concepts involved in the NURBS construction. For additional details the interested reader can refer to [30].

NURBS result from weighted combinations of B-spline functions. To define a set of  $n$  B-spline functions of order  $p$  in a univariate parametric space  $\xi \in [0, 1]$ , the knot vector  $\kappa_\xi$  is defined as follows

$$\kappa_\xi = [\xi_1, \xi_2, \dots, \xi_{n+p+1}]^T. \quad (35)$$

Denote  $N_{ap}(\xi)$  as the B-spline basis function of order  $p$  in the  $a$ -th knot span  $\xi \in [\xi_a, \xi_{a+1})$ . The following recursive equations can be used to compute the univariate B-spline basis function  $N_{ap}(\xi)$  [30]

$$N_{a0}(\xi) = \begin{cases} 1 & \xi_a \leq \xi < \xi_{a+1} \\ 0 & \text{otherwise} \end{cases}, \quad (36)$$

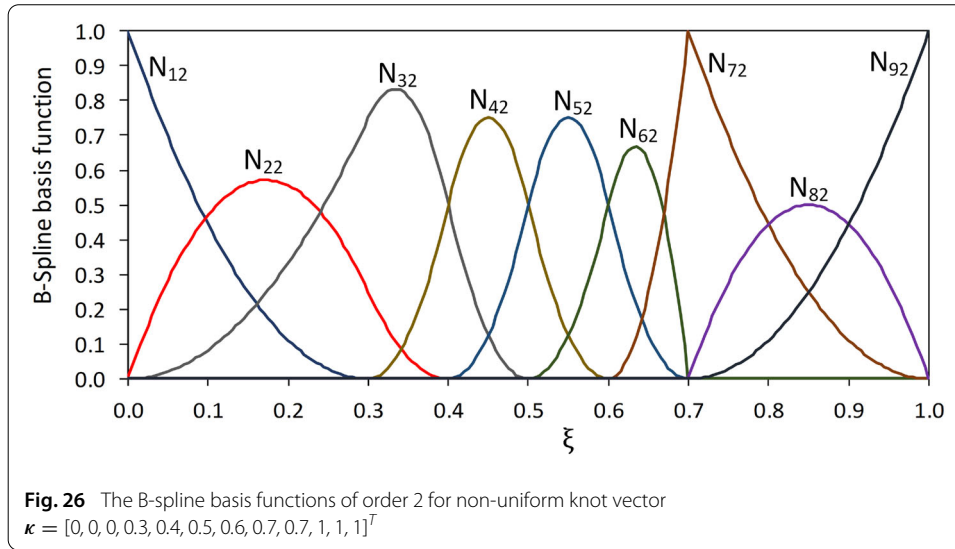
for  $p = 0$ , and for  $p > 0$

$$N_{ap}(\xi) = \frac{\xi - \xi_a}{\xi_{a+p} - \xi_a} N_{a(p-1)} + \frac{\xi_{a+p+1} - \xi}{\xi_{a+p+1} - \xi_{a+1}} N_{(a+1)(p-1)} \quad (37)$$

For illustrative purposes a set of basis functions of order 2 using a non-uniform knot vector  $\kappa_\xi = [0, 0, 0, 0.3, 0.4, 0.5, 0.6, 0.7, 0.7, 1, 1, 1]^T$  is shown in Fig. 26.

A proper choice of the knot vectors allows obtaining rich behavior of the basis functions and enough flexibility to describe complex geometries. For example, the basis functions given in Fig. 26 define an interpolation at the ends of the interval  $[0, 1]$  and also at the position of  $\xi = 0.7$  due to the repeated knot at these locations.

The NURBS basis functions can be obtained using a rational weighted sum of the B-splines basis functions. With  $w_i$  the weight, the univariate NURBS basis function,  $R_i^p(\xi)$



reads [30]

$$R_i^p(\xi) = \frac{N_{ip}(\xi) w_i}{\sum_{\alpha=1}^n N_{\alpha p}(\xi) w_{\alpha}}. \quad (38)$$

The bivariate NURBS basis functions  $R_{ij}^{pq}(\xi, \eta)$  in 2D domains and trivariate NURBS basis functions  $R_{ijk}^{pqr}(\xi, \eta, \zeta)$  in 3D domains can be obtained using tensor product of univariate basis functions

$$R_{ij}^{pq}(\xi) = \frac{N_{ip}(\xi) N_{jq}(\eta) w_{ij}}{\sum_{\alpha=1}^n \sum_{\beta=1}^m N_{\alpha p}(\xi) N_{\beta q}(\eta) w_{\alpha\beta}}, \quad (39)$$

and

$$R_{ijk}^{pqr}(\xi) = \frac{N_{ip}(\xi) N_{jq}(\eta) N_{kr}(\zeta) w_{ijk}}{\sum_{\alpha=1}^n \sum_{\beta=1}^m \sum_{\gamma=1}^l N_{\alpha p}(\xi) N_{\beta q}(\eta) N_{\gamma r}(\zeta) w_{\alpha\beta\gamma}}, \quad (40)$$

respectively.

In the previous equations  $p$ ,  $q$ , and  $r$  are the order of B-splines in directions  $\xi$ ,  $\eta$ , and  $\zeta$ , respectively and  $\xi = (\xi, \eta, \zeta)$  is the coordinates vector in the computational domain, and  $w_{ijk}$  denote the weighting parameters. All the univariate basis functions are defined based on the B-splines defined in their respective coordinates. The knot vector  $\kappa_{\eta}$  and  $\kappa_{\zeta}$  applying in the coordinates  $\eta$  and  $\zeta$  read

$$\kappa_{\eta} = [\eta_1, \eta_2, \dots, \eta_{m+q+1}]^T \quad \eta \in [0, 1], \quad (41)$$

and

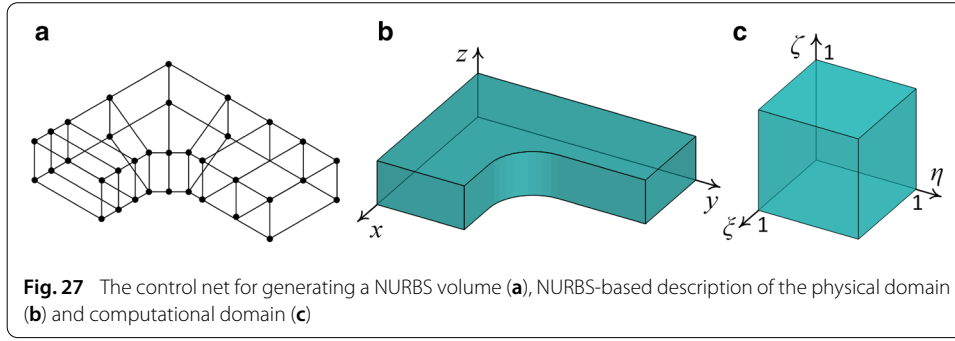
$$\kappa_{\zeta} = [\zeta_1, \zeta_2, \dots, \zeta_{l+r+1}]^T \quad \zeta \in [0, 1], \quad (42)$$

respectively, being  $n$ ,  $m$  and  $l$  the number of basis functions in directions  $\xi$ ,  $\eta$  and  $\zeta$  respectively.

Using the NURBS basis functions just described, the single variate curve  $C$ , bivariate surface  $S$  and trivariate volume  $V$  can be described from

$$\mathbf{x} = \sum_{i=1}^n R_i^p(\xi) \mathbf{P}_i \quad \mathbf{x} \in C, \quad (43)$$

$$\mathbf{x} = \sum_{i=1}^n \sum_{j=1}^m R_{ij}^{pq}(\xi) \mathbf{P}_{ij} \quad \mathbf{x} \in S, \quad (44)$$



**Fig. 27** The control net for generating a NURBS volume (a), NURBS-based description of the physical domain (b) and computational domain (c)

and

$$\mathbf{x} = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^l R_{ijk}^{pqr}(\boldsymbol{\xi}) \mathbf{P}_{ijk} \quad \mathbf{x} \in V, \quad (45)$$

respective, where,  $\mathbf{P}$  refers to the control points (the vertices of the so-called control net) in the physical domain.

Equations (43)–(45) represent the application that maps any point  $\boldsymbol{\xi}$  in the computational domain  $\Omega_{\boldsymbol{\xi}}$  to a point  $\mathbf{x}$  in physical domain  $\Omega_{\mathbf{x}}$ .

Figure 27 illustrates the just described construction, where the control net and control points are shown in the left, the physical domain  $\Omega_{\mathbf{x}}$  described using the NURBS-based mapping (45), in the central image, and the corresponding computational domain  $\Omega_{\boldsymbol{\xi}}$ , the unit cube shown in the right image.

Now, using the notation  $\mathbf{x}^T = (x, y, z) \equiv (x_1, x_2, x_3)$  and  $\boldsymbol{\xi}^T = (\xi, \eta, \zeta) \equiv (\xi_1, \xi_2, \xi_3)$ , for facilitating the mapping description compactness, the terms involved in the transformation of the differential operator from  $\Omega_{\mathbf{x}}$  to  $\Omega_{\boldsymbol{\xi}}$  read

$$\frac{\partial x_a}{\partial \xi_b} = \sum_{i=1}^n \sum_{j=1}^m \sum_{k=1}^l \frac{\partial R_{ijk}^{pqr}(\boldsymbol{\xi})}{\partial \xi_b} P_{ijk}^a, \quad a, b = 1, 2, 3, \quad (46)$$

where  $P_{ijk}^a$  refers to the  $a$ -component of  $\mathbf{P}_{ijk}$ .

The previous equation involves the NURBS derivatives. Thus, for example the derivative of  $R_i^p(\boldsymbol{\xi})$  reads

$$\frac{d}{d\xi} R_i^p(\boldsymbol{\xi}) = w_i \frac{N'_{ip}(\boldsymbol{\xi}) W(\boldsymbol{\xi}) - N_{ip}(\boldsymbol{\xi}) W'(\boldsymbol{\xi})}{W(\boldsymbol{\xi})^2} \quad (47)$$

where,  $W(\boldsymbol{\xi})$  is the denominator of Eq. (38). The derivatives of bivariate and trivariate basis functions can also be obtained following the same rationale, and can be found in [30].

As it can be noticed, the main drawback of the proposed mapping is that expressions like (47) have not a separate structure, compromising the effectiveness of the PGD solver. For that purpose we apply a separated approximate representation described in B.

## Appendix B: SAR constructor

Consider function  $g(\boldsymbol{\xi})$ . It might be given analytically or it may be possible to numerically evaluate it at any point  $\boldsymbol{\xi}$ . We are interested in obtaining a SAR of  $g(\boldsymbol{\xi})$  in terms of a set of approximation functions in different directions and a set of coefficients as given in Eq. (13). Any approximation basis could be considered in  $\mathbf{M}$  to find the coefficients in  $\mathbf{G}$  related to a finite sum involving  $NG$  terms providing an approximation accurate enough.

We proceed in an iterative manner, and we consider that we have already calculated the first  $(n - 1)$  modes and we would like to find the next mode,  $n$ .

The residual at the present iteration reads

$$r^{n-1}(\xi) = g(\xi) - \sum_{i=1}^{n-1} \prod_{j=1}^{ND} M_j^T(\xi_j) G_{ji}. \quad (48)$$

Therefore, the  $n$ -th mode must be obtained in such a way that makes an approximation for the residual, i.e.

$$\prod_{j=1}^{ND} M_j^T G_{jn} \approx r^{n-1}(\xi). \quad (49)$$

To find the coefficients  $G_{jn}$  that provides the best approximation of the residual  $r^{n-1}(\xi)$ , the error

$$\|e\| = \int_{\Omega_\xi} \left( \prod_{j=1}^{ND} M_j^T G_{jn} - r^{n-1}(\xi) \right)^2 d\xi, \quad (50)$$

is enforced to be minimum by prescribing null derivatives

$$\frac{\partial \|e\|}{\partial G_{1n}^\alpha} = 0, \quad \frac{\partial \|e\|}{\partial G_{2n}^\beta} = 0 \quad \& \quad \frac{\partial \|e\|}{\partial G_{3n}^\gamma} = 0, \quad (51)$$

where,  $G_{1n}^\alpha$ ,  $G_{2n}^\beta$  and  $G_{3n}^\gamma$  refer to the  $\alpha$ -th,  $\beta$ -th and  $\gamma$ -th component of corresponding vectors.

Consider, for instance

$$\frac{\partial \|e\|}{\partial G_{1n}^\alpha} = 0, \quad (52)$$

it results

$$\begin{aligned} & \int_{\Omega_\xi} M_1^\alpha \left( M_1^T G_{1n} \right) \left( M_2^T G_{2n} \right)^2 \left( M_3^T G_{3n} \right)^2 d\xi \\ &= \int_{\Omega_\xi} M_1^\alpha \left( M_2^T G_{2n} \right) \left( M_3^T G_{3n} \right) r^{n-1}(\xi) d\xi, \end{aligned} \quad (53)$$

that reduces to

$$\begin{aligned} & \left( \int_0^1 M_1^\alpha M_1 d\xi \right)^T G_{1n} \\ &= \frac{\int_0^1 \int_0^1 \int_0^1 M_1^\alpha \left( M_2^T G_{2n} \right) \left( M_3^T G_{3n} \right) r^{n-1}(\xi) d\zeta d\eta d\xi}{\int_0^1 \left( M_2^T G_{2n} \right)^2 d\eta \int_0^1 \left( M_3^T G_{3n} \right)^2 d\zeta}, \quad \forall \alpha \end{aligned} \quad (54)$$

By considering the other two relations in Eq. (51), each one applying  $\forall \beta$  and  $\forall \gamma$  respectively, a nonlinear discrete system is obtained whose solution result in the vectors  $G_{1n}$ ,  $G_{2n}$  and  $G_{3n}$ .

### Appendix C: Separated geometry mapping

As before stated in Section , the most appealing features of the PGD is reducing of a high dimensional problem to a series of problems defined in lower dimensional spaces.

To apply this method in non-separable geometries one needs first to map the problem from the physical domain  $\Omega_x$  into the computational domain  $\Omega_\xi$ . The main issue, as

previously discussed, is the non-separable expression of the terms related to the Jacobian of the coordinate transformation.

The derivatives transform according to

$$\begin{bmatrix} \partial(\cdot)/\partial x \\ \partial(\cdot)/\partial y \\ \partial(\cdot)/\partial z \end{bmatrix} = \mathbf{h} \begin{bmatrix} \partial(\cdot)/\partial \xi \\ \partial(\cdot)/\partial \eta \\ \partial(\cdot)/\partial \zeta \end{bmatrix}, \quad (55)$$

where,  $\mathbf{h}$  is the inverse of the Jacobian,  $\mathbf{J}$ .

As previously indicated the coordinate transformation (46) leads to non separable expressions that compromise the PGD effectiveness. An alternative procedure consists in constructing a separated representation of the coordinates mapping

$$x(\xi) = \sum_{i=1}^{NX} \prod_{j=1}^{ND} M_j^T X_{ji}, \quad (56)$$

$$y(\xi) = \sum_{i=1}^{NY} \prod_{j=1}^{ND} M_j^T Y_{ji} \quad (57)$$

and

$$z(\xi) = \sum_{i=1}^{NZ} \prod_{j=1}^{ND} M_j^T Z_{ji} \quad (58)$$

where,  $NX$ ,  $NY$  and  $NZ$  are the number of modes required to reach the desired level of accuracy in the SAR of  $x(\xi)$ ,  $y(\xi)$  and  $z(\xi)$  respectively.

Using the same rationale, we can derive the SAR of Jacobian determinant and the transformation derivatives, respectively

$$|\mathbf{J}| = \sum_{i=1}^{NJ} \prod_{j=1}^{ND} M_j^T J_{ji}, \quad (59)$$

and

$$h_{ab}(\xi) = \sum_{i=1}^{NH^{ab}} \prod_{j=1}^{ND} M_j^T H_{ji}^{ab}, \quad a, b = 1, 2, 3. \quad (60)$$

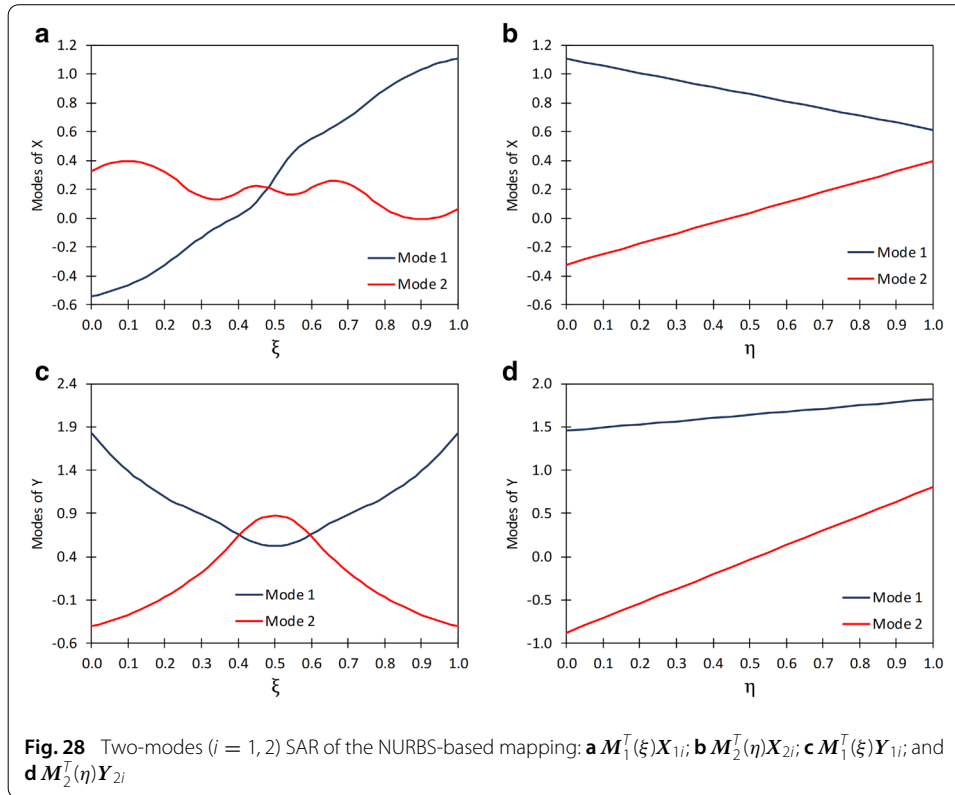
To illustrate the SAR of NURBS we consider the case given in Fig. 6, whose left image represents a 2D NURBS surface and its control net. When considering second order B-splines along the direction  $\xi$  ( $p = 2$ ) and one along direction  $\eta$  ( $q = 1$ ), and using the knots

$$\kappa_\xi = [0, 0, 0, 0.25, 0.4, 0.5, 0.6, 0.75, 1, 1, 1]^T, \quad (61)$$

and

$$\kappa_\eta = [0, 0, 1, 1]^T \quad (62)$$

the SAR of coordinate  $x(\xi)$  and  $y(\xi)$  are constructed by using the procedure just described, where a single mode representation, i.e.  $NX = NY = 1$ , results in the reconstructed geometry depicted in Fig. 6(centre). As it can be noticed in Fig. 6(right) two SAR modes  $NX = NY = 2$  suffice for generating an almost perfect representation of the mapping. The different modes involved in the approximation are depicted in Fig. 28. The maximum error of SAR of  $x$  and  $y$  are given in Table 1 proving that only two modes largely suffices for capturing the geometric details.

**Table 1** Maximum error in the separated approximate representation of coordinates  $x$  and  $y$ 

# modes	Maximum error in $x$ representation	Maximum error in $y$ representation
1	1.572e-01	7.656e-01
2	7.772e-16	3.109e-15

## Appendix D: PGD constructor

First, we define the operator

$$\widehat{\mathbf{M}}_j^a = \begin{cases} \frac{\partial \mathbf{M}_a}{\partial \xi_a} & \text{if } a = j \\ \mathbf{M}_j & \text{else} \end{cases}. \quad (63)$$

By considering the above operator the derivatives of  $T$  and  $T^{*d}$  read

$$\begin{aligned} \frac{\partial T}{\partial \xi_b} &= \frac{\partial}{\partial \xi_b} \left( \sum_{i=1}^{n-1} \prod_{j=1}^{ND} \mathbf{M}_j^T \mathbf{T}_{ji} + \prod_{j=1}^{ND} \mathbf{M}_j^T \mathbf{T}_{jn} \right) \\ &= \sum_{i=1}^{n-1} \prod_{j=1}^{ND} \widehat{\mathbf{M}}_j^T \mathbf{T}_{ji} + \prod_{j=1}^{ND} \widehat{\mathbf{M}}_j^T \mathbf{T}_{jn} \end{aligned} \quad (64)$$

and

$$\frac{\partial T^{*d}}{\partial \xi_a} = \frac{\partial}{\partial \xi_a} \left( \mathbf{T}^{*dT} \mathbf{M}_d \prod_{\substack{j=1 \\ j \neq d}}^{ND} \mathbf{M}_j^T \mathbf{T}_{jn} \right) = \mathbf{T}^{*dT} \widehat{\mathbf{M}}_d^a \prod_{\substack{j=1 \\ j \neq d}}^{ND} \widehat{\mathbf{M}}_j^T \mathbf{T}_{jn}, \quad (65)$$

respectively.

Received: 23 February 2021 Accepted: 6 September 2021

Published online: 04 October 2021

**References**

1. Ammar A, Huerta A, Chinesta F, Cueto E, Leygue A. Parametric solutions involving geometry: a step towards efficient shape optimization. *Comput Methods Appl Mech Eng.* 2014;268C:178–93.
2. Ammar A, Ghnatios Ch, Delplace F, Barasinski A, Duval JL, Cueto E, Chinesta F. On the effective conductivity and the apparent viscosity of a thinrough polymer interface using PGD-based separated representations. *Int J Num Methods Eng* (In press).
3. Azaiez M, Ben Belgacem F, Casado J, Chacon T, Murat F. A new algorithm of proper generalized decomposition for parametric symmetric elliptic problems. *SIAM J Math Anal.* 2018;50(5):5426–45.
4. Bazilevs Y, Beirão da Veiga L, Cottrell JA, Hughes TJR, Sangalli G. Isogeometric analysis: approximation stability and error estimates for h-refined meshes. *Math Models Methods Appl Sci.* 2006;16(07):1031–90.
5. Bogner B, Leygue A, Chinesta F, Poitou A, Bordeu F. Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity. *Comput Methods Appl Mech Eng.* 2012;201:1–12.
6. Bogner B, Leygue A, Chinesta F. Separated representations of 3D elastic solutions in shell geometries. *Adv Model Simul Eng Sci.* 2014;1:4. <http://www.amses-journal.com/content/1/1/4>.
7. Bordeu F, Ghnatios Ch, Boulze D, Carles B, Sireude D, Leygue A, Chinesta F. Parametric 3D elastic solutions of beams involved in frame structures. *Adv Aircraft Spacecraft Sci.* 2015;2(3):233–48.
8. Chinesta F, Leygue A, Bordeu F, Aguado JV, Cueto E, Gonzalez D, Alfaro I, Ammar A, Huerta A. PGD-based computational vademecum for efficient design, optimization and control. *Arch Comput Methods Eng.* 2013;20:31–59.
9. Chinesta F, Keunings R, Leygue A. The Proper Generalized Decomposition for Advanced Numerical Simulations. A primer. *Springerbriefs in Applied Sciences and Technology*, Springer 2014.
10. Chinesta F, Huerta A, Rozza G, Willcox K. Model Order Reduction. Chapter in the *Encyclopedia of Computational Mechanics*, Second Edition. In: Erwin Stein, René de Borst & Tom Hughes Edt., Wiley: New York; 2015.
11. Cottrell JA, Hughes TJR, Reali A. Studies of refinement and continuity in isogeometric structural analysis. *Comput Methods Appl Mech Eng.* 2007;196(41):4160–83.
12. Cottrell JA, Hughes TJR, Bazilevs Y. *Isogeometric analysis toward integration of cad and FEA*. New York: Wiley; 2009.
13. Ghnatios Ch, Abisset-Chavanne E, Binetruy Ch, Chinesta F, Advani S. 3D modeling of squeeze flow of multiaxial laminates. *J Non-Newt Fluid Mech.* 2016;234:188–200.
14. Evans JA, Hughes TJR. Discrete spectrum analyses for various mixed discretizations of the Stokes eigenproblem. *Comput Mech.* 2012;50(6):667–74.
15. Ghnatios Ch, Xu G, Visonneau M, Leygue A, Chinesta F, Cimetiere A. On the space separated representation when addressing the solution of PDE in complex domains. *Discr Contin Dyn Syst.* 2016;9(2):475–500.
16. Ghnatios C, Abisset-Chavanne E, Ammar A, Cueto E, Duval JL, Chinesta F. Advanced spatial separated representations. *Comput Methods Appl Mech Eng.* 2019;354:802–19.
17. Ghnatios C, Cueto E, Falco A, Duval JL, Chinesta F. Spurious-free interpolations for non-intrusive PGD-based parametric solutions: application to composites forming processes. *Int J Mater Form* (In press).
18. Gonzalez D, Ammar A, Chinesta F, Cueto E. Recent advances in the use of separated representations. *Int J Num Methods Eng.* 2010;81(5):637–59.
19. Hughes TJR, Cottrell JA, Bazilevs Y. Isogeometric analysis: CAD finite elements NURBS exact geometry and mesh refinement. *Comput Methods Appl Mech Eng.* 2005;194:4135–95.
20. Hughes TJR, Reali A, Sangalli G. Duality and unified analysis of discrete approximations in structural dynamics and wave propagation comparison of p-method finite elements with k-method NURBS. *Comput Methods Appl Mech Eng.* 2008;197(49):4104–24.
21. Hughes TJR, Evans JA, Reali A. Finite element and NURBS approximations of eigenvalue boundary value and initial value problems. *Comput Methods Appl Mech Eng.* 2014;272:290–320.
22. Kazemzadeh-Parsi MJ. Isogeometric analysis in solution of unconfined seepage problems. *Comput Math Appl.* 2019;78:66–80.
23. Manzoni A, Salmoiraghi F, Heltai L. Reduced Basis Isogeometric Methods (RB-IGA) for the real-time simulation of potential flows about parametrized NACA airfoils. *Comput Methods Appl Mech Eng.* 2015;284(1):1147–80.
24. Lacourt S, Ryckelynck D, Forest S, de Rancourt V, Flouriot S. Hyper-reduced direct numerical simulation of voids in welded joints via image-based modeling. *Int J Num Methods Eng.* 2020;121:2581–99.
25. Ladevèze P. The large time increment method for the analyze of structures with nonlinear constitutive relation described by internal variables. *C R Acad Sci Paris.* 1989;309:1095–9.
26. Ladevèze Pierre, Passieux J-C, Néron David. The latin multiscale computational method and the proper generalized decomposition. *Comput Methods Appl Mech Eng.* 2010;199(21–22):1287–96.
27. Leygue A, Chinesta F, Beringhier M, Nguyen TL, Grandidier JC, Pasavento F, Schrefler B. Towards a framework for non-linear thermal models in shell domains. *Int J Num Methods Heat Fluid Flow.* 2013;23(1):55–73.
28. Lipton S, Evans JA, Bazilevs Y, Elguedj T, Hughes TJR. Robustness of isogeometric structural discretizations under severe mesh distortion. *Comput Methods Appl Mech Eng.* 2010;199(5):357–73.
29. Perotto S, Carlino MG, Ballarin F. Model reduction by separation of variables: a comparison between Hierarchical Model reduction and. *Proper Generalized Decomposition*. 2018 arXiv preprint [arXiv:1811.11486](https://arxiv.org/abs/1811.11486).
30. Piegli L, Tiller W. *The NURBS Book*. Monographs in Visual Communication, Second ed., Springer-Verlag, 1977.
31. Pruliere E. 3D simulation of laminated shell structures using the Proper Generalized Decomposition. *Compos Struct.* 2014;117:373–81.
32. Reyes R, Codina R, Baiges J, Idelsohn S. Reduced order models for thermally coupled low Mach flows. *Adv Model Simul Eng Sci.* 2018;5(1):28.



33. Senecal JP, Ji W. Characterization of the proper generalized decomposition method for fixed-source diffusion problems. *Ann Nucl Energy*. 2019;126:68–83.
34. Sevilla R, Fernandez-Mendez S, Huerta A. NURBS-enhanced finite element method (NEFEM). *Int J Num Methods Eng*. 2008;76(1):56–83.
35. Tertrais H, Ibanez R, Barasinski A, Ghnatios Ch, Chinesta F. On the Proper Generalized Decomposition applied to microwave processes involving multilayered components. *Math Comput Simul*. 2019;156:347–63.
36. Vidal P, Gallimard L, Polit O. Explicit solutions for the modeling of laminated composite plates with arbitrary stacking sequences *Composites Part B - Engineering*. 2014;60:697–706.
37. Vidal P, Gallimard L, Polit O. Shell finite element based on the Proper Generalized Decomposition for the modeling of cylindrical composite structures. *Comput Struct*. 2014;132:1–11.
38. Vidal P, Gallimard L, Polit O. Assessment of variable separation for finite element modeling of free edge effect for composite plates. *Compos Struct*. 2015;123:19–29.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)

---