CrossMark

A comprehensive framework for modeling heterogeneous objects

Francisco Conde-Rodríguez[1] · **Juan-Carlos Torres[2]** ·
Ángel-Luis García-Fernández[1] · **Francisco-Ramón Feito-Higueruela[1]**

Abstract Many real objects are heterogeneous. They are composed of diverse materials, which are present in varying proportions. Materials inside the solid do not have to be uniformly distributed. So, methods capable of accurately model not only the boundary of the solid, but also the distribution of material in every single point of its interior, are needed. In this paper we propose a new framework for modeling heterogeneous objects. The framework is comprehensive as it characterizes precisely heterogeneous objects, defines an adequate mathematical model that captures the essence of such objects, and a computational representation to represent the modeled objects in a computer. Our framework is based on Bézièr hyperpatches and solves the main problems of this mathematical tool. We have implemented it completely in order to check whether it is possible to precisely model real objects.

Keywords Object modeling · Heterogeneous objects · Bézièr

✉ Francisco Conde-Rodríguez
fconde@ujaen.es

Juan-Carlos Torres
jctorres@ugr.es

Ángel-Luis García-Fernández
algarcia@ujaen.es

Francisco-Ramón Feito-Higueruela
ffeito@ujaen.es

[1] Departamento de Informática, University of Jaén, EPS, Campus las lagunillas s/n, 23071 Jaén, Spain

[2] Departamento de Lenguajes y Sistemas Informáticos, ETSIIT, University of Granada, c/Periodista Manuel Saucedo Aranda, s/n, 18071 Granada, Spain

1 Introduction and motivation

Heterogeneous objects are composed of different materials whose proportion throughout the solid is variable. These objects benefit from the best features due to the mix of materials they are made from. The mixture of materials that compose heterogeneous solids, is called heterogeneous material or functionally graded material. Heterogeneous materials are used to manufacture many types of real objects, e.g., medical implants, engine parts or competition tires.

Designing heterogeneous objects is not an easy task. It is necessary to have a computer application, based on a well stated framework, to address all the complex aspects of the design process.

In this paper, we describe a comprehensive framework that allows representing heterogeneous objects in a simple and efficient way. The framework is based on Bézièr hyperpatches, further developed and extended to solve the main problems of this well-known formulation. A hyperpatch is a continuous, three-parameter, single-valued function that defines the coordinates of the set of points comprising the object exterior as well as its interior. The shape of the object is defined by geometric coefficients also called control points.

Our framework is able to maintain continuity between parametric cells, and also allows representing discontinuities if necessary, both in the geometry and the gradation of the used materials. It also keeps the validity of the represented object.

We also developed a simplified mesh of geometric coefficients, that allows a human operator to stay focused on the main structure of the object and adjust the fine details in a very simple and intuitive manner. The editing is performed by moving geometric coefficients in a process assisted by the framework, allowing to automatically maintain first-order continuity in those regions that need it. It is also possible to

interactively create new object models with the aid of techniques such as extrusion, sweeping, or revolution of existing cells.

All this is done while keeping a relatively low number of parametric cells, which simplifies its recording, processing, and transmission through a network.

We developed a computer application to test whether the framework can accurately represent real objects (see Fig. 1). Using our application, we modeled a real motorcycle tire from scratch, following precisely the blueprints included in the patent [11]. The whole process only took 6 h.

The remainder of this paper is organized as follows. In Sect. 2, some of the research works already carried out in this field are briefly described. Our comprehensive framework is described in Sect. 3; Sect. 3.1 defines precisely what a heterogeneous object is and a valid material distribution, Sect. 3.2 describes heterogeneous regular objects (hr-objects) which are the mathematical model inside our framework, Sect. 3.3 describes heterogeneous composite Béziér hyperpatches that are the computational representation we use, and Sects. 3.4 and 3.5 describe how our framework keeps two fundamental properties: validity and continuity. Next, in Sect. 4 we explain how heterogeneous objects can be edited, both in geometry and material distributions, in a flexible and simple way, using our framework. In Sect. 5 we describe some of the operations that our framework provides to construct models of heterogeneous objects from scratch. In Sect. 6 several examples of real objects modeled with a computer application using our
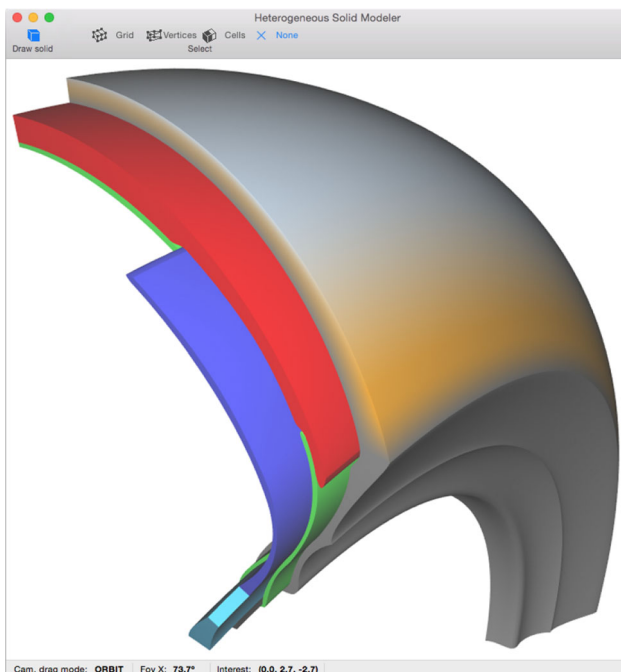

Fig. 1 Real heterogeneous motorcycle tire precisely modeled with our computer application

framework are shown. And finally, the conclusions of this work are summarized in Sect. 7.

2 Related work

A precise characterization of a heterogeneous object is presented in [7] by Kumar and Dutta. The mathematical model for heterogeneous objects (r_m-set / r_m-object) proposed by the authors captures very well the essence of this type of objects, so it is quite an appropriate mathematical model for computational representations of heterogeneous objects. The authors also presented a computational representation by means of non-manifold B-reps based on r_m-objects.

Traditional solid modeling methods, like boundary representation (B-rep), constructive solid geometry (CSG), or enumeration, are not the most suitable for heterogeneous objects. This is due to the fact that they either do not allow representing the inside of the object, or use very simple functions for precisely modeling the interior.

Research on the use of hyperpatches as a tool for modeling heterogeneous objects [1] started during the 1980s. Hyperpatches have an extraordinary capacity to precisely represent each point inside the object by allowing to designate the desired material to every point. However, there is a high degree of freedom in the mathematical tool, which needs to be refined so it can be a useful tool.

Several attempts have been made to refine hyperpatches:

Schmitt, Pasko and Schlick [14] suggested a method to model heterogeneous objects that uses FRep as the basic model combined with trivariate B-Splines primitives by means of CSG operations. This method can be very useful for rapid prototyping or virtual sculpting.

Qian and Dutta [12] suggested using B-Spline hyperpatches to represent heterogeneous objects. They focused on making it easy for the user to assign the materials to each point of the object considering the desired properties for that specific point. For this task, the authors defined a diffusion process.

In [5], and later in [3] and [4], a model based on trivariate simplex splines was presented. This model is useful when constructing a representation of a heterogeneous object using a triangle mesh as the starting point. It allows obtaining very accurate tetrahedral spline decompositions. Variable continuity is defined between 0 and C^{n-1}, being n the degree of the splines. The generated simplex B-Spline mesh is huge, even for objects with relatively simple geometries, because the models have been constructed from triangulations. The editing, as with any hyperpatch, is done by moving the control points.

Yang and Qian [16] describe a method for constructing complex geometries with B-Spline hyperpatches defined

from curves using a lofting process. For certain problems, this method can be a good approach.

Warkhedkar and Bhatt [15] used regular B-Spline hyperpatches to accommodate the slices from a CT scan and adjust the distribution values in the vertices of the B-Spline control mesh to represent the human body. The visualization is done using classical techniques, such as marching cubes.

In [6], the authors describe a method based on functionally graded material (FGM), very adequate for rapid prototyping, for which it is not necessary to precisely model very complex geometries.

These current methods have certain weaknesses, such as:

- The classical formulation of hyperpatches produces completely continuous models or with no continuity at all. Mixed continuity is not allowed in classic hyperpatches. For example: Cubic B-Spline hyperpatches have coefficients that are aligned in regular grids and produce C^2 continuous objects. They cannot model discontinuities, both in geometry or in distribution of materials, between adjacent cells. Knot insertion and other techniques try to solve this problem but make the editing process intricate.
- Hyperpatches *per se* do not address the problem of validity (the hyperpatch function *per se* is not injective), which is a very important property in a object modeling framework, because it prevents the modeling of nonsense objects.
- The natural way to edit a hyperpatch is by moving its geometric coefficients, however if these coefficients are treated independently, without the aid of the framework to perceive and maintain the existing relations among them, the editing is very unintuitive and difficult for modeling new objects.
- Other current methods like CSG or B-rep do not allow a precise representation of both the inside of the object and its boundary.
- Despite its ability to make rapid prototypes, methods based on CSG are not well suited to modeling complex geometries that cannot be obtained by boolean operations.
- Compact computational representations are preferable since they produce a finite set of symbols that facilitate its processing, storage and transmission.

3 Our framework

Traditionally, to describe a new framework for modeling objects, a three-level approach has been used [9,13]:

- *Physical objects* They are the objects that need to be studied or manipulated. They may or may not exist.
- *Mathematical models* Physical objects cannot be modeled in a computer, so it is necessary to adopt an adequate

idealization thereof. These idealized objects must have a clear and intuitive connection with physical objects, and at the same time, be simple enough so that they can be represented in a discrete and limited computer.
- *Computational representations* The final step in the modeling activity is to assign to the mathematical model a representation that can be suitable for its use in a computer.

We use this approach, so in the next three sections we explain each of these levels.

3.1 Material distribution

In this section, we characterize the physical objects we are trying to model.

Kumar and Dutta present in [7] a great mathematical characterization of heterogeneous objects. We use its characterization with slight variations.

Heterogeneous objects are composed of different constituent materials (also known as *primary materials*), which are present in varying proportions throughout the object. It is assumed that the number of primary materials n is finite. The material at a point $\mathbf{p}$ in a object $\mathbf{S}$ is a combination (mixture) of the n primary materials specified by the volume fractions of these materials. Thus, the $\mathbb{R}^n$ set, with a vectorial space structure, is a mathematical space that defines the material distribution, each dimension representing a specific primary material.

Let $\mathbf{M} = \{m_1, m_2, \ldots, m_n\}$ be a set of primary materials, where each material m_i is a literal (the name of that material). Let $n = |\mathbf{M}|$ be the cardinality of the set $\mathbf{M}$. The *material distribution* of any given point $\mathbf{p}$ of a heterogeneous object is represented by a tuple $\mathbf{a} = (a_1, a_2, \ldots, a_n) \in \mathbb{R}^n$, where each element a_i of the material distribution is the fraction of the volume occupied by the ith material at that point.

The volume fractions of each primary material must add up to 1, otherwise the objects in the real world are senseless. In other words, not all possible material distributions $\mathbf{a} \in \mathbb{R}^n$ are appropriate. The valid space of material distributions $\mathbf{V}$ can be defined as:

$$\mathbf{V} = \left\{ \mathbf{a} \in \mathbb{R}^n \ \middle| \ \sum_{i=1}^{n} a_i = 1, \quad a_i \geq 0 \right\} \tag{1}$$

A material distribution $\mathbf{a}$ is valid only if $\mathbf{a} \in \mathbf{V} \subseteq \mathbb{R}^n$; i.e., a material distribution is valid if its elements a_i are the coefficients of a convex combination of a set of primary materials $\mathbf{M}$.

A combination operation is defined over the set $\mathbf{V}$ of valid material distributions. Let $\mathbf{a}, \mathbf{b} \in \mathbf{V}$ be two valid material distributions, then its combination is given by:

$$a \oplus_\alpha b = \alpha \cdot a + (1 - \alpha) \cdot b, \quad \alpha \in [0, 1], \tag{2}$$

where α is an scalar that defines the contribution of each material distribution to the final result.

The operation of combination $\oplus_\alpha$ is closed in $\mathbf{V}$. We think this formulation of the combination operation $\oplus_\alpha$ is more simple and straightforward than other proposed formulations.

3.2 Heterogeneous regular object

In this section we characterize the mathematical model we use in our framework.

The r_m-objects (made up by r_m-sets) proposed by Kumar and Dutta [7] are the appropriate mathematical idealization for heterogeneous objects. Here, we use r_m-objects as the mathematical model for a heterogeneous object, but with slight variations.

We force the material function to be piecewise continuous, in order to model discontinuities in the material distribution. This is a very important capability of our framework as it allows abrupt change of materials throughout the modeled object.

The mathematical model we suggest is the hr-object, based on the r_m-sets and the r_m-objects of Kumar and Dutta. Let $\mathbf{M}$ be a set of primary materials, and let $n = |\mathbf{M}|$ be the cardinality of the set $\mathbf{M}$.

A *heterogeneous regular object (hr-object)* (see Fig. 2) is defined as a ternion, $\mathbf{O} = \{\mathbf{S}, \mathbf{V}, F\}$, where:

- $\mathbf{S}$ is an r set. $\mathbf{S} \subseteq \mathbb{R}^3$ is a limited set with its boundary defined by rigid and regular algebraic functions.
- $\mathbf{V} \subseteq \mathbb{R}^n$ is the set of all valid material distributions.
- F is the material function; it assigns a valid material distribution $\mathbf{a}$ (see Eq. 1) to each point $\mathbf{p}$ of the r set. The F
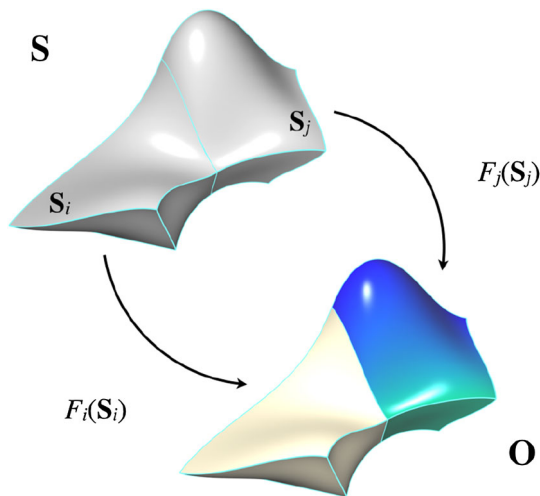
function defines the way primary materials are distributed throughout the volume of the object.

Unlike other proposed models, the domain of our material function F is not $\mathbb{E}^3$. This make it possible to apply deformations to the modeled heterogeneous objects without losing information on the material distributions.

F is a piecewise continuous function:
Let $\{\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_p\} \subseteq \mathscr{P}(\mathbf{S})$ be a partition of $\mathbf{S}$, i.e.:

$$\mathbf{S}_i \cap \mathbf{S}_j = \emptyset, \quad i \neq j \quad \text{and} \quad \bigcup_{i=1}^{p}{}^{*}\mathbf{S}_i = \mathbf{S}.$$

$$F(\mathbf{p}) = \begin{cases} F_i : \mathbf{S}_i \to \mathbf{V} \\ F_i(\mathbf{p}) = \mathbf{a}, \quad \mathbf{p} \in \mathbf{S}_i \quad 1 \leq i \leq p \end{cases} \tag{3}$$

Each function F_i is continuous in the $\mathbf{S}_i$ set in which it is defined. We name each of the subsets $\mathbf{S}_i$ an hr-set (every volume in which the F function is continuous).

As can be seen in Eq. 3, the material function F defines the way the materials are distributed in the volume of the object. It assigns a valid distribution of materials to each point $\mathbf{p}$ of the r set.

The material function F does not have to be continuous in the whole r set $\mathbf{S}$. This allows modeling complex objects with discontinuities both in geometry and material distribution.

The *external geometric boundary* of the regular heterogeneous object $\mathbf{O}$, $b_{\text{ext}}(\mathbf{O})$, is the boundary of the regularized union of the $\mathbf{S}_i$ sets.

$$b_{\text{ext}}(\mathbf{O}) = b\left(\bigcup_i {}^{*}\mathbf{S}_i\right), \quad 1 \leq i \leq p$$

The points that do not belong to $b_{\text{ext}}(\mathbf{O})$, but belong to $b(\mathbf{S}_i), 1 \leq i \leq p$, that is, those points that are not in the boundary of the regular heterogeneous object but only in the boundary of some of the hr-sets that form it, are called *internal boundary points*.

$$b_{\text{int}}(\mathbf{O}) = b_{\text{ext}}(\mathbf{O}) \setminus \bigcup_i b(\mathbf{S}_i), \quad 1 \leq i \leq p$$

Internal boundary points can be part of more than one hr-set. They are irregular points that may have two or more associated material distributions.

Let $\underline{\mathbf{p}} \in b_{\text{int}}(\mathbf{O})$ be a point belonging to the internal boundary points of the hr-object $\mathbf{O}$.

Let $\mathbf{L} = \{l_1, \ldots, l_m\} \in \{1, \ldots, p\}$, be the set formed by the indices of the hr-sets to which the point $\underline{\mathbf{p}}$ belongs:

$$\underline{\mathbf{p}} \in \bigcup_{l_i}^{\mathbf{L}} b(\mathbf{S}_{l_i}) \subseteq b_{\text{int}}(\mathbf{O}), \text{ and let } l_{\min} = \min(\mathbf{L})$$

We define the material function on this internal boundary point $F(\underline{\mathbf{p}})$ as:

$$F(\underline{\mathbf{p}}) = F_{l_{\min}}(\underline{\mathbf{p}})$$

Fig. 2 Structure of a hr-object

Thus the material function F is injective in these points, and therefore the hr-object is valid.

Note that $\mathbf{M}$ is not part of the mathematical definition of heterogeneous regular object, because the only thing that matters is its cardinality $n = |\mathbf{M}|$ (number of primary materials for a specific problem). Applications using hr-objects will assign the meaning of each material.

3.3 Heterogeneous composite Bézier hyperpatches

Finally, in this section, we characterize the computational representation we use in our framework.

Hyperpatches (also known as parametric objects or trivariate objects) are a mathematical tool that has been successfully used to represent objects [10]. They have a clear advantage over other classical tools, such as boundary representation (B-rep), constructive solid geometry (CSG), or decomposition models (exhaustive enumeration, octrees, etc.): they allow to accurately represent not only the boundary of the object, but also the interior. A *hyperpatch* is a set of points whose coordinates are the result of a vectorial, continuous, single-valued function:

$$\mathbf{B}(u, v, w) = (x(u, v, w), y(u, v, w), z(u, v, w)),$$

where the parametric variables u, v, and w are usually limited to values within the closed interval [0, 1].

This vectorial function defines the coordinates of the set of points that make up the object. For every 3-tuple of values of the parametric variables (u, v, w), the $\mathbf{B}$ function provides the coordinates of a point $\mathbf{p} = (x, y, z) \in \mathbb{R}^3$ of the object. Thus, a hyperpatch is a parametric projection of a solid domain (a unit size cube) in the three-dimensional space. The domain of the projection is known as *parametric space* $\mathbb{U}$, and the projection range is called *modeling space* $\mathbb{R}^3$.

The *geometric form* of the hyperpatch is:

$$\mathbf{p}(u, v, w) = \sum_{i=0}^{3} \sum_{j=0}^{3} \sum_{k=0}^{3} \mathbf{g}_{ijk} B_i^3(u) B_j^3(v) B_k^3(w)$$
$$u, v, w \in [0, 1],$$

where $\mathbf{g}_{ijk} = (x_{ijk}, y_{ijk}, z_{ijk})$ are the *geometric coefficients* that define the shape of the object and B^3 are the *cubic blending functions* that combine these coefficients to obtain all the points of the object. The geometric coefficients are also known as *control points*. As can be seen, the shape of the object is directly linked to the geometric coefficients.

There are several types of blending functions that can be used to combine geometric coefficients. For this work, we have chosen cubic Bézier blending functions [8].

$$B_0^3(t) = (1-t)^3 \qquad B_1^3(t) = 3t(1-t)^2$$
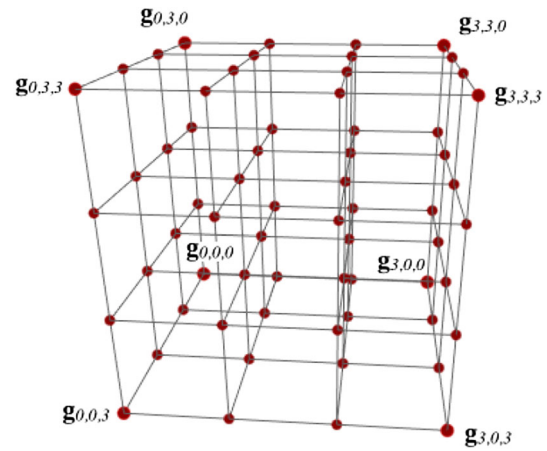$$B_2^3(t) = 3t^2(1-t) \qquad B_3^3(t) = t^3 \tag{4}$$

Fig. 3 Three-dimensional array of $4 \times 4 \times 4 = 64$ geometric coefficients with indices i, j, and k varying from 0 to 3

In the tricubic case, $4 \times 4 \times 4 = 64$ geometric coefficients are used to define a hyperpatch (as can be seen in Fig. 3). It is usually useful to think that these coefficients are organized in a three-dimensional array with indices i, j, and k varying from 0 to 3.

A set of boundary elements is associated to each tricubic Bézier hyperpatch, perfectly defining its boundary (see Fig. 4):

- Six surfaces that are obtained by fixing the value of one of the parametric coordinates u, v, w, which takes the value 0 or 1, while the other two run freely along the [0, 1] interval. They are called *isoparametric boundary surfaces*. For example $\mathbf{B}(0, v, w)$ or $\mathbf{B}_{0vw}$.
- Twelve curves obtained by fixing the value of two of the parametric coordinates u, v, w, which take the value 0 or 1, while the third runs freely along the [0, 1] interval. These are called *isoparametric boundary curves*.
- Eight points obtained by fixing the value of the three parametric coordinates u, v, w, which take the value 0 or 1. These are called *corner points*.

The Bézier functions were chosen because:

- They interpolate the eight corner points of a parametric Bézier hyperpatch, that is, the hyperpatch passes through these points.
- The tangents to each corner point are defined by only its neighboring geometric coefficients (in the array of 64 coefficients). Thus, moving a control point close in the array to a corner point, have a clear effect on the shape of the hyperpatch around that corner point.

Bézier hyperpatches accurately model their interior. However, it is not possible to determine material distribution at
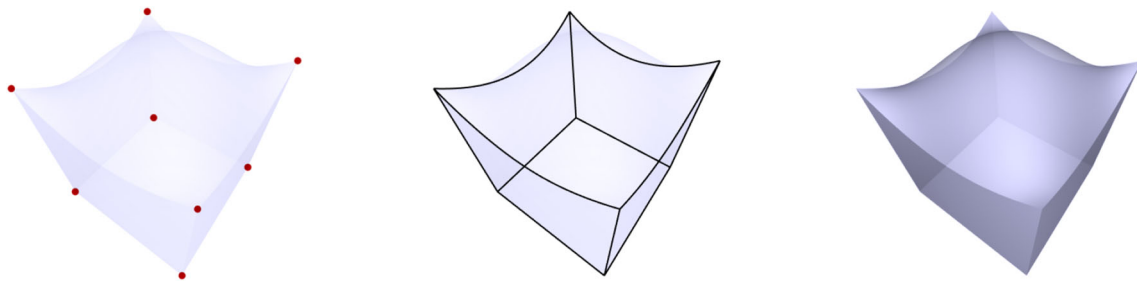
Fig. 4 Boundary elements associated to a hyperpatch. From *left to right* corner points, boundary isoparametric curves and boundary isoparametric surfaces
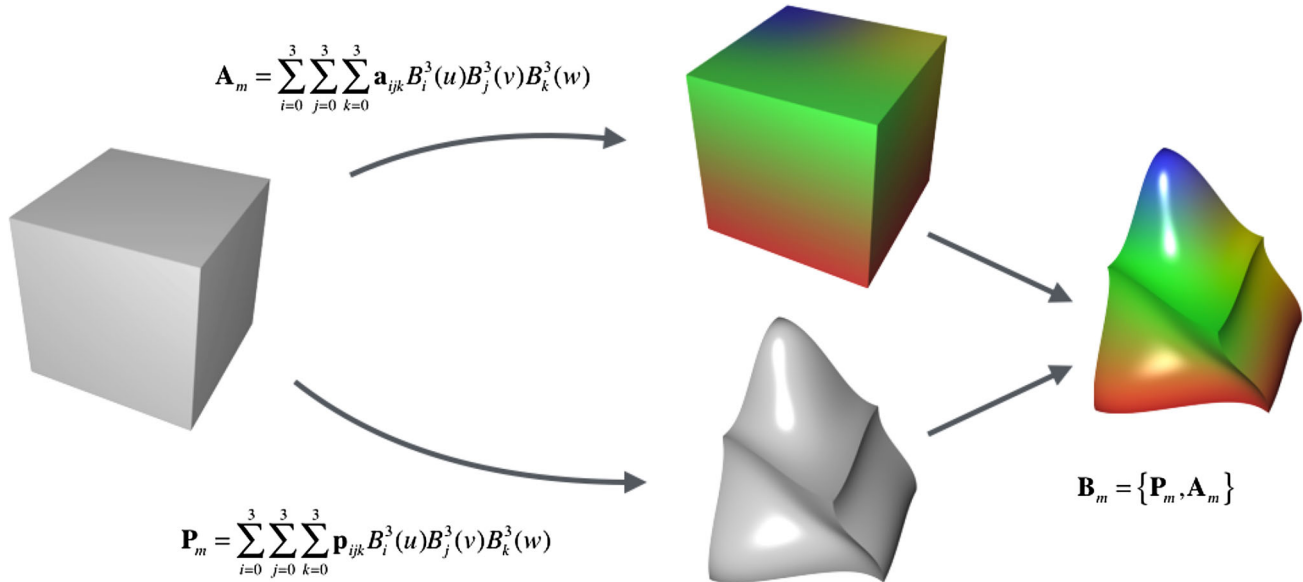

$$\mathbf{A}_m = \sum_{i=0}^{3}\sum_{j=0}^{3}\sum_{k=0}^{3}\mathbf{a}_{ijk}B_i^3(u)B_j^3(v)B_k^3(w)$$

$$\mathbf{P}_m = \sum_{i=0}^{3}\sum_{j=0}^{3}\sum_{k=0}^{3}\mathbf{p}_{ijk}B_i^3(u)B_j^3(v)B_k^3(w)$$

$$\mathbf{B}_m = \{\mathbf{P}_m, \mathbf{A}_m\}$$

Fig. 5 A heterogeneous Béziér hyperpatch $\mathbf{B}_m$, is composed by a set $\mathbf{P}_m$ of points that defines the geometry of the hyperpatch, and a set $\mathbf{A}_m$ of material distributions that defines the gradation of materials throughout the hyperpatch

every point; thus, it is necessary to define the *heterogeneous Béziér hyperpatch* as follows [1]:

$$\mathbf{B}_m = \sum_{i=0}^{3}\sum_{j=0}^{3}\sum_{k=0}^{3}\mathbf{h}_{ijk}B_i^3(u)B_j^3(v)B_k^3(w)$$
$$u, v, w \in [0, 1], \tag{5}$$

where $\mathbf{h}_{ijk} = (\mathbf{p}_{ijk}, \mathbf{a}_{ijk})$ are the *heterogeneous geometric coefficients,* which are pairs of values. The elements

$$\mathbf{p}_{ijk} = (x_{ijk}, y_{ijk}, z_{ijk}) \in \mathbb{R}^3$$

are the points that define the shape of the hyperpatch, and the elements

$$\mathbf{a}_{ijk}= (a_{1_{ijk}}, a_{2_{ijk}}, \ldots, a_{n_{ijk}})\in \mathbf{V} \subseteq \mathbb{R}^n$$

are the material distributions defining the gradation of the different materials throughout the hyperpatch.

Note that a heterogeneous Béziér hyperpatch $\mathbf{B}_m$ is a set formed by two sets:

$$\mathbf{B}_m = \{\mathbf{P}_m, \mathbf{A}_m\}$$

The set $\mathbf{P}_m$ defines the shape of the hyperpatch, and the set $\mathbf{A}_m$ defines the gradation of the different materials throughout it (see Fig. 5).

$$\mathbf{P}_m = \sum_{i=0}^{3}\sum_{j=0}^{3}\sum_{k=0}^{3}\mathbf{p}_{ijk}B_i^3(u)B_j^3(v)B_k^3(w) \quad u, v, w \in [0, 1]$$

$$\mathbf{A}_m = \sum_{i=0}^{3}\sum_{j=0}^{3}\sum_{k=0}^{3}\mathbf{a}_{ijk}B_i^3(u)B_j^3(v)B_k^3(w) \quad u, v, w \in [0, 1]$$

A *heterogeneous composite Béziér hyperpatch*, $\mathbf{B}$, is a finite collection of heterogeneous Béziér hyperpatches.

$$\mathbf{B} = \{\mathbf{B}_1, \mathbf{B}_2, \ldots, \mathbf{B}_m\},$$

where $\mathbf{B}_i = \{\mathbf{P}_i, \mathbf{A}_i\}$.

$\forall \quad \mathbf{P}_i \in \mathbf{B}_i, \mathbf{P}_j \in \mathbf{B}_j \quad i \neq j, \quad \mathbf{P}_i \cap^* \mathbf{P}_j = \emptyset$. That is, the hyperpatches are pairwise disjoint, and at most, only share one, two or four corner points; one or four isoparametric boundary curves; and/or one isoparametric boundary surface.

A heterogeneous composite Béziér hyperpatch can be seen as a cell decomposition, in which each cell is a heterogeneous Béziér hyperpatch. From now on, we are going to use either *parametric cell* or hyperpatch to refer to the heterogeneous Béziér hyperpatches in this decomposition.

3.4 Validity

In any framework for object modeling, validity is a very important property [9,13], as it ensures that meaningless models of objects cannot be obtained. Therefore, in this section we will study the validity of objects modeled by using our framework.

If additional conditions are not established, the classic hyperpatch formulation does not guarantee the validity of the objects represented. For that reason, in 2001, we developed a validity condition for b-spline hyperpatches that easily extends to the Béziér case [2]. Our validity condition works by checking relative positions of geometric coefficients in each parametric cell. Thus, after editing the solid by moving some geometric coefficients it is only necessary to check the validity in those parametric cells containing these coefficients. This process is very fast and can be done in real time.

Our condition of validity is sufficient but not necessary. This means that some parametric cells, that are in fact valid, are rejected. However, this is an advantage of our validity condition, since the points inside the rejected heterogeneous Béziér hyperpatches are not regularly distributed (see Fig. 6). Parametric cells with points regularly distributed inside them are preferable when modeling heterogeneous objects.

3.5 Continuity

There is zero-order continuity C^0 between two parametric cells $\mathbf{B}_l$ and $\mathbf{B}_m$, if they share 16 boundary geometric coefficients. Let $\mathbf{B}_l(1, v, w) = \mathbf{B}_m(0, v, w)$ be the shared surface, then the coefficients $\mathbf{h}_{l_{3jk}} \in \mathbf{B}_l$ and $\mathbf{h}_{m_{0jk}} \in \mathbf{B}_m$ have to be equal $\forall j, k \in \{0, 1, 2, 3\}$.

Note that the reasoning is the same in the case of shared surfaces with fixed values of v or w.

This zero-order continuity condition is too restrictive, so we developed a more flexible one. This is another contribution of our work. In our framework, zero-order continuity can be established for each of the four corner points of the shared surface independently (see Fig. 7).

For a given corner point, only three geometric coefficients have effect in the shape of the boundary isoparametric surface around it. They are its immediate neighbors. So, to obtain
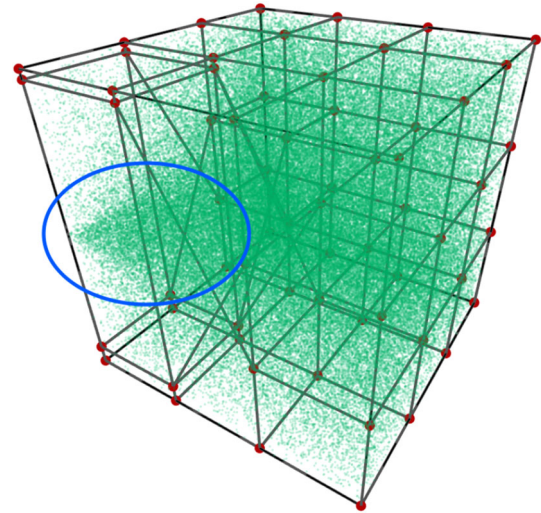

Fig. 6 This parametric cell *is valid*, however, our condition of validity rejects it. This is an *advantage of our framework*, because the points inside this cell are not regularly distributed (*encircled points in blue*). Parametric cells with regular distributions of points inside them are preferable when modeling heterogeneous objects
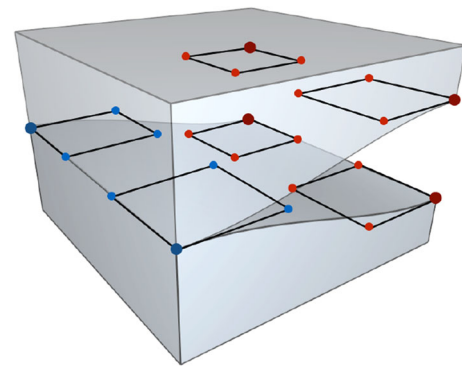

Fig. 7 Zero-order continuity between two parametric cells can be established at corner point level. These two parametric cells share the corner points in *dark blue* and their immediate neighbors so they have C^0 continuity in the *left side*. In the *right side*, each cell has different corner points and different neighbors, so they do not have C^0 continuity

zero-order continuity around a given corner point, both cells have to share it, and have to share its immediate neighboring geometric coefficients.

Let $\mathbf{h}_{l_{3,3,3}} = \mathbf{h}_{m_{0,3,3}}$ be the shared corner point, then the neighbor geometric coefficients $\mathbf{h}_{l_{3,3,2}} = \mathbf{h}_{m_{0,3,2}}$, $\mathbf{h}_{l_{3,2,3}} = \mathbf{h}_{m_{0,2,3}}$ and $\mathbf{h}_{l_{3,2,2}} = \mathbf{h}_{m_{0,2,2}}$ have to be equal in order to obtain zero-order continuity around that corner point (see Fig. 7).

Our condition is very flexible and allows to model complex shapes easily.

Notice that a heterogeneous geometric coefficient $\mathbf{h}_{ijk} = (\mathbf{p}_{ijk}, \mathbf{a}_{ijk})$ has two components: a position in three-dimensional space $\mathbf{p}_{ijk} \in \mathbb{R}^3$ and a material distribution $\mathbf{a}_{ijk} \in \mathbf{V} \subset \mathbb{R}^n$, so zero-order continuity C^0 applies to both the geometry and the distribution of the materials.

Both components are independent, so it is possible to establish C^0 continuity between two parametric cells for the geometry (the shape of the modeled object) but not for the material distribution.

We will study now first-order continuity C^1. Let $\mathbf{B}_l$ and $\mathbf{B}_m$ be two parametric cells that share a boundary isoparametric surface. Let $\mathbf{B}_l(1, v, w) = \mathbf{B}_m(0, v, w)$ be the points of the shared surface.

The heterogeneous Béziér hyperpatches have first-order continuity C^1 in the shared surface if all the isoparametric curves with constant values of v and w passing through each point of the shared surface, have equal tangent vectors to these points on both cells.

$$\mathbf{B}_l^u(1, v, w) = \mathbf{B}_m^u(0, v, w) \quad \forall v, w \in [0, 1] \qquad (6)$$

The tangents to these points depend on the 16 geometric coefficients that define the shared surface, and their immediate neighbors on each cell. So, in order to ensure C^1 continuity, each group of three coefficients must form two vectors that must be equal.

Let $\mathbf{h}_{*, j, k}$ be one geometric coefficient shared between $\mathbf{B}_l$ and $\mathbf{B}_m$. Let $\mathbf{v}_l = \mathbf{h}_{*, j, k} - \mathbf{h}_{l_{3-1}, j, k}$ the vector formed by the shared coefficient and its immediate neighbor in $\mathbf{B}_l$. And let $\mathbf{v}_m = \mathbf{h}_{m_{0+1}, j, k} - \mathbf{h}_{*, j, k}$ the vector formed by the shared coefficient and its immediate neighbor in $\mathbf{B}_m$. Then, there is C^1 continuity between $\mathbf{B}_l$ and $\mathbf{B}_m$ in the shared surface, if (see Fig. 8):

$$\mathbf{v}_l = \mathbf{v}_m; \quad \mathbf{h}_{*, j, k} - \mathbf{h}_{l_{3-1}, j, k} = \mathbf{h}_{m_{0+1}, j, k} - \mathbf{h}_{*, j, k} \quad \forall j, k \in \{2, 3\} \qquad (7)$$

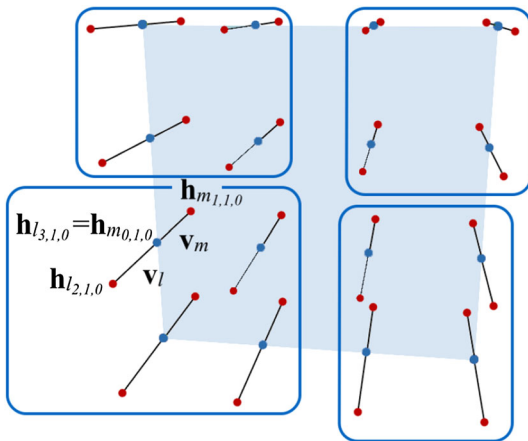**Fig. 8** The tangents to the points on the shared surface (*light blue*) are defined by the 16 geometric coefficients of the surface (*dark blue*) and their immediate neighbors (*red*) on each cell. In order to ensure C^1 continuity, each group of three coefficients must form two vectors that must be equal. In our framework, C^1 continuity can be established at corner point level, so the continuity condition of Eq. 8 can be set for each group of four shared coefficients independently (*encircled points in blue*)

Note that the reasoning is the same in the case of shared surfaces with fixed values of v or w.

Again, this first-order continuity condition is too restrictive, so we developed a more flexible one. In our framework, C^1 continuity can be established for each corner point of the shared surface independently (see Fig. 8).

After editing an immediate neighbor $\mathbf{h}_{l_{3-1}, j, k}$ of a shared geometric coefficient $\mathbf{h}_{*, j, k}$ it is easy to calculate the position of the corresponding coefficient $\mathbf{h}_{m_{0+1}, j, k}$ as:

$$\mathbf{h}_{m_{0+1}, j, k} = 2\mathbf{h}_{*, j, k} - \mathbf{h}_{l_{3-1}, j, k} \qquad (8)$$

Note that the reasoning is the same in the case of shared surfaces with fixed values of v or w.

A heterogeneous geometric coefficient $\mathbf{h}_{ijk} = (\mathbf{p}_{ijk}, \mathbf{a}_{ijk})$ has a position in three-dimensional space and a material distribution. Let us study first-order continuity in both cases separately.

The position of a heterogeneous geometric coefficient is a point in the three-dimensional space $\mathbb{R}^3$, so the result of the operation $2\mathbf{p}_{*, j, k} - \mathbf{p}_{l_{3-1}, j, k}$ is always a valid point in $\mathbb{R}^3$.

However, the same operation cannot be applied on the material distribution, as the material distribution of a heterogeneous geometric coefficient has to be a point in $\mathbf{V} \subset \mathbb{R}^n$.

For example, if we use a set $\mathbf{M} = \{m_0, m_1\}$ of two primary materials, $|\mathbf{M}| = 2$, only the points of $\mathbb{R}^2$ that lie in the line segment with endpoints $(1, 0)$ and $(0, 1)$ represent valid material distributions.

Suppose $\mathbf{a}_{l_{3-1}, j, k} = (0.5, 0.5)$ and $\mathbf{a}_{*, j, k} = (0.2, 0.8)$, applying Eq. 8, $\mathbf{a}_{m_{0+1}, j, k} = (-0.1, 1.1)$ which is a not valid material distribution.

In order to maintain C^1 continuity for material distributions automatically, we developed a C^1 continuity condition for material distributions different to Eq. 7. Given the material distribution $\mathbf{a}$ on a corner point shared by two parametric cells $\mathbf{B}_l$ and $\mathbf{B}_m$, we ensure C^1 continuity in the distribution of materials around that corner point, by assigning the same material distribution $\mathbf{a}$ to all the immediate neighboring coefficients in both cells (see Fig. 10).

As can be seen in Fig. 9 this continuity condition for the distribution of material allows us to model many interesting continuous transitions between two materials (for example, transitions with shape of cylinders or spheres). It is important to note that this is not a linear interpolation between the distributions of material in each of the eight corner points of each parametric cell, but a cubic interpolation.

It is also possible to have C^1 continuity between two cells in some corner points and discontinuity in other corner points shared by the same cells (see Figs. 9, 10).

Of course, if a greater degree of accuracy in the transition between a material and another one is required, continuity always can be adjusted manually.
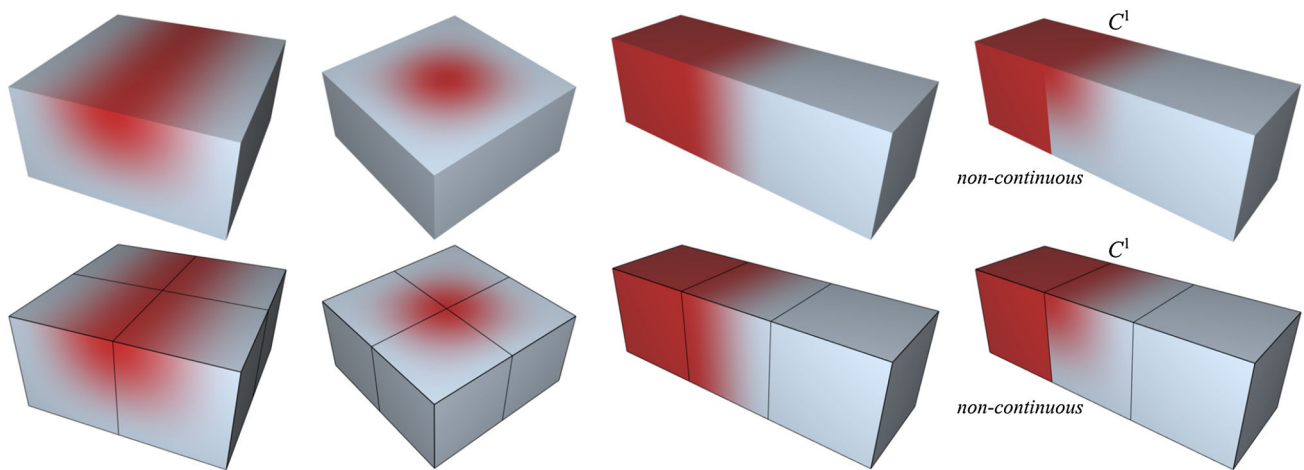
Fig. 9 Our first-order continuity condition for distribution of material allows to model many interesting continuous transitions between two materials, for example with shape of cylinders or spheres (*left side*). It

is also possible to combine discontinuity and first-order continuity in the same shared surface between two cells (*right side*)
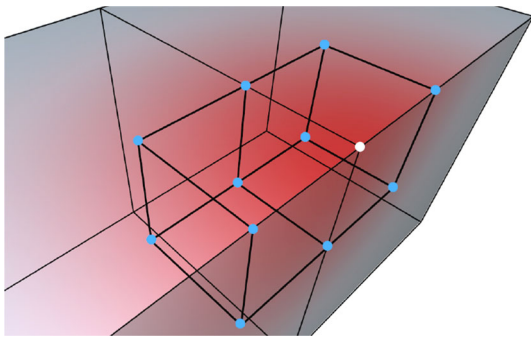

Fig. 10 Our framework maintains C^1 continuity for material distributions between two cells in a given corner point (*white point*), by assigning the same material distribution to its immediate neighboring points (*blue points*)

We do not consider second-order continuity C^2, because this type of continuity cannot be maintained locally, as it involves all geometric coefficients of both cells $\mathbf{B}_l$ and $\mathbf{B}_m$, and consequently all other adjacent cells. That means that, if we establish second-order continuity between two cubic parametric cells, all adjacent cells have to be also C^2 continuous, and then it is not possible to model discontinuities in the object. This makes it poorly flexible for modeling complex objects. Furthermore, second-order continuity has moderate use. For example, most mechanical parts do not need it [10] (page 95).

However, it is possible to obtain second-order continuity between two adjacent cells if necessary by using our framework while maintaining flexibility, just by increasing the degree of the Béziér blending functions.

4 Editing objects

We have specifically designed this model to make the editing process of heterogeneous objects an easier task.

4.1 Editing geometry

Because the shape of an object is directly linked to its geometric coefficients, the natural way to edit a hyperpatch is by moving its geometric coefficients. This generates immediate results.

However, if the geometric coefficients are treated independently, without the framework aid to perceive and maintain the existing relations, the editing process becomes very confusing and difficult for a human operator.

Figure 11 shows that even in a very simple composite hyperpatch (only six parametric cells), the large number of coefficients makes knowing which points have to be edited to generate a certain effect difficult. We have developed a *simplified mesh of geometric coefficients* where only the boundary curves and the corner points are shown.

In the Béziér case, corner points have maximum influence on the object because the blending functions interpolate them (the object passes through them); thus, they are the first level of interaction presented to the user.

When the user selects a corner point, the system shows a *grid of immediate neighbors*. Each corner point has seven contiguous coefficients in the $4 \times 4 \times 4$ grid of geometric coefficients that defines a parametric cell. These seven coefficients are the immediate neighbors of the corner point (see Fig. 12). In general, each corner point is shared by eight adjacent parametric cells and therefore has 26 neighboring control points arranged in a $3 \times 3 \times 3$ grid. This is the second level of interaction presented to the user.

When the user moves a corner point, our framework automatically applies the same movement to all the immediate neighbors, maintaining the continuity of the adjacent parametric cells on that corner point, as well as the general shape of the hyperpatch around it (see Fig. 14b).
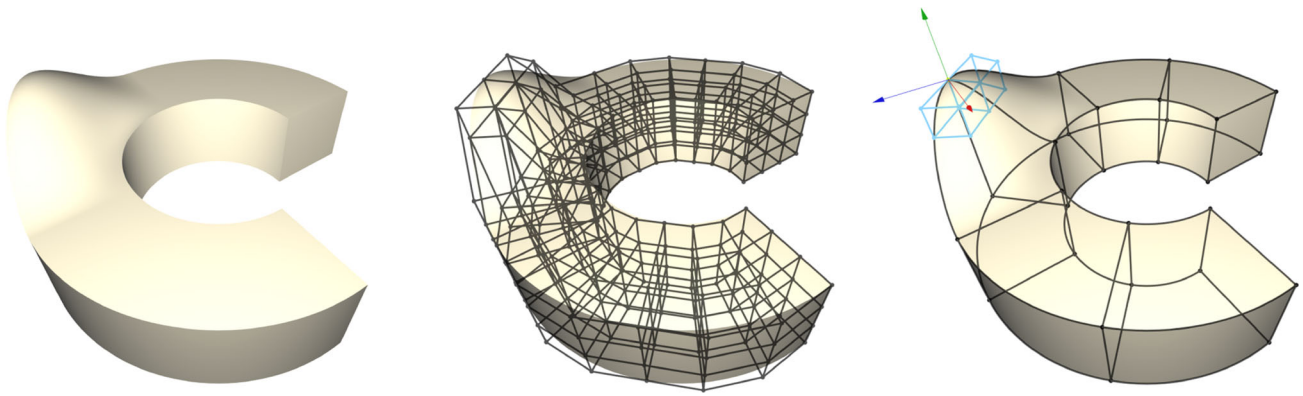
Fig. 11 Even with a simple model, a classical mesh of geometric coefficients (*center*) makes it difficult to localize which points need editing to obtain the desired effect. The image on the *right* shows our simplified mesh, which greatly eases the task
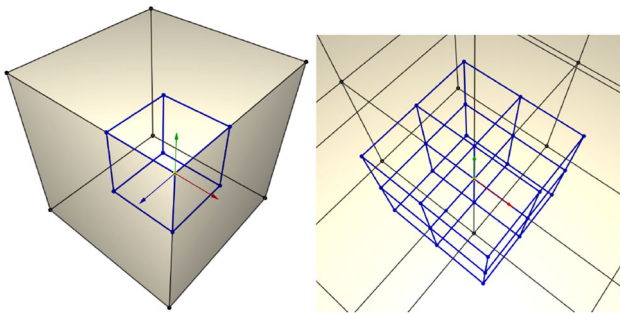

Fig. 12 (*Left*) each corner point has seven contiguous coefficients in the $4 \times 4 \times 4$ grid of geometric coefficients that defines a parametric cell. (*Right*) in general, each corner point has 26 neighboring control points arranged in a $3 \times 3 \times 3$ grid
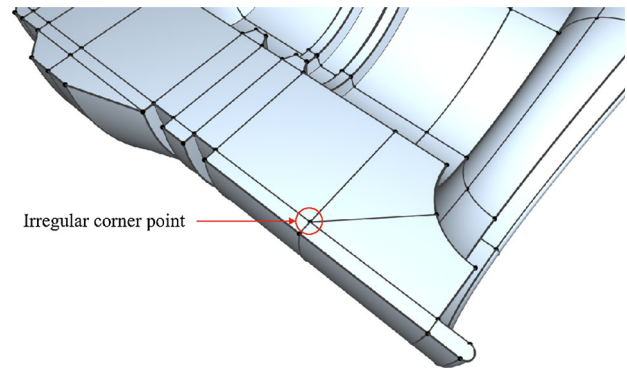

Fig. 13 Our framework is not limited to regular grids of coefficients. In this image we can see a real mechanical part modeled with our framework, with irregular corner points

The framework also allows moving the neighboring geometric coefficients of a selected corner point, for precise adjustments of the shape of the heterogeneous composite Béziér hyperpatch on that point. If the corner point is marked as continuous, when the user moves an immediate neighbor, the system automatically compensates the position of some of the others immediate neighbors by applying Eq. 8, thus maintaining C^1 continuity. If the corner point is marked as non-continuous, when the user moves a neighbor, the system does not modify the neighboring geometric coefficients (see Fig. 14c, d).

Our framework is not limited to regular grids in which each corner point is shared by exactly eight parametric cells, so a given corner point can have an arbitrary number of immediate neighbors. In these irregular corner points it is not possible to maintain continuity automatically; this is an unsolved problem; however, our system can integrate these points, because certain types of discontinuity are very useful for modeling complex objects (see Figs. 13, 14).

4.2 Editing material distributions

When editing the distribution of materials inside a heterogeneous composite Béziér hyperpatch our framework shows the parametric cells that compose it, and the corner points.

If the user selects cells, the system allows to apply the same material distribution to all the 64 geometric coefficients of each cell. This is a uniform distribution of materials. It can be very useful to model those parts of the solid that are homogeneous.

If the user selects a corner point, the system allows to apply the same material distribution to the seven neighbor geometric coefficients of the corner point in each cell that share it. This is a C^1 continuous gradation of materials (see Fig. 15).

If some cells have been selected and the user selects a corner point belonging to them, the system allows to apply the same material distribution to the neighboring geometric coefficients of the corner point but only in each selected cell. This is a C^1 continuous gradation of materials among the selected cells and a non-continuous gradation of material among the non selected cells which also share the selected corner point (see Fig. 15).

It is also possible to apply a different material distribution to each coefficient of a parametric cell manually. This way the user has a very precise control of the gradation of materials.
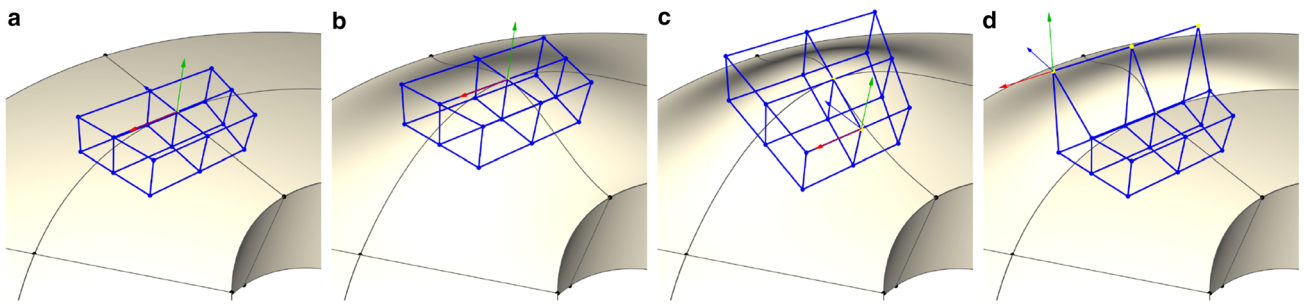
Fig. 14 Hyperpatch editing. From *left to right*: **a** before editing, **b** moving a corner point, **c** moving a neighboring point with continuity activated, **d** moving three neighboring points with no continuity acti-
vated. Note that in **b** and **c** the framework automatically moves the neighboring points accordingly to maintain C^1 continuity
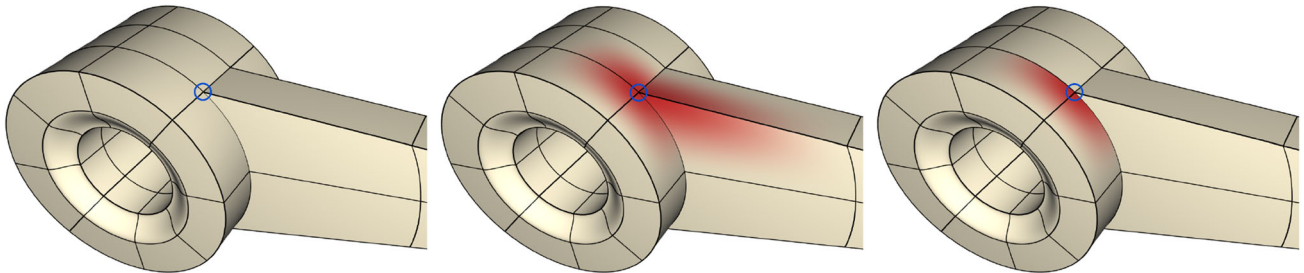

Fig. 15 Editing material distributions. If the user selects a corner point shared among several cells (*left*), the framework can apply a given material distribution to all cells that share the corner point (*center*) or to some
cells (*right*). It depends on whether or not there are selected some cell. Thus it is possible to establish non-continuous or C^1 continuous gradations of materials among cells, if necessary
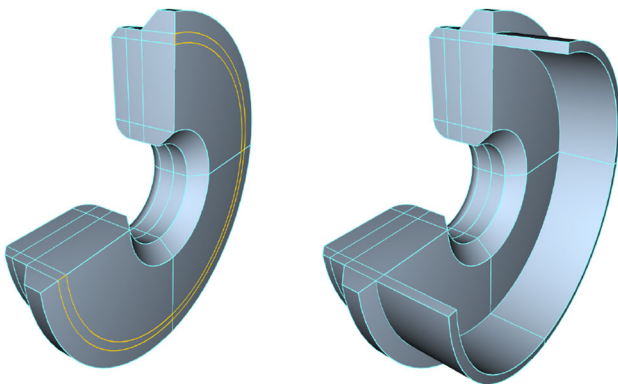
Fig. 16 Extrusion along the normal of three parametric cells from one of their boundary isoparametric surfaces to generate a more complex geometry
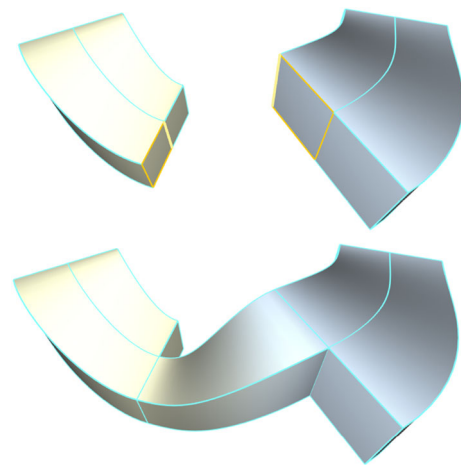

Fig. 17 Blending of two parametric cells. The blended cell is C^1 continuous both for the geometry as for the distribution of materials

However, this manual process is only necessary in a few cases.

In all cases the framework checks automatically the validity of the material distribution, ensuring that all the elements of the applied material distribution are the coefficients of a convex combination of the set of primary materials (see Eq. 1).

5 Creating new objects

A heterogeneous composite Béziér hyperpatch is constructed by adding parametric cells to it. This way of working simpli-

fies the creation of new objects. When a user adds a new cell, our framework automatically performs the following tasks:

- It checks whether the added parametric cell is injective (valid).
- It checks that the new cell does not intersect any existing cell, or that both cells only share a corner point, a boundary isoparametric curve, or a boundary isoparametric surface.
- If the new cell shares a boundary isoparametric surface with any other cell in the hyperpatch, our framework

checks whether there is C^1 continuity in each of the four shared corner points, and, it classifies each point as continuous or non-continuous.

Incorporating new cells can be done using several intuitive methods like:

- *Extrusion* of existing cells (see Fig. 16).
- *Blending* of two parametric cells. The new blended cell can be C^1 continuous both for the geometry as for the distribution of materials if needed (see Fig. 17).

- *Revolution* of a Bézier surface around an axis.
- *Lofting* of a Bézier surface along a Bézier path.
- *Addition from a library* of predefined cell structures (see Fig. 18).
- *Addition from a file* containing a previously saved model. This allows to merge together several objects.

In all cases our framework ensures the validity of the generated heterogeneous composite Bézier hyperpatch. A new parametric cell is only added to the model if it is valid.
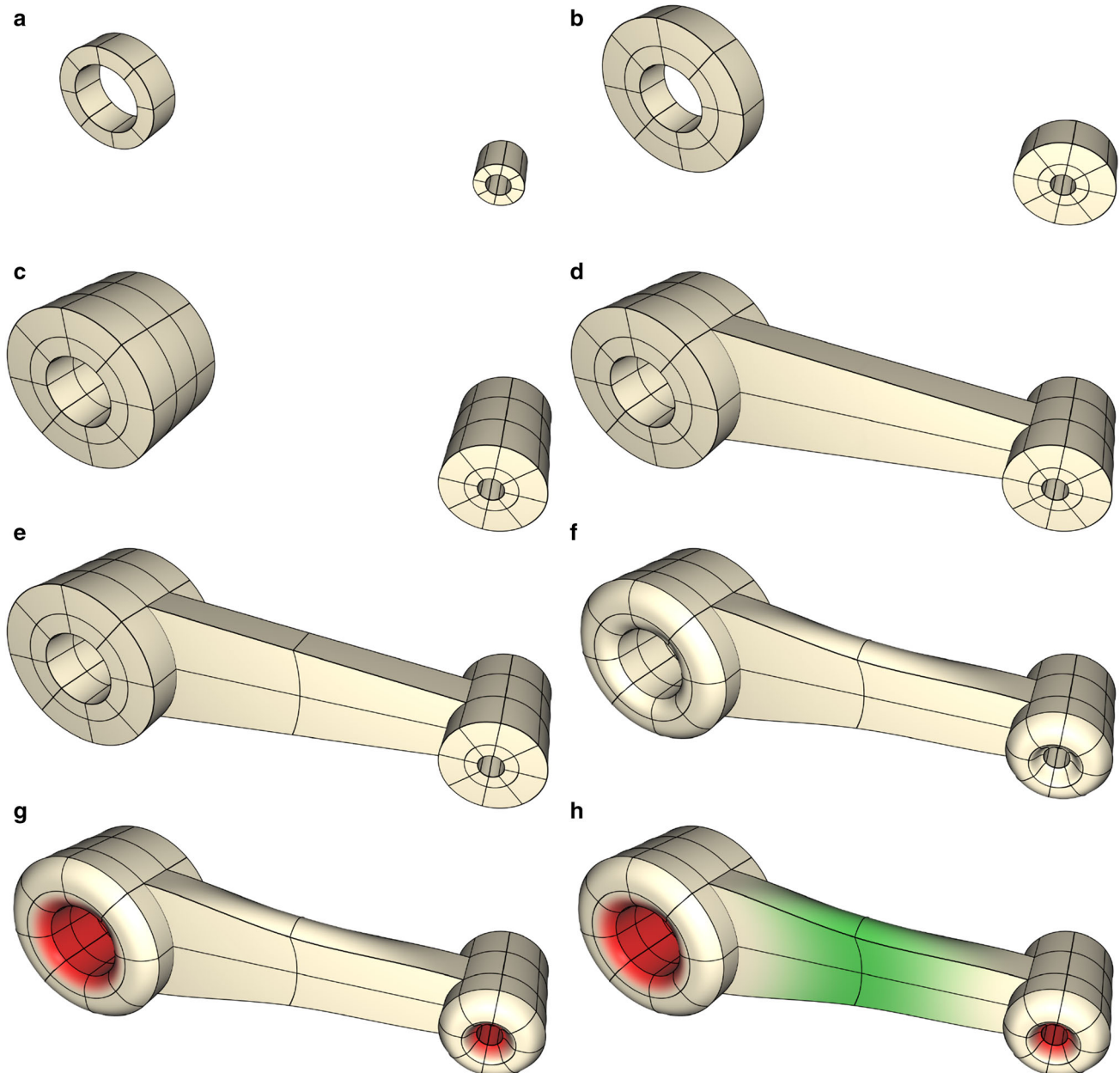

Fig. 18 Creating a heterogeneous object in eight steps. From *top to bottom*, and from *left to right*: **a** adding two rings from a library of predefined cell structures; **b** and **c** extrusion of the rings to obtain a wider structure; **d** non-continuous blending of two groups of two cells to connect both rings; **e** subdivision of the two blended cells; **f** editing of the external cells of the rings in order to obtain a more rounded shape; **g** and **h** setting a C^1 continuous gradation of materials among selected cells
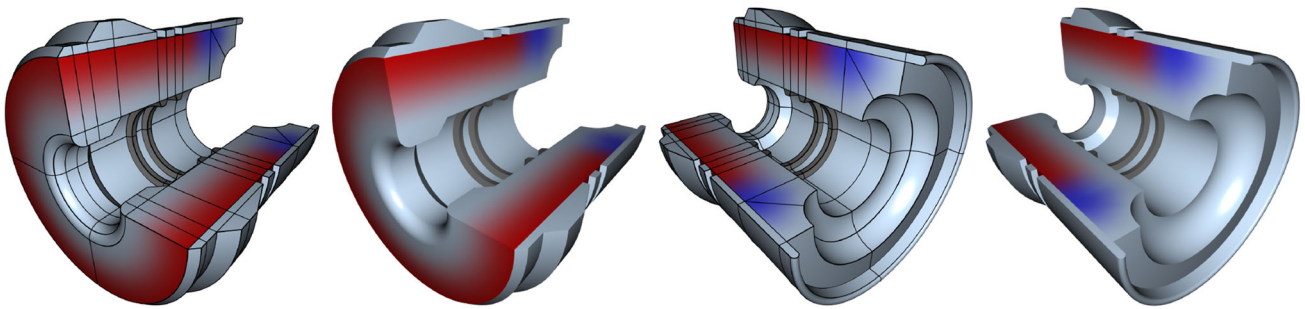
Fig. 19 *Example 1* Mechanical part modeled from scratch using the computer application based on our framework
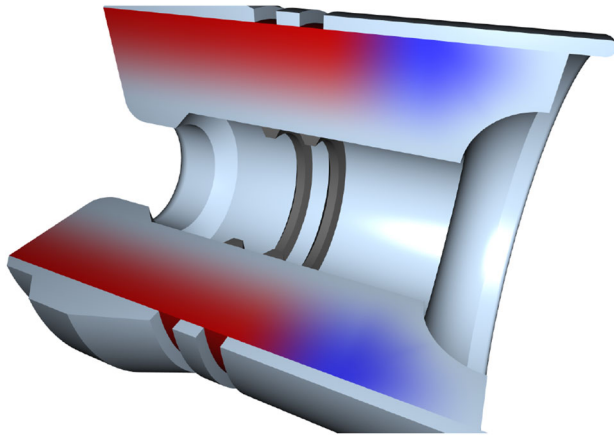

Fig. 20 *Example 2* Heterogeneous motorcycle tire modeled following precisely the blueprints included in the patent [11]

Fig. 21 Detailed view of the mechanical part showing its heterogeneous internal structure

6 Examples

We developed a computer application to test whether the framework can accurately represent real objects. Using our application we have modeled several objects.

We modeled a mechanical part (see Figs. 19, 20) composed of four materials (light grey, dark grey, blue and red in the figure). In some zones of the object, the transition among the materials is a C^1 continuous gradation. The number of parametric cells needed to accurately model the part is low (100 cells).

The most interesting heterogeneous object we have modeled is a motorcycle tire (see Figs. 21, 22). We followed the exact blueprints of a patented tire [11]. The whole modeling process of the shown tire only took 6 h. The amount of cells needed only was 204.

An interesting feature of the framework is that the whole tire, including all its components (tread, body ply, belt, sidewall, etc.) is *modeled as a single object,* which is very convenient for simulations. This is possible due to the ability of the framework to model discontinuities in material distributions in the various parametric cells that make up the object.
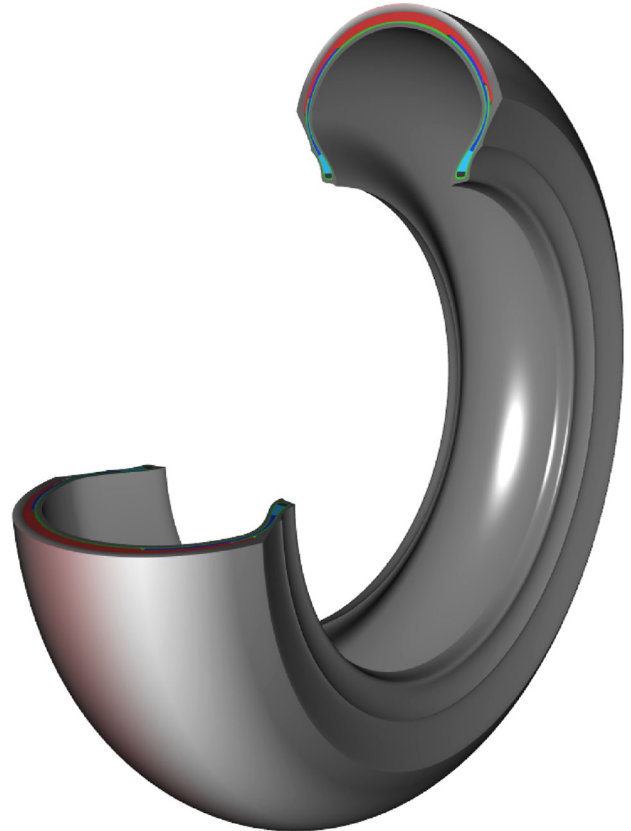
7 Conclusions

In this work, we have presented a comprehensive framework that allows representing heterogeneous objects in a simple and efficient way. The framework is based on Béziér hyperpatches and solves the main problems of this well-known formulation.

Validity of the represented objects is kept automatically by checking relative positions of geometric coefficients in each parametric cell. Our validity condition not only rejects nonvalid cells, it rejects also cells with interior points not well distributed. This is very useful when modeling heterogeneous objects.
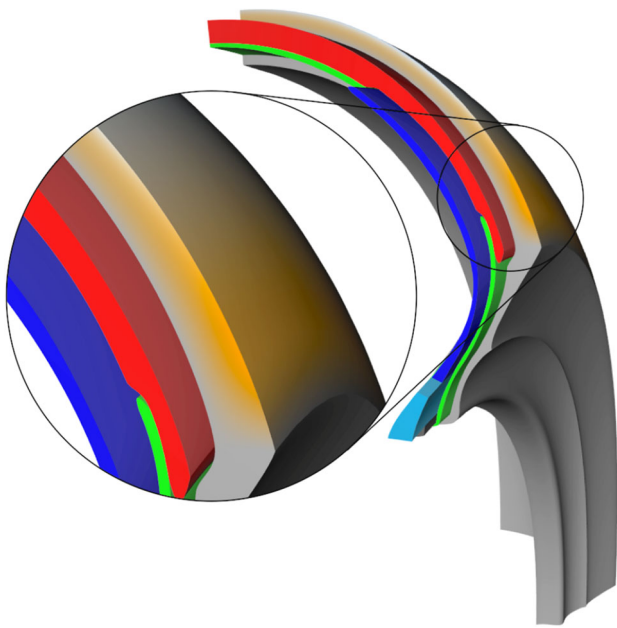
Fig. 22 Detailed view of the heterogeneous tire. The tread is composed by a continuous combination of three materials

In addition, we have presented a simplified mesh of geometric coefficients that allows a human operator to focus on the main structure of the object and adjust the fine details in a very simple and intuitive way. The editing is performed by movements of geometric coefficients in a process assisted by the system. The framework simplifies the editing process by automatically maintaining continuity (both in the geometry as in the distribution of materials) at the corner points that require it. Furthermore, our framework can also manage irregular non-continuous corner points, for modeling complex objects that require discontinuities.

The presented framework can also create new object models in an interactive manner with the aid of techniques such as extrusion, sweeping, or revolution of existing parametric cells while maintaining a relatively low number of parametric cells.

Finally, we have presented several complex objects shaped with a computer application using our framework. The application allows testing the ability of the model to accurately represent real objects.

7.1 Future work

Our framework can be extended to represent *Porous materials* as it is possible to use the emptiness as a primary material. This is perfectly coherent with the methods used for layered manufacturing (LM), in which a special material is used to fill in the empty spaces during the manufacturing process and later this special material is removed leaving a gap.

Modeling internal microstructures on several scale levels inside the object is also possible. Each parametric cell is the mapping of a unit cube into three-dimensional space, so we can define the microstructure in this unit size domain.

References

1. Casale, M., Stanton, E.: An overview of analytic solid modeling. IEEE Comput. Graph. Appl. **5**(2), 45–56 (1985). doi:10.1109/MCG.1985.276402
2. Conde Rodriguez, F.A., Torres Cantero, J.: A simple validity condition for b-spline hyperpatches. In: Proceedings of Eurographics, 2001, short presentations (2001)
3. Hua, J., He, Y., Qin, H.: Multiresolution heterogeneous solid modeling and visualization using trivariate simplex splines. In: ACM Symposium on Solid Modeling and Applications, SM, pp. 47–58 (2004)
4. Hua, J., He, Y., Qin, H.: Trivariate simplex splines for inhomogeneous solid modeling in engineering design. J. Comput. Inf. Sci. Eng. **5**(2), 149–157 (2005). doi:10.1115/1.1881352
5. Jackson, T., Liu, H., Patrikalakis, N., Sachs, E., Cima, M.: Modeling and designing functionally graded material components for fabrication with local composition control. Mate. Des. **20**(2–3), 63–75 (1999). doi:10.1016/S0261-3069(99)00011-4
6. Kou, X., Tan, S.: Heterogeneous object modeling: a review. CAD Comput. Aided Des. **39**(4), 284–301 (2007). doi:10.1016/j.cad.2006.12.007
7. Kumar, V., Dutta, D.: An approach to modeling and representation of heterogeneous objects. J. Mech. Des. Trans. ASME **120**, 659–667 (1998). doi:10.1115/1.2829329
8. Lasser, D.: Bernstein-bézier representation of volumes. Comput. Aided Geom. Des. **2**(1–3), 145–149 (1985). doi:10.1016/0167-8396(85)90018-4
9. Mantyla, M.: An Introduction to Solid Modeling. Computer Science Press, College Park, MD (1988)
10. Mortenson, M.E.: Geometric Modeling. Wiley, New York (1985)
11. Ochiai, K.: Tire for motorcycle. Sumitomo Rubber Industries, Ltd. In: Kobeshi (JP) Assignee. US Patent 7,201,198 (2007)
12. Qian, X., Dutta, D.: Physics-based modeling for heterogeneous objects. J. Mech. Des. Trans. ASME **125**(3), 416–427 (2003). doi:10.1115/1.1582877
13. Requicha, A.G.: Representations for rigid solids: theory, methods, and systems. ACM Comput. Surv. **12**(4), 437–464 (1980). doi:10.1145/356827.356833
14. Schmitt, B., Pasko, A., Schlick, C.: Constructive sculpting of heterogeneous volumetric objects using trivariate b-splines. Vis. Comput. **20**(2–3), 130–148 (2004). doi:10.1007/s00371-003-0236-9
15. Warkhedkar, R., Bhatt, A.: Material-solid modeling of human body: a heterogeneous b-spline based approach. CAD Comput. Aided Des. **41**(8), 586–597 (2009). doi:10.1016/j.cad.2008.10.016
16. Yang, P., Qian, X.: A b-spline-based approach to heterogeneous objects design and analysis. CAD Comput. Aided Des. **39**(2), 95–111 (2007). doi:10.1016/j.cad.2006.10.005

Francisco Conde-Rodríguez received a B.Sc. in Computer Science from the University of Granada in 1992 and a M.Sc. in Computer Science from the University of Granada in 1994. He is a lecturer at the University of Jaén since 1994. His research interests include Solid Modeling, volume visualization and human-computer interaction.

Ángel-Luis García-Fernández received a B.Sc. in Computer Science from the University of Jaén in 1998, a M.Sc. in Computer Science from the University of Granada in 2000, and his Ph.D. in 2007 from the University of Granada. He is a lecturer at the University of Jaén since 2004. His research interests include Solid Modeling and multidisciplinary applications of Computer Science.

Juan-Carlos Torres received is M.Sc. in Physics in 1982 and his Ph.D. from the Universidad de Granada in 1992. He is presently a full professor of Software at the University of Granada. His research interests include solid modeling, geometric algorithms, virtual reality and geographic information systems. He has more than 100 publications. He is presently the Head of the Virtual Reality Lab of the University of Granada. He was Co-Chair of the Eurographics 2003 Conference, Co-Chair of SIACG'04 and Co-Chair of Digital Heritage 2015 in Granada. He has been the Chair of the Eurographics Spanish Chapter and a member of the Eurographics Executive Committee. http://lsi.ugr.es/~jctorres/.

Francisco-Ramón Feito-Higueruela received the B.Sc. degree in mathematics from the University Complutense of Madrid and the Ph.D. degree in computer science from the University of Granada, Spain. Currently, he is a full professor at the Escuela Politécnica Superior of Jaén, being the head of the Graphics and Geomatics Research Group. He was the head of the Department of Computer Science from 1993 to 1997, the vice chancellor in charge of studies and quality from 1997 to 1999, and the vice chancellor in charge of research and international affairs from 1999 to 2002. His research interests include formal methods for computer graphics, geometric modeling, computational geometry, geographical information sciences, GIS 3D and Smart cities.