



HAL
open science

Retroactive Decomposition For Edge Matching Process in Parametric Systems

Mehdi Baba-Ali, Xavier Skapin, David Marcheix

► **To cite this version:**

Mehdi Baba-Ali, Xavier Skapin, David Marcheix. Retroactive Decomposition For Edge Matching Process in Parametric Systems. 2010. hal-02463998

HAL Id: hal-02463998

<https://hal.science/hal-02463998>

Preprint submitted on 2 Feb 2020

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.



Laboratoire SIC
STIC CNRS FRE n°2731, Bât. SP2MI, Téléport 2,
Bvd Marie et Pierre Curie, BP 30179,
86962 Futuroscope Chasseneuil Cedex, France
Téléphone : +33(0)5.49.49.65.67
Télécopieur : +33(0)5.49.49.65.70



***Retroactive Decomposition For Edge
Matching Process in Parametric Systems***

M. Baba-ali, X. Skapin and D. Marcheix
{baba-ali, skapin}@sic.univ-poitiers.fr, marcheix@ensma.fr

2010

Research Report N° 2010-xx

Retroactive Decomposition For Edge Matching Process in Parametric Systems

M. Baba-ali, X. Skapin and D. Marcheix
{baba-ali, skapin}@sic.univ-poitiers.fr, marcheix@ensma.fr

2010

Abstract

Nowadays, parametric systems support history-based, constraint-based and feature-based modeling. One of the main tasks of these systems is to maintain the design history. Unfortunately, most of them fail during the reevaluation phase when various kinds of topological changes occur. This issue is known as "persistent naming" which refers to the problem of identifying entities in an initial model and matching them in the reevaluated model, especially for edges resulting from non-planar intersections. Although related works ensure the uniqueness of the references to topological entities, they often neglect the shape characteristics of surfaces and give results different from those expected during design reevaluation. For this reason, those works are still not enough powerful to find the right mapping in some situations. We propose in this paper a method to add some additional information about surfaces to improve such works. We compute those information by making a decomposition of surfaces into sub-surfaces: hump(s) and/or hollow(s). Moreover, when the faces which intersect are deformed during the reevaluation phase, a retroactive decomposition operation is applied. This last enables to map the sub-surfaces resulting from the initial decomposition in order to have the right edge mappings.

Keywords: Persistent naming; CAD; Parametric design; Reevaluation

1 Introduction

The feature-based parametric modeling has acquired, during the last 15 years, an increasingly important place in domains such CAD, architecture or geology. One of the reason for such success lies in the ability of feature-based systems to allow a rapid change of the geometry by a simple change of feature's parameter. Nevertheless, these systems share a number of shortcomings related to their inner structure [9]. For example, lack of common definition of features make the data exchange and sharing difficult, if not impossible; chronological dependency of features in the modeling process reduces the flexibility of modeling sequences; the sequential and automatic reevaluation of feature would easily generate a geometry and a topology different from those expected.

All of those problems are pretty clearly related to one basic modeling issue: the semantics of feature, which describes the user's intents, is not captured throughout the modeling process. In fact, feature and its parameters (dimension values and references to topological entities: vertices, edges and faces) depend on the geometric elements computed in previous steps. This dependence on geometry alters the interpretation of features.

The first consequence of not taking semantic in feature specification into account is persistent naming problem. In current commercial parametric systems, a feature could refer to topological entities resulting from old features of the modeling process. Then, when those features are reevaluated, the first one may refer to a different entity unexpectedly, or even cannot find reference. As a result, an unforeseen geometry is generated.

A typical example is shown in Fig. 1, where the initial model is designed by means of a parametric specification containing four successive operations. The fourth one consists of rounding an edge e_2 resulting from the intersection of the top face f_1 of the swept block with the lateral face f_2 of the slot. A parameter of this constructive gesture in the parametric specification is then a reference to this edge e_2 . If the initial model is saved after this fourth step, the current instance no longer contains edge e_2 : it was removed by the rounding function. Thus the rounding function can not use this edge as an input parameter. Therefore "names" are needed to represent and distinguish the entities referenced in the parametric specification whether or not they exist in the current instance. These names must be unambiguous and must contain some characterizations that enable to retrieve the same or the corresponding entities when the model is reevaluated. This characterization is more complex for parametric models, for which the entities and the number of entities change from one evaluation to another. Let us return to the above example, but this time in the reevaluated model. We notice that, at step 3, edge e_2 has been split into edges e_3 and e_4 . At step 4, the problem is to determine which edge(s) has(ve) to be rounded. Thus, the problem is to match edge e_2 with edges e_3 and e_4 despite topology changes. It is thus necessary to have, in addition to the naming mechanism, a robust matching mechanism regarding reevaluation.

Most of existing approaches on persistent naming ([1, 5, 12, 13]) define several naming mechanisms: a face naming mechanism and other mechanisms defined especially for entities of lower dimension (vertices, edges). In addition, a few amount of literature has been published on the persistent naming of both higher dimensional (volumes, hyper-volumes) or aggregative entities (wires, shells¹, aggregation of volumes). In this context, the recent paper by [10] proposes a solution for processing shells. He underlines the inability of face-based systems to define an unique naming mechanism for the aggregative structures and the entities of

¹A shell is an aggregate of faces.

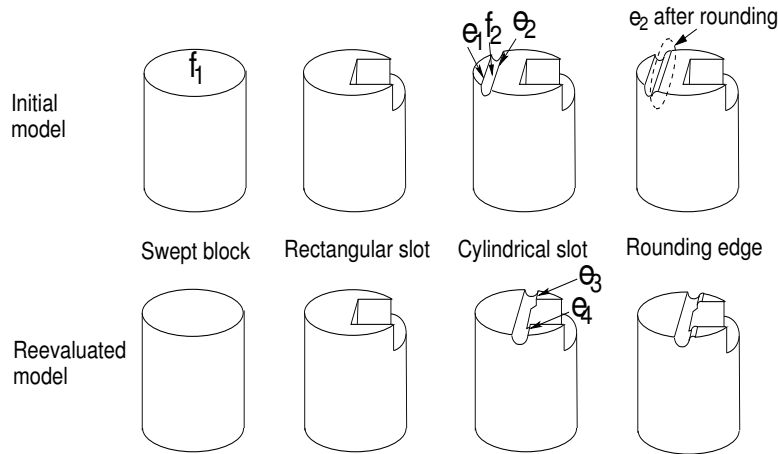


Figure 1: Naming and matching problems.

any dimension. In order to overcome this shortcoming, an edge-based hierarchical persistent naming system (see figure 2) should be implemented.

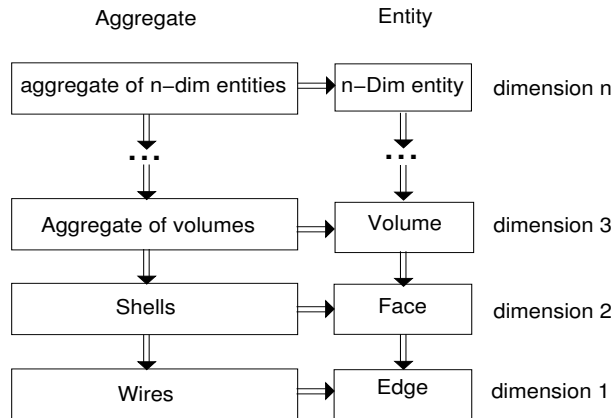


Figure 2: An edge-based hierarchical system for persistent naming problem. Each n -dimensional entity and each aggregate of n -dimensional entities is composed of $(n - 1)$ -dimensional entities.

To address this issue, the first task consists in defining a robust name structure and a matching technique for edges. This task is partially achieved in [2] for the edges resulting from planar intersections. However, it remains problematic when non-planar surfaces intersect. Recent works (Biddarra [5], Wang [12] or Wu [13]) deal with this issue by using some additional information, but those information are still not enough powerful to find the right mapping in some situations. The reason is that those approaches do not consider sufficiently the shape characteristics of surfaces. As the shape of a surface is the main intuitive criterion which allows the user to identify what he sees and that curvature constitutes an elementary unit of the shape perception.

The example presented in figure 3 illustrates pertinence and stability problems of Wu's and Wang's naming approaches during the reevaluation process. In this example, the parametric

model contains four constructive gestures: construction of a block by extrusion of the front face, construction of face F , difference Boolean operation between the block and face F and then rounding of the edge e_3 .

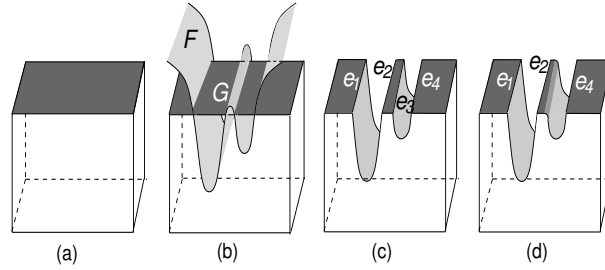


Figure 3: Parametric model consisting of four modeling operations.

During the reevaluation process, the position of F is changed and the topology of the object too. Figure 4 depicts different parts resulting from the difference Boolean operation between surfaces F and G of figure 3(c). Let us parts of figure 4 two by two, and assume that the second part is the result of the reevaluation of the first one after changing the position of F . In order to be able to reevaluate the fourth constructive gesture (rounding operation) and all the more as the topology is now different, it is necessary to calculate in the reevaluated model the edge corresponding to e_3 (i.e. to calculate the mapping of edge e_3). The mapping calculus takes the geometrical characteristics of each edge (for example, the geometrical position on a parametric surface for [13] and the orientation for Wang [12]) into consideration and it returns the results in table 1 (see the 4th and 5th column).

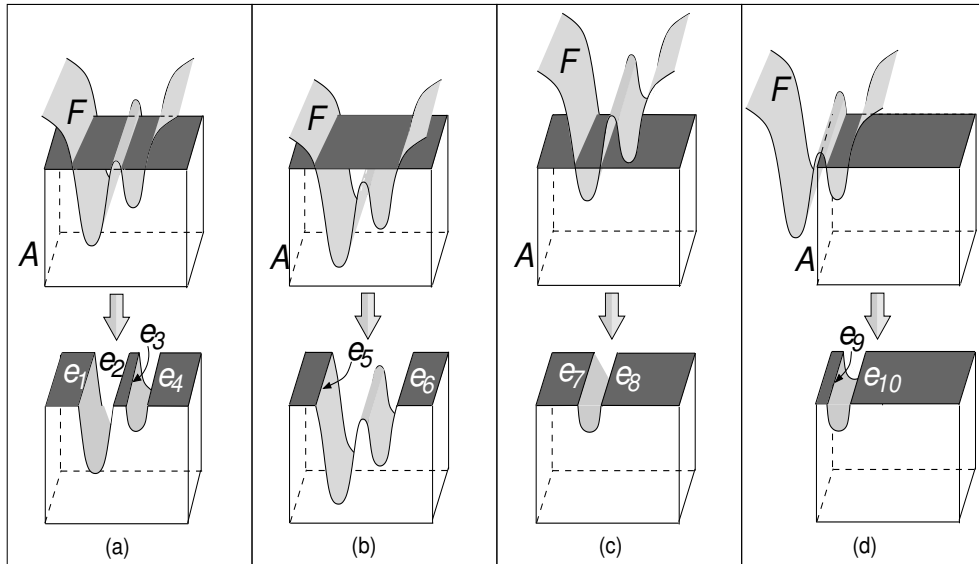


Figure 4: Different parts resulting from the Boolean difference operation of figure 3 as parameter values change.

Most of those results are different from those expected (not suitable with the user’s design intents). In fact, the mappings to be found depend on three characteristic regions (two hollows and one hump) of surface F . For example, in the fourth case, edge e_5 should be mapped to edge e_7 and e_8 because e_5, e_7 and e_8 result from the intersection of the left hollow of F with the top face of block A : i.e. G . In this case, the edge e_6 is excluded from the matching process because it results from the intersection of the right hollow of F with G . Moreover, since edges e_7 and e_8 do not have the same orientation, then e_5 is mapped to e_7 .

Table 1: Computation of the correspondance between edges showed in figure 4. $edge_1 \leftrightarrow edge_2$ means that $edge_1$ is mapped to $edge_2$.

Case	Matching between	Expected mappings	Wu’ edges mapping	Wang’ edges mapping
1	part a and part b	$e_1 \leftrightarrow e_5$ $e_4 \leftrightarrow e_6$	$e_1 \leftrightarrow e_5$ $e_2 \leftrightarrow e_6$	$e_1 \leftrightarrow e_5$ $e_2 \leftrightarrow e_6$
2	part a and part c	$e_1 \leftrightarrow e_7$	$e_1 \leftrightarrow e_7$ $e_2 \leftrightarrow e_8$	$e_3 \leftrightarrow e_7$ $e_2 \leftrightarrow e_8$
3	part a and part d	$e_3 \leftrightarrow e_9$ $e_4 \leftrightarrow e_{10}$	$e_1 \leftrightarrow e_9$ $e_2 \leftrightarrow e_{10}$	$e_3 \leftrightarrow e_9$ $e_2 \leftrightarrow e_{10}$
4	part b and part c	$e_5 \leftrightarrow e_7$	$e_5 \leftrightarrow e_7$ $e_6 \leftrightarrow e_8$	$e_5 \leftrightarrow e_7$ $e_6 \leftrightarrow e_8$
5	part b and part d	$e_6 \leftrightarrow e_{10}$	$e_5 \leftrightarrow e_9$ $e_6 \leftrightarrow e_{10}$	$e_5 \leftrightarrow e_9$ $e_6 \leftrightarrow e_{10}$
6	part c and part d	no mapping	$e_7 \leftrightarrow e_9$ $e_8 \leftrightarrow e_{10}$	$e_7 \leftrightarrow e_9$ $e_8 \leftrightarrow e_{10}$

Therefore, to find right mappings we should decompose the topological surface into characteristic regions (sub-surfaces). We use, for that, the surface decomposition process we have defined in [4]. The replacement of the original surface with sub-surfaces in the wang’s, Wu’s and bidarra’s matching process in addition to the orientation information of edges lead to the right mappings.

This decomposition does not work anymore when the surfaces which interact are deformed (as we can see in figure 5). Since, in this case the resulting sub-surfaces are not the same as before. Indeed, in figure 5, the surface F of figure 4 is deformed into a new surface F' (for example by moving the control points in the parametric expression of the surface). The edges resulting from the intersection of F ’s sub-surfaces (resp. F' ’s sub-surfaces) with G can not be correctly mapped because the sub-surfaces of F are different from those of F' . For instance, the edges of the resulting part in figure 4(a) (e_1, e_2, e_3 and e_4) can not be correctly mapped to the edges of the resulting part in figure 5(b) (e_{15}, e_{16}, e_{17} and e_{18}) because, the right hollow of F does not completely correspond to the right hollow of F' . As a result, with a classical decomposition [4], we do not know to locate edges e_{15}, e_{16}, e_{17} and e_{18} by report to surface F and vice versa (i.e. edges e_1, e_2, e_3 and e_4 by report to surface F').

So, when we consider that the right hollow of F corresponds to the two right hollows of F' (since in reality the right hollow of F is deformed into two right hollows of F'), the mapping between edges is possible. The principle is to map the edges resulting from the left hollow of

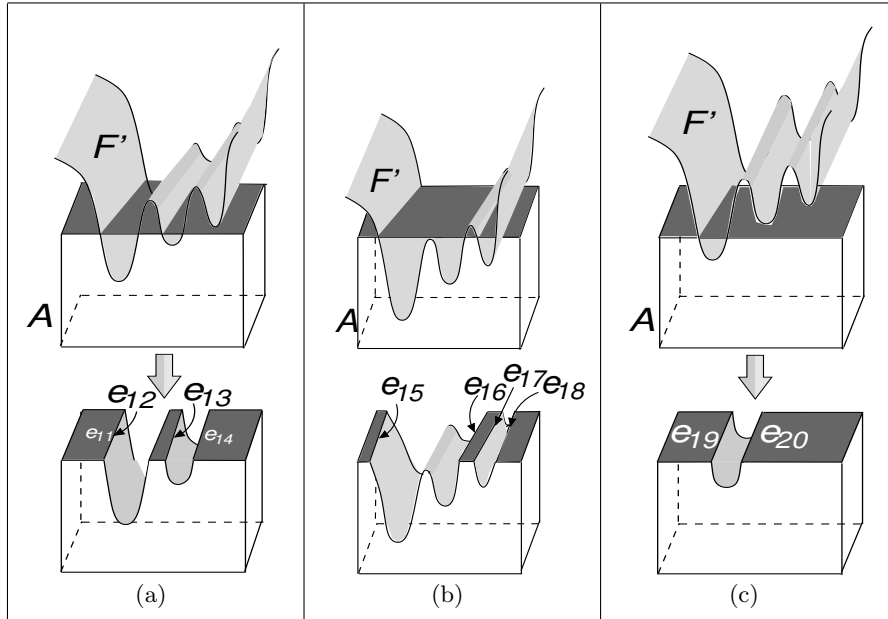


Figure 5: Different parts resulting from the Boolean difference operation of figure 3 as surfaces which interact are deformed.

F with the edges resulting from the left hollow of F' (resp. to map the edges resulting from the right hollow of F with the edges resulting from the two right hollows of F' and the edges resulting from the central hump of F with the edges resulting from the left hump of F'). This can be done by defining a retroactive decomposition method.

In this paper, we intend to define retroactive decomposition method to carry out the matching between edges when the surfaces which interact are deformed. This method is based on the decomposition process defined in [4]. It copies the sub-surfaces of a surface in the initial model on the corresponding deformed surface in the reevaluated model. This enables to locate from which sub-surface interaction the generated edges come from. The goal is to capture more semantics and thus to have the right mappings

The paper is structured as follows. In Section 2, related research work is reviewed and we give some details on Wang's approach. In Section 3, we explain how our decomposition method offers better results than Wang's. Section 4 is dedicated to the recursive decomposition process for addressing the deformed surfaces problem. We give some concluding remarks and perspectives in Section 5.

2 Related works

Every persistent naming approach [9] addresses the same fundamental issues: characterizing and matching topological entities. In planar context, the proposed solutions [1, 6, 8, 7] use either purely topological information (including topological neighborhood, local topological orientation), or both topology and geometry (parameterization of local spaces, for example). In non-planar cases, the litterature methods [5, 13, 12] use some additional geometrical information to remove ambiguity, when several entities have the same topological neighborhood (for example, the characterization of the edges resulting from the intersection of two

cylinders). Unfortunately, those solutions are not enough.

2.1 Biddara’s approach

For Bidarra and al. [5], the solution to persistent naming problem is feature-based. They consider a parametric feature model only as interrelated instances of persistent entities: reference classes, declarative feature classes and procedural feature classes. The procedural features, e.g. chamfers and blends, are associated with edges whose references may present ambiguity in the case of intersection of non-planar faces. Thus, for solving this ambiguity, [5] defined an original method. This method consists in automatically defining (one or more pairs of) half-spaces so that no pair of edges lies in the same (combination of) half-space(s). It is efficient but too restrictive since this solution remains available only for applications lying upon quadric surfaces: planar, cylindrical, spherical, and toroidal. The remainder of this paper will not deal with such method.

2.2 Wu’s approach

Wu and al. [13] consider parametric models as a sequence of regularized operations of construction between a part body and the body of an original feature. As the body of an original feature is created, every face of this feature is attached with a name (called *ON* for Original Name). Then a Boolean operation is conducted between the part and the body with names. *ONs* are stored along with geometry and are propagated to the descendant faces, in the case of scission, fusion or modification, which makes it possible to link each contingent face to its invariant initial face. An *ON* is made of an identifier of constructive operation (Sweep feature, Revolve feature, Chamfer feature), and identifiers which make it possible to know which invariant entities the resulting face comes from. For example, in the case of the Chamfer illustrated in figure 6, the part is made by subtracting one block from the other and chamfering edges e_1 and e_2 . Although the original feature body contains eight faces, actually only faces (a, b, e, f) and (b, c, d, e) are the feature faces. Moreover, faces F_1 , F_2 and F_3 are the feature faces in the cut Extruded feature, whose identity value is assumed to be 6. This Extruded feature has a profile (as the Sweep feature), which consists of four elements, $I1$, $I2$, $I3$ and $I4$. The profile’s identity is 5. The *ON* of a swept entity is given by :

$$ON(F) = [FeatID, FeatID_p, ID_{element}, FeatID_{path}, ID_{trajectory}]$$

where *FeatID* is the feature identity of the Sweep, *FeatID_p* (resp. *FeatID_{path}*) is the identifier of the profile (resp. of the path), and *ID_{element}* (resp. *ID_{trajectory}*) identify one profile edge (resp. one trajectory edge). In case of Extrude feature, which is formed by sweeping a profile along the normal of the profile, the *FeatID_{path}* and *ID_{trajectory}* can be set to zero. Then, the context of the *ON* for F_1 is :

$$ON(F_1) = \{6, 5, ID_{I2}, 0, 0\},$$

When an entity (vertex, edge or face) is referenced, *ONs* are propagated along the descendant faces, so several faces can be assigned the same *ON*. Wu and al. propose to use a *RN*

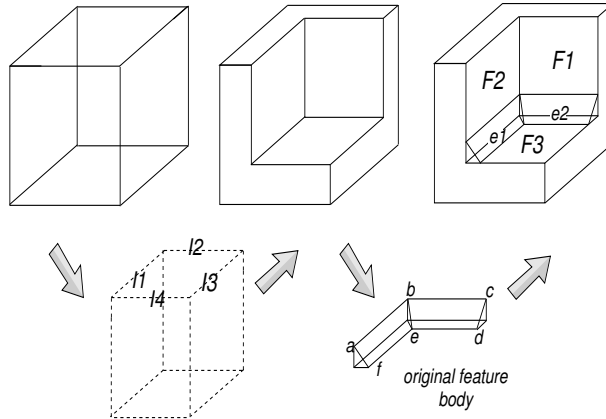


Figure 6: Name chamfer feature.

(called *RN* for Real Names) made of the initial *ON* and some additional information allowing to eliminate ambiguity in construction (called *PSI* for Parametric Space Information). When regenerating the whole part, all the real names in the part model should be retrieved to prepare data for the reevaluation.

Those purely geometric information are based on the parameter space (u, v) of the surfaces they rely on. An arrangement of the subdivided faces in u or v directions, based on the distance between a characteristic representative point and the origin of the reference coordinate system, makes it possible to associate a different number with each face. To illustrate how to obtain the parametric space information of a topological entity, let us take again the example of figures 4(b) and (c) and compute the parametric space of surfaces F and G (the top face of block A in figure 4). The result is shown in figure 7 and the coordinates of each point are depicted in table 2. According to the principle of selecting feature points, as defined in [13], the feature points of topological faces G_1, G_2, G_3, G_4, F_1 and F_2 are A, B, C, D, E and N , respectively. After sorting the order of those subdivided faces, the real names of G_1, G_2, G_3, G_4, F_1 and F_2 are $[ON(G), 1, 2], [ON(G), 2, 2], [ON(G), 1, 2], [ON(G), 2, 2], [ON(F), 1, 1]$ and $[ON(F), 1, 1]$ respectively.

Table 2: (u, v) value for each point.

Point	A	B	C	D	E	P
(u, v)	(0, 0)	(0.8, 0)	(0, 0)	(0.52, 0)	(0.15, 0.1)	(0.55, 0.1)

Other topological entities (vertices, edges) are named following the *RN* of the adjacent faces and, in a similar way, with an information on the parameter space. In the previous case, the real names of edges e_5, e_6, e_7 and e_8 are $[RN(G_1), RN(F_1), 1, 1], [RN(G_2), RN(F_1), 1, 1], [RN(G_3), RN(F_2), 1, 1]$ and $[RN(G_4), RN(F_2), 1, 1]$, respectively.

As discussed above, if a feature references some topological entities, these entities should be recorded as topological names (real names). So, while rebuilding the whole part or editing a feature, those referenced entities should be retrieved from the real names recorded in the feature. In the example of figure 7, the part of figure 4(b) is reevaluated to the part of

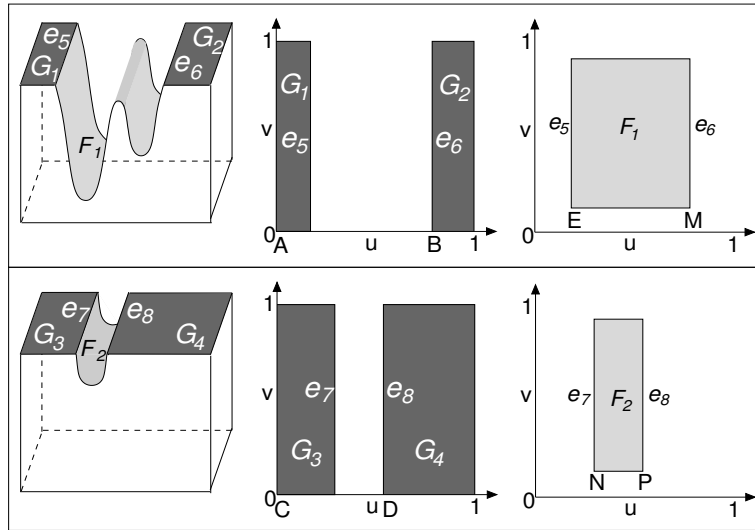


Figure 7: Computing the parametric space information.

figure 4(c). Then, by using the real names of edges e_5 and e_6 , we retrieve edges e_7 and e_8 in the new context (case 4, column 4 in table 1): the real names of G_3 and G_4 are the same as those of G_1 and G_2 , respectively. Unfortunately, this mapping is not valid because e_6 does not match e_8 . Obviously, e_6 results from the intersection of the left hollow of face F with face G in the part of figure 4(b) whereas e_8 results from the intersection of the right hollow of the same face with G in the part of figure 4(c).

The real names used by Wu guarantee the uniqueness of the references (any reference should be globally unique) in the identifying process, but not the relevance of the matching after the reevaluation step.

2.3 Wang's approach

Wang and al. [12] propose a general definition of a parametric family, based on the principle of geometric continuity introduced by Shapiro and al. [11]. They develop a "semantic id scheme" based on geometry rather than on topology because they consider that geometry is more robust and topology is volatile in the parametric family. In this surface-based scheme, *id* of entity includes information of construct relation (surface segment) and geometric meaning of identification (orientation and/or adaptation). For an *id* of an entity, the surface segment includes the reference to its corresponding unbounded geometric entity and the references to its boundary surface *ids*. For example, in figure 6, a face F_2 is generated by a cut feature c . If one assume that faces F_1, F_2, F_3, F_4 and F_5 are referring to plans S_1, S_2, S_3, S_4 and S_5 respectively, then F_1 will have the name $FACE(CUT(c) :: SURFACE(S_1) - SURFACE(S_2), SURFACE(S_3), SURFACE(S_4), SURFACE(S_5))$. And the edge e_3 will have the name $EDGE(CUT(c) :: CURVE(SURFACE(S_1) + SURFACE(S_5)) - SURFACE(S_2), SURFACE(S_4))$.

This characterization works well only when two planes intersect at one edge. In other cases (intersection of quadratic or higher order surfaces), The information of orientation and/or adaptation is necessary. Fig. 4 depicts some instances of this configuration. For Wang and al. the orientation of the surface $f(x, y, z) = 0$ at point $p = (x, y, z)$, $I_\sigma(f, p)$, corresponds

to its first gradient:

$$I_\sigma(f, p) = \nabla f(x, y, z) = \left[\frac{\sigma}{\sigma x} \quad \frac{\sigma}{\sigma y} \quad \frac{\sigma}{\sigma z} \right]^T f(x, y, z).$$

Also, the orientation of edge e (straight or curved) which result from the intersection of two surfaces $f(x, y, z) = 0$ and $g(x, y, z) = 0$, at point $p = (x, y, z)$, corresponds to the cross product of the first gradients of f and g at point p :

$$I_{11}(f, g, p) = I_\sigma(f, p) \times I_\sigma(g, p).$$

In figure 4(a), plane $G : z = -2$ intersect with surface $F : z + \frac{1}{15}e^{-(x+5)^2+4} + e^{-0.09x^2+1} = 0$. The four intersecting edges e_1, e_2, e_3 and e_4 , bounded by planes $y + 1 = 0$ and $y - 1 = 0$, have orientations $[0 \ -3.1879 \ 0]^T$, $[0 \ 2.2362 \ 0]^T$, $[0 \ -0.6637 \ 0]^T$ and $[0 \ 0.6647 \ 0]^T$, respectively.

Extra care should be given to cases that two or more edges have the same orientation (such as the left and the right edges resulting from the intersection of plane $z = 0$ with surface $x^3 - x - z = 0$). In this situation, an additional information (adaptation) besides orientations is needed². The adaptation of the surface $f(x, y, z) = 0$ at point $p = (x, y, z)$, $I_\sigma^2(f, p)$, is defined as the second order gradient:

$$I_\sigma^2(f, p) = \nabla^2 f(x, y, z) = \left[\frac{\sigma^2}{\sigma x^2} \quad \frac{\sigma^2}{\sigma y^2} \quad \frac{\sigma^2}{\sigma z^2} \right]^T f(x, y, z).$$

The adaptation of the intersection edge made by surfaces f and g can be defined as

$$I_{12}(f, g, p) = I_\sigma(f, p) \times I_\sigma^2(g, p).$$

$$I_{21}(f, g, p) = I_\sigma^2(f, p) \times I_\sigma(g, p).$$

$$I_{22}(f, g, p) = I_\sigma^2(f, p) \times I_\sigma^2(g, p).$$

So now, while reevaluating whole features, the *id*'s of referenced entities, both in the initial (old *ids*) and reevaluated (new *ids*) models are compared. Indeed, Wang begin by matching the entities having the same "surface segment". Then, he considers that if only one edge results from the intersection of two surfaces, or no additional geometric information (orientation/adaptation/gradient) is included in either of the old and new *ids*, there is an exact mapping for *ids*. For all other cases, the matching is based on closeness of edges. The k -closeness of edge e_1 (resulting from the intersection of two surfaces f_1 and g_1) and edge e_2 (resulting from the intersection of two surfaces f_2 and g_2), at points p_1 and p_2 belonging to e_1 and e_2 , respectively, is defined by

$$k\text{-close}(f_1, g_1, f_2, g_2, p_1, p_2) = |p_1 - p_2| + \sum_{i=1}^k \sum_{j=1}^k |I_{ij}(f_1, g_1, p_1) - I_{ij}(f_2, g_2, p_2)| \quad (k \geq 0)$$

Thus, for each pair of edges to map, Wang computes k -close³. An $m \times n$ closeness matrix R is generated by listing each of the reevaluated model's edges as row indices and each of

²In the same way, if adaptation still not enough then a higher order gradient must be computed.

³Generally, k is the highest order of surface gradient in the edge's *id*.

the initial model's edges as column indices. Once the closeness matrix is built, the mapping of edges can be done by selecting the lowest k -close value in each row. let us take again the example of figure4(a) and assume that the plane is changed to $z = -0.5$, two new edges e_5 and e_6 (see figure 4(b)) need to be mapped to old edges e_1, e_2, e_3 and e_4 . The 1-closeness matrix is computed at the intersection points with plane $y = 0$.

$$0 - close \begin{pmatrix} 1.6311 & 2.8719 & 4.8478 & 8.4375 \\ 10.2634 & 8.4786 & 6.3633 & 2.9076 \end{pmatrix}$$

$$1 - close \begin{pmatrix} 3.4742 & 6.4529 & 5.5289 & 10.447 \\ 13.8417 & 10.3244 & 7.4174 & 3.1819 \end{pmatrix}$$

The lowest value in the first row (resp. in the second row) corresponds to the 0-close between edge e_5 and e_1 (resp. edges e_6 and e_4). Therefore, the matching process of Wang maps edges e_5 and e_6 to e_1 and e_4 , respectively.

Despite the satisfactory results achieved, Wang's approach presents some anomalies:

1. when an $m \times n$ matching is carried out (where $m > n$), a wrong mapping can be done. In the previous example, if we assume that the model of figure 4(b) is first constructed. The 1-close matrix, in this case, is equal to the transposed of the 1-close matrix presented above. And this matrix leads to the wrong mapping of e_1, e_2 and e_3 to e_5 , and e_4 to e_6 .
2. when one considers the edges of figure 4(c) as the result of intersection of plan $G : z = -3.5$ and surface $F : z + \frac{1}{15}e^{-(x+2)^2+4} + e^{-0.09(x-3)^2+1} = 0$. Then, the plan G and surface F are changed to $z = -2$ and $z + \frac{1}{15}e^{-(x+5)^2+4} + e^{-0.09x^2+1} = 0$. We obtain the following 1-close matrix

$$1 - close \begin{pmatrix} 4.6894 & 6.7979 & 3.2123 & 7.3957 \\ 9.8238 & 2.9172 & 4.3425 & 5.2495 \end{pmatrix}$$

which leads to wrong mapping of e_7 to e_3 and e_8 to e_2 .

3. this method uses faces in the implicit format: it uses the corresponding unbounded surfaces of those faces. Unfortunately, in most of modeling systems, many objects are not defined using implicit surfaces.

3 Shape-based matching process

Solving the persistent naming problem requires: (1) the characterization of topological entities (vertices, edges, faces) resulting from modeling operations, then (2) the definition of a matching process capable of determining the corresponding entities. In [12], Wang has proposed new method for matching edges resulting from the intersection of non-planar faces. Unfortunately, the fact that the shape characteristics of surfaces are neglected, this new matching process is still not enough powerful to find the right mapping in some situations (as shown in figure 4). In order to remove those shortcomings, we propose to include these characteristics (defined in [4]) into the matching process.

In this section, we present our decomposition process which is introduced in [4]. We show how the method leads to better results than Wang's et those information will be useful to understand the recursive decomposition method introduced in Section 4.

3.1 Surface decomposition

In a non-planar modeling, all methods use geometric properties to solve the persistent naming issue. Although they ensure the uniqueness of the references to topological entities, they often neglect the shape characteristics of surfaces and give results different from those expected.

When manipulating the model, shape characteristics such as curvature are easier to perceive than the topology of surfaces. The reason is that: a) the shape is the main criterion which allows the user to identify what he sees; and b) the curvature constitutes an elementary unit of the shape perception. As we see in figure 4, the shape of F allows the distinction between the left and the right hollows of this surface. This leads us to consider a topological surface as a collection of sub-surfaces discernible and designable by the shape.

Matching process first decomposes surfaces based on the curvature criterion. The surfaces involved in this decomposition are the invariant surfaces: i.e. surfaces that can be, completely and unambiguously, characterized by the structure of a modeling operation and its input parameters, independently of involved values. In figure 4, invariant surfaces include all faces of block A , surface F , the face resulting from the rounding gesture, etc. The resulting sub-surfaces correspond either to hollows or humps. Actually, the decomposition process is performed following the steps below:

1. the calculation of local extremums (minimums and maximums) in a 3-dimensional space: a local maximum corresponds to the top of a hump whereas a local minimum corresponds to the bottom of a hollow (in x and y directions).
2. the calculation of inflexion curves: an inflexion curve separates two local extremums.
3. the merge of sub-surfaces: when a sub-surface does not contain a local extremum, it is merged into another one. The goal is to obtain one sub-surface per local extremum.

3.1.1 Calculation of local extremums

If one considers the studied surface as a function with two variables $f(x, y)$, a local extremum is a point $p_e = (x_e, y_e)$ that satisfies: 1) $f_x(x_e, y_e) = 0$; 2) $f_y(x_e, y_e) = 0$ (where f_x and f_y are the first derivative functions with respect to x and y) and 3) $f_{xx}(x_e, y_e) \times f_{yy}(x_e, y_e) - (f_{xy}(x_e, y_e))^2 > 0$ (where f_{xx} , f_{yy} and f_{xy} are the second derivative functions with respect to x , y and both x and y , respectively).

As an example, we take the following function: $f(x, y) = x * e^{(-x^2-y^2)}$ (see figure 8).

3.1.2 Calculation of inflexion curves

The inflexion curves are a collection of points $p_i = (x_i, y_i)$ which satisfy: 1) $f_{xx}(x_i, y_i) = 0$ and 2) $f_{yy}(x_i, y_i) = 0$. They delineate the surface parts (sub-surfaces) containing each local extremum. The inflexion curves of surface $f(x, y) = x * e^{(-x^2-y^2)}$ delineate twelve sub-surfaces R_1, R_2, \dots, R_{12} , as shown in figure 9.

However, some generated sub-surfaces are without local extremums. In figure 9, sub-surfaces $R_1, R_2, R_3, R_5, R_6, R_8, R_9, R_{10}, R_{11}, R_{12}$ do not contain extremums depicted in

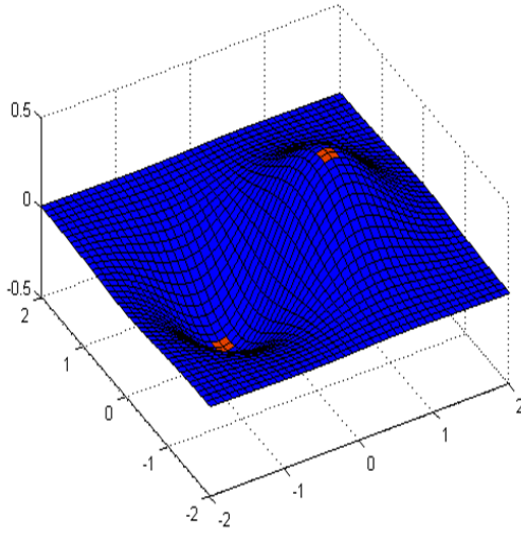


Figure 8: Computing a local extremums (red points).

figure 8. Therefore, a sub-surface merging is necessary in order to obtain one sub-surface per local extremum.

3.1.3 Merge of sub-surfaces

In order to obtain a surface decomposition with one local extremum per sub-surface, we merge the sub-surfaces bounded by the inflexion curves. To do so, we compute the distance between each point of a sub-surface without local extremum and a local extremum. The processed points are assigned to the nearest sub-surface with local extremum.

For the example of figure 8, sub-surfaces R_1, R_2, R_3, R_{11} and R_{12} are merged with sub-surface R_4 and sub-surfaces R_5, R_6, R_8, R_9 and R_{10} are merge with sub-surface R_7 , resulting respectively in sub-surfaces Z_1 and Z_2 (see figure 10).

To improve Wu's and Wang's approaches, we substitute the topological faces characterizing each edge of the model by sub-surfaces resulting from the decomposition operation. In figure 11, the decomposition of surface F, taken from the example of figure 4, gives three sub-surfaces (called S_1 for the left hollow, S_2 for the central hump and S_3 for the right hollow). By considering these sub-surfaces, the orientation information and the topological neighborhood of edges to map we can define the following matching method.

3.2 Matching process

Our matching process takes into consideration the sub-surfaces resulting from the decomposition operation presented in the previous sub-section. It characterizes the edges following both ON of the adjacent sub-surfaces (such as defined by Wu) and an additional information indicating orientation or adaptation (such as defined by Wang). Therefore, the matching is performed between the edges having the same topological neighborhood (in

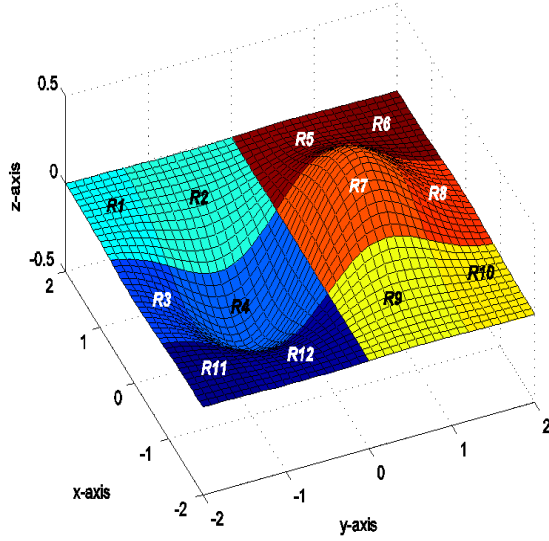


Figure 9: Computing an inflexion curves.

terms of sub-surfaces) by computing a suitable matrix k -close. For instance, let us take again the two matching configurations presented in section 2.3. In the first case, surface $F : z + \frac{1}{15}e^{-(x+5)^2+4} + e^{-0.09x^2+1} = 0$ intersect with plane $G : z = -2$ (see figure4 (a)) during the construction phase then plane $G : z = -0.5$ (see figure4 (b)) during the reevaluation phase. With our name structure, the edges resulting from these two intersections are characterized as follows:

- $e_1 = [ON(S_1), ON(G), [0 \ -3.1879 \ 0]^T]$;
- $e_2 = [ON(S_2), ON(G), [0 \ 2.2362 \ 0]^T]$;
- $e_3 = [ON(S_2), ON(G), [0 \ -0.6637 \ 0]^T]$;
- $e_4 = [ON(S_3), ON(G), [0 \ 0.6647 \ 0]^T]$;
- $e_5 = [ON(S_1), ON(G), [0 \ -1.3448 \ 0]^T]$ and
- $e_6 = [ON(S_3), ON(G), [0 \ 0.3904 \ 0]^T]$.

The edges to be mapped are: e_1 with e_5 and e_4 with e_6 . Furthermore, when we consider the intersection between surface $F : z + \frac{1}{15}e^{-(x+5)^2+4} + e^{-0.09x^2+1} = 0$ with plane $G : z = -3.5$ (see figure 4(c)) then plane $G : z = -2$ (see figure4 (a)), the matching between resulting edges e_1, e_2, e_3, e_4, e_7 and e_8 leads to the wrong mappings with Wu's and Wang's approaches (see table 1). Contrariwise, when we characterizes edges e_7 and e_8 of figure 4(c) as follows:

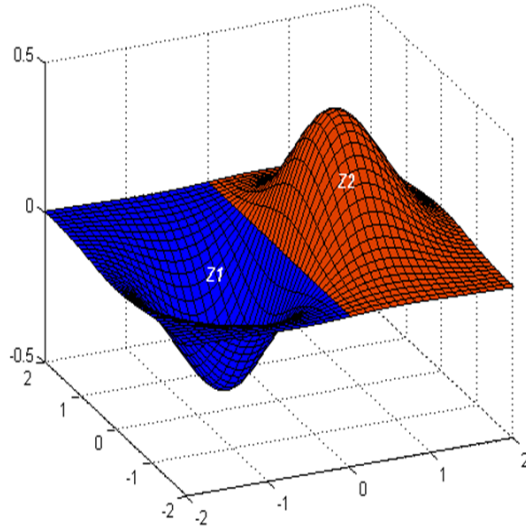


Figure 10: Surface decomposition.

- $e_7 = [ON(S_1), ON(G), [0 \ -2.3033 \ 0]^T]$ and
- $e_8 = [ON(S_1), ON(G), [0 \ 2.1582 \ 0]^T]$.

The edges to be mapped are: e_1 with e_7 and e_8 . The corresponding 1-closeness matrix is equal to:

$$1 - close \left(\begin{array}{cc} 4.6894 & 6.7979 \end{array} \right)$$

and thus the right mapping is e_1 to e_7 instead of e_3 and e_2 to e_7 and e_8 (as cited in subsection 2.3).

4 matching process after surface deformation

As explained above, the decomposition process improves the results obtained by Wang's approach. The principle is to map the edges resulting from the same pair of sub-surfaces. When the shape of the surface which interact is saved, the matching process of Wang does not cause any problems or difficulties (the sub-surfaces which interact remain the same both in the initial and the reevaluated models). In figure 11, the sub-surfaces S_1, S_2 and S_3 resulting from the decomposition of surface F and surface G (see figure 4) remains the same, both in the initial and in the reevaluated models. This changes, nevertheless, in the case where the surfaces which intersect are deformed after the reevaluation phase. In figure 5, the deformation of F into surface F' generates 5 sub-surfaces: S_4, S_5, S_6, S_7 and S_8 (see figure 12), instead of 3. As a consequence of this difference, the classical decomposition does not work anymore. A solution can be done by defining a retroactive decomposition method.

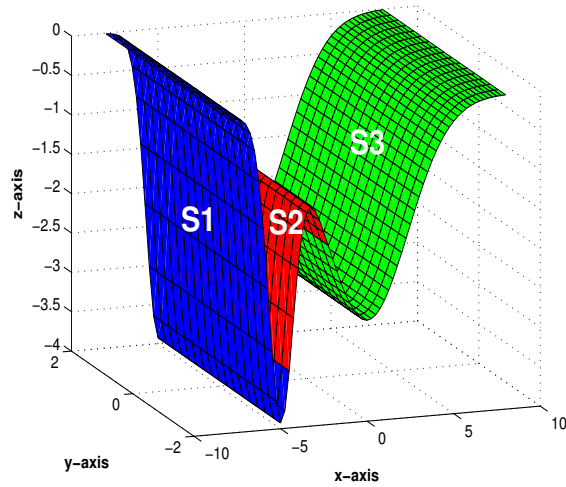


Figure 11: Decomposition of surface F from figure 4.

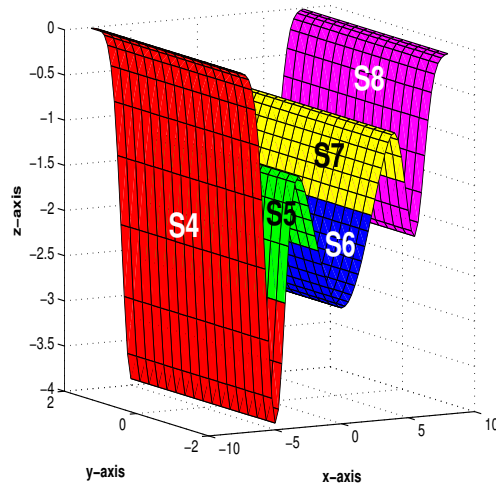


Figure 12: Decomposition of surface F' from figure 5.

4.1 Retroactive decomposition method

Suppose we have two surfaces which intersect during the construction process, F_1 and F_2 , and two other surfaces which intersect during the reevaluation process, F_3 and F_4 , such as F_3 and/or F_4 result from the deformation of F_1 and F_2 , respectively. An example of this configuration corresponds to the intersection of surface F with surface G during the construction process (as we can see in figure 4) then the intersection of surface F' with surface G during the reevaluation process (see figure 5). The matching between the edges which result from the intersection of F_1 with F_2 and F_3 with F_4 requires the retroactive decomposition of all those surfaces as follows:

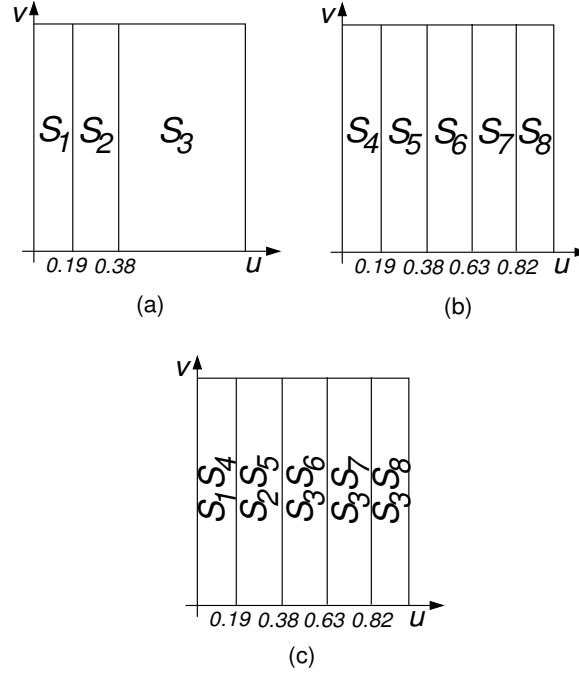


Figure 13: Retroactive decomposition operation in the parametric space. (a) Parametric space of F . (b) Parametric space of F' . (c) Copy of the decomposition of the parametric spaces of surface F on the parametric spaces of surface F' .

1. decomposition of surfaces F_1 , F_2 , F_3 and F_4 according to the principle defined in section 3.1. Let us be $R_{i1}, R_{i2}, \dots, R_{in_i}$ the sub-surfaces resulting from the decomposition of F_i and $1 \leq i \leq 4$. For the previous example, the decompositions of F and F' are depicted in figures 11 and 12, respectively. The decomposition of surface G generates sub-surface G itself;
2. computation of the parametric space of F_1 , F_2 , F_3 and F_4 in order to get a parametric description of resulting sub-surfaces. The parametric space of surfaces F and F' are depicted in figures 13(a) and (b), respectively;
3. copy the decomposition of the parametric spaces of surfaces F_1 and F_2 on the parametric spaces of surfaces F_3 and F_4 , respectively. This copy is performed as follows: for each point $p = (u, v)$ belonging to the parametric space of surface F_1 and F_3 (resp. F_2 and F_4), if $p \in R_{1j}$ and $p \in R_{3k}$ (resp. $p \in R_{2m}$ and $p \in R_{4n}$) then $p \in R_{1j}R_{3k}$ (resp. $p \in R_{2m}R_{4n}$) where $1 \leq j \leq n_1$, $1 \leq k \leq n_3$, $1 \leq m \leq n_2$, and $1 \leq n \leq n_4$. In figure 13, the copy of the parametric representation of sub-surfaces S_1 and S_2 of surface F (see figure13(a)) on the parametric representation of surface F' (see figure 13(b)) is depicted in figure13(c). The consequence on the analytic representation of surfaces F and F' is depicted in figures 14(a) and 14(b), respectively.

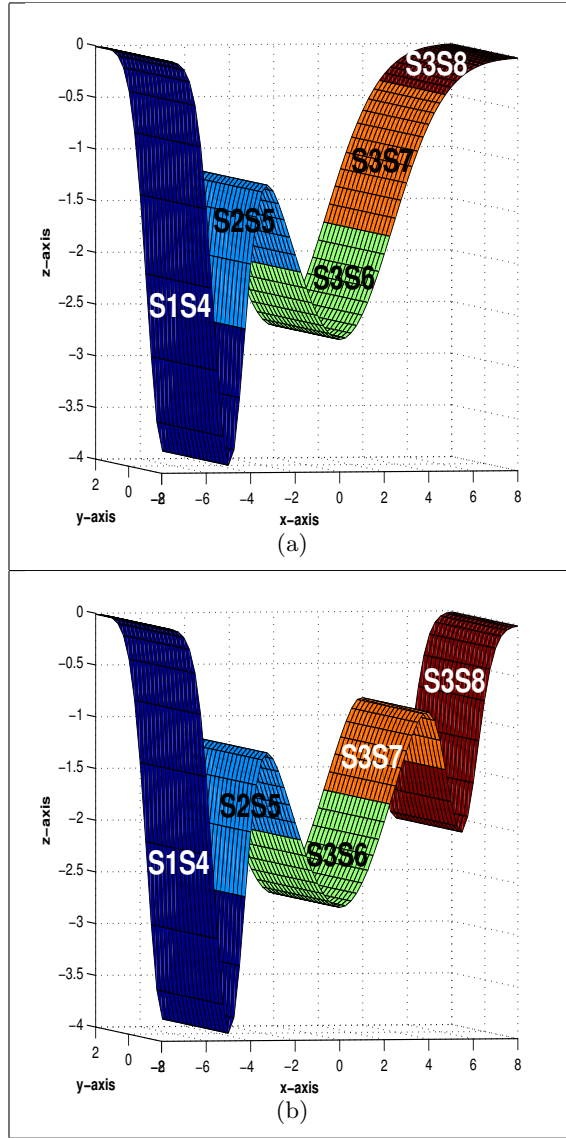


Figure 14: Retroactive decomposition applied on surfaces F and F' .

As explained above, the matching is performed between the edges having the same topological neighborhood (in terms of sub-surfaces) by computing a suitable matrix k -close. In order to show the effectiveness of our matching process, let us take the part of figure 4(a) and reevaluate it by deforming surface F into F' for get the part of figure 5(b). In the first case, surface F :

$$z + \frac{1}{15}e^{-(x+5)^2+4} + e^{-0.09x^2+1} = 0$$

intersect with plane $G : z = -2$ whereas in the second case deformed surface F' :

$$z + e^{-(0.4x+0.3)^2+1} + \frac{1}{15}e^{-(x+5)^2+4} + e^{-(x-4.5)^2+0.5} = 0$$

intersect with plane $G : z = -0.7$. The matching process of Wang gives the following 1-closeness matrix at the intersection points with plan $y = 0$:

$$1 - \text{close} \begin{pmatrix} 2.7763 & 11.8958 & 11.618 & 15.812 \\ 6.6644 & 7.9381 & 10.7131 & 10.4596 \\ 5.7679 & 5.5023 & 5.6972 & 9.3589 \\ 10.7065 & 1.3912 & 3.6379 & 4.4506 \end{pmatrix}$$

That is means that edges e_1 and e_2 are mapped on edge e_{15} and edges e_3 and e_4 are mapped on edge e_{16} whereas those mappings are obviously wrong. So, with our naming process, we characterize first edges e_1, \dots, e_4 and edges e_{15}, \dots, e_{18} as follows:

- $e_1 = [ON(S_1S_4), ON(G), [0 \ -3.1879 \ 0]^T]$;
- $e_2 = [ON(S_2S_5), ON(G), [0 \ 2.2362 \ 0]^T]$;
- $e_3 = [ON(S_2S_5), ON(G), [0 \ -0.6637 \ 0]^T]$;
- $e_4 = [ON(S_3S_7), ON(G), [0 \ 0.6647 \ 0]^T]$;
- $e_{15} = [ON(S_1S_4), ON(G), [0 \ -1.7971 \ 0]^T]$;
- $e_{16} = [ON(S_3S_7), ON(G), [0 \ 0.6139 \ 0]^T]$;
- $e_{17} = [ON(S_3S_7), ON(G), [0 \ -0.9167 \ 0]^T]$ and
- $e_{18} = [ON(S_3S_8), ON(G), [0 \ 1.3033 \ 0]^T]$.

According to those characterization, the edges to be mapped are: e_1 with e_{15} , e_4 with e_{16} and e_{17} . Edges e_2, e_3 and e_{18} does not have potential corresponding edges. The computation of 1-close matrix is necessary only for the matching between edges e_4 with edge e_{16} and e_{17} . It is equal to

$$1 - \text{close} \begin{pmatrix} 3.6379 & 4.4506 \end{pmatrix}$$

As a result the obtained mapping is e_4 to e_{16} .

5 Conclusion

In this paper, we have proposed a new method for matching edges resulting from the intersection of non-planar faces. our method takes into consideration the shape of non-planar faces and it considers Wang's naming approach in order to have the right mappings. Moreover, some additional information about surfaces are computed to improve the matching between edges. In fact, our method takes into consideration the shape of non-planar faces and it considers each surface as a collection of hollow(s) and/or hump(s). In practice, a decomposition operation according to the shape characteristics of surfaces is implemented. When the faces characterizing edges are deformed during the reevaluation phase, a retroactive decomposition operation is applied. That enables to map the sub-surfaces resulting from the initial decomposition in order to have the right mappings.

The perspectives of this work are: firstly, to use our method on several domain as CAD, architecture and geology; then, to integrate our method into the hierarchical persistent naming system we are currently developing.

References

- [1] D Agbodan, D Marcheix, G Pierra, C Thabaud. A Topological Entity Matching Technique for Geometric Parametric Models. SMI '03: Proceedings of the Shape Modeling International. IEEE Computer Society, 2003.
- [2] M Baba-ali, D Marcheix, X Skapin. An edge-based matching technique for geometric parametric models. Workshop on Computational Topology in Image Context CTIC 08, 2008.
- [3] M Baba-ali, D Marcheix, X Skapin. An Edge Matching Technique for Non-Planar Face Intersections in Geometric Parametric Models. SMI '09: Proceedings of the Shape Modeling International 2009, 201-208. IEEE Computer Society, 2009.
- [4] M Baba-ali, D Marcheix, X Skapin. A Method To Improve Matching Process by Shape Characteristics in Parametric Systems. Computer Aided Design & Applications 2009, 6(1-4): 341-350.
- [5] R Bidarra, PJ Nyirenda, WF Bronsvort. A feature-based solution to the persistent naming problem. Computer Aided Design & Applications 2005, 2(1-4): 517-526.
- [6] V Capoyreas, X Chen, CM Hoffmann. Generic naming in generative, constraint-based design. Computer-Aided Design 1996, 28(1): 17-26.
- [7] CM Hoffmann and R Juan. Erep: An editable, high-level representation for geometric design and analysis. Selected and Expanded Papers from the IFIP TC5/WG5.2 Working Conference on Geometric Modeling for Product Realization, 129-164. North-Holland Publishing Co., 1992.
- [8] J Kripac. A mechanism for persistently naming topological entities in history-based parametric solid models. SMA '95: Proceedings of the third ACM symposium on Solid modeling and applications, 21-30. Salt Lake City, Utah, United States, ACM, 1995.
- [9] D Marcheix, G Pierra. A survey of the persistent naming problem. SMA '02: Proceedings of the seventh ACM symposium on Solid modeling and applications, 13-22. Germany, ACM, 2002.
- [10] D Marcheix. A Persistent Naming of Shells. CAD-CG '05: Proceedings of the Ninth International Conference on Computer Aided Design and Computer Graphics, 259-268. IEEE Computer Society, 2005.
- [11] V Shapiro, DL Vossler. What is a parametric family of solids? SMA '95: Proceedings of the third ACM symposium on Solid modeling and applications, 43-54. Salt Lake City, Utah, United States, ACM, 1995.
- [12] Y Wang, BO Nnaji. Geometry-based semantic ID for persistent and interoperable reference in feature-based parametric modeling. Computer Aided Design 2005, 37(10): 1081-1093.
- [13] J Wu, T Zhang, X Zhang, J Zhou. A face based mechanism for naming. Recording and retrieving topological entities. Computer-Aided Design 2001, 33(10): 687-698.