Computer-Aided Design and Applications

http://www.cadanda.com

A Method to Improve Matching Process by Shape Characteristics in Parametric Systems

Mehdi Baba-ali[1], David Marcheix[2] and Xavier Skapin[3]

[1]XLIM-SIC University of Poitiers, babaali@sic.univ-poitiers.fr
[2]LISI-ENSMA, marcheix@ensma.fr
[3]XLIM-SIC University of Poitiers, xavier.skapin@univ-poitiers.fr

ABSTRACT

In current parametric systems, the persistent naming issue (based on edge mappings of intersecting surfaces) is not as fully supported as it should be. Unpredictability and ambiguity of models often happen during design reevaluation within systems. This reference deficiency is widely treated in the literature, especially about non-planar entities during design construction. 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. We propose in this paper a method to add some additional information about surfaces to improve such works. We compute those information by decomposing surfaces according to hump(s) and/or hollow(s). More precisely, our method use local extremums and inflexion curves to obtain one hump or hollow per sub-surface. The existing matching processes replace every surface with their corresponding sub-surfaces, leading to the right edge mappings.

Keywords: persistent naming, shape characteristics, matching process.
DOI: 10.3722/cadaps.2009.341-350

1. INTRODUCTION

Reuse of part model is an important issue in CAD, architecture or geology domains. Nowadays, feature-based parametric systems enable easy conception and modification of parts. Indeed, it has been estimated that almost 80% of all design tasks consists in adapting an existing model [9] by some simple modification of dimension values, constraint relations and feature definitions.

A feature-based parametric system contains the topological representation (i.e. *n*-Dim entities: vertex, edge, face, volume and topological adjacency relations between these entities), geometric representation (i.e. embedding of these entities) of an object, a set of parameters (characteristics of the object) and a set of constraints (equations or functions) applied to the object. By extension, a parametric modeler is a geometric conception system which not only preserves the explicit geometry of the designed object (so called *parametric object* or *current instance*), but also the sequence of commands used for generating it (so called *design process*, *constructive gestures* or *parametric specification*).

The majority of current feature-based parametric systems are known as *procedural* or *history-based* systems because parametric specification can be regarded as a composition of modeling functions, where each function is attached via its parameters to topological entities defined in some previous states of the model. Referenced entities must then be named in a persistent way in order to be able to reevaluate the model in a coherent way. Particularly, when reevaluation leads to topological

modifications, the references between entities used during the design process are difficult to match in the new context, giving results different from those expected.
This problem is known as "*topological naming*" when names use only topological information and "*persistent naming*" when names also involve other kind of information such as geometry of feature orientation [3] and [6]. The persistent naming mechanism should enable unambiguous identification of geometric and topological entities of the parametric model during the construction process, in order to retrieve them in the reevaluated model. Let us take the example shown in Fig. 1 to illustrate this issue.

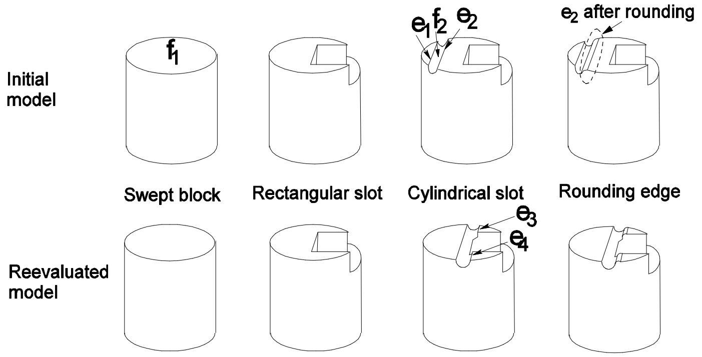


Fig. 1: Naming and matching problems.

In this example, the initial model is designed by means of a parametric specification containing four successive operations. The fourth one consists in 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 cannot 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 we look at 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.
Different works in this domain focus on the persistent naming of atomic entities (vertices, edges or faces) in planar cases. Others recent works (Wang [11], Biddarra [2] or Wu [12]) consider the same problem in the non-planar context which is the most difficult one; and the persistent naming of curves (non-planar instances of edges) gathers the most part of difficulties.
In the solutions proposed by [1, 8, 9], the topological entities are unambiguously characterized by an appropriate discriminating attribute. However, this characterization leads to pertinence and stability problems of the naming mechanism during the reevaluation process. The example presented in Fig. 2 illustrates the difficulty for those works to deal with this problem. In this example, the parametric model includes the Boolean difference operation. For this operation, different reevaluations of a same object generate parts. Each one of them results from the intersection between block *A* and surface *F*. Let us consider those parts 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 continue the reevaluation process after the Boolean operation, it is necessary to calculate the correspondence between the entities of these two parts: especially, edges resulting from the intersection of the top face of *A*, and *F*. The mapping calculus of those edges is geometry oriented. It uses the geometrical characteristics of

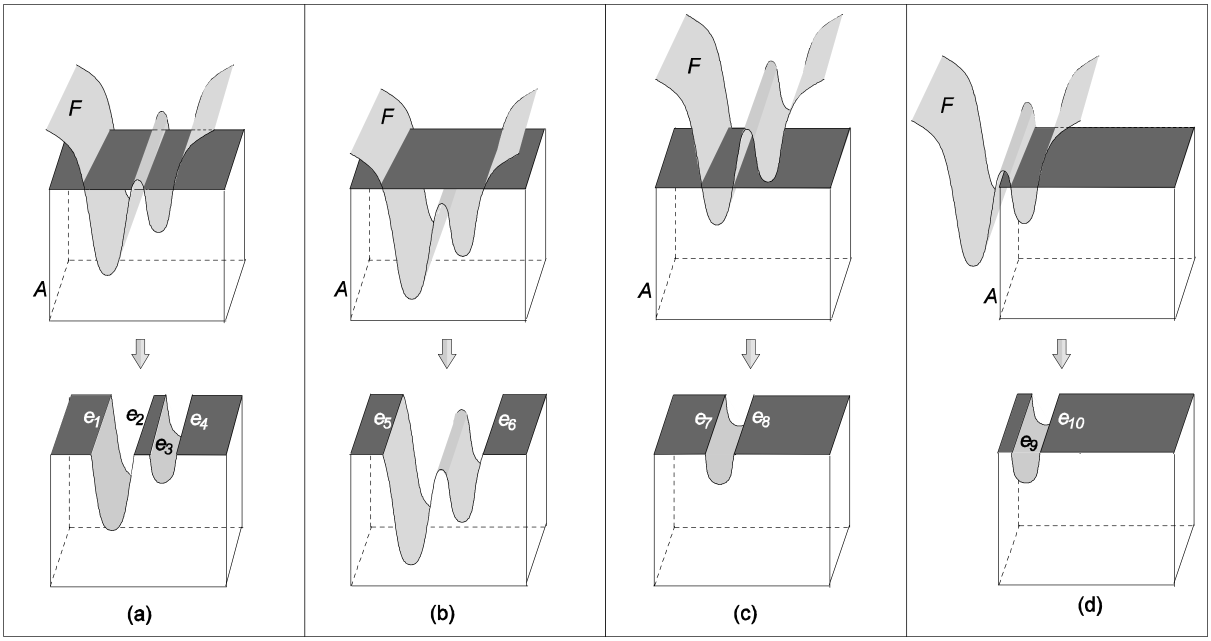


Fig. 2: Different parts resulting from the Boolean difference operation as parameter values change.

each edge (for example, the geometrical position on a parametric surface for Wu and the orientation for Wang) and it returns the results of Tab. 1 (see the 4^{th} column).

Those results, except for case 2, are different from those expected (not suitable with the user's design intents). The mappings to be found depend on two characteristic regions (hollows) of surface *F*. For example, in the first case, edges e_1 and e_4 should be mapped to edges e_5 and e_6 respectively, because e_1 and e_5 (resp. e_4 and e_6) result from the intersection of the left (resp. right) hollow of *F* with the top face of *A*.

In this paper, we intend to calculate some additional information about surfaces to improve the matching between topological entities. Our process is based on existing solutions like [8, 9] and it enables: a) to keep the uniqueness of references when two or more entities have the same topological neighborhood; b) to take into account the inherent semantics of different constructive gestures (representing the user's design intents). We consider that 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. Actually, a surface decomposition, according to the curvature criterion, is carried out. The naming solutions take into consideration the sub-surfaces resulting from the decomposition process. The replacement of the original surface with sub-surfaces leads to the matching techniques to give the right mappings.

Case	*Correspondance between*	*Expected mappings*	*Edges mapping*
1	part a and part b	$e_1 \Leftrightarrow e_5$ and $e_4 \Leftrightarrow e_6$	$e_1 \Leftrightarrow e_5$ and $e_2 \Leftrightarrow e_6$
2	part a and part c	$e_1 \Leftrightarrow e_7$ and $e_2 \Leftrightarrow e_8$	$e_1 \Leftrightarrow e_7$ and $e_2 \Leftrightarrow e_8$
3	Part a and part d	$e_3 \Leftrightarrow e_9$ and $e_4 \Leftrightarrow e_{10}$	$e_1 \Leftrightarrow e_9$ and $e_2 \Leftrightarrow e_{10}$
4	Part b and part c	$e_5 \Leftrightarrow e_7$	$e_5 \Leftrightarrow e_7$ and $e_6 \Leftrightarrow e_8$
5	Part b and part d	$e_6 \Leftrightarrow e_{10}$	$e_5 \Leftrightarrow e_9$ and $e_6 \Leftrightarrow e_{10}$
6	Part c and part d	no mapping	$e_7 \Leftrightarrow e_9$ and $e_8 \Leftrightarrow e_{10}$

Tab. 1: Computation of the correspondence between edges showed in Fig. 2. $edge_1 \Leftrightarrow edge_2$ means that $edge_1$ is mapped to $edge_2$

The paper is structured as follows. In Section 2, related research work is reviewed. We have chosen to complete Wu's approach [12] with our method, since his work is a good representative of related works. Therefore, mechanisms of Wu's method are given in Section 3, whereas we describe our work in Section 4. We give some concluding remarks and perspectives in Section 5.

2. RELATED WORK

Every persistent naming approach [8] addresses the same fundamental issues: characterizing and matching topological entities. In planar context, the topological entities are characterized by a single topological neighborhood. For that reason, the proposed solutions ([1], [4], …, [6]) are based on topology rather than on geometry. However, in non-planar cases, topology is not generally sufficient. Then, the literature methods use some additional geometrical information to remove ambiguity (for example, the characterization of the edges resulting from the intersection of two cylinders).

2.1 Characterization Based on Topology

Following the pioneer work of Hoffmann and Juan [5], several authors have proposed solutions for naming topological entities, in particular the solution of Kripac [6, 7].
Kripac presents a topological ID system which names a face based on a step id (identifying the particular creation face step during the modeling operations), a face index within that particular step, and the type of the corresponding surface. Edges and vertices are identified by the names of adjacent faces. Each model maintains a face modeling history during the construction. This history is used to map the new entities to the old ones if the topology of the model is changed. This matching process involves expensive graph isomorphism procedures in each model reevaluation and is not adapted to non-planar contexts.

2.2 Characterization Based on Geometry

A second approach consists in using geometry besides topology when entities with the same topological neighborhood are referenced. Recent works, which investigate in this direction, are proposed by Bidarra [2], Wang [11] and Wu [12].
For Bidarra and al., 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, [2] 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.
Wang and al. propose a general definition of a "parametric family" (Shapiro [10]), based on geometric continuity. 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, prefixing ids with feature namespaces transform the original flat namespace to an organized logical naming hierarchy. The ids identify themselves descriptively by the procedure of feature operations. Each of them may include additional geometry information (orientation and gradient) when multiple curves are formed by intersecting the same set of surfaces. The id structure described by Wang is not fundamentally different from the one proposed by Bidarra. Moreover, the use of higher order gradient to disambiguate edge designation can not reflect all geometry changes.
Wu proposes a naming mechanism which uses both topology and geometry to solve persistent naming when ambiguities appear. Wu's approach is quite representative of other related works. We therefore propose to complete his approach with our method. We describe Wu's method in the following section, and we show its shortcomings for some cases of edge matching process.

3. WU'S APPROACH

Wu and al. 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. If it is needed to reference an entity (face, edge or vertex) in the part body when creating some later features, they should be recorded with other names (called *RN* for Real Names), which are derived from original names. When regenerating the whole part, all the real names in the part model should be retrieved to prepare data for the reevaluation. *ON*s 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 sweep illustrated in Fig. 3, the part is made by subtracting one block from the other one 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. 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, 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, and $FeatID_p$ and $FeatID_{path}$ are the identifiers of the profile and the path, $ID_{element}$ and $ID_{trajectory}$ repectively identify one profile edge and 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:

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

When an entity is referenced, *ON*s are propagated along the descendant faces, so several faces can be assigned the same *ON*. Wu and al. propose to use a *RN* made of the initial *ON* and some additional information allowing to eliminate ambiguity in construction (called *PSI* for Parametric Space Information). Those purely geometric information are based on the parameter space (u, v) of the surfaces the faces relie 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 Figs. 2(b) and (c) and compute the parametric space of surfaces *F* and *G* (the top face of block *A* in Fig. 2). The result is shown in Fig. 4 and the coordinates of each point are depicted in Tab. 2. According to the principle of selecting feature points, as defined in [12], 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.

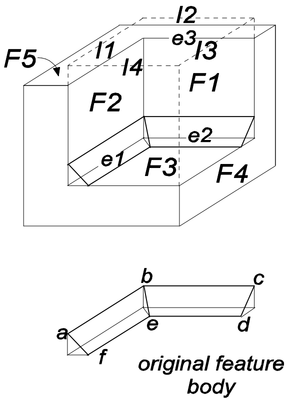


Fig. 3: Name chamfer feature.

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 Fig. 4, the part of Fig. 2(b) is reevaluated to the part of Fig. 2(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 Tab. 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 the top face of block *A* in the part of Fig. 2(b) whereas e_8 results from the intersection of the right hollow of the same face in the part of Fig. 2(c).

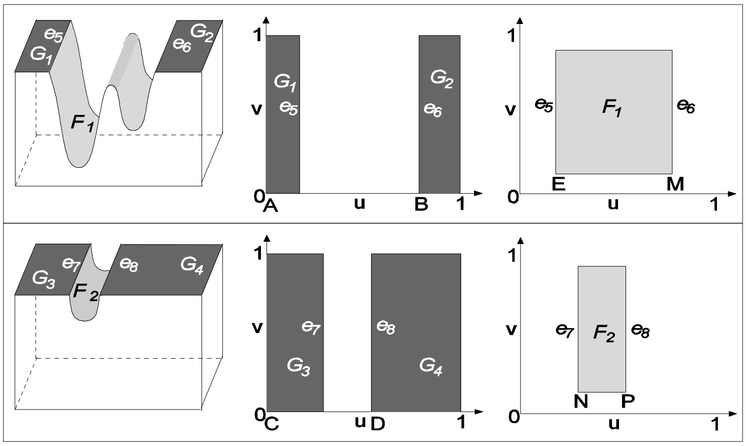


Fig. 4: Computing the parametric space information.

Following the method developed by Wu, the solutions proposed by Wang and Bidarra enable to characterize in an unique way every topological entity when ambiguities appear. Nevertheless, those methods do not always succeed in solving the matching issue. We propose, in the following section, a solution based on the shape characteristics of the modelized objects.

4. OUR APPROACH

In a non-planar context, 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. Our goal is not to define yet another naming method but to provide a solution which improves the results of such methods.

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 2, 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 discernable sub-surfaces by the shape characteristics.

Our method decomposes surfaces based on the curvature criterion and characterizes atomic entities (sub-surfaces, curves and vertices) as follows:

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

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

- each sub-surface is named following its *RN*, such as defined by Wu;

- curves and vertices are named following both the *ON* of the adjacent sub-surfaces and an additional information indicating the orientation of these entities. When one considers a curve c (resp. a vertex v) as the result of the intersection of two surfaces (resp. two curves): $f(x, y, z) = 0$ and $g(x, y, z) = 0$ (resp. $c_1(x, y, z) = 0$ and $c_2(x, y, z) = 0$), then the orientation of c (resp. v) at point $p = (x_p, y_p, z_p)$ is defined as: $\nabla c = \nabla f(x_p, y_p, z_p) \times \nabla g(x_p, y_p, z_p)$ (resp. $\nabla v = \nabla c_1(x_p, y_p, z_p) \times \nabla c_2(x_p, y_p, z_p)$) (where $\nabla = [\frac{\partial}{\partial x}\ \frac{\partial}{\partial y}\ \frac{\partial}{\partial z}]^T$ is the first gradient operator in Cartesian coordinates) [11]. In Fig. 4, plan $z = 0$ intersects with left hollow of surface F: $x^2+z+1 = 0$. The two intersecting curves e_7 and e_8, bounded by planes $y - 1 = 0$ and $y + 1 = 0$ have orientations $[0\ -2\ 0]^T$ and $[0\ 2\ 0]^T$, respectively.

The 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.

4.1 Calculation of Local Extremes

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: $f(x, y) = x * e^{(-x2-y2)}$ (see Fig. 5).

4.2 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 $f(x, y) = x * e^{(-x2-y2)}$ delimite twelve sub-surfaces $R_1, R_2, \ldots, R_{12}$, as shown in Fig. 6.

However, some generated sub-surfaces are without local extremums. In Fig. 6, 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 Fig. 5. Therefore, a sub-surface merging is necessary in order to obtain one sub-surface per local extremum.

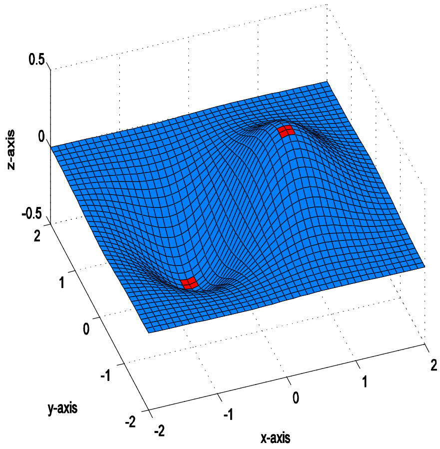


Fig. 5: Computing a local extremes (red points).

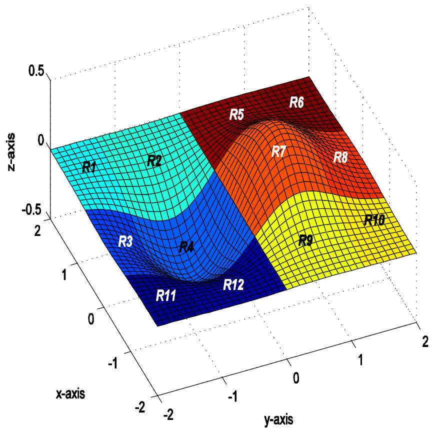


Fig. 6: Computing an inflexion curves.

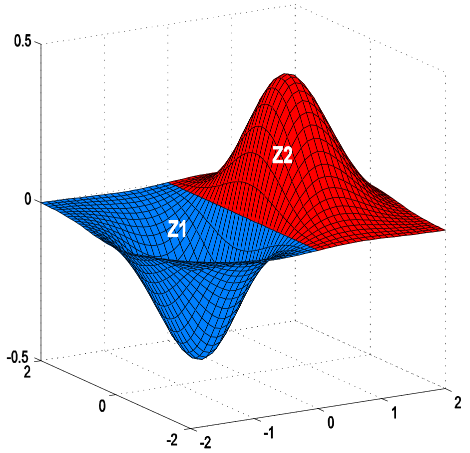


Fig. 7: Surface decomposition.

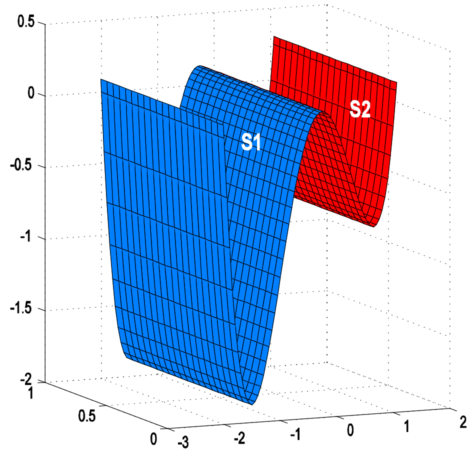


Fig. 8: Decomposition of surface *F* from Fig. 2.

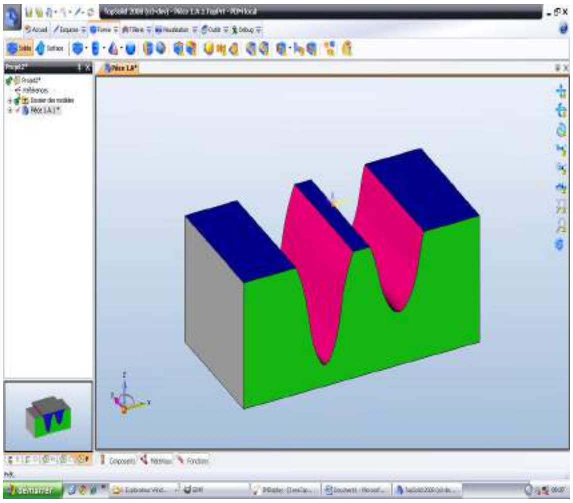
4.3 Merge of Sub-surfaces

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 Fig. 5, sub-surfaces R_1, R_2, R_3, R_{11} and R_{12} are merged with subsurface 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 Fig. 7).

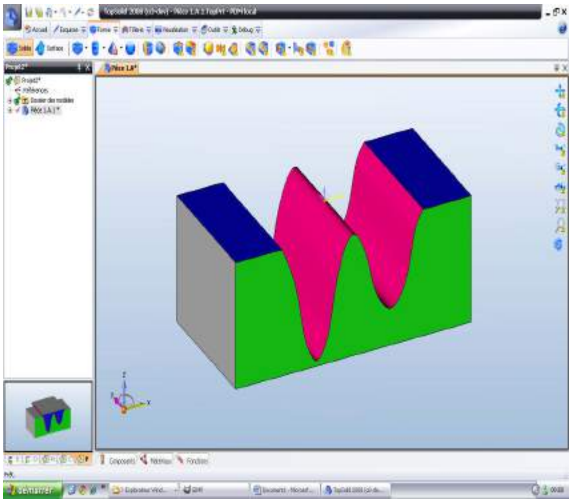
To improve Wu's and Wang's approaches, we use sub-surfaces resulting from the decomposition operation and the name structure defined in section 4. In Fig. 8, the decomposition of surface *F*, taken from Fig 2, gives two sub-surfaces (called *S*1 for the left hollow and *S*2 for the right one). By considering these sub-surfaces and the orientation information, the initial mapping results (column 4 in table 1) change. In fact, the comparison between edge's names of the Tab. 3 leads to the expected mappings depicted in Tab. 1 column 3.

Edge	*Edge's name*
e_1	[*ON*(S_1),*ON*(G),[*0 -2 0*]^T]
e_2	[*ON*(S_1),*ON*(G),[*0 2 0*]^T]
e_3	[*ON*(S_2),*ON*(G),[*0 -1 0*]^T]
e_4	[*ON*(S_2),*ON*(G),[*0 1 0*]^T]
e_5	[*ON*(S_1),*ON*(G),[*0 -2 0*]^T]
e_6	[*ON*(S_2),*ON*(G),[*0 1 0*]^T]
e_7	[*ON*(S_1),*ON*(G),[*0 -2 0*]^T]
e_8	[*ON*(S_1),*ON*(G),[*0 2 0*]^T]
e_9	[*ON*(S_2),*ON*(G),[*0-1 0*]^T]
e_{10}	[*ON*(S_2),*ON*(G),[*0 1 0*]^T]

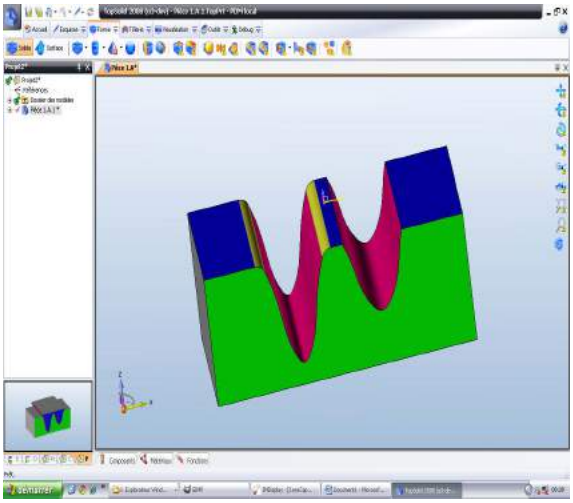
Tab. 3: Characterization of edges.



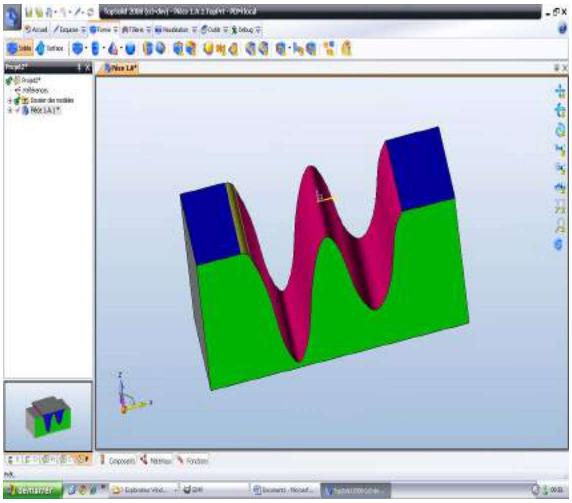
(a) The initial model corresponding to the part of Fig. 2(a)



(c) The reevaluated model corresponding to the part of Fig. 2(b)



(b) Rounding edges e_1 and e_2 of the part of Fig. 2(a)



(d) Reevaluation of the rounding operation

Fig. 9: Parametric process applied on parts of Figs. 2(a) and (b) by using TopSolid® Software with our matching approach.

Fig. 9 illustrates how the shape characteristics (sub-surfaces resulting from the decomposition operation) can improve the matching approaches of Wu and Wang. Figs. 9(a) and (c) show parts resulting from the difference Boolean operation between a block and surface *F* (see Fig. 2(a) and (b)). When the rounding operation is applied on edges e_1 and e_2 of the initial model (see Fig. 9(b)), our matching approach maps e_1 and e_4 on e_5 and e_6 (see case 1 in Tab. 1), so the rounding operation is applied on edge e_5 only in the part of Fig. 9(c) (see the result in Fig. 9(d)).

5. CONCLUSION AND FUTURE WORKS

In this paper, we have proposed some additional information about surfaces to improve the matching between topological entities. 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 criterion is implemented. This operation uses local extremums and inflexion curves in order to delimite hollow(s) and/or hump(s). In the end, the decomposition operation generates one sub-surface per local extremum. The existing matching processes replace each surface with its corresponding sub-surfaces, leading to the right edge mappings.
The perspectives of this work are: firstly, to test the robustness of our method on several examples taken from domain as CAD, architecture and geology; then, to integrate our method into the hierarchical persistent naming system we are currently developing.

6. REFERENCES

[1] Agbodan, D.; Marcheix, D.; Pierra, G.; Thabaud, C.: A topological entity matching technique for geometric parametric models. In SMI '03: Proceedings of the Shape Modeling International 2003, page 235, Washington, DC, USA, 2003, IEEE Computer Society.

[2] Bidarra, R.; Nyirenda, P.-J.; Bronsvoort, W.-F.: A feature-based solution to the persistent naming problem, Computer Aided Design & Applications, 2(1-4), 2005, 517–526.

[3] Capoyleas, V.; Chen, X.; Hoffmann, C.-M.: Generic naming in generative, constraint-based design, Computer-Aided Design, 1996, 28(1), 17–26.

[4] Chen, X.: Representation, Evaluation and Editing of Feature-Based and Constraint-Based design, Ph.D. thesis, Department of Computer Sciences, Purdue University, West Lafayette, Indiana, 1995.

[5] Hoffmann, C.-M.; Juan, R.: EREP: an editable high-level representation for geometric design and analysis. Technical Report CER-92-24, Department of Computer Sciences, Purdue University, West Lafayette, Indiana, 1993.

[6] Kripac, J.: A mechanism for persistently naming topological entities in history-based parametric solid models. In SMA '95: Proceedings of the third ACM symposium on Solid modeling and applications, New York, NY, USA, 1995, 21-30.

[7] Kripac, J.: Topological ID system: A mechanism for persistently naming topological entities in history-based parametric solid models. Ph.D. Thesis, University of Czech Technical, Prague, 1993.

[8] Marcheix, D.; Pierra, G.: A survey of the persistent naming problem, Proceedings of the seventh ACM symposium on Solid modeling and applications, 2002, 13–22.

[9] Shah, J.-J.; Mäntylä, M.: Parametric and feature-based CAD/CAM: Concepts, techniques, applications, 1995.

[10] Shapiro, V.; Vossler, D.-L.: What is a parametric family of solids, Proceedings of the third ACM symposium on solid modeling and applications, May 17–19. Utah: Salt Lake City; 1995. p. 43–54.

[11] Wang, Y.; Nnaji, B.-0.: Geometry-based semantic id for persistent and interoperable reference in feature-based parametric modeling, Computer Aided design, 37, 2005.

[12] Wu, J.; Zhang, T.; Zhang, X.; Zhou, J.: A face based mechanism for naming, recording and retrieving topological entities, Computer Aided Design, 33(10), 687–698, 2001.