7[e] Conférence Internationale de MOdélisation et SIMulation - MOSIM'08 - du 31 mars au 2 avril 2008 – Paris- France
« Modélisation, Optimisation et Simulation des Systèmes : Communication, Coopération et Coordination.».

CLASSIFYING CONSTRAINTS FOR CURVE AND SURFACE MODELLING

Vincent CHEUTET

LISMMA Supméca Paris
3 rue Fernand Hainaut
93407 saint Ouen cedex
vincent.cheutet@supmeca.fr

RÉSUMÉ : *Computer-Aided Geometric Design modellers are now based on powerful mathematical curve and surface models, but there is still a considerable need for efficient tools to handle, analyze and modify these objects. Designing product shapes using geometric operations on free-form curves and surfaces is still a tedious task. Moreover, designers would prefer to use meaningful tools to concentrate on design objectives expressed in terms of functionalities and constraints related to engineering topics and technical matters.*
This explains why constraint modelling in CAGD is an important challenge for the forthcoming years and the paper aims at suggesting a classification of the different constraints which ought to be taken into account in a constraint based modelling CAGD software.

MOTS-CLES : *constraint, shape modelling, classification*

1 INTRODUCTION

In our current society of consumption, aesthetics plays a crucial role in the definition of a product and so of its target market. Even if two products satisfy the same basic functionalities and have similar performances, the shape of the product often makes the difference to the eyes of the customers. As a consequence, shape has increased in complexity to open new emotional fields. Free-form curve and surface models have significantly increased the quality of the designed objects assuming the ability of the user to handle the different parameters, generally defined in terms of geometry. Improving the ergonomics and efficiency of these design handles in order to manipulate, analyze and modify objects is still a research issue. Moreover, when a designer uses a CAGD (Computer-Aided Geometric Design) software, his/her expectations would be to express and solve the problem as a relevant problem of design. The latter problem to solve is given in terms of functionalities and various types of constraints including geometry, engineering topics, mechanics, manufacturing, know-how, industrial context, economy, ... and is sometimes badly structured, especially at the beginning of the process. This explains why requirements for curve and surface modelling techniques are currently changing and more and more oriented toward a constraint based modelling approach. This new approach will directly lead first to express constraints and then to solve a high number of various constraints. It thus raises a lot of difficult issues while it corresponds to an actual need in industry.

But the industrial requirements for manufactured objects are complex. It may be impossible to model a real shape with a unique surface so that the object must be defined by a set of surfaces connected together through continuity constraints (tangency or curvature). Moreover, even for one surface, the mathematical models currently applied (B-spline or NURBS) cannot always represent the reality and/or the topology of the shape. As a result, the user must define a larger surface and then define a restriction curve on the surface which surrounds its relevant part. The curve, given in the parametric definition plane is called trimmed curve and the relevant surface is called trimmed surface. This approximate operation raises well-known inaccuracies. We point out this particular problem because the different connected surfaces defining an object are most of the time trimmed surfaces.

Figure 1 proposes a real example of a digital model of a car door defined with several trimmed B-spline surfaces. One of the patches is described (with and without its control network) with a hole inside, defined by trimming curves. The previous example illustrates that constraint modelling in CAGD is an important challenge for the forthcoming years.

Details on global approaches developed in CAD (Computer-Aided Design) modellers (parametric modelling, variational modelling, feature-based modelling, declarative modelling) and existing constraint-based modelling techniques for the most frequent types of curves and surfaces (parametric, multiresolution, implicit, subdivision…) can be found in (Cheutet et al., 2008). From this survey, this paper proposes a classification of the different constraints which ought to be taken into account in a constraint-based modelling CAGD software.

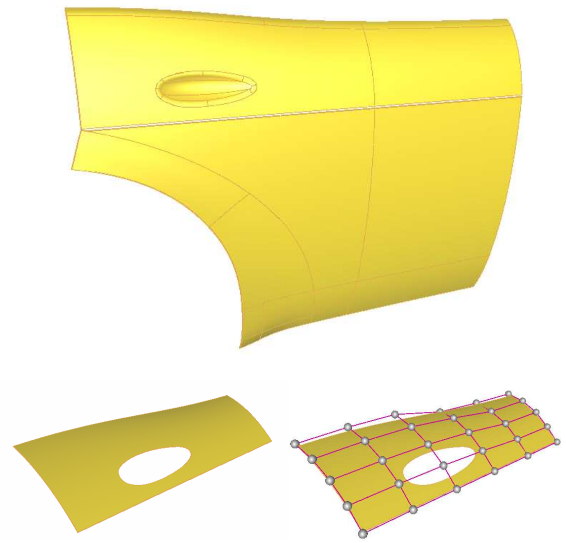

Figure 1. An example of a digital model of a car door, with the definition of one patch according to its control network (courtesy of s.t.s.).

2 CONSTRAINT EXPRESSION

One of the main difficulties in shape generation and modification is the expression of the user's intents and their translation into a set of constraints compatible with the chosen geometric model and the software modelling environment. Three aspects increase the complexity for expressing constraints:

- The first one is to take into account the user's background or the user's skills: he/she can be either a stylist, a designer, a manufacturer, etc. Each of these users has a different point of view on the product and its shape. The users express different types of constraints about product modifications during the product development process. These constraints contribute to a so-called product view defined from the appropriate information: product shape, mechanical, technological...

- At the same time, the expression of constraints depends of the type of software application used. Their expression can differ when using a CAD modeller or a Virtual Environment with haptic devices. In the first case, the constraints will be directly attached to the model geometry. In the second one, some of the constraints could be expressed in terms of forces to fit capabilities of haptic devices.

- Expressing the constraints is related to the underlying geometric model of a component: a constraint will be differently written if the geometric model is a parametric surface or a surface mesh. Moreover, the difficulty in specifying constraints is increasing when the geometric model type is not uniform over the whole product. Some areas can for instance be described by NURBS surfaces and others by surface meshes. It can also happen that the geometric model of the constraining entity is different from the one of the constrained entity, like a surface mesh constrained by a B-spline curve. These configurations correspond to hybrid models, and are a priori unusual in a CAD modeller. But considering an industrial product through its set of components with all their possible geometric representations can lead to such cases.

From a complementary point of view, the constraints may be related to the product itself rather than to its geometry. It adds the difficulty of formulating constraints that have a meaning at the product level, while incorporating some geometric parameters (e.g. as available in the CAD modellers). Here are two examples of these high level constraints:

- For a manufacturer, one component of the product can be designed to be moulded, and in this case, this component must be extractable from its mould. This high-level constraint (at the product level) can be decomposed into low-level constraints where some of them can be attached to the component geometry by constraining the draft angle of some areas on the surface with respect to the parting plane of the component.

- For a designer, a component of the product should not break during product use. However, this constraint only incorporates mechanical quantities like the maximum stress into the component. Since it needs the strength of the component material and the boundary conditions on the component, this constraint cannot be directly decomposed into lower level constraints referring to the component geometry. Only a Finite Element Analysis (FEA) or similar mechanical analysis methods can provide the proper parameters for this constraint. The FEA cannot be easily incorporated inside a CAD modeller and the extraction of constraints attached to the component shape becomes even more complex.

As described, the notion of constraints during a design phase can be very large. This notion is commonly used at all of its successive steps, even if some users don't bear in mind the same meaning for these design constraints. We first propose in the following a taxonomy of the constraints classically used for curve and surface modelling and secondly, the various ways a user can express these constraints. Four semantic levels exist for these constraints in the context of shape generation and modification, according to the type of the constrained entity:

- *Semantic level 1*: constraints attached to one element of the component shape: this includes some local constraints used to manipulate its shape like the point constraints.

- *Semantic level 2*: constraints expressed between two or more geometric elements of the component shape: for instance, to preserve the integrity of the geometric model of the component during the modification of its shape, like the continuity of its shape.

- *Semantic level 3*: constraints attached to the whole component shape like a volume constraint, for example.

- *Semantic level 4*: constraints not only attached to the component shape but linking its geometry with other parameters of the component, i.e. parameters not directly expressing geometric constraints like functional or engineering ones as illustrated above.

The user first specifies a constraint of any given category. In order to handle it in a second time, a geometric modeller can need to decompose this constraint into a set of constraints of complementary categories (see an example in section 3.2). But this decomposition cannot be always achieved and the modeller must be able to handle constraints of different categories at the same time. Moreover, the decomposition is not unique and each modeller has its own way to perform the decomposition.

Then, according to the different approaches described in the introduction, two ways of specifying constraints can be proposed that cover all the semantic levels described previously:

- *Strict constraints*: which are named classically **constraints** in the literature. The modeller has to strictly respect them during the shape creation and manipulation processes. As an example, the current sketchers in CAD modellers are only using this type of constraints.

- *Soft constraints*: these constraints are used in the declarative modelling approach to allow the description of the object properties, but also to describe free-form surfaces in the other approaches. They can express the final aspect of a component shape or at least, the expectation to obtain a solution close to it.

The above analysis highlights the complexity of expressing the constraints attached to a product. In the next subsections, the constraints classically used for curve and surface modelling are described in more details, according to the categories previously defined in this section.

3 STRICT GEOMETRIC CONSTRAINTS

This section describes the **strict constraints** commonly used in shape modelling and part of the first two categories previously listed (constraint attached to one geometric element or between two or more geometric elements of a component). Because they are directly related to geometric parameters, the literature named them as ***geometric constraints***. Structuring them requires three complementary concepts:

- *The constrained entity*: either a curve or a surface.

- *The constraining entity*: either a point, a curve or a surface.

- *Global or local effects*, according to the size of the area on which they have a direct influence. **Global** designates configurations where the entire entity considered is modified. **Local** means that only an arbitrary subset of the entity is subjected to shape changes.

Using these three concepts, Tables 1 and 2 can be considered to classify these constraints.

		constrained entity	
		curve	surface
constraining entity	point	point constraint	point constraint
	curve	shape matching, symmetry	curve constraint (character line...)
	surface	projection	shape matching, symmetry, connection...

Table 1. Constraints with a geometric entity as reference (direct constraints)

	constrained entity	
	curve	surface
local	tangent, curvature, ...	tangent plane, principal curvatures ..
global	length, area,...	area, volume, inertia,...

Table 2. Constraints without a geometric entity as reference (indirect constraints).

When the constrained entities and the constraining ones have a manifold dimension superior to 0 (i.e. are a curve or a surface), these constraints can require an approximation process and hence are generally decomposed into a set of point constraints. This situation often arises when specifying continuity constraints across several geometric elements of a component. Actually, the continuity conditions across trimmed patches cannot be correctly expressed using the real geometric entities describing the corresponding surface or volume (Farin, 1996).

As a consequence, some of the constraints can be structured in a hierarchical manner: an upper level constraint is translated into a set of lower level constraint elements and so forth until receiving a set of point constraints if this elementary type of constraints is the only common denominator to express the desired constraint.

3.1 Local constraints

Local geometric constraints are used to locally control a shape, i.e. over a subset of the geometric elements describing the overall shape. The control is achieved through a set of point constraints like position constraints, tangent and normal constraints, curvature constraints, etc. These constraints correspond to the first line of Table 1, i.e. where the constraining entity is a point. But they also correspond to the second line of Table 2, i.e. with a local effect on the shape.

These constraints provide an ideal tool for direct shape manipulation. The designer picks a point on a curve or a surface and moves it. It remains to enforce the curve/surface to pass through a new user-specified location using the properties of the underlying geometric model of curve/surface (see figure 2). Repeating this operation with small displacement steps leads to a curve/surface select-and-drag tool which continuously deforms the curve/surface. Virtual sculpting techniques make use of this type of constraints.

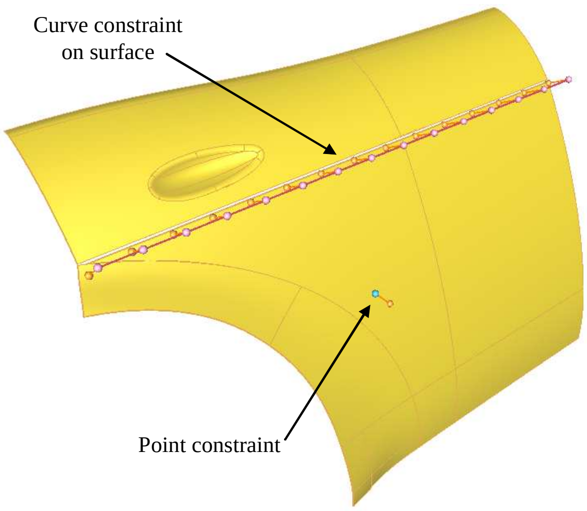

Figure 2. Point and line constraints on the car door model.

These constraints are also the most basic constraints, when the decomposition of constraints is required as stated above.

Let **P** be a point of the curve/surface, $\mathbf{P_u}$ (resp. $\mathbf{P_v}$) be the first derivative vector of the curve/surface in the *u* (resp. *v*) direction at the point considered. Each local constraint can be written as follows:

- *Position*: $\overrightarrow{PM}=\vec{0}$, where **M** can be either a 3D point prescribed by the user or a point of another curve/surface.
- *Tangent*: $\overrightarrow{P_u}.\overrightarrow{D}=0$, $\overrightarrow{P_v}.\overrightarrow{D}=0$ (the last one only exists in case of a surface), where **D** can be either a vector prescribed by the user or the normal vector of another curve/surface.

According to the type of curve/surface considered, the derivative $\mathbf{P_u}$ can be different if the type of curve/surface is no longer parametric but implicit, subdivision, etc. It is also important to note that the position constraint is defined by one vector constraint and so corresponds to three scalar constraints, and the tangent constraint is defined by two scalar constraints.

Position, tangent and normal directions are linear constraints and can efficiently be solved with appropriate methods depending on the underlying geometric model:

- NURBS: methods enumerate as follows: specification of an area of influence, least-squares fitting with linear constraints (SVD, QR factorization (Golub and Van Loan, 1996)).
- FFD: solid model forming the control lattice referred to in the previous section.
- Particular surface models that give direct access to these quantities: interpolating spline curves (position, tangent, curvature), triangular interpolating surfaces (position, tangent, normal), (Loop, 1994), (Peters, 1995), (Piper, 1987), (Hahmann and Bonneau, 2000), (Hahmann and Bonneau, 2003), (Yvart et al., 2005).

3.2 Curve constraints on a surface

These constraints enforce the surface to match a given curve, in a context of shape modification (see Table 1). One of the current applications is for car aesthetic design: the product is described by a set of curves at some stage of the product specification and this set can be seen as the framework of the product shape model. These curves correspond not only to the object profiles, symmetry lines and selected sections but also to significant lines strongly affecting the product shape, the character lines. Thus, the surface model of the product is directed by these curves (Cheutet et al., 2005a). In such a context, the stylists are manipulating a product shape, i.e. its surfaces, through the prescription of curves meaningful for them. Therefore the surfaces have to be constrained by these reference curves.

In the case of continuous surfaces like NURBS surfaces, these curve constraints are, most of the time, decomposed into a set of point constraints, with additional parameters related to the application domain (see figure 2). This discretization process has a strong influence on the resulting shape. This points out a very complex problem: the curve discretization must not only be too coarse to lose some meaningful variations of the initial curve but must also take into account the distribution of the degrees of freedom over the initial surface to avoid as a result over-constrained configurations. For the specific case of NURBS surfaces, the decomposition is not necessary when the constraining curve can be directly matched with an iso-parametric curve of the surface (Michalik and Bruderlin, 2004). However, this case is a very particular one, and is not often encountered in industrial configurations.

When the underlying surface is defined as a mesh, the constraint decomposition process is mandatory since the surface model is a discrete geometric model.

If a curve is constrained by a surface, with a projection constraint for instance, the defined constraint is exactly the same as the curve constraint described at the beginning of this section. The surface is then the constraining element and therefore is not modified during the solving process.

3.3 Entity constraints on entities of the same dimension

This section addresses either constraints used to express intrinsic relationships between geometric elements of a product (from the second semantic level, as described in section 2), i.e. continuity conditions between curves/patches, or constraints related to shape matching configurations, from the first semantic level (see Table 1). Shape matching constraints refer to configurations where a geometric entity, e.g. a curve, has to match the shape of a "reference" geometric entity, e.g. an arc of circle. In the case of continuity conditions, as in figure 1, the shape can be defined by a set of B-spline patches and the continuity conditions in position, tangency and/or curvature between them have to be taken into account to preserve the model integrity during the shape transformation.

Considering the geometric continuity conditions, if the constrained geometric elements are trimmed patches, which are the most common case, the constraints need to be decomposed once more. Firstly, a surface continuity constraint is expressed as a curve constraint for each boundary curve of a surface patch. Secondly, each curve constraint is decomposed into a set of point constraints. When performing such decomposition, only an approximate continuity condition is achieved between the corresponding elements. An exact geometric continuity can be obtained if and only if the common boundaries between the two patches are not trimming lines, and if the two lines, which are iso-parametric curves, have the same degree and the same number of control points (Farin, 1996). The latter requirements can be easily achieved with classical knot insertion of degree elevation algorithms, once a common degree and knot vector is constructed for the surfaces considered. This is obtained at the expense of increasing the number of parameters of the surfaces and/or considering the previous algorithms as a complementary set of constraints.

In shape matching configurations, a curve/surface is constrained by a "reference" geometric element of the same dimension, with for instance:

- *Curve constraints on curve*: a curve is constrained to become a circle or a straight line...

- *Surface constraints on surface*: a surface is constrained to become planar, or part of a cylinder or a sphere...

In the previous examples, the predefined geometric element is a primitive one or part of a primitive one, i.e. segment, circle, plane, cylinder, sphere. If the predefined geometric element is totally defined, e.g. in the case of a primitive element, all their parameters and position are defined. The decomposition into a set of point constraints is the generic way to express these constraints.

But sometimes, and particularly when the predefined geometric element is a primitive one not entirely constrained, the decomposition process is not the only solution. For instance, a subset of the bump surface of the streamliner (see figure 2) has to be planar but the position of the plane is not constrained in any way: using point constraints is not relevant since it prescribes the position of the plane. A new set of constraints has been proposed by (Cheutet et al., 2005b) to preserve or insert the desired shape constraints during a deformation process. As an example, in case of planarity constraints on a NURBS surface, a corresponding set of constraints is expressed by the equation:

$$\overrightarrow{\mathbf{n_0}}.\overrightarrow{\mathbf{P_0P}} = \mathbf{0}$$

where $\mathbf{n_0}$ is the normal vector of the plane, $\mathbf{P_0}$ is a point of the plane and $\mathbf{P}$ can be either a surface point or directly a surface control point. Using such constraints will not constrain the position and the orientation of the plane.

Apart from such specific configurations, the generic way to express these constraints is still the decomposition into a set of point constraints, as for curve constraints on surfaces.

3.4 Global constraints

This section describes constraints that act on the whole curve/surface (constraints from semantic level 3). They cannot be decomposed into a set of point constraints like the other examples because they refer to some integral properties of the associated curve/surface.

In 2D space, curves can be constrained to preserve either a prescribed area or a constant length or to preserve some symmetry with a predefined axis during the deformation process (Elber, 2001), (Sauvage et al., 2004), (Hahmann et al., 2005). In particular, (Elber, 2001) has worked on the preservation of the internal area of a B-spline curve during the deformation process and has demonstrated that this constraint is linear with respect to the coordinates of the curve control points. In fact, if one considers a regular closed planar polynomial parametric curve **C**(t) = (x(t), y(t)), the enclosed (signed) area **A** equals (using Green's theorem):

$$A = \oint \left[x(t)\dot{y}(t) - \dot{x}(t)y(t) \right] dt$$

$$A = \oint \left| C(t) \times \dot{C}(t) \right| dt$$

This equation is clearly bilinear in the coordinates of the control points. Other constraints, like computing the moments, exist and have been studied by (Gonzalez-Ochoa et al., 1998) and (Elber, 2000).

In 3D space, volume preservation is important for achieving realistic deformations of solid objects in computer graphics (Lasseter, 1987). Different techniques for volume preservation exist, according to the type of the underlying geometric model:

- B-spline surfaces (Elber, 2001),
- trivariate Bézier solids: least squares-energy minimization coupled with volume constraint (Rappoport et al., 1995),
- B-rep solids: global volume (Hirota et al., 1999),
- implicit surfaces: discretized volume approximations (Desbrun and Gascuel, 1995).

4 SOFT CONSTRAINTS

These constraints can be of various types which can be difficult to express in a mathematical form. For example one can expect a rather flat surface, a bumbled one or an elliptic one. Introducing these constraints into a constraint based modeller remains an open challenge.

This section also describes constraints that are related to a functional to minimize. These constraints do not have to preserve a strict value during the deformation process. They rather indicate a user's tendency for the desired curve/surface behaviour after deformation. As an example, a global constraint on volume preservation imposes a strict value for the volume that the final shape has to satisfy, whereas inserting a soft constraint on volume tends to minimize the volume variation between the initial shape and the resulting one, and so the process accepts a variation of volume.

Free-form surfaces of components used in CAD modellers are usually based on a set of trimmed patches. This implies that the number of parameters involved in constraint-driven modifications, i.e. the number of control polyhedron vertices, is generally far greater than the number of user's constraints, at least globally. Therefore, the corresponding problem is often under-constrained, hence this type of constraint is used as a criterion for a deformation engine to choose one solution among all those verifying the strict constraints. Classically, the deformation engine is based on one soft criterion related to the shape fairness, in order to obtain the smoothest and the most graceful shapes. The criteria often correspond to the minimization of an energy having a physical meaning and lead to natural surfaces. Different interpretations exist and have been implemented for each geometric model:

- Non-linear functionals that derive from elasticity theory:

$$\int \kappa^2(t)\,dt \text{ or } \int \left(\kappa_1^2 + \kappa_2^2\right) dA$$

- Linearized versions of these energy functionals, in order to accelerate and simplify the computations:

$$\varepsilon = \int_\sigma \left(\alpha\,\text{strech} + \beta\,\text{bend}\right) d\sigma$$

One minimization criterion is chosen for a given user objective, but (Pernot et al., 2004) have demonstrated that soft constraints can also be used to monitor the shape deformation (see figure 3). In this case, soft constraints can be arranged in various ways and the user can choose one shape among a continuous set of solutions, using a single control parameter. For this purpose, the user initially chooses two predefined behaviours of the shape, i.e. two predefined criteria, and a set of solutions is generated as a linear combination of the solutions obtained for each sole criterion. To further increase the range of solutions, different criteria over a set of connected sub-domains covering the surface deformation area can be also defined.

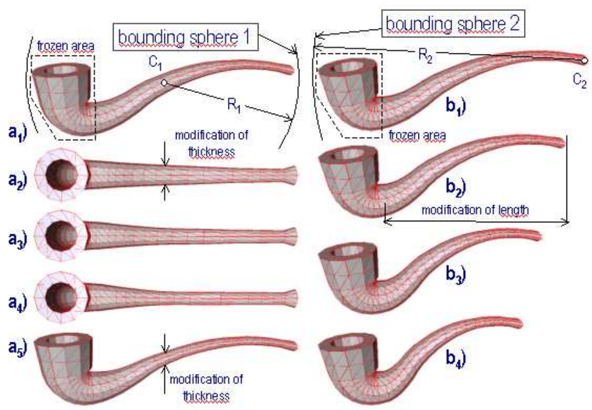

Figure 3. Global shape modifications using solely the multi-minimizations and their predictive behaviours (from Pernot et al. 04).

Two examples are depicted on figure 3. No geometric constraints are specified and the various shapes are obtained when the parameters of the multi-minimizations vary. The pipe is immersed inside a bounding sphere centred at a user-specified point $\mathbf{C_i}$, i.e. $\mathbf{C_1}$ for the example (a) and $\mathbf{C_2}$ for the example (b), and used to define locally the basic quantities to be minimized (Pernot et al., 2004). More precisely, in the proposed examples, the more the control vertices of the geometry are far from the centre of the sphere, the more the initial shape defined by these vertices is preserved. The relative influence between these two types of quantities is controlled by a single parameter which enables the generation of a wide variety of shapes (figures a_1 to a_5 and figures b_1 to b_4). The sphere of the example (a) has been centred in the middle of the pipe which enables a modification of the thickness of the pipe. If the bounding sphere is moved at the extremity $\mathbf{C_2}$ of the pipe (example (b)), a modification of the length of the pipe is obtained.

5 ENGINEERING CONSTRAINTS

This section deals with constraints of the fourth semantic level. In this category, we could find all constraints related to the mechanical behaviour of a product, such as changing the shape of a component in some areas while maintaining the maximum stress value in a given area (see for example the works of Allaire et al., 2005}). The considered constraints are characterized by the fact that they simultaneously incorporate in their expression geometric parameters as well as mechanical or technological parameters that can be needed at a given stage of the product development process.

As mentioned before, these constraints cannot usually be decomposed into constraints of the other semantic levels since other quantities than the geometric ones are involved in these constraints. They are decoupled from the solving process set up for geometric constraints. Moreover, the results of the constraints evaluation require the result of a specific algorithm, like a Finite Element Analysis: these constraints are seen by the designer as black boxes. The results obtained through these "black boxes" are then incorporated into a geometric constraint solving process. Then, the user can analyze a solution or obtain some clues about which parameters have to be modified.

6 SOLVING A SET OF CONSTRAINTS

Once all the constraints have been expressed, the solving process can begin. The objective in this section is not to address and analyze the methods for solving a set of constraints attached to a free-form curve/surface but to characterize some of the configurations often faced when manipulating such free-form objects. The current issue according to the specification of the constraints is: what are the right number and the right position of the constraints, according to the number and the location of the degrees of freedom (DOF) of the underlying geometry? Globally, different configurations can appear:

- *Under-constrained problem*: there are fewer constraints than DOF.
- *Iso-constrained problem*: there is the same number of constraints and DOF.
- *Over-constrained problem*, with three sub-cases:
    - consistent: the added constraints are coherent with the other ones,
    - semi- consistent: it is possible to transform the problem into an iso-constrained one by adding DOF to the geometric model,
    - fully consistent when constraints express contradictory requirements and lead to no satisfactory solution.

As an example, to define a plane, four coplanar points create a consistent over-constrained problem but four non-coplanar points generate a fully consistent over-constrained one. If the initial geometry is a patch only defined by four non coplanar control points, a planar area can be defined over the initial surface by adding DOF in the geometric model to produce the desired freedom required to insert the planar area.

The configuration of a set of constraints can however be even more complex: a problem can be globally under-constrained and locally over-constrained, semi-consistent (etc.), i.e. for some subset of constraints characterizing a specific geometric configuration. Tools to detect these configurations exist but only with a restricted set of geometric constraints applied on primitive elements like points, lines, planes, etc. In the case of free-form curves/surfaces, no such tool is currently available.

Most of the time and especially in styling activities, the initial problem is globally under-constrained (see section 4) and a surface deformation is performed to obtain one solution. The solution of the deformation process can be compared to adding a soft constraint to the initial under-constrained problem. This is a way to select one solution among all possible ones. This problem structure reflects also the fact that some designer's constraints can hardly be expressed as geometric constraints, like the surface fairness. Hence, soft constraints can be used to let the designer adjust the shape in accordance to complementary parameters that cannot be incorporated into geometric constraints.

But applications can expect more than one solution as the declarative modelling approach. The user can ask for a structuring of all the solutions and for tools to browse this structure. This structuring phase can be even more interesting when a large set of engineering constraints is applied on the product but is decoupled from the geometry generation and manipulation phases. At this moment, few people are working on this subject.

7 CONCLUSION

The paper focus on the constraints, either geometric or not, attached to free-form curves and surfaces. These constraints are the key elements to define and manipulate a product shape, when defining an environment.

The proposed presentation structures and performs a first synthesis about the existing constraints that can applied, according to the level of semantic attached to the constraint, and to the type of manipulation. This proposed synthesis does not describe how the constraints are implemented according to the underlying geometric representation used to describe the shape in the concerned environment, but several references have been added to find further details on these topics.

One main issue is the analysis of a set of constraints applied on free-form shapes. If currently several algorithms exist to solve these constraints according to the type of geometric representations, there is a lack on establishing rules to detect over-constrained configurations or hierarchies between constraints, according to the proposed classification.

ACKNOWLEDGMENT

The author would like to thank Marc Daniel, Stefanie Hahmann, Raphael La Greca, Jean-Claude Léon, Robert Maculet, David Ménegaux and Basile Sauvage for their contribution in this work.

The author wants to thank the CNRS (Centre National de la Recherche Scientifique) to have provided support through the "Action Spécifique Modélisation Géométrique sous Contraintes" and also through the MathSTIC Project "Solveurs Géométriques".

REFERENCES

Allaire, G., De Gournay, F., Jouve, F., Toader, A.-M., 2005. Structural optimization using topological and shape sensitivity via a level set method. *Control and Cybernetics*, Vol. 34, pp. 59-80.

Cheutet V., Catalano C.E., Pernot J.P., Falcidieno B., Giannini F., Léon J.C., 2004. 3D Sketching with Fully Free Form Deformation Features for Aesthetic Design. *Eurographics Symposium Proceedings on Sketch-Based Interfaces and Modeling,* Grenoble, p. 9-18.

Cheutet V., Daniel M., Hahmann S., La Greca R., Léon J.C., Maculet R., Ménegaux D., Sauvage B., 2008. Constraint Modeling for Curves and Surfaces in CAGD: A Survey, *accepted to International Journal of Shape Modelling*.

Cheutet V., Pernot J.P., Léon J.C., Falcidieno B., Giannini F., 2005. Insertion of Planar Areas into Free Form Surfaces in Early Product Design. *ASME DETC-DAC*.

Desbrun M., Gascuel M.P., 1995. Animating Soft Substances with Implicit Surfaces. *SIGGRAPH '95 Conference Proceedings*, p. 287-290.

Elber G., 2000. Linearizing the area and volume constraints. *Technical Report CIS-2000-04*.

Elber G., 2001. Multiresolution curve editing with linear constraints. *6th ACM/IEEE Symposium on Solid Modeling and Applications*, A. Arbor, Michigan, p. 109-119.

Farin G., 1996. Curves and Surfaces for Computer Aided Geometric Design. *4th ed. New York Academic Press*.

Golub G.H., Van Loan C.F., 1996. Matrix Computation. *The John Hopkins University Press*, 3rd ed. Baltimore and London.

Gonzalez-Ochoa C., Mccammon S., Peters J., 1998. Computing Moments of Objects Enclosed by Piecewise Polynomial Surfaces. *ACM Transaction on Graphics*, vol. 17(3), p. 143-157.

Hahmann S., Bonneau G.P., 2000. Triangular G^1 interpolation by 4-splitting domain triangles. *Computer Aided Geometric Design*, vol. 17, p. 731-757.

Hahmann S., Bonneau G.P., 2003. Polynomial Surfaces Interpolating Arbitrary Triangulations. *IEEE Transactions on Visualization and Computer Graphics*, vol. 9(1), p. 99-109.

Hahmann S., Sauvage B., Bonneau G.P., 2005. Area Preserving Deformation of Multiresolution Curves. *Computer Aided Geometric Design*, vol. 4(22), p. 349-367.

Hirota G., Maheshwari R., Lin M.C., 1999. Fast Volume-preserving Free-form Deformation Using Multi-level Optimization. *Proceedings of Solid Modeling '99*, p. 234-245.

Lasseter J., 1987. Principles of Traditional Animation Applied to 3D Computer Animation. *Computer Graphics Proceedings (SIGGRAPH '87)*, ACM Press, p. 35-44.

Loop C., 1994. A G^1 triangular spline surface of arbitrary topological type. *Computer Aided Geometric Design*, vol. 11, p. 303-330.

Michalik P., Bruderlin B.D., 2004. Constrained-based design of B-spline surfaces from curves. *Solid Modeling and Applications Conference,* p.213-223.

Pernot J.P., Guillet S., Léon J.C., Falcidieno B., Giannini F., 2004. Multi-Minimisations for Shape Control of Fully Free Form Deformation Features. *Shape Modelling Conference*.

Peters J., 1995. C^1-surface splines. *SIAM Journal of Numerical Analysis*, vol. 32(2), p. 645-666.

Piper B., 1987. Visually smooth interpolation with triangular Bézier patches. *Geometric Modeling: Algorithms and new Trends*, G. Farin, editor, p. 221-233.

Rappoport A., Sheffer A., Bercovier M., 1995. Volume-preserving Free-form Solids. *Proceedings of Solid Modeling 95*, p. 361-372.

Sauvage B., Hahmann S., Bonneau G.P., 2004. Length Preserving Multiresolution Editing of Curves. *Computing*, vol. 72, p. 161-170.

Yvart A., Hahmann S., Bonneau G.P., 2005. Hierarchical triangular splines. *ACM Transactions on Graphics*, vol. 24(4), p. 1-18.