ITERATIVE GEOMETRIC DESIGN FOR ARCHITECTURE

IVO STOTZ[1], GILLES GOUATY[2], YVES WEINAND[3]

[1] Dipl. Arch. EPFL, Ecole Polytechnique Fédérale de Lausanne, IBOIS, Switzerland, ivo.stotz@epfl.ch
[2] Dipl. Ing. DEA, Ecole Polytechnique Fédérale de Lausanne, IBOIS, Switzerland, gilles.gouaty@epfl.ch
[3] Prof. Dr. , Ing. EPFL, Arch. ISA, Ecole Polytechnique Fédérale de Lausanne, IBOIS, Switzerland, yves.weinand@epfl.ch

Editor's Note: Manuscript submitted 19 December 2008; revision received 14 April 2009; accepted 16 April. This paper is open for written discussion, which should be submitted to the IASS Secretariat no later than December 2009.

ABSTRACT

This interdisciplinary research project presents a corporation of architects, mathematicians and computer scientists. The team researches new methods for the efficient realization of complex architectural shapes. The present work investigates methods of iterative geometric design inspired by the work of Barnsley. Several iteratively constructed geometric figures will be discussed in order to introduce to the notion of transformation driven geometric design. The design method studied allows interacting with the design forming affine transformations and generates discrete geometries.
Further, the handling of specific constraints is discussed. Geometrical and topological constraints aim to facilitate production of architectural free form objects. A surface method based on vector sums is studied. It allows designing free form surfaces that are entirely composed of planar quadrilateral elements. The combination of the proposed surface method and transformation driven iterative design provides new form-finding possibilities while satisfying a certain number of material and construction constraints.
Finally, the findings are tested on a series of applications. The studied test scenarios aim to evaluate the advantages of discrete geometric design in terms of efficient integrated production of free form architecture.

Keywords: *architecture, applied discrete geometry, IFS, timber construction*

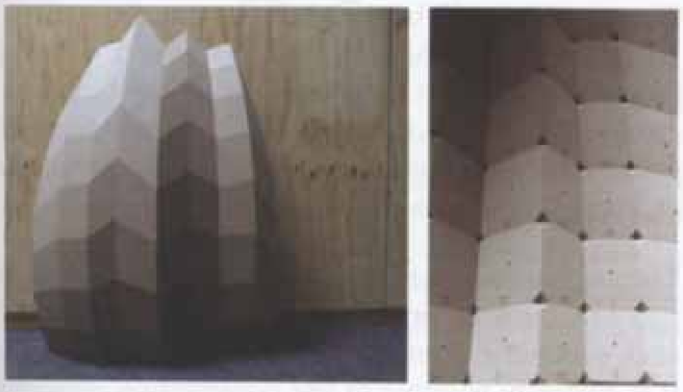

Figure 1. *Example of a free-form object designed using the presented surface method*

1. INTRODUCTION

In order to present the geometric design method studied, some mathematical background needs to be explained. Before explaining the principles of transformation driven geometric design, a series of historical examples will be discussed. This will introduce the reader to the methods of iterative geometric design. The relation between the mathematical method of geometric surface design and the physically constructed building will be shown by examples in the second part of this presentation.

2. MATHEMATICAL BACKGROUND

2.1 Of monster Curves...

The Cantor set (cf. figure 2), also called Cantor dust, is named after the German mathematician Georg Cantor. It describes a set of points which lies on a straight line. In the end of the 19th century, this figure attracted the attention of mathematicians because of its apparently contradictory properties. Cantor himself described it as a perfect set, which is nowhere dense [1]. Further properties, such as self-similarity, compactness and discontinuity, have been studied years later.

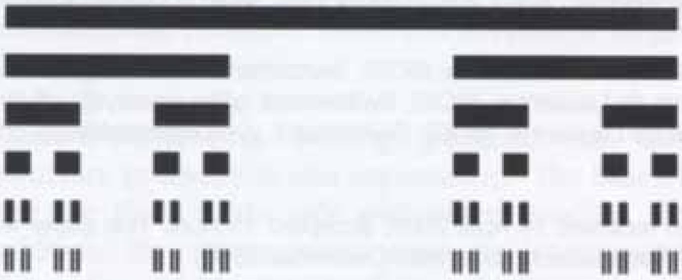

Figure 2. Cantor Set

The geometrical construction of the Cantor set can be explained as follows: Take a straight line segment, divide it into three parts of equal length and remove its middle third; divide again each of the resulting line segments and keep removing their middle thirds. If you repeat this for each of the new line segments, you will end up with the Cantor set.

The Von Koch curve belongs among the first found and best known fractal objects. In 1904, the Swedish mathematician Helge Von Koch described it for the first time in [2]. The Curve is constructed stepwise. Beginning from a straight line, there results a meandering curve with strange properties:

- It does not possess a tangent, which means that it can not be differentiated
- The length of any of its sections is always infinite

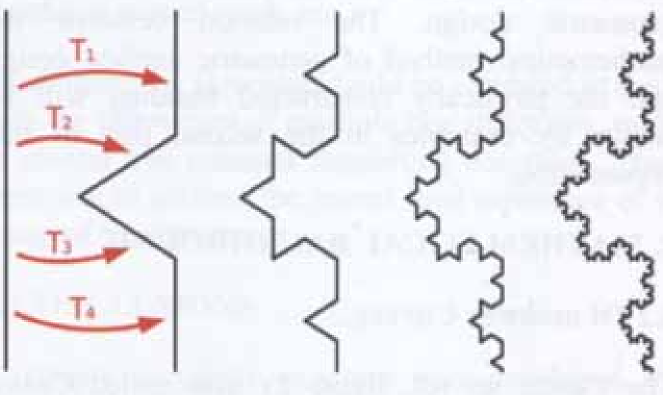

Figure 3. Von Koch Curve

The geometrical construction of the Von Koch curve is iterative, where each of the construction steps consists of four affine geometric transformations. The primitive is a section of a straight line, which is scaled, rotated and displaced by each of the transformations $\{T_1 ... T_4\}$. Per construction step, four duplicates are generated of which each will produce four more duplicates in the next construction step (cf. figure 3).

2.2 ... and Iterative Geometric Figures

The strange properties of the aforementioned objects led the mathematicians to name them "monster curves". In 1981, based on Hutchinson's operator [3], Barnsley defined a formalism which was able to describe such objects in a deterministic way [4]. His IFS-method (cf. section 2.3 Iterated Function Systems) consists in a set of contracting functions that are applied iteratively. In our case, a function is an affine geometric transformation. Iterative means that the construction is done step by step. The input of a construction step is the result of the step before.

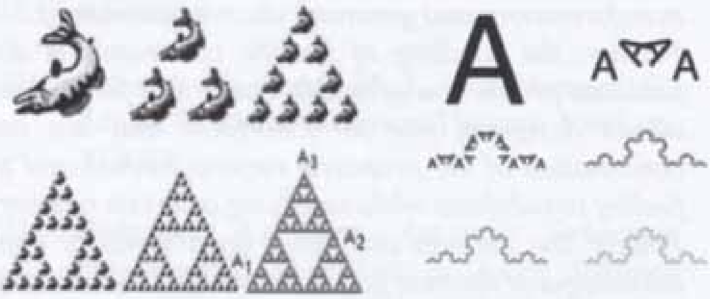

Figure 4. Sierpinski triangle and Von Koch curve according to Barnsley's IFS-formalism

What is really new in Barnsley's detection is that the resulting geometric figures are not defined by the primitive used, but rather by its transformations. As shown in figure 4, the construction of a Sierpinski triangle may use a fish as primitive. Analogous to this, the Von Koch curve might be constructed based on the letter "A". The result we end up with remains strictly the same.

The conclusion, that it is theoretically possible to use any form of primitive for the construction of such geometric figures, led us to the hypothesis that it is basically possible to use construction elements as primitives. Instead of using fishes like Barnsley, we would rather use construction elements such as beams or panels etc.

Figure 5. Iterative construction of a Bezier curve

In order to top off this series of introductory examples, we would like to address briefly the Bezier curve. In 1959, De Casteljau discovered a method for the construction today called the Bezier curve. De Casteljau's method [5] is based on iterative construction (cf. figure 5), which is highly similar to the construction of a Von Koch curve. The actual Bezier curve was analytically described by Bezier in 1961 as a polynomial function, which presents the headstone of today's CAD software.

2.3 Iterated Function Systems (IFS)

The geometric figures of the examples shown in the beginning of this section are all defined by a set of transformations $\{T_i\}$. As Barnsley teaches us, the result is indifferent to the initial object on which is applied the set of transformations. This is true for the limit state, also called attractor A, which is the resulting figure if the set of transformation $\{T_i\}$ is applied an infinite number of times. The attractor A is the unique non-empty compact such that

$$A = \bigcup_i T_i A$$

The attractor is a theoretical object, which we will not discuss any further. We will work on the intermediate construction steps that are given by the following sequence:

$$K_{n+1} = \bigcup_i T_i K_n$$

Where K represents the initial figure, the so called germ. K might be any arbitrary object as for example a fish, as shown in figure 4, or a straight line segment, as used for the construction of a von Koch curve or a Bezier curve (cf. figures 3 and 5). Finally, the following equation shall be mentioned, which relates the geometric figure K at construction step $n \to \infty$ to the attractor A:

$$\lim_{n \to \infty} \left(K_n \right) = A$$

Within the following, the modeled objects will be defined as the projection of an IFS via a projection operator P.

3. DISCRETE ITERATIVE GEOMETRIC DESIGN

The strange properties of the geometric figures discussed in section 2.1 and 2.2 are concerning the limit state, the attractor. For practical applications, the theoretical object of the attractor is far less relevant than its intermediate construction state K_n. In the scope of this work, the construction state K_n has following properties, which are beneficial for the application of free-form geometries in architecture:

- K_n is computational point by point
- The resulting geometry is always expressed by a finite number of elements

3.1 Transformation driven Geometric Design

As stated in section 2.3, the final aspect of the iteratively constructed geometry is defined by the transformations. In order to control the resulting figure, the geometric design method has to provide solutions, which will allow acting on the transformations used for the construction of the figure.

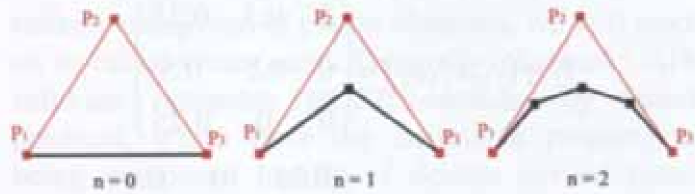

Figure 6. *First construction steps of a Bezier curve*

Let us discuss the example of an iteratively constructed Bezier curve as shown in figure 6. The figure shows the first construction steps of a Bezier curve with three control points P shown in red.

$$P = (p_1, p_2, p_3) = \begin{pmatrix} p_{1,x} & p_{2,x} & p_{3,x} \\ p_{1,y} & p_{2,y} & p_{3,y} \\ p_{1,z} & p_{2,z} & p_{3,z} \end{pmatrix}$$

At the beginning ($n = 0$), the initial figure K consists in the line connecting the end points of the control polygon $[p_1, p_3]$. Per construction step, two transformations $\{T_1, T_2\}$ are applied on each element. Below, the first three steps of the construction sequence are shown.

$$PK_0 = PK$$
$$PK_1 = PT_1K_0 \cup PT_2K_0 = PT_1K \cup PT_2K$$
$$PK_2 = PT_1K_1 \cup PT_2K_1 = PT_1T_1K \cup PT_1T_2K \cup PT_2T_1K \cup PT_2T_2K$$
$$PK_3 = PT_1K_2 \cup PT_2K_2 = PT_1T_1T_1K \cup ... \cup PT_2T_2T_2K$$

Each element of PK_i can be computed by following the construction tree shown in figure 7.

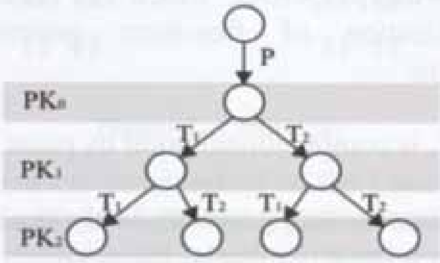

Figure 7. Construction sequence: Tree representation

For our work, transformation matrices are used to describe affine geometric transformations. We work with barycentric coordinates, which means that each computed point is based on a combination of the entry points. In the present example of a Bezier curve with three control points, each transformation can be expressed by one 3x3 matrix:

$$T_1 = (c_1, c_2, c_3) = \begin{pmatrix} 1 & 0.5 & 0.25 \\ 0 & 0.5 & 0.5 \\ 0 & 0 & 0.25 \end{pmatrix}$$
$$T_2 = (c_4, c_5, c_6) = \begin{pmatrix} 0.25 & 0 & 0 \\ 0.5 & 0.5 & 0 \\ 0.25 & 0.5 & 1 \end{pmatrix}$$

The above shown transformation matrices are pretty standard. They work with fixed values conducting always to the same result: a smooth Bezier curve with three control points.

A closer look at the transformation matrices of the Bezier curve shows that they have certain values in common. For example: The column c_1 of T_1 as well as the last column c_6 of T_2 is fixed. Further, the last column of T_1 is identical to the first column of T_2. These dependencies guarantee that the resulting curve will be **continuous**. The rest of the values are free and can be modified at the designer's desire.

If we want to generalize the above mentioned findings, we can say that certain values of the transformation matrices are constrained whereas others are free. Based on the example of an iteratively constructed continuous curve with two transformations and three control points, the general scheme of the transformation matrices is:

$$T_1 = \begin{pmatrix} 1 & c_{2,1} & c_{3,1} \\ 0 & c_{2,2} & c_{3,2} \\ 0 & c_{2,3} & c_{3,3} \end{pmatrix}, \quad T_2 = \begin{pmatrix} c_{3,1} & c_{5,1} & 0 \\ c_{3,2} & c_{5,2} & 0 \\ c_{3,3} & c_{5,3} & 1 \end{pmatrix}$$

In order to act on the free values of the transformation matrices, we defined a graphical data input method. We introduce the set of points $\{s_i\}$, which we call "subdivision points" (cf. figure 8). Each point s_i corresponds to the projection via P of a column c_i.

$$s_i = Pc_i$$

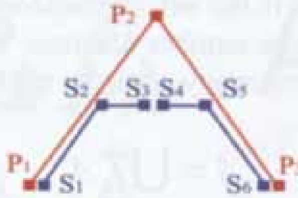

Figure 8. Representation of control and subdivision points

By this relation, the values of the columns c_i can be obtained by the position of the subdivision points s_i relative to the position of the control points P (cf. figure 9), using the equation below:

$$c_i = P^{-1}s_i$$

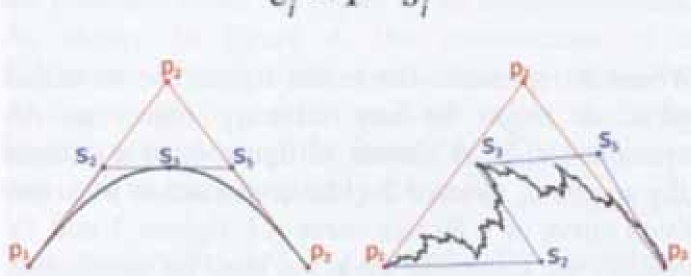

Figure 9. Acting on the transformation matrices by manipulating the subdivision points (blue)

3.2.1 Potential of transformation driven geometric design

Whether a figure is smooth or rough depends only on the affine geometric transformations. The same curve might be smooth or rough. By changing the subdivision parameters, respectively the smoothness and the roughness can be adjusted, as shown in figure 10. The input of the subdivision parameters is given by the position of what we named subdivision points. Alongside the control points, which are widely known in classical CAD-software, subdivision points augment the variety of design possibilities. They provide a graphical way to manipulate the affine geometric transformations, which are expressed in the user-unfriendly form of n-dimensional matrices that work under the skin of the graphical user interface.

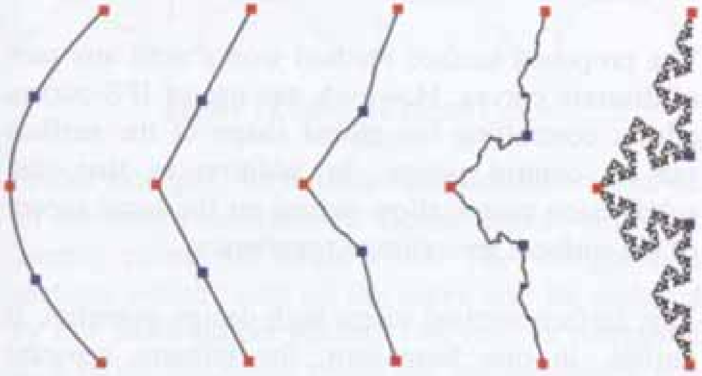

Figure 10. IFS-Curve curve design: adjustment of roughness / smoothness

3.3 Constrained Geometric Design

The goal is to develop design strategies that make the design and the production of free-form surfaces easier. Therefore, the geometric design should meet certain topological and geometrical constraints. The constraints are mainly dictated by physical and production conditions, coming from the field of construction. Within the following, a few examples of different constraints are presented.

E.g. an important point might be that the free-form object will be built out of planar timber panels. According to this, the geometrical constraint demands that the virtual 3D-model must be constituted completely of planar parts. We will work on this constraint in the sections 3.4 and 4.2.

In section two, we have presented a series of iteratively constructed objects. Not all of them are suitable for physical realization. The Cantor set for instance, is just a set of discontinued line fractions. Since we generally need material continuity (unless designing ornaments or such), we will have to verify that the created elements building up the geometric figure are connected with each other. Continuity represents a topological constraint.

In order to avoid complex detailing of the nodes of a wire frame structure, it is advantageous to know the number of bars coinciding in one node. To fix the number of bars per node to 6, we might work with surfaces which are entirely composed of regular triangular faces. This is a topological constraint.

On the one hand, constraints will make the physical realization of free-form objects easier. On the other hand, they may limit the design possibilities, and therefore restrict the form-finding process, which we want to avoid as far as possible.

3.4 A constrained surface model

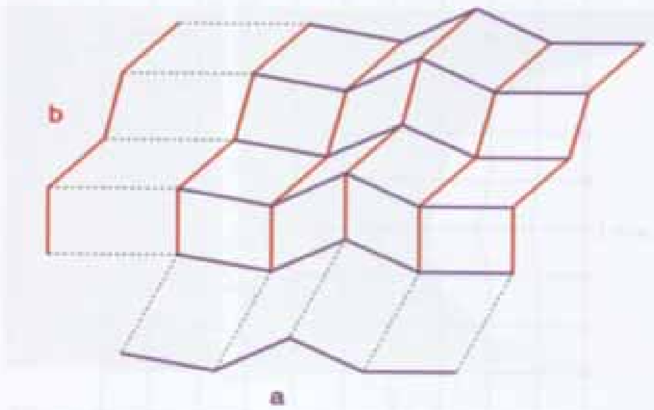

Figure 11. Surface design by vector sum

In order to create iterative surfaces, which are entirely composed of planar elements, we will work on so-called vector sums. Generally, classical CAD-software computes NURBS-surfaces by tensor products, which have the unsuitable property of being composed locally of double curved faces. Great effort is needed for their production. The principle of using vector sums, more percisely Minkowski sums [6], for the generation of free-form surfaces has already been studied by Schlaich [7] and Glymph [8]. Such surfaces are combinations of two curves. Figure 11 shows the curves *a* and *b*. The vector sum of any two segments of the curves (*a*,*b*) creates a parallelogram, which is part of the entire surface. The surface is completely composed of parallelograms and therefore it meets the geometrical constraint which requires that all its parts must be planar.

The discrete curves *a* and *b* used for the construction of the surface are represented by two list of points *A* and *B*. The resulting vector sum surface is represented by the quad mesh *M*:

$$M_{i,j} = M_{0,0} + \overline{A_0 A_i} + \overline{B_0 B_j}$$

Note: $M_{0,0}$ is the origin of the quad Mesh. It may be any arbitrary point in the design space.

The design possibilities of vector sums are limited compared to NURBS surfaces. As it, the quad mesh generated by vector sums is composed entirely of parallelograms. In order to augment the design capabilities, Schlaich [7] and Glymph [8] extended the method such that the resulting quad mesh is composed not only of parallelograms but also of trapezoids.

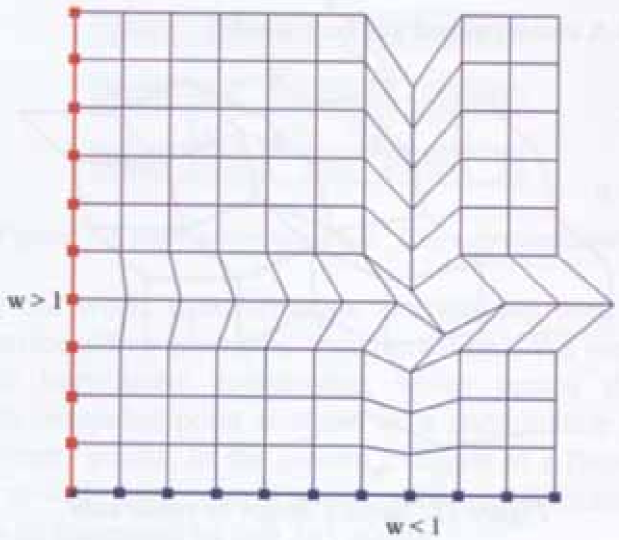

Figure 12. *Deformation of vector sum meshes by editing the weight of certain points.*

In order to extend the design capabilities, we employ methods of projective geometry. The IFS-formalism will be extended by the possibility of assigning different weights to its control and subdivision points. Each point will be defined by four coordinates (w,x,y,z). Assigning different weights w to the points allows deforming the quad mesh more or less locally. Hereby, the resulting quad mesh is no longer composed of parallelograms only but more generally speaking of convex planar quadrilaterals. To illustrate the effect of point weight editing on vector sums, figure 12 shows the simple case of a regular quad mesh where two point weights have been edited. Figure 13 shows the same principle applied to more complex meshes.

Within the scope of projective geometry, the coordinates (w , x , y , z) are called homogeneous coordinates. A point (x , y , z) in the model space $\mathbf{R}^3$ with a weight w has the corresponding homogeneous coordinates (w , xw , yw , zw). Reciprocally, each point of homogeneous coordinates (w , x , y , z) has corresponding $\mathbf{R}^3$ coordinates $(x/w , y/w , z/w)$. The passage from homogeneous coordinate to $\mathbf{R}^3$-coordinates is called projection.

In order to work with IFS-subdivision in homogeneous coordinates, the control points P must be defined in homogeneous coordinates. Hereon, the IFS-subdivision and the computation of the quad mesh M are done using homogeneous coordinates. Finally, the obtained result is projected back to the model space $\mathbf{R}^3$.

The proposed surface method works with any pair of discrete curves. However, the use of IFS-curves allows controlling the global shape of the surface via its control points. In addition to that, the subdivision points allow acting on the local aspect of the surface: smoothing / roughening.

This surface method offers high design potential. It unifies, in one formalism, the hitherto separate paradigms of the "smooth" and the "rough". Furthermore, it verifies a certain number of geometric constraints, allowing the optimization of the production of free-form architecture. Within the following, this will be shown by examples.

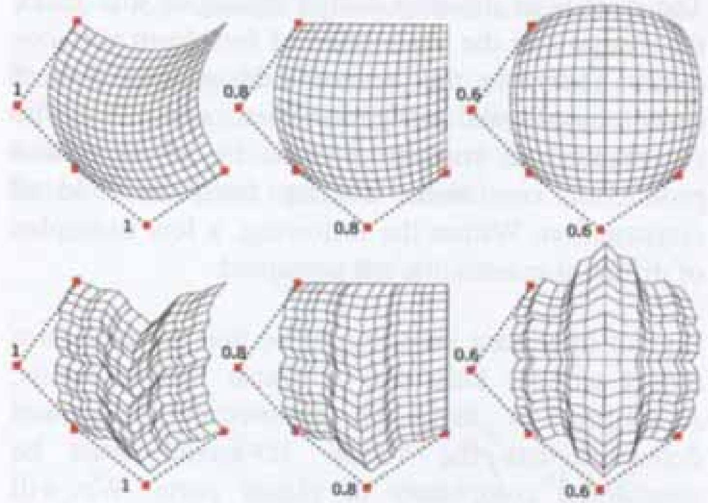

Figure 13. *Point weight editing of smooth and rough vector sum surfaces.*

4. APPLICATIONS

In order to realize physical buildings out of discrete virtual geometries, the elements, which constitute the 3D-models, are replaced by constructional elements. For an iteratively designed curve, the line sections will be substituted by linear constructional elements, such as planks or beams. In the case of a discrete surface, we replace its faces by planar constructional elements (panels, plates etc.). The substitution of geometric elements by constructional elements poses a certain number of questions as the geometric figures do not have physical dimensions like thickness. We will first discuss the more demonstrative case of a two-dimensional figure: the Bezier curve.

4.1 Discrete Bezier vault structure

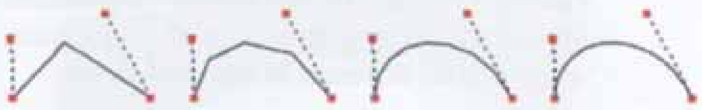

Figure 14. *Iterative Bezier Curve*

In this example, we build a vault structure based on an iteratively constructed Bezier curve with four control points (cf. figure 14). The straight line sections which build up the curve will be replaced by raw sawn timber planks. The vault is composed of a series of curves which are put jointlessly one next to the other. The planks are then screwed together in order to compose a massive timber vault structure. The shape of the vault's section can be controlled via the control points of the Bezier curve. Figure 15 shows a shape study where the curve's control points have been deformed such that the resulting shape is a meandering element with inflection points. The line segments of the underlying discrete Bezier curve have been replaced by constructional elements.

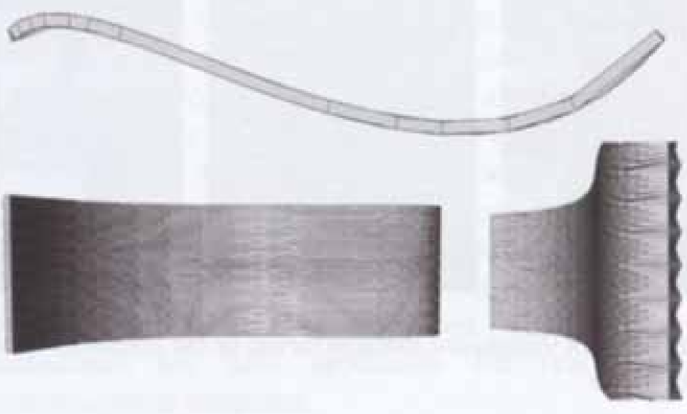

Figure 15. *Shape Study*

Once the shape has been defined, the curve will be subdivided into its parts until we will obtain adequate length for the constructional elements. On the one hand, the lengths of the elements should not be longer than the prevalent planks existing on the market. On the other hand, the subdivision should be fine enough to obtain a smooth rendering of the curve.

The relevant dimensions, which are necessary for the production of the constructional elements, are directly induced by the geometric figure. The lengths of the planks correspond to the lengths of the curve's line sections. The chamfer angle can also be deduced from the geometric model (bisector angle of two adjacent line segments). The design is therefore limited to two steps:

- Shape control, via the control points
- Subdivision control by choosing the adequate level of iteration

Figure 16. *Reduced Scale Model*

Figure 16 shows a reduced scale model of such a massive timber vault structure. The different planks have been cut by a computer numerically controlled routing table.

The question of how to partition a free-form object into a coherent set of constructional elements becomes obsolete because it is directly given by the iterative geometrical construction method. A direct link from design to production plans has been established, which is an important cost and time factor for the production of free-form architecture.

4.2 Shell structure – feasibility test

In this section we will discuss the application of an iteratively constructed free-form surface as a panel construction. The surface method used for the design of the free-form objects has been described in section 3.4. The design used for the realization of the larger prototype (cf. figure 21 in section 4.3), was mainly driven by following parameters: On the one hand, we wanted to realize a small domelike structure, presenting a smooth arc in its longitudinal section. On the other hand, we designed a rough curve, providing folds to the transversal section of the structure.

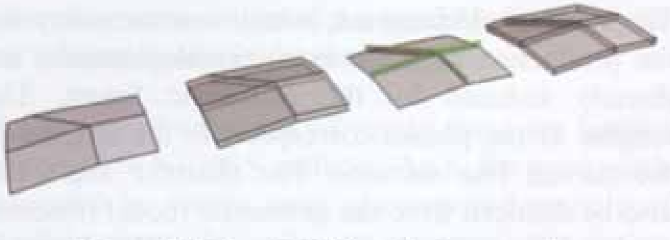

Figure 17. Parallel Offset Mesh generation

In the present example, the faces that compose the surface are replaced by planar timber panels. The choice of the thickness of the timber panel is important as the virtual 3D-surface does not present any thickness. A volume model has to be derived from the surface model. The thickening process is illustrated by figure 17. First, we generate a parallel offset surface, which holds a constant distance to the initial surface. The distance corresponds to the thickness of the timber panel. Second, the bisector planes are calculated, we will use them later for the chamfer cut of the panels. In this way, we design free-form objects that are entirely built up of planar constructional elements. Figure 18 shows an example of an IFS-surface whose faces have been entirely replaced by thickened constructional elements.

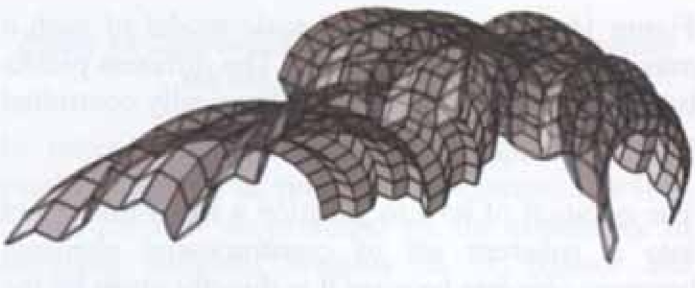

Figure 18. Thickened IFS-surface

4.3 Integrated Manufacturing

In order to test the established digital production chain, we did first produce an extract of an iteratively designed free-form surface by a 5-axis CNC-machine. The procedure to get from the geometry data of the constructional elements to the machine code has been mainly automated. To realize the parts of such complex shapes, the following work steps are necessary:

- A unique address for each constructional element is necessary for the logistical reason that the different elements can be assembled in the right place.
- Each element has to be oriented according to the coordinate system of the CNC-machine, the dimensions of the raw material and fiber direction of the plywood panel.
- Automatic generation of the machine code for each element: The material properties, the type of machine and the nature of the cutting tools are of the highest importance for integrated production of the elements, which are all different in size and shape.

Figure 19 shows a sequence of the machining process. The production of each multi part plate has been split into three work steps:

- Piercing of the fixation holes. Each part is screwed on the machining table.
- Engraving of the addresses of each element.
- Contour cut: machining of the actual element.

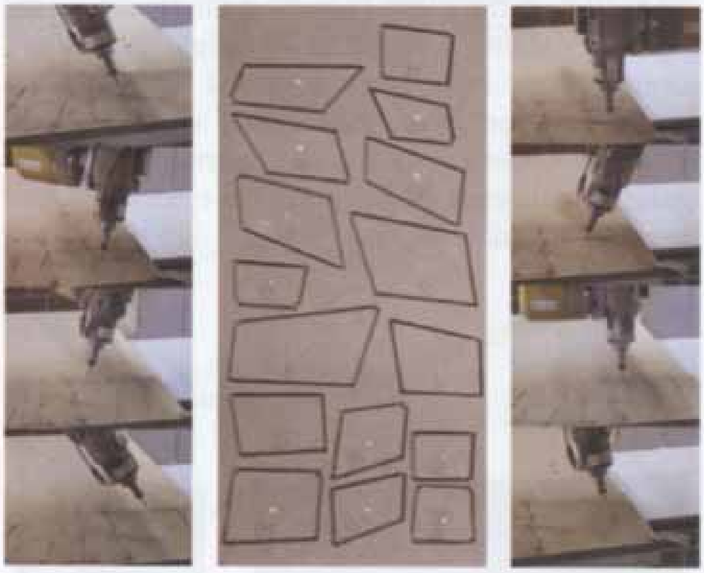

Figure 19. Integrated manufacturing of the constructional elements

The assembled manufactured elements give an accurate rendering of the surface designed on the computer screen. This shows that practical realization of iteratively constructed surfaces becomes possible.

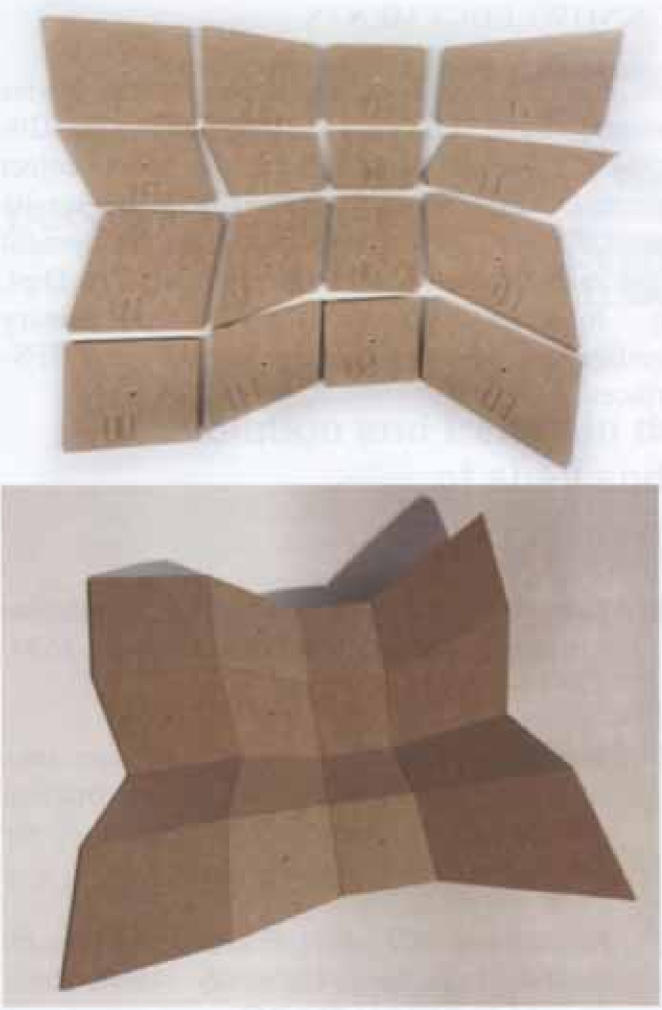

Figure 20. Assembling of the partial prototype

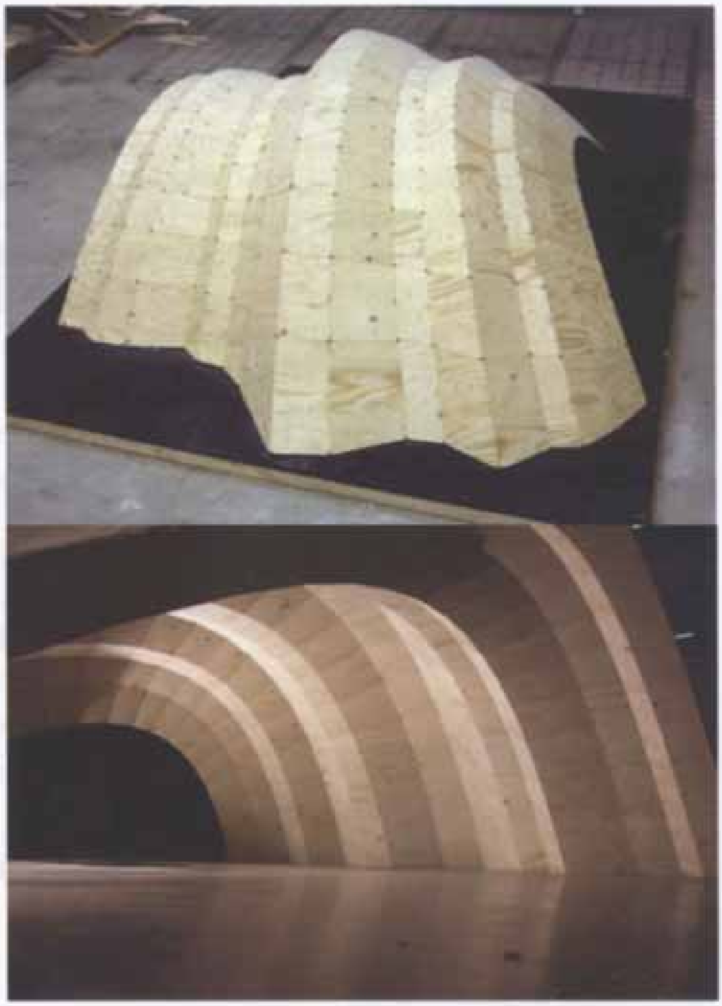

Figure 21. Second prototype

After the realization of an early partial prototype (cf. figure 20), we went on testing the method on a more complex structure which was composed out of 256 constructional elements. The shell structure shown in figure 21 presents a small vault spanning over four and a half meters; its plates are 10mm thick spruce plywood. Although the realized objects remain till today relatively small-sized, they allowed us to verify the validity of the proposed design method, since the employed manufacturing techniques are also applicable for real scale constructional elements.

5. DISCUSSION AND OUTLOOK

The presented applications show that the design and construction of free-form surfaces using our method requires a relatively small planning effort. Thus, several problems showed up during the manufacturing process because of the extremely low tolerances permitted by the perfectly fitting pieces. Big scale free-form buildings will probably loosen these tolerances, but the logistics and the montage will probably get more complicated. The efficiency of the method presented is only proved insofar as the processing of the data, from design to production, last only a few moments.

Further, the possibility of the design method, that allows the generation of rough and smooth objects, could be employed for the design of bearing plate structures, using the rigidity of the fold to improve the structural strength of the free-form surfaces.

Figure 22 illustrates a preliminary investigation on the bearing potential of such folded structures. On its left side, a smooth symmetrical plate structure is shown. On its right side the same model with added folds. In order to get a rough idea of the global structural behavior we imagined a plate structure spanning over twelve meters, built of 20mm plywood panels (spruce). First, we designed a smooth domelike shell structure. Hereon, we applied asymmetrical load of $F_z = 1500N/m^2$, which is a typical design value for snow load in central Switzerland. The maximum deflection occurred along the z-axis, which was about at 266mm. After adding folds, the FEM-analysis showed a maximum deflection value of 15mm.

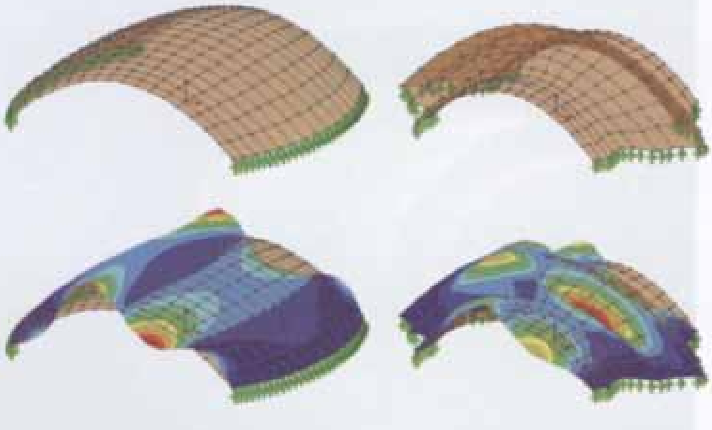

Figure 22. *Comparison of the a smooth and rough IFS-surface under asymmetrical load*

Note: This analysis was assuming hinge joints between the individual constructional elements. In reality, the joints may present a certain bending rigidity. Since the constructional elements are fixed along all four sides (except the border panels) any possible rotation is greatly limited by the system. For real scale applications, we think that adequate detailing is highly important. Further information about the joints of folded timber plate structures can be found in [9] and [10]. Buri et al. [9] used 2mm folded steel plates for joining massif block timber panels of 40mm. Haasis et al. studied in [10] the bending rigidity of screwed connections.

The above mentioned thoughts about detailing and structural properties of the presented work are possible subjects for future investigations.

In the future, we will continue the development of bigger and more complex objects. The potential of the new design method for free-form surfaces is far from being exhausted. We hope to meet further occasions to test our method on applications such as suspended ceilings, free-form facades, climbing walls, halls or shell structures.

ACKNOWLEDGEMENTS

This research has been supported by the Swiss National Fund (200021-112103) and (200020-120037/1). We also would like to thank our project partner, Dr. Eric Tosan from the LIRIS, Université Lyon I, France, for his support on the development of the mathematical model. Special thanks to Dipl. Ing. Johannes Natterer for the preliminary investigation on the bearing capacity of IFS-surfaces.

REFERENCES

[1] ***Cantor G.***, De la puissance des ensembles parfait de points, *Acta Mathematica 1884*, p.381—392.

[2] ***von Koch H.***, Une courbe continue sans tangente, obtenue par une construction géometrique élémentaire, *Arkiv för Matematik 1*, 1904, p.681-704.

[3] ***Hutchinson J. E.***, Fractals and self-similarity. *Indiana University Mathematics Journal* 30, 1981, p.713-747.

[4] ***Barnsley M. F.***, Fractals Everywhere, *Academic Press*, 1988

[5] ***De Casteljau P.***, Courbes à poles, INPI, 1959

[6] ***Minkowski H.,***. Geometrie der Zahlen. Leipzig, *Teubner*, 1896.

[7] ***Schlaich J., Schober H.***. Filigrane Kuppeln. Beispiele, Tendenzen und Entwicklungen, *Tec21*, 2002, vol..128, no.12, p.21-27.

[8] ***Glymph J., Shelden D., Ceccato C., Mussel J., Schober H.,*** A parametric strategy for freeform glass structures using quadrilateral planar facets, *Automation in Construction*, 2004

[9] ***Buri H., Weinand Y.,*** Gefaltet, Holztragwerke, *Tec21*, 2009, no. 8, p.18-22.

[10] ***Haasis M., Weinand Y.,*** Origami –folded plate structures. Engineering. *10th World Conference on Timber Engineering*, 2008, Miyazaki, Japan