U.P.B. Sci. Bull., Series A, Vol. 76, Iss. 1, 2014 ISSN 1223-7027

ALGORITHMIC PROCEDURES TO GENERATE SPATIAL COMPLEX GEOMETRIES

Ion CINCA[1], Andrei RĂDUCANU[2]*, Maria NICOLAE[3], Vasile Dănuț COJOCARU[4]

This paper presents a way to use parametric design in order to obtain complex spatial geometric forms addressed to freeform architectural objects. By practical building of the designed object, the viability of this design procedure is proven. In this case, the abilities of parametric design were used to determine the distinct shape of the "cell" as constitutive unit. The algorithm determines the shapes and dimensions based on the implemented geometrical relations between the shape of the cells and the base surface, as well as other input parameters.

Key words: complex geometries, parametric design, algorithmic procedures

1. Introduction

In the present are opened new possibilities concerning the realization of complex geometrical forms in architecture/constructions domain. By using rationalized constitutive elements assembled in respect with a specific composition law it is possible to generate structures with complex geometries, free-form types. An inconvenient of an irregular volumetric form is that it presumes currently the use of irregular, custom made constitutive elements, every element being different-unique, with a particular position in structure assembly, impossible to be changed. Consequently, the custom type industry is in a period of rapid expansion, stimulated of designer's needs to fulfil their no limits concepts.

Based on newest approaches, relationships between *form-structure-material-fabrication* are now reformulated. Design orientation towards material resources by "material based design" is a new trend with a theoretical fundamentals now coagulating [1,2,3]. The designer is now directly implicated in technologies materialization in all stages, from concept, to final production stages. New adaptive design procedures could adjust their structure or properties according to user demands.

[1] Assoc. Prof., University POLITEHNICA of Bucharest, Materials Science and Engineering Faculty, 313 Splaiul Independentei, Bucharest, RO 060042, e-mail: ion.cinca@mdef.pub.ro
[2]*Corresponding author: Ph.D. Stud., University POLITEHNICA of Bucharest, Materials Science and Engineering Faculty, 313 Splaiul Independen□ei, Bucharest, RO 060042, e-mail: zoster_andrei@yahoo.com
[3] Prof., University POLITEHNICA of Bucharest, Materials Science and Engineering Faculty, 313 Splaiul Independentei, Bucharest, RO 060042, e-mail: nicolae_maria@yahoo.com
[4] Assoc. Prof., University POLITEHNICA of Bucharest, Materials Science and Engineering Faculty, 313 Splaiul Independentei, Bucharest, RO 060042, e-mail: dan.cojocaru@mdef.pub.ro

In the present are created conditions to develop new concepts, processes, and design digital media based on advanced knowledge of designers in emerging domains such as smart geometry and mathematic algorithms [4,5,6].

This way a dynamic design system [7] is in progress with key-frame simulation and analysis tools called parametric or generative design, totally different from CAD, CAE, CAM known procedures.

2. Parametric design procedures and design stages

The design process based on parametric systems uses the computational capabilities to automate design stages. Besides an augmented efficiency, costs reducing, optimization and accuracy, parametric processes have as other objective an even more large design space exploring. Digital parametric design media have at this moment a series of specific procedures: shape grammars [8], Lindenmayer-systems [9], cellular automata [10,11], genetic algorithms [12], swarm intelligence [13,14]. Each of these procedures has some limitations which makes them applicable only in certain cases. Otherwise, there are also some similarities between them. In ideal conditions, the tendency would be towards an integrated parametric design system, which can include multiple procedures, as a new interactive generation of design instruments. This last desiderate is now only at concept stage. As it fallows, it will be briefly characterized the parametric design procedures listed above.

2.1 Parametric design procedures

Shape grammars: represent a set of rules which can be applied to generate a certain language or design rules [15,16,17]. Shape Grammars procedure can be used as design instrument to generate design languages or as an analysis instrument to understand a design already made. Accordingly to [8] can be defined four base components of Shape Grammars procedure: a finite set of forms; a finite set of symbols; a finite set of rules for forms; an initial form.

Lindenmayer-systems, (L-systems): L-systems [9] are mathematic algorithms known for the possibility of generating types of forms which have intrinsic the capacity to develop characteristics similar to biological growth. Lindemayer type procedures were used to have solutions to different complex problems referring to urbanism or urban development [18].

Cellular automata: The algorithm describes a growing process in a defined tram (bi-dimensional or tri dimensional), whose logic of new cell (elements) generation is based on neighbours positions of déjà pre-existing cells [10,11,19]. Cellular automata procedure was used to solve some problems in architecture and urbanism. The results obtained by cellular automata are similar to a tissue which is developing based on cell simple generation rules.

Genetic algorithms: Genetic algorithms represent mathematic algorithms inspired of evolutionary processes in nature [12]. Analogous operators (genes, genetic codes, genetic structure, etc) are used to describe a population status in a process which wants to optimise a so called "fitness function". Analogous operators currently used are: **gene** the smallest unity of a genotype; **genetic codes**: they are numbers or letters used for codification inside a genotype; **genetic description**: they are generic operations used in evolutionary systems which can be utilized also in genetic description; **genetic structure**: a set of genes with a certain order or a certain relationship; **genotype**: it represents the genetic structure of a design (not necessarily the physical aspect); **phenotype**: the observable properties of design, its form for example. Genetic algorithms were used in design with different aims, such as: optimisation [20,21], space distribution, design partition [22,23,24,25].

Swarm intelligence: this procedure is the most recently one, its concept is placed at early ”90” years. Using this procedure with operators named “agents”, there are defined rules which are deduced from social or collective behaviours [26,27,28]. The idea of self-organization is referring to coherent results which appear at global level in a system as a result of behaviours at small scale (agents behaviour), strictly in immediate proximity, without to be *conscious* of result at global level.

Perspectives for a possible integration in a single design system: till this moment cannot be presented examples of situations in which more than one procedure were used together to obtain solutions for a certain problem. Anyway, among specialists there is a consent concerning the idea that there is not yet realized a theoretical base on which a common language can transgress different parametric design procedures presented at this paragraph. Although, only at conceptual level exist efforts to propose a frame in which can be identified certain translation points inside a design process in which would be possible transition from a procedure to another. In this acceptation, in function of some limits, it would be possible to *navigate* on different routes depending on approached procedures and design process development.

2.2. Design stages

The first stage is to **define the logic that governs the way in which constitutive spatial structures elements connect** each other, firstly as a simplified geometry.

The second stage is **the implementation of the geometric principles** (logic) in a 3D modelling software. Once the logic is defined and the established algorithm runs, an array of possible solutions is generated. In this vast array of possibilities, the algorithm can search for a solution that performs exceptionally well. The design digital medium is Rhinoceros 3D, in conjunction with Grasshopper, which is a plug-in that enables to define algorithms. Grasshopper has an advantage over "traditional" ways of defining an algorithm (which use programming languages) consisting in a graphical interface. This way the successive steps became easier to see and mentally manage. As well it is friendlier to designers and saves time.

Finding a solution that performs well structurally is the third stage: the algorithm is used in order to search solutions which tend to be structurally as efficient as possible. Grasshopper has a function build in, called Galapagos, which browses through sets of various possible solutions.

Testing the structural performance of the realised spatial structure is the last design stage: it is done based on parameters regarding it's composing elements. Elements are implemented in extensions of the algorithmic software which are specialized on structural engineering analysis; Karamba, Kangaroo, Geometry Gym, RhinoBIM are these extensions. These are plug-in for Grasshopper and therefore work in an algorithmic way as well and therefore can easily perform calculations for large numbers of objects that have particular structural situations within the overall freeform shape. Once the different structural tests are performed, the results need to be compiled in a single number, the "fitness factor" which tells which solution is "better". This process is done for each solution that gets tested in Galapagos.

An algorithm stops when a solution is within an acceptable range of performance is achieved.

3. Experimental

One of the most powerful characteristic of parametric design is the fact that complex geometrical spatial shapes can be designed using an "adaptive module" as constitutive element (called further "cell"); this means that constitutive elements of complex geometrical spatial structure follow the same logic but respond to their particular position in the whole, thus being different in shape.

This kind of approach is now feasible for low budget projects due to recent developments in parametric design software, allowing one to manage large amounts of parametric data, as well as in mass customization tools that became a lot more accessible.

In this paper is presented an application which uses parametric design to obtain a spatial curved large dimensions object, similar to an igloo, generic called further Hexigloo (http://www.archdaily.com/146764/). This application uses such a variable module. The idea was to design a geometric architectural system in order to produce an igloo-like structure that people can enter at a street-festival in Bucharest, in this way promoting this new approach to architectural design. This work was developed at a generative design workshop at the "Ion Mincu"

University of Architecture and Urbanism in Bucharest and was presented at the Architecture Annual 2010 Bucharest.

3.1 Algorithm concept and realization for spatial curved large dimension object

Data referring only to first two stages in design process are presented in this paper: in the first stage is defined the logic that governs the way in which constitutive elements(cells) connect each other to generate the spatial structure and in the second stage is realized the implementation of the geometric principles (logic) in the 3D parametric modelling software. A shape grammar type algorithm was realised.

As a general principle, an algorithm is conceived in order to allow one to apply its generating logic onto different input geometry. Once the algorithm is realized, it can be applied numerous times to generate the geometry for a number of different input surfaces.

For this project it was envisaged to use hexagonal cells. The algorithm was developed in the idea to divide any given surface (not necessarily planar) into hexagonal cells.

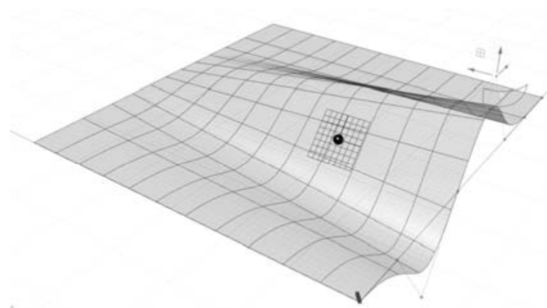

Fig. 1. Local coordinates system for a curved surface in NURBS software

The software used to elaborate the digital algorithm is Rhinoceros in conjunction with Grasshopper. This is a NURBS software, therefore any surface has a inherent double coordinate domain, similar to XOY in a Cartesian space, the difference being the coordinates follow the shape of the surface.

In a NURBS software, any surface is seen as an infinitely elastic and deformable rectangle which has a local coordinates system that gets deformed together with the deformation of the surface (Fig. 1). The unit division along the local coordinate system is therefore not constant and it is not equal to one unit in the Cartesian system.

The writing of the algorithm is an action on an abstract conceptual level. This means that the term surface is only a generic one.

It is important to mention that *the algorithm describes a process* and its execution materializes the design when it meets the input factors.

The algorithm was developed without a specific surface shape in mind. It is meant to be generally applicable to any given surface.

The process of dividing the surface into hexagons consists in a method that works as follows. The surface is reproduced as a flat rectangle in the Cartesian XOY system and is overlapped by a hexagonal grid. The coordinates of the corners that make up this grid are then "mapped" onto the surface, meaning they are interpreted as local coordinates within the surface. These mapped points are then connected using the same connection order as in the flat grid, resulting in the "mapped" grid onto the surface.

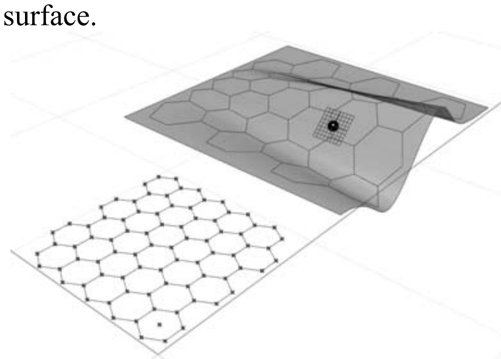

Fig. 2. Mapping hexagons onto the surface

In figure 3 one can see that a cell which is used to construct the spatial structure has itself a 3-dimensional shape, which was designed to have a hole on the upper side and one on the lower part; fig. 3 presents a rendered image of one possible outcome of the algorithm using hexagonal cells.

To obtain the virtual model of the cells' volumetric shapes, the following steps were undertaken.

The mapped hexagons are extruded downwards, in order to build up the lateral sides of the module (cell) (Fig. 4).

The lower part of the cells will be formed by hexagonal pyramids (Fig. 4) that are later virtually cut lower or higher, depending on the size of the hole that is determined for each module.

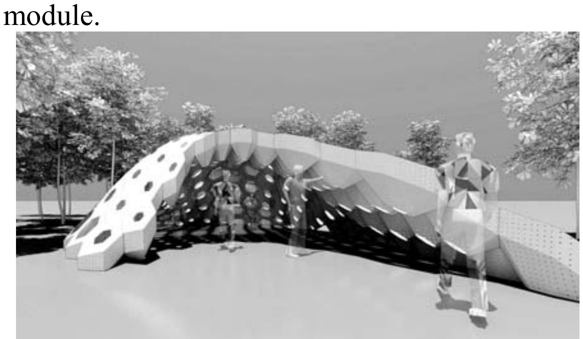

Fig. 3. Rendered image of one variant of Hexigloo

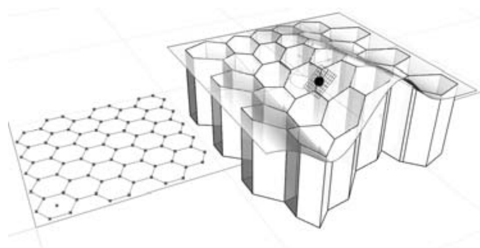

Fig. 4. Extrusion of the hexagonal contours

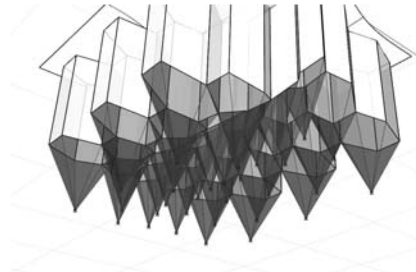

Fig. 5. Forming the pyramids

The dimension of the holes is influenced by an external geometric shape, usually called "attractor". In this case, it is a 3-dimensional curve that influences the dimension of the holes. The distance from the centre of each module to the closest point on the curve is measured (shown in fig. 6 with arrows). This parameter will influence the distance with which the original hexagon is offset (to offset means to construct a parallel curve to an existing one).

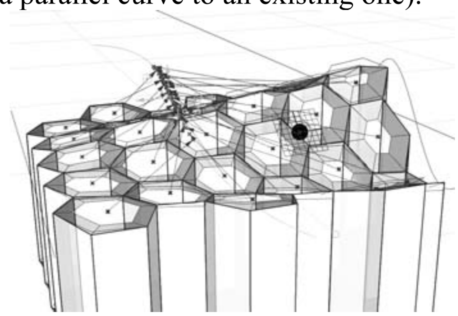

Fig. 6. Different hole sizes depending on the distance of the cells to the attractor

Firstly, a domain of possible hole dimensions needs to be established for the modules. The maximum hole size needs to leave a certain amount of brim that is defined arbitrarily and is valid for all cells. The minimum size hole is also defined arbitrarily, so that each cell is assured to have a perceptible hexagonal hole.

Once start and end values of the domain are established, the numeric values of the distances (from the centres of the modules to the attractor) need to be "scaled" to fit the domain of possible offset values. Therefore the largest distance from centre to attractor would equal the largest acceptable offset amount (and therefore smallest resulting hole) and vice-versa. In order to do this, a sub-algorithm needed to be developed which worked as follows. This algorithm can be used in any situation where values need to be "scaled" in order to fit a different domain.

The formula used is:

$$b = (a - B1)[(B2 - B1)/(A2 - A1)] + B1 \qquad (1)$$

where: *a* is initial number (to be scaled); *b* is scaled number (which is what needs to found); *A*1 is start point of initial domain; *A*2 is end of initial domain; *B*1 is start of target domain; *B*2 is end of target domain.

In order to "scale" the number for another domain, what matters is the difference in size between the two domains (initial and target). This is expressed in a ratio between the size of the initial domain and the target domain:

The ratio or "scale factor":

$$(B2 - B1)/(A2 - A1) \qquad (2)$$

is used to "scale" the input number, but we are not interested in the number relative to a random/arbitrary position of zero, but rather to the size of the domain, that is why we subtract the start number of the domain.

Regarding values for: $(a - B1)$ (3)

after the number is scaled, it needs to be "shifted" forwards or backwards in the new system of reference, depending on where the target domain is located, consequently the start value of the target domain is added.

The resulting numbers: $+B1$ (4)

are used in the offset operation and determine the hole size of each cell.
The hole size dictates the level at which the inverted pyramid of the cell needs to be cut (therefore also the end height of the truncated pyramid).

3.2 Results

Modules execution and assembling. The 3-dimensional cells were digitally unfolded onto a single plane using a build-in function of the 3D modelling software.

The contours were then laser cut out of corrugated fibreboard and folded and glued together into their final shape (fig. 7).

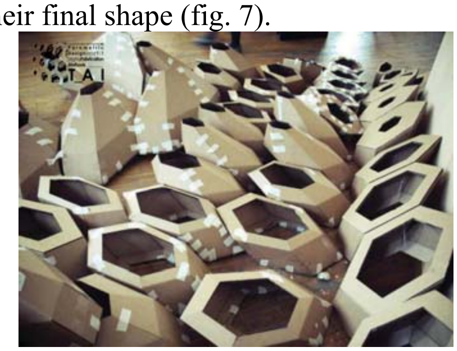

Fig. 7. Build cells

A map of connectivity for the cells was produced and they were assembled into zones that were not too big to be manageable by 2 people.

This sub-assemblies were then connected on site. The cells were put together using zip-ties and hot-melt adhesive.

The image of the spatial structure, named Hexigloo, obtained after the cell assembling, is presented in fig. 8.

Fig. 8 Finished spatial structure-Hexigloo

4. Conclusions

This paper presents a way to use parametric design in order to obtain complex spatial geometric forms addressed to freeform architectural objects.

By practical building of the designed object, the viability of this design procedure is proven.

In this case, the abilities of parametric design were used to determine the distinct shape of the "cell" as constitutive unit. The algorithm determines the shapes and dimensions based on the implemented geometrical relations between the shape of the cells and the base surface, as well as other input parameters, such as cell size, height of the overlapping sides of the cells, etc. The base surface can be changed and the adapted result is returned in a split second. This is valid also for changes of the "attractor", resulting in different dimensions for the holes. This almost instant feedback is possible due to the automated execution of the different steps that form the algorithm. Without this kind of approach, adapting the design to new parameters (such as a different bass surface, different attractor, etc) would require tedious work that is very time-consuming.

The next step is to extend the developed models by using other materials, metallic one for example, in order to study the behaviour of the materials in a real case scenario from a material engineering point of view and implement the conclusions in the design process in order to maximize the output.

R E F E R E N C E S

[1]. *Rivka Oxman*, Informed tectonics in material-based design, Design Studies, **vol. 33**, 2012, pp. 427-455

[2]. *Ann Heylighen and Matteo Bianchin*, How does inclusive design relate to good design? Designing as a deliberative enterprise, Design Studies, **vol. 34**, 2013, pp. 93-110

[3]. *Nada Bates-Brkljac*, Assessing perceived credibility of traditional and computer generated architectural representations, Design Studies, **vol. 30**, 2009, pp. 415-437

[4]. *H. Whitehead*, The origins of smart geometry, L. A. Jeffress (Ed.), Inside Smart Geometry, New York: John Wiley, 2013
[5]. *M. Burry*, From descriptive geometry to Smart, L. A. Jeffress (Ed.), Inside Smart Geometry, New York: John Wiley, 2013
[6]. *H. Whitehead*, The practice of smart geometry, L. A. Jeffress (Ed.), Inside Smart Geometry, New York: John Wiley, 2013
[7]. *Vishal Singh and Ning Gu*, Towards an integrated generative design framework, Design Studies, **vol. 33**, 2012, pp. 185-207
[8]. *G. Stiny*, Introduction to shape grammars. Environment and planning B: Planning and Design, **vol.7**, 1980, pp. 343-351.
[9]. *A. Lindenmayer*, Mathematical models for cellular interaction in development I. Filaments with one-sided inputs. Journal of Theoretical Biology, **vol. 18**, 1968, pp. 280-289
[10]. *J. von Neumann*, The general and Logical theory of automata. In L. A. Jeffress (Ed.), Cerebral mechanisms in behavior-the Hixon symposium, New York: John Wiley,1951, pp. 1-41
[11]. *S. Wolfram*, A New Kind of science. Wolfram Media Inc. 2002
[12]. *J. H.Holland*, Adaptation in natural and artificial systems. Ann Arbor: University of Michigan Press, 1975
[13]. *J. L. Deneubourg*, Application de l'ordre par fulctuations a la description de certaines etapes de la construction du nid chez les termites. Insect Soc, **vol. 24**, 1977, pp. 117-130
[14]. *A. Payman*, Swarm intelligence. Pasadena, CA: Jet PropulsionLaboratory, 2004
[15]. *T. W. Knight*, Designing a shape grammar. In J. S. Gero, & F. Sudweeks (Eds.), Artificial intelligence in design, The Netherlands: Kluwer Academic Publishers, pp. 499-516, 1998
[16]. *H.Koning and J. Eizenberg*, The language of the Prairie: Frank Lloyd, Wright's Prairie houses. Environment and Planning, **vol. 8**, 1981, pp. 295-323
[17]. *J. P. Duarte*, A discursive grammar for customizing mass housing: the case of Siza's houses at Malagueira. Automation in Construction, **vol. 14**, 2005, pp. 265-275.
[18]. *G. Kelly and H. McCabe*, Interactive generation of cities for real-time applications. In ACM SIGGRAPH 2006 research, Boston, Massachusetts: ACM, pp. 44, 2006
[19]. *R. Krawczyk*, Architectural interpretation of cellular automata. In Generative art 2002.
[20. *G. N.Bullock, M. J.Denham, I. C.Parmee and J. G. Wade*, Developments in the use of the genetic algorithm in engineering design. Design Studies, **vol. 16**, 1995, pp. 507-524
[21]. *I. Hybs and J. S. Gero*, An evolutionary process model of design, Design Studies, **vol. 13**, 1992, pp. 273-290
[22]. *P. S. Coates*, Review paper: some experiments using agent modelling at CECA. In 7 th generative art conference 2004
[23]. *J.S. Gero and V.S. Kazakov*, A genetic engineering approach to genetic algorithms, Evolutionary Computing, **vol. 9**, 2001, pp. 71-92
[24]. J. Jo and J. S Gero, Space layout planning using an evolutionary approach. Artificial Intelligence in Engineering, **vol. 12**, 1998, pp. 149-162
[25]. *L.Caldas, and L. K. Norford*, Architectural constraints in a generative design system: Interpreting energy consumption levels. In Building simulation 2001. International Building Performance Simulation Association, 2001
[26]. *K. Carley*, Sociology: computational organization theory. Social Science Computer Review, **vol. 12**, 1994, pp. 611-624.
[27]. *J. C.Kunz, R. E. Levitt and Y. Jin*, The virtual design team: a computational simulation model of project organizations. Communications of the Association for Computing Machinery, **vol. 41**, 1998, pp. 84-92.
[28]. *E. Bonabeau, M.Dorigo, and G. Theraulaz*, Swarm intelligence: From natural to artificial systems. Oxford University Press, 1999