R. Stouffs, P. Janssen, S. Roudavski, B. Tunçer (eds.), *Open Systems: Proceedings of the 18th International Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA 2013)*, 821–829. © 2013, The Association for Computer-Aided Architectural Design Research in Asia (CAADRIA), Hong Kong, and Center for Advanced Studies in Architecture (CASA), Department of Architecture-NUS, Singapore.

MESH GRAMMARS

Procedural Articulation of Form

Michael HANSMEYER and Benjamin DILLENBURGER
Swiss Federal Institute of Technology Zurich, Zurich, Switzerland
hansmeyer@arch.ethz.ch, dillenburger@arch.ethz.ch

Abstract. We introduce a formal grammar as a computational approach to the generation of design. While existing shape-grammars transform primitive shapes as lines or rectangles, the presented production system specifically addresses polyhedral objects described by three-dimensional meshes composed of vertices, edges and faces. The parameters of the transformation rules are sensitive to topological and topographical properties of the selected input mesh. We demonstrate that this approach allows the creation of new ornamental structures and can lead to a new language of architectural forms.

Keywords. Generative; procedural; subdivision; shape grammars.

1. Introduction

This paper introduces a new approach for procedural design of architectural forms and ornaments called “mesh grammars”. Using procedural design, one can generate complex shapes that cannot be drawn with an acceptable effort directly via user input. With small adjustments to the rules, a wide range of variations can be calculated in a short time.

A number of procedural modelling techniques exists to create three-dimensional models from sets of rules: L-Systems and fractals are often used to model plants or landscapes, while shape grammars have been applied on a theoretical level in the fields of art and architectural design. Despite their prevalence in academia, shape grammars have not gained a widespread relevance as a generative design tool in architectural practice. Their use has primarily been limited to analysing, encoding and re-applying existing architectural composition rules.

The motivation of our research is to develop a procedural instrument that allows design of new, complex architectural forms and novel structures of ornamentation. We therefore extend the possibilities of existing shape grammars with a grammar for the articulation and refinement of polygonal meshes.

This paper first gives a short overview and discussion of the related concept of shape grammars. In the second part, a formalism called "mesh grammars" is introduced. Finally, we present examples of forms generated by applying these mesh grammars, and we conclude with an outlook. In summary, this paper makes following contributions that have not been demonstrated:

- A formal grammar addressing multi-facetted mesh geometry.
- Context-sensitive parametric rules responsive to both topological and topographical attributes on a local and global level.
- Application of the mesh grammars to the generation of architectural artefacts.

2. Conceptual Similarity to Shape Grammars

Stiny and Gips (1971) introduced shape grammars as a computational technique for rule-based design. Shape grammars can be seen as the graphical correspondence of production-systems used for languages. The rules of production systems are composed of two parts: a left-hand side a sensory precondition ('if' statement), and right-hand side action ('then' statement). In shape grammars, these two parts are defined as:

- Recognition of a particular shape and selection of the corresponding transformation.
- Replacement of the shape by a new shape, transformed according to the specification of the applied rule.

Shape grammars are a useful instrument for computer-aided architectural design under formal aspects. Composition principles can be digitally encoded as rules. The combination of different rule sets enables a large variety of designs and high detailing with relatively small programming effort. Architectural shape grammars have been mostly used for analysing composition rules of built architecture (Flemming, 1987) and for the computational reproduction and recombination of existing "styles" (Duarte, 2005; Müller et al., 2006).

Despite the theoretical work on shape grammars as an analytical tool during the last thirty years, shape grammars have neither found widespread implementation as generative tool for design, nor are they documented as design method for built architecture. The potential of a rule-based procedural design as generator of new architectural forms is as of yet largely unrealized. The authors identify two broad manners in which a rule-based design tool can be extended to from an analytical tool to a generative instrument.

2.0.1. *Moving from primitive shapes to polyhedral objects*

As the rules of shape grammars mostly use primitive shapes as input (such as points, lines, planes and solids), the amount of implemented rules is often very high. For instance, the shape grammars used to generate floor plans of Palladian villas (Stiny and Mitchell, 1972) consist of 69 rules that are processed sequentially in eight steps.

The rules of the presented mesh grammars focus on polyhedral three-dimensional objects. These objects are described as polygonal meshes composed of connected faces, edges and nodes. The process output is not an agglomeration of isolated shapes but again a mesh, i.e. a network of connected elements.

2.0.2. *Context-sensitive parametric modifications*

The rules of shape grammars usually operate using affine transformation, addition, splitting, or direct replacement of shapes. When using parametric rules, these operations are adapted according to properties such as size or proportion of the input-shape.

The rules of the presented approach include vertex-based three-dimensional operations of shape-internal deformations with context-sensitive parameters. They deform the mesh at each location in a differentiated way depending on the local and global context.

2.1. THE MESH GRAMMAR PROCESS

The starting condition of the process is a polygonal mesh of any level of detail and complexity. The process itself consists of applying a rule to this mesh in order to produce an output mesh (Figure 1). This application can be repeated for a fixed number of iterations or it can continue until the mesh exhibits a specific attribute.

The rule itself has two components: an attribute of the mesh that is to be measured, as well as a mapping function that specifies how the measured value is

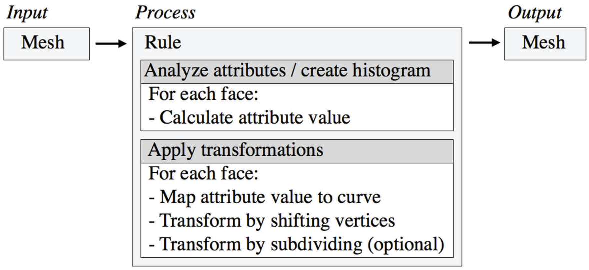

Figure 1. Mesh grammar flow chart.

converted into a transformation of the mesh. Mesh transformations consist of translation of vertices along faces' normal vectors. Optionally, transformation can include subdivision of the mesh.

2.2. MESH ATTRIBUTES

Mesh attributes are measured at the level of individual faces. They can broadly be classified as either topographical or topological, and one can distinguish between attributes that are local and can be measured using only a single face, and those that are global and measure a face attribute relative to its context (Figure 2).

The spatial distribution of any attribute can be monitored at any iteration by visualizing faces' values as a grey-scale map (Figure 3). This provides a preview of how vertices can be translated.

	Topographical	Topological
Local	• Area • Perimeter • Jaggedness • Parallelness • Planarity	• Number of neighbors • Number of vertices
Global	• Curvature • Orienation • Distance to point/ border/centroid /etc.	• Nodal distance to border • Nodal distance to corner

Figure 2. A selection of mesh attributes.

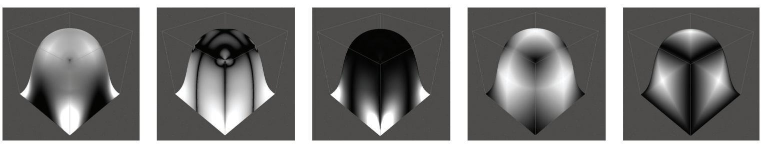

Figure 3. Attributes of a subdivided mesh visualized as shades of grey. From left to right: curvature, planarity, jaggedness, distance to initial vertex, and distance to initial edge.

2.3. MAPPING FUNCTION

The link between a measured attribute value and a translation operation is a mapping function. This function, usually in the shape of a curve, specifies how an attribute value is converted into a translation of vertices. An underlying histogram that measures the distribution of either absolute or relative (percentile) attribute values can be used to monitor and control the mapping (Figure 4). The histogram allows an easy specification of clipping values, and can provide visualization of distributions using different scales on the x-axis – such as a logarithmic scale.

Mapping functions provide control of a spatial transformation of the geometry through the use of a linear pattern. The histogram of attributes indexes the three-dimensional structure of the mesh in a linear order. By mapping this distribution of values to the parameters of the rules with various curve-functions, the resulting transformation of the mesh is manifested in three dimensions (Figure 5).

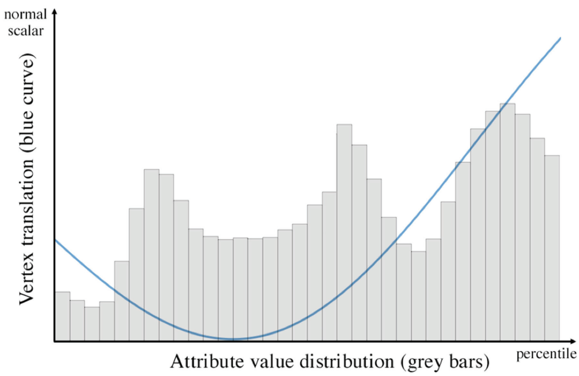

Figure 4. Mapping function with underlying histogram.

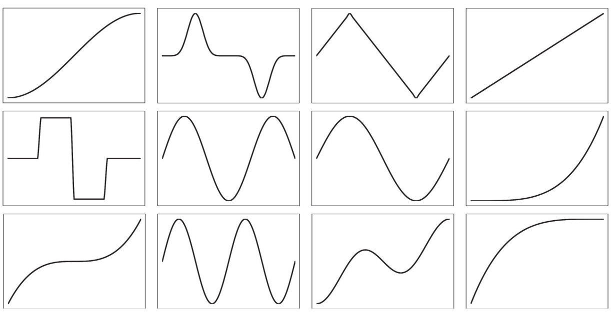

Figure 5. A selection of mapping functions.

2.4. MESH TRANSFORMATION

Mesh transformation always entails a translation of vertices to generate a new output mesh, and can optionally be accompanied by mesh subdivision.

2.4.1. Translation vectors

Vertices are translated using their normal vectors, which in turn are multiplied by a translation scalar. The translation scalar is specified by the mapping function. The unscaled extrusion vector can be calculated based on the dimensions of the face, or calculated as a unit vector that in turn can be scaled corresponding to the current iteration.

2.4.2. Mesh subdivision

Transformation can optionally entail mesh subdivision after each extrusion operation. This enables the algorithm to work at increasingly smaller scales: first iterations can control the overall form, while successive iterations affect surface development and eventually the creation of a microstructure or skin (Figure 6). By

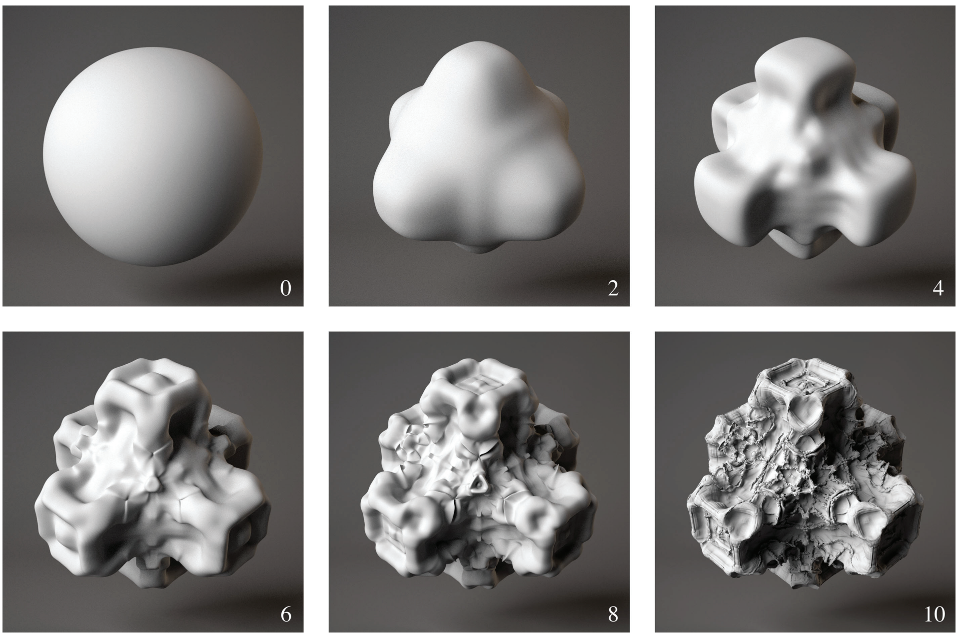

Figure 6. Successive application of a single mesh grammar rule to a sphere.

using the Catmull-Clark scheme (Catmull and Clark, 1978), for instance, it is possible to further differentiate the mesh by specifying individual extrusion factors for corner points, edge points, and midpoints of the new mesh (Hansmeyer, 2010).

3. Conclusion

3.1. SIMILARITIES AND DIFFERENCES TO SHAPE GRAMMARS

Mesh grammars can extend the range of applications of shape grammars. As a generative design tool, they provide an unseen vocabulary of forms and ornament that cannot be created using traditional design methods (Figure 7). As meshes are a generic description format of three-dimensional geometry, the presented procedural approach can easily be adapted to given architectural contexts and to different scales of architectural elements.

Although the presented process is deterministic – as the same combination of rules applied on the same mesh produces always the same result – it is difficult

Figure 7. Mesh grammars with a spherical input mesh (rendering). *(Continued on next page…)*

Figure 7. (Continued...)

to foresee the outcome. As attributes of one mesh always result from attributes of the preceding mesh, small changes in the weighting of parameters can propagate and lead to new and oftentimes surprising results. Thus, while the design is only truly controllable in an iterative trial-and-error process, this unpredictability offers the chance of discovering unexpected results in an open formal range.

With the inclusion of parametric rules into the grammars, local attributes of the mesh are uncovered, articulated and accentuated. Through the context-sensitivity of the transformations, the resulting structures maintain an organic connection to the input shape. The generated ornament appears to evolve out of the mesh rather than being an afterthought. Structure and ornament are fused into a single entity. Even the simplest input mesh with only a single rule can evolve into a highly articulated shape within less than 10 iterations.

3.2. OUTLOOK

The potential of mesh grammars as a generative tool for architecture could be further explored through a refinement of the presented transformation routines. This could include, for instance, the integration of physical constraints or of geometrical constraints such as collision-detection.

It has recently become possible to materialize the forms hat this mesh grammar process can generate. The development of both subtractive and additive digital fabrication methods is happening hand-in-hand with the increase in computational power necessary to calculate high-resolution meshes. Fort he first time, geometric complexity is no longer an impediment, but in its malleability it becomes an opportunity.

References

Catmull, E. and Clark, J.: 1978, Recursively generated b-spline surfaces on arbitrary topological surfaces, *Computer Aided Design*, **10**, 350–358.

Duarte, J. P.: 2005, A discursive grammar for customizing mass housing: the case of Siza's houses at Malgueira, *Automation and Construction*, **14**(2), 265–275.

Flemming, U.: 1987, More than the sum of parts: the grammar of Queen Anne houses, *Environment and Planning B: Planning and Design*, **14**, 323–350.

Hansmeyer, M.: 2010, Subdivision beyond smoothness, *Proc. Computational Aesthetics*, London.

Müller, P., Wonka, P., Haegler, S., Ulmer, A. and Van Gool, L.: 2006, Procedural modelling of buildings, *ACM Trans. Graph.*, **25**(3), 614–623.

Stiny, G. and Gips, J.: 1972, Shape grammars and the generative specification of painting and sculpture, *Information Processing '71*, North-Holland, Amsterdam, 1460–1465.

Stiny, G. and Mitchell, W. J.: 1978, The Palladian grammar, *Environment and Planning B: Planning and Design*, **5**(1), 5–18.