Can a Computer Implementation Based on Set Grammars Allow Emergent Shapes?

Luís Romão,
School of Architecture and Planning, Massachusetts Institute of Technology, USA, and Faculdade de Arquitectura, UTL, Portugal
lromao@mit.edu

Abstract. *In a previous paper, (Romão 2005) it was shown that a designer could create his own rules and combine them for application in a deterministic way using a computational device based on Set Grammars called SGtools. Using this device, the designer can assess the results in a visual manner and then change the rules without any knowledge of a programming language. This work examines whether SGtools can deal with emergence by coupling the representational abilities of Set Grammars with the search power of hybrid algorithm inspired on Genetic and Taboo search algorithms. The use of this search algorithm enhances the ability of the designer to explore solutions in practical time thereby enabling him/her to find unexpected, emergent solutions.*

Keywords: *Shape Grammars, Generative Design; Prediction and Evaluation.*

Introduction

Two strategies have been used in visual art: grabbing shapes as objects or formulating objects from shapes. Francisco d'Ollanda (1955) mentioned these strategies through the words of Michelangelo Buonarotti, when he argued for differences between the Flemish painters and the Italian ones. According to Ollanda, Michelangelo said that Flemish painters drew things back and forth to compose a painting, using landscapes, saints, and other symbolic "objects", which Italian painters did not. The relevance of this quotation is that its implicitly identifies two processes of representing art through shapes that can be used for transposing processes and representation to computation. The statement of Kurt Badt, quoted by Rudolf Arnheim (1996), about how impressionist and pos-impressionists saw shapes can help to illustrate this argument. As he pointed out, both strategies apply for creativity but not in the same way:

The Symbolists derived their representation of the world from individual objects; they built it around single figures, composed it of objects, in Latin: res. Their intention was that of realists, regardless of the meaning they attributed to the objects. The Impressionists proceeded from impressions of the whole, from a connexion of things, into which these things had grown and which they had created by their natural growth. ... In their conception of the world and in the intention of their art, which had the task of showing that conception, the Impressionists were naturalists (using the word nature in its original sense of nasci: being born, wanting to become, growing.) This means that there was in fact a profound difference between the two artistic tendencies. But there is no difference of rank or value between the two conceptions of reality. They are two equally good aspects of the same thing. For this reality of the world exists, in man's conception, as connexion but also as segregation because the two can be thought of and represented only in mutual relation.

The first part of this statement shows some relevance for our argument: symbolists represent the world from a collection of objects. In opposition, others represent reality as a whole framed by connections. For different reasons, artists decide to use one of the approaches, or both. Therefore, the idea of designing with shapes as symbols with hierarchical relations is not new, and it is worth to explore in computation.

Background

In recent literature, several authors have claimed for a shift of paradigm in design and computation (Mitchell, 1990; and Emdanat, 1999). The procedure of transposing representation and process in computation could follow the bottom up approach used by Rodney Brooks (2005) in the cog robot project. He says: It turns out to be easier to build real robots than to simulate complex interactions with the world, including perception and motor control. Leaving those things out would deprive us of key insights into the nature of human intelligence. In this approach, methods of interaction are evaluated and tested in order to fulfill a distant goal. The concept of starting a design process from almost nothing is common in design practice. After the initial conceptual sketches, a huge collection of regulations and technical issues is considered in the project development. A question then arises, how can one conceive a tool that supports this incremental development of the design process? How can one assure that this tool is flexible enough to allow the exploration of different formal universes without jeopardizing productivity?

Terry Knight (2003) stated that Shape Grammars (SG) are characterized to support intelligibility, and evolutionary computation systems for productivity. Some examples of combination of SG and of Genetic Algorithms (GA) or of Simulated Annealing (SA) have been experimented inspired by those claims, however, the expectation of a final and a clear production result is lower for designs that are beyond the engineering design context. The methods of research have been used to evaluate building performance, or predict energy conservation (Caldas and Rocha, 2001) or design structural analyses components (Shea and Cagan, 1999). This type of simulation and some articulated process and representation are nicely done. The characteristics of these areas have certain well-defined values that can be represented by algorithm operations. However there are other aspects of design more difficult to quantify. It is necessary to answer to some minimal requirements: to promote a good computer representation of shapes and associated elements, to promote a process framework to manipulate that representation and to promote a resourceful control to act on information. In addition, John Gero (1999) introduced the concept of constructive memory to answer the need of explaining some of the phenomena that occurs in designing and design thinking. He identifies a gap between "reinterpretation" and "unexpected discovery". He also argues that emergence can happen by a reinterpretation of existent shapes. In fact, natural metaphor process seems to miss human memory representation. In fact, it was introduced a combination of GA and TS (Hansa, 2003) in some experiments. This mixing was justified because TS process narrows too much the field of search in opposition with the other. Nevertheless, one may wonder that by introducing a metaphor of memory could approach design process because instead of having a single, although huge, blur line (or space) of search, it could happen instead of different lines (or spaces) of search.

As Samir Emdanat (1999) stated, coupling generative systems and a meta-heuristic research method provides: a compact representation of the design space that can be subjected to systematic explorations to discover feasible design topologies, and an exploration that can be efficiently directed to feasible segments of the design space from which designs that can exhibit highest utility are selected. Other methods and combinations of them have been used in design with shape grammar; the paradigmatic example is of the Christine Shea and John Cagan (1999) work. It was used the simulated annealing process for structural analyses. Other example is combining GA and SG (Rosenman 1996). Outside architectural design, GA have also been tried by mixing metaphors of memory or cultural transmission, which improved GA performance (Spector 1996) in author opinions.

Therefore, SG were selected because they proved to be a solid background in terms of computer representation and analyses, and GA were selected as a computer process of search with the addition of a metaphor of memory because they have proved to be a consistent method of search where productivity is applied. The memory metaphor was added because it is part of a human process of thinking.

SGtools is a local tool that performs under set grammar guidance, a method that can control representation, but the question is if it can be combined with method search strategies. The user defines a grammar by visual means and rules that can be stored and retrieved. A shape in a vocabulary defines a grammar from a group of rules based on the isometric properties of a rectangle, for 2D purposes or of an oblong, for 3D ones. One can play with different grammars in combination that are applied at the same search space. Designs are evaluated visually by one person or more in a group. One can substitute a shape by other shape after being defined a spatial relation with rectangle or oblong shapes. Equivalence shapes can be any shape. This tool seems adequate to represent shapes and to keep in memory information about shapes, and work together with a set of algorithms to represent processes.

Shape Grammar parameters

Generative grammars are made of finite descriptions that explain an infinite set of designs in a known language (Stiny 1980). This formalism is specified by a vocabulary of shapes described as symbols and a group of transformations to generate new arrangements of shapes. As a design production system, it can generate known designs and new ones in the same style. That is the main strength of the generative systems that provide a concise and computable representation of design space. In its basic formal definition, it can be described as: G = [U, L, T, I]. Where U states for a vocabulary of non-terminal shapes, L states for terminal shapes, T states for a set of transformations and I for an initial shape. This production system works with a set of rewrite rules type A->B, when A is found in C the rule is applied under given transformations.

Only Euclidian and affinity transformations are allowed. Rules of addition and subtraction are applied to shapes described in a zero-dimension. Concepts of shape equivalence and color grammar can operate over shapes through rules. The SGtools uses parts of this framework device described earlier.

Research optimization methods

There are not optimal solutions on research methods and the same can be said for designs. Algorithms that support creative design should help to produce surprising designs: randomness is needed. Playing with chance has to be combined with a goal-oriented search. It has to be an effective way of exploring on a searching space. Therefore, what seems to be interesting is a division of labor between a human designer who is equipped to deal with ill-defined problems, and a machine that can formulate a great range of possible designs.

Genetics algorithms

GA have been used for a quite some time in the area of design: a metaphor that simulates some aspects of evolution of nature, by modeling its behavior on a formal machine. These concepts and procedures are being largely applied to architecture descriptions and processes (Testa and al., 2001). John Gero (1999) summarizes the advantages of this method: it can produce unexpected results due to the random search; it can produce complexity, which is important if designer needs it; fitness can change over evolutionary time, which is very important during a design process; and fitness can be humanly evaluated, which supports an effective goal-oriented process. The blind search allows unexpected shapes that can become interesting by human visual evaluation.

Taboo Search

It is also introduced another stochastic method of research that roots from the 70s, which Fred Glover (1996) presented for the first time in 1996. Many others had contributed for its formalization. The use of memory is the main key of this method. It keeps track of local information and simultaneously of the whole process, which Glover calls the dynamic neighborhood search techniques. A general definition can be: a solution of x moves to a solution x1 after n moves, if x1 is in the group of N*(x), and not in the group of (taboo list). Taboo lists can have different qualities.

Framework representation

Processes that are applied here are inspired on those described earlier. A mix of both processes is introduced along with SG to produce design results. The question is; what are the tangible strategies to apply to represent a good approach to human design process. In terms of functionality, it means that a designer can work with any shape under SGtools framework, but he needs to use a programming language to alter the process search strategies, but not variable values. The presented example is very simple.

Process of SG encoding

This example is made for a two-dimensional scenario. The vocabulary is a rectangle of any dimension that is called in SGtools by frame, plus any other shape from the vocabulary of AutoCAD software. This second shape can later be changed through a method of shape equivalence inspired on the concept of Knight (1981). All Euclidian transformations are applied to a rectangle regarding its isometric properties. In this example, rules are four to limit this exploration. There are rules for shapes and operators for the search process that are dependent and listed into genotypes. For set grammars, it was defined a very common group of spatial relations. There are no terminal rules. In formal terms, this is a grammar composed by these parts: Sg = (U, L, T, I). Where, U stands for any possible AutoCAD shape. L stands for a labeled rectangle shape, and I for an initial shape. In figure 1, rules are shown and it is possible equivalent shapes.

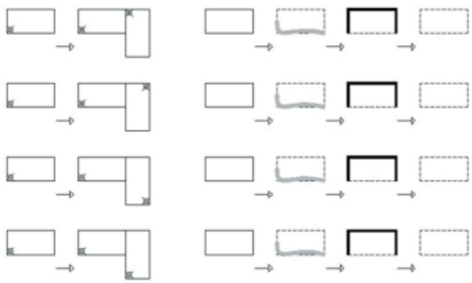

Figure 1 Rules definition and possible shape equivalence

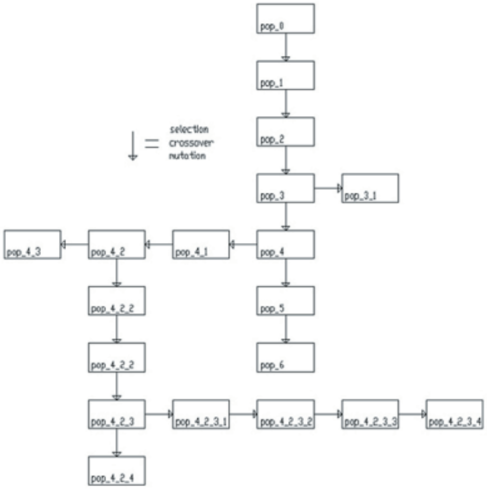

Figure 2 Populations can be stored and retrieved.

Process of GA encoding

A genotype contains information to be manipulated by crossover and mutation operators, while a phenotype is used for selection purposes. For this experiment, parameters are described as follows. Chromosome is represented by a binary format. It is composed of an identification of a shape, an associated SG rule and a number of rule applications. For example a chromosome |111 00 11| has respectively a shape that belongs to a grammar rule and its position in the space, a grammar rule with an associated shape, and three rule applications.

Fitness is given by human visual evaluation. A user should select from a range of possible shape rules, which one should prevail over others. For this example, it was implemented a fitness function with two possible options: selection of horizontal or of vertical lines. One may say that horizontal lines should prevail over the others. Other possible selection decision was a selection by the type of shape grammar rules. These criteria are applied randomly over existent population.

Crossover will work by binary representation and two points crossing. For example: |111 00 01| and |100 01 10 | may be the parents of |111 00 10| and |100 01 01| chromosomes. The correspondence in this case: |111 00 01| means a shape and its position of a range of eight possibilities. The second part represents a shape grammar rule that is associated with the shape. The last part represents the number of rules application that ranges from 0 to 3. The process of pairing selection and mutation are random. In this case only half of the whole population can do crossover processes, where Pc= 0.5.

Mutation will occur by flipping one gene value. For example, |010 00 11| may go to |010 01 11|, which means that the selected shape is changed by a shape of the other rule, but the position in space will be the same. This process is applied to a small percentage of population, Pm < 0.001.

Parameters can be changed for selection and the ways they are applied. In this case, it was applied roulette-wheel selection method. For example, the number of vertical/horizontal lines and rule type can be altered by human evaluation. For the first case, it was applied 80/20 percentages. For the second, the chosen rule is removed by 10% of the total rule population and by 30% for the others rules. These values can be altered during the search. Population size and results are controlled by the user. At any time can be introduced a number of possible elements of a population.

The selection process also includes a pre-defined function that does not admit duplication of a same shape at the same space. A new population is created and kept into a new layer. Only current population is seen, other populations are frozen, but they can be retrieved and new populations can derive from them.

A new population derives from an old one by means of a selection. The user can track different generations and can retrieve them. From an old selected population can derivate a new line of search. There are no limitations to the number of population proposed and evaluated, neither to the lines of direction of search. For example, in figure 2 it is shown different lines of direction that the search

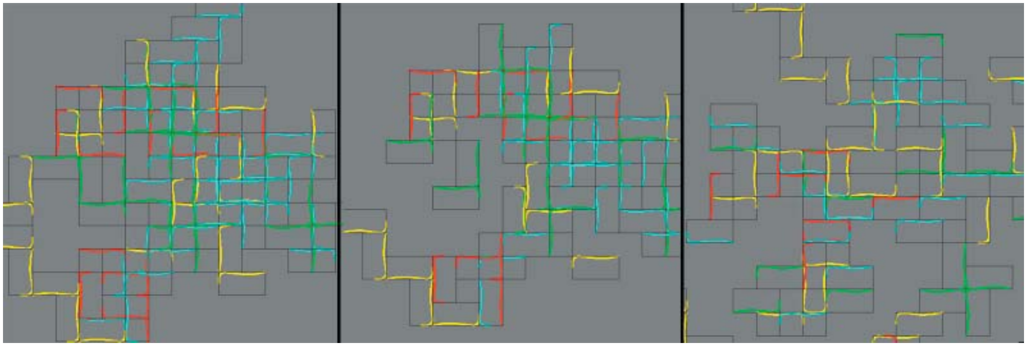

Figure 3 Populations 2, 3 and 9.

can take. Not only a line direction can change, but also it is possible to create a new direction from an old population. It is similar with the process of retrieving an old drawing previously dropped into a litter basket.

Conclusions and further work

One of the goals of this paper was to inquire if a computer implementation could combine metaphors of genetic algorithm process and shape grammars representation to obtain interesting goal-oriented design approach. It was seen that is needed further exploration to prove validity, not only with this simple grammar but also by introducing grammars that are more complex. Memory metaphor seems relevant because older populations could be reevaluated to provide new reformulated ones.

The goal for emergence results in a limited way and can play an important role for design. It is gotten by human visual evaluation over populations. There is not computer emergence though. Populations that embody options are in the universe of

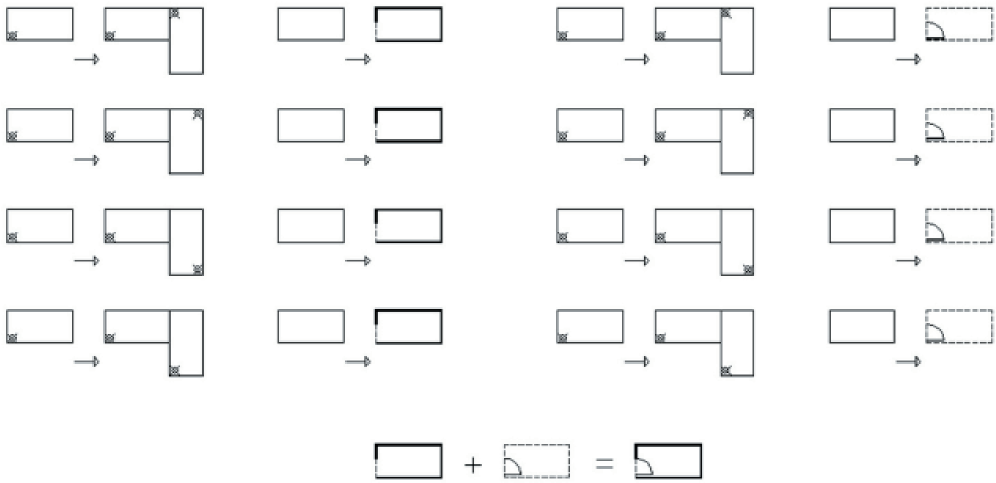

Figure 4 Different rules of different grammars can interact; the holistic relation is given by the user

the grammar but they are difficult to reach by a systematic evaluation. There is also the possibility from existent design shapes being formulated new rules, which augment enormously possible search spaces. During the process of search, the user can create new rules shaped by other rules and shapes. For example, rule_1 is A->tA, and rule_2 is B->tB both are part of a rule_3 C->t1C, where C contains A and B shapes and t1 is different of previous t(s) values but they work all together in a parallel format.

There was other goal: creating a tool that could ”grow” along with the comprehension of the user in terms of design and mainly in terms of tools. This approach is not new (Testa and al., 2001). The process of interrelation between computer and designer, using this application or a different one, could be a good strategy to stronger relations between designers and computers. It was introduced both spatial and visual features to promote and to better control offspring. The populations illustrated in figure 3 could be used to support a detached house for example. There are no constrains of areas spaces or topologic relations between them.

Real tests should be made against real design programs, in terms of early stages. It was not presented any completed design result yet. The same grammar and the same search process could lead to different design results depending on the designer. Even a same designer could reach different solutions in different times. This process only focuses the morphology aspects of design and aspects of emergence will be explored only in terms of designer evaluation. Nevertheless, more work must be done to generalize this framework and improve user interfaces.

An important aspect should be noted. There is not only one linear path of search, even blurred and large, but many. These directions were tracked and the end of the process is decided by the user. He goes as far as he can handle. It seems odd or randomly but it was argued that an introduction of the memory metaphor narrows the space of search. During the process of search, shapes can be changed and tested. By changing shapes, previous population may need to be evaluated.

To conclude, a genetic approach was applied, mixed with concepts that are outside the genetic definition. This will alter results profoundly. It is thought that certain phenomena are artificial regarding those dependent of the goals or purposes of the designer. Results are different for each designer, in opposition to natural laws that are controlled by natural phenomena, which tend to a same result.

Acknowledges

This work was made during my PhD thesis of the School of Architecture and Planning of MIT. It has been supported by the Fundação Calouste Gulbenkian and the Fundação Luso-Americana para o Desenvolvimento, and by the School of Architecture of Universidade Técnica de Lisboa where author is currently teaching. The author also wants to thank the anonymous reviewers for the useful observations and to Jose Duarte for his support.

References

Arnheim, R.: 1997, Visual Thinking, Berkeley: University of California Press.

Brooks, R.: http://www.ai.mit.edu/projects/humanoid-robotics-group/cog/overview.html: May 25 2005.

Caldas, L. and Rocha, J.: 2001, A generative design system applied to Siza´s school of architecture at Oporto, CAADRIA 2001 Proceedings of the Sixth Conference on Computer Aided Architectural Design Research in Asia. Sydney 19-21 April, pp. 253-264.

Emdanat, S., Vakalo, E.G. and Birmingham, W.: 1999, Solving form-making problems using shape algebras and constraint satisfaction, Architectural Computing from Turing to 2000,

eCAADe Conference Proceedings, Liverpool (UK) 15-17 September, pp. 620-625.
Hamza, K, Mahmoud, H., Saitou, K.: 2003, Design optimization of N-shaped roof trusses using reactive taboo search, Applied Soft Computing 3, Elsevier B.V. pp. 221–235.
Holanda, F. 1955, Diálogos de Roma: da pintura antiga, Lisboa : Sá da Costa, translated into English by Charles Holroyd, Michael Angelo Buonarroti, London, Duckworth and co., 1911.
Gero, John S.: 1999, Constructive memory in design thinking, in Architectural Science Review 42:3-5.
Glover, F.: 1996, Tabu search and adaptive memory programming-advances, applications and challenges, in Interfaces in Computer Science and Operations Research. Barr, Helgason and Kennington, (eds), Kluwer Academic Publishers.
Knight, W. T.: 1981, Languages of designs: from known to new, Environment and Planning B: Planning and Design, vol. 8, pp. 213-238.
Knight, W. T.: 2003, Either/Or -> And, Environment and Planning B: Planning and Design 2003, vol. 30, pp. 327-338.
Mitchell, William J.: 1990, A New Agenda for Computer-Aided Design, The Electronic Design Studio: Architectural Knowledge and Media in the Computer Era, CAAD Futures '89 Conference Proceedings, Cambridge (Massachusetts / USA), pp. 1-16.
Romão, L.: 2005, SGtools: A computer tool for exploring designs with set grammars, CAAD Futures '05 Conference Proceedings, Vienna, Austria, 20-22 June 2005, pp. 53-62.
Rosenman, M.A.: 1996, The generation of form using an evolutionary approach, J.S. Gero and F. Sudweeks (eds), Artificial Intelligence in Design Ì96, pp .643-662.
Shea, K. and Cagan, J.: 1999, The design of novel roof trusses with shape annealing: assessing the ability of a computational method in aiding structural designers with varying design intent, Design Studies, 20 (1) pp. 3-23.
Spector, L. and Luke S.: 1996. Cultural transmission of information in genetic programming, in Genetic Programming 1996: Proceedings of the First Annual Conference, J. Koza et al, (eds), Cambridge: MIT Pres, pp. 200-208.
Stiny, G.: 1980, Introduction to shape grammar and shape formalism. Environment and Planning B: Planning and Design, vol. 7, pp. 343-51.
Testa P, O'Reilly, Weiser D., Ross I.: 2001, Emergent Design: a crosscutting research program and design curriculum integrating architecture and artificial intelligence, Environment and Planning B: Planning and Design, V. 28:(4) July, pp. 481-498.