Issues in Building a Shape Grammar Interpreter for Customizing Mass Housing

JOSÉ PINTO DUARTE
Massachusetts Institute of Technology

1. INTRODUCTION

This paper was developed in the context of a research that is concerned with the customisation of mass housing. This research is based on three arguments. First, shape grammars can provide the technical apparatus to make the rules of a designer's design language explicit. Second, a computer program encoding such a grammar—an interpreter of the language — would allow the designer to use his design rules more effectively. Finally, shape grammars and their interpreters provide a suitable tool for customising the design of mass housing. The first argument was settled with the presentation of a grammar for the language of houses designed by the Portuguese architect Álvaro Siza at Malagueira (Duarte, 1999), a 1,200 houses development. The grammar accounts for the generation of the existing designs used in its development (Figure 1) and it can also be used to generate new designs in the language. A new design (Figure 2, t5) by the author of the grammar—the second author—was placed among existing designs and Siza--the first author—was not able to distinguish it from his own designs. In addition, the grammar was used by other designers to design houses for specific clients in a set of experiments. These designers were not knowledgeable of the architectural and the cultural contexts in which the designs and the grammar were developed. The result also was, to a variable extent, a set of customised designs in the language (Figure 3). The second argument is discussed in this paper, and the third argument will be the subject of future research.

For the second argument to be settled one needs to build an interpreter that makes possible an effective use of the Malagueira grammar. A possible interpreter would randomly generate houses in the grammar. The designer would then populate the housing development with such houses. This use, however, is not the most effective as one could not guarantee that the generated houses would match the prospective users' needs. Mitchell (1990) illustrated the need for shape grammars by comparing a designer's attempt to design without one to Gulliver's Lilliputans attempt to write books by randomly combining words. A grammar guarantees

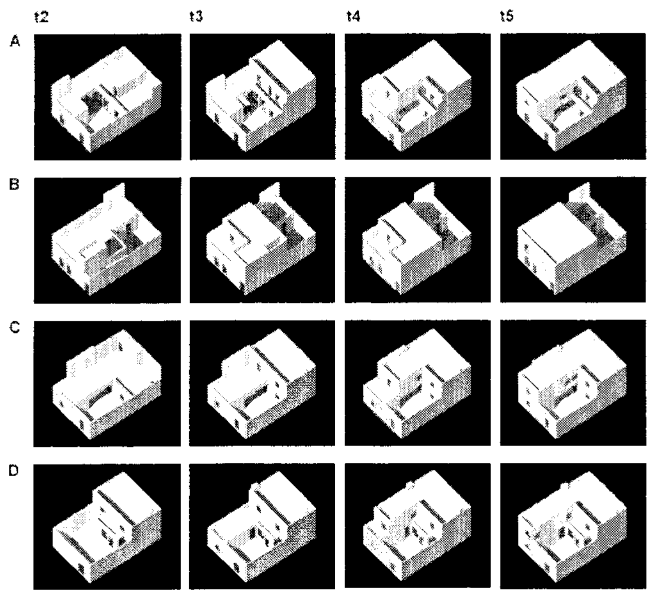

Figure 1 – Part of the corpus of existing designs by Siza, the first author. Rows depict the two to five-bedroom variations of housetypes A, B, C, and D.

that English sentences will be generated, but one problem remains, how can one assure that the grammatically correct sentences will say what we are trying to convey? To expect this is as hopeless as expecting the random concatenation of words to generate English sentences. So, there are two parts to the problem, one is concerned with the generation of legal designs—designs in the language, the other with the generation of suitable designs—designs that match requirements given at the outset. The power of an interpreter would, thus, be considerably higher, if one were able to generate both. The designer could then use the interpreter as a tool in the dialogue with the client. It is this sort of interpreter that we intend to build. This discussion highlights the difficulties involved and aims at describing how different state of art techniques can help in that endeavor.

The problem of building grammars and interpreters that generate suitable designs is foreign to previous shape for

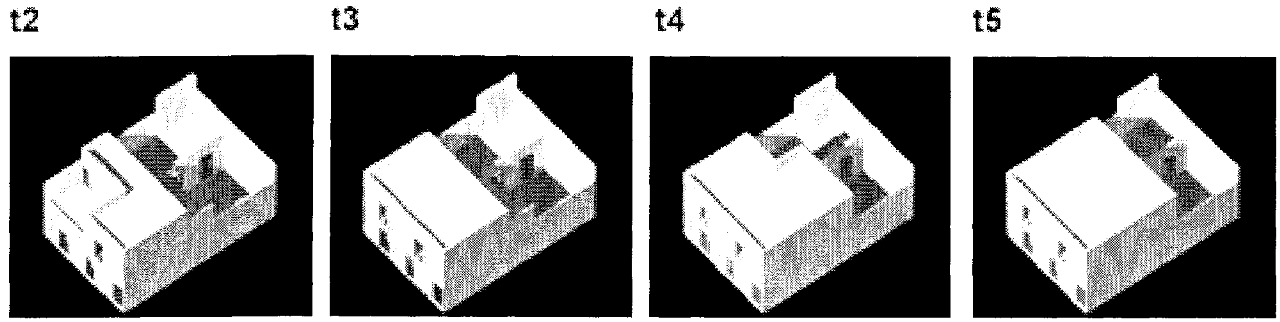

Figure 2 - Novel designs by the author of the grammar, the second author.

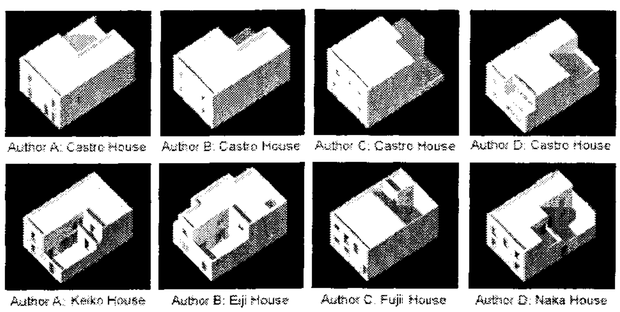

Figure 3 - Designs by users of the grammar, the third authors, for specific clients. Top row: houses by different authors for the same client. Bottom row: houses by different authors for different clients.

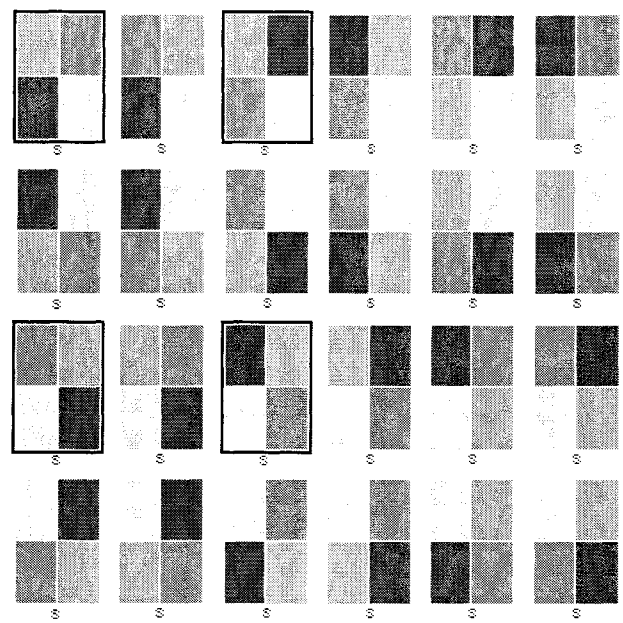

Figure 4 - The first step in the grammar is the dissection of the lot into three or four functional zones, patio, living, sleeping, and service zones to form a basic pattern. The picture shows 24 out of the 192 possible patterns. Different zones are represented with different shades of gray. Patterns used by Siza are highlighted with a bold frame. The letter "S" indicates the location of the street.

instance. Heisserman's Genesis interpreter (1991) when loaded with these grammars was able to randomly generate legal houses, but it lacked a mechanism that tied the final design to user-specified requirements. Some engineering grammars and their interpreters, however, have been developed with the goal of generating optimised solutions for given design contexts. This was the case, for instance, of Reddy and Cagan's (1995) truss design grammar, and of Shea's essays of discrete structures (1997). The generation of designs that match given contexts requires some sort of control mechanism to act upon the generation. The development of such a mechanism requires one to address three different but related issues concerning the purpose, the nature, and the timing of the mechanism.

2. CONTROL MECHANISM: PURPOSE

As for the purpose, one can identify three intermediate goals of increasing demand, implying the generation of legal, suitable, and optimal designs. The achievement of the first goal by a shape grammar alone depends very much on how much knowledge is encoded in the grammar, which, in turn, depends on the configuration of the space of legal solutions. If this space is continuous and contained, it is feasible to develop the grammar to encode such a space. On the other hand, if the space is discontinuous and sparse, it might be difficult to develop a grammar that only generates legal designs because its exact contour is unknown or because it might require too many rules.

The Malagueira grammar initially fell into the first case, as it only allowed the generation of basic patterns identical to the five that Siza designed. This seemed to be too restrictive, however, and so the grammar was later enlarged to include all the possible 192 basic patterns that can be inferred from Siza's dissecting rules (Figure 4). This enlargement seemed justified because different patterns could be needed to generate houses that satisfied new users with different needs. This was, in fact demonstrated by a set of experiments in which design subjects were given the restrictive version of the grammar and asked to use it in the generation of houses for specific clients with a life-style different from the initial Malagueira dwellers. In a first set of experiments, the subjects were not allowed to change the grammar rules, whereas in a second they were allowed to do so. Results of the first set of experiments show that without changing the rules it was not possible to satisfy the clients' requirements in some of the cases. Results of the second set of experiments also showed that most of the changes required in the grammar were just to allow the generation of basic patterns different from those initially used by Siza. The enlargement of the range of patterns raised the issue of whether the new patterns corresponded to designs in the language. During regular conversations of the author of the grammar with Siza to discuss the evolving grammar, Siza's intuition was that the 192 patterns seemed to be in the grammar from a formal viewpoint. Siza also pointed out that the lifestyles of the Malagueira dwellers evolved since the first house was designed twenty years ago. Current dwellers tend to have higher social status and a more urban lifestyle. Such intuitions were confirmed by the experimental results in which houses with new underlying patterns satisfied the needs of dwellers with different lifestyles and were still perceived as being in the language. Therefore, both for formal and functional aspects, it was appropriate to enlarge the space of design solutions.

Such an enlargement, however, turns the problem of generating a legal solution, into that of generating a suitable solution. Once it is intuitively agreed that the 192 patterns lead to designs in the language, all the solutions generated by the grammar are legal (provided that similar intuitions validate possible design configurations at later stages of the derivation). However, once one ties the generation of a solution to constraints given at the outset, all the solutions in the grammar will have to be suitable. Therefore, a solution will be legal (and suitable) as long as it satisfies the design requirements. But should it be optimal? Given the nature of the problem, in which some criteria might be conflicting, or even ill defined, it seems difficult to accept the existence of a global optimum. In fact, one of the major issues that the experimental subjects had to face was to design a solution for an over-constrained problem. The task became one of choosing which criteria to satisfy.

It was not uncommon for the subjects to ask for a new meeting with the client to re-evaluate the goals. Moreover, results also showed that there might be several satisfactory solutions, each representing different trade-offs among the several criteria. Compare, for instance, the different houses generated for the same client by different designers (Figure 3). In such circumstances the problem is not amenable for representation as an optimisation problem. Therefore, one should not be concerned with the search for a global optimum, but rather with providing the designer and the client with several satisfactory solutions from which to choose.

3. CONTROL MECHANISM: NATURE

As for the nature of the control mechanism, the second issue, two approaches are possible. In the first, this mechanism is embedded in the grammar. In this case, it can be included in the rules through control conditions involving labels. In the Malagueira grammar, control conditions restrict the allocation of a room through dissection to finding rooms with appropriate function labels. For instance, a transitional space is allocated if a service room with an adjacent living room is found. Nevertheless, experience suggests that this approach alone is not enough to cover all the details in a housing program required to guarantee the generation of an appropriate design. For instance, it is necessary to keep track of the lists of desired and allocated spaces to prevent the allocation of non-desired spaces. Or, in more elaborate approaches, it might be required to constrain rule application to the performance of the design from certain viewpoints such comfort, security, or cost. Labels are not good for book keeping or for dealing with such performance criteria. Their use for these purposes is cumbersome and not very intuitive.

An alternative approach considers the development of an additional, external mechanism. Such a mechanism can be a description grammar or a directed stochastic algorithm. The use of one does not necessarily exclude the use of other, and it can even be shown that they are equivalent by writing stochastic algorithms in the form of description grammars (Knight, in progress). In our discussion, however, we will maintain the view that distinguishes between them.

Directed stochastic search

In the directed stochastic alternative, the mechanism would check the evolving design for fitness as to guarantee that appropriate designs were generated. In this approach, the generative grammar defines the space of design solutions and the evaluation mechanism looks for an appropriate solution within that space. Such was the approach proposed by Cagan and Mitchell (1993) with their Shape Annealing algorithm, linking a grammar to a simulated annealing technique, and then followed by Reddy and Cagan (1995), by Shea (1997), by Shea and Cagan (1997) and by Shea et al (1997). Shape annealing is a variation of the more general simulated annealing method. The main difference between shape annealing and the more general method is that in shape annealing the changes that are made to evolving design in an attempt to bring it closer to the optimum are more informed because they follow shape rules. This eliminates non-feasible design configurations. Simulated annealing is an example of a heuristic method for optimisation. The formulation of an optimisation problem requires the proper identification of an objective function with n variables, and a set of constraints that bound the values that those variables can take. There are three major difficulties in formulating the problem of generating a customised house as an optimisation problem. The first difficulty is defining an objective function due to the subjectivity of some of the criteria. The second is that the problem is over-constrained, that means that there is no solution in the space defined by the variables because the constraints imposed on the variables are too restrictive. Finally, the third difficulty is that the problem is ill-defined, that is, the goal is not clear and not sufficient criteria is given to specify what constitutes an optimal solution.

Description grammars

The concept of description grammar was developed by Stiny (1981) to account for features of designs not covered by shape grammars. A shape grammar specifies how designs can be generated. A description grammar describes the design in terms of other features considered relevant according to some criteria of interest. The relation between the shape grammars and description grammars is such that for each shape rule there is a corresponding description rule. plus an additional description rule corresponding to the initial shape. As the grammar rules are applied to the evolving design, the corresponding description rules are applied to the evolving description. Thus, as the generation of the design evolves, the description of the design is constructed. Stiny further suggests that the description grammar can be considered a grammar of another language and that it would be possible to translate back and forth between the two languages. The hypothesis that one can immediately raise is whether one could use such a translation mechanism to obtain the design from the description. This translation process, however, is not straightforward, as one needs to overcome three problems.

The first problem is that of fixing the contents of the description, that is, which categories to include. In the Malagueira grammar, we are faced with difficulties raised by the fact that neither the documents gathered in Siza's office, nor the interviews with him or his collaborators provided any written descriptions of the existing designs. Therefore we will have to use another strategy to determine them. Choosing the system of categories is crucial, as they will determine what questions to ask the user and how to derive the design. Assuming a solution to this problem, we have to discover how to arrive at the design from the description, the second problem.

A possible solution is to obtain the design directly from the description. In this solution, one uses the grammar to generate all the possible designs and all the matching descriptions, and then browses this catalogue in search of the design or designs that match the given description. However, if the space of design solutions is very large or infinite, the process becomes ineffective or even impossible. An alternative solution is to obtain the design from the description through an indirect process that can be summarised as follows. First, move from the given description to the initial description by applying description rules in reverse and store in memory what these rules were as well as the sequence of their application. Then obtain the initial shape from the initial description. When the last description rule is applied, find the corresponding grammar rule and apply it to the initial shape; proceed recursively with grammar rule applications until the design is derived. This alternative also raises some problems, as at each stage of this backward-chaining process there might be the choice of applying more than one description rule. How does one decide which one to choose? Shall one try all, evolving different descriptions in parallel? But then, what if the number of possible description processes grows exponentially? Which among them does one decide to pursue, and which are left out? We will need some sort of search strategy, which means that an external mechanism, outside of the grammar might be needed.

The discussion above assumes that one is able to get the description of the desired design. But what if we cannot get the full description in first place? We are faced with the third problem. In design, the problem is often ill defined and part of the designer's task is to clarify what the problem is. In fact, the user might not know enough about his desired house to provide a full description. Two possibilities lie ahead. In the first, one needs a process to arrive at a full description from the given incomplete description. In fact, there might be several full descriptions that can be derived, meaning that there might be several designs that match the incomplete description. In which case one needs to decide which full description to pick up to start the process of searching for a matching design or the exponential aspect of the problem increases. In the second possibility one starts the backward, inference process from the incomplete description to arrive at the corresponding incomplete design, and then finishes the design. Here, one may encounter problems if the given incomplete description contains information corresponding to later stages of the design derivation and is missing information of earlier stages, making it difficult to perform the backward-inference process.

4. CONTROL MECHANISM: TIMING OF THE EVALUATION

The third problem concerns the timing of the evaluation, that is, when in the generation should the control mechanism actuate. There are two possibilities. In the first, control happens as the generation evolves. This was the case of the shape-annealing algorithm proposed by Cagan and Mitchell for optimising designs with a grammar of half-hexagons. The design goal was to fill in rectangles with half-hexagons put together in accordance with the rules defined by the grammar. The algorithm would randomly select a rule and then evaluate the design after its application and compare it to the design before rule application. If it were better, the rule would be applied. If it were worse, there was still a probability for the new design to be selected. This probability, higher at the beginning of the generation, would lower as the design approached the end. In the second possibility control takes place at the end of the generation.

The framework proposed by Reddy and Cagan for truss design, and then taken further by Shea falls into this category. In their grammars the process starts with a complete design that is the simplest answer to the structural problem one wants to solve. This design represents a legal configuration but not necessarily a feasible or functional solution, as some constraints might not be respected. Then, shape grammar rules are applied to transform the design. Evaluation then takes place to assess the transformation in much the same way as in Cagan and Mitchell's initial algorithm. The assessment of the complete design is effective in the case of the truss grammars, because the grammar is very simple.

In the case of the Malagueira grammar such a procedure would not be cost and time effective because of the complexity of the design problem and the grammar. Moreover, it would be difficult to develop transformation rules to apply to a complete design. Therefore, the problem calls for a different sort of approach. Preliminary studies suggest the possibility of applying transformation rules to intermediate stages of the design. For instance, transformation rules can be developed that apply to the basic patterns to improve area distribution among the different functional zones taken into consideration user preferences. Transformation rules can also be applied to the openings to improve natural light. Thus, a possible approach might to consider different intermediate stages or steps in the derivation of a design and to do some sort of "optimisation" at each step, focusing on appropriate goals and using appropriate techniques. Such an approach corresponds to using techniques known in artificial intelligence as problem reduction and means-end analysis (Winston, 1993).

5. CONCLUSIONS

Building an interpreter for a housing grammar that solves the problem of generating customized designs requires an effective control of the generation to guarantee that the solution matches the specified requirements. In choosing which control mechanism to use one needs to address issues related to the purpose, the nature, and the timing of such a mechanism. The purpose implies to decide whether legal, suitable, and optimal designs ought to be generated. Given that the house customization problem is often over-constrained or ill-defined, it is often impossible to frame it as an optimization problem.

On the other hand, the generation of legal designs is not sufficient. In customizing mass housing the goal is to generate suitable designs. As far as the nature of the control mechanism is concerned, the basic issue is to decide whether the control is internal or external to the shape grammar. Internal mechanisms such as labels are limited in their control ability because it is often cumbersome to use from the description, and to find a way to deal with incomplete descriptions.

Finally, given the complexity of the problem of designing a house, the problem needs to be decomposed into sub-problems and different control mechanisms that address such problems and actuate at different stages of the generation might need to be used. Current research is concerned with the development of an interpreter that answers all of these.

REFERENCES

Cagan J., and Mitchell W. J., "Optimally Directed Shape Generation by Shape Annealing", in *Environment and Planning B* 20 (1993): 5-12.

Duarte, J. P., "Democratized Architecture: Grammars and Computers for Siza's Mass Housing" in Proceedings of the International Conference on "Enhancement of Computational Methods in Engineering and Science", (Elsevier: 1999): 729-739.

Flemming U. "More than the Sum of the Parts: The Grammar of Queen Anne Houses", in *Environment and Planning B*, 17 (1987): 323-350.

Heisserman, J., 1991, "Generative Geometric Design and Boundary Solid Grammars", Ph.D. dissertation, Carnegie-Mellon University, Pittsburgh.

Knight (in progress). "Computing with Emergent Shapes", Massachusetts Institute of Technology.

Koning H. and Eizenberg J., "The Language of the Prairie: Frank Lloyd Wright's Prairie Houses", in *Environment and Planning B*, 8 (1981): 295-323.

Mitchell W. J., "The logic of Architecture", (MIT Press, Cambridge: 1990).

Reddy G. and Cagan J., "An improved Shape Annealing Algorithm for Truss Topology Generation", in Journal of Mechanical Engineering, June, Vol. 117 (1995): 315-321.

Shea K., 1997, " Essays of Discrete Structures: Purposeful Design of Grammatical Structures by Directed Stochastic Search", Ph.D. thesis, Department of Mechanical Engineering, Carnegie-Mellon University.

Shea, K. and J. Cagan, "Innovative Dome Design: Applying Geodesic Patterns with Shape Annealing", Artificial Intelligence in Engineering Design, Analysis and Manufacturing, 11 (1997): 379-394.

Shea, K., J. Cagan, and S. J. Fenves, "A Shape Annealing Approach to Optimal Truss Design with Dynamic Grouping Members", ASME Journal of Mechanical Design, 119 (3) (1997): 388-394

Stiny G., "A Note on the Description of Designs", in *Environment and Planning B* 8, pages (1981): 257-267.

Winston, P., "Artificial Intelligence", (Addison Wesley: 1993).