ARTICLE IN PRESS
AUTCON-01585; No of Pages 9
Automation in Construction xxx (2013) xxx–xxx
Contents lists available at SciVerse ScienceDirect
Automation in Construction
journal homepage: www.elsevier.com/locate/autcon

A general indirect representation for optimization of generative design systems by genetic algorithms: Application to a shape grammar-based design system

Vasco Granadeiro [a,d,*], Luis Pina [b], José P. Duarte [c,d], João R. Correia [d], Vítor M.S. Leal [e]

[a] IN+, Instituto Superior Técnico, Technical University of Lisbon, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
[b] Pina Robotic Solutions, Rua Eduardo Coelho 43-3 Esq., 1200-165 Lisboa, Portugal
[c] CIAUD, Faculty of Architecture, Technical University of Lisbon, Rua Sá Nogueira, 1349-055 Lisboa, Portugal
[d] Instituto Superior Técnico/ICIST, Technical University of Lisbon, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
[e] Institute of Mechanical Engineering, Faculty of Engineering of the University of Porto (IDMEC-FEUP), Rua Dr. Roberto Frias, 4200-465 Porto, Portugal

ARTICLE INFO

Article history:
Accepted 18 May 2013
Available online xxxx

Keywords:
Generative design system
Shape grammars
Parametric design
Optimization
Genetic algorithm
Representation

ABSTRACT

Generative design systems coupled with objective functions can be efficiently explored through the use of stochastic optimization algorithms, such as genetic algorithms. The first step in implementing genetic algorithms is to define a representation, that is, the data structure representative of the genotype space and its mathematical relation to the data of the phenotype space — the variables of the real problem. This can be a hard task, particularly if the design system contains dependency between variables. This paper presents a general representation, which enables the use of standard variation operators, allows defining both continuous and discrete variables from a single type of gene and is easily adaptable to different problems, with a larger or smaller number of variables. This representation was created to solve the representation problem in the design system for Frank Lloyd Wright's prairie houses, a shape grammar that was converted into a parametric design system.

1. Introduction

A generative design system is a model composed of a set of computational rules that are applied to generate alternative designs [1–3]. In addition to enabling the desired design variety, rules encode constraints to create only the intended output. Therefore, through a well-designed system of rules, generative design systems have the capability of maintaining stylistic coherence and design identity while generating diverse designs.

The advantage of having alternative designs is to allow choosing the one that better fits a certain context or achieves a defined objective. Frequently, this can be translated into an objective function, linked to the design system. Within the flexibility of the design system, variables (in the form of rule applications and/or parameters) can be edited to improve the value of the objective function, looking for the best performing design. However, for a large number of variables, search algorithms are a much more efficient method to explore the solution space. Of course, this requires design systems to be programmed in numerical computing languages.

Exhaustive search is the ideal method for small solution spaces. Exact optimization algorithms, such as linear programming, branch-and-bound and dynamic programming, are very proficient in finding the best solution in convex solution spaces. However, in general, generative design systems involve very large and non-convex solution spaces. Therefore, stochastic optimization algorithms are the most appropriate search method, even though they do not guarantee finding the global optimal solution. Among them, genetic algorithms have been proven to find high quality solutions in large solution spaces and in reasonable time periods [4].

The first step in solving an optimization problem with genetic algorithms is to define an adequate representation and a corresponding genotype–phenotype mapping. The real-world problem involves variables. Genetic algorithms work with genes, the corresponding entities for the variables. The representation is the definition of the genes and the genotype–phenotype mapping is the mathematical relation between genes and variables. The informal term “genetification” is sometimes used as a mnemonic for the task of creating the representation and the genotype–phenotype mapping. The two main requirements for a representation are encoding all the possible solutions of the problem and enabling the application of the variation operators to them (crossover and mutation).

* Corresponding author at: IN+, Instituto Superior Técnico, Technical University of Lisbon, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal. Tel.: +351 918506061.
E-mail addresses: vascogranadeiro@ist.utl.pt, vgrana@media.mit.edu (V. Granadeiro), luismppina@gmail.com (L. Pina), jduarte@fa.utl.pt (J.P. Duarte), jcorreia@civil.ist.utl.pt (J.R. Correia), vleal@fe.up.pt (V.M.S. Leal).

0926-5805/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.autcon.2013.05.012

Although some bibliography focuses specifically on this subject, crafting or choosing an adequate representation is not an easy task, since it is still the result of intuitive analysis and the latter depends on some experience in the field [5].

The motivation for the research presented in this paper was the problem of finding an adequate representation for the design system for Frank Lloyd Wright's prairie houses, with its optimization in foresight [6,7]. The objective function for the design optimization problem, provided that it involves a non-convex solution space, is independent from the representation and its requirements. Several objective functions are possible, ranging from simple geometric features – like footprint area, external surface, volume, etc. – to more complex performance measures, like energy consumption [8], daylight illuminance [9], acoustic performance [10], structural fitness [11], and so on.

Although coupling design systems and genetic algorithms is not a novel approach, this specific study entailed some singularities. The design system for Frank Lloyd Wright's prairie houses is a shape grammar. Although a conversion method was outlined in [7], for converting the grammar (a non-numerical design system) into a parametric design system (a numerical design system), the latter, among other characteristics, contains dependency between variables. Consequences of this characteristic restrict the use of the most obvious representation, the direct representation [5], unless customized, problem-specific, variation operators are crafted. However, the objective of this study was solving not only this one problem but also to create a general representation and corresponding genotype–phenotype mapping, which could be implemented on other design systems with similar characteristics. As such, an essential feature for this representation was to use standard variation operators [4], such as those included in common commercial optimization tools.

The literature already includes some examples of coupling shape grammars with genetic algorithms, but in none the representation was sufficiently explicit or generic to enable its application to this design system. In [12], a direct representation was used and variation operators were customized to the problem. In [13], the representation used is not explicit. Nevertheless, it does not cover all the variables of the design system, since some variables remain constant during the optimization runtime, due to limitations of the software used. This means that the optimization process, in each run, is not exploring all the solution spaces but only part of it. In [14], tree data structures are used in the representation, each tree being formed by a different number of nodes. This representation was crafted for a shape grammar composed of one rule only, which basically divides a shape into two, and this can be repeated indefinitely. Although interesting, this representation is not adequate for Frank Lloyd Wright's design system, due to large differences between the two grammars.

The design system presented in [15] is not a shape grammar, it is a parametric design system, but its constrained structure, in which “each variable is associated with explicit upper and lower limits”, resembles the structure of the converted Frank Lloyd Wright's design system, presented in [7]. In addition, this design system also involves defining continuous and discrete variables to generate designs. A direct representation was used but, in this case, it was possible to apply a standard crossover and a standard mutation (however, not all standard variation operators could be applied). An analogous representation could be crafted for Frank Lloyd Wright's design system, although this would not be generic, as intended. The representation proposed here can be seen as a generalization of this one.

The remainder of the paper is organized in the following way. Section 2 presents the grammar for Frank Lloyd Wright's prairie houses and its conversion to a parametric design system. Section 3 explains the problem that the representation must solve. Section 4 presents and discusses the proposed representation. Section 5 validates the proposed representation, by solving the identified problem and successfully being used in an optimization problem. Section 6 draws the main conclusions from this study.

2. Grammar for Frank Lloyd Wright's prairie houses

Shape grammars comprehend a formalism defined by sets of shape transformation rules that apply in sequence to generate a design from an initial shape [16]. They have been used in different areas of design, not only to encode the rules underlying a set of existing designs but also to create new design systems. Due to dealing with shape transformations, the potential of this formalism rests in its ability to encode certain topological variations that are difficult to encode in other generative design systems. The downside of using this formalism is that although some shape grammar interpreter shells exist [3], they only allow developing simple computer implementations.

The grammar for Frank Lloyd Wright's prairie houses is a powerful generator of diverse designs. Each house is formed by a number of blocks, the main compositional elements of the design system. The grammar rules determine the number of blocks, their shape, size, location and function. The original grammar for Frank Lloyd Wright's prairie houses is fully described in the paper by Koning and Eizenberg [17]. Continuing the research presented in [7], a simplified version of the original grammar, developed there, was also used in the research presented here. Fig. 1 shows the first nine rules of this new grammar — to create the fireplace (rule 1), to create the core unit, composed of the living zone (rules 2 to 5) and the service zone (rules 6 and 7), and to add the obligatory extensions (rules 8 and 9). Depicting the remaining fifty-one rules is not necessary for the demonstrations in this paper. For a complete depiction and description of this new design system, readers should refer to the paper by Granadeiro et al. [7].

Both the original and this new grammar for Frank Lloyd Wright's prairie houses belong to a subcategory of shape grammars, named *set* grammars [18]. In shape grammars, subshape recognition and shape emergence [19] are used to detect shapes, and rules can be applied an infinite number of times. Set grammars are a simpler form of shape grammars, more constrained, in which subshape recognition, shape emergence and iterative rule application are not freely used. The conversion method described in [7] and abridged next is suitable for set grammars. Other examples of set grammars are the grammar for Palladio's villas [20], the grammar for Queen Anne houses [21] and the grammar for Siza's houses at Malagueira [22].

2.1. Conversion of the grammar into a parametric design system

A parametric design system consists of a geometric model that encodes formal features, common to the designs of its solution space. The formal features are the topological relations between the various parts of the designs and their dimensional variations. These relations are defined as a set of equations with variables (parameters, hence the term “parametric”) that can take values within defined ranges [23]. In this manner, the model is controlled by equations and alternative designs are obtained just by altering the parameters in the equations. Relations can be numerous and complex, involving several equations and variables, but, unlike in shape grammars, they always involve symbolic information, including numbers, and, therefore, there are numerous platforms that can be used for programming parametric design systems. Parametric design has been used in many areas, such as architecture [24], building construction [25], mechanical products [26], aircrafts [27] and even to model human bodies [28].

As mentioned above, the new grammar for Frank Lloyd Wright's prairie houses was already converted into a parametric design system [7]. To accomplish this, the shape transformation rules of the former were translated into equations and variables, using *variants programming*, a parametric design technique [23]. This technique consists basically in using procedural programming to develop design models. The program

Please cite this article as: V. Granadeiro, et al., A general indirect representation for optimization of generative design systems by genetic algorithms: Application to a shape..., Automation in Construction (2013), http://dx.doi.org/10.1016/j.autcon.2013.05.012