Artificial Intelligence for Engineering Design, Analysis and Manufacturing

cambridge.org/aie

AI EDAM special issue: advances in implemented shape grammars: solutions and applications

Sara Eloy[1], Pieter Pauwels[2] and Athanassios Economou[3]

[1]Department of Architecture and Urbanism, Instituto Universitário de Lisboa (ISCTE-IUL), ISTAR-IUL, Lisbon, Portugal; [2]Department of Architecture and Urban Planning, Ghent University, Ghent, Belgium and [3]School of Architecture, College of Design, Georgia Institute of Technology, Georgia, USA

Guest Editorial Promises Of Shape Grammars

Cite this article: Eloy S, Pauwels P, Economou A (2018). AI EDAM special issue: advances in implemented shape grammars: solutions and applications. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **32**, 131–137. https://doi.org/10.1017/S0890060417000634

Key words:
shape grammar; implementation; interpreters; technical design; interfaces; use cases

Author for correspondence: Sara Eloy, E-mail: sara.eloy@iscte.pt

Abstract

This paper introduces the special issue "Advances in Implemented Shape Grammars: Solutions and Applications" and frames the topic of computer implementations of shape grammars, both with a theoretical and an applied focus. This special issue focuses on the current state of the art regarding computer implementations of shape grammars and brings a discussion about how those systems can evolve in the coming years so that they can be used in real life design scenarios. This paper presents a brief state of the art of shape grammars implementation and an overview of the papers included in the current special issue categorized under technical design, interpreters and interface design, and uses cases. The paper ends with a comprehensive outlook into the future of shape grammars implementations.

Introduction

This special issue frames the topic of computer implementations of shape grammars, both with a theoretical and an applied focus. Articles in this issue display the state of the art of these computer implementations, as well as their applicability in real design scenarios. The development of this special issue was combined with a special workshop "Advances in Shape Grammars: Implemented Shape Grammars", chaired by Pieter Pauwels and Sara Eloy and held in Evanston, Chicago preceding the Design Cognition and Computation 2016 Conference (DCC2016).

During the DCC workshop (Pauwels & Eloy, 2016), the 23 participants had the opportunity to discuss the recent developments in shape grammar research. The aims of the workshop were: (i) to enable hands-on demonstrations, (ii) to receive on-topic feedback and discussions, (iii) to discuss recent developments, (iv) and to discuss on three topics: technical design, interface design, and use cases. These last three topics were also the main topics that were planned to be addressed in the AIEDAM special issue. Terry Knight and Thanos Economou each gave a brief talk in the beginning followed by the hands-on demonstrations provided by Thomas Grasl with GRAPE and Andrew Li with The Interpreter Project during which they showed the potentials and drawbacks of both interpreters. Presentations were done by Hau Hing Chau "Exploring lattice structures in shape grammar implementations", and Luca Zimmerman "A generative design framework integrating spatial grammars, simulation and optimization", allowed to discuss technical advancements. Sara Garcia with "A multipurpose chair grammar implementation", and Nikolaos Vlavianos with "Shape grammars Augmented Reality (SGr): A novel rule-based method for designing with eyes" brought the use of shape grammars in a practical context.

The starting point of this special issue is the current state of the art regarding computer implementations of shape grammars and a discussion about how those systems can evolve in the coming years so that they can be used in real-life design scenarios. Shape grammars have been researched for more than 40 years now reaching several areas of design as architecture, engineering, and product design. Besides the intellectual challenge involved in the development and use of shape grammars, the potential they would have in addressing real-life design problems is enormous. Several shape grammar implementations have been developed the last years but a true impact in the design methods is still to be proven possible and illustrated. To be able to show shape grammars to industry partners, the grammarians need to implement them in working software prototypes that fully demonstrate their potentials and their generative power. Besides the focus on pure computer implementations of shape grammars, this special issue also called for contributions regarding real design scenario applications. Several recent computer implementations use parametric design tools to generate design alternatives. Nevertheless, most of these implementations realize only a part of the potentials that were originally attributed to shape grammars, thereby leaving some important features as emergence, interactivity, and ambiguity behind in favor of other features. Considering this

CAMBRIDGE
UNIVERSITY PRESS

https://doi.org/10.1017/S0890060417000634 Published online by Cambridge University Press

myriad of implementation approaches, computer-implemented shape grammar interpreters might be usable for diverse application scenarios, ranging from shape grammars for personalized customization for mass-housing to flexible, on-the-spot design grammars that are able to evolve with the design process of the designer.

With the above two main tracks (computer implementations and real-world use cases), this special issue aims at: (1) addressing current shape grammar design computer tools and (2) discussing future paths of implementations towards the real use case scenarios. In the next subsections, we present a brief state-of-the-art overview (the section "State of the art"), to be complemented with the state-of-the-art review of the articles in this special issue. In the section "Advances in implemented shape grammars", we present a short summary of the articles that are included in the current special issue, after which we give a comprehensive outlook into the future in the section "An outlook".

State of the art

Shape grammars applications have been developed for over 40 years addressing several areas of design, most particularly architectural design, engineering, and product design. Significantly, several shape grammars implementations developed over the last 10 years have managed to successfully integrate early pioneering work on shape representation and computation and begin to explore emerging applied technologies, including parametric design tools, generative design tools, procedural modeling tools, information modeling applications, and rule-based design systems. These recent applications offer extensive support for Euclidean and parametric rules, visual definition of rules, interactive rule application including manual, semi-automatic, and automatic modes, and address issues of interoperability between shape grammar interpreters and state-of-the-art modeling tools.

Parallel to these advances, several classifications have been proposed over the last 15 years to classify shape grammar applications: some concentrate on the tasks for programs that implement shape grammars, for example, generation, parsing, and inference tasks and their interactions with CAD modelers (Gips, 1999). Others focus on technical and/or expressive characteristics of interpreters, including underlying computing language, subshape recognition, the dimensionality of shapes, etc. (Chau, 2002). Others focus on usage in design, including general interpreters versus specific domain applications; schematic design versus design development; industrial strength interpreters versus proof-of-concept applications, etc. (Chase, 2010; McKay et al., 2012); and others focus on the systematization of concepts (Garcia, 2017).

Our approach builds upon all these trajectories and generously recasts them in two distinct approaches to reflect upon the current-state-of-the-art as well as speculate on the design of new shape grammar applications: the first one privileges the design of software that allows the visual querying of a design in any conceivable way. The second privileges the design of software that allows the specification of visual rules in any conceivable way. Both approaches complement each other and both address the core of the shape grammar formalism: the dynamic play between seeing and doing and the need to accommodate both in a seamless manner in design. Note that these two approaches are related to the classic distinction in the shape grammar discourse between general interpreters and specific design domain applications, but not entirely. For example, existing general interpreters may not support emergence at all, or they may support some types of emergence, say, for example, identification of polygons, stars, and so on, but not all possible shapes and spatial relations; and existing specific design domain applications may support visual specification of rules or not while they may not support any emergence none so ever. Still the initial distinction between general interpreters and specific domain interpreters is useful, and is used below in Table 1 to rework the evolving shape grammar software list in the bibliography currently given in a variety of sources (see, e.g., Gips, 1999; Chau et al., 2004; Kunkhet, 2011; McKay et al., 2012).

Advances in implemented shape grammars

In this section, we give a brief summary of the diverse papers that are included in this special issue. The papers chosen for this volume reflect current research and concerns within the shape grammar research community on two main topics: (i) technical advancements and (ii) practical implementations both concerning interface and/or real-life design scenarios. Hence, we have divided the accepted papers into a number of categories, namely: (i) technical design, (ii) interpreters and interface design, and (iii) use cases.

Technical design

On the first topic, technical design, five articles have been included. The selected papers in this category are highly technical in nature and delve into very specific advances that can be made on the purely technical level. They include existing software implementations, yet advance beyond the current state of the art by giving a detailed impression of the strengths and weaknesses of the available systems, and making suggestions for future development opportunities at the heart of technology.

The paper by *Wortmann* and *Stouffs* discusses the algorithmic complexity of shape grammar implementations and categorizes existing implementations by their algorithmic complexity. Instead of just discussing subshape detection and emergence as the way to discuss algorithm complexity, the authors consider complexity in terms of the number of potential target shapes (or possibilities for rule application). More specifically, the authors describe how different sets of transformations (isometries, similarities, affinities, and so on) admit a number of potential possibilities for rule application in both shape grammars and parametric shape grammars and how important this is to secure design freedom. In the end, the authors propose new matching algorithms for non-parametric and parametric shape grammar implementations along with an analysis of their complexity.

Chau, Mckay, Earl, Behera, and Pennington provide a robust framework for a general shape grammar interpreter that brings together various theoretical inquiries that have remained so far somewhat independent. The key idea of the work to use lattices to transform rapidly and on-the-fly a given shape grammar to a set grammar that can be computed by a machine before it reverts back to a shape grammar representation. The results are very convincing: the structure of the design depends on the rules that are used to structure it; the emergent parts of the design (temporal atoms) are all represented as nodes in a lattice and are all computed efficiently; all rule matchings and resulting applications are readily visualized; and as if all these are not enough, the computations are all using curves in two-dimensional (2D) and 3D

https://doi.org/10.1017/S0890060417000634 Published online by Cambridge University Press

Table 1. List of existing general shape grammar interpreters and purpose-built design shape grammar applications

General interpreters	References
shape grammar interpreter (SGI)	Krishnamurti (1982)
Shape generation system	Krishnamurti and Giraud (1986)
SG interpreter	Chase (1989)
GRAIL	Krishnamurti (1992)
Shape grammar system	Stouffs (1994)
GEdit	Tapia (1999)
Shape grammar editor	Work by Shelden in 1996, cited by Gips (1999)
GraphSynth	Anon (n.d.)
U13 shape grammar implementation	Chau et al. (2004)
SGI for rectilinear forms	Trescak et al. (2009)
Parametric SG interpreter	Yue et al. (2009)
SG development system	Li et al. (2009)
Subshape Detector and SD2	Jowers et al. (2010)
Grape: U12 and U13 SG interpreters	Grasl and Economou (2013)
Shape grammar implementation	Strobbe et al. (2013)
Grape: agent-based rule decision	Grasl and Economou (2014)
Specific design domain interpreters	**References**
Shepard–Metzler analysis	Gips (1974)
Simple interpreter	Gips (1975)
Palladio grammar	Stiny and Mitchell (1978)
Queen Anne houses	Flemming (1987)
Genesis	Heisserman (1991)
Grammatica	Carlson (1993)
Genesis (Boeing)	Heisserman (1994)
Basic grammar	Duarte and Simondetti (1997)
EifForm	Shea (2000)
3D shape grammar	Piazzalunga and Fitzhorn (1998)
SG-Clips	Chien et al. (1998)
3D Shaper	Wang (1998)
Coffee maker grammar	Agarwal and Cagan (1996)
MEMS grammar	Agarwal et al. (2000)
Shaper 2D	McGill (2002)
Yingzao fashi grammar	Li (2002)
Harley Davidson	Pugliese and Cagan (2002)
Grammar use and interaction	Chase (2002)
Buick	McCormack and Cagan (2004)
Coca-Cola grammar	Chau et al. (2004)
Cross-over vehicle grammar	Orsborn et al. (2006)
Digital camera design parametric grammar	Lee et al. (2012)
Malagueira	Duarte (2005), Duarte and Correia (2006)
Shape designer	Wong et al. (2005)
Marrakech Medina grammar	Duarte et al. (2007)
Tibet Tangka grammar	Zhang and Lin (2008)
Baltimore Row-house	Aksamija et al. (2010)
QI curves	Jowers (2006), Jowers and Earl (2011)
Design synthesis and shape generation	McKay et al. (2011)
Urban grammar for Praia	Beirão et al. (2009)
Shape grammar and Tangible augmented reality	Chen et al. (2009)
Humanoid grammar	Fiedler and Ilčík (2009)
Shape Grammar Machining Planning	Ertelt and Shea (2010)
Interactive 3D Spatial Grammar System (Spapper)	Hoisl and Shea (2011)
Grappa	Grasl (2012)
Thonet chair grammar	Barros et al. (2011)
SG parsing via reinforcement learning	Teboul et al. (2011)
Rabo-de-Bacalhau grammar	Eloy and Duarte (2015), Strobbe et al. (2016)
Entelechy grammar	Ligler and Economou (2015)
Dirksen grammar	Park and Economou (2015)
Multipurpose chair grammar	Garcia and Romão (2015)

space (along with straight lines too) and both under a similar representation so that a straight line can be represented as a degenerate Bezier curve. The work is nicely presented within a useful overview of current shape grammar interpreters and its contribution is discussed within a more critical overview of visual ambiguity in creative design.

The paper by *Stouffs* aims at the implementation of a generic shape grammar implementation, similar to the aim pronounced by Chau et al. Stouffs indicates how algebras of shapes have been defined for elements of different kinds, as well as for shapes augmented with varying attributes. Grammar forms could hence potentially be expressed in terms of a direct product of basic algebras. Stouffs extends this algebraic approach in this paper: he derives combinations of basic shape algebras with attribute algebras. This algebraic abstraction at the same time serves as a procedural abstraction, giving insights into the modular implementation of a general shape grammar interpreter for different grammar forms. In addition, Stouffs considers practical limitations on algebraic compositions of basic shape algebras with attribute algebras.

The paper by *Whiting, Cagan and Leduc* discusses the possibility of efficiently generating grammars in design and analysis from arbitrary sets of data. Indeed, the authors point to the lack of automated ways to induce grammars from arbitrary structured datasets. They argue that machine translation methods can allow inducing grammars from coded data. Their proposed process for Efficient Probabilistic Grammar Induction for Design includes four steps: (1) extracting objects from the data; (2) forming structures from the object; (3) expanding structures into rules based on frequency; and (4) finding rule similarities that lead to consolidation or abstraction. To evaluate this method, grammars are induced from generated data, architectural layouts and 3D design models of Andrea Palladio's Villa Foscari, La Malcontenta. This evaluation demonstrates that this method is capable of automatically offering usable grammars which are functionally similar to grammars produced by hand.

Zimmermann, Chen, and Shea discuss the adoption of spatial grammars in engineering applications by providing a method that enables the automated link between the designs generated by the spatial grammar [developed in *spapper* (Hoisl & Shea, 2011)] and their evaluation through finite-element analysis. The authors propose a framework that combines a 3D spatial grammar interpreter with automated finite-element analysis and stochastic optimization using simulated annealing and tests it in the automated design and optimization of spokes for inline skate wheels. The results presented verify that the framework can generate structurally optimized designs within the style and additive manufacturing constraints defined in the spatial grammar, and produce a set of topologically diverse, yet valid design solutions.

Interpreters and interface design

On the topic of interpreters and interface design, two key contributions have been included. They both document existing software implementations of shape grammars, including their newest features and technological advances. Both articles focus on the connection with the end user, giving an idea of what sort of features can be made available to an end user.

Li reports the implementation of a grammar editor using Rhinoceros3D and Python, and a stand-alone general interpreter that supports subshape detection. The grammar can be modified directly in Rhinoceros3D and exported to the interpreter, and a derivation file can be exported back from the interpreter to Rhino. Users have an intuitive way of using the system being shielded from most of the subdomain tasks. This shape grammar implementation has been developed from the earlier work by Chau et al. (2004) through the inclusion of a graphical interface to shapes and rules. The current implementation also evolves from the stand-alone interpreter report in Li et al. (2009). *Li* reports as well the good feedback of participants during several classes and workshops where he tested the implementation under design scenarios.

Grasl and *Economou* describe the structure of a generic parametric shape grammar interpreter named GRAPE. The interpreter is based on graph grammars and supports emergence, parametric rules, and several types of geometric objects. The shape grammar engine is an agent-based rule selection system designed to be independent of other packages and platforms. Several plugins for commercial CAD packages, such as Rhino, AutoCAD and Revit, and one web application have been created. Users may create rules by directly using the GrGen.NET graph grammar rule modeling language or via the visual editor implemented in the web interface. This paper evolves from a previous article by the authors, namely Grasl & Economou (2013), by discussing the graph model, the visual rule editor, incorporating additional geometries (curvilinear and 3D), and introducing rule selection agents. A number of designs developed in structured workshops starting from existing grammars or from scratch show telling results and the potential of the interpreter too.

Use cases

Finally, we included two use case contributions which document applications of shape grammars in product design and urban design. These articles indicate how computer implementations of shape grammar system can be used in practice and where some of the key difficulties lie in building a wide end-user base. Both papers present practical implementations close to real-life design scenarios. They discuss the technological advances of the documented systems in close relation with advances in design and decision support tools, including parametric design tools, generative design tools, optimization algorithms and approaches, and semantics-based applications.

The problem addressed by *Beirão* and *Duarte* in their paper is the development of generic grammars that, instead of generating designs specific from one design language may encode design principles that are general and common to a large group of designs. A generic urban grammar is then presented which was inferred from specific urban grammars that resulted from the analysis of different *corpi* in the same design domain. To implement the grammar authors converted the shape grammar into a parametric design model and implemented it using Rhinoceros and Grasshopper to facilitate computer implementation. According to the authors, parametric design interfaces are more suitable for designers than state-of-the-art shape grammar's interpreters and represent better visual stimuli. Nevertheless, with the adoption of such process emergence is not supported. Such a design system can be used by designers that starting from existing elements in the urban area, references, may create an urban design solution step by step.

The second case contribution is by *Garcia* and *Leitão* that present a multipurpose chair grammar and its implementation in the design tool ChairDNA. The aim of this research is the development of a parametric set grammar design tool that allows the generation and exploration of chair design alternatives. Such a system could be used during the conceptual stages of chair design to help designers explore design alternatives that simultaneously are unexpected solutions and comply with the chair design's restrictions. In order to help designers non-familiar to shape grammars to use the interface authors opted to translate the set grammar into graphical user interface elements that are presented to users and control rule application and the derivation process. In this sense, the grammar is cast primarily as a schema whereas the designers pass numerical values to variables and sets of variables to instantiate a model based on these numbers. The authors provide an evaluation of ChairDNA done by design students and design practitioners. In this evaluation, designers assessed several criteria related to usability and others related to the usefulness of ChairDNA as a design tool. The results appear to be positive and useful providing insights on how to improve the tool and integrate it into daily life design practice.

An outlook

This special issue contributes to the debate on how shape grammars are finding their way into the design process and will, as such, extend beyond the current state of the art. A question raised in the call concerns with what more can be done in shape grammars' implementation to make them useful to a real-life design scenario. Another issue raised relates to the path being followed by current research and its focus aiming both at the design market and at the academic challenge of innovation.

Future challenges highlighted both by authors and by current literature cover four main topics: (i) the creation of new algorithms

https://doi.org/10.1017/S0890060417000634 Published online by Cambridge University Press

to expand the power of computer design generation, (ii) the inclusion of evaluation strategies during the computer generation process, (iii) the development of computer implementations that are easy to use and adaptable to designers' knowledge and work processes, (iv) and the development of specific and generic shape grammars that can be used immediately in the design practice.

The need to create *new algorithms*, highlighted in this issue by Wortmann and Stouffs, contributes to the expansion of computer-automated design systems' generation power and therefore the possibilities to make a larger type of visual calculations reachable. Further research is needed in order to extend the presented bounds and algorithms to higher dimensions and other types of geometries.

Parallel to the generation capabilities of shape grammars, design solutions need to be *evaluated* using several criteria. The main aim is that the processes of evaluation can be conducted during the generation process and not just at the end in such a way that the generation process may only produce valid design solutions. Zimmermann et al. bring an example of how this process is helpful for structural optimization problems that require manufacturing constraints, a variety of solutions, and the compliance with a personal style.

Concerning *the development of computer implementations*, two main lines of research exist. On one hand, authors aim at implementing a general shape grammar interpreter that includes all the shape grammar formalisms but does not have the concern of generating designs with a programming purpose (as housing, product design, etc.). On the other hand, there are authors that sacrifice some shape grammar characteristics, as emergence, so that a feasible computer-automated design system is developed. The evolution of software is a common concern, also referred by Li, and Grasl and Economou, since the now existing compatibility between interpreters and design software may be lost in future updates. Even though, the next steps will include the development of interpreters inside commonly used CAD systems. This will lead to a less formal way of using grammars and, as stated by Li, to a more congenial one to designers. Simultaneously, the development of shape grammar design tools that enable designers with few or no programming or grammar knowledge to use the generative power of such systems should also play a role in the current development.

The *developments of specific and generic shape grammars* that can be immediately used in design processes will foster its use by designers. Grammars that are applied to specific design problems in for example, product design and architecture may have an immediate use by designers if they respond to an existing problem. Also, the application of the concept of generic grammars to several design domains by designers from different fields can create specific styles, specific contexts, customizations, or any combinations of these.

Acknowledgements. We would like to thank Yan Jin, the former Editor-in-Chief of *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* (*AIEDAM*), for supporting the proposal and development of this special issue, and Amaresh Chakrabarti, the journal's current Editor-in-Chief for his help in realizing it. We would also like to praise the authors for their contributions and views and also the reviewers for making this issue possible.

References

Agarwal M and Cagan J (1996) A blend of different tastes: the language of coffee makers. *Environment and Planning B: Planning and Design* **25**(2), 205–226.

Agarwal M, Cagan J and Stiny G (2000) A micro language: generating MEMS resonators by using a coupled form – function shape grammar. *Environment and Planning B: Planning and Design* **27**(4), 615–626. Available at http://journals.sagepub.com/doi/10.1068/b2619

Aksamija A, *et al.* (2010) Integration of knowledge-based and generative systems for building characterization and prediction. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **24**(1), 3–16. Available at http://www.andrew.cmu.edu/user/ramesh/pub/distribution/journal/aiedam-S0890060409990138-integrationOfKnowledge-based.pdf

Anon, GraphSynth (n.d.) *Department of Mechanical Engineering, University of Texas in Austin.* Available at http://designengrlab.github.io/GraphSynth/indexP.htm (Accessed 30 July 2017).

Barros M, Duarte JP and Chaparro BM (2011) Thonet chair design grammar: a step towards the mass customization of furniture. In *Proceedings of the 14th International Conference on Computer Aided Architectural Design Futures*, Liege, Belgium, pp. 181–200.

Beirão J, Duarte JP and Stouffs R (2009) An Urban Grammar for Praia: Towards Generic Shape Grammars for Urban Design. In *Computation: The New Realm of Architectural Design, 27th eCAADe Conference Proceedings*, Istanbul, Turkey, pp. 575–584.

Carlson C (1993) *Grammatical Programming: An Algebraic Approach to the Description of Design Spaces.* Pennsylvania, USA: Carnegie-Mellon University.

Chase S (2010) Grammar based design tools : Issues of representation and interaction. *Slide presentation.* Available at http://vbn.aau.dk/files/57369300/DV_workshop_keynote_Chase.pdf.

Chase SC (1989) Shapes and shape grammars: from mathematical model to computer implementation. *Environment and Planning B: Planning and Design* **16**, 215–242.

Chase SC (2002) A model for user interaction in grammar-based design systems. *Automation in Construction* **11**(2), 161–172. Available at http://linkinghub.elsevier.com/retrieve/pii/S0926580500001011

Chau HH (2002) *Preserving Brand Identity in Engineering Design Using a Grammatical Approach Preserving Brand Identity in Engineering Design.* Leeds, UK: University of Leeds.

Chau HH, *et al.* (2004) Evaluation of a 3d shape grammar implementation. In Gero JS (ed.). *Design Computing and Cognition'04.* Dordrecht, The Netherlands: Kluwer Academic Publishers, pp. 357–376.

Chen IR, Wang X and Wang W (2009) Bridging shape grammar and Tangible Augmented Reality into collaborative design learning. In *2009 13th International Conference on Computer Supported Cooperative Work in Design.* IEEE, pp. 468–473. Available at http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4968103.

Chien S-F, *et al.* (1998) SG-Clips: a System to Support the Automatic Generation of Designs from Grammars. In *Proceedings of the Third Conference on Computer-Aided Architectural Design Research in Asia (CAADRIA).* Osaka, Japan: Osaka University, pp. 445–454.

Duarte J and Simondetti A (1997) Basic grammars and rapid prototyping. In *EG-SEA-AI Workshop '97, Applications of Artificial Intelligence in Structural Engineering.* Finland: Tampere University.

Duarte JP (2005) Towards the mass customization of housing: the grammar Siza's houses at Malagueira. *Environment and Planning B: Planning and Design* **32**(3), 347–380.

Duarte JP and Correia R (2006) Implementing a description grammar for generating housing programs online. *Construction Innovation Journal on Information and Knowledge Management in Construction* **6**, 203–216.

Duarte JP, Rocha J and Soares GD (2007) Unveiling the structure of the Marrakech Medina: a shape grammar and an interpreter for generating urban form. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **21**(4). Available at http://www.journals.cambridge.org/abstract_S0890060407000315 (Accessed 28 June 2017).

Eloy S and Duarte JP (2015) A transformation-grammar-based methodology for the adaptation of existing housetypes: the case of the "rabo-de-bacalhau". *Environment and Planning B: Planning and Design* **42**(5), 775–800.

Ertelt C and Shea K (2010) Shape grammar implementation for machining planning. In *Design Computing and Cognition 2010 Workshop.*

Fiedler S and Ilčík M (2009) *Procedural Human Posing Using CGA Grammars.* Available at https://www.cg.tuwien.ac.at/research/publications/2009/fiedler-2009-php/fiedler-2009-php-report.pdf

https://doi.org/10.1017/S0890060417000634 Published online by Cambridge University Press

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.

Garcia S (2017) Classifications of shape grammars. In Gero JS (ed.). *Design Computing and Cognition DCC'16*. Cham: Springer, pp. 229–248.

Garcia S and Romão L (2015) A design tool for generic multipurpose chair design. In Celani G, Sperling D and Franco J (eds). *Computer-Aided Architectural Design: The Next City – New Technologies and the Future of the Built Environment: 16th International Conference, CAAD Futures 2015*. São Paulo, Brazil: Springer, pp. 600–619. Available at http://link.springer.com/chapter/10.1007/978-3-662-47386-3_33.

Gips J (1974) Shephard-Metzler analysis: a syntax oriented program that performs a three dimesional perceptual task. *Pattern Recognition* **6**, 189–199.

Gips J (1975) *Shapes Grammars and their Uses: Artificial Perception, Shape Generation and Computer Aesthetics*. Basel, Switzerland: Springer. Available at http://books.google.pl/books/about/Shape_Grammars_and_Their_Uses.html?id=ajVzST9ihCcC&pgis=1.

Gips J (1999) Computer implementation of shape grammars. In *Workshop on Shape Computation*, pp. 1–11. Available at http://www.shapegrammar.org/implement.pdf.

Grasl T (2012) Transformational Palladians. *Environment and Planning B: Planning and Design* **39**(1), 83–95.

Grasl T and Economou A (2013) From topologies to shapes: parametric shape grammars implemented by graphs. *Environment and Planning B: Planning and Design* **40**(5), 905–922.

Grasl T and Economou A (2014) Towards controlled grammars: Approaches to automating rule selection for shape grammars. In *32nd eCAADe Conference Proceedings*, Newcastle, UK, pp. 357–363.

Heisserman JA (1991) *Generative Geometric Design and Boundary Solid Grammars*. Pennsylvania, USA: Carnegie-Mellon University.

Heisserman JA (1994) Generative geometric design. *IEEE Computer Graphics and Applications* **14**, 37–45.

Hoisl F and Shea K (2011) An interactive, visual approach to developing and applying parametric three-dimensional spatial grammars. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **25**(4), 333–356.

Jowers I (2006) *Computation with Curved Shapes: Towards Freeform Shape Generation in Design*. Milton Keys, UK: The Open University.

Jowers I and Earl C (2011) Implementation of curved shape grammars. *Environment and Planning B: Planning and Design* **38**(4), 616–635. Available at http://epb.sagepub.com/lookup/doi/10.1068/b36162.

Jowers I, *et al.* (2010) Shape detection with vision: Implementing shape grammars in conceptual design. *Research in Engineering Design* **21**, 235–247.

Krishnamurti R (1982) *SGI: A Shape Grammar Interpreter*. Milton Keynes, England: The Open University.

Krishnamurti R (1992) The arithmetic of maximal planes. *Environment and Planning B: Planning and Design* **19**, 431–464.

Krishnamurti R and Giraud C (1986) Towards a shape editor: The implementation of a shape generation system. *Environment and Planning B: Planning and Design* **13**(4), 391–404.

Kunkhet A (2011) *Harmonised Shape Grammar in Design Practice*. Staffordshire University. Available at http://www.kunkhet.com/research/MPhilPhD-A.Kunkhet-2011.pdf

Lee HC, Herawan T and Noraziah A (2012) Evolutionary grammars based design framework for product innovation. *Procedia Technology* **1**, 132–136. Available at http://linkinghub.elsevier.com/retrieve/pii/S22120173 12000278.

Li AI, *et al.* (2009) A prototype system for developing two and three dimensional shape grammars. In *Proceedings of 14th International Conference Computer-Aided Architectural Design Research in Asia*, pp. 717–726.

Li AI-K (2002) A prototype interactive simulated shape grammar. In Koszewski K and Wrona S (eds). *Design Education: Connecting the Real and the Virtual, Proceedings of the 20th Conference on Education in Computer Aided Architectural Design in Europe*, Warsaw, Poland, pp. 314–317.

Ligler H, Economou A (2015) Entelechy I: towards a formal specification of John Portman's domestic architecture. In Martens B, Wurzer G, Grasl T, Lorenz WE and Schaffranek R (eds). *Real Time – Proceedings of the 33rd eCAADe Conference*. Vienna, Austria: Vienna University of Technology, pp. 445–452.

McCormack JP and Cagan J (2004) Speaking the Buick language: capturing, understanding and exploring brand identity with shape grammars. *Design Studies* **25**(1), 1–29.

McGill MC (2002) Shaper2D: visual software for learning shape grammars. In *Design education: Connecting the Real and the Virtual, Proceedings of the 20th Conference on Education in Computer Aided Architectural Design in Europe*, Warsaw, pp. 148–151.

McKay A, *et al.* (2011) Computer-aided design synthesis: an application of shape grammars. *International Journal of Product Development* **13**(1), 4–15.

McKay A, *et al.* (2012) Spatial grammar implementation: From theory to useable software. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **26**(2), 143–159. Available at http://www.journals.cambridge.org/abstract_S0890060412000042.

Orsborn S, *et al.* (2006) Creating cross-over vehicles: defining and combining vehicle classes using shape grammars. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **20**, 217–246.

Park J and Economou A (2015) The Dirksen variations: towards a generative description of Mies' courthouse language. In Martesn B (ed.). *Real Time – Proceedings of the 33rd eCAADe Conferenceeal Time: Extending the Reach of Computation*. Vienna, Austria: Vienna University of Technology, pp. 453–462.

Pauwels P and Eloy S (2016) *Workshop Advances in Shape Grammars: Implemented Shape Grammars*, Evanston, Chicago, US.

Piazzalunga U and Fitzhorn P (1998) Note on a three-dimensional shape grammar interpreter. *Environment and Planning B: Planning and Design* **25**(1), 11–30. Available at http://epb.sagepub.com/lookup/doi/10.1068/b250011.

Pugliese M and Cagan J (2002) Capturing a rebel: modelling the Harley-Davidson brand through a motorcycle shape grammar. *Research in Engineering Design* **13**(3), 139–156.

Shea K (2000) eifForm: a generative structural design system. In *2000 ACSA Technology Conference: Emerging Technologies and Design: The Intersection of Design and Technology*, pp. 87–91.

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

Stouffs R (1994) *The Algebra of Shapes*. Pittsburgh, PA: Carnegie Mellon University.

Strobbe T, De Meyer R and Van Campenhout J (2013) A generative approach towards performance-based design. In *Computation and Performance, Proceedings of the 31st eCAADe Conference*, pp. 627–634. Available at http://papers.cumincad.org/data/works/att/ecaade2013_052.content.pdf.

Strobbe T, *et al.* (2016) A graph-theoretic implementation of the Rabo-de-Bacalhau transformation grammar. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **30**(2), 138–158. Available at http://www.journals.cambridge.org/abstract_S08900604 16000032.

Tapia M (1999) A visual implementation of a shape grammar system †. *Environment and Planning B: Planning and Design* **26**, 59–73.

Teboul O, *et al.* (2011) Shape grammar parsing via reinforcement learning. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference*. Available at http://vision.mas.ecp.fr/Personnel/teboul/files/cvpr11_teboul.pdf.

Trescak T, Rodriguez I and Esteva M (2009) General shape grammar interpreter for intelligent designs generations. In *Proceedings of the 2009 6th International Conference on Computer Graphics, Imaging and Visualization: New Advances and Trends, CGIV2009*, pp. 235–240.

Wang Y (1998) *3D Architecture Form Synthesizer*. Cambridge, USA: Massachusetts Institute of Technology.

Wong W-K, *et al.* (2005) Designing 2D and 3D shape grammars with logic programming. In *10th Conference on Artificial Intelligence and Applications*, Taiwan.

Yue K, Krishnamurti R and Grobler F (2009) Computation-friendly shape grammars: detailed by a sub framework over parametric 2D rectangular shapes. In *Joining Languages, Cultures and Visions: CAADFutures 2009*, Montreal, pp. 757–770.

Zhang W and Lin S (2008) Research on Tibet Tangka based on shape grammar. In *2008 9th International Conference on Computer-Aided Industrial Design and Conceptual Design*. IEEE, pp. 373–376. Available at http://ieeexplore.ieee.org/document/4730591/.

https://doi.org/10.1017/S0890060417000634 Published online by Cambridge University Press

Sara Eloy is an Assistant Professor at Instituto Universitário de Lisboa (ISCTE-IUL). Her main research interest is shape grammars and the possibilities of using them in real design scenarios. Other areas of research are CAAD, the use of immersive virtual and augmented Reality for the design process, and the analysis of the building space namely considering space perception and space syntax. She graduated in Architecture (FAUTL 1998) and has a PhD in Architecture (IST UL 2012) where she investigated on a transformation grammar-based methodology for housing rehabilitation. She is the Director of the Information Sciences and Technologies and Architecture Research Center (ISTAR-IUL), and was the Director of the Department of Architecture and Urbanism (DAU), and of the Integrated Master in Architecture at ISCTE-IUL in Lisbon between 2013 and 2016.

Pieter Pauwels is an Assistant Professor at Ghent University, Department of Architecture and Urban Planning. He holds Master (2008) and PhD degrees (2012) in Engineering: Architecture from Ghent University. He teaches Computer-Aided Design (CAD) and Building Information Modelling (BIM) to architectural engineering students and industry professionals. His research focuses on Building Information Modelling; Linked Building Data; ontologies (ifcOWL); regulatory compliance checking; (semantic) web technologies; information integration, exchange, and management. He actively chairs two technical international community groups, namely the W3C Community Group on Linked Building Data and the BuildingSMART Int'l Linked Data Working Group (LDWG). In that context, he is actively involved in the definition and management of an ifcOWL ontology that can be used as a recommended standard for capturing and publishing building data using the latest web technologies.

Athanassios Economou is a Professor in the College of Design at Georgia Institute of Technology. Dr Economou's teaching and research are in the areas of shape grammars, computational design, computer-aided design, and design theory. He is the Director of the Shape Computation Laboratory, a research group that explores how the visual nature of shape can be formally implemented with new software technologies to assist seeing and doing in design in new ways. Dr Economou holds a Diploma in Architecture from NTUA, Athens, Greece, an MArch from USC, and a PhD in Architecture from the University of California, Los Angeles (UCLA).

https://doi.org/10.1017/S0890060417000634 Published online by Cambridge University Press