

MIT Open Access Articles

Generating a Visual Language of Performance-Driven Configurations for the Principal Façade of a Prototype Sustainable House

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Kotsopoulos, Sotirios D. "Generating a Visual Language of Performance-Driven Configurations for the Principal Façade of a Prototype Sustainable House." Nexus Network Journal 17, no. 3 (October 26, 2015): 851–874.

As Published: <http://dx.doi.org/10.1007/s00004-015-0273-5>

Publisher: Springer Basel

Persistent URL: <http://hdl.handle.net/1721.1/103157>

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of Use: Article is made available in accordance with the publisher's policy and may be subject to US copyright law. Please refer to the publisher's site for terms of use.



Nexus Network Journal – Submission Cover Sheet

Author : Sotirios D Kotsopoulos

Contact address: School of Arts, Humanities and Social Sciences
Massachusetts Institute of Technology
77 Massachusetts Ave.
Cambridge, MA 02139 USA

E-mail : skots@mit.edu

Title of paper Generating a Visual Language of Performance-Driven Configurations for the Principal Façade of a Prototype Sustainable House

Category **Research**

Keywords : shape grammars, symmetry, patterns, light simulation, performance, rule-based architecture.

Keywords assure that your article can found by search engines; for help with keywords, see *NNJ* website <http://www.nexusjournal.com/for-authors/submission-guidelines/keyword.html>

Article abstract:

A performance-driven application of shape grammars is presented. A parametric shape grammar that generates a language of pattern designs for the principal façade of a prototype house – featuring a 5×20 matrix of electrochromic windows – based equally on performance and aesthetic criteria is described. The adjustment of the chromatism and light transmittance of each windowpane on the façade enables the adjustment of solar radiation at the house interior. The novel aspect of the grammar is that it encodes performance constraints of interior daylight illuminance and associates them to visual, symmetry principles of two-dimensional pattern generation. Twelve parametric rules account for the generation of the façade pattern language and five subclasses account for the symmetry of the patterns in the language

Brief biography (100-150 words, 1000 characters spaces included **maximum**) describing your work and interests (will be inserted at the end of the article).

Sotirios D Kotsopoulos studies the application of computation and procedural thinking in creative design. He uses formal, generative means like shape algebras, rule schemata and rules, to address the constructive associations of form, structure and performance in design. He holds a PhD in Design and Computation from MIT.

Generating a Visual Language of Performance-Driven Configurations for the Principal Façade of a Prototype Sustainable House

Abstract: A performance-driven application of shape grammars is presented. A parametric shape grammar that generates a language of pattern designs for the principal façade of a prototype house – featuring a 5×20 matrix of electrochromic windows – based equally on performance and aesthetic criteria is described. The adjustment of the chromatism and light transmittance of each individual windowpane on the façade enables the adjustment of solar radiation at the house interior. The novel aspect of the grammar is that it encodes performance constraints of interior daylight illuminance and associates them to visual, symmetry principles of two-dimensional pattern generation. Twelve parametric rules account for the generation of the façade pattern language and five subclasses account for the symmetry of the patterns in the language.

1 Introduction

Programmable building components and materials that can vary their states and their performance and aesthetic attributes in response to external stimuli introduce new challenges for designers who wish to deploy them in effective and aesthetically subtle ways. The use of programmable materials in building facades, in particular, provides an occasion for the precise adjustment of natural light, ventilation and heat, and affects how a building is perceived from the public street. Natural light, air and heat are essential elements of building comfort and performance that users favor most of the time, but the configuration of windows on the principal elevation of a building is also typically loaded with wide-ranging expressive connotations.

Randomly generated configurations caused by the transition of the states of programmable materials in building components could satisfy the performance premises underpinning their deployment without acknowledging their aesthetic potential. This present paper describes a performance-driven application of shape grammars enabling the visual treatment of a programmable façade covering the principal elevation of a prototype house in Trento, in northern Italy. The grammar satisfies specific performance requirements and generates visual pattern designs out of the provisions needed for their satisfaction, by conjoining symmetry principles of two-dimensional pattern generation and constraints of daylight illuminance.

The prototype house in Trento was designed and implemented by the author and a group of cross-disciplinary researchers as part of an experiment on sustainable living and home automation, conducted within the Green Home Alliance, a collaboration between the Design Lab at the Massachusetts Institute of Technology and the Fondazione Bruno Kessler in Trento. The performative strategy underlying the design of the prototype is threefold: firstly, it aims to minimize the use of artificial lighting, heating and cooling by enabling precise adjustment of the incoming natural light and heat through the management of the programmable façade; secondly, it aims to exploit the thermal capacity of the building envelope, to store heat; thirdly, it aims to prompt the residents to adopt energy-efficient behaviors. A key aspect of the overall approach is the fine management of the home system dynamics by a model-based control that leverages an Iterative Risk Allocation (IRA) algorithm to provide robust planning in the context of environmental management [Graybill 2012].

After receiving feedback from sensors – tracking the illuminance, temperature, and humidity of the interior and exterior environment – the control specifies a predictive plan (plan executive)¹ for the management of all systems, in association with statistical climatic data, and the schedules and preferences of the users. Under uncertain conditions the control offers a probabilistic guarantee that the desired comfort levels will be maintained at minimum energy expense. Parallel management of the Heating Ventilating and Air Conditioning (HVAC) system, the programmable façade, and the thermal conservation capabilities of the envelope, contributes in achieving this objective.

The primary motivation for this project was to improve the energy efficiency in the residential sector, which in 2012 is accounted for 21% of total primary energy consumption, in the United States. A parallel motivation was to supply an environmentally conscious mode of building original expressive means that align with technological innovation. Challenging the purpose of the inoperable curtain wall provided an extra motive. Inoperable curtain walls enclose air-and-sound sealed environments, restrict the interaction with the public street, and are energy intensive. The ethnographic studies accompanying this research indicated that in the Trentino region building façades serve considerably more subtle functional and expressive demands than what a typical inoperable curtain wall can serve. The grammar aims to respond to these deep-rooted expectations by pointing to original expressive possibilities without neglecting the quest for energy efficiency [Kotsopoulos et al. 2014].

¹ The plan executive describes the execution of a sequence of events that the system performs. This plan is predictive because it predicts the events for a finite amount of time, based on explicit assumptions and risk constraints.

The programmable façade is a matrix of 5×20 windows, $700 \text{ mm} \times 700 \text{ mm}$ each, arranged in columns and rows (Figure 1) and oriented towards south. Each windowpane integrates an overlay of electrochromic glass enabling the adjustment of solar radiation and heat, and a Poly Dispersed Liquid Crystal (PDLC) film enabling the adjustment of privacy and view.

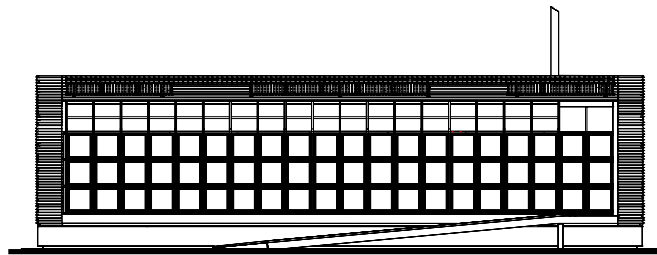


Figure 1. The south elevation of the prototype, incorporating a 5×20 matrix of electrochromic, individually addressable windows.

The management of solar radiation and heat is possible through the modification of the chromatism and light transmittance of the electrochromic layer, which can be switched between tinted and clear states. In the tinted state the windows absorb the unwanted solar heat and glare, and in the clear state maximize the incoming sunlight and heat. Hence, on a hot summer day a number of electrochromic windows are set in the tinted state to minimize solar transmittance and protect the house interior from direct sun exposure, while on a cold winter day they are set clear to maximize the exposure of the high

thermal mass envelope to the winter sun.

The parametric shape grammar presented here generates a language of façade patterns by determining the state of the electrochromic layer of the windowpanes. Patterns involving the same number of tinted windowpanes yield a specific value of interior daylight illuminance and are categorized in the same performative *equivalence class*. Twenty-seven equivalence classes account for the illuminance of the patterns in the language, namely: all 100 windows are tinted, or 0 windows are tinted, or any number between 50 and 75 windows are tinted. The grammar generates façade patterns involving any number between 50 and 75 tinted windows. Five subclasses account for the symmetry of the patterns in the language: (1) no symmetry, (2) rotational symmetry, (3) reflectional symmetry about the vertical axis, (4) reflectional symmetry about the horizontal axis, and (5) rotational and reflectional symmetry.

After a review of the background of this research, the performative and the visual scopes of the façade pattern language is described. The performative scope determines how variation in the number and distribution of tinted windows affects the levels of illuminance at the house interior. The visual scope determines how symmetry principles affect the distribution of tinted windows and the formation of patterns on the façade. Based on both of these, a façade pattern grammar is specified, including the algebras, the rules, and the derivation stages, as well as the count of façade patterns and their classification based on illuminance and symmetry criteria.

2 Background

The general conventions of visual calculation, which are adopted here, have been carefully crafted over time providing a formidable framework for the study of design languages [Stiny 1980, Knight 1994]. Stiny [1977] presented a parametric shape grammar encoding a set of conventions driving the generation of traditional Chinese ice-ray lattice patterns. In this parametric grammar Stiny applied subdivisions on an existing rectangular frame, to form ornamental window and grille designs. Stiny and Mitchell [1978a, b] described a parametric shape grammar encoding stylistic principles of generation and enumeration of Palladian villa plans. An account of the dual character, generative / expressive of spatial rule systems was presented in Knight [2005]. Knight examined four grammars producing different kinds of patterns, namely Greek meanders, ice ray lattices, Palladian plans and structural building screens, and discussed the creative features of the grammars as systems. Two noteworthy articles on optimization and performance-driven generative design are Luebke and Shea [2005], and Shea et al. [2005]. Luebke and Shea [2005] examined how navigating the performance space of a design solution promotes design thinking and exhibited the association between design variations and performance. Shea et al. [2005] used performance-driven design methods based on modeling of conditions and performance and pointed that such methods enable the designers to develop more meaningful input design models, and solutions.

This research extends the above contributions in two ways. First, like in Stiny [1977], and Stiny and Mitchell [1978a, b] it provides the full productive spectrum (generation,

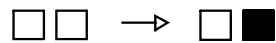
enumeration) of a parametric shape grammar encoding a design language, but instead of only encoding stylistic conventions of form, the present grammar treats in parallel explicit performance constraints of light adjustment, and symmetry principles of pattern generation [Knight 2005]. Second, like Luebke and Shea [2005], and Shea et al. [2005], performance-driven generative methods are introduced to produce designs, but the present grammar is also the algorithm driving the transitions of a given programmable architectural element – a building façade – that integrates specific electroactive material technologies. The grammar acknowledges the expressive potential of these technologies and exploits their performance and aesthetic affordances.

3 Shape grammars

A shape grammar is a generative and expressive device consisting of a calculating part and a syntactic-interpretive part. The calculating part engages an algebraic framework in which elements of 0, 1, 2 and 3 dimensions (points, lines, planes, solids, or combinations of these) are used in calculations that may happen in a space of 0, 1, 2 or 3 dimensions. The syntactic-interpretive part consists of production rules confining the syntactic (structure) and semantic (meaning) attributes of sets of products, which are called *languages*.

An algebra X_{ij} formalizes the interaction of i -dimensional elements on a j -dimensional space [Stiny 1991]. Hence, the algebra U_{12} formalizes the computations that designers execute with lines on the graphic plane: it captures the interaction of one-dimensional elements, lines ($i=1$), on the two-dimensional plane ($j=2$). This treatment can be

expanded to include algebras with labeled points (V_{ij}), which serve the naming of elements, and algebras with colors and properties like weights (W_{ij}) which serve their distinction in desired ways [Stiny 1992]. Product algebras can be formed as combinations of algebras to allow the execution of calculations with labeled and colored forms. Within this algebraic framework, elements are composed or combined with the aid of production rules that apply on an initial shape, under some Euclidean transformation (translation, reflection, rotation etc.). A product algebra $\langle U_{12} \times U_{22} \rangle$ is used in the next example of a shape rule that is similar to the rules used in the façade pattern grammar:

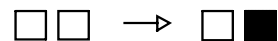


The rule is of the general form $x \rightarrow y$, where x is the shape depicted on the left side and y the shape depicted on the right side of the rule. In the context of the façade pattern grammar planes and their linear boundaries are used to represent the windowpanes and their frames, while rules are used to capture the action of tinting. A rule checks a number of neighboring windows as depicted on the left side and tints some number of these, as depicted on the right side. Because the grammar currently captures only the fully tinted and clear states of the windows, a symbolic representation in which windows are treated as switches that can be set “on” (1) or “off” (0) could have served the purpose of control, leaving the visual aspect implicit. The use of lines and planes instead of points (1,0) emphasizes the visual aspect of façade pattern generation. Furthermore, the visual aspect would be more prominent if patterns including windows with various degrees of tinting are introduced. This is technically possible because the electrochromic material can attain tinting states within a four-valued scale, namely 0%, 6%, 20% or 100%. An extension of the present grammar guiding the generation of such patterns will be objective of future

research.

Parametric rules of the form $g(x) \rightarrow g(y)$ are introduced [Stiny 1977] to supply more open ended means of composition. Parametric rules involve parameterized labeled shapes x and y with $x\langle u, w \rangle$ and $y\langle v, s \rangle$. An assignment g to the parameters of the parameterized shapes u and v , and the parameters of the labeled parameterized points w and s , results in shapes $g(u)$ and $g(v)$ and labeled points $g(w)$ and $g(s)$. More general expressions [Stiny 2006, 2011] called rule schemata enable the representation of shapes by the means of functions and arguments (assignments) to variables. In a rule schema of the form $x \rightarrow y$ the occurrences of x and y can be recursively related to generate an indefinite number of rules that associate the objects in the left and right hand side in explicit ways. For example, if all occurrences of y are Euclidean transformations t of the occurrences of x , then the rule schema $x \rightarrow y$ is rewritten as $x \rightarrow t(x)$. Alternatively, if the occurrences of y are determined to be parts (prt) of the occurrences of x , then the rule is rewritten as: $x \rightarrow \text{prt}(x)$, and if the occurrences of x are determined to be boundaries (b) of the occurrences of x , then the rule is rewritten as $x \rightarrow b(x)$, and so on. The association between conditions x and products y is explicitly determined through an operation f , so that y is expressed as a function $f(x)$ of x , and a rule schema can obtain the general form: $x \rightarrow f(x)$.

Along these lines, the shape rule:



can be described as an instance of the rule schema:

$$x \rightarrow \text{prt}^c(x) + (b^{-1}(\text{prt}(x))) .$$

In this expression $\text{prt}(x)$ indicates a part of x while $\text{prt}^c(x)$ indicates the complement of $\text{prt}(x)$. Moreover, $b^{-1}(x)$ denotes the reverse boundary of x , which is the shape that has x as its boundary. If x is the shape on the left side, then some part of x is tinted by applying a uniform tone to its area $b^{-1}(\text{prt}(x))$, while its complement $\text{prt}^c(x)$ remains unchanged. The rule of the example applies on a shape C capturing two neighboring window cells:



to produce shape C' :



under a transformation t in two ways, plain or under mirror reflection (or 180° rotation).

These correspond to the ways the shape on the left side of the rule can be matched on C :



Multiple windows can be tinted when a rule is applied in parallel at different locations of the façade matrix. This allows accounting for the symmetry properties of the produced patterns. The next expression of a rule schema captures formally this capability:

$$x \rightarrow \sum t [\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))].$$

4 A grammar of light and symmetry

The number and distribution of tinted windows can be determined to ensure a constant supply of sufficient levels of daylight at the house interior. A parametric grammar is defined that generates façade patterns that are always suitable while acknowledging the interior illuminance and symmetry properties of the patterns.

4.1 Light and performance

The performative premises pertaining to the adjustment of natural light were extrapolated through light simulation with *Relux Professional*. It thus became possible to determine how variation in number and distribution of tinted windows affects natural light in the house interior. The simulations accounted for the performance of the electrochromic layer while acknowledging the yearly daylight conditions in Trento (Figure 2).

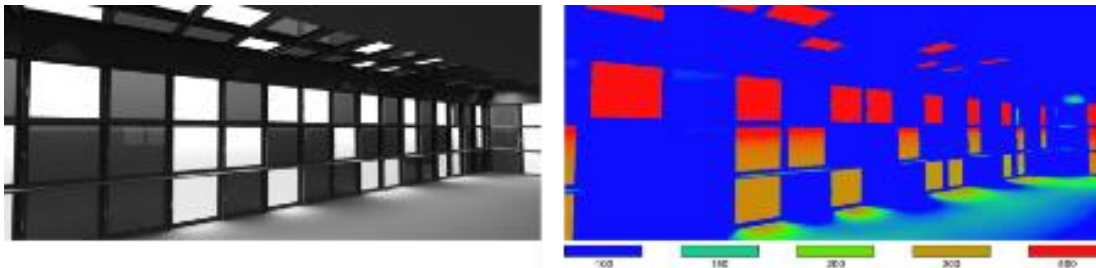


Figure 2. Ray-tracing (left) and illuminance simulation (right) of a façade pattern with *Relux*.

Typical outputs of the simulations included the minimum, maximum and average values of illuminance, the uniformity values and average daylight factor, the Isolux maps for assigned planes, and the tridimensional illuminance diagrams. Typically the interior daylight conditions are determined by the average illumination E_{ave} , the uniformity G_1 and the daylight factor D_{ave} . The minimum daytime value of illuminance E_{min} was set to 500 lux. Uniformity G_1 captures the smoothness of daylight distribution defined by the ratio E_{min}/E_{ave} . A satisfactory value is $G_1 = 0.5$. Lastly, the daylight factor represents a physical feature of windows, which is constant and set to $D_{ave} = 3$.

A link was determined between visual configuration and interior illuminance, and general principles were extracted, applicable to any façade pattern. After establishing that the

performance of two distinct patterns can remain constant throughout the year, and that a pattern can be modified from one configuration to another while maintaining a constant value of interior illuminance, it was established that any two distinct patterns with the same number of tinted windows (*coverage ratio*) could yield invariable illuminance. Hence, visual configurations could vary while retaining a constant number of tinted windows and satisfying any preset value of interior daylight illuminance, average daylight factor, and uniformity G_1 .

The notion of *equivalence class* was introduced to designate classes of patterns having the same coverage ratio and yielding proximate values of interior illuminance regardless of visual configuration. A predictive model was specified to determine what the desired coverage ratio is during any daylight condition. The model associates coverage ratio and daylight illuminance E_{ave} during specified time intervals for each day of the year. Two performance constraints encapsulating the findings of the simulations were encoded in the grammar to guarantee the generation of suitable patterns:

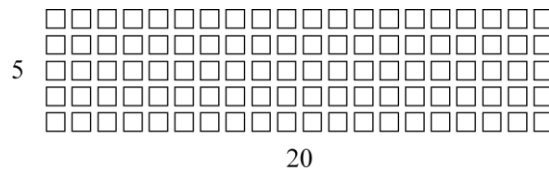
- I. Out of 100 electrochromic windows, 50 to 75 need to be active (*tinted*) to secure luminosity levels above the threshold value, $E_{min} \geq 500$ lux.
- II. If n is the number of consecutive active windows in a row, $n \leq 3$ to secure smooth light distribution, $E_{min} / E_{ave} \geq 0.5$

Three general states were determined, based on the simulations: (a) 0 % active, a single rule is required, *do nothing*, (b) 50 % - 75 % active, a generative grammar is used to

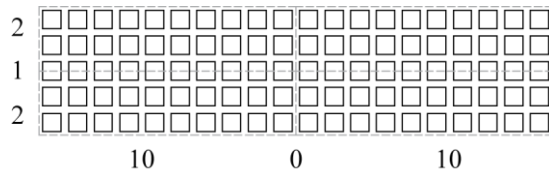
produce façade patterns, (c) 100 % active, a single rule is required, *activate all*. The grammar is used to activate any number of windows within the range 50 - 75.

4.2 Symmetry and rule schemata

The programmable curtain wall covering the south elevation of the house forms a 5×20 matrix of windows:



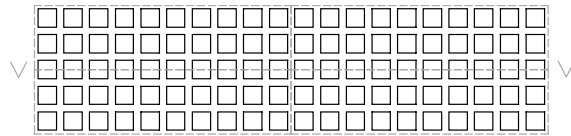
The 5×20 matrix is partitioned to enable the designation of rule schemata that make possible the generation of patterns with diverse symmetry:



The above partition specifies a distribution 2 1 2 of window columns, and 10 0 10 of window rows. The partitioning of the façade follows the suggestions of Leon Battista Alberti regarding the numbers of buildings [Alberti 1988: 303-305]. Alberti's system is based on the numbers 1, 2, 3 and 4 and their squares and cubes. As Hersey [1976: 29] points out, Alberti used a Pythagorean practice in which any number can be reduced to the desired "cubic form" by isolating a central cubic factor and flanking this with two equal cubic remainders. Alberti recommended that these numbers be used in the distribution of windows on elevations, and rooms in plans [Wittkower 1971: 113-116].

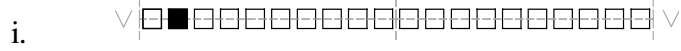
The arrangement and count of such features must comprise equal entities on either side of a real or imagined central axis, where numbers are analyzed until concluding into binary structures composed of cubic numbers.

Auxiliary graphic elements – namely axes and labels – were introduced in the description of the window matrix that is illustrated with black lines in the algebra U_{12} . Number 5 is written as $2 \ 1 \ 2$, and number 20 as $10 \ 0 \ 10$. The axes depicted as dashed grey lines in the algebra W_{12} determine the partitioning, while a pair of labeled points (V, V) in the algebra V_{02} enables the distinction between vertical and horizontal axis. Any two neighbouring partitions are bilaterally symmetrical along the vertical or horizontal axes. A description is formed in the product $\langle U_{12} \times W_{12} \times V_{02} \rangle$:

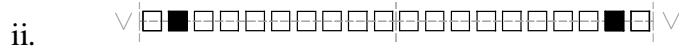


Rules can apply in a nested fashion: first, a rule can apply plain or under reflection; second, it can apply in parallel, on different locations of the matrix, under various transformations t , rotations, or reflections. Seven modes of rule application are formed, and are organized in two classes. The first class involves only the window row along the horizontal axis. The second class involves the remaining four 2×10 window rows. To generate a façade pattern it is mandatory to use one rule from each class.

In the first class a rule applies in two modes:

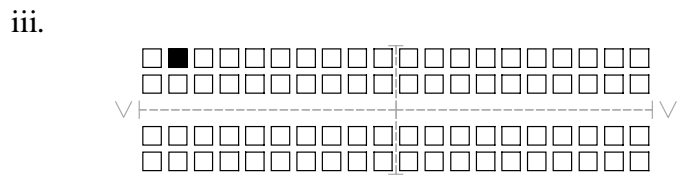


Concisely: $-I_i$

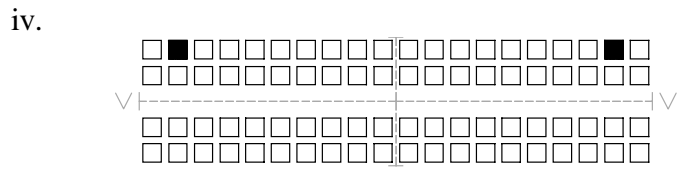


Concisely: $-I_i \quad I_i$

In the second class a rule applies in five modes:

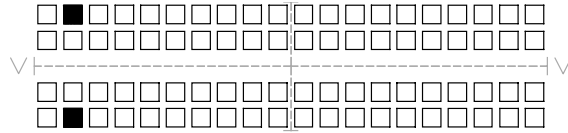


Concisely: I_i

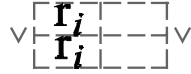


Concisely: $I_i \quad I_j$

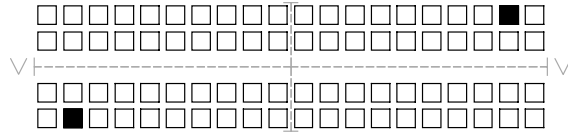
v.



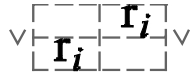
Concisely:



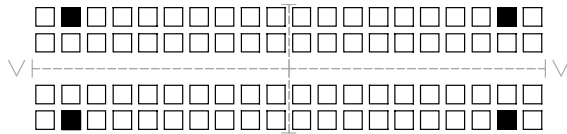
vi.



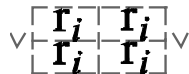
Concisely:



vii.



Concisely:



The modes (i) and (iii) require the application of a single rule r_i . They are captured by the rule schema:

$$x \rightarrow \text{prt}^c(x) + (b^{-1}(\text{prt}(x))).$$

The modes (ii) and (iv) involve parallel application of a rule r_i and of a copy of r_i under a transformation t_{rv} that is a mirror reflection along the vertical axis. They are captured by the rule schema:

$$x + t_{rv}(x) \rightarrow [\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))] + t_{rv}[\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))].$$

Mode (v) involves parallel application of a rule r_i and of a copy of r_i under a different transformation t_{rh} that is a mirror reflection along the horizontal axis. It is captured by the rule schema:

$$x + t_{rh}(x) \rightarrow [\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))] + t_{rh}[\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))].$$

Mode (vi) involves parallel application of a rule r_i and of a copy of r_i under a transformation t_{180} that is 180° rotation around the center point of the matrix. It is captured by the rule schema:

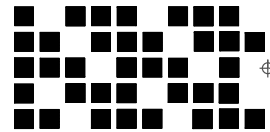
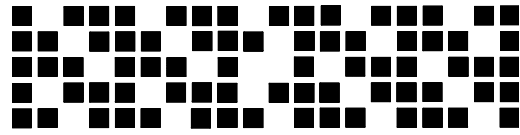
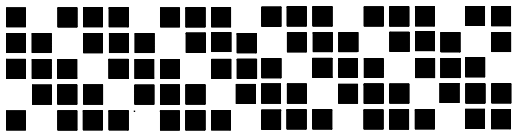
$$x + t_{180}(x) \rightarrow [\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))] + t_{180}[\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))].$$

Mode (vii) involves parallel application of r_i and of three copies of r_i under mirror reflection along the vertical t_{rv} , and the horizontal axes t_{rh} , and 180° rotation around the center point of the matrix t_{180} . Mode (vii) is captured by the rule schema:

$$x + t_{rv}(x) + t_{rh}(x) + t_{180}(x) \rightarrow [\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))] + t_{rv}[\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))] + t_{rh}[\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))] + t_{180}[\text{prt}^c(x) + (b^{-1}(\text{prt}(x)))].$$

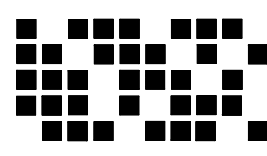
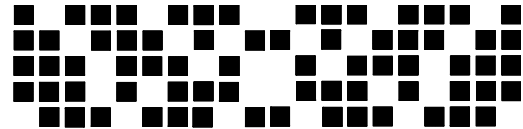
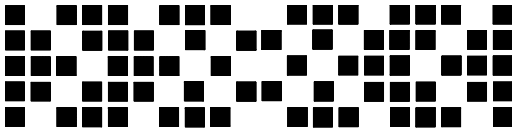
Five subclasses account for the symmetry of the patterns in the language. All seven rule schemata can be used to generate patterns with no symmetry. Combination of rule schema (ii) with (vi) or (vii) generates patterns with rotational symmetry. To generate a pattern with reflectional symmetry along the vertical central axis, rule schemata (ii) and (iv) can be combined. To generate a pattern with reflectional symmetry along the horizontal central axis, the rule schemata (i) or (ii) can be combined with (v) or (vii). To generate a pattern with reflectional and rotational symmetry only the rule schemata (ii)

and (vii) can be combined. Visual examples of patterns generated by combinations of the rule schemata (i)-(vii) are shown in Figure 3.



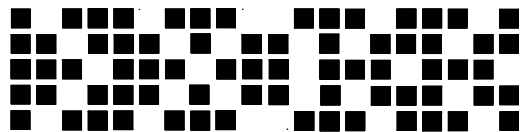
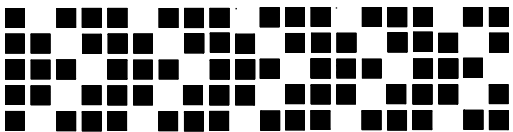
(i), (iii) no symmetry

(ii), (vi) 180° rotation



(ii), (vii) reflection and 180° rotation

(ii), (iv) reflection along the vertical axis



(i), (v) reflection along the horizontal axis

(i), (vii) reflection along the horizontal axis

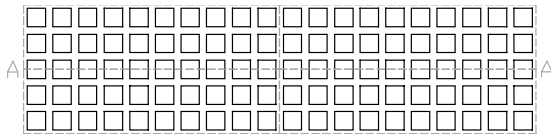
Figure 3. Visual examples of patterns generated by combinations of rule schemata.

5 The façade pattern grammar

The façade pattern grammar involves a set of twelve parametric rule instances applying on the initial shape of the inactive window matrix, and satisfying the two performance constraints I, II. The rules apply in three consecutive stages A, B, C. The section presents the algebras, the rules, and the derivation stages of the parametric grammar.

5.1 Algebras

The necessary graphic elements for deriving a pattern are finalized in this section, based on product algebras U_{ij} , W_{ij} and V_{ij} that allow the illustration of the windowpanes, their frames, and the auxiliary elements, and enable the execution of calculations with labeled and colored shapes. The initial shape is the augmented description of the 5×20 matrix of inactive (clear) windows, including grey line axes and labels. All graphic elements are manipulated on the plane ($j=2$). This description is introduced as a convenient simplification, since the actual curtain wall lies in two plane surfaces meeting at an angle of $\sim 120^\circ$, in three-dimensional space (as shown in Figure 1). Labeled points A, A along the horizontal axis, are used to indicate the stage of the derivation:



Three possible graphic illustrations of windows are determined, *tinted* (active), *clear* (inactive), and the auxiliary state, *excluded*. When window cells are marked *excluded* they become inaccessible to further computation. In this way they are distinguished from the remaining *clear* windows, which can be still switched into *tinted*. At the terminating stage of a derivation all *excluded* window cells turn into *clear*.

Inactive (*clear*) windows are represented by their outlines, as black linear squares, in the algebra U_{12} . The symbolic description that is assigned to inactive windows is simply x . Active (*tinted*) windows are represented as black planar squares in the algebra U_{22} . The symbolic description that is assigned to tinted windows is $b^{-1}(x)$. *Excluded* windows are represented as grey planar squares in the algebra W_{22} . The symbolic description that is assigned to excluded windows is $w(b^{-1}(x))$, with w denoting a weighted version of $b^{-1}(x)$. Table 1 presents the shapes, symbols, states and algebras used in representing windows.




shape			
symbol	x	$b^{-1}(x)$	$(w(b^{-1}(x)))$
state	clear(inactive)	tinted (active)	excluded
algebra	U_{12}	U_{22}	W_{22}

Table 1. Shapes, symbols, states, and algebras used in representing window cells.

Auxiliary graphic elements, like axes, are depicted with grey dashed lines in the algebra W_{12} , and labeled points A, B, and C in the algebra V_{02} denote the distinction between

horizontal and vertical axis, and indicate the ongoing stage of the derivation. The overall algebraic component serving the execution of calculations forms the product:

$$\langle U_{12} \times W_{12} \times U_{22} \times W_{22} \times V_{02} \rangle.$$

A typical derivation step $C \Rightarrow C'$ is expressed as follows:

$$\langle U_{12} \times W_{12} \times U_{22} \times W_{22} \times V_{02} \rangle \Rightarrow \langle U_{12} \times W_{12} \times U_{22} \times W_{22} \times V_{02} \rangle.$$

Components U_{12} , W_{12} , W_{22} , U_{22} , or V_{02} can be substituted with the empty shape. For example, the initial shape is expressed in the product $\langle U_{12} \times W_{12} \times \emptyset \times \emptyset \times V_{02} \rangle$.

The derivation step in which all window cells are marked either as excluded or tinted, and all linear boundaries are erased ($U_{12} \Rightarrow \emptyset$) is:

$$\langle U_{12} \times W_{12} \times U_{22} \times W_{22} \times V_{02} \rangle \Rightarrow \langle \emptyset \times W_{12} \times U_{22} \times W_{22} \times V_{02} \rangle.$$

At the end of a derivation all excluded window cells turn clear, the grey tone is erased ($W_{22} \Rightarrow \emptyset$) and the linear boundaries are redrawn in the U_{12} component:

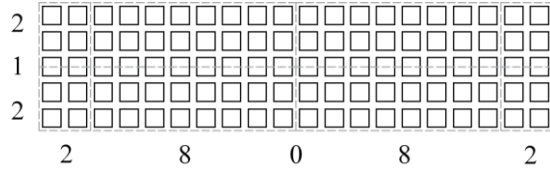
$$\langle \emptyset \times W_{12} \times U_{22} \times W_{22} \times V_{02} \rangle \Rightarrow \langle U_{12} \times W_{12} \times U_{22} \times \emptyset \times V_{02} \rangle.$$

Finally, in the terminating step all the auxiliary graphic elements are erased:

$$\langle U_{12} \times W_{12} \times U_{22} \times \emptyset \times V_{02} \rangle \Rightarrow \langle U_{12} \times \emptyset \times U_{22} \times \emptyset \times \emptyset \rangle.$$

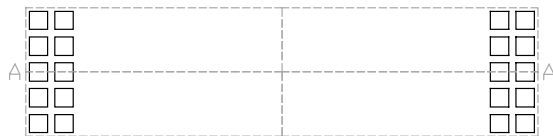
5.2 Derivation stages

The application of rules takes place in three stages A, B, and C, the productive objectives of which are explained next. The windows are represented in this section with their linear boundaries. All three stages A, B, and C, are required in order to generate a façade pattern. Each stage includes a number of productive rules and a terminating rule, applied at the end to introduce the successive stage, or to terminate the process. The partitioning of the façade is refined to reflect the application of rules at the different stages:

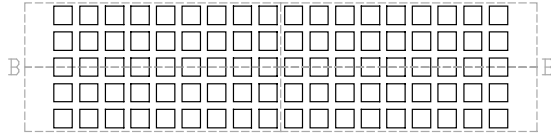


The partition describes a distribution of 2 1 2 window columns, and 2 8 0 8 2 window rows. Hence, the distribution 10 0 10 of columns is further analyzed into 2 8 0 8 2 (or $2 \cdot 2^3 \cdot 0 \cdot 2^3 \cdot 2$). This new distribution better reflects how rules apply on particular locations of the window matrix at the three stages A, B, and C.

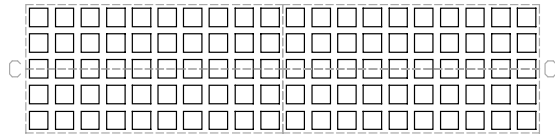
Stage A – the initiating stage – includes four rules in total (not shown here). These apply on the outermost four columns of the matrix. Stage A is terminated when the outermost window columns are tinted or excluded. The windows that are affected in stage A are shown next:



Stage B – the productive stage – includes five rules in total. At this stage pattern generation proceeds until all available windows are either tinted or excluded. The window cells that can be affected in stage B are depicted next:



Stage C – the terminating stage – includes three rules in total. At this stage pattern generation proceeds until all window cells that are marked as excluded are set clear, and all auxiliary graphic elements, such as axes and labels, are erased. The shapes that can be affected in stage C include the entire matrix:

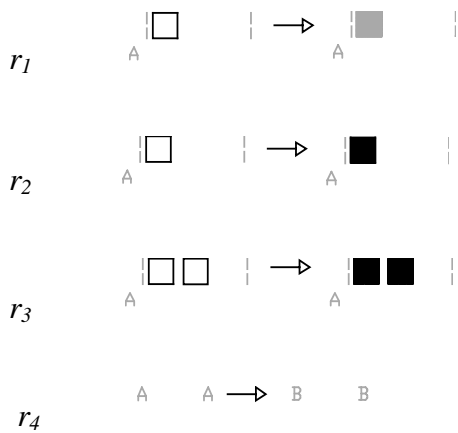


5.3 Parametric rule instances

The twelve parametric rule instances of the grammar involve parameterized labeled shapes, or points, and are named with the letter r_i (with $i = 1, 2, \dots, 12$). A rule may apply while taking into account the state of a single window and the state of a maximum of four more windows in the same row. A neighbourhood of windows involves at most m windows in the same row, with $m \leq 5$. Each productive rule scans a neighbourhood of windows, as depicted on the left hand side of a rule, and modifies the state of a number of windows as depicted on the right hand side of the same rule. Productive rules always apply to exhaust the available windows. The number of the tinted windows can be 0, 1, or 2. Rules apply in three stages A, B, and C. Stage A initiates the derivation. Stage B is the main productive stage. Stage C terminates the derivation.

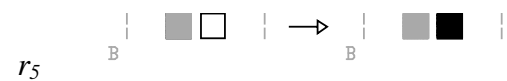
5.3.1 Stage A

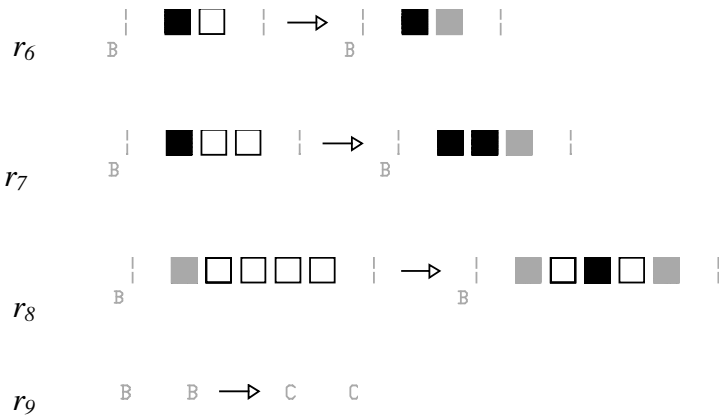
Productive rules 1-3 initiate the process. They include parameterized labeled shapes and they apply on the outermost left or right window columns. Rule 1 scans one window cell and excludes it by applying a grey tone to its area. Rules 2 and 3 scan and activate one or two cells respectively. Rules 1-3 apply until all the outermost left and right window cells are either tinted or excluded. Rule 4, involving only labeled parametric points, terminates stage A and introduces stage B, as shown below:



5.3.2 Stage B

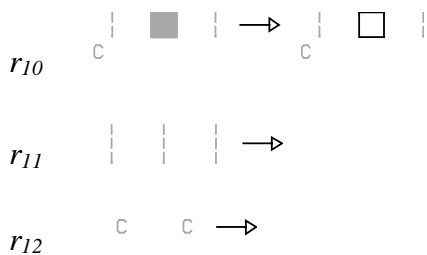
Productive rules 5-8 include parameterized labeled shapes. They can affect the entire matrix except from the outermost left and right window columns. Rules 5 and 6 scan two window cells and either activate or exclude one. Rule 7 scans three cells, activates one cell and excludes one cell. Rule 8 scans five cells, activates one cell and excludes one cell. Rules 5-8 apply until all available cells are either tinted or excluded. Rule 9, terminates stage B and introduces stage C:





5.3.3 Stage C

Productive rules *10*, *11* include parameterized labeled shapes. Rule *10* erases the grey markers from the excluded windows anywhere on the matrix and turns them into clear. Rule *11* eliminates the auxiliary grey axes. Rules *10*, *11* apply until all excluded cells are turned into clear, and all axes are erased. Rule *12* involves the erasing of the labeled parametric points, and applies last to terminate the process:

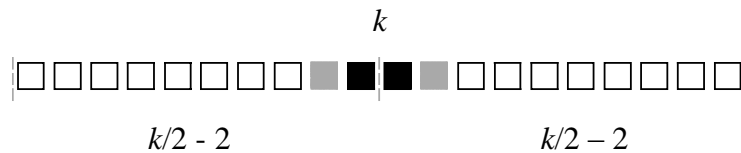


6 Counting façade patterns

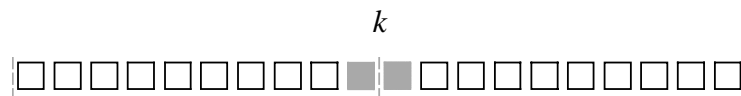
This section presents the count of patterns belonging in each of the symmetry subclasses in the pattern language. Five subclasses account for the symmetry of the patterns in the language, namely: (1) no symmetry, (2) rotational symmetry, (3) reflectional symmetry about the vertical axis, (4) reflectional symmetry about the horizontal axis, and (5)

rotational and reflectional symmetry. Class (1) includes patterns that can be generated through translation on the plane. Class (2) includes patterns that remain identical upon 180° rotation. Classes (3) and (4) include patterns that remain identical when reflected about the central axis of the façade, either vertical or horizontal. Class (5) includes patterns for which the distinction between horizontal and vertical axis is unnecessary. The five subclasses are denoted S_0 , S_{rot} , $S_{\text{ref v}}$, $S_{\text{ref h}}$, and $S_{\text{ref, rot}}$.

If n denotes the number of tinted windows, the interest is for $50 \leq n \leq 75$. The window rows of the façade are numbered $i = 1, 2, \dots, 5$, with k_i denoting the number of tinted windows in row i . The n active windows are distributed across the rows with the requirement $\sum k_i = n$ and $0 \leq k_i \leq 20$. The function $P(k, m)$ is defined to be the number of possible ways to tint k window cells in a row of length m window cells, according to the rules of the grammar. Let $S(k)$ denote the number of arrangements with k tinted cells in a row of length 20 that are bilaterally symmetric about the center. Two cases can be distinguished in calculating $S(k)$. First, the two centermost cells are tinted. Then, the cells to either side of the two centermost cells must remain clear. Thus, there are $P(k/2 - 2, 8)$ possible arrangements:



Second, the two centermost cells are clear. There are $P(k/2, 9)$ possible arrangements:



$k/2$ $k/2$

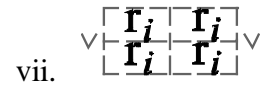
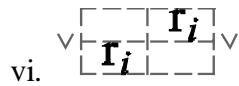
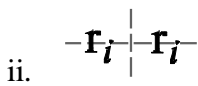
Therefore:

$$S(k) = \begin{cases} P(k/2, 9) + P(k/2 - 2, 8) & \text{if } k \text{ even.} \\ 0 & \text{if } k \text{ odd.} \end{cases}$$

Initially the subclass S_0 of patterns that do not involve symmetry is counted. All modes of rule application may be used to generate a pattern in S_0 and all possible configurations are included in the class. The total number of patterns in S_0 is the product of $P(k_i, 20)$ over each row i summed over all possible arrangements of n tinted cells across the five rows:

$$E_{S_0}(n) = \sum_{\substack{k_1, \dots, k_5 \\ \sum k_i = n \\ 0 \leq k_i \leq 20}} \prod_i P(k_i, 20).$$

For the subclass S_{rot} of patterns involving rotational symmetry the analysis proceeds separately for the windows of the horizontal axis and those of the remaining partitions. Only mode (ii) can be used on the horizontal axis, since the center row must be bilaterally symmetric. The number of arrangements is $S(k_3)$. The modes (vi) and (vii) can be used on the remaining partitions:



This implies that the configuration of the bottom two rows is determined by that of the top two rows (or vice versa). Hence, it is sufficient to count the arrangements of one section. Let $E_2(n)$ denote the number of arrangements of n tinted cells across two rows.

This value can be calculated similarly to E_{S_0} as follows:

$$E_2(n) = \sum_{\substack{k_1, k_2 \\ k_1 + k_2 = n}} \prod_i P(k_i, 20).$$

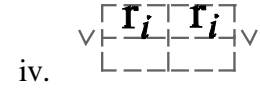
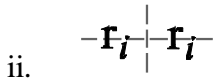
$$0 \leq k_i \leq 20$$

The counting of configurations in the subclass S_{rot} is calculated as follows:

$$E_{S_{\text{rot}}}(n) = \sum_{\substack{k_{12}, k_3 \\ 2k_{12} + k_3 = n \\ 0 \leq k_3 \leq 20 \\ 0 \leq k_{12} \leq 40}} S(k_3) \cdot E_2(k_{12}).$$

where k_{12} denotes the total number of tinted windows between rows 1 and 2.

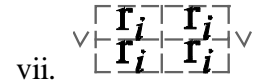
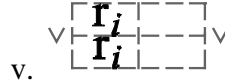
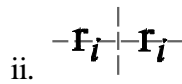
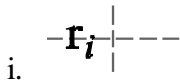
For the $S_{\text{ref v}}$ subclass of patterns involving reflectional symmetry about the vertical axis, the calculation is similar to S_0 except that only the modes (ii) and (iv) can be used, since each row is bilaterally symmetric:



For even n the count in the $S_{\text{ref v}}$ subclass is:

$$E_{S_{\text{ref v}}}(n) = \sum_{\substack{k_1, \dots, k_5 \\ \sum k_i = n \\ 0 \leq k_i \leq 20}} \prod_i S(k_i).$$

To generate a pattern in the subclass $S_{\text{ref h}}$ of patterns involving reflectional symmetry about the horizontal axis, the modes (i), (ii), (v), and (vii) can be used:



The modes (i) and (ii) can be used on the center row and the number of arrangements is $P(k_3, 20)$. For the remaining four partitions it is sufficient to count the number of arrangements for one section of top or bottom rows. As in the S_{rot} subclass this number is

$E_2(n)$. Summing over all arrangements of n tinted cells across the rows calculates the count:

$$E_{S_{\text{ref h}}}(n) = \sum_{\substack{k_1, k_3 \\ 2k_1 + k_3 = n \\ 0 \leq k_3 \leq 20 \\ 0 \leq k_1 \leq 40}} P(k_3, 20) \cdot E_2(k_1).$$

For the $S_{\text{ref, rot}}$ subclass of patterns involving rotational and reflectional symmetry, only the modes (ii) and (vii) can be used. The mode (ii) implies that there are $S(k_3, 20)$ arrangements for the center row:

ii.

vii.

To calculate the arrangements for the remaining four 2×10 sections, define $E_{2S}(n)$ to be the number of arrangements of n tinted cells across 2 rows such that each row is symmetric. E_{2S} is calculated similarly to E_{S_0} :

$$E_{2S}(n) = \sum_{\substack{k_1, \dots, k_2 \\ k_1 + k_2 = n \\ 0 \leq k_i \leq 20}} \prod_i S(k_i).$$

Summing over the arrangements of n across the rows, the count is calculated as:

$$E_{S_{\text{ref rot}}}(n) = \sum_{\substack{k_1, k_3 \\ 2k_1 + k_3 = n \\ 0 \leq k_3 \leq 20 \\ 0 \leq k_1 \leq 40}} S(k_3) \cdot E_{2S}(k_1).$$

The complete count of patterns in each symmetry class is provided in Table 2.

<i>n</i>	50	51	52	53	54
S_0	1.751×10^{27}	1.191×10^{27}	7.566×10^{26}	4.484×10^{26}	2.472×10^{26}
S_{rot}	1.466×10^{13}	0	9.541×10^{12}	0	5.397×10^{12}
$S_{ref\ h}$	6.971×10^{15}	5.664×10^{15}	4.433×10^{15}	3.338×10^{15}	2.415×10^{15}
$S_{ref\ v}$	5.069×10^{12}	0	2.961×10^{12}	0	1.488×10^{12}
$S_{ref,\ rot}$	1.521×10^7	0	1.155×10^7	0	7.948×10^6
<i>n</i>	55	56	57	58	59
S_0	1.265×10^{26}	5.986×10^{25}	2.612×10^{25}	1.048×10^{25}	3.844×10^{24}
S_{rot}	0	2.628×10^{12}	0	1.087×10^{12}	0
$S_{ref\ h}$	1.675×10^{15}	1.113×10^{15}	7.067×10^{14}	4.278×10^{14}	2.463×10^{14}
$S_{ref\ v}$	0	6.351×10^{11}	0	2.266×10^{11}	0
$S_{ref,\ rot}$	0	5.111×10^6	0	2.893×10^6	0
<i>n</i>	60	61	63	64	65
S_0	1.285×10^{24}	3.891×10^{23}	2.590×10^{22}	5.611×10^{21}	1.069×10^{21}
S_{rot}	3.759×10^{11}	0	0	2.406×10^{10}	0
$S_{ref\ h}$	1.345×10^{14}	6.943×10^{13}	1.541×10^{13}	6.562×10^{12}	2.597×10^{12}
$S_{ref\ v}$	6.616×10^{10}	0	0	2.717×10^9	0
$S_{ref,\ rot}$	1.523×10^6	0	0	2.708×10^5	0
<i>n</i>	66	67	68	69	70
S_0	1.774×10^{20}	2.529×10^{19}	3.053×10^{18}	3.063×10^{17}	2.494×10^{16}
S_{rot}	4.180×10^9	0	5.300×10^8	0	4.534×10^7
$S_{ref\ h}$	9.468×10^{11}	3.167×10^{11}	9.578×10^{10}	2.606×10^{10}	6.248×10^9
$S_{ref\ v}$	3.434×10^8	0	2.753×10^7	0	1.049×10^6
$S_{ref,\ rot}$	8.550×10^4	0	2.150×10^4	0	4.096×10^3
<i>n</i>	71	72	73	74	75
S_0	1.597×10^{15}	7.700×10^{13}	2.615×10^{12}	5.547×10^{10}	5.507×10^8
S_{rot}	0	2.285×10^6	0	5.018×10^4	0
$S_{ref\ h}$	1.295×10^9	2.299×10^8	3.131×10^7	3.537×10^6	1.756×10^5
$S_{ref\ v}$	0	0	0	0	0
$S_{ref,\ rot}$	0	0	0	0	0

Table 2. Count of façade patterns based on symmetry and number of active windows.

Restricting the production to any n in the range 50-75 tinted windows, yields 1.285×10^{24} distinct patterns in total, which are more than the number of stars in the observable

universe. These patterns are produced by the control system of the house that uses the grammar to generate façade patterns, without storing them in the system's memory.

At all times the control can generate façade patterns autonomously, in response to the weather conditions, in order to maintain desirable levels of visual comfort and to secure the long-term optimization of performance. Alternatively, the symmetry and illuminance settings can be determined by user preferences. In the absence of preset preferences all settings are determined autonomously by the control. After receiving feedback from the sensors (approximately every minute) that track the illuminance, temperature, and humidity conditions of the interior and exterior environment, the control associates these data with statistical climatic information, and the preferences of the inhabitants, to specify a twenty-four-hour plan executive, determining the states of the façade, the HVAC, etc. This programming routine is repeated every hour, and a new plan executive is specified. In this way, the control re-adjusts the states of all systems to counterbalance any event may interfere with the long-term (24 h) optimization goal.

The interaction with the control takes place via a graphical, tablet-based user interface (UI) application running on an Android tablet, which permits users to store or overwrite preferences in real time. When a user provides input, the control applies grammatical rules to generate a pattern based on this input. After calculating the number of tinted windows required to reach the desired level of interior illuminance, the control initiates the process of pattern generation to produce a pattern that falls within the prescribed symmetry and performance subclass (for example, a pattern that has reflectional symmetry $S_{\text{ref } v}$ about the vertical axis, and includes 74 tinted windows).

The grammar produces only acceptable patterns. The modes of rule application govern the symmetry properties of the patterns, and the grammatical rules are a visual encoding of the performance provisions I and II. After the desired number of tinted windows is specified, the control selects the proper rule modes to produce the desired symmetry subclass (example: for $S_{\text{ref } v}$ symmetry the modes (ii) and (iv) must be used) and applies the rules to reach this number (example: 74). The control counts the number of tinted windows and stops applying rules (in stage B) when the required number is reached, prompting the system to terminate the process (stage C).

Table 2 shows that numerous façade patterns meet any given symmetry and daylight requirements. This enables the control to alternate from one façade pattern with a particular symmetry and performance, to another. Alternatively, by designating specific symmetry subclasses, or patterns, the users can prompt the control to favor these patterns.

7 Discussion

The grammar presented here is part of an experiment in which electroactive materials and responsive methods of building control are integrated in the design of a residential prototype unit. The advances in optimization and material engineering research encourage the execution of similar experiments in building automation and energy management. The concern is that these advances, with all their precise planning and consideration of performance will not produce hospitable living environments. Against a strictly technocratic, resource- and energy-intensive vision, stands the possibility to employ these technologies in subtle ways. The present grammar responds to this

challenge by providing an environmentally sensitive solution without neglecting the need for human centric design. It enables the prototype to meet the requirement of low energy, while emphasizing the expressive role of building façades. Without prescribed states the programmable façade functions as an active matrix of apertures, enabling the users to determine how to engage with the neighbors and the public street. Hence, the grammar allows the façade to become a medium of self-expression, mirroring the transitory dispositions of the users into the urban landscape. Furthermore, the grammar acknowledges the expressive potential of the deployed material and control technologies and exploits their affordances in a systematic way, by satisfying explicit daylight requirements and generating elegant pattern designs out of the provisions needed for their satisfaction.

Although the present grammar deals only with two states of the electrochromic material, namely the fully “tinted” and fully “clear” states – because these two affect more decisively the daylight performance – the electrochromic material can attain states within a four-valued tinting scale, namely 0%, 6%, 20% or 100%. An extension of the present grammar that generates patterns including windows with various degrees of tinting would be technically possible will be objective of future research.

The absence of quantifiable data from the deployment of the described system in real use conditions for long period of time is the main shortcoming of this research, and the main obstacle to its further development. The precise assessment of the described system will be subject of future research after quantifiable data from its use will become available.

Preliminary simulation results over a week-long scenario show that significant energy savings of 19.8 % can be reached.

It is likely that in the future programmable materials and responsive methods of building control will attain a higher degree of influence in architecture. Similar to how electrification profoundly affected architecture at the close of the nineteenth century, the supply of computational power into built environments transforms their performance and aesthetic attributes, and introduces new design challenges related to the management of essential aspects of architecture, such as light, air, and heat. New experiences between the users and the inhabitable environment enabled by programmable materials and computation can be imagined, and new questions arise regarding the role of simulation, human-machine interaction, and Artificial Intelligence methods, on building performance, comfort, and aesthetics.

As cities become increasingly connected, buildings are turning into automated, responsive cells, replete with automated devices. But the propagation of responsive devices guarantees neither the conservation of energy, nor the quality of the habitable environment. Designers are called to mitigate the strains and constraints of the available “smart” systems in favor of the users, to produce cohesive, human-centric, responsive environments. In this context, the designer’s task is no longer to design static forms, but also processes and experiences through the coordination of dynamic building systems. Grammars could play a key role in providing the formal and visual means to plan the transitions of the various building systems, and to envision not only environmentally

sound and aesthetically rich buildings, but also engaging processes and experiences. The parametric shape grammar that generates a façade pattern language for the management of the programmable façade of the prototype in Trento is an exemplary design application of this kind, merging visual computation, responsive methods of building control, and material engineering research, and pointing towards original design possibilities without disregarding the quest for efficient management of the energy resources.

Acknowledgment

This research was conducted within the Green Home Alliance between the Design Lab at the Massachusetts Institute of Technology and the Fondazione Bruno Kessler in Trento, northern Italy. Thanks are due to Prof. Athanassios Economou for useful remarks.

References

- Alberti, Leon Battista. 1988. *On the Art of Building in Ten Books*. J. Rykwert, N. Leach, and R. Tavernor, trans. Cambridge, MA: MIT Press.
- Graybill, Wesley. 2012. Robust, Goal-Directed Planning and Plan Recognition for the Sustainable Control of Homes. Master's Thesis, Massachusetts Institute of Technology.
- Hersey, George L. 1976. *Pythagorean Palaces: Magic and Architecture in the Italian Renaissance*. Ithaca and London: Cornell University Press.
- Knight, Terry. 1994. *Transformations in Design: A Formal Approach to Stylistic Change and Innovation in the Visual Arts*. Cambridge: Cambridge University Press.

- Knight, Terry. 2005. Creativity • Rules. In: *Computational and Cognitive Models of Creative Design VI*, J. S. Gero and M. L. Maher, eds., 155-174. Sydney: Key Centre of Design Computing and Cognition, University of Sydney.
- Kotsopoulos, Sotirios D., Leonardo Giusti, Federico Casalegno. 2014. Designing synchronous interactions for the fenestration system of a prototype sustainable dwelling. In: *Proceedings of the 7th International Conference of the Arab Society for Computer Aided Architectural Design (ASCAAD2014)*, Effat University, Saudi Arabia, March 31-April 3, 291-301.
- Luebkehan, Chris and Kristina Shea. 2005. CDO: Computational design + optimization in building practice. *The Arup Journal* **3**: 17-21.
- Shea, Kristina, Robert Aish, and Marina Gourtovaia. 2005. Towards integrated performance-driven generative design tools. *Automation in Construction* **14**(2): 253-264.
- Stiny, George. 1977. Ice-ray: a note on the generation of Chinese lattice designs. *Environment and Planning B* **4**: 89-98.
- Stiny, George. 1980. Introduction to shape and shape grammars. *Environment and Planning B: Planning and Design* **7**(3): 343-351.
- Stiny, George. 1991. The algebras of design. *Research in Engineering Design* **2**: 171-181.
- Stiny, George. 1992. Weights. *Environment and planning B: Planning and design* **19**: 413-430.
- Stiny, George. 2006. *Shape: Talking about Seeing and Doing*. Cambridge, MA: MIT Press.

- Stiny, George. 2011. What Rule(s) Should I Use? *Nexus Network Journal* **13**(1): 15-47.
- Stiny, George. and William J. Mitchell. 1978a. The Palladian grammar. *Environment and Planning B: Planning and design* **5**(1), 5-18.
- Stiny, George. and William J. Mitchell. 1978b. Counting Palladian plans. *Environment and Planning B: Planning and design* **5**(2): 189-198.
- Wittkower, Rudolf. 1971. *Architectural Principles in the Age of Humanism*. New York, London: N. W. Norton & Company.