Nexus Netw J (2015) 17:963–980
DOI 10.1007/s00004-015-0267-3

NEXUS NETWORK JOURNAL
Architecture and Mathematics

RESEARCH

Shapes and Other Things

Terry Knight[1]

Published online: 22 August 2015

Abstract Shape grammars have offered a unique computational theory of design over the past forty or so years. Although the focus of shape grammar theory has been on shapes and designs, the material objects or things that might comply with shapes have also been considered. In this paper, I trace the history of approaches for specifying material properties and things through shape grammars. I identify early trends and their limitations, and then propose a new possibility. In early approaches, material things were viewed through the lens of shapes. I argue for a new approach in which shapes are viewed through the lens of material things. Shape grammars are adapted to define *making grammars* for computing things. Shapes are just one of many things that can be made with these grammars. I conclude with a discussion of the relationship of designing and making, and suggest that designing is a kind of making.

Keywords Algebra · Design theory · Shape grammars · Materials

Introduction

In their seminal 1972 paper introducing shape grammars, "Shape Grammars and the Generative Specification of Painting and Sculpture", George Stiny and James Gips distinguished between the description or specification of an object and the object itself. They proposed this distinction to advance a theory of design: "Formally defining the specification of an art object independently of the object itself provides a framework in which theories of design and aesthetics can be developed" (Stiny

✉ Terry Knight
tknight@mit.edu

[1] Department of Architecture, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Room 7-304F, Cambridge, MA 02139, USA

and Gips 1972: 125). And they concluded: “For design theory in the visual arts, this [a generative specification] means that the definition and solution of design problems can be based on the specification of an art object instead of the object itself” (Stiny and Gips 1972: 134). The specification they proposed was, of course, exceptionally original and novel. It was founded on shape and shape grammar, a specification that was simultaneously generative and visual. It launched the beginnings of a new, computational theory of design.

Stiny and Gips’s early distinction between *visual shapes* that are central to *design* and objects or *material things* that are central to *making* runs through many subsequent and important developments in shape grammar theory and applications. Stiny reiterated the distinction almost two decades later in another pivotal shape grammar paper (Stiny 1990). In responding to the main question of his paper “What is a design?”, Stiny cautioned against an answer that “confuses designs and things complying with them” (Stiny 1990: 97) and reminded readers that designs are “used to describe things for making” (Stiny 1990: 97). He then went on to detail the multi-faceted devices and descriptions—including visual shapes—that might comprise a design. Still later, in his 2006 book *Shape*, he repeated verbatim his definition of designs as descriptions of things for making (Stiny 2006: 225).

The history of shape grammar theory and applications, from 1972 to the present, is the history of an expanding, evolving theory of design. A significant aspect of this history is the development of formal, computational mechanisms to expand design descriptions to be more expressive of the material things that might comply with them—to narrow the gap between designs and the things made with them. A review of this history is timely. The recent, wide-spread interest in physical artifacts, materials, material culture, material practices, and new production processes across a spectrum of disciplines from the humanities and social sciences to science, engineering, and design has propelled research on new computational paradigms for designing and making. This includes new work in the domain of shape grammars.

Here, I trace the story of the various efforts to consider materiality and physical things through shape grammars and derivative formalisms. My goal is to motivate more work and new ideas, in other words, to keep the story going. To keep the story going, new theories and mechanisms that extend beyond visual shapes, but that have the depth and rigor of theories of shape, are needed. The beginnings of one such theory is described here.

Back to the story so far: I tell it in three parts. In the first part, I review early proposals to include materiality and physical things in shape grammars, and some of the applications that followed from these proposals. The formal mechanisms from this period all stem in one way or another from visual shapes. In the second part, I describe some of the complications that arise with these proposals, in particular, from viewing material things through the lens of shapes. In the third part, I argue for a new approach. I review new work by Knight and Stiny (2015) in which we treat material things in grammars in terms of properties of the things themselves rather than through shapes. Essentially, we adapt a shape grammar theory of design to propose a new, computational theory of making. In a reversal of previous approaches, we view shapes through the lens of material things. Shapes are just one of many kinds of things that can be computed with grammars. I conclude with a reappraisal of the relationship of designing and making. From the perspective of shape grammars, designing is a kind of making.

Shapes

The 1972 Stiny and Gips paper is best known as the first publication on shape grammars. Equally noteworthy is Stiny's and Gips's attention to objects and the making of material things. Their formalism for a generative specification was for the specification of objects. It had two parts: a shape specification consisting of a shape grammar for generating 2D or 3D shapes, and a material specification for determining how the shapes could be materialized as paintings or as sculptures, for example, how to paint or color the different areas in a shape in order to physically produce a painting. A novel proposal for the computational description and generation of material things was thus laid out. The two parts of this proposal are worth elaborating, particularly in light of the developments that followed.

The specification of material things begins with shapes. The unique perceptual and creative features of shapes as defined for shape grammars did not begin to be elaborated until a decade later (Stiny 1982a, b). Further, shapes were defined in the 1972 paper only in terms of straight lines. However, it is clear in this first paper that shapes in a shape grammar are special. They are much different from symbols or words in a phrase structure grammar, which inspired the invention of shape grammars. Shapes in a shape grammar are ambiguous. They do not have fixed or predefined parts. A shape can be decomposed in any way, into any parts, so that any part of a shape is open to a rule application. Additionally, when shapes are combined, any overlapping or coinciding elements in the shapes fuse—there are no hidden or layered elements in a shape. When shapes fuse, new, emergent shapes may result.

The specification of material things requires augmenting shapes with other descriptions. A generative specification is defined formally as an ordered pair consisting of a shape specification and a material specification. Material rules are illustrated in parallel with shape rules. However, material (painting) rules are not generative or defined in terms of shapes, but in terms of sets indicating the levels or stages in which the areas in a shape are generated. Additionally, painting rules are applied to a shape after it is generated by shape rules.

I turn now to some of the work that these two early ideas, respectively, made possible.

Shapes

In 1975, Stiny and Gips each published books based on their doctoral dissertations on shape grammars (Stiny 1975; Gips 1975). They each revisited their earlier work on generative specifications, but their books diverged in focus otherwise. Stiny detailed the visual and formal properties of shapes and how these properties work in shape computations. Gips focused on computer implementations and applications of shape grammars. To facilitate the implementation of shape grammars in computer programs, Gips developed a symbolic characterization of shape grammars. Instead of using shapes as the components of rules, he used symbols to denote shapes, thus avoiding the complexities involved in computing with shapes digitally as opposed to manually.

Stiny's and Gips's alternative characterizations of shape grammars laid out alternative bases for computing designs for physical things. In a 1982 letter to the editor of *Environment and Planning B*, Stiny articulated these alternatives as: shape grammars versus set grammars (Stiny 1982a: 113). Shape grammars treat shapes as spatial entities whereas set grammars treat shapes as symbolic entities, that is, as discrete elements of a set. Stiny's letter was an addendum of sorts to his earlier paper "Kindergarten grammars: designing with Froebel's building gifts" (Stiny 1980). The Froebel blocks are physical things. Stiny noted in his letter that set grammars are apt for representing recursive procedures that emulate physically adding and subtracting blocks in accordance with spatial relations. With set grammars, he wrote: "The integrity of the compositional units [blocks] in designs is thus preserved" (Stiny 1982a: 114). In other words, set grammars are appropriate when things like blocks should behave like discrete physical objects that remain constant and never fuse, and not like unpredictable and ambiguous shapes.

The idea of set grammars was taken up in a host of later works, most with the dual goals of simplifying computer implementations of shape grammars and representing physical things. Two of the many examples of this work suffice to illustrate these dual motives. Jeff Heisserman introduced *boundary solid grammars* for representing and generating solid models for specifying products in architecture and engineering design (Heisserman 1991). He implemented boundary solid grammars in the program *Genesis.* The solids in boundary solid grammars were defined as discrete elements as in set grammars. In one of the first commercial, automated applications of shape grammar based-work, Heisserman applied boundary solid grammars to the design of aircraft piping at Boeing (the details of which have not been made public). Around the same time, William Mitchell proposed *functional grammars* as specialized shape grammars for generating designs that meet functional, structural, or construction requirements. Mitchell defined functional grammars as "atomic systems" with fixed, architectural "shape units" as the primary elements (Mitchell 1991: 171). He noted that functional grammars thus mitigate the problems of computer implementations of standard shape grammars. Further, rules record only "physically feasible possibilities for interfacing elements" (Mitchell 1991: 168).

While set grammars may seem an intuitive choice for computing designs for physical things, shape grammars offered other possibilities. For example, Djordje Krstic (2001) defined new types of shape grammars that take into account physical or engineering properties of shapes. He did so by exploring combinations and variations of the shape algebras that defined the operations underpinning computations with shapes. Building on earlier work by Christopher Earl (1997) on shapes and their boundaries, Krstic described new types of shape grammars defined with algebras for compound shapes, such as solids together with their planar boundaries. These new grammars included "collision-protecting" grammars (Krstic 2001: 159–160) that model machine assembly processes in which physical parts must not collide.

Just recently, Benay Gürsoy and Mine Özkar (2015) proposed a different way to consider physical things with shape grammars. They are interested in the exploration of things and their materiality in creative making processes. They do not propose a new formalism; shapes in shape grammars represent material things in the usual, visual way, either as spatial or symbolic entities. Instead, their focus is on interpreting the formal actions of shape rules as physical/haptic actions or manipulations of real things. In a promising case study, they show how actions specified by shape rules—such as rotations and translations—can be applied systematically and by-hand to physical things to gain insights into their material behaviors and to generate novel, often unanticipated, physical outcomes.

All of the work described above started from the notion of shape, whether spatial or symbolic, to specify designs for physical things. With the exception of Gürsoy and Özkar's work, materials were generally not considered, only the physicality of the things. And for the most part, work focused on designs for physical things made of rigid elements, perhaps because of the domains—architecture and engineering—in which the work took place. The explicit treatment of materials and other properties developed in other work that explored how shapes could be combined with other descriptions.

Shapes and Other Descriptions

Stiny and Gips's 1972 generative theory of design linked shape information with material information. Their theory advanced in new directions in the decades that followed. Two different but interconnected approaches to associating materials or other properties with shapes took root. These approaches were defined mostly in terms of the algebras that support computations with grammars.

In one approach, properties were included alongside of, or in parallel with, shapes. That is, properties were defined in algebras that worked in combination with shape algebras. This *parallel* approach was embedded in a larger ambition to develop a general theory of design. In the second approach, properties were integrated with shapes. That is, properties were incorporated directly within shape algebras. This *integrated* approach was aimed, in particular, at the specification of materials or other properties. I trace the developments of these two approaches in turn, beginning with the general, parallel approach.

About a decade after the introduction of shape grammars, Stiny presented a formalism for the description of designs in his paper "A note on the description of designs" (1981). The formalism was very reminiscent of the 1972 generative specification. However, objects were not foregrounded, and there was no reference to the 1972 paper. The formalism was comprised of a shape grammar for generating and describing the spatial composition of designs, and a description function—or set of rules—for describing designs in other terms. Specifically, each shape rule had a corresponding description rule. The shape rules and the description rules applied in parallel to generate designs along with descriptions of them. A salient difference between this formalism and the 1972 one was that here design descriptions were generative and implemented simultaneously with shape rules. Stiny also defined descriptions as open-ended: "the details in a description ultimately depend on what features and properties of designs are of interest" [Stiny 1981: 265). As examples, he cited "purpose, function, or use, meaning, type, or form" (Stiny 1981: 257). Materials were not mentioned, but obviously they were not excluded by definition. Stiny gave a simple example in which shape rules for generating 3D designs made of blocks were matched with description rules that described the generated designs in terms of rooms, walls, windows, and other architectural components. Later applications of description schemes showed the generality and versatility of this design formalism. Andrew I-kang Li (2000) used a description scheme in conjunction with a shape grammar for traditional Chinese buildings to describe building sections. José Duarte (2001) used a description grammar in parallel with a shape grammar for the Malagueira houses of Álvaro Siza, to describe house designs in terms of their spatial, topological, site, cost, aesthetic, and other features.

The idea of describing designs through linked, parallel computations progressed in the next decade, primarily through the work of Stiny. In his paper "Formal Devices for Designs" (1989), Stiny turned the spotlight on algebras, the underlying framework for computations with grammars. He focused here on parallel computations based in the combination, or Cartesian product, of algebras. He looked in particular at the algebra for labeled shapes—the combination of the algebra for shapes and the algebra for labeled points.[1] Parallel computations in the algebra for labeled shapes, he wrote, generate designs described in terms of form—using shapes—and function—using labels (1989: 186). Of note, Stiny defined designs here for the first time as elements (ordered n-tuples) in a relation, and as specifications of objects for making: "Form-function relations contain designs: their elements describe objects for making" (1989:181). Both ideas recall his 1972 work. Further, he incorporated his earlier notion of description schemes to generalize the idea of computing in the Cartesian product of shape and label algebras, to the Cartesian product of any number of algebras. "In this way", he wrote, "n-ary relations containing designs comprised of descriptions in appropriate domains of interest can be defined, either in shape grammars when the algebra for labeled shapes is used repeatedly to form Cartesian products … or in description schemes when other algebras are used as well" (Stiny 1989: 187).

Stiny consolidated his notion of design as a relation in a paper published the following year. In "What is a design?" (1990), he separated the concepts of shapes and designs, conflated in earlier papers. Shape, he proposed, was just one of many possible components of a description of a design defined by a relation. In his words: "*A design is an element in an n-ary relation among drawings* [*shapes*]*, other kinds of descriptions, and correlative devices as needed*" (1990: 97; italics original). The relations that define designs, he wrote, are defined in algebras that are the Cartesian products of other algebras. These algebras are in general wide-ranging; they "contain the expressive devices used by specialists in different domains of interest, say, descriptions of form, of function, or of construction, manufacture, and assembly, and each has to be characterized in its own right" (Stiny 1990: 97). Stiny's inclusive definition of design followed: "Designs follow from a confluence

[1] Labels are ancillary, symbolic devices used in shape rules to control rule applications and to identify features of interest in designs.

of considerations, some dealing with form, and others dealing with aesthetics, function, material, construction, and so on" (1990: 103).

Stiny expanded on algebras, in particular algebras for shapes, in his paper "The Algebras of Design" (1991). Up to this point, shapes had been defined exclusively in terms of lines. Here, Stiny considered other ways to define shapes that corresponded to the diversity of drawings and models that designers use in practice. He generalized algebras for shapes made of lines to algebras for shapes made of points, lines, planes, or solids. These new algebras, either separately or in Cartesian products, provided the basis for parallel shape computations with shape grammars. And they could be combined, of course, with algebras for other descriptive devices.

Applications of these ideas took off across a range of design domains. For example, Manish Agarwal, Jonathan Cagan, and Katherine Constantine (1999) aligned a shape grammar for coffeemaker designs with a set of cost rules, that together generated coffeemaker designs along with their manufacturing costs. Li (2001) presented a comprehensive grammar for traditional Chinese building designs defined in the Cartesian product of sixteen different algebras. The grammar generated designs as members of a relation among spatial, diagrammatic, and numerical descriptions of designs. In other words, he defined a sixteen part parallel grammar. More recently, Krstic (2015) defined a parallel grammar for Rascian church plans together with justified permeability graphs.

The research above was directed toward a general theory of design, one based on multiple design descriptions and devices working in "conjunctive combination", as Stiny aptly put it (1989: 183). Materials were included implicitly in various formalisms, but were not singled out for attention. They could be included as one of many possible descriptions defined with a grammar. For example, building materials were included in Duarte's description rules for calculating the construction costs of Malaguiera house designs (2001: 475ff). Specific approaches for associating properties, including materials, with shapes developed in a related line of research in which properties were associated directly with shapes and their algebras. I turn now to this specific, integrated approach to materials.

Returning to 1989: In the paper "Color grammars: designing with lines and colors" (Knight 1989), I introduced a formalism for including properties or "qualities" of shapes in shape grammars. Qualities were defined broadly to include graphic qualities such as color, texture or shading, or other qualities such as materials or functions that belong to built designs or physical things (1989: 419). At the time, shape grammars were defined only in terms of shapes made of lines, together with labels—that is, in terms of labeled shapes. I proposed adding a third, quality component to the rules of a shape grammar so that compositions defined with shapes and qualities could be generated. Color, one of the more familiar qualities associated with works of art and architecture, was used as an example. Hence, I coined the term *color grammar* to describe this new kind of grammar. However, any quality could be substituted for color in the definitions that were given. Definitions could also be extended to include multiple qualities.

The quality component added to the rules of a grammar was called a *color field*, or more generally a *quality field*. A color field was defined both spatially and qualitatively—in other words, it was a shape together with qualities. Spatially, a color field was specified with 2D planes or 3D solids, thus extending for the first time shapes made of lines to shapes made of planes or solids (similar to Stiny's later (1991) proposal). Spatially, a color field behaved just like a shape. A color field could be divided into parts in unlimited ways, and when combined with another field, any overlapping parts fused. Qualitatively, a color field was "filled" or associated with one or more colors or qualities. Qualitative distinctions between color fields raised problems and opportunities that did not arise with shapes. When two color fields overlapped in combination, the qualities of the overlapping regions needed to be defined. To resolve this issue, I suggested the notion of "ranking" (1989: 424). Basically, when two color fields overlapped, either one field could be ranked above the other with one field "dominating" the other, or the two fields could be ranked equally. When a field dominated another, it covered and concealed the other field; when neither field dominated, the qualities of the fields blended to define a composite quality.

Color fields in combination with shapes and labels defined colored labeled shapes. Figure 1, taken from (Knight 1989), gives an example of a colored labeled shape.

Colored labeled shapes formed the components of the rules of a color grammar. The operations, relations, and transformations needed to compute with colored labeled shapes were also described. Although not mentioned explicitly, these could form the bases for algebras for colored labeled shapes, similar to Stiny's algebras for labeled shapes (1989: 174). Indeed, in both his 1990 and 1991 papers on algebras and designs, Stiny made reference to colors grammars and the idea of algebras for spatial regions containing properties such as colors (1990: 103; 1991: 181). Abstract examples of color grammars were illustrated in the 1989 paper, some to demonstrate the effects of different rankings using the same color grammar. In a subsequent paper, "Mughul gardens revisited" (Knight 1990), I applied color grammars to generate landscaping variations of Mughul gardens. Different colors were used to denote different kinds of landscaping. Colors together with different rankings showed how the same shape rules could lead to different garden designs.

In 1992, Stiny extended and contextualized his algebras of shapes within a broader computational framework for design. In his paper "Weights" (1991), he supplemented algebras for shapes with algebras for labels and algebras for "weights". The algebras for shapes included shapes of any dimension—point, lines, planes, and solids. The algebras for labels included shapes of any dimension augmented with labels or symbols. The algebras for weights included shapes of any dimension augmented with properties of any kind. The notion of a weight generalized nicely the earlier notion of a color field. While a color field was defined only in terms of planes and solids, weights were defined in terms of shapes of any

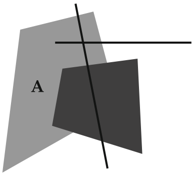


Fig. 1 A colored labeled shape, redrawn from (Knight 1989: 434)

dimension. Otherwise the two notions were equivalent. Shape, label, and weight algebras could be used alone or in any combination to support computations with shape grammars. In combination, these algebras contained compound shapes made up of shapes, labels, and weights, similar to colored labeled shapes. Figure 2 shows an illustration of a compound shape from Stiny's 1992 paper.

Operations, relations, and transformations for weight algebras were given in detail, and in a way that opened up weights to a variety of interpretations. Weights could refer to formal properties of shapes, or to physical or intentional ones. In his later book *Shape*, Stiny described weights as including graphical qualities such as color and texture, or any material property (2006: 216).

In 2000, Agarwal, Cagan, and Stiny presented the first application of weights. The authors defined a shape grammar for the design of MEMS (micro-electromechanical systems) resonators. The shape grammar combined shapes made of lines, weights made of planar regions, and labeled points. The authors argued that the use of weights allowed them to simplify the shape grammar and, more importantly, to model "more closely the physical design of a resonator that employs overlapping planes of different materials to create the final device" (2000: 625). Numerical values were assigned to weights representing the main elements of a resonator. These values were ordered (as in ranking in a color grammar) to determine which elements occlude others when elements overlap in a computation. More recently, Lynne MacLachlan and Iestyn Jowers (2014) explored the use of weights for modeling material properties of multi-material surfaces, that is, surfaces produced by new fabrication technologies that mix different materials with different properties in a single fabricated object. Like Agarwal et al. (2000), MacLachlan and Jowers associated numerical values with weights, in this case, to model the hardness and flexibility of materials. Different orderings or rankings of values were defined to explore different material outcomes. Their examples included linear orderings on weights to determine which material "dominates" another (2014: 5) that were like the rankings by color described in Knight (1989: 426).

Complications

Research on shapes and other descriptions was part of a larger effort to advance a comprehensive, computational theory of design. Within this theory, designs were of interest as specifications for physical things, or often as abstract entities in their own right. Work on physicality and materiality was always rooted in visual shapes describing designs for physical things, whether or not these shapes

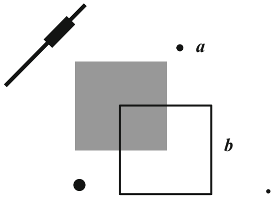


Fig. 2 A compound shape, redrawn from (Stiny 1992: 423)

were treated spatially or symbolically. Material descriptions could be added, either in parallel with shapes or integrated with them. This work over forty plus years has been rigorous and in-depth. Nonetheless, complications can arise when things are described through visual shapes, with or without additional material descriptions. Unintuitive, unexpected, or unwanted behaviors and outcomes can occur.

To begin with, consider the description of physical things in terms of shapes, putting materials aside for the moment. Conceptual differences between shapes and things can lead to quandaries. Shapes, as defined by Stiny, have special properties and behaviors tailored to visual perception—they are ambiguous and they fuse in combination. They behave in ways that physical things may not. Indeed, this issue was observed early on by Stiny in his 1980 "Kindergarten" paper. He referred to a certain "unpleasantness" (1980: 443) that results from the fusion of shapes. For example, when a rule is applied to subtract a cube (defined with lines) from a contiguous cube, an incomplete cube is produced (see Fig. 3). Clearly, this is not desirable if shapes are to be interpreted as physical blocks.

He also noted the "confusion" (1980: 439) that might result from spatial ambiguity, for example, when new shapes are produced in a computation that are not shapes added explicitly by the application of rules. In such cases, rules might apply in unintended ways. For instance, a rule might be applied inadvertently to a pillar made up of parts of two intersecting oblongs (1980: 440). Stiny suggested that solving these problems could be "very tricky" (1980: 440) and require special or ad-hoc solutions. Keep in mind, though, that these problems may not be problematic at all, but desirable in fact, when shapes are exploited for their visual creative possibilities and rules are pursued for their "unruliness", as Lionel March (1996) once put it.

At the time of the Kindergarten grammar paper, shapes were defined with lines only, so shapes made of planes or solids in 3D might seem a better bet for specifying physical things. Planar shapes would not solve the subtraction problem above, but solid shapes would. However, solids would then pose problems for addition. Solid shapes representing discrete physical things can merge in combination so that the separate shapes disappear (see Fig. 4).

Problems like these might be avoided through special composite algebras such as those described by Krstic (2001). Krstic's algebras for shapes and their boundaries are, indeed, suitable for modeling physical things in many engineering applications (2001: 157). However, different operations in these algebras might need to be defined on a case-by-case basis depending on whether or not physical parts, especially parts of different materials, are meant to maintain their integrity when they are added to designs or removed from designs.

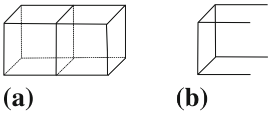


Fig. 3 **a** Two contiguous cubes made of lines and **b** the result of subtracting the rightmost cube

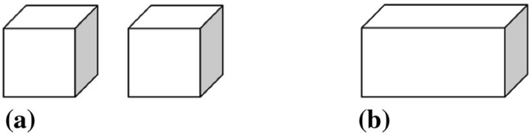


Fig. 4 **a** Two cubic solids **b** When combined side-by-side, the two cubes produce one solid

Another way to model physical things is through set grammars. But set grammars have limitations too. They have been applied successfully to describe discrete physical things made of rigid materials. But they may not be suited for modeling discrete physical things made of flexible or deformable materials. For example, sheet materials may remain discrete in combination, but internally a sheet material such as paper may be foldable (decomposable into parts) in unlimited ways like a shape.

Associating materials with shapes, through colors and weights, affords other opportunities for modeling physical things. Weights and colors can disambiguate parts of shapes representing different physical things in productive ways. However, because colors and weight values are associated with visual shapes, unwanted situations like those described above can still occur. For instance, if the two cubes in Fig. 4a are weights with the same material properties, they would still merge as shown in Fig. 4b. They could be distinguished through their linear boundaries, but other issues might emerge if subtraction is used.

All of the problems described here can be resolved in one way or another through some combination of formal and technical devices involving shapes with labels, weights, colors, or other descriptions. But these solutions may not be pretty or intuitive. An alternative is to consider materials and physical things from a different perspective—a perspective based on the original, defining qualities of shape grammars. With this perspective, things are not viewed through the lens of visual shapes. Rather, shapes are viewed through the lens of things.

Things

The core features of shape grammars make for a unique, computational theory of design, one aligned especially well with creative design practice. Shape grammars promote an *improvisational, perceptual*, *and action-oriented* approach to designing. The simple two-rule grammar depicted in Fig. 5 begins to illustrate this. Each of the rules shown takes a shape—a square or an L-shape—and moves it in space along a diagonal. Registration marks fix the before and after locations of the shapes. The rules can apply using different transformations or orientations allowing for moves along a diagonal in different directions. The example computation begins with a double L-shape. It reveals outstanding features of this shape grammar and shape grammars in general. The rules are nondeterministic and, importantly, can apply to emergent shapes in a computation—shapes that are not added explicitly by the rules. Many other computations, besides the one shown, are possible. In each step of a computation, the user can choose on the spot what shapes to see and what action to take next.

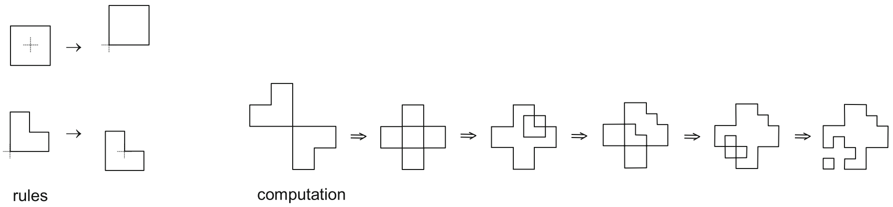


Fig. 5 A shape grammar and one of many possible shape computations using the rules

Shape computations, like the one in Fig. 5, are based on perception and action. They are about seeing and drawing with basic spatial elements, using one's eyes and hands, to make shapes. From this point of view, shape grammars ally well with contemporary theories of making in the social sciences and humanities. Recent studies in these fields have probed the active, contingent, sensory, and embodied nature of the making of physical things, from paintings to pots to buildings. They have highlighted the role of materials in the dynamic formation of things, especially how the physical and sensual qualities of materials can give rise to emergent, unplanned outcomes. The work of Ann-Sophie Lehmann (2015) on materials, Tim Ingold on skilled practice (2010), Lambros Malafouris (2004) on embodiment, and David Howes on the senses (2005) are just a few examples of the range of recent investigations into making. Shape grammars connect well to these ideas. Shape computations can be regarded a kind of making activity: the basic spatial elements are the materials for making and the shapes computed are the things made.

In this light, shape grammar theory offers a natural basis for a computational theory of making. Designing with shape grammars is about *doing* (*drawing*) and *seeing* with basic *spatial elements* to make *shapes*. George Stiny and I have recently extended this definition of designing to propose a definition of making: making is *doing* and *sensing* with *stuff* to make *things* (Knight and Stiny 2015). I summarize the details of our work here, beginning with informal definitions of the terms we use. *Doing* is an action such as drawing, knotting, folding, typing, throwing, stomping, and so on. *Sensing* includes any one or more of our senses. Both doing and sensing can be done with "tools". Tools might be our bodies' "tools" such as our hands or our eyes, or tools might be extensions of our bodies, for example, pencils or eyeglasses. Both doing and sensing might include actions or sensings by a machine as well as by a person. *Stuff* includes physical materials like gases, liquids, or solids with properties that can be visual, acoustic, mechanical, geometric, and so on. A little more abstractly, stuff can be points, lines, planes, and solids. *Things* are finite objects made of stuff. For example, shapes are things made of lines; origami are things made of paper; paintings are things are made of watercolor.

With these definitions in hand, shape grammars for computing or making shapes can be adapted to *making grammars* for computing or making things. To support grammars for things, new kinds of algebras for the stuff of things are essential. Whereas algebras for the stuff of shapes are defined in terms of what we can draw and see with spatial elements, algebras for the stuff of things generalize this idea to encompass other kinds of doings and sensings with other kinds of stuff. Algebras for stuff are intended to formalize properties of materials that are relevant to artists, designers, or makers—properties that may be very different from those of interest to scientists or technologists. Algebras for stuff are meant to capture the sensory and experiential properties of stuff that are pertinent to making. The properties of interest for making may be likened to the properties of interest in manipulating substances in alchemy—"substances that were known by what they look and feel like" (Ingold 2011: 213)

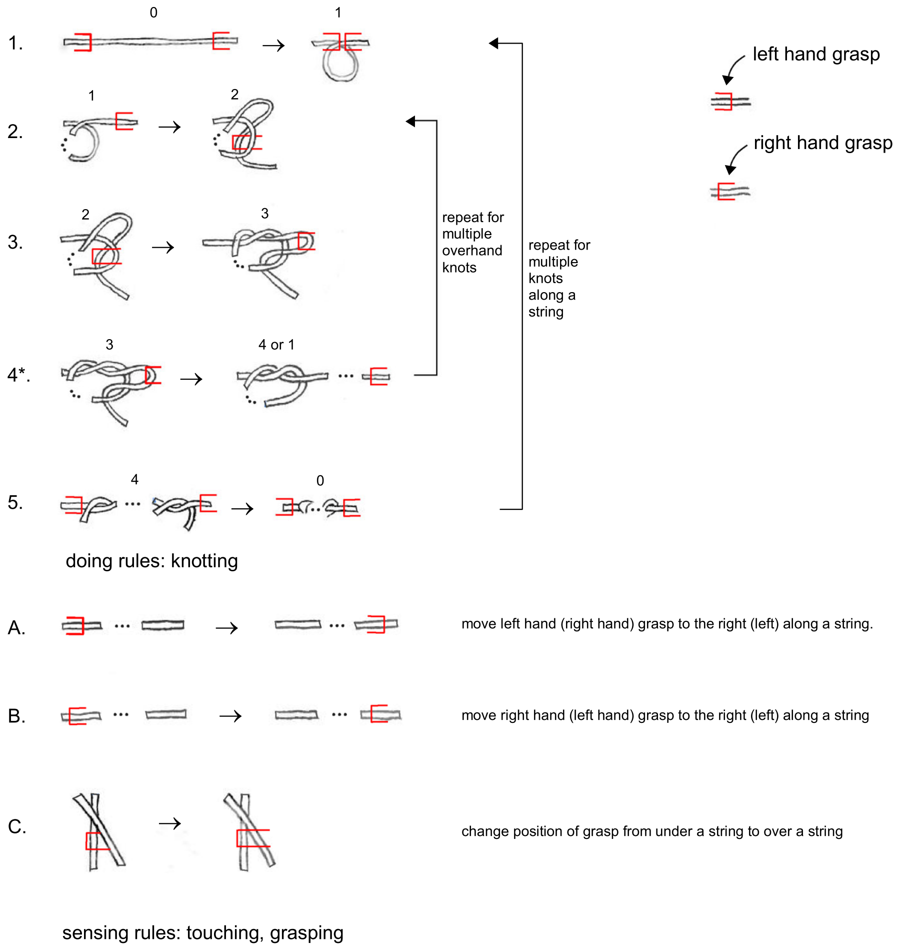
With algebras for stuff as foundation, *making grammars* for computing material things can be defined. In our preliminary work (2015), Stiny and I give an example of a making grammar and its algebra. I sketch the highlights here. The grammar is for knotting strings, a highly tactile making activity. The grammar generates single and multiple overhand knots along a string. It includes:

Things	knotted strings in 3D
Stuff	strings
Doing	knotting (looping, pulling, etc.)
Sensing	touching (grasping, focusing attention, repositioning) with the hands

Figure 6 shows the knotting grammar. There are two sets of rules: doing rules for knotting and sensing rules for touching or grasping. The brackets in both sets of rules indicate the locations of left and right hand grasps, using thumb and forefinger, along a string. The knotting rules apply to change the configuration of a string by looping and pulling the string in different ways to tie knots. The touching/grasping rules apply to change a grasp location along a string (rules A and B) or to change the position of a hand (rule C) from under a string to over a string. They do not change the configuration of the string. Rule 4* is both a knotting and grasping rule: it applies to pull a string through a loop while sliding or moving a grasp along the string. The labels 0, 1, 2, 3, and 4 in the doing rules control the sequence in which these rules apply. Sensing rules can be applied at any time. All of the rules apply under the same transformations as shape grammars, so that knots can be made in different orientations and directions. The grammar is a highly schematized version of an actual knotting process. The knotting rules capture natural stopping or stable points in a continuous tying process.

The algebra for the knotting grammar is defined in terms of the unique properties of strings. It is motivated by differences and similarities with the algebra for shape elements, in particular, lines. Lines fuse in combination, but strings are independent in combination. Further, both lines and strings can have indefinite parts, but lines can have parts in them anywhere whereas the parts of strings are limited by knots and their locations. Details of the string algebra are given in (Knight and Stiny 2015). As we note there, our algebra for strings might be extended to other materials, for example, surfaces such as paper where folds would correspond to knots. An arrangement or array of areas and folds is one enticing possibility.

Interpreting the rules in a knotting grammar in terms of our string algebra is fairly straightforward. Essentially, the pieces of a string(s) in the left side of a rule must be included in a string(s) that is part of the string to which the rule applies. The result of applying a rule or a sequence of rules in the order given in the grammar is a knot corresponding to the tying operation in the algebra. A computation of a knotted


1.
0
1
2.
1
2
3.
2
3
4*.
3
4 or 1
5.
4
0
repeat for multiple overhand knots
repeat for multiple knots along a string
left hand grasp
right hand grasp
doing rules: knotting
A.
B.
C.
sensing rules: touching, grasping

move left hand (right hand) grasp to the right (left) along a string.

move right hand (left hand) grasp to the right (left) along a string

change position of grasp from under a string to over a string

Fig. 6 A Making Grammar: knotting with string

string is shown in Fig. 7. The computation is sequential, but separated into two columns to distinguish doings from sensings.

The knotting grammar and string algebra are simple, but suggestive of exciting possibilities for grammars for other kinds of things and algebras for other kinds of stuff. In our 2015 paper, Stiny and I give two other examples. One example demonstrates how shape computations can be reconceived as a kind of making. The shape grammar of Fig. 5 is recast as a making grammar, that is, as a grammar for making shapes. The rules of the making grammar include doing rules and sensing rules. The doing rules are rules for drawing, and are identical to the rules in Fig. 5. The sensing rules are rules for seeing. These rules identify, or visually grasp, a shape to be altered by a drawing rule. They are like Stiny's "identity rules" (Stiny 1996). Another example we describe is considerably more complex and we only speculate about its development. This is a grammar for watercolor painting. The

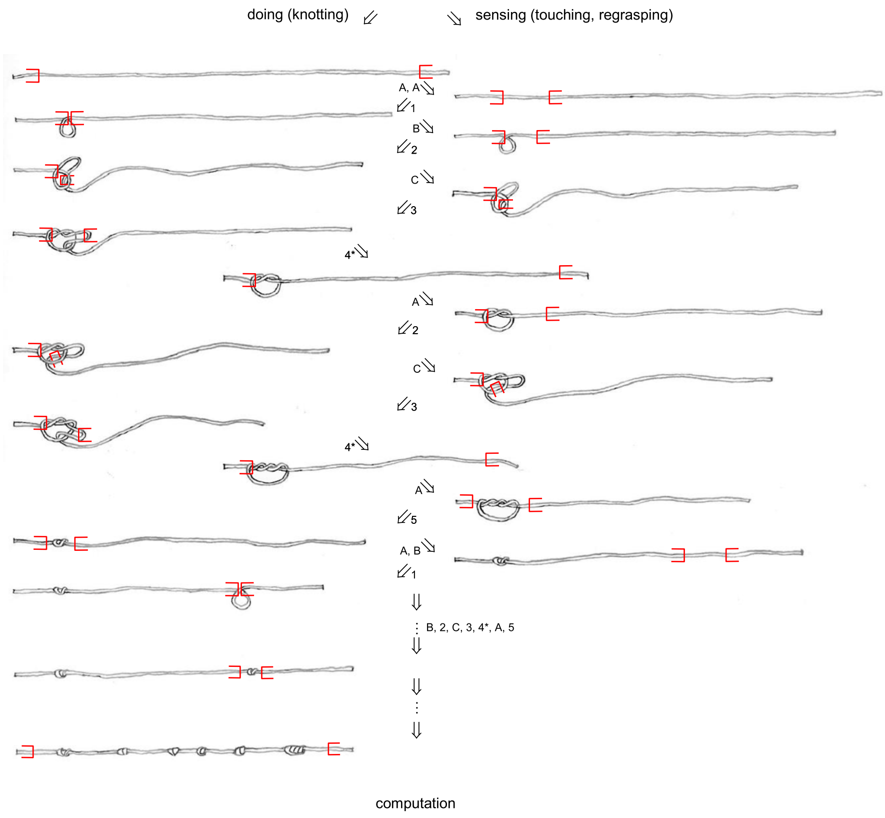


Fig. 7 Knotting a string with the grammar of Fig. 6

challenges of understanding and describing computationally a rich, highly skilled practice such as watercolor painting are many. The stuff, the doings, and the sensings are complicated, variable, and interrelated. Multiple different materials with different behaviors and properties interact with one another in different ways, in different sequences, and with different tools and techniques. Each different material might require a different algebra, and these may need to be combined in an algebra for entities of various kinds. But these intricacies add to the appeal of discovering, capturing, and understanding in a new way a creative making activity like painting.

Discussion

In 1972, the separation of the specification of a thing—that is, the design for a thing—and the thing itself was critical in establishing a new, computational theory of design. Design was then, and still is, concerned mostly with creating representations and plans for things that can be materialized subsequently. However, in recent years advances in readily accessible fabrication and prototyping machines have spurred interest among designers in generating and engaging more directly with material things. Outside the field of design, work ranging from cultural studies of making practices (traditional and digital) to research into technological innovations and applications of new materials and manufacturing devices, has bolstered this growing interest. Designing and making are more and more intertwined, and inquiries into their relationships are expanding.

Within the field of computational design—beginning with its early origins in computer-aided design (CAD) up to the present day—research has focused on design and designing, as the field's name implies. Shape grammars have provided one of the most enduring theories of design within this field. But shape grammar theory has not remained fixed. Diverse trajectories of research on computer implementations, practical applications, pedagogy, and the philosophy and mathematics underlying shape grammars have shaped the development of shape grammar theory. And, as I have traced it here, the history of shape grammar theory includes important explorations into ways to account for materials and physical things. From the beginning, this work has emanated from visual shapes and stretched the limits of descriptions based on shapes.

Now we have a different approach to making but one inspired by the unique features of shape grammars as a computational theory of design. Shape grammars have long promoted design as an activity that demands perceptual, active engagements with materials, that is, seeing and drawing with shapes. In fact, shape grammars have always promoted designing as a kind of making. Making shapes is hardly different from making knots or making paintings. The recent work outlined here points to new directions for shape grammars in the terrain of material things and their making.

References

Agarwal, M., Cagan, J., and K. G. Constantine. 1999. Influencing generative design through continuous evaluation: Associating costs with the coffeemaker shape grammar. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AI EDAM)* **13**: 253–275.

Agarwal, M., Cagan J., and G. Stiny. 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.

Duarte J. P. 2001. Customizing Mass Housing: A Discursive Grammar for Siza's Malagueira Houses. Ph.D. thesis, Massachusetts Institute of Technology.

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

Earl, C. F. 1997. Shape boundaries. *Environment and Planning B: Planning and Design* **24** (5): 669–687.

Gips, J. 1975. *Shape Grammars and Their Uses: Artificial Perception, Shape Generation and Computer Aesthetics.* Basel: Birkhaüser.

Gürsoy, B. and M. Özkar. 2015. Visualizing making: Shapes, materials and actions. *Design Studies* **41** (forthcoming).

Heisserman, J. A. 1991. *Generative Geometric Design and Boundary Solid Grammars,* Ph.D. thesis, Carnegie Mellon University.

Howes, D. (Ed.). 2005. *Empire of the Senses: the Sensual Culture Reader.* Oxford: Berg.

Ingold, T. 2010. The Textility of Making. *Cambridge Journal of Economics* **34**: 91–102.

Ingold, T. 2011. *Being Alive: Essays on movement, knowledge and description.* NY and London: Routledge.

Knight, T. W. 1989. Color grammars: designing with lines and colors. *Environment and Planning B: Planning and Design* **16** (4): 417–449.

Knight, T. W. 1990. Mughul gardens revisited. *Environment and Planning B: Planning and Design* **17** (1): 73–84.

Knight, T. and G. Stiny. 2015. Making grammars: From computing with shapes to computing with things. *Design Studies* **41** (forthcoming).

Krstic, D. 2001. Algebras and grammars for shapes and their boundaries. *Environment and Planning B: Planning and Design* **28** (1): 151–162.

Krstic, D. 2015. Language of the Rascian School: Analyzing Rascian Church Plans via Parallel Shape Grammar. In *Design Computing and Cognition '14*, eds. J. S. Gero and S. Hanna, 421–436. Switzerland: Springer International Publishing.

Lehmann, A-S. 2015. The Matter of the Medium. Some Tools for an Art Theoretical Interpretation of Materials. In: *The Matter of Art: Materials, Technologies, Meanings 1200*-1700, eds. C. Anderson, A. Dunlop, and P. H. Smith, pp. 21–41. Manchester: Manchester University Press.

Li, A. I-k. 2000. Integrating symbolic and spatial information in shape grammars, with an example from traditional Chinese architecture. In: *Proceedings of the Fifth Conference of the Association of Computer Aided Architectural Design Research in Asia* (CAADRIA 2000), 245–253.

Li, A. I-k. 2001. *A Shape Grammar for Teaching the Architectural Style of the Yingzao Fashi*. Ph.D. thesis, Massachusetts Institute of Technology.

MacLachlan, L. and I. Jowers. 2014. Formalising flexible multi-material surfaces as weighted shapes. In: *Creative Shape Modeling and Design,* SIGGRAPH Asia 2014 Proceedings, Article 3, 1–3.

Malafouris, L. 2004. The cognitive basis of material engagement: Where brain, body and culture conflate. In: *Rethinking Materiality: The Engagement of Mind with the Material World*, eds. E. DeMarrais, C. Gosden, and C. Renfrew, 53–62. Cambridge: McDonald Institute Monographs.

March, L, 1996. Rulebound unruliness. *Environment and Planning B: Planning and Design* **23** (4): 391–399.

Mitchell, W. J. 1991. Functional grammars: An introduction. In: *Computer Aided Design in Architecture '91: Reality and Virtual Reality*, eds. G. Goldman, G. and M. Zdepski, 167–176. Troy, NY: Association for Computer Aided Design in Architecture, 1991.

Stiny, G. 1975. *Pictorial and Formal Aspects of Shapes and Shape Grammars.* Basel: Birkhauser.

Stiny, G. 1980. Kindergarten grammars: designing with Froebel's building gifts. *Environment and Planning B: Planning and Design* **7** (4): 409–462.

Stiny, G. 1981. A note on the description of designs. *Environment and Planning B: Planning and Design* **8** (3): 257–267.

Stiny G. 1982a. Spatial relations and grammars. *Environment and Planning B* **9** (1): 113–114.

Stiny, G. 1982b. Shapes are individuals. *Environment and Planning B* **9** (3): 359–367.

Stiny, G. 1989. Formal devices in design. In: *Design Theory 8*, eds. S. A. Newsome, W. R. Spillers, and S. Finger, 173–188. New York: Springer-Verlag.

Stiny, G. 1990. What is a design? *Environment and Planning B: Planning and Design* **17** (1): 97–103.

Stiny, G. 1991. The algebras of design. *Research in Engineering Design* **2** (3): 171–181.

Stiny G. 1992. Weights. *Environment and Planning B: Planning and Design* **19** (4): 413–430.

Stiny, G. 1996. Useless rules. *Environment and Planning B: Planning and Design* **23** (2): 235–237.

Stiny, G. 2006. *Shape: Talking About Seeing and Doing*. Cambridge, MA: The MIT Press.

Stiny, G. and J. Gips. 1972. Shape Grammars and the Generative Specification of Painting and Sculpture. In: *Information Processing* 71, ed. C. V. Freiman, 1460–1465. Amsterdam: North-Holland. Republished 1972 in: *The Best Computer Papers of* 1971, ed. O. R. Petrocelli, 125–135. Philadelphia: Auerbach.

Terry Knight is a Professor of Design and Computation in the Department of Architecture at the Massachusetts Institute of Technology. She conducts research and teaches in the area of computational design, with an emphasis on the theory and application of shape grammars. Her book *Transformations in Design* (Cambridge University Press, 1994) is a well-known introduction to the field of shape grammars, and she has published extensively on shape grammar-related topics in design research journals. Her recent research is in the new area of Computational Making, where she is exploring the incorporation of material, sensory, and improvisational aspects of making things into grammars. She has served on the editorial boards of *Languages of Design*, *Environment and Planning B: Planning and Design*, *Journal of Mathematics and the Arts*, *ArchiDoct*, and *Design Science Journal*. She holds a BFA from the Nova Scotia College of Art and Design, and an MA and PhD in Architecture from the University of California, Los Angeles.