Environment and Planning B, 1980, volume 7, pages 343-351

Introduction to shape and shape grammars

G Stiny
Centre for Configurational Studies, The Open University, Milton Keynes MK7 6AA, England
Received 26 November 1980

Abstract. The definitions pertaining to the shape grammar formalism are developed in detail.

In this paper, we take a whirlwind tour through the *shape grammar* formalism, and the definitions and ideas on which it is based. The formal machinery for the algorithmic definition of *languages* of two- and three-dimensional spatial designs is thus established.

Shape

Definition: A shape *is a limited arrangement of straight lines defined in a cartesian coordinate system with real axes and an associated euclidean metric.*

In order to formalize this idea, the following definitions pertaining to lines are used: A *line* l, $l = \{p_1, p_2\}$, is determined by any set of two distinct points p_1 and p_2, called the *end points* of the line. A line always has limited but nonzero length. Two lines are equal if and only if they have the same end points. A point p is *coincident* with a line l having end points p_1 and p_2 if and only if p is an end point of l or the length of l is equal to the sum of the lengths of the two lines l', $l' = \{p_1, p\}$, and l'', $l'' = \{p, p_2\}$, having end points p_1 and p, and p and p_2, respectively. Two lines l_1 and l_2 are *colinear* if and only if the end points of l_1 and l_2 are all coincident with a line determined by two of these end points. Notice that colinearity is an *equivalence* relation for lines.

Every shape is specified by a finite set of lines, no two of which can be combined to form a single line. Two unequal lines combine to produce one whenever
(1) the two lines share an end point and the remaining end point of one line is coincident with the other line;
(2) both end points of one line are coincident with the other line;
(3) one end point of each line is coincident with the other line; or
(4) the two lines share an end point and this point is coincident with the line formed by the two remaining unshared end points.

In these four cases, the two lines are colinear. The elements in the set of lines specifying a shape are called *maximal lines*, as they are not parts of longer lines in the shape. The shape specified by a set of maximal lines can be represented graphically by drawing the lines in the set. The shape specified by the set containing no maximal lines is called the *empty shape*, and is denoted by s_ϕ. Intuitively, the empty shape is a blank space.

This method of specifying shapes is used to ensure that the specification of a shape is unique. Because a line can always be represented as a collection of multiple colinear line elements, if we were to allow any finite set of lines to specify a shape, then every shape but the empty shape would have an unlimited number of different specifications. In this case, the set of maximal lines for a shape would be the smallest set of lines specifying it.

Subshape and identity relations for shapes
One shape is a subshape (part) of another shape whenever every line of the first shape is also a line of the second shape. More precisely, a line is *in* a shape if and only if its end points are coincident with a maximal line of the shape. Thus, a shape s_1 is a *subshape* of a shape s_2 (denoted by $s_1 \leq s_2$) if and only if each maximal line of s_1 is in s_2. Notice that the empty shape is a subshape of every shape.

It is easy to see that every shape but the empty shape has an unlimited number of distinct subshapes. Take any maximal line of a shape s. Any two distinct points coincident with this line form a new line that is a subshape of s. Now there are an unlimited number of distinct pairs of such points and, hence, an unlimited number of distinct subshapes of s. Further, any finite combination of such lines defined in possibly different maximal lines of s is also a subshape of s.

Two shapes are identical whenever they have the same lines. More precisely, the shapes s_1 and s_2 are *identical* (denoted by $s_1 = s_2$) if and only if each is a subshape of the other. In this case, the sets of maximal lines specifying the shapes are equal.

When shape equality is defined by the shape identity relation, the subshape relation is a *partial order* on sets of shapes. The set of all subshapes of a given shape and the subshape relation determine a *lattice*. This particular lattice is infinite in extent; it is a complete representation of the interrelations between all of the possible component elements of the shape. A finite lattice is determined by the set of shapes containing the empty shape, the maximal lines for a given shape, and all of the shapes specified by some combination of these lines, together with the subshape relation.

Boolean operations for shapes
The *shape union* of shapes s_1 and s_2 (denoted by $s_1 + s_2$) is the shape consisting of all of the lines in s_1 or s_2 or produced by combining lines in s_1 or s_2. A maximal line of s_1 and a maximal line of s_2 can combine to form a new, longer maximal line in their shape union. Thus, there may be lines in this shape that are in neither s_1 nor s_2. The shapes s_1 and s_2 are both subshapes of the shape $s_1 + s_2$.

The *shape intersection* of shapes s_1 and s_2 (denoted by $s_1 \cdot s_2$) is the shape consisting of just those lines in both s_1 and s_2. The shape $s_1 \cdot s_2$ is a subshape of the shape s_1 and a subshape of the shape s_2.

The *shape difference* of shapes s_1 and s_2 (denoted by $s_1 - s_2$) is the shape consisting of just those lines in s_1 that are not also lines in s_2. The shape $s_1 - s_2$ is always a subshape of the shape s_1 but need not be a subshape of the shape s_2.

The operations of shape union, intersection, and difference treat shapes in the same basic way as the set-theoretic operations of union, intersection, and difference treat sets. More precisely, the set of all subshapes of a given shape s and the operations of shape union and intersection form a *Boolean algebra*. That is, shape union and intersection are both closed on this set. These operations are commutative, and each is distributive relative to the other. The empty shape is the identity element for shape union; the shape s is the identity element for shape intersection. The complement $\bar{t}$ for any shape t in the set is given by the shape difference of s and t, that is, $\bar{t} = s - t$. As $\bar{t}$ is a subshape of s, it is also in the set. Notice that the finite set of shapes containing the empty shape, the maximal lines for a given shape s, and all of the shapes specified by some combination of these lines, together with the operations of shape union and intersection also forms a Boolean algebra.

Transformations of shapes
The euclidean transformations provide for new shapes to be produced by changing the location, orientation, reflection, or size of a given shape. These *transformations* are *translation*, *rotation*, *reflection*, *scale*, or finite *compositions* of them. A transformation that does not involve scale is called an *isometry*.

A transformation τ of a shape s is the shape denoted by $\tau(s)$. Any transformation of the empty shape is the empty shape.

Two shapes are geometrically similar when one can be changed into the other by a transformation. More precisely, a shape s_1 is *similar* to a shape s_2 if and only if there is a transformation τ such that $\tau(s_1)$ is identical to s_2. The shapes s_1 and s_2 are *congruent* if and only if the transformation τ is an isometry.

The sets of shapes S^+ and S^*

A finite set of shapes may be used as the vocabulary for the formation of other shapes. It is said that a shape is *made up* of elements in a given set of shapes whenever it is the shape union of transformations of shapes in this set.

The set of all shapes made up of shapes in a given set of shapes S is denoted by S^+. In mathematical terminology, the set S^+ is the least set containing all of the shapes in the set S that is closed under shape union and the transformations. For example, if the set S contains only one shape consisting of a single straight line, then the set S^+ contains all possible shapes made up of one or more maximal lines. Any such shape is just the shape union of its maximal lines, which are transformations of the shape (line) in the set S, and, hence, is an element in the set S^+. For a given set of shapes S, the set of shapes S^* contains in addition to all of the shapes in the set S^+ the empty shape s_ϕ.

Labelled shapes

Aspects of a shape can be distinguished by labelling it. The following simple definitions are needed for this purpose.

A *labelled point* $p{:}A$ is a point p with a symbol A associated with it. Two labelled points $p_1{:}A_1$ and $p_2{:}A_2$ are the same if and only if the points p_1 and p_2, and the symbols A_1 and A_2 are the same. A transformation τ of a labelled point $p{:}A$ is the labelled point $\tau(p){:}A$, where $\tau(p)$ is the point produced by applying τ to p. The symbol associated with a labelled point is invariant under the transformations. A transformation τ of a set of labelled points P is the set of labelled points $\tau(P)$ produced by applying the transformation τ to each labelled point in P.

A labelled shape consists of two parts: a shape and a set of labelled points. More precisely, a *labelled shape* σ is given by an ordered pair $\sigma = \langle s, P\rangle$, where s is a shape and P is a finite set of labelled points. The labelled points in the set P are located with respect to the shape s. These labelled points may be coincident with the lines in s, but this need not be the case. A labelled shape σ can be represented graphically by drawing the shape s, and indicating the occurrences of the labelled points in the set P.

The labelled shape consisting of a shape s but with no symbols associated with it is denoted by $\langle s, \emptyset\rangle$, where $\emptyset$ is the set of labelled points containing no elements. Symbols may be associated with the empty shape s_ϕ to produce a labelled shape $\langle s_\phi, P\rangle$, where P is a nonempty set of labelled points. The *empty labelled shape* is given by $\langle s_\phi, \emptyset\rangle$ and corresponds to a blank space.

Relations and operations on shapes can be extended to labelled shapes:
For labelled shapes σ_1 and σ_2 given by $\sigma_1 = \langle s_1, P_1\rangle$ and $\sigma_2 = \langle s_2, P_2\rangle$, σ_1 is a *subshape* of σ_2 (denoted by $\sigma_1 \leq \sigma_2$) if and only if the shape s_1 is a subshape of the shape s_2, and the set P_1 is a subset of the set P_2. The two labelled shapes σ_1 and σ_2 are *identical* (denoted by $\sigma_1 = \sigma_2$) if and only if each is a subshape of the other.

The *shape union* of σ_1 and σ_2 (denoted by $\sigma_1 + \sigma_2$) is the labelled shape consisting of the shape union of s_1 and s_2 and the union of the sets P_1 and P_2. That is, $\sigma_1 + \sigma_2 = \langle s_1 + s_2, P_1 + P_2\rangle$.

The *shape intersection* of σ_1 and σ_2 (denoted by $\sigma_1 \cdot \sigma_2$) is the labelled shape consisting of the shape intersection of s_1 and s_2 and the intersection of the sets P_1 and P_2. That is, $\sigma_1 \cdot \sigma_2 = \langle s_1 \cdot s_2, P_1 \cdot P_2\rangle$.

The *shape difference* of σ_1 and σ_2 (denoted by $\sigma_1 - \sigma_2$) is the labelled shape consisting of the shape difference of s_1 and s_2 and the difference of the sets P_1 and P_2. That is, $\sigma_1 - \sigma_2 = \langle s_1 - s_2, P_1 - P_2 \rangle$.

A transformation τ of a labelled shape σ, $\sigma = \langle s, P \rangle$, is the labelled shape $\tau(\sigma)$ given by $\langle \tau(s), \tau(P) \rangle$. Two labelled shapes are *similar* if there is a transformation that makes one identical to the other. If the transformation is an isometry, then the labelled shapes are *congruent*.

The sets of labelled shapes $(S, L)^+$ and $(S, L)^*$
Labelled shapes can be formed from given vocabularies of shapes and symbols. It will be said that a labelled shape σ is *made up* of shapes in a set S and symbols in a set L whenever it has one of the following three forms:
(1) $\sigma = \langle s, \emptyset \rangle$, where s is a shape in the set S^+;
(2) $\sigma = \langle s_\phi, P \rangle$, where P is a finite, nonempty set of labelled points in which any symbol associated with a point is an element of the set L; or
(3) $\sigma = \langle s, P \rangle$, where s is a shape and P is a set of labelled points satisfying conditions (1) and (2), respectively.
The set of all labelled shapes made up of shapes in the set S and symbols in the set L is denoted by $(S, L)^+$. Notice that if S contains only one shape consisting of a single straight line, then the set $(S, L)^+$ contains all possible shapes made up of one or more maximal lines and labelled in all possible ways with symbols from L. The set of labelled shapes $(S, L)^*$ contains in addition to all of the labelled shapes in the set $(S, L)^+$ the empty labelled shape $\langle s_\phi, \emptyset \rangle$.

Families of shapes
So far, we have considered only individual shapes. A family of shapes can be defined in terms of a given shape by allowing its component elements to be dimensioned in accordance with certain specified criteria.

More precisely, a *family* of shapes is defined by a *parameterized shape* s, which is obtained by allowing the coordinates of the end points of the maximal lines in a given shape to be variables. A particular member of this family is determined by an *assignment* g of real values to these variables. These values may be required to satisfy certain specified conditions. The result of applying the assignment g to the parameterized shape s is the shape denoted by $g(s)$.

Parameterized shapes and their assignments may be considered as generalizations of transformations applied to shapes. In addition to allowing the locations, orientations, reflections, and sizes of shapes to be changed, parameterized shapes provide for shapes to be distorted in certain ways. Assignments to parameterized shapes can, in general, vary any spatial aspect of a shape, for example, angles and intersections of lines and the ratios between lengths of lines, so long as lines remain straight.

Families of labelled shapes can be defined by parameterized labelled shapes. A *parameterized labelled shape* σ is given by $\sigma = \langle s, P \rangle$, where s is a parameterized shape, and P is a finite set of labelled parameterized points. A *labelled parameterized point* p:A is a labelled point where the coordinates of p are variables. A member of the family of labelled shapes defined by σ is determined by an assignment g of real values to all of the variables associated with end points of maximal lines in s or elements in P. These values may be required to satisfy certain specified conditions. The labelled shape produced by applying g to σ is given by $g(\sigma) = \langle g(s), g(P) \rangle$.

The shape grammar formalism
The shape grammar formalism allows for *algorithms* to be defined directly in terms of labelled shapes and parameterized labelled shapes. Each such algorithm defines a language of shapes.

Shape grammars

A *shape grammar* has four components:

(1) S is a finite set of shapes;

(2) L is a finite set of symbols;

(3) R is a finite set of *shape rules* of the form $\alpha \rightarrow \beta$, where α is a labelled shape in $(S, L)^+$, and β is a labelled shape in $(S, L)^*$; and

(4) I is a labelled shape in $(S, L)^+$ called the *initial shape*.

In a shape grammar, the shapes in the set S and the symbols in the set L provide the building blocks for the definition of shape rules in the set R and the initial shape I. Labelled shapes generated using the shape grammar are also built up in terms of these primitive elements.

A shape rule consists of two labelled shapes, one on each side of the arrow. These labelled shapes and the initial shape I are made up of shapes in the set S and symbols in the set L. Neither the labelled shape on the left-hand side of a shape rule nor the initial shape are allowed to be the empty labelled shape $\langle s_\phi, \emptyset \rangle$, but the labelled shape on the right-hand side of a shape rule is so allowed. Shapes in shape rules and the initial shape are labelled to help guide the shape generation process.

A shape rule $\alpha \rightarrow \beta$ *applies* to a labelled shape γ when there is a transformation τ such that $\tau(\alpha)$ is a subshape of γ, that is, $\tau(\alpha) \leq \gamma$. Unless stated otherwise, it is assumed that τ is a general transformation. In this case, α is similar to some part of γ. However, one reserves the right to restrict τ to special kinds of transformations. For example, one may require τ to be an isometry. In this case, α is congruent to some part of γ.

The labelled shape produced by applying the shape rule $\alpha \rightarrow \beta$ to the labelled shape γ under the transformation τ is given by $[\gamma - \tau(\alpha)] + \tau(\beta)$. This labelled shape is formed by *replacing* the occurrence of $\tau(\alpha)$ in γ with $\tau(\beta)$. That is, one first takes the shape difference of γ and $\tau(\alpha)$, and then takes the shape union of this labelled shape and $\tau(\beta)$. Notice that the application of the shape rule has the effect of erasing the occurrence of $\tau(\alpha)$ in γ whenever β is the empty labelled shape.

Labelled shapes are generated by a shape grammar by applying the shape rules one at a time to the initial shape or to labelled shapes produced by previous applications of shape rules. A given labelled shape γ is *generated* by the shape grammar if there is a finite series of labelled shapes beginning with the initial shape and ending with γ such that each term in the series but the first is produced by applying a shape rule to its immediate predecessor.

A shape grammar defines a set of shapes called a *language*. This language contains all of the shapes s generated by the shape grammar that have no symbols associated with them, that is, labelled shapes of the form $\langle s, \emptyset \rangle$. Each of these shapes is derived from the initial shape by applying the shape rules; each is made up of shapes or subshapes of shapes in the set S.

(Shape grammars can also be used to define languages of labelled shapes, for example, languages containing certain kinds of architectural plans or mathematical diagrams. In this case, the definition of a shape grammar is extended to have two disjoint sets of symbols L_1 and L_2. Symbols in L_1 are nonterminal or auxiliary ones; symbols in L_2 are terminal ones. Shape rules are defined in terms of the symbols in the set $L_1 + L_2$. Labelled shapes in the language defined by the shape grammar have only terminal symbols associated with them.)

The definition of shape grammars is made clear by a simple example. Consider the shape grammar of figure 1 which is specified by giving its shape rules and initial shape. In general, when no confusion can result, this simplified method of specification is used. The labelled shapes in the two shape rules and the initial shape are made up of a square and the symbol •.

The labelled shape on the left-hand side of both shape rules consists of a square and the symbol • located at the midpoint of one of its edges. The labelled shape on the right-hand side of the first shape rule consists of this square and another one inscribed in it. Each vertex of the inside square coincides with the midpoint of a different edge of the outside square. The symbol • is located at the midpoint of an edge of the inside square. The labelled shape on the right-hand side of the second shape rule consists of the square on its left-hand side.

The initial shape is a labelled square with the symbol • at the midpoint of one of its edges.

The generation of a shape using the shape grammar of figure 1 is shown in figure 2. The first shape rule is applied to the initial shape in step 1 and to the resulting labelled shape in step 2. This shape rule applies only to labelled shapes that contain a square with the symbol • associated with the midpoint of one of its edges. The shape rule inscribes a square, also labelled in this way, in the labelled square corresponding to its left-hand side and erases the symbol • associated with this square. As a result, the symbol • is always associated with the midpoint of an edge of the most recently inscribed square. Thus, the first shape rule can be applied to the initial shape and to each labelled shape produced during the shape generation process at most one time. Because the labelled shape on the left-hand side of the first shape rule is a subshape of the labelled shape on its right-hand side, it can be applied again to any labelled shape already produced by applying it. The second shape rule is applied in step 3 of the generation. The left-hand side of this shape rule is identical to the left-hand side of the first shape rule, and hence it applies in identical circumstances. The second shape rule erases the symbol • associated with the midpoint of an edge of a square to produce a shape in the language defined by the shape grammar. The other steps in the generation do not contain shapes in this language, as the symbol • is associated

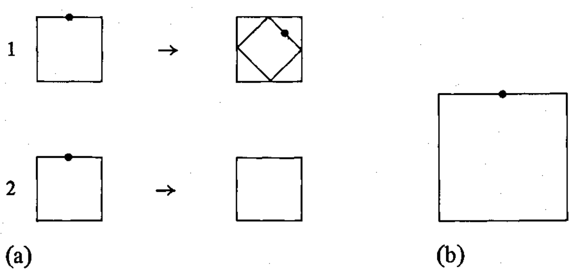


Figure 1. A simple shape grammar that inscribes squares in squares. (a) Shape rules, (b) initial shape.

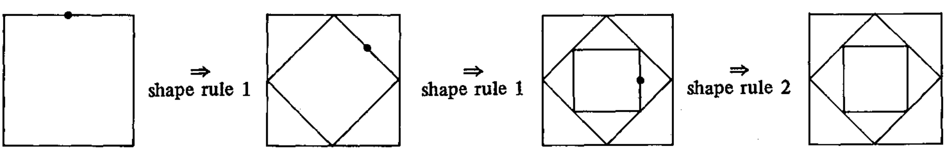


Figure 2. Generation of a shape using the shape grammar of figure 1.

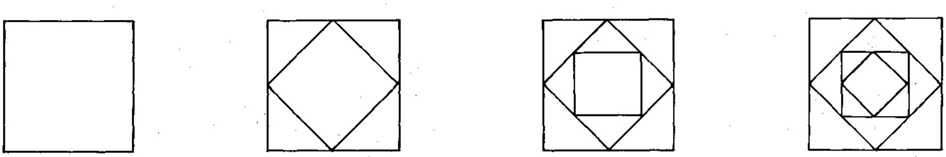


Figure 3. Some shapes in the language defined by the shape grammar of figure 1.

with each of the shapes in these steps. No shape rule can be applied after the second shape rule has been used, because both shape rules require the occurrence of the symbol • to be applied.

The language defined by the shape grammar of figure 1 contains shapes consisting of n ($\geqslant 1$) squares, one inscribed in another. Some of these shapes are shown in figure 3. Notice that the bounding square for all such shapes is always the same.

Parametric shape grammars

Parametric shape grammars are an extension of shape grammars in which shape rules are defined by filling in the open terms in a general schema. A shape rule *schema* $\alpha \rightarrow \beta$ consists of parameterized labelled shapes α and β, where no member of the family of labelled shapes specified by α is the empty labelled shape. Whenever specific values are given to all of the variables in α and β by an assignment g to determine specific labelled shapes, a new shape rule $g(\alpha) \rightarrow g(\beta)$ is defined. This shape rule can then be used to change a given labelled shape into a new one in the usual way. More precisely, if there is a transformation τ that makes $g(\alpha)$ a subshape of the given labelled shape, then this occurrence of the labelled shape $\tau[g(\alpha)]$ can be *replaced* with the labelled shape $\tau[g(\beta)]$. Unless explicitly restricted, τ is a general transformation. It will be said that a shape rule schema applies to a labelled shape whenever it defines a shape rule that applies to the labelled shape.

The implications of these generalizations are illustrated in the following simple example. Consider the parametric shape grammar given in figure 4, which may be viewed as a generalization of the shape grammar defined in figure 1. Where the shape grammar generates shapes by inscribing squares in squares, the parametric shape grammar generates shapes by inscribing convex quadrilaterals in convex quadrilaterals.

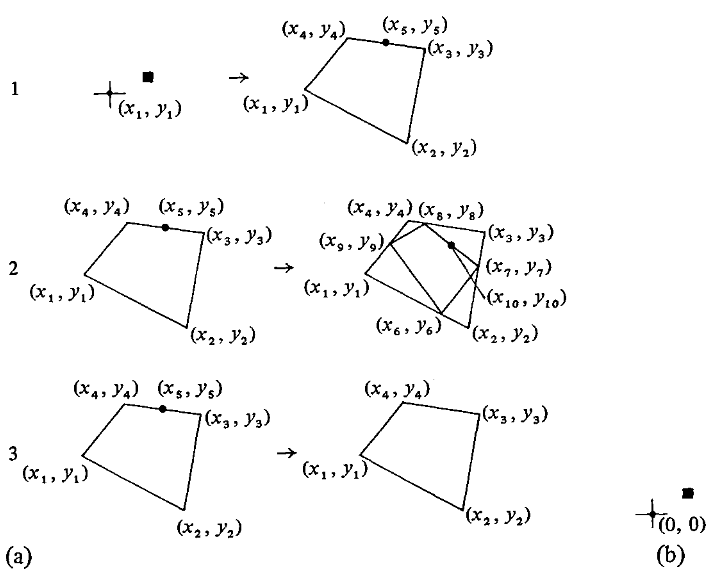


Figure 4. A simple parametric shape grammar that inscribes convex quadrilaterals in convex quadrilaterals. (a) Shape rule schemata, (b) initial shape.

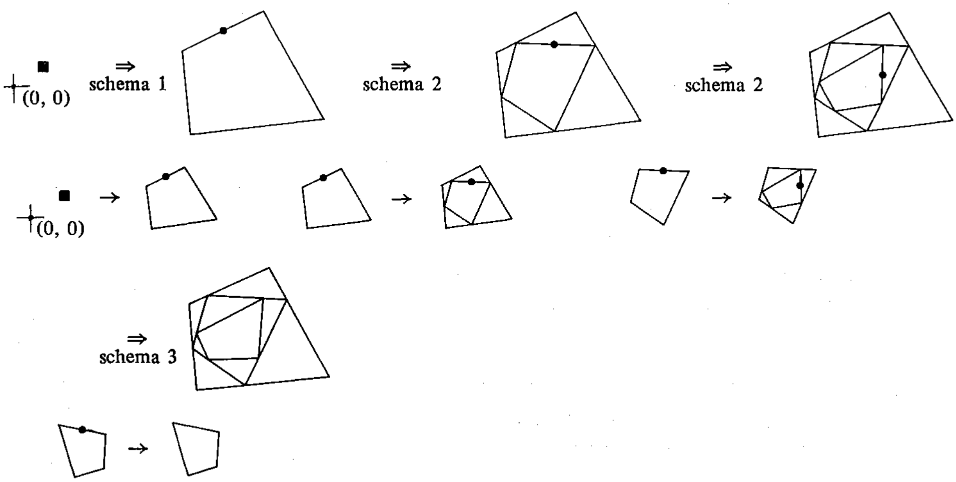
The parametric shape grammar contains three shape rule schemata:

1. The first schema defines shape rules that replace a point labelled by the symbol ■ with a convex quadrilateral having one of its edges labelled by the symbol •. The left-hand side of the schema consists of a labelled parameterized point (x_1, y_1):■; the right-hand side consists of a parameterized quadrilateral q with vertices at the points (x_1, y_1), (x_2, y_2), (x_3, y_3), and (x_4, y_4) and a labelled parameterized point (x_5, y_5):•.

Values assigned to the variables in the schema satisfy these conditions:
(a) The points (x_1, y_1), (x_2, y_2), (x_3, y_3), and (x_4, y_4) are the vertices of a convex quadrilateral.
(b) The point (x_5, y_5) is the midpoint of the line with end points (x_3, y_3) and (x_4, y_4).
2. The second schema defines shape rules that inscribe one convex quadrilateral in another one so that each of the vertices of the inside one is coincident with a different edge of the outside one. The left-hand side of the schema consists of the parameterized quadrilateral q and the labelled parameterized point (x_5, y_5):•; the right-hand side consists of the parameterized quadrilateral q, another parameterized quadrilateral r with vertices at the points (x_6, y_6), (x_7, y_7), (x_8, y_8), and (x_9, y_9), and a labelled parameterized point (x_{10}, y_{10}):•. Values assigned to these new variables satisfy these conditions:
(c) The points (x_6, y_6), (x_7, y_7), (x_8, y_8), and (x_9, y_9) are coincident with the lines having end points (x_1, y_1) and (x_2, y_2), (x_2, y_2) and (x_3, y_3), (x_3, y_3) and (x_4, y_4), and (x_4, y_4) and (x_1, y_1), respectively, but are not these points.
(d) The point (x_{10}, y_{10}) is the midpoint of the line with end points (x_7, y_7) and (x_8, y_8).
Conditions (a) and (c) guarantee that any two quadrilaterals determined by assigning values to the variables in q and r are both convex.
3. The third schema defines shape rules that erase the symbol • from the edge of a convex quadrilateral. The left-hand side of the schema consists of the parameterized quadrilateral q and the labelled parameterized point (x_5, y_5):•; the right-hand side consists of the parameterized quadrilateral q.

The initial shape of the parametric shape grammar consists of the labelled point (0, 0):▪. Notice that the initial shape is a labelled shape and not a parameterized labelled shape.

The generation of a shape using the parametric shape grammar of figure 4 is shown in figure 5. The first schema is applied to the initial shape in step 1. The schema is used to define the shape rule shown beneath the step. This shape rule changes the initial shape into a labelled convex quadrilateral. The second schema is applied in steps 2 and 3 to the labelled shapes produced in steps 1 and 2. The schema is used to define the shape rules shown beneath steps 2 and 3. Each of these shape rules has a left-hand side that is similar to the quadrilateral with an edge marked by the symbol • in the labelled shape to which it is applied. The shape rule inscribes a new quadrilateral



with an edge marked by the symbol • in the distinguished quadrilateral and erases the symbol • associated with it. Notice that the recurrence of the symbol • in the schema prevents it from being used to inscribe more than one quadrilateral in any given quadrilateral. For any shape rule defined by the second schema, there are other shape rules, also defined by the schema, with left-hand sides similar to the labelled quadrilateral in the right-hand side of the original shape rule. Consequently, the second schema can be applied again to any labelled shape already produced by applying it. The third schema is applied in step 4 to the labelled shape produced in step 3. The schema is used to define the shape rule shown beneath the step. This shape rule terminates the shape generation process by erasing the symbol •.

The language defined by the parametric shape grammar of figure 4 contains shapes consisting of n ($\geqslant 1$) convex quadrilaterals, one inscribed in another. The shape generated in figure 5 as well as the shapes drawn in figure 6 are members of this language. Notice that such shapes are not bounded by a fixed quadrilateral.

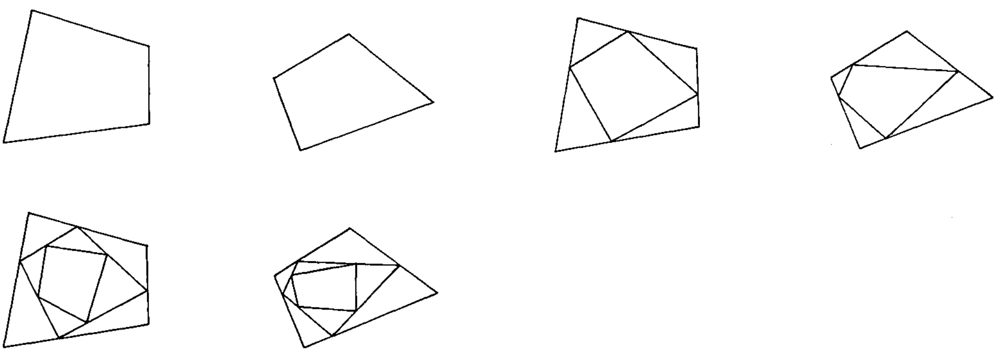


Figure 6. Some shapes in the language defined by the parametric shape grammar of figure 4.

Discussion

The division between shape grammars and parametric shape grammars reflects the usual separation between euclidean (similarity) transformations and other more general ones traditional in the study of spatial patterns using symmetry groups. As a result, shape grammars are best used to define languages of shapes with proportional relationships determined by arithmetic or geometric series. The more general kinds of transformations allowed in parametric shape grammars preserve straight lines, but may vary their relative dimensions and the angles between them. Parametric shape grammars can thus be used to define languages of shapes with proportional relationships determined in any way whatever. To appreciate fully the value of parametric shape grammars in this sense, the reader is invited to find a shape grammar that generates just the shapes in the language defined by the parametric shape grammar of figure 4. Of course, special kinds of shape grammars involving, for example, just affine transformations or just perspective transformations can be defined without difficulty.

Acknowledgements. Preparation of this paper was supported in part by a grant from the Science Research Council. The figures were prepared by Hank Koning and Julie Eizenberg.