A Theoretical Framework for the Analysis and Derivation of Orthogonal Building Plans and Sections

E-G. Vakalo and A. Fahmy
College of Architecture & Urban Planning
The University of Michigan
USA.

1 ABSTRACT

Architects are generally perceived as “Formgivers with an extraordinary gift” (Ackerman, 1980:12). Implicit in this statement is the belief that the operations that architects employ to compose their designs are the product of a creative faculty that is beyond the reach of rational discourse, and thereby cannot be subjected to logical investigation. This view is detrimental to the advancement of knowledge about architectural composition and adversely affects both practice and education in architecture. More specifically, it prevents the architectural community from acquiring of a more refined conception about how architects derive their designs.

In contrast to this view, this study demonstrates that architectural form-making is amenable to logical analysis. In specific, this is to be done through a theoretical and computational framework that describe and explain the tasks involved in the making of orthogonal building plans and sections. In addition to illustrating the susceptibility of architectural form-making to logical analysis, the frameworks proposed in this study overcome the limitations of previously established theories that deal with architectural form-making. These can be divided into two categories: normative and positive theories.

Normative theories include architectural treatises and manifestos. A major limitation of normative theories is that they have limited explanatory power. Their concern is with promoting a specific aesthetic ideology and prescribing rules that can be used to derive compositions that conform to it. Therefore, they cannot be used to explain form-making in general. Positive frameworks, such as shape grammar, rely on rules to describe derivation and analysis processes. Nevertheless, they do not provide a comprehensive description of the tasks involved in architectural form-making. This causes the relation between the rules and compositional tasks to be ambiguous. It also affects adversely the ability of these frameworks to provide architects with a complete understanding of the role of compositional rules in derivation or analysis processes.

2 THE THEORETICAL AND COMPUTATIONAL FRAMEWORK

The theoretical framework suggests that the process of deriving orthogonal building plans and sections is a rational and objective activity. The process of deriving orthogonal building plans and sections is objective because it is amenable to logical analysis. It is rational because it involves the conscious application of a specific set of operations with the intent of accomplishing two immutable tasks: the grouping of the building’s functional units and the creation of architectural space.

The grouping of functional units is regarded to be a top-down activity that occurs at the level of the geometric structure. It involves the division of the overall configuration of orthogonal plans or sections into rectangles and squares that represent the functional units. The creation of architectural space is considered to occur at the level of the spatial structure and to entail the establishment of spatial relations between the elements that define architectural space.
The computational framework proposed in this study derives from the theoretical framework. It consists of a geometric and a spatial grammar. The rules of the geometric grammar apply to the lines of the geometric structure. The rules of the spatial grammar apply to the elements that define architectural space (e.g., walls, columns, beams, and slabs). The geometric grammar describes explicitly the spatial relations among the lines of the geometric structure and enumerates the operations involved in the grouping of the functional units. The spatial grammar describes explicitly the spatial relations among the edges and corners of the spatial structure and enumerates the operations involved in the creation of architectural space.

2 THE GEOMETRIC GRAMMAR

The geometric grammar is a quadruple \((V, \sum, R, I)\). \(V\) is a finite set of shapes that contains the shapes produced by applying the shape rule schemata to the initial shape. Initially, it consists of the initial shape. \(\sum\) is a dot located at the lower left corner of the shape to which the shape rule schemata are applied. \(I\) is the initial shape: a square having the width of one unit (Figure 1). \(R\) is a set that consists of four rule schemata.

\[
\text{Shape Rule Schema R1:} \quad X \quad Y \\
\text{Nlv= 1, Nlh=1, Dx=(1/2), Dy=(1/2)}
\]

\[
\text{Shape Rule Schema R2:} \\
\text{Nlv= 1, Nlh=1, Dx=(1/2), Dy=(1/2)}
\]

\[
\text{Shape Rule Schema R3:} \\
\text{Nlv= 0, Nlh=1, Dx=(1/2), Dy=(1/2)}
\]

\[
\text{Shape Rule Schema R4:} \\
\text{Nlv= 1, Nlh=0, Dx=(1/2), Dy=(1/2)}
\]

Figure 1: The Proposed Grammar

There exist four rule schemata in \(R\): (1) a start rule schema \(R_1\), (2) an operational rule schema \(R_2\), (3) a label movement rule schema \(R_3\), and (4) a termination rule schema \(R_4\). The start rule schema \(R_1\) applies to the initial shape. It performs a scaling transformation to it and multiplies its width and height by the factors \(X\) and \(Y\), respectively. It also attaches a label to the lower left corner of the resulting shape (Figure 2).

\[
\text{XxL, YxL} \\
\text{XxL, YxL} \\
\text{XxL, YxL}
\]

Figure 2: Examples of Applying Shape Rule Schema R1

Shape rule schema \(R_2\) applies to a labeled shape only once. The shape to which \(R_2\) is applied is called the mother-shape. The shapes produced by \(R_2\) are called the sub-shapes of the mother-shape. \(R_2\) divides the mother-shape into a series of sub-shapes, attaches a string to the sub-shapes, and moves the label to the lower left vertex of the sub-shape sharing the lower left vertex of the mother-shape. To divide the mother-shape, \(R_2\) adds rectangles and squares that represent the functional units to it. The strings attached to the sub-shapes are intended to distinguish clearly between them. The movement of the label indicates the sub-shape to which the next rule will be applied.

The application of \(R_2\) is governed by 4 parameters: \(Nlv, Nlh, Dx, \) and \(Dy\). \(Nlv\) and \(Nlh\) specify the number of vertical and horizontal lines to be added to the mother-shape, respectively. \(Dx\) is a \(k\)-tuple of real numbers, such that \(k=Nlv\). The entries in \(Dx\) specify the ratios of division for the vertical lines to be added to the mother-shape.
Dy is a n-tuple of real numbers, such that n = Nh. The entries in Dy specify the ratios of division for the horizontal lines to be added to the mother-shape.

The ordering of the sub-shapes is based on the relations β, E and Π. The relation β is defined as follows: (1) if the y components of the upper left vertices of two sub-shapes A and B are equal, (A, B) ∈ β. β is an equivalence relation. In specific, it divides the sub-shapes into disjoint equivalence classes [S1], [S2],..., [Sn] that are marked by a key: the y component of the coordinate positions of the upper left vertices of their member shapes.

The relation E is defined as follows: if and only if the key for [S1] is smaller than the key of [S2], the pair ([S1], [S2]) is included in E. The order of an equivalence class [S], O(S), is equal to the number of equivalence classes with keys smaller than the key of [S]. The relation Π applies only to the members of an equivalence class [S] produced by β. Specifically, if two sub-shapes s1 and s2 are in [S] and the x component of the upper left vertex of s1 is smaller than the x component of the upper left vertex of s2, (s1,s2) is included in Π. The order of a sub-shape within an equivalence class, O[S](s), equals the number of shapes whose upper left corners have x-coordinates smaller than s.

The string attached to a sub-shape results from concatenating the string of the mother-shape, the order of the equivalence class to which it belongs, and its order in the class. For example, consider applying R1 to the initial shape, such that X=9 and Y=9. Consider applying R2 to the resulting labeled shape, such that Nlv=2, Nh=2, Dx = (1/3, 2/3), Dy = (1/3, 2/3) (Figure 3a). The application of R2 creates nine sister-shapes: A, B, C, D, E, F, G, H, and I. A, B, and C belong to the equivalence class [K1], K1=0. D, E and F belong to the equivalence class [K2], K2=3. G, H and I belong to the equivalence class [K3], K3=6. O([K1]), O([K2]), and O([K3]) are 0, 1, and 2, respectively (Figure 3c). O[K1](A), O[K1](B), O[K1](C), O[K2](D), O[K2](E), O[K2](F), O[K3](G), O[K3](H), O[K3](I) are equal to 0, 1, 2, 0, 1, 2, 0, 1 and 2, respectively (Fig. 3d). Finally, the strings attached to A, B, C, D, E, F, G, H, I are 00, 10, 20, 01, 11, 21, 02, 12, 22, respectively (Figure 3e).

Shape rule schema R3 moves the label from the lower left vertex of a sub-shape to the lower left vertex of any sub-shape produced during derivation or analysis process. The application of R3 is governed by the parameter Cs which specifies the string attached to the sub-shape to which the label is to be moved (Figure 4).
2.1 Examples of the Analytical Capabilities of the Grammar

To demonstrate the analytical capabilities of the grammar, it will be used to analyze the geometric structures of three buildings (Figure 5): (1) the plan of the Ideal Church by Serlio, (2) the plan of the Texas House 6 by Hejduk, and (3) the section of the Smith house. The analysis of the Ideal Church, the Texas House 6, and the Smith house are illustrated in Figures 6, 7, and 8, respectively.
Figure 7: The Analysis of the Geometric Structure of the Plan of the Texas House 6 by Hejduk

Figure 8: The Analysis of the Geometric Structure of the Section of the Smith House by Meier
3 THE CELL STRUCTURE

The cell structure is a four-tuple \((E, S, L, A)\). \(A\) includes the vertices of the squares and rectangles constituting the geometric structure of orthogonal building plans or sections. \(L\) is the set of vectors connecting them. \(S\) is a set of squares having the width \(w\), such that the center of each square is coincident to a vertex in \(A\). \(E\) is a set of sets. Each set in \(E\) is composed of three adjacent rectangles, \(r_1, r_2, \text{ and } r_3\), such that: (1) \(r_1\) is adjacent to a square \(s_1\) in \(A\), (2) \(r_3\) is adjacent to a square \(s_2\) in \(A\), (3) \(r_2\) is adjacent to \(r_1\) and \(r_3\), and (4) the centerlines of \(r_1, r_2, \text{ and } r_3\) are collinear. (Figure 9).

![Figure 9: Geometric Structure, Cell Structure, and Cell in the Cell Structure](image1)

The vertices of the cell structure are represented by the twin-tuple \((x, y)\), such that \(x\) and \(y\) are their coordinate positions in a two dimensional coordinate system whose origin is the lower left vertex of the rectangle or square that encloses the geometric structure of the plan or section being analyzed or derived. A vector \(v\) in the cell structure is represented by the four-tuple \((S_x, S_y, E_x, E_y)\). \(S_x\) and \(S_y\) are the coordinate positions of the start point of the vector \(v\). \(E_x\) and \(E_y\) are the coordinate positions of the endpoint of \(v\): \(S_x < E_x\) or \(S_y < E_y\).

The squares and rectangles of the cell structure are represented by the four-tuple \((P_1, P_2, P_3, P_4)\): \(P_1, P_2, P_3, \text{ and } P_4\) are the coordinate positions of their upper left, upper right, lower right, and lower left vertices, respectively. The vectors \(V_1 (P_1, P_2), V_2 (P_3, P_2), V_3 (P_4, P_3), \text{ and } V_4 (P_4, P_1)\) are their upper, right, bottom, and left defining edges (Fig. 4.17), respectively. \(M_1\) and \(M_2\) are the medians of the squares and rectangles of the cell structure. The start point of \(M_1\) is the midpoint of the vector \(V_3\). Its endpoint is the midpoint of \(V_1\). The start point of \(M_2\) is the midpoint of \(V_4\). Its endpoint is the midpoint of \(V_2\) (Figure 10).

![Figure 10: The Points, Edges, and Medians of the Squares and Rectangles of the Cell Structure](image2)

3.1 The Architectural Components of the Cell Structure

Four architectural components constitute the edges and corners of the cell structure of orthogonal architectural compositions in plan and section. They are the I, L, T, and X-shaped components. An I-shaped component is composed of a rectangle \(r\), such that \(r\) is not adjacent to a square in \(S\) (Figure 11a). An L-shaped component is composed of a square \(s\) and two rectangles \(r_1\) and \(r_2\), such that: (1) the top defining vector of \(s\) is the bottom defining vector of \(r_1\) and the right defining vector of \(s\) is the left defining...
vector of \( r_2 \), (2) the right defining vector of \( s \) is the left defining vector of \( r_1 \) and the bottom defining vector of \( s \) is the top defining vector of \( r_2 \), (3) the bottom defining vector of \( s \) is the top defining vector of \( r_1 \) and the left defining vector of \( s \) is the right defining vector of \( r_2 \), or (4) the left defining vector of \( s \) is the right defining vector of \( r_1 \) and the top defining vector is the bottom defining vector of \( r_2 \) (Figure 11 b).

![Figure 11: (a) I-shaped Components and (b) L-shaped Components](image)

A T-shaped component is formed by a square \( s \) and three rectangles \( r_1 \), \( r_2 \), and \( r_3 \), such that: (1) the top defining vector of \( s \) is the bottom defining vector or \( r_1 \), the right defining vector of \( s \) is the left defining vector of \( r_2 \), and the bottom defining vector of \( s \) is the top defining vector of \( r_3 \), (2) the right defining vector of \( s \) is the left defining vector or \( r_1 \), the bottom defining vector of \( s \) is the top defining vector of \( r_2 \), and the left defining vector of \( s \) is the right defining vector of \( r_3 \), (3) the bottom defining vector of \( s \) is the top defining vector or \( r_1 \), the left defining vector of \( s \) is the right defining vector of \( r_2 \), and the top defining vector of \( s \) is the bottom defining vector of \( r_3 \), or (4) the left defining vector of \( s \) is the right defining vector of \( r_1 \), the top defining vector of \( s \) is the bottom defining vector of \( r_2 \), and the right defining vector of \( s \) is the left defining vector of \( r_3 \) (Figure 12).

![Figure 12: T-Shaped Components](image)

An X-shaped component is formed by a square \( s \) and four rectangles, \( r_1 \), \( r_2 \), \( r_3 \), and \( r_4 \) such that: (1) the top defining vector of \( s \) is the bottom defining vector of \( r_1 \), (2) the right defining vector of \( s \) is the right defining vector of \( r_2 \), (3) the bottom defining vector of \( s \) is the top defining vector of \( r_3 \), and (4) the left defining vector of \( s \) is the right defining vector of \( r_4 \) (Figure 13).

![Figure 13: An X-shaped Component](image)

The binary relation \( \delta \) is defined over the vertices of the geometric structure of the orthogonal plan or section being derived or analyzed (e.g., the elements of the set \( A \)): \( \delta \) (\( P_1 \), \( P_2 \)) is true, if \( P_1y > P_2y \), \( P_1y = P_2y \) and \( P_1x < P_2x \), or \( P_1x = P_2x \) and \( P_2y = P_2y \). The order of a point \( P \) in \( A \), \( O(P) \), is equal to the number of points \( P_1 \) in \( A \), such that \( \delta \) (\( P \), \( P_1 \)) is true.

The relation \( \delta \) is employed to derive the strings that distinguish between the architectural components making up the cell structure. The derivation of strings
involves the following operations. First, the relation $\delta$ is applied to the vertices of the geometric structure (e.g., the elements of $A$) underlying the cell structure (Figure 14a). Second, the order of each point $O(P)$ is computed (Figure 14b). Third, a matrix is constructed such that: (1) a column $C_i$ in the matrix represents one point $P$ in $A$, such that $i = O(P) - 1$, and (2) a row $R_j$ in the matrix represents one point $P$ in $A$, such that $j = O(P) - 1$ (Figure 14c).

Fourth, each architectural component in the cell structure is mapped to a cell in the matrix, and the string attached to it is derived (Figure 14d and e). If the component is an L, T, or X-shaped component and the center of its square $s$ coincides with a point $P_i$ in $A$ represented by the column $C_i$ and the row $R_i$, the component is mapped to the cell $C_{ii}$ and the string $i.i$ is attached to it. If the architectural component is an I-shaped component and one of its medians is included in a vector $V(P_i, P_j)$ in $E$, the component is mapped to the cell $C_{ij}$ and the string $i.j$ is attached to it.

Figure 14: (a) The Application of the Relation $\delta$ to the Vertices of the Geometric Structure of the Cell Structure in Figure 10b, (b) the Order of the Points Constituting the Geometric Structure, (c) the Matrix (d) the Mapping of Architectural components to the Cells of the Matrix, and (e) the Strings Attached to the Architectural Components of the Cell Structure in Figure 10b.
The spatial grammar applies to labeled I, L, T, and X-shaped components. A labeled component is the hatched component. The grammar consists of five architectural rule schemata: (1) the operational rule schemata R1 and R2, (2) the label movement rule schema R3, (3) the transformation rule schemata R4 and R5.

Architectural rule schema R1 is concerned with inserting voids into labeled I-shaped components. R1 performs the following operations to a labeled I-shaped component: (1) divides it into a series of I-shaped components, and assigns to each of the produced components the value 0 or 1, (4) if a component is assigned the value 0, it is considered to be void, and (5) if a components is assigned the value 1, it is considered to be solid.

Architectural rule schema R1 divides a labeled I-shaped component by adding to it vertical or horizontal vectors. If the magnitude of the median M2 of the I-shaped component to which R1 is applied is larger than the magnitude of its median M1, a set of vertical vectors are added to it. The magnitude of the added vectors is equal to the magnitude of M1. The start points of the added vectors are coincident to its bottom defining vector; their endpoints, to its top defining vector. If the magnitude of the median M1 of the I-shaped component to which R1 is applied is larger than the magnitude of M2, R1 adds to it a series of horizontal vectors. The magnitude of the added vectors is equal to the magnitude of M2. The start points of the added vectors are coincident to its left defining vector; their endpoints, to its right defining vector.

The application of R1 is governed by 4 parameters: Nd, Rd, Sd, and Nc. Nd specifies the number of added vectors. Rd is a k-tuple of real numbers that are smaller than one, such that k = Nd. Each entry in Rd corresponds to one of the vectors added by R1 and specifies the ratios of division for it. Sd is a k-tuple of binary values, such that k = Nd + 1. Each entry in Sd corresponds to an I-shaped components produced by the intersection of the added vectors with the defining edges of the I-shaped component to which R1 is applied. The entries in Sd can be assigned the values 0 or 1. If an entry is assigned the value 0, the corresponding component is considered to be void. If an entry is assigned the value 1, the corresponding component is considered to be solid. Nc specifies the string attached to the component to which the label is to be moved after the application of R1.

After the application of R1 a virtual I-shaped component is constructed. The vertices of the virtual I-shaped component are coincident with the vertices of the original I-shaped component to which R1 was applied. The virtual component is denoted with the dotted outline. Figure 15 illustrates examples of applying R1.

![Figure 15: Examples of Using Architectural Rule Schema R1](image-url)
Architectural rule schema R2 applies to labeled I, L, T, and X-shaped components. It is concerned with deleting the square and rectangles constituting them. Its application is governed by the parameters Q and Nc. Q is a five tuple of binary values. The first, second, third, fourth, and fifth entries in Q correspond to the square s, the rectangles r1, r2, r3, and r4 making up the component to which the architectural rule R2 is applied, respectively. If an entry is assigned the value 0, the corresponding rectangle or square is considered to be void. If an entry is assigned the value 1, the corresponding square or rectangle is considered to be solid.

If R2 is applied to an I-shaped component only the first entry is used. If R2 is applied to an I-shaped component to which R1 was applied. All the squares and rectangles produced by the application of shape rule schema R1 are deleted. If R2 is applied to a T-shaped component, the first four entries are employed. The fifth entry is assigned the value 0. If R2 is applied to an X-shaped component, five entries are used. Similarly to R1, Nc specifies the string of the component to which the label is to be moved after the application of R2. Examples of the application of architectural rule schema R2 are illustrated in Figure 16.

Architectural rule schema R3 is a label movement rule. Its application is governed by the parameter Nc that specifies the string attached to the component to which the label is to be moved. Initially, the label is attached to the component located at the lower left corner of the cell structure (Figure 17). Labeled components are hatched.

Architectural rule schema R4 applies to labeled I, L, T, and X-shaped components. The application of R4 is governed by four parameters: K, Trx, Try, and Nc. K is an n-tuple of strings, such that n is equal to the number of the rectangles to which R4 is
applied. If R4 is applied to an I-shaped component that was divided using R1, K is assigned the value $\phi$, and the translation is applied to all the rectangles and squares constituting it. If R3 is applied to L, T, and X-shaped components, entries in K can be assigned the values s, r1, r2, r3 and r4. If an entry is assigned the value s, the translation is applied to the square s. If an entry is assigned the value r1, r2, r3, or r4, the translation is applied to the corresponding rectangle. Trx and Try specify the value of the translation in the x and y directions, respectively. Nc specifies the string of the component to which the label is to be moved after the application of the rule. Figure 18 illustrates examples of the application of architectural rule schema R4.

Architectural rule schema R5 scales labeled I, L, T, and X-shaped components. Four parameters control its application: K, Fp, Sx, and Sy, and Nc. K is an n-tuple of strings, such that n is equal to the number of the rectangles to which R5 is applied. If R5 is applied to an L, T, or X-shaped component, entries in K can be assigned the values s, r1, r2, r3, and r4. If an entry is assigned the value s, the scaling transformation is applied to the square s that enter into the composition of the component. If an entry is assigned the value r1, r2, r3, or r4, the scaling transformation is applied to the corresponding rectangle.

Fp determines the fixed point of the scaling operation. If Fp is assigned the value 0, the center of the component to which R5 is applied is considered to be the fixed point of the scaling transformation. If Fp is assigned the value 1, the upper left corner of the square or rectangle to which R5 is applied is considered to be the fixed point of the scaling transformation. If Fp is assigned the value 2, the lower right corner of the square or rectangle to which R5 is applied is considered to be the fixed point of the scaling transformation.

If R5 is applied to an I-shaped component that was divided using R1, the scaling transformation is applied to all the squares and rectangles making up the component, and K is assigned the value $\phi$. Nc specifies the component to which the label is to be moved after the application of the rule (Figure 19). Finally, to terminate the derivation or analysis the variable Cn in R1, R2, R3, R4, and R5 is assigned the $\phi$. 

Figure 18: Examples of the Application of Architectural Rule Schema R4
4.1 The Analysis Abilities of the Architectural Grammar

To demonstrate the analysis abilities of the architectural grammar formulated above, it will be used to analyze Hejduk’s Texas House 4 (Figure 20) and the section of the Smith House by Meier (Fig. 21).
Figure 20: The Analysis of the Spatial Structure of the Plan of Texas House 6
Continuation of Figure 20
Continuation of Figure 20
Continuation of Figure 20
Continuation of Figure 20
Continuation of Figure 20
Figure 21: The Analysis of the Spatial Structure of the Section of Meier's Smith House
Continuation of Figure 21
Continuation of Figure 21
Continuation of Figure 21
Continuation of Figure 21
Continuation of Figure 21
5 CONCLUSION

A theoretical framework that provides a comprehensive description of architectural form-making in plan and section was proposed. On the basis of this framework, a computational framework that describes the operations involved in the creation of the morphological structure of orthogonal building plans and sections was formulated and tested.

An important advantage of the frameworks proposed in this study over normative architectural theories that deal with form-making is that they are generalizable: they explain the process of composing the morphological structure of orthogonal building plans and sections in general. In contrast to shape grammar, the rules of grammars proposed in this study derive from a theoretical framework that provides a comprehensive description of the tasks and operations involved in architectural form-making.

Specifically, there is a one to one correspondence between the tasks identified by the theoretical framework and the ones fulfilled through the application of the grammars. For example, the computational model divides derivation and analysis processes in two steps. The first is concerned with the grouping the functional units and is achieved through the application of the geometric grammar; the second, the creation of architectural space and is achieved through the application of the spatial grammar.

There is also a one to one correspondence between the operations identified by the theoretical framework and the rules of the grammars. For example, the application of the geometric grammar comprises of the top-down application of a division rule to the cells of the geometric structure; the application of the spatial grammars, the application of two types of rules. The first type inserts voids into the edges of the spatial structure. The second applies scaling and translation operations to architectonic elements.

Isolating the tasks involved in the derivation or analysis is important because it eliminates the operations that are irrelevant to the task in hand. For example, at the level of the geometric structure, only the operations that contribute to the realization of the grouping the functional units are considered. This focuses the attention of the user on these operations and provides a clear understanding of their role in the derivation or analysis.
However, it must be noted that in comparison to shape grammar, the application of the grammars proposed in this study includes a large number of computations. This makes derivation and analysis processes error-prone and time consuming. It may also distract the user from focusing on the morphological operations employed in these processes. Nevertheless, it is believed that this limitation can be overcome through a computer program that relieves the user from these tasks.

Finally, the ability of the computational framework to describe the operations involved in the process of creating the morphological structure of orthogonal building plans and sections suggest that this process includes two immutable tasks: the grouping of the functional units and the creation of architectural space. The grouping of the functional units involves the top-down division of the plan or sections; the creation of architectural space, the insertion of voids into architectonic elements and the application of scaling and translation operations to these elements.

In turn, this leads us to believe that architectural form-making is a rational and objective activity. In specific, the susceptibility of the process of creating orthogonal plans and sections to logical analysis provides reasonable ground for believing that architectural form-making in general is amenable to logical analysis. Similarly, the materiality of the tasks and operations constituting the form-making process suggests that it is driven by reason.

6 REFERENCES