



# On Visual-Mechanical Synthesis in Shell Structures

Rizal Muslimin<sup>1</sup> 

Published online: 22 December 2016  
© Kim Williams Books, Turin 2016

**Abstract** While form-finding and optimization methods in shell structures have been widely advanced, discourses regarding an ideation process of how a designer or engineer may synthesize a precedent into novel designs have been less explored, as such ideation mainly relies on his/her intuition in representing structural problems. In this paper, we present a computational method to synthesize shell designs through the interplay between mechanical idealization and visual computation using traditional craft as a reference. Examples of how to generate a folding structure from weaving and tensegrity structure from folding show not only the way to convert craft logic from one type to another in shell structures but also the method to explicitly interpret the designer's ideation process in generating a certain structural design.

**Keywords** Shell · Structure · Shape grammar · Gestalt · Tensegrity · Folding · Weaving · Embedding · Idealization · Optimization

## Introduction

In shell design, structural idealization can be considered a process of schematizing an actual shape into a formal representation to simulate its physical behavior. Structural idealization is followed by an optimization process whereby internal and external conditions are modified to find the desired solution that also satisfies its structural behavior.

---

✉ Rizal Muslimin  
rizal.muslimin@sydney.edu.au

<sup>1</sup> Faculty of Architecture, Design and Planning, the University of Sydney, Darlington, NSW 2008, Australia

Many forms of shell idealization have been applied in architectural design by mimicking a certain precedent. This mimicking strategy assumes that the structural behavior from the precedent's scale could predict physical performance in a larger size, as the load-stress distribution is considered proportional across scale. For instance, structural parameters embedded in natural shapes (e.g., a fish scale, spider web, and insect exoskeleton) or in loaded material (e.g., tensioned soap-film and hanging chains) serve as a schema to simulate the physical behavior of the shape on a larger scale (Hensel et al. 2004; Otto 1973; Isler 1986). By modifying and equating the external and internal energy to this schema, one could receive immediate physical feedback and proportional performance in the design. Various studies and projects have advanced design opportunities on mimetic idealization. D'Arcy Thompson's, Otto's and Isler's structural diagrams, for instance, showcase comprehensive models of how shells' structures from nature or physical models could perform in architecture (Thompson 1992; Isler 1986; Pearce 1978). However, while shell structures have been widely promoted by enumerating a number of possible designs, discourse on the *ideation*, i.e., the manner in which one perceives and determines the ways a shape can be idealized into a shell structure, has been explored less often. The current ideation process mainly relies on designer or engineer intuition in representing structural problems. Otherwise, one would depend on the available methods, such as from Otto's or Isler's physical model or from the expansive reference for biomimicry strategies. Yet vast numbers of references from natural or cultural artifacts does not necessarily mean the ideation process is straightforward, as it depends on the moment one sees a trivial object as a potential precedent to formulate a structural design strategy. Often, the creative process during this moment occurs in a tacit manner, which makes it challenging for others to learn or repeat the success stories.

Moreover, whereas mimetic idealization in shell design lends itself to the knowledge that the precedent shape has been optimized generatively in its environment, the strategy for assembling shell architecture is mainly derived in a top-down fashion, i.e., by tessellating a continuous surface into discrete modules to fit a certain construction technique such as in reinforced-concrete, timber-frame, and brick-masonry. This may be caused by the assumption that, although the load and force distribution within a shape might be proportional in different sizes, the joint mechanism in the precedent (e.g., interwoven fabric mesh or soap film) is not always transferrable across scale. Consequently, the limited number of construction methods at the local configuration (e.g., the joint system) could compromise the boundless design possibilities derived from the form-finding method for the shell's global configuration (i.e., the overall shape).

In this paper, our study focuses more on interpreting the ideation process regarding how one learns from a precedent and generates a new structural reconfiguration. We investigate this issue by harnessing traditional craft as an alternative source to enrich shell design. Some traditional crafts, in this case basket-weaving and paper-folding, maintain similar properties to shell structures in architecture and they are mainly executed with a bottom-up approach by first figuring out how to join and configure the shapes at the local scale before assembling them into a desired product at the global scale. With less constraint on

what the global appearance of the shell would look like, the bottom-up approach could help to produce various emergent shell designs by using different types of crafts as precedents from the local scale. Thus, in our study, we aim less on the quantity of how many parametric designs can be incrementally derived and more on the quality of how unique the new design typologies could be in comparison to their precedents.

## Methods

Our methods focus on the ideation process before the design is idealized and optimized. In generating part of a shell, we rely on the association and disassociation process between seeing a shape and registering its mechanical description. Hence, we use the term *Gestalt Force* to highlight the role of visual perception in investigating a shape apart from its description. We maintain that in order to liberate the structural ideation process, our visual perception needs to be able to work independently yet in close proximity to mechanical registration (Muslimin 2014). In explaining this method, our paper is structured in three sections: first, we will introduce visual embedding, a process of visually embedding a shape from an object and transforming it into a different design using shape grammar rules (Stiny and Gips 1971). Second, we will explain mechanical embedding, i.e., a procedure to register mechanical descriptions into a shape based on the Finite Element idealization procedures (Bathe 1996). Third, we will show examples of how visual and mechanical embedding works together to generate various idealized structural configurations for the shell from traditional craft as the precedent, i.e., weaving and paper folding.

Shape Grammar is a computational design method to analyze and synthesize design with a set of shape rules. In contextualizing a design, shape grammar's descriptive function allows meaning to be attached to the shape, in this case, structural description (Stiny 1981). It is through this mechanism that the synergy between shell typology and shape grammar's methods can be reciprocal. On one hand, the shell's geometry provides a versatile platform for calculating shape across different dimensions. In its physical embodiment, many shell designs were represented in linear and planar shapes, such as with chains and fabric in Gaudi's model, cable nets and soap-film in Otto's experiment, and long lumbars and concrete in Candela's roof constructions (Garlock and Billington 2008). In digital computation, the shell has been explored in poly-dimensional representation. For example, in dome typology through the shape annealing method (Mitchell 1998; Shea and Cagan 1997) and the graphic-static with polyhedral form and force schema (Akbarzadeh 2016), where a line served as a vector in computing the equilibrium of a three-dimensional shape. On the other hand, rule-based iterations were able to enrich new design variations, as demonstrated in a shell's derivative geometry such as the two-dimensional truss via evolutionary algorithm (Mueller 2014), grammar-based Graphic-Static (Lee et al. 2015), and fractal-based trusses (Rian 2014). Most of these methods generated various structural designs with simple rules and schemas

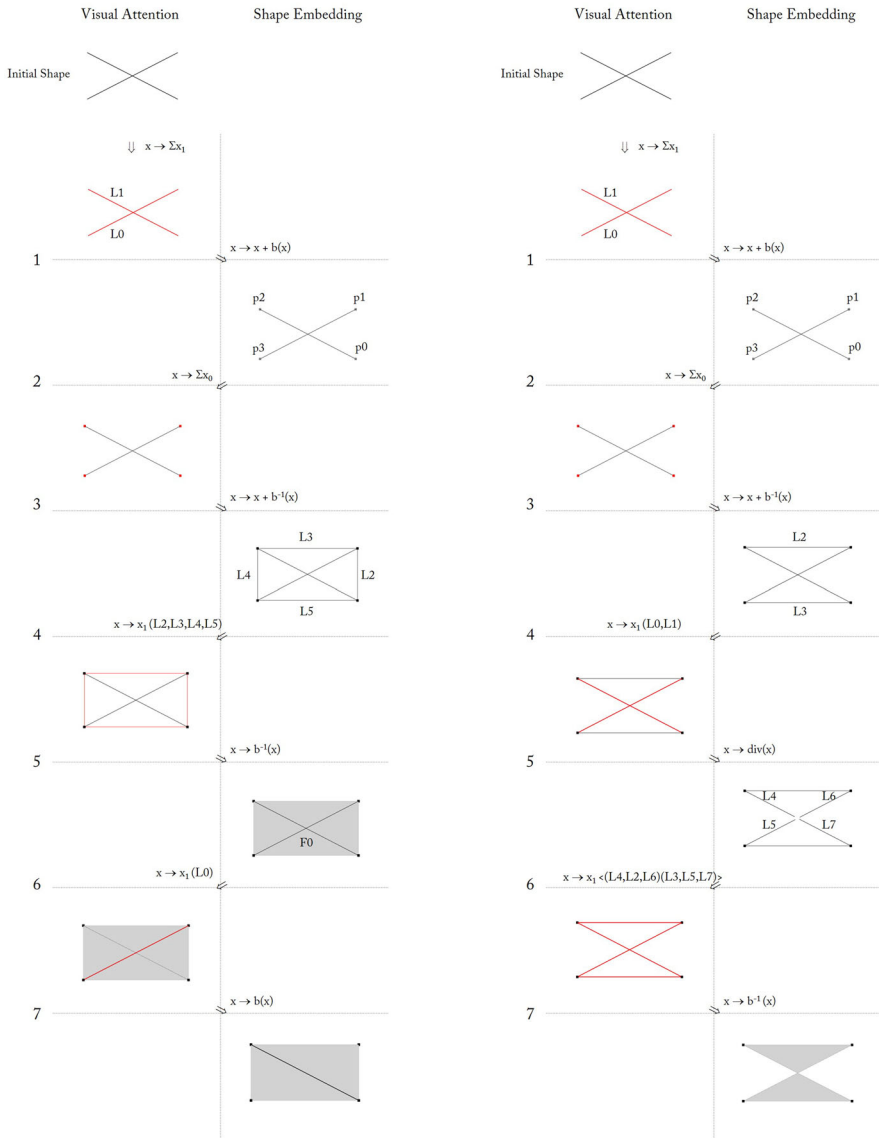
using shapes that have been embedded with predefined mechanical description prior to the optimization process (e.g., supporting points or force-flow lines).

In light of recent generative design approaches in structural optimization, here we highlight a different avenue in which shell designs could be derived beyond the sum of enumerative design iterations using another of shape grammar's key principles, *embedding*; a process to associate and disassociate ways of seeing and describing a shape (Knight and Stiny 2015). The method introduced in this paper is rather uncommon in the way in which shapes are being synthesized across dimensions in a non-linear fashion and along which the mechanical meaning is not tightly attached to the shape, hence, this allows room for ambiguity in perceiving how the shape could structurally behave. By relaxing the shape and meaning relationship in the structural ideation process, the role of shape grammar in this method is to embrace such an ambiguous, yet traceable process, to venture a new kind of structural reconfiguration.

## Visual Embedding

To begin with, we consider a shape symbolized by 'x,' without any mechanical description (Fig. 1). Several reduction rules govern the way shape x is visually computed. To declare which shape is being currently focused on, Identity Rule:  $x \rightarrow x_{ij}$  highlights a shape with variables i and j, respectively representing shape and space dimension. That is,  $x \rightarrow x_{13}$  means 'see one-dimensional line(s) in a three-dimensional space'. If j is not defined, it means see shape  $x_i$  in any dimensional space (Stiny 2008).

A set of boundary rules:  $x \rightarrow b(x)$ , returns a boundary of a shape. Variable  $b(x)$  denotes the boundary of the shape x in the lower dimension (i.e.,  $x_{i-1}$ ). For instance, a boundary of a 3D solid comprises 2D faces; a face's boundaries are lines, and lines' boundaries are points. Rule  $x \rightarrow b^{-1}(x)$  returns the inverse boundary of a shape into a higher dimension (i.e.,  $x_{i+1}$ ). The inverse boundary of two points is a line, three or more connected lines will return a face and so forth. The boundary rule can be nested to skip a level of boundary dimensions. Applying rule  $x \rightarrow b(b(x))$  to a 2D face, for instance, will return the face's vertices instead of lines. However, as the boundary and the shape are independent of one another in the reduction rule, they are not tightly bound (Stiny 2011). For instance, applying a boundary rule,  $x \rightarrow b(x)$ , to two crossing lines will produce four lines' endpoints, but applying the inverse-boundary rule,  $x \rightarrow b^{-1}(x)$ , to these four endpoints will not only retrieve the previous two lines, but also return other lines that share the points as their boundaries (see step 3 in Fig. 1). Correspondingly, because the endpoints are independent from the line, erasing the points does not necessarily mean erasing the line. In addition to boundary rules, part rules  $x \rightarrow prt(x)$  will return part of a shape within the same dimension, for instance, shorter line segments within a continuous line or a bunch of planes within a surface. In the reduction rule, however, part rule can produce many shapes, e.g., different ways of segmented lines overlapping each other in a line. To simplify this, rule  $x \rightarrow div(x)$  splits a shape into  $n$  non-overlapping discrete parts.



**Fig. 1** Visual embedding

Figure 1 shows boundary and inverse-boundary rule iterations on two interlacing lines to derive new shapes. Of the two iterations, the first three steps produce similar results—four points. For the third and fourth steps however, the inverse boundary rule,  $x \rightarrow b^{-1}(x)$ , begins to show different shape compositions as a result of applying the rule only to specific points. The boundary of the four points in step 2 could be four lines (a square) or two triangles. The seventh step of the iterations shows a new shape where the original knot typology has been fully altered, as some

lines have disappeared into endpoints as a result of applying the boundary rule to the lines.

The examples in Fig. 1 show that shape relationship across dimension visual embedding is less tight than that in most structural analysis software. While in the latter, shapes across different dimensions are tightened together, either by integral or derivative shapes, the shape relationship in visual embedding is ambiguous, letting the designer wander the shapes across dimensions and bind and unbind them with the (inverse) boundary shapes as the eye wishes. Hence, by separating the visual attention and shape embedding independently, this creates a many-to-many shape relationship.

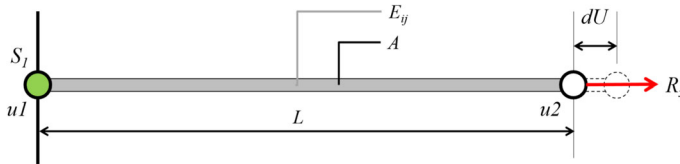
## Mechanical Embedding

In registering the mechanical description for the shape, we use the Finite Element Method (FEM) to represent the way in which an object is being idealized into a mathematical model, i.e., as an assemblage of vertices, edges, and faces that define the object's shape. In *Finite Element Procedures*, Bathe (1996) describes the process of idealizing physical objects into mathematical models in four phases. The geometry phase represents the kinematic shape of a physical object. The material phase assigns material properties to the kinematic shape. The loading phase applies an external load onto the object. The boundary condition defines the degree of freedom at the supporting point of the shape. Here, we represent Bathe's idealization process as a set of descriptive functions to apply a mechanical property to the shape. While  $x$  stands for a shape being visually attended, the function-embedded list of mechanical attributes  $\langle(L, A), E, R, S\rangle$  is as follows:

- Function  $x \rightarrow x + (L, A)$  defines the shape as kinematic parametric schema, where (L) represents the shape's length and (A) represents the shape's cross sectional profile area. Other geometric variables pertaining to its kinematic properties such as trigonometric variables can be included in this function.
- Function  $x \rightarrow x + E_{ij}$  assigns Degree of Material Elasticity (E) or (Young Modulus). The shape is coded by the proportion of strain (i) and stress (j): the greater the (j/i) ratio, the stiffer the material, and vice versa. For simplicity reasons, in this paper, we do not use the actual E value; instead, we define  $E_0$  as a pseudo-elastic material by increasing the shape's degree of freedom with rule  $x \rightarrow \Sigma \text{div}(x)$ , and  $E_1$  as rigid material, by reducing the number of a shape's segments into one with rule  $x \rightarrow \Sigma \text{div}^{-1}(x)$ .
- Function  $x \rightarrow x + R_{xyz}$  applies external force (R) to the endpoints of each line, represented by a vector to indicate the direction and magnitude in which the load is applied, hence,  $x$  is (an array of)  $x_0$ . Vector R's degree of freedom is defined by a transformation space. Gravity, for instance, is noted as  $R_{-z}$  or  $R_{00-1}$  as it applies vertically in z axis.
- Function  $x \rightarrow x + S_{xyz}$  constrains the supporting condition of a shape (S). The support constrains the shape's degree of freedom in space dimension (x, y, z) in responding to external forces R and the internal stiffness defined by (E, A, and L). For instance,  $S_x$  means a shape can only move on the x axis,  $S_{xy}$  means it can

move on a two-dimensional plane, and  $S_{xyz}$  or  $S_0$  means the shape is free to move in a three-dimensional space, otherwise  $S_1$  or fixed support.  $S_{x'}$  means it can rotate along normal vector  $x'$ .

Given these mechanical attributes, the shape's physical behavior can be calculated by satisfying the principle of virtual equilibrium equation to solve the unknown point of displacement ( $dU$ ) in a shape as the following example on a simple one-dimensional tensioned element shows:



where the displacement can be obtained using:

$$\frac{EA}{L} dU = R$$

The mechanical embedding functions show that the visually attended shape ( $x$ ) has to be declared before applying the mechanical description and variables in each function. Figure 2 shows an example of embedding mechanical descriptions into a plain knot using the above rules. The iteration in the middle column identifies shapes that are visually attended by rule  $x \rightarrow x_{ij}$ , and the iteration in the left column shows the embedded mechanical properties of the visually attended shapes.

### Weaving to Tensegrity

With the visual embedding rules and mechanical embedding function set, in the following we show the reconfiguration process by alternating between visual and mechanical embedding while being bridged by identity rule:  $x \rightarrow x_{ij}$ . Here, the identity rule highlights the visual attention in deciding which parts of the shape will be embedded either by visual or mechanical embedding for the subsequent iteration.

The example in Fig. 3 begins with an isometric view of two overlapping rigid lines from a weaving structure as an initial condition to derive different structural configurations. Shapes in the middle columns highlight the visually focused shape, while the shape in the right column shows visual embedding and the left column shows mechanical embedding. The shapes in the far left of the page show the consequences in the overall configuration. Rule  $x \rightarrow x + R_z$  has been applied to the points where two forces go downward and two others go upward  $\langle P_0 - R_{+z}, P_1 - R_{-z}, P_2 - R_{+z}, P_3 - R_{-z} \rangle$ , without support, such that:  $x \rightarrow x_0 + S_0$ , the lines will move away from each other. While the lines are relatively close, the following iteration is applied:

1. Rule  $x \rightarrow \Sigma x_0$  draws visual attention to all the points  $\langle P_0, P_1, P_2, P_3 \rangle$  (the sum means 'select all').

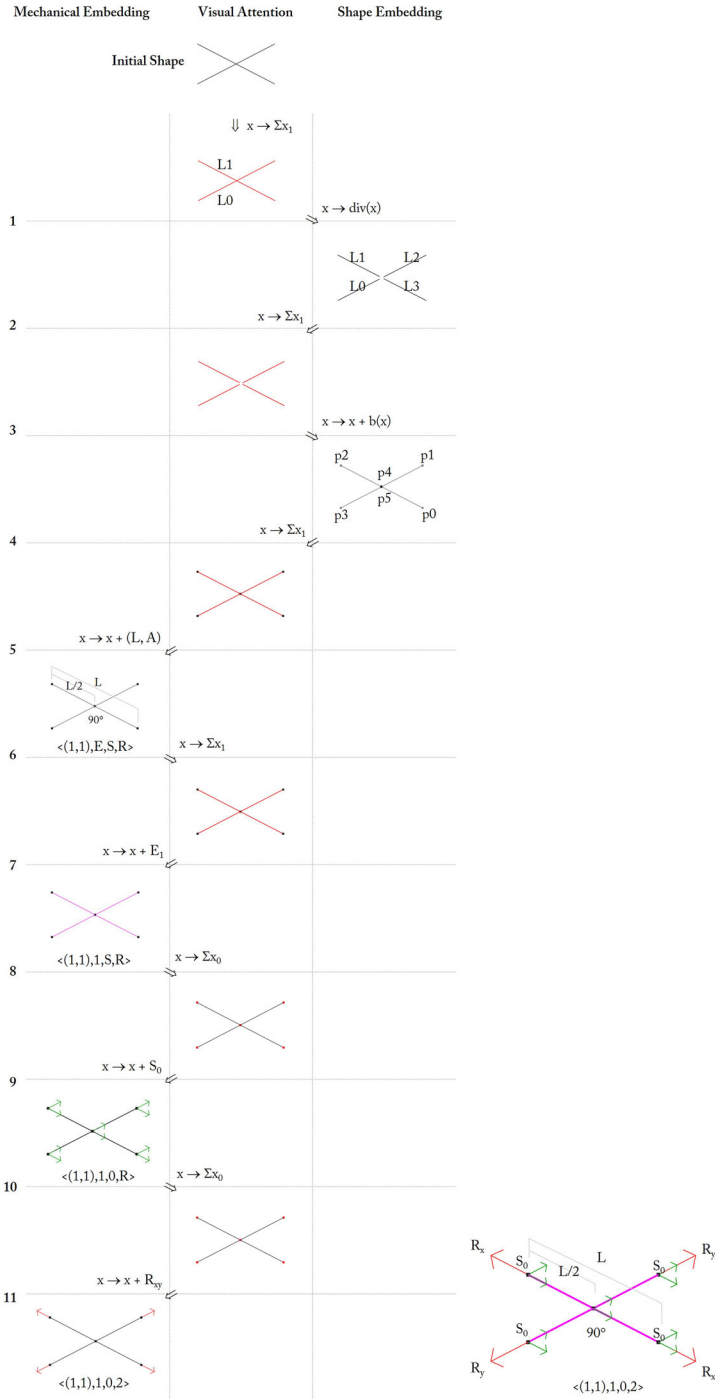


Fig. 2 Mechanical embedding



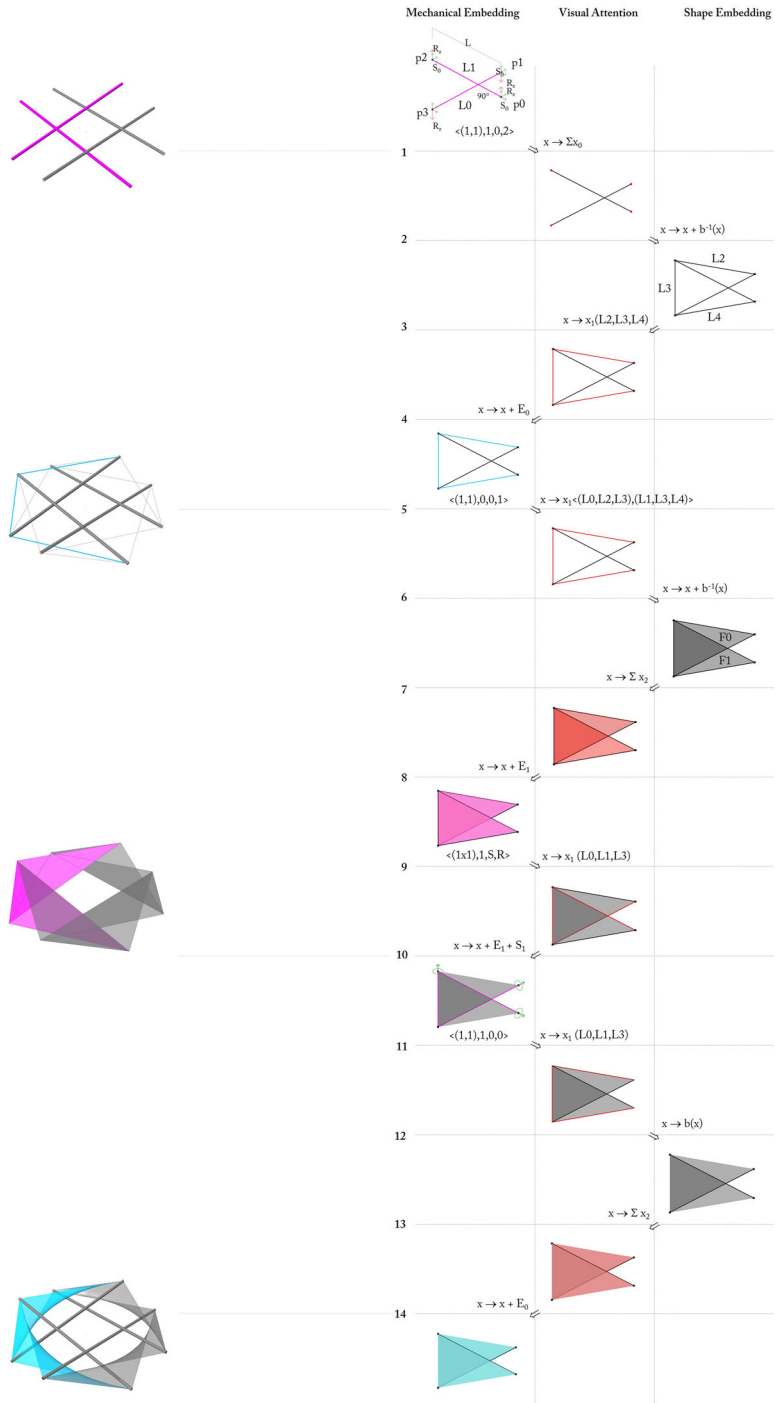


Fig. 3 Visual-mechanical embedding on deriving weaving into tensegrity into folding

2. Rule  $x \rightarrow x + b^{-1}(x)$  retrieves three lines as the inverse boundary of the points so the two lines are now connected by the new lines.
3. Rule  $x \rightarrow x_1$  focuses visual attention on the new lines  $\langle L_2, L_3, L_4 \rangle$ .
4. Rule  $x \rightarrow x + E_0$  applies elastic material to the lines, thus the lines become flexible.

The consequence of this iteration profoundly changes the original weaving physical behavior. The initial lines in the knot which were overlapping one another in a compressive mode are now tensioned by new flexible lines. If we map this new configuration to the weaving composition by substituting the initial knot with the new one, we will have new physical behavior that resembles a tensegrity structure. We can then group these iterations to establish a weaving-to-tensegrity function (G1):  $x_0 \Rightarrow x + b^{-1}(x) \Rightarrow x_1 \Rightarrow E_0$ , or in plain English, ‘in the woven knot, see points, draw new lines from them, now see the new lines and apply elastic material into them.’

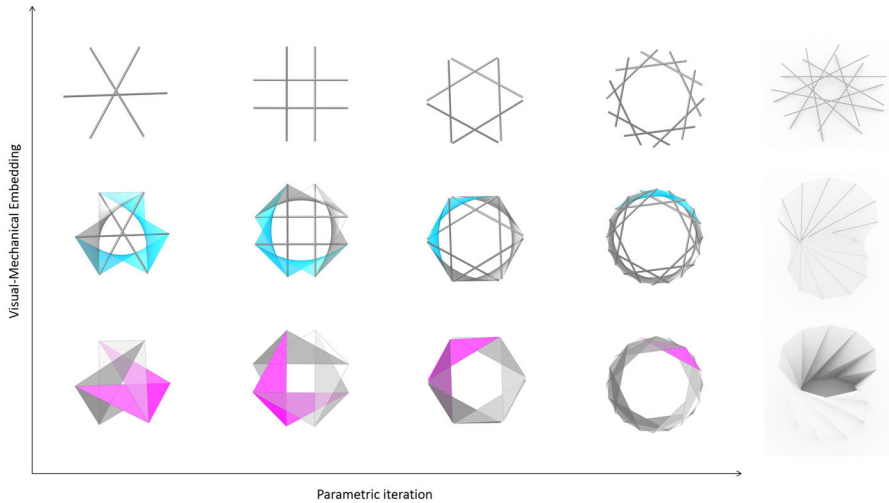
Using this function, the ideation process of turning interwoven designs into a tensegrity structure as demonstrated by Snelson (2003) can be represented more explicitly and, for the same reason, we may continue to alter this structure to develop other different configurations. Next, we continue the iteration from the last state:

5. Rule  $x \rightarrow x_1$  draws visual attention focus to two groups of lines: (L0, L2, L3) and (L1, L3, L4).
6. Rule  $x \rightarrow x + b^{-1}(x)$  integrates the lines into two triangular faces  $\langle F_0, F_1 \rangle$ .
7. Rule  $x \rightarrow \Sigma x_2$  draws attention to all faces.
8. Rule  $x \rightarrow x + E_1$  applies non-elastic material to the faces, thus, the faces become rigid  $\langle F_1 - E_1, F_2 - E_1 \rangle$ .
9. Rule  $x \rightarrow x_1$  draws visual attention, apply to lines  $\langle L0, L1, L3 \rangle$ .
10. Rule  $x \rightarrow x + E_1 + S_{x-y}$  applies one rotational degree of freedom and rigidity to the line, thus, it becomes a hinge between the two faces.

At this state, the previous tensegrity module, composed of line compositions, becomes an assemblage of faces embedded with rigid material connected with a hinge. Mapping this face’s assemblage to the weaving composition will return a folding structure (Sierra 2016; Jackson 2011). We continue on from this iteration to see what we can do with the folding:

7. Rule  $x \rightarrow x_1$ , returns visual attention back to the lines (L0, L1, L3).
8. Rule  $x \rightarrow b(x)$ , returns the point of the line and erases the three lines.
9. Rule  $x \rightarrow \Sigma x_2$ , highlights all the face.
10. Rule  $x \rightarrow x + E_0$ , applies elastic material to the face. Thus, two rigid lines are now connected by an elastic face in their endpoints.

Examples in Figs. 3 and 4 show only a few of many ways in which mechanical and visual embedding can work in tandem to generate different structural configurations. Different visual attention leads to different geometry and mechanical consequences, as there are many shapes to focus and modify in each step. The generated designs also show that the points in each step never left the initial position

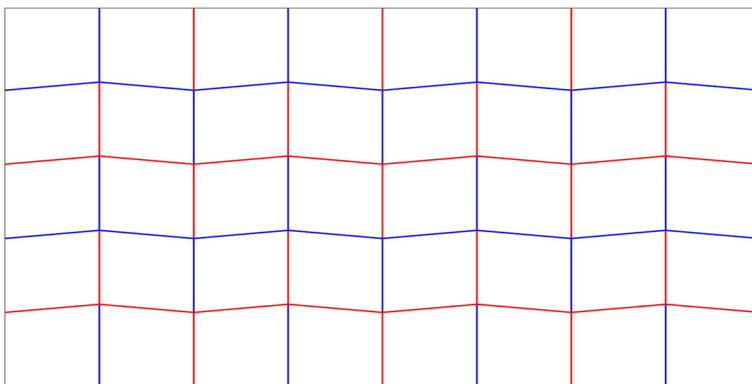


**Fig. 4** Comparison between parametric iteration and visual-mechanical embedding

even after series derivation. This allows the function to be applied locally to a certain part of a structural configuration while keeping its connectivity with the neighboring members without compromising the global configuration.

**Function Assembly**

There are several ways to compile this iteration. One way is to create a new function, G2, by nesting the previous function, G1, into this process to show how to turn a tensegrity module into a folding structure, such that  $G2: (G1) \Rightarrow x_1 \Rightarrow x + b^{-1}(x) \Rightarrow \Sigma x_2 \Rightarrow x + E_1 \Rightarrow x_1 \Rightarrow x + E_1 + S_{x-y}$  or we can deploy the G1 function and come up with a direct weaving-to-folding function such that  $G3: x_0 \Rightarrow x + b^{-1}(x) \Rightarrow x_1 \Rightarrow E_0 \Rightarrow x_1 \Rightarrow x + b^{-1}(x) \Rightarrow \Sigma x_2 \Rightarrow x + E_1 \Rightarrow x_1 \Rightarrow$



**Fig. 5** Miura fold crease pattern

$x + E_1 + S_{x-y}$ . With deployed function we can modify variables in existing functions. For instance, instead of applying rigid material in function G3, we can change an elastic into a rigid one by turning variable  $E_0$  into  $E_1$  and apply a boundary rule to omit the line in the last part, thus,  $x_1 \Rightarrow b(x)$  instead of  $x_1 \Rightarrow S_1$ . This function will return a compressive-tension system where the rigid lines are linked by a fabric instead of cables (Peña 2011). Moreover, the iteration can also be assembled into a more abbreviated function such as G4:  $W \rightarrow T'$ ; G5:  $T \rightarrow F'$ ; G6:  $W \rightarrow F'$ , where  $T'$ ,  $F'$  and  $W'$  are parametric schema embedded with the shape's mechanical description and conditional parameter ( $W$  = weaving,  $T$  = tensegrity and  $F$  = folding).

The open format (G1, G2, and G3) is more transparent to understand how the translation across structural configurations works and is more accessible as an open-source to generate other possible designs by tweaking the function, thus more suitable for pedagogical and inventive use. The compiled format (G4, G5, and G6) is more opaque and pragmatic, yet practical and ready to iterate other existing designs by transforming a certain precedent as initial shape, thus more in line with design practice.

### Inversed Function

In a similar manner, we can make an inverse function to generate new structure from a different direction. Consider function G2:  $T \rightarrow F'$  that converts tensegrity to folding, which includes iteration:  $x \Rightarrow x_1 \Rightarrow x + b^{-1}(x) \Rightarrow \Sigma x_2 \Rightarrow x + E_1 \Rightarrow x_1 \Rightarrow x + E_1 + S_{x-y}$ , or in words, 'in tensegrity, see lines and draw face from them. See the new faces and apply rigid material onto them. See the the lines between the faces and apply pin-joint into it to create hinge mechanism'. (Note that if a pin-joint is not applied, the configuration will behave as a folded-plate structure.)

We can inverse function G2 by reversing the iteration  $x_1 \Rightarrow x + b^{-1}(x) \Rightarrow \Sigma x_2$  into  $\Sigma x_2 \Rightarrow x + b(x) \Rightarrow x_1$ . Therefore, we have the inverse function G7:  $F \rightarrow T'$  to convert folding into tensegrity with iteration  $x \Rightarrow \Sigma x_2 \Rightarrow x + b(x) \Rightarrow x_1 \Rightarrow E_1 \vee E_0$ , or in words 'in folding, see faces, draw lines from them. Apply rigid material onto the mountain crease and flexible material onto the valley crease'. Or to be more precise:  $x \Rightarrow \Sigma x_2 \Rightarrow x + b(x) \Rightarrow x_1 \Rightarrow (x_{2\ i} \angle x_{2\ i+1} < 180^\circ): E = E_1 \vee (x_{2\ i} \angle x_{2\ i+1} > 180^\circ): E = E_0$ , where the boolean operator OR ( $\vee$ ) checks the angle between two faces: if it is larger than 180 degrees, it's a mountain crease, and therefore apply rigid material to the crease, otherwise, apply flexible material.

### Folding to Tensegrity

To demonstrate the inverse function, this section shows how to generate a novel compressive/tension configuration from a particular folding structure. For this case, we use the Miura fold pattern in Fig. 5. Founded by Koryo Miura in 1970, the Miura fold was originally purposed as a deployable mechanism to compactly package and deploy a large membrane into space (Miura 1985). In its further

development, the pattern has become one of the most popular folding styles in the field of design and engineering.

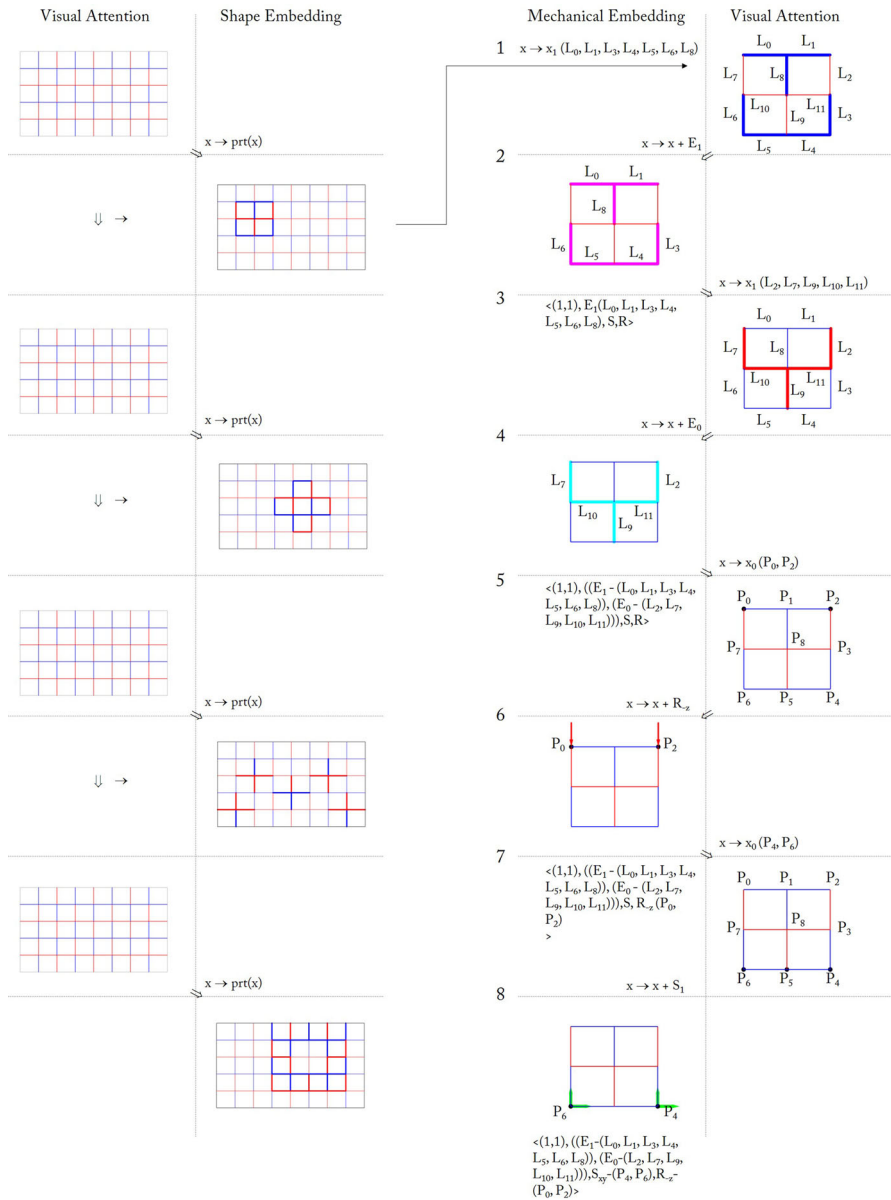
Miura crease patterns consist of a tessellated parallelogram; a quadrilateral with two parallel sides with no right angles. Miura tessellation performs parallel lines in one direction (horizontal) and zigzag lines in the other direction (vertical), where the lines are alternating between mountain and valley folds. With this configuration, Miura crease patterns can be flattened with rigid material, according to its material thickness (Lebée 2015).

In this case, we use a square pattern to show the ways in which a visual and mechanical embedding principle applied onto Miura's folding pattern can convert its structural property from deployable into a fixed yet flexible structure. Note that this goal is rather different to most of the other structure experiments with folding whereby the Miura fold is being fixed into a rigid folded-plate structure, such as the folded timber plate (Weinand 2009). By converting it into a flexible structure, we aim to generate a new structural configuration that can maintain the lightweight quality of the original Miura fold. For this reason, we apply a folding-to-tensegrity function ( $G7: F \rightarrow T'$ ) to convert the Miura fold into a tensegrity structure, where membrane or plate properties may occur during the computation. To anticipate over-elastic behavior by the inplane stiffness only (as in membrane) and over-rigid structure by outplane stiffness only (as in plate), our visual-mechanical embedding process reconfigures the interplay between tension and compression using flexible and rigid material to provide bending stiffness throughout the shell surface.

To begin with, we applied visual embedding onto the Miura fold module and identified the mountain/valley crease patterns on the surface. The example in the left of Fig. 6 shows four of the many possible modules that can be embedded from the pattern by shifting the area of visual attention. Note that the empty rule  $\rightarrow$  means: do nothing with the shape, whereas rule  $x \rightarrow \text{prt}(x)$  means: see shape  $x$  and focus only on a specific part of the shape  $x$ . While empty rule  $\rightarrow$  seems trivial, it is important to neutralize the visual attention state and clear previous description (memory) to allow other design opportunities.

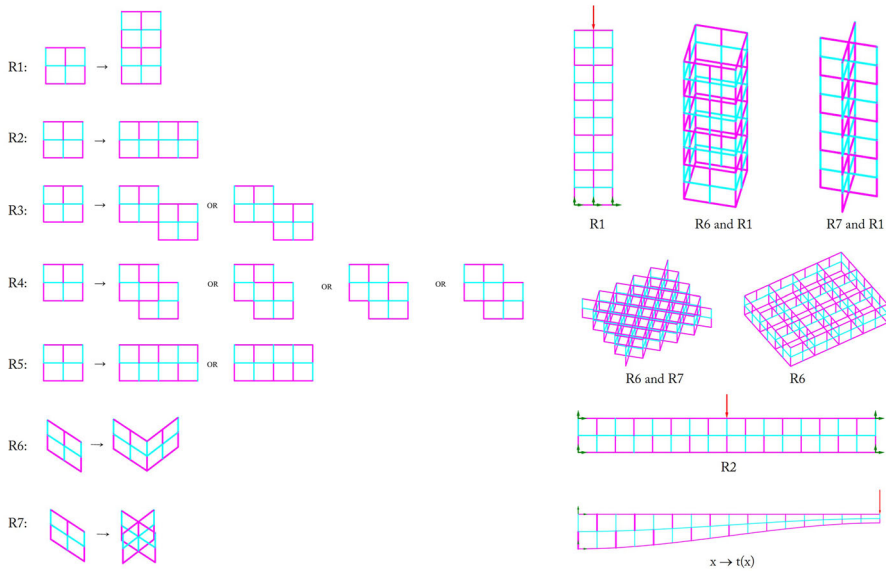
In the following, we show the ways material properties, gravity loads, and the boundary conditions are assigned to one of the modules with function  $G9: F \rightarrow T'$ :

1. Rule  $x \rightarrow x_1$  draws visual attention to mountain lines (blue)  $\langle L_0, L_1, L_3, L_4, L_5, L_6, L_8 \rangle$ .
2. Rule  $x \rightarrow x + E_1$  applies rigid material to the lines (magenta)  $\langle E_1 - (L_0, L_1, L_3, L_4, L_5, L_6, L_8) \rangle$ .
3. Rule  $x \rightarrow x_1$  draws visual attention to valley lines (red)  $\langle L_2, L_7, L_9, L_{10}, L_{11} \rangle$ .
4. Rule  $x \rightarrow x + E_0$  applies flexible material to the lines (magenta)  $\langle E_0 - (L_2, L_7, L_9, L_{10}, L_{11}) \rangle$ .
5. Rule  $x \rightarrow x_0$  draws visual attention to the points on the top  $\langle P_0, P_2 \rangle$ .
6. Rule  $x \rightarrow x + R_{-z}$  applies gravity load to the points  $(R_{-z} (P_0, P_2))$ .
7. Rule  $x \rightarrow x_0$  draws visual attention to other points in the bottom  $\langle P_4, P_6 \rangle$ .
8. Rule  $x \rightarrow x + S_1$  constrains the boundary condition onto the points as fixed supports  $(S_1 - (P_4, P_6))$ .



**Fig. 6** Visual and mechanical embedding on the Miura fold

The resulting mechanical embedding shows a new structural configuration derived from the Miura fold with tensegrity behavior: some flexible cables are now in tension to support the rigid beam, while some of the the rigid beams are in compression to hold the tension.



**Fig. 7** Additive rules (*left*) and example results (*right*)

### Additive Iteration

With this new module, we then apply the additive rule with linear transformation (e.g., translation, rotation, and reflection) to multiply the module in two- and three-dimensional space to create a larger surface. Figure 7 shows translation and reflection rules in two- and three-dimensional space to create a composite of tension and compressive elements, with some examples of resulting space-truss typologies in the horizontal (e.g., roof) and vertical directions (e.g., column). These results can be modulated by using the same rules within a certain grammar or through parametric modulation with a schema:  $x \rightarrow t(x)$ . For instance, at the local configuration, each of the structural elements—cables, beams, floors, walls—can be further optimized by either incrementally changing the material properties (e.g., the cable's elasticity) or the component's dimension to equate external force (e.g., the floor's thickness or beam's height). At the surface's global configuration, optimization to adapt for different architectural functions can be achieved by modulating the ratio of scale, height, width, and depth, or by increasing/decreasing the number of additions in multiplying the surface or subdivision to increase the surface's density.

### Recursive Embedding

Embedding works in a non-linear manner. If the mechanical meanings in Fig. 7 are to be converted back into the folding context, (e.g., rigid lines become mountain lines and flexible lines become valley lines) then the result could be a different type of flattenable and unflattenable creases composition. In any iteration stage,

backward or forward, visual and mechanical embedding can be applied recursively to generate more structural designs. Figure 8a shows visual embedding iteration applied to the resulting shape from Fig. 6. Applying  $x \rightarrow x + b(x)$  returns the lines' endpoints, and then applying inverse boundary  $x \rightarrow b^{-1}(x) + E_1$  to the points returns rigid diagonal lines (pink). Inverse boundary rules are then repeated to these new lines and the lines at the perimeter, and a new surface appears: a rigid arrowhead. In the 3D module, the same inverse boundary produces a square shape in a different plane (Fig. 8b). Applying rigid properties  $x \rightarrow b^{-1}(x) + E_1$  to these shapes produced a rigid surface (e.g., a floor). With new surfaces, the volumetric structure becomes more stable with less deflection. The same is applied to the second iteration (lower row), only this time, it is applied to the flexible lines. Here, the inverse boundary of flexible lines are then applied with function:  $x \rightarrow b^{-1}(x) + E_0$  to produce a flexible surface (fabrics/cyan), which holds the rigid lines in a comparable configuration to the previous cables. The generated surface, whether rigid or flexible, allows the new structural configuration to be more adaptive in responding to certain architectural requirements such as privacy and climate (Fig. 8c).

In the following example, we show a different mechanical relationship from the same module. Visual attention is focused on the middle horizontal line and function  $x \rightarrow x + E_0$  changes a rigid line into a flexible line (Fig. 9a). As a result, the rigid lines are now separated into an array of single rigid lines connected by cables. The same inverse-boundary rule is then applied in the third step and new diagonal lines appear. Although the module looks similar to the one in Fig. 8, the mechanical

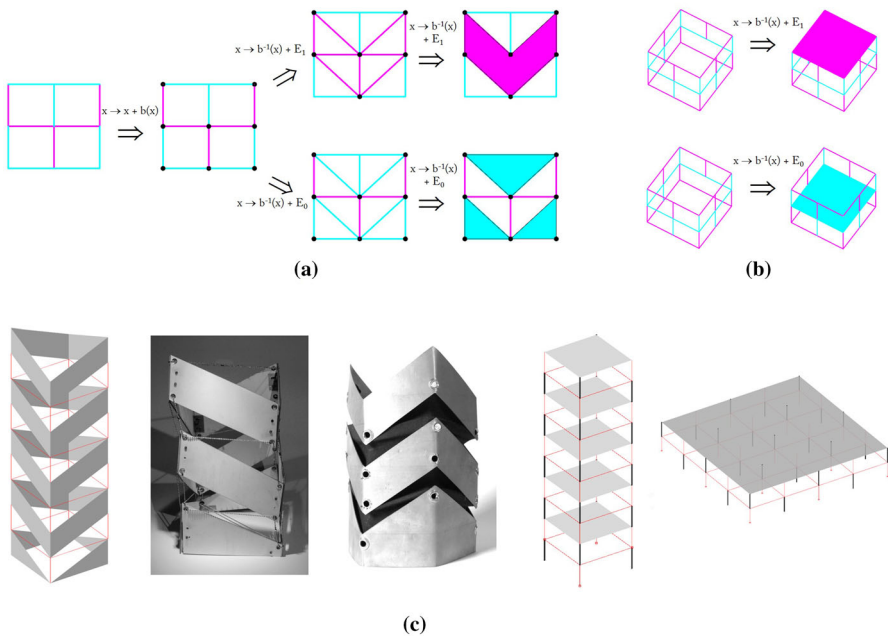
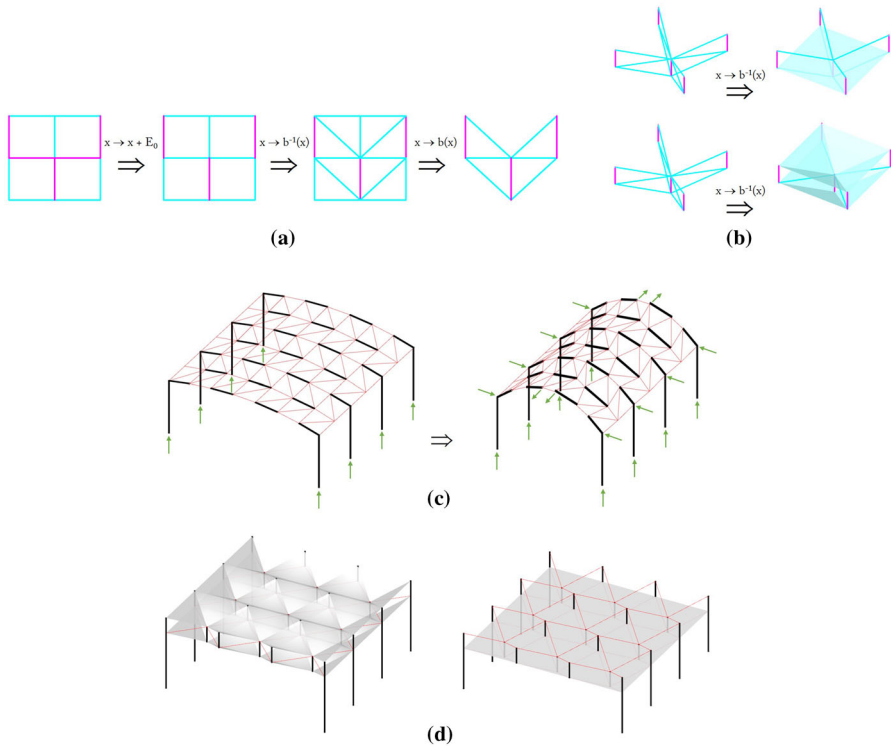


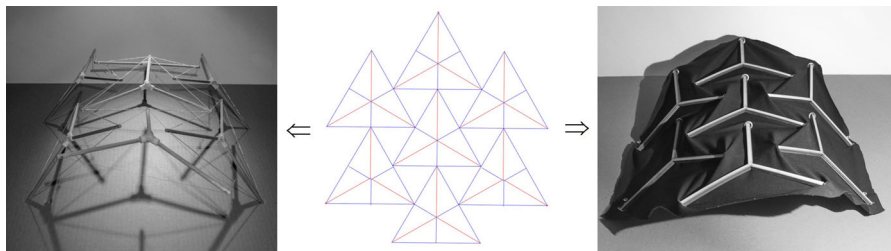
Fig. 8 New generated typology in vertical and horizontal orientation





**Fig. 9** The two-dimensional array of modules shows different ways of assembling surface elements

performance is different. The roof structure in Fig. 9 shows how the new configuration would affect the structural configuration as a whole. The new two-dimensional module is multiplied in the x-y plane (see Fig. 9c). If vertical loading is applied on the surface, this configuration will collapse due to the rotational movement along the horizontal lines. However, this deflection can be minimized by applying lateral loads from the side as a compressive reaction onto the vertical load. As a result, it increases the shell’s bending stiffness (see similar strategy in Fig. 10). Thus, in this case, the structure is optimized not by increasing material properties or



**Fig. 10** Ron Resch Folding Tessellation converted into flexible shell with struts-cable (left) and struts-fabric (right) with function G7:  $F \rightarrow T'$

dimensions, but rather by changing the loading direction, which is commonly practiced in traditional craftsmanship. The fact that this design is collapsible in one direction means it could serve as a deployable shelter. Note that if  $x \rightarrow b^{-1}(x)$  applied to all flexible lines, this configuration will generate a stiff tent structure composed with struts and fabric (Kojima 2013).

In a subsequent rule iteration in Fig. 9a, instead of repeating the inverse boundary to create a surface, in the fourth step, the boundary rule  $x \rightarrow b(x)$  is applied to the upper and lower horizontal lines to get only the endpoints as the boundary of the lines. In other words, it leaves empty lines with only the endpoints remaining intact. The result is yet another new module with a zig-zag configuration where the boundary condition is set on rigid lines on the sides to pull out the flexible cables. The rules in Fig. 9b show a three-dimensional transformation of this module. The inverse-boundary rule  $x \rightarrow b^{-1}(x) + E_0$  adds a two-dimensional flexible surface to this module: a single membrane (top) and a double-membrane module (bottom). This creates a composition where the cables and membranes tensionally brace, hold, and lock all the rigid lines together within the space-structure. With fewer rigid components, the structure becomes lighter (Fig. 9d).

## Conclusion

In this paper, we have formulated a computational design method to enhance the ideation process in shell structures with visual and mechanical embedding. The output quality, in regards to how unique the new designs are compared to the precedents, shows unexpected yet applicable new designs, not only in their physical differences but also in terms of compressive-tension force variances. Inspired by the bottom-up approach in traditional crafts, the additive computation was able to generate various compositions and orientations for shell typology, e.g., tower, plates, and vault, along with other possible features e.g., flexibility and deployability.

With such explicit representation, the intuitive process of structural ideation becomes more transparent. This representation allowed us to interpret the designer's ideation process in developing their structural systems. In converting weaving into tensegrity, for example, our iteration could interpret the creative moment of how Kenneth Snelson perceived the woven shape in inventing tensegrity structure. Furthermore, the iteration from tensegrity to folding shows promising links to allow the structural principles to travel across various configurations. Hence, it paves the way for an engineer/designer to learn across disciplinary boundaries to expand the shell structure's design language.

While the examples show a promising path for incorporating visual calculation and mechanical registration in structural design, this method needs more elaboration to develop a rigorous, yet intuitive model that could produce more valuable outputs. To ensure the outputs' constructability, the visual embedding and mechanical registration process needs to be controlled with conditional rules, for instance, by constraining the mechanical values such as the maximum displacement and bending stress, or design requirements such as shape proportion, dimension, and material

opacity. Hence, for future investigations, the parametric modulation needs to be seamlessly integrated with a subsequent optimization process, such as Finite Element analysis and evolutionary algorithms.

To ensure the method's versatility, further study of different types of crafts is essential to show how effective this method is in generating new structural designs from various precedents. There are many possibilities beyond the given examples, because flipping the mountain-valley crease into flexible or rigid material, for instance, is just one way to embed mechanical meaning from the Miura fold. The Miura fold itself is just one of many other folding styles, and folding is just one type of craft. Moreover, while weaving-to-tensegrity examples suggest how a designer learns from others (e.g., how Snelson was inspired by a basket-weaver's design), the method can also investigate how a designer learns from nature to derive a new structural design (e.g., how Miura may learned from Hornbeam leaves).

## References

- Akbarzadeh, M., T. Van Mele and P. Block. 2016. Three-Dimensional Graphic Statics: Initial Explorations with Polyhedral Form and Force Diagrams. *International Journal of Space Structures* 31 (2–4): 217–226.
- Bathe, Klaus-Jürgen. 1996. *Finite Element Procedures*. Englewood Cliffs, N.J: Prentice Hall.
- Garlock, M.E.M. and Billington, D.P., 2008. *Félix Candela: engineer, builder, structural artist*. Princeton University Art Museum.
- Hensel, Michael, Achim Menges, and Michael Weinstock. 2004. *Emergence: Morphogenetic Design Strategies*. Academy Press.
- Isler, Heinz. 1986. *Heinz Isler, Schalen: Katalog zur Ausstellung*. Stuttgart: K. Kramer.
- Jackson, Paul. 2011. *Folding Techniques for Designers: From Sheet to Form*. Laurence King Publishing.
- Knight, T. and G. Stiny. 2015. Making Grammars: From Computing with Shapes to Computing with Things. *Design Studies* 41: 8–28.
- Kojima, Kazuhiro. 2013. C + A Coelacanth and Associates: MOOM Tensegritic Membrane Structure. *Designboom Architecture & Design Magazine*. May 15. <http://www.designboom.com>.
- Lebée, A., 2015. From Folds to Structures, a Review. *International Journal of Space Structures*, 30(2), pp.55–74.
- Lee, J., Fivet, C. and Mueller, C., 2015. Modelling with forces: grammar-based graphic statics for diverse architectural structures. In *Modelling Behaviour* (pp. 491–504). Springer International Publishing.
- Mitchell, W.J. 1998. Articulate Design of Free-form Structures. In: *Artificial Intelligence in Structural Engineering*, ed. Ian Smith, 223–234. Berlin: Springer-Verlag.
- Miura, Koryo. 1985. Method of Packaging and Deployment of Large Membranes in Space. *The Institute of Space and Astronautical Science Report* 618: 1.
- Mueller, C.T. 2014. Computational Exploration of the Structural Design Space. Ph.D. thesis, Massachusetts Institute of Technology.
- Muslimin, R., 2014. EthnoComputation: On Weaving Grammars for Architectural Design. Ph.D. thesis, Massachusetts Institute of Technology.
- Otto, Frei. 1973. *Tensile Structures*. Cambridge, Massachusetts: The MIT Press.
- Pearce, Peter. 1978. *Structure in Nature is a Strategy for Design*. Cambridge, Massachusetts: MIT Press.
- Peña, Diana Maritza. 2011. Application of Tensegrity to Tensile-Textile Constructions: Formfinding and Structural Analysis. *Journal of the International Association for Shell and Spatial Structures* 52 (2): 67–81.
- Rian, I. and M. Sassone. 2014. Fractal-Based Generative Design of Structural Trusses Using Iterated Function System. *International Journal of Space Structures* 29 (4): 181–204.
- Shea, K. and J. Cagan. 1997. Innovative Dome Design: Applying Geodesic Patterns with Shape Annealing. *Artificial Intelligence for Engineering, Design, Analysis and Manufacturing* 11 (05): 379–394.

- Sierra, Fernando. 2016. Tensegrity or Folding? *Flickr* <https://www.flickr.com/photos/elelvis/316797710/>. Accessed July 15.
- Snelson, Kenneth. 2003. *Kenneth Snelson: Sculpture*. New York: Marlborough.
- Stiny, G. 1981. A Note on the Description of Designs. *Environment and Planning B: Planning and Design* 8 (3): 257–267.
- Stiny, G. 2011. What Rule(s) Should I Use? *Nexus Network Journal* 13 (1): 15–47.
- Stiny, G. 2008. *Shape: Talking About Seeing and Doing*. Cambridge, Massachusetts: The MIT Press.
- Stiny, G. and Gips, J., 1971, August. Shape Grammars and the Generative Specification of Painting and Sculpture. In *IFIP Congress (2)* (Vol. 2, No. 3).
- Thompson, D'Arcy Wentworth. 1992. *On Growth and Form: The Complete Revised Edition*. New York: Dover Publications.
- Weinand, Yves. 2009. Innovative Timber Constructions. *Journal of the International Association for Shell and Spatial Structures (J. IASS)* 50 (2): 111–120.

**Rizal Muslimin** is a lecturer at the Faculty of Architecture, Design, and Planning, the University of Sydney. He holds a Ph.D. in Architecture: Design and Computation from Massachusetts Institute of Technology in 2014. He received his Master and Bachelor degrees from Institut Teknologi Bandung and Universitas Parahyangan, both in Indonesia. His research interest centered at the intersection of craft, architecture and computational making, including ethnocomputational studies on investigating the logic underlying the vernacular and contemporary design.