A Poe(Gene)tic Algorithm Method to Compute Gradient Spatiality

Tugçe Darcan
Institute of Science & Technology, Istanbul Technical University, Turkey
tugcedarcan@gmail.com

Ethem Gürer
Institute of Science & Technology, Istanbul Technical University, Turkey
ethemgurer@gmail.com

ABSTRACT

Related to the very common use of contemporary evolutionary methodologies, metaphorical relations coming out between architectural design and different structures (open and closed) and also new forms of spatiality are now being discussed. We are trying in this research, to query such a relationship between design and poetic language. In this regard, this paper concerns how haiku, well-known Japanese poem, may turn out to be an unfolding layer within the act of designing, by standing as a sort of syntactic generator. Genetic algorithms are benefited to compute the existing formalism in haiku structure, which gives rise to 'gradient spatiality'.

KEYWORDS: evolutionary design; genetic algorithms; poetic language; haiku; gradient spatiality.

Introduction

Language, as generally assumed, is one of the main forms of human representation. It may be classified as common language, body language, machine language, poetic language etc. Poetic language, due to its elaborately chosen vocabulary and wise associations formed between its elements (words), differs from other linguistic representations. It offers a broad meaning nest within a formalized grammatical structure. In other words, poetry possesses a hierarchical formalization of syntax which can be handled through a computational algorithm.

When it is time to discuss about the reflections of poetic language on architecture, *haiku* is the most frequently mentioned type of the poetic domain. Introduced to world literature by Japanese poet Matsuo Bashō in the 17th century, traditional Japanese haiku is considered by Antoniades (1992) as a highly powerful sample of semantic density that can attract architects who trace the trails of new spatial experiences. It is important to note that the multiplicity in the semantics of a haiku stems from not only the open-endedness of its discourse but also the formal hierarchy of words and lack of punctuation marks. In this regard, haiku can be said to have an organized formalism. Known as one of the shortest poem types in the world literature, it seems to preserve potential to act like a plain vertebra for computational design in the metaphorical sense. Especially the traditional form of haiku consisting of three lines and generally seventeen syllables (five in the first line, seven in the second, and five in the third), demonstrates a defined structure to be examined, evaluated and computed.

In the theoretical sense, the technical method proposed to decompose the syntactic structure of haiku poem and to interpret it within design thinking rests on basic rules of genetic algorithms, which remain the most recognized form of evolutionary algorithms (EAs) as Whitley (2001) pointed out. Starting from the pioneering studies of John Holland in the 1960s and 1970s, numerous works have made it possible to generate solutions to design problems by using techniques inspired by nature. In accordance with this generative character of GAs, general properties of haiku poems are examined and new haiku poems are put forth based on evolutionary principles. Next, physical and digital models of the new haiku offspring are generated as frames and finally, they are integrated giving rise to 'gradient spatiality'.

Method

In our study, the motivation comes from the plain syntactic configuration of haiku. As a seventeen-syllable poem organized into three lines, traditional Japanese haiku presents a predefined set of syntactic structures. We prefer to investigate haiku poems which are either translations or originally written pieces in English. The primary reason is that English is an inflexional language. Inflexion allows exchanging words between two haiku poems with little or no need for grammar correction. Another reason to study on English haikus is the fact that their syntactic structure has more variations when compared to traditional ones in Japanese.

Use of genetic algorithms in generative design approach makes it possible to handle the haiku syntax from an evolutionary perspective. A randomly selected pair of haiku poems is considered as "parents" of future "child" haikus. Initially, syntactic structures of poems are coded individually. Secondly, they are evaluated relatively to each other according to syntactic arrangement in their lines. Later, they are "hybridized" in search of new haiku poems with words "inherited" from both of the parent poems. At the beginning of the coding process, adjectives, nouns and verbs of the first and the second poem are marked by the initial letter of their types, as A, N or V. These initials are treated as "genes" to be replaced by a word of the same type in the other parent. Prepositions and articles are not included in genes so that grammatical structure can be transferred to the "next generation" correctly when word exchange occurs. As the next step, initials are sequentially numerated. Finally, dominance value is attributed to each initial, depending on the length of the lines in the poem. The dominance of an initial is calculated by the division of the maximum number of initials in one line to the total number of initials in its own line.

After coding parents, they are evaluated as a pair in terms of similarity between their syntactic structures. When there is no difference in the number of initials in all of the three lines, the parents are defined as "symmetric". If there is disparity in the number of initials in at least one line, the parents are defined as "asymmetric" (see fig. 1).

Design Interpretation: Evaluation and Hybridization of Haikus

To start with, two poems are randomly selected from the literature of English haiku: haiku 1 by Nikola Nilic and haiku 2 by Banea Stefan. The poems are then encoded according to the method based on certain word types (see section 2). After dominance values are attributed to encoded words, gene schemas of poems are seen to have different organizations, which cause them to be asymmetric parents (see fig. 2).

When it is time to hybridize these asymmetric parents, a limited range of crossover is permitted in which only one gene is exchanged between the code strings of parents. As another constraint, crossover is allowed to occur only between the child haikus just produced in the former step. In other words, slightly altered codes of child haikus become the parent codes of the next generation. The crossover schema in the figure 2 demonstrates the six possible paths used for gene (word) exchange.

According to the constraints mentioned above, a table (see fig. 3) is prepared in which both the genotypes (poems) and code strings (phenotype codes) are evolved along with the gradual progress of hybridization. At the each step of the crossover process, the resulting forms of code strings are numerated according to their precedence in the evolutionary process and the initial parent poem from which their syntax is inherited. Every single code string also represents dominance values which can be embodied through geometrical and topological interpretation. Initially, the code strings, namely, the phenotype codes are physically modeled via the use of a specific method.

Phenotype Representation of Haiku Offspring

A visualization method is identified in order to not only simulate the syntactic rhythm of haikus three-dimensionally but also make use of child haikus as generators of new phenotypes within the same design language. Therefore, phenotype codes of every child haiku is decided to be physically modeled as enclosed frames. First, the code strings of child haikus are physically modeled as strings of rods whose length are defined by the dominance value of each gene (word). When there is no change in the word type, the next rod (word) is attached to the end of the previous one in the way that both of them can be rotated on parallel planes. Provided that word type alters, the following rod is attached to the former one in the way that they can be rotated on planes perpendicular to the base plane of the former ones.

Secondly, rod strings of child haikus are closed, forming frames. At this level, studying on physical models provides the opportunity to acknowledge that relative rotational directions of rods introduce multiple possibilities for enclosure of frames. Once enclosed, configuration of a frame can still be changed by adjustment of moveable joints. However, geometric domain of motion is restricted by length of rods and their rotation planes. When the frame tends to organize itself in a certain formal configuration within the limited movement domain, the joints can be fixed. Thus, among all the possible alternatives, structurally most stabile frame configurations are chosen.

After individual modeling of phenotype codes, formal evolution of phenotypes is aimed to represent. Based on the numerical values in the names of phenotype codes, frame models are sequentially aligned in two groups. Next, groups of frames are placed one after another (see fig. 4). All of them fixed on a plane along an approximately linear axis, frames are seen to bring about a formal rhythm which is visible both in the elevation and perspective views.

Thus, in the method developed for three-dimensional representation of haikus, two types of conversion take place. As already mentioned, dominance value of nouns, adjectives and verbs is used as the parameter determining the length of rods. Hence, numerical distribution of words into lines, in other words, the formal structure of a haiku is converted to dimensional rhythm for the rods of a frame. Secondly, word type alteration in a genotype is used as the parameter to determine planes of rotation for two rods attached to each other. In this way, the syntactic structure of a haiku is converted to mobility for joints of rods.

Digital Representation of Phenotypes

As for the digital representation phase, together with the Grasshopper plug-in, Rhinoceros 3D modeling software is preferred to use for two main reasons. First, Rhinoceros 3D facilitates ease of modeling non-uniform rational B-spline (NURBS) surfaces, making it possible to connect the frames by forming quadrilateral surfaces instead of triangulation. Secondly, Grasshopper interface provides the chance to represent and manipulate dynamic forms in ease.

The process of digitalization begins with the definition of rods of frames as connected lines which can be rotated relatively to one another. After all the six rods are represented as six lines in equal lengths to their physical versions, the open endpoint of the first line and the last line are required to intersect so that a frame can be formed. However, multiple axes for relative rotations and three different values of line length make it almost impossible to close frames by trying to manually fix the angular positions of lines. Therefore, the 'Galapagos' component, which is the evolutionary solver of Grasshopper, is utilized to be able to adjust the positions of every group of six lines. In order to function the 'Galapagos' component, all of the sliders determining the relative rotational motion of frames are identified as 'Genome'. The distance between the open endpoints of the first line and the last line is measured by the 'Distance' component. Connecting the distance value to the 'Fitness' part, the 'Galapagos' component becomes ready to function. As the distance value has to be equal to zero for frame formation, minimization of the distance is defined as the target of the evolutionary process. Genetic algorithm is run by the 'Galapagos' component to minimize the shortest distance between the two endpoints up to the threshold value of zero. After a certain number of generations are produced, endpoints finally intersect in precision at the possible highest level. If not 0.000000, groups of lines are closed as frames at a distance of around 0.001112 mm between the two endpoints (see fig. 5). In this way, lines update the coordinates of their endpoints in the way that they can be stabilized, forming a rigid frame. At this point, it is important to underline that groups of lines have different tendencies to organize their positions when the evolutionary solver is run at varying initial configurations. This demonstrates the presence of multiple possibilities for both enclosure of frames and the final configurations of the frame groups. When all of the frames are stabilized in desired forms by the help of the evolutionary solver, they may also be claimed to reach one of the true poses in terms of their self-standing capability. To benefit from the opportunities of digital modeling interface, the rigid frames are organized in a dynamic way along x, y and z axes, forming a vertebra of curvilinear tube. As for the final step, the edge points of frames are connected by lines to nearest edge points of both the previous and the next frame. This method results in the formation of quadrilateral surfaces at a minimum level of curvilinearity since there is no need for triangulation as all the frames possess the same number of edge points. Organized along different curvilinear routes (see fig. 5), final forms present non-uniform volumes which have gradient spatiality.

Remarks on Design Interpretation

Compared to standard flow of genetic algorithm, the applied crossover technique may be regarded as a small-scale evolutionary process. However, such a decelerated evolution provides us with two crucial opportunities. First, reduced rate of differentiation per iteration gives the chance to have a better understanding of the evolution occurring in both phenotypes (frames) and genotypes (poems). Second, narrowness of the gene pool results in a three-dimensional formal gradient along the sequential array of phenotypes. In addition, it also gives rise to a gradient pattern in semantics of the haiku poems, namely, genotypes.

Overall, the method can be regarded as a demonstration of possible degrees of genetic alteration at a time, ranging between the minimum and the maximum thresholds of evolution. The initial forms of selected poems and their three-dimensional representations as frames exhibit the minimum amount of evolution, that is, the case when no hybridization occurs. As for the final genotypic and phenotypic results after evolution, all of the genes in the code strings of two parent poems are seen to interchange, which demonstrates the theoretically maximum amount of evolution that may take place between only one pair of parents.

When the crossover is terminated after six iterations, numeric values in the code strings of both poems are totally exchanged (see fig. 3). Due to the dissimilarity of the two syntactic structures and choices for gene exchange paths, the order of dominance values in the final code strings differ from the very first sequence of the other parent poem. However, when a certain configuration of frame is presented, it sometimes becomes impossible to ensure on which level of the crossover it was produced. For instance, it is not possible to distinguish frame 1.6 from frame 2.1 by just comparing their phenotypic appearances since both of them have exactly the same sequence of dominance values.

At this point, genotypes, semantically evolved forms of the two haiku poems make it possible to ensure the level of hybridization of which a frame came out. As the encoded vocabularies of the two poems are completely distinct, none of the members of the new haiku offspring is identical to one another. Yielding different meanings, each haiku exhibits a unique genotype for each phenotypic representation. In addition, by means of comparison between the vocabularies of an evolved haiku and its initial form, the number of exchanged words can easily be detected. Since only one word is exchanged at each level of hybridization, the number of new words in a haiku is equal to the number of genetic iterations that came along so far.

To sum up, haikus standing for genotypes represent not only the content but also the quantity of evolution. Here, it seems possible to indicate that haikus as genotypes have superiority over binary genotypes that are most commonly used in computational applications of genetic algorithm. As a numerical representation consisting of 0s and 1s, binary coding cannot go beyond the degree up to which the code strings of dominance values contribute to the clarification of the 'blind' process of evolution. Haikus, however, point out where a phenotype should be positioned as a crystallization point within the uncontrolled duration of evolution. Hence, such an approach not only seeks for a way to deepen the understanding of evolutionary design process but also offers the chance to gain control over the genotypic by-products of the duration.

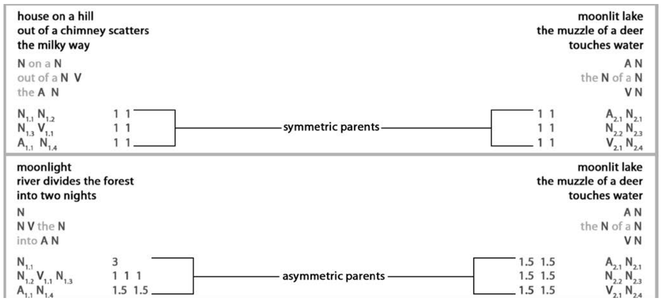


Fig. 1. Examples of symmetric and asymmetric parents (haiku by Dan Doman on the top left, by Nikola Nilic on the left down and by Banea Stefan on the right)

Concluding remarks

In this study, handled as an organized structure, poetry is taken as one of the poses for uncovering formalism in the language and then in design. Poetry, in this regard, is claimed to play an unveiling role within the act of designing, since it has been re-formed in a computational paradigm by evolutionary techniques, via the use of genetic algorithm. In the last 20 years, as Koza (1992) highlighted, genetic programming has become an important new sub-area of evolutionary algorithms in design. Gürer and Çağdaş (2009) claim that evolutionary methods are preferred by an ascending number of users as they systematize the design process through a holistic evaluation and create a huge solution domain with high quality. There is a large body of literature concerning design studies focused on both concepts and rules of genetic algorithms (GAs). As a similar study, haiku poems have been transmogrified by an algorithmic layout according to GA rules. Alterations occurring in each level of the process have been utilized to generate a gradient spatiality, in the search of poetic taste in digital design and intrinsic potentials in evolutionary computing.

References

Antoniades, A. C. 1992. *Poetics of Architecture: Theory of Design*. New York: Van Nostrand Reinhold.

Gürer, E., Çağdaş, G. 2009. An emergent form generation method for supporting conceptual design. Çağdaş, G., Çolakoğlu, B. (Eds.), *Proceedings of the eCAADe 2009* (pp. 167-172). Turkey.

Koza, J. 1992. *Genetic programming: a paradigm for genetically breeding computer population of computer programs to solve problems.* Cambridge, MA: MIT Press.

Whitley, D. 2001. An overview of evolutionary algorithms: practical issues and common pitfalls. *Information and Software Technology,* 43, 817-831.

Anakiev, D., Kacian J. 2001. *Knots: The Anthology of Southeastern European Haiku Poetry.* Retrieved in april 2012, from http://www.iyume.com/haiku/haikuex.html

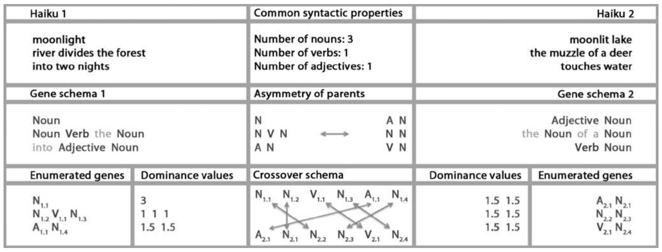
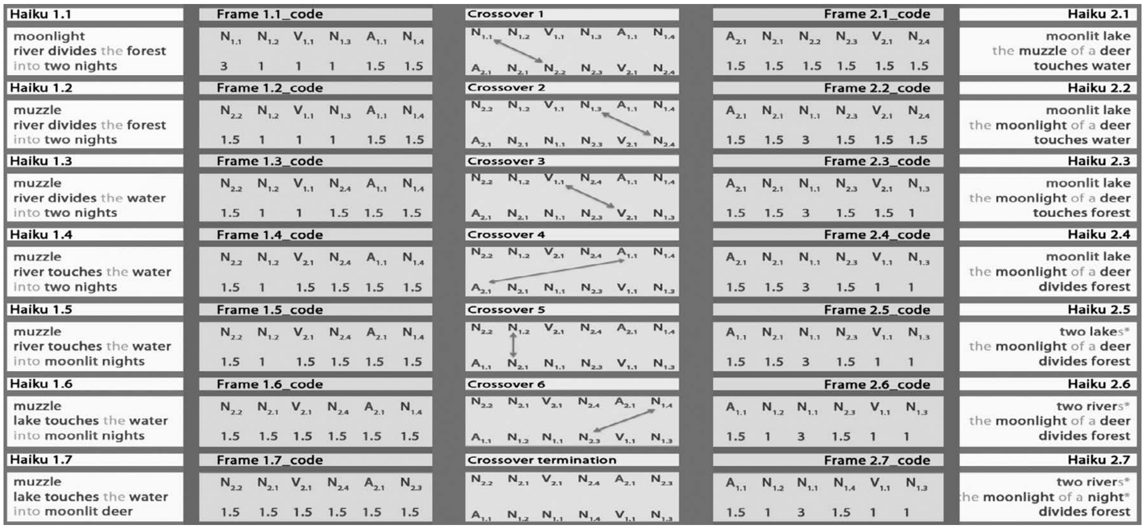
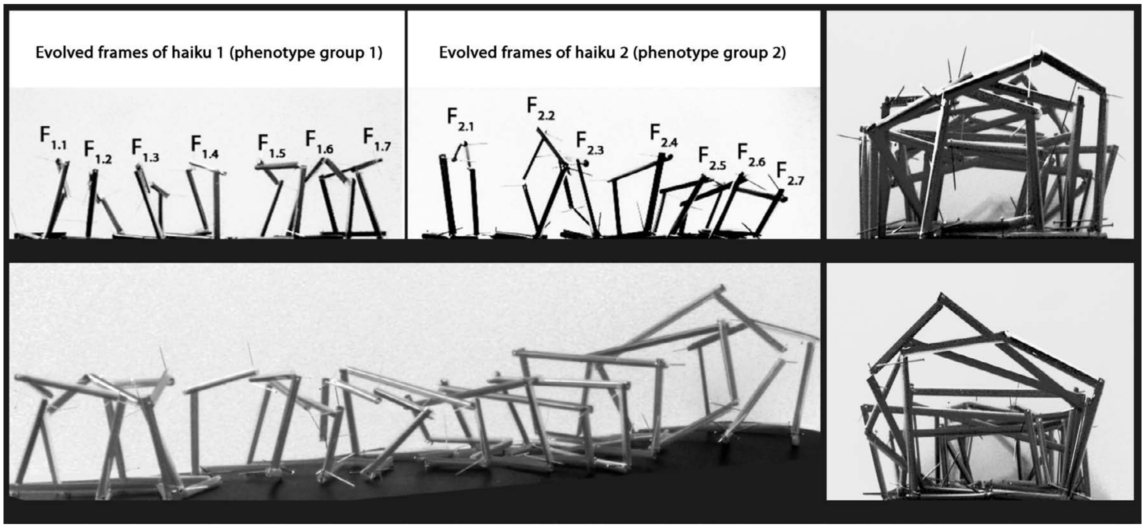


Fig. 2. Enumeration and evaluation of a randomly selected pair of haikus (haikus by Nikola Nilic on the left and by Banea Stefan on the right)



Haiku 1.1	**Frame 1.1_code**	**Crossover 1**	**Frame 2.1_code**	**Haiku 2.1**
moonlight river divides the forest into two nights	$N_{1.1}$ $N_{1.2}$ $V_{1.1}$ $N_{1.3}$ $A_{1.1}$ $N_{1.4}$ 3 1 1 1 1.5 1.5	$N_{1.1}$ $N_{1.2}$ $V_{1.1}$ $N_{1.3}$ $A_{1.1}$ $N_{1.4}$ $A_{2.1}$ $N_{2.1}$ $N_{2.2}$ $N_{2.3}$ $V_{2.1}$ $N_{2.4}$	$A_{2.1}$ $N_{2.1}$ $N_{2.2}$ $N_{2.3}$ $V_{2.1}$ $N_{2.4}$ 1.5 1.5 1.5 1.5 1.5 1.5	moonlit lake the muzzle of a deer touches water
Haiku 1.2	**Frame 1.2_code**	**Crossover 2**	**Frame 2.2_code**	**Haiku 2.2**
muzzle river divides the forest into two nights	$N_{2.2}$ $N_{1.2}$ $V_{1.1}$ $N_{1.3}$ $A_{1.1}$ $N_{1.4}$ 1.5 1 1 1 1.5 1.5	$N_{2.2}$ $N_{1.2}$ $V_{1.1}$ $N_{1.3}$ $A_{1.1}$ $N_{1.4}$ $A_{2.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{2.1}$ $N_{2.4}$	$A_{2.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{2.1}$ $N_{2.4}$ 1.5 1.5 3 1.5 1.5 1.5	moonlit lake the moonlight of a deer touches water
Haiku 1.3	**Frame 1.3_code**	**Crossover 3**	**Frame 2.3_code**	**Haiku 2.3**
muzzle river divides the water into two nights	$N_{2.2}$ $N_{1.2}$ $V_{1.1}$ $N_{2.4}$ $A_{1.1}$ $N_{1.4}$ 1.5 1 1 1.5 1.5 1.5	$N_{2.2}$ $N_{1.2}$ $V_{1.1}$ $N_{2.4}$ $A_{1.1}$ $N_{1.4}$ $A_{2.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{2.1}$ $N_{1.3}$	$A_{2.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{2.1}$ $N_{1.3}$ 1.5 1.5 3 1.5 1.5 1	moonlit lake the moonlight of a deer touches forest
Haiku 1.4	**Frame 1.4_code**	**Crossover 4**	**Frame 2.4_code**	**Haiku 2.4**
muzzle river touches the water into two nights	$N_{2.2}$ $N_{1.2}$ $V_{2.1}$ $N_{2.4}$ $A_{1.1}$ $N_{1.4}$ 1.5 1 1.5 1.5 1.5 1.5	$N_{2.2}$ $N_{1.2}$ $V_{2.1}$ $N_{2.4}$ $A_{1.1}$ $N_{1.4}$ $A_{2.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{1.1}$ $N_{1.3}$	$A_{2.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{1.1}$ $N_{1.3}$ 1.5 1.5 3 1.5 1 1	moonlit lake the moonlight of a deer divides forest
Haiku 1.5	**Frame 1.5_code**	**Crossover 5**	**Frame 2.5_code**	**Haiku 2.5**
muzzle river touches the water into moonlit nights	$N_{2.2}$ $N_{1.2}$ $V_{2.1}$ $N_{2.4}$ $A_{2.1}$ $N_{1.4}$ 1.5 1 1.5 1.5 1.5 1.5	$N_{2.2}$ $N_{1.2}$ $V_{2.1}$ $N_{2.4}$ $A_{2.1}$ $N_{1.4}$ $A_{1.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{1.1}$ $N_{1.3}$	$A_{1.1}$ $N_{2.1}$ $N_{1.1}$ $N_{2.3}$ $V_{1.1}$ $N_{1.3}$ 1.5 1.5 3 1.5 1 1	two lakes* the moonlight of a deer divides forest
Haiku 1.6	**Frame 1.6_code**	**Crossover 6**	**Frame 2.6_code**	**Haiku 2.6**
muzzle lake touches the water into moonlit nights	$N_{2.2}$ $N_{2.1}$ $V_{2.1}$ $N_{2.4}$ $A_{2.1}$ $N_{1.4}$ 1.5 1.5 1.5 1.5 1.5 1.5	$N_{2.2}$ $N_{2.1}$ $V_{2.1}$ $N_{2.4}$ $A_{2.1}$ $N_{1.4}$ $A_{1.1}$ $N_{1.2}$ $N_{1.1}$ $N_{2.3}$ $V_{1.1}$ $N_{1.3}$	$A_{1.1}$ $N_{1.2}$ $N_{1.1}$ $N_{2.3}$ $V_{1.1}$ $N_{1.3}$ 1.5 1 3 1.5 1 1	two rivers* the moonlight of a deer divides forest
Haiku 1.7	**Frame 1.7_code**	**Crossover termination**	**Frame 2.7_code**	**Haiku 2.7**
muzzle lake touches the water into moonlit deer	$N_{2.2}$ $N_{2.1}$ $V_{2.1}$ $N_{2.4}$ $A_{2.1}$ $N_{2.3}$ 1.5 1.5 1.5 1.5 1.5 1.5	$N_{2.2}$ $N_{2.1}$ $V_{2.1}$ $N_{2.4}$ $A_{2.1}$ $N_{2.3}$ $A_{1.1}$ $N_{1.2}$ $N_{1.1}$ $N_{1.4}$ $V_{1.1}$ $N_{1.3}$	$A_{1.1}$ $N_{1.2}$ $N_{1.1}$ $N_{1.4}$ $V_{1.1}$ $N_{1.3}$ 1.5 1 3 1.5 1 1	two rivers* he moonlight of a night* divides forest

Fig. 3. Semantic evolution of haikus (genotypes) and evolution of frame codes (phenotype codes)
* indicates where a correction is made in case of grammatical incongruity



Evolved frames of haiku 1 (phenotype group 1)

Evolved frames of haiku 2 (phenotype group 2)

Fig. 4. Perspective view, front and back elevations of the physical model of sequential frames

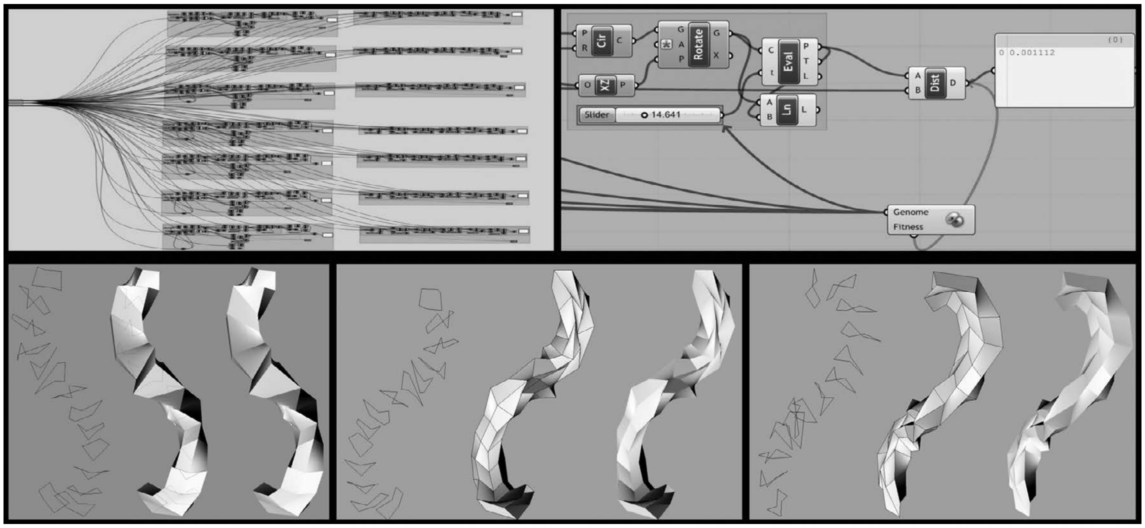


Fig. 5. Galapagos component functioning via the use of genetic algorithm and different organizations of frames