Proceedings of the ASME 2023
International Design Engineering Technical Conference
IDETC2023
August 20–23, 2023, Boston, MA

IDETC2023-117063

DRAFT: GENERATIVE DESIGN OF STATISTICALLY SELF-SIMILAR MECHANICAL STRUCTURES

Noah Hill[1], Matt Ebert[1], Mena Maurice[1], Vinayak Krishnamurthy[1,2,*]

[1]J. Mike Walker '66 Department of Mechanical Engineering, Texas A&M University, College Station, TX
[2]Department of Computer Science and Engineering (affiliated), Texas A&M University, College Station, TX

ABSTRACT

We present a novel methodology to generate mechanical structures based on the idea of fractal geometry as described by the chaos game. Chaos game is an iterative method that generates self-similar point-sets in the limiting case within a polygonal domain. By computing Voronoi tessellations on these point-sets, our method generates mechanical structures that adopts the self-similarity of the point-sets resulting in fractal distribution of local stiffness. The motivation behind our approach comes from the observation that a typical generative structural design workflow requires the ability to generate families of structures that possess shared behavioral (e.g. thermal, mechanical, etc.) characteristics making each structure distinct but feasible. However, the generation of the alternatives, almost always, requires solving an inverse structural problem which is both conceptually and computationally challenging. The objective of our work is to develop and investigate a forward-design methodology for generating families of structures that, while not identical, exhibit similar mechanical behavior in a statistical sense. To this end, the central hypothesis of our work is that structures generated using the chaos game can generate families of self-similar structures that, while not identical, exhibit similar mechanical behavior in a statistical sense. Furthermore, each family is uniquely identifiable from the parameters of the chaos game, namely, the polygonal domain, fractional distance, and number of samples. We present a systematic study of these self-similar structures through modal analysis and demonstrate a preliminary confirmation of our hypothesis.

Keywords: Fractals, Voronoi Tessellation, Generative Design, Chaos Game

1. INTRODUCTION

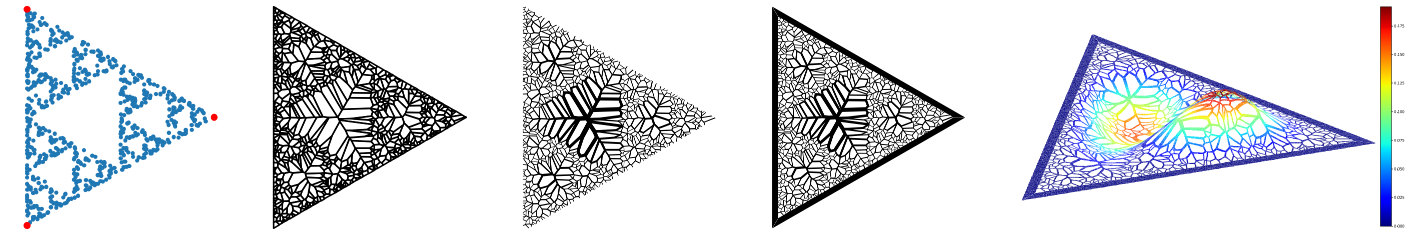
Generative design of engineered structures is now a popular area of research spanning domains including structural mechanics, acoustics, and thermo-fluidics. In a typical generative structural design workflow, the designer defines a spatial domain along with some mechanical conditions and constraints and the modeling system generates a population of feasible structural alternatives to choose from. A fundamental requirement for a such a workflow is the ability to generate families of structures that possess shared behavioral (e.g. thermal, mechanical, etc.) characteristics making each structure "*distinct but feasible*". However, the generation of the alternatives, almost always, requires solving an inverse structural problem which is both conceptually and computationally challenging [1–3].

The objective of this work is to develop and investigate a forward-design methodology for generating families of structures that, while not identical, exhibit similar mechanical behavior in a statistical sense. More importantly, we seek a methodology that offers explicit parameters to control the mechanical behavior of a structural system. To achieve these goals, we introduce an algorithm to generate a new class of structures, namely *self-similar* structures, inspired by fractal geometry. Our methodology is based on the well-known fractal algorithm known as the chaos game, which is a simple and powerful method to generate fractal geometry. Using point-sets generated using the chaos game, our methodology utilizes the well-known Voronoi tessellation to generate self-similar structures with statistically shared mechanical behavior (Figure 1).

1.1 Rationale & Background

Fractals offer a unique property that helps generate similar structures since fractals can be defined as consisting of smaller parts similar to itself, commonly called recursive self-similarity [4]. Self-similarity is an important property of natural structures (e.g. trees, nacre structures, etc.) and is particularly useful in structural design problems [5, 6]. However, much of the literature primarily uses L-systems and grammar-based algorithms [5, 7]. The common approach of investigation in these approaches is to determine the right parameters for the algorithm to generate the optimal structure for a given application rather than explore methods to generate an entire design space of potentially feasible designs. As a result, very little is explored or understood regard-

*Corresponding author: vinayak@tamu.edu
Documentation for `asmeconf.cls`: Version 1.34, April 28, 2023.



(a) Point Generation (b) Voronoi of Generated Points (c) Area Based Line Thickening (d) Adding Grips (e) Modal Analysis Results

FIGURE 1: THE METHOD USED TO GENERATE THE POINTS (A), VORONOI DECOMPOSITION (B), AND THE RESULTING STRUCTURE(C). IN ORDER TO TEST THE STRUCTURE WE ADD GRIP SECTIONS ON THE EXTERIOR OF THE STRUCTURE WITH SMALL GAPS BETWEEN ADJACENT GRIPS (D). TO EVALUATE THE STRUCTURES WE UTILIZED MODAL ANALYSIS (E). THE PARAMETERS USED FOR GENERATION ARE $n = 3$, $\lambda = 0.5$, AND $t = 750$.

ing stochastic similarity in mechanical or other physical properties of structures generated using existing fractal-based structural design.

In contrast to prior works, our objective is to embody the idea of generating populations of feasible alternatives rather than generating one optimal solution to a structural problem. In order to achieve this objective, we identify the chaos game as a potent direction for algorithmic investigation of self-similar structure generation. Chaos game is a well-known iterative process that can be used to create fractal geometry in the form of point-sets through repetitive and randomized generation of points using a polygonal domain [8, 9]. Chaos game has been used in the biomedical field [10, 11], plant modeling [12], and computer graphics [4].

In order to understand why chaos game is an interesting direction, let us consider a simple example — the Sierpinski gasket. A typical way to generate the Sierpinski gasket is through iterative subdivision using L-systems and turtle graphics [13]. However, the Sierpinski triangle can also be generated by using the chaos game by starting with a triangle and generating a series of points based on randomized linear interpolation with the vertices of the triangle (Figure 2). What is critical to note here is that while the L-systems approach gives the exact geometry of the gasket for each iteration, the chaos game only does so in the limiting case (i.e. when the number of iterations tend to infinity). This has two implications. First, the number of iterations in the chaos game controls the level of subdivisions for the gasket. Second, even for the same number of finite iterations, one gets a completely different point-set because of randomization. In conjunction, both these implications mean that for a given polygon (a triangle in this case), an entire family of geometrically *similar* point distributions (and therefore structures) can be generated by using merely a few parameters (the number of iterations and the interpolation parameter). The question is whether this geometric similarity carries forward into mechanical behavior.

1.2 Technical Approach

Chaos game is an iterative method that generates self-similar point-sets in the limiting case within a polygonal domain. By computing Voronoi tessellations on these point-sets, our method generates mechanical structures that adopt the self-similarity of the point-sets resulting in the fractal distribution of local stiffness.

Our work aims to generate families of non-identical structures with similar mechanical behavior, which are generated in the same manner. In this way, our process is a generative method to create structures. Furthermore, each family is uniquely identifiable from the parameters of the chaos game, namely, the polygonal domain, point interpolation distance, and the number of samples. We present a systematic study of these self-similar structures through modal analysis and demonstrate a preliminary confirmation of our hypothesis.

2. RELATED WORK

Our work spans multiple overlapping fields of research in structural design each of which is quite extensive. Here, we discuss works that are either methodologically or contextually relevant to our work.

2.1 Unit-cell Structural Design

Structural design has a rich history with several algorithms that seek to develop structural systems with specific physical properties. For example, work by Chu et al. [14] considers the design of cellular structures especially for additive manufacturing. Similarly, we see several works that focus on lattice structures for creating auxetic (negative Poisson's ratio) structures [15–17]. What is common in these approaches is that they are based on arrangements of some or the other form of a unit cell (often symmetric) and the arrangements are constrained according to some underlying grid-structure. The idea is that one can tune macro- and meso-scale properties by varying a few parameters pertaining to the unit cell geometry [18]. However, the design of the unit-cell, in itself, is not a trivial task. In fact, it is either ad hoc and based on trial-and-error or requires significant expertise and intuition.

2.2 Topology Optimization

Another widely practiced research direction in this regard is that of topology optimization, wherein the typical goal is to optimize (maximize or minimize) a specific criterion specified by the designer [19, 20]. In these cases, the designer explicitly defines

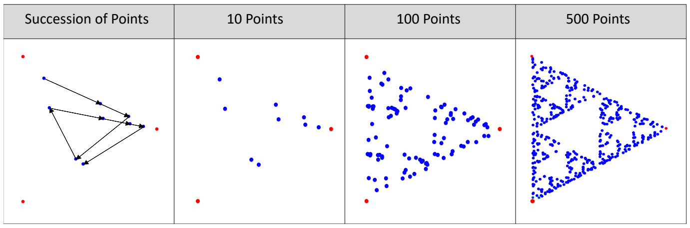


FIGURE 2: CREATION OF POINTS USING CHAOS GAME ALGORITHM WHERE n = 3 AND λ = 0.5

the mechanical loads, constraints, and boundary conditions, and a computer generates a single structure that meets the criteria with the set conditions. In these cases, only a single *optimal* design is generated, and if the designer needs a different structure, the design problem must be redefined. There are also works is based on using topology optimization combined with generative design based on L-Systems for the creation of graph-based structures [21, 22].

2.3 Learning-based Structural Design

Many recent works consider the inverse design approach using machine learning methods to generate 2D structures with tunable properties [23]. Recently, we see topology optimization approaches in conjunction with deep learning methods such as convolutional neural networks (CNN) [24] to generate multi-scale structures spanning micro- and macro-scales. Genetic algorithms have also been used to create optimal designs using a Pareto frontier and multiple objectives [25]. There has also been uses of machine learning algorithms in order to optimize biologically inspired patterns [26].

2.4 Voronoi-based Cellular Structures

One method to create a large number of patterns revolves around using Voronoi decomposition of points in order to obtain cellular structures. One area that has used this is the design of 2D infill structures for 3D printing[27]. In a similar manner 3D Voronoi decomposition can be used to create 3D printed structures with anisotropic behavior which can be helpful in certain application with uneven or varied loading conditions present [28]. Another way that Voronoi decomposition has been used is to generate foam structures which has application in many different areas such as topological interlocking and energy absorption [29]. One interesting application of this is in 2.5D tile generation wherein different 2D Voronoi layers are stacked on top of one another in order to create a 3D structure from 2D Voronoi decomposition [30]. Voronoi decomposition has also been used to create metamaterial structures which be very beneficial since they do not rely on the design of a single unit cell but rather the design of the Voronoi sites [31, 32].

2.5 Fractal-based Structural Design

The chaos game has been used to explore the mechanical properties of fractals when applied as lattice structures [33–35]. This is done by geometrically defining a structure through lengths, widths, and thicknesses, which can be modified to change mechanical behavior. Similar works have done experiments on hierarchical structures, which can be defined by self similarly at multiple dimensions [36–38].

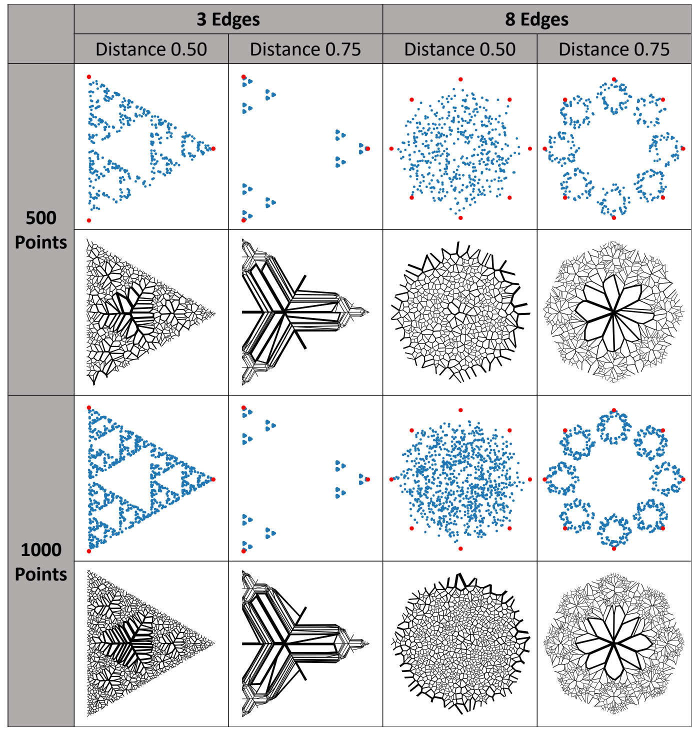


FIGURE 3: STRUCTURES GENERATED USING DIFFERENT COMBINATIONS OF PARAMETER SETS ARE SHOWN HERE. THE EFFECTS OF VARYING NUMBER OF VERTICES (n), NUMBER OF POINTS (t), AND FRACTIONAL DISTANCE MOVED ALONG THE LINE (λ) ON THE GENERATED POINTS AND STRUCTURES ARE DISPLAYED.

2.6 Our Work

Our work leverages the strength of fractal-based approaches (especially the chaos game) for inducing controllable stochastic variability along with Voronoi-based approaches for elegant topology generation for structure generation. This powerful combination provides advantages over prior approaches by introducing an intuitive way for both parametric control as well as structure generation. In effect, this provides us a means for direct creation of entire families of structures that behave in some stochastically similar manner without the need for inverse design.

3. CONCEPTUAL PRELIMINARIES

Chaos theory has been used to define a method for creating fractals, commonly called the chaos game [39–41]. The chaos game is an algorithm used to generate a fractal that is described as an Iterative Function System, which is a set of pairs of linear mappings [39, 40]. Many different fractals can be generated using different parameters. One of the most common fractal patterns that can be created using this process is the Sierpiński triangle [41, 42]. In fact, the same process that is used to create the Sierpiński gasket and the Menger sponge, can be used to create fractals for other polygons.

3.1 Chaos Game

The process that the chaos game uses to generate fractals is defined as follows (Figure 2). Consider a polygon, P_n, where n is the number of vertices of the polygon. The polygon's vertices are defined by the set of points $\{p_1, p_2, ..., p_n\}$. For each vertex, the value of $p_i = (cos\theta_i, sin\theta_i)$, where $\theta_i = \frac{2\pi * i}{n}$. This creates a regular polygon with n number of vertices. Consider a randomly placed point $q_0 \in \mathbb{R}^2$. We define a function $R(n) : [1, n] => i$ which provides the random integer, i, from 1 to n. Using this function, the i-th vertex of P_n, which is p_i, is chosen. Using this randomly selected vertex, p_i, and the randomly chosen point, q_0, a new point can be defined using the following equation $q_1 := q_0 + \lambda(p_i - q_0)$, where λ is the fractional distance moved along the line connecting the vertex, p_i, and the initial point, q_0, and defined as $\lambda \in [0, 1]$. This process can now be repeated for the new point, q_1. Using t to represent the current iteration, the following equation for finding the next point can be defined.

$$q_{t+1} \leftarrow q_t + \lambda(p_{R(n)} - q_t) \tag{1}$$

3.2 Parameters

This gives use three parameters for the chaos game: n, λ, and t. The first parameter n is the number of vertices of the polygon, and these vertices are often called attractors in previous literature [39]. Different regular polygons will be generated depending on the number of vertices, changing the options available for p_i and the possible values of q_t (Figure 3). The second parameter is λ, or the fractional distance moved along the line connecting p_i and q_t. The fractional distance moved along the line, λ is confined to be between 0 and 1. When $\lambda = 0$, $q_{t+1} = q_t$, conversely when $\lambda = 1$, $q_{t+1} = p_i$. This means that if λ is closer to 1, the points will be more clustered towards the vertices of the polygon, while if λ to 0, the points will be more clustered in the center of the polygon (Figure 3). The third parameter is t or the number of iterations being run. The current iteration, t, is defined as $t \in [t_0, T]$, where $t_0 > 0$ and T is the number of iterations. t_0 must be greater than 0 because the initial point q_0 is not guaranteed to be within the bounds of P_n. Changing the number of points generated changes how densely the polygon is filled, which changes the structure being generated and how that structure may react when tested (Figure 3).

3.3 Generating Points

By defining all three parameters, a set of points can be generated using the defined algorithm (Figure 2). The set of points is defined as $S := \{q_t\}$, where all q_t are inside the main polygon. The set of points will be called the *chaos sites*. These *chaos sites* can be used to create the structures being studied (Figure 1a).

Since the number of points being generated is finite, it is possible to get different sets of points with the same parameters since the starting random point differs each time. Additionally, successive iterations of $R(n)$ return different chosen points each time even if the initial point is the same. It is for this reason that this process is generative since, for the same set of parameters, there will be a different (but similar) set of points generated (Figure 4). Different sets of parameters will also result in very different sets of points (Figure 3). For example, five runs with the polygon being a triangle, fractional distance moved along the line being 0.5, and the number of points being 750 will produce five triangles that have a similar clustering of points, but the structure will not be the same across all five runs (Figure 4).

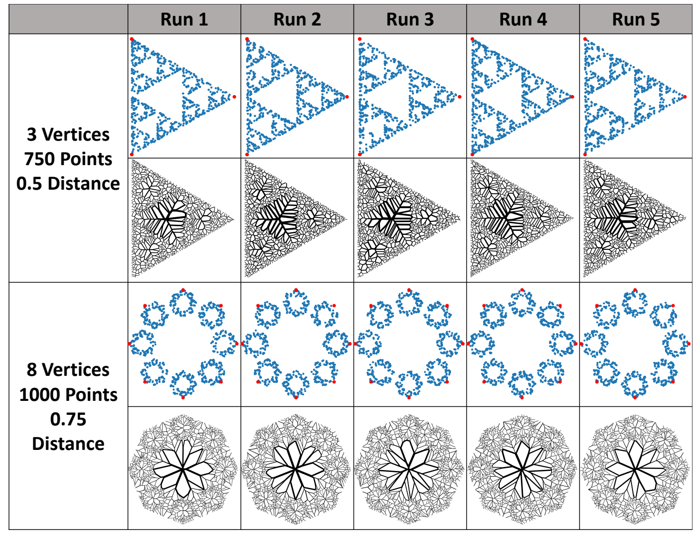


FIGURE 4: FIVE DIFFERENT RUNS FOR THE SAME SET OF PARAMETERS PRODUCE SIMILAR, BUT NOT THE SAME, RESULTS FOR THE GENERATED POINTS AND VORONOI DECOMPOSITION SHOWING THAT THE PROCESS IS GENERATIVE. TWO DIFFERENT SETS OF PARAMETERS ARE SHOWN HERE.

4. STRUCTURE GENERATION METHODOLOGY

The *chaos sites*, S, generated using the Chaos Game algorithm can be used to create a structure. The methodology used to generate this structure has two main components: (1) computing a Voronoi tessellation of S, (2) thickening the edges of the Voronoi tessellation.

4.1 Voronoi Decomposition

Using the points contained in S as Voronoi sites, a Voronoi tessellation is computed resulting in a partitioning of a given polygonal domains. To create our structures, a Voronoi tessellation is calculated for the *chaos sites*, S, creating a set of cells for the structure. Note that the distribution of points in S is an approximation of some fractal geometry. As a result, the tessellation of this set results in a partition wherein the cell areas are distributed in a manner inversely proportional to the point density thereby giving the structural a fractal-like property. The unbounded cells resulting from Voronoi tessellation are trimmed according to the polygonal domain boundary (Figure 1b).

4.2 Edge Thickening

Given a Voronoi tessellation of the polygonal domain, our next step is to thicken the edges in the tessellation. As such, this can simply be achieved by creating offset polygons for each Voronoi cell in the tessellation. However, an important consideration here is that the fractal-like distribution of the chaos sites results in a high variation in the areas of the Voronoi cells. Therefore, we employ an adaptive strategy wherein, we determine the offset for each cell based on the measure of the cell areas normalized with respect to largest and smallest cells in the tessellation (Figure 1c). Specifically, we begin by computing the maximum and minimum cells areas (A_{max} and A_{min} respectively. For a given Voronoi cell with an area A, we then compute the *normalized area*, $\hat{A}$, as follows (Equation 2):

$$\hat{A} = \frac{A - A_{min}}{A_{max} - A_{min}} \tag{2}$$

Consider an edge e shared by two Voronoi cells f_j and f_j. Then, the thickness of the edge is given by $\tau_e = \omega_1 + \omega_2$, where ω_i and ω_j is the offset applied to f_i and f_j. Note that for a non-adaptive thickening, this would simply amount to $\tau_e = 2\omega$ where $\omega_1 = \omega_2 = \omega$ is a constant offset. However, in our adaptive case, the idea is to compute the offset based on the normalized areas. For this, we define ω_{max} and ω_{min} as the maximum and minimum possible polygon offsets respectively. For a cell f_i with normalized area $\hat{A}_i$, the offset ω_i is calculated by linearly interpolating between the offset limits as (Equation 3):

$$\omega_i = \hat{A}_i(\omega_{max} - \omega_{min}) + \omega_{min} \tag{3}$$

Therefore, the thickness of an edge shared by two Voronoi cells f_j and f_j with normalized areas $\hat{A}_i$ and $\hat{A}_j$, the thickness $\tau_e = (\hat{A}_i + \hat{A}_j)(\omega_{max} - \omega_{min}) + 2\omega_{min}$.

4.3 Grip Generation for Modal Analysis

In our work, we specifically aim to investigate our shape generation methodology in terms of comparing structures based on their natural frequencies. In order to do so, we implemented a grip generation step in our computational framework. This is primarily done to ensure the application of appropriate boundary conditions for our analysis. Having said this, this is not a fundamentally necessary step toward the generation of the actual self-similar structure and we have added it for completeness.

To create the grips, we first generate an outward offset for the polygonal domain. This results in a region between the original and the offset polygon. We then split this region into n pieces (where n is the number of sides of the domain) simply by connecting each pair of corresponding vertices in the two polygons. This results in n distinct grips rather than a single body (Figure 1d). After the grips have been added, the sample is extruded and triangulated to make it a 3D structure.

5. EXPERIMENTAL DESIGN

The central claim behind our method is that the geometric similarities induced by each family of parameters of the chaos game reflects in the corresponding mechanical behavior. In order to investigate this claim, we conducted a series of numerical experiments wherein we used modal analysis as our choice of physical characterization. The idea was to perform comparative statistical analyses of natural frequencies of structures across different selective parameter families (polygonal domain — n, number of iterations — T, fractional distance — λ). Below, we provided details regarding the design of our experiments.

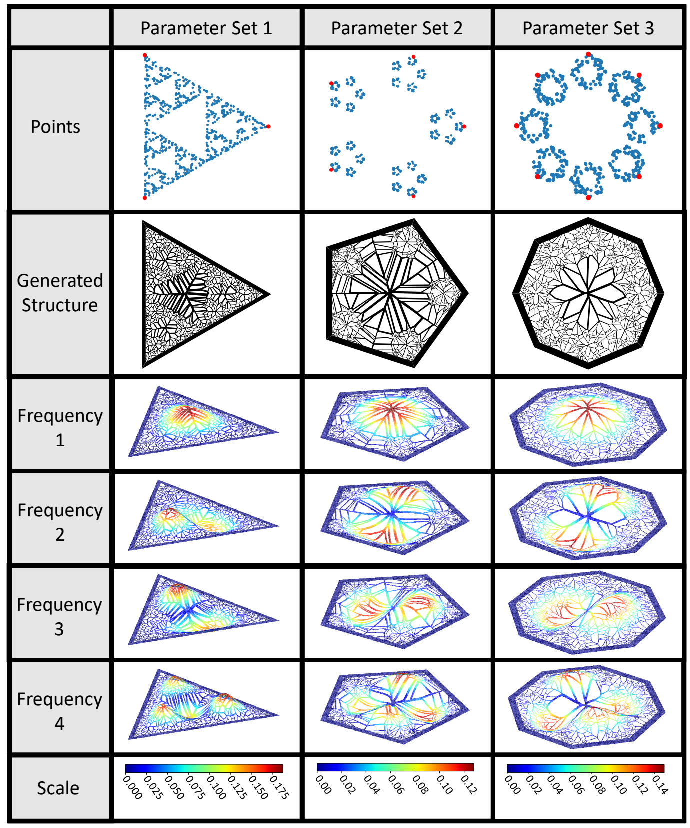


FIGURE 5: DIFFERENT PARAMETERS FOR SHAPE GENERATION ARE SHOWN HERE. THE DEFORMATION OF THE FIRST 4 MODES IS ALSO SHOWN WITH THE SCALE BAR SHOWING THE TOTAL DEFORMATION IN METERS. AN IMPORTANT NOTE IS THAT THE GRIPS SHOWN IN THE GENERATED STRUCTURE ARE DISCONNECTED BY A VERY SMALL MARGIN.

5.1 Modal Analysis

We chose modal analysis as our context for two reasons, First, it gives is a concrete physical context (vibrations of a dynamical system) which we can easily quantify in terms of natural frequencies. Second, the modes of a structural system are fundamentally connected to the geometry and topology of the system. Consequently, this would allow us to make objective comparisons of the mechanical behavior of each structure by examining the natural frequencies and mode shapes of each geometry [43, 44]. Based on these reasons, modal analysis is an ideal candidate for a preliminary exploration of how statistical shape similarities can carry over to mechanical properties.

We implemented our experiments using ANSYS. Each grip's nodes are fixed which prevents the grips from moving allowing only the inside structure to be examined with the modal analysis. Once the grips are fixed, the modal analysis is performed, and the results can be analyzed.

5.2 Hypothesis

We predict that using the same parameters, the generated structures will not have statistically different mechanical properties over multiple iterations. This hypothesis aims to test if the parameter space defined by the chaos game and using our generation methodology results in structures that belong to the same family which could all be *feasible* design solutions. Ultimately this would show that our methodology is a generative process for creating structures.

If the previous statement holds true, we expect that as we change λ then there should be a linear relationship in the frequency magnitude. This relationship can be expected because λ is the distance moved along a line connected the randomly selected polygon point and the previous point. For this reason we expect a linear relationship between λ and the magnitude of the frequency.

5.3 Experiment 1

To test the hypothesis that structures generated with the same set of parameters will have statistically similar results, the same set of parameters was simulated 100 iterations. The parameters n, λ, and t were set to 3 vertices, 0.5 fractional distance moved along the line, and 750 points, respectively. The first ten natural frequencies were found for each of the 100 runs in order for comparison. For each shape generated, we recorded 10 natural frequencies along with the deformation of the shape.

5.4 Experiment 2

The goal of the second hypothesis is to determine the relationship between λ and the resulting frequencies. In order to test this, a second experimental setup was devised where with a set value of n, the value of λ would vary. The number of points was set be set to 750 points. Additionally for λ the values were the following: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. For each shape generated, we recorded 5 natural frequencies along with the deformation of the shape. A total of 25 total iterations were run at each value of λ. Additionally the values of n were chosen to be 3, 5, 6, and 8. This resulted in a 255 total tests for each value of n.

6. RESULTS

6.1 Experiment 1

A one-way ANOVA shows that the null hypothesis is rejected. In other words, there is there is a significant difference in the population means of the data. This, in turn, supports our central hypothesis that structures generated using the same parameters have similar mechanical properties. This is further reinforced by the facts that standard deviations for each given population is low (Table 1). It is worth noting, however, that the standard deviations generally increase as the frequency increases. This is indicated by the observation that the standard deviation for frequency 1 is $2.04Hz$, whereas for frequency 10 it increases to $13.06Hz$.

As modal frequency increases, the magnitude of the frequencies begin to pair off with one another, with frequency 1 being the exception as it is the only category dissimilar to the other frequencies (Figure 6). We can observe this pairing off effect in frequencies 2 and 3 which have similar results at around a frequency of $440Hz$, and frequencies 4, 5, and 6 which also share similar results at around a frequency of $700Hz$. Frequencies 7, 8, 9, and 10 also pair together at a value around $1000Hz$. An interesting observation is that at higher frequency numbers the number of frequencies in a pair increases. This is observed first with the first pairing only containing two frequencies, the next having three frequencies and the next having four frequencies. Since many of these frequencies pair together, we used a two-sample t-test in order to verify that these frequencies are from two independent populations, rather than samples from a single population. In each case, the p-value for comparison was < 0.005 showing that it is likely that the pairing originates from two separate populations. Similarly to the pairing behavior of the frequencies, it can be observed that the standard deviations of the different frequency groups pair together. This can be further demonstrated as the standard deviation difference between frequencies 2 and 3 is $0.39Hz$ and are near $3.4Hz$, whereas, the standard deviations for frequencies 4, 5, and 6 are all near $7Hz$.

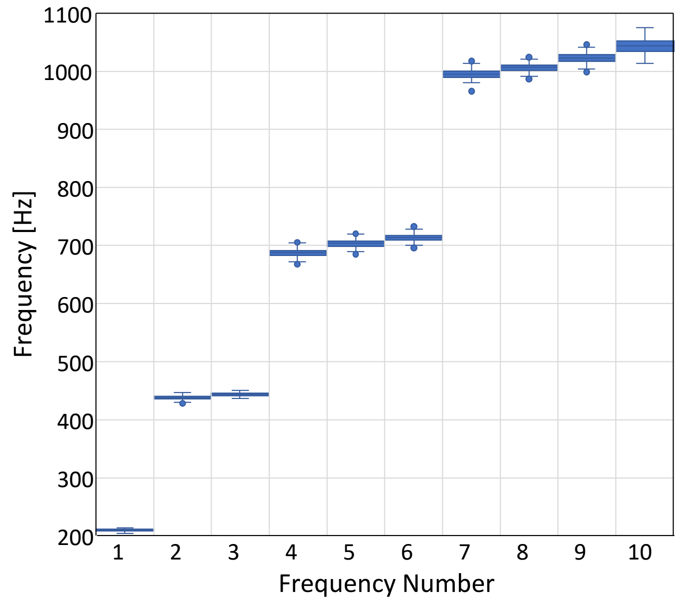


FIGURE 6: FIRST 10 FREQUENCIES FOR THE SAME SET OF PARAMETERS (n = 3, λ = 0.5, AND t = 750) AFTER 100 RUNS.

6.2 Experiment 2

For experiment 2, we tested if the fractional distance moved along the line, λ, has a linear relationship with the frequency. For each of the different polygons tested we conducted a two-way ANOVA which showed that the null hypothesis was rejected in each case, meaning that there are significant differences in mean values for each of the two variables, λ and frequency number. Additionally, the tests also showed that there is an interaction between the two factors. Some general results were found for all the polygons tested. There appears to be a distance at which each frequency value peaks for each polygon (Figure 7). For example, for the triangle (n = 3), the frequency values peak at λ = 0.4. Also, the lowest frequency value for the structures tested tends to occur at λ = 0.9 (Figure 7). Further, frequencies 1-3 change less as λ increases from one value to the next, while frequencies 4-5 change more as λ increases.

6.2.1 Triangle. For the triangle structures (n = 3), the frequency magnitudes tend to form pairs at many values of λ (Figure

Frequency Number	1	2	3	4	5	6	7	8	9	10
Mean (Hz)	209.5	438.0	443.9	687.3	703.3	713.2	994.5	1006.7	1022.7	1043.7
Standard Deviation (Hz)	2.04	3.61	3.22	7.63	7.05	6.68	8.69	7.30	8.43	13.06

TABLE 1: AFTER 100 RUNS WITH THE SAME INPUT PARAMETERS THE AVERAGE AND STANDARD DEVIATION ARE SHOWN.

7). This can be seen as frequency 1 is often dissimilar to the other values and frequencies 2 and 3 are often close in magnitude. In a similar manner, frequencies 4 and 5 tend to pair together. For the triangle structures, the pairings stay consistent up to $\lambda = 0.6$; afterwards, the pairings are no longer similar to $\lambda < 0.6$. Further, for $\lambda = 0.9$, the magnitudes of all five frequencies are close; however, through t-test comparison, it was determined that each frequency likely belongs to its own population. For $\lambda = 0.1$, the standard deviation of the frequency values is higher than the standard deviations of the same frequencies for the other λ values. More specifically, the standard deviations of frequencies 4 and 5 are high compared to the standard deviations of frequencies 4 and 5 for the other λ values. The magnitude of the frequency values peak near $\lambda = 0.4$, afterwards there is a steady decrease in the values. The magnitudes of the first five frequencies are also higher than the magnitudes of the other polygons with a maximum near $850Hz$.

6.2.2 Pentagon. For the pentagon structures ($n = 5$), the frequency magnitudes form similar pairs to the triangle structures, except the pairings stay consistent up to $\lambda = 0.8$ (Figure 7). Further, for $\lambda = 0.9$ the magnitudes of all the frequencies except frequency 1 are close in magnitude which is similar to what was observed with the triangle at $\lambda = 0.7$. The magnitude of the frequency values peak near $\lambda = 0.5$. As a whole, the standard deviations for the frequency values are small for the pentagon structures. The magnitudes of all the frequency values are smaller than their counterparts (same λ and frequency number) for the triangle structures, appearing to be around half the magnitude in most cases. The maximum frequency value is around $400Hz$.

6.2.3 Hexagon. The frequency magnitudes for the hexagon structures ($n = 6$) follow the same pairing as the pentagon, but stay consistent longer, including $\lambda = 0.9$ (Figure 7) which is the highest value of λ that maintains the pairing. The magnitude of the frequency values peaks around $\lambda = 0.5$, with a maximum frequency value of around $350Hz$. Frequencies 4 and 5 at $\lambda = 0.1$ and $\lambda = 0.9$ have a higher standard deviation than the other frequency standard deviations. For 0.2 through 0.8 fractional distance moved along the line ($\lambda = 0.2 - 0.8$), the standard deviations of the magnitudes of the frequencies are smaller in comparison with the previous values. The hexagon structures also tend to have lower frequency magnitudes than their counterparts for the pentagon and triangle; however, there is not as steep of a decrease from pentagon to hexagon as there was from triangle to pentagon.

6.2.4 Octagon. The same pairing seen before in the other structures occurs for the octagon structure ($n = 8$) (Figure 7). The pairings stay consistent up to and including 0.9 fractional distance moved along the line ($\lambda = 0.9$), just like the hexagon structures. Frequencies 4 and 5 also do not drop in magnitude as much from $\lambda = 0.7$ to $\lambda = 0.9$ as they did for the hexagon structures. The magnitude of the frequency values peak near $\lambda = 0.5$, similar to the pentagon and hexagon. The octagon structures also tend to have lower frequency magnitudes than their counterparts for the hexagon. For 0.7 through 0.9 fractional distance moved along the line ($\lambda = 0.7 - 0.9$), the magnitudes of frequency 1 are close to each other. This same pattern occurs for frequencies 2 and 3.

Some interesting general results can also be noted. There seems to be a point where the distance along the line starts affecting the frequency values less (Figure 7). For the pentagon, hexagon, and octagon structures where $\lambda = 0.7 - 0.9$, the magnitudes for frequency 1 are similar. For the hexagon and octagon structures where $\lambda = 0.7 - 0.9$, the magnitudes for frequencies 2-3 are similar. Also, as the number of vertices, n, increases, the pairings stay consistent for higher values of λ (Figure 7). For the triangle, the pairings are only consistent up to $\lambda = 0.5$, while for the octagon, the pairings are consistent up to $\lambda = 0.9$. Further, as the number of vertices, n, increases, the magnitude of the frequency values decreases except for when pairing happens (Figure 7). For example, frequency 1 for $\lambda = 0.1$ for the triangle is higher than frequency 1 for $\lambda = 0.1$ for the pentagon, which is higher than frequency 1 for $\lambda = 0.1$ for the octagon. However for $\lambda = 0.9$, since the triangle ($n = 3$) does not have pairing at this λ, the magnitudes of frequencies 2 and 3 actually increase for the pentagon ($n = 5$) because pairing occurs.

The results can also be visualized by analyzing the mode shapes and displacement of the structure at the frequencies found (Figure 5). Several interesting results were found. For frequency 1 with all the polygons, the largest displacement occurred in the center of the structure. For frequencies 2 and 3 for all the polygons, there is a hill (positive displacement in z) on one side, while there is a valley (negative displacement in z) on the other side. Frequencies 2 and 3 have similar-looking structural displacements, but the hills and valleys are located in different sections. The displacement of the triangle structure for frequency 4 does not look like the displacement of the pentagon and octagon structures for frequency 4. The triangle structure has three hills located near the vertices and one valley located in the center of the structure. The pentagon and hexagon structures have two hills and two valleys and appear similar to each other.

7. DISCUSSION

7.1 Connecting Mechanics to Geometry

Overall, we observe a strong connection between mechanical properties and the chaos game parameters. For example, an increase in the number of vertices of the polygon generally results in a decrease in the magnitude of the frequencies. We also observe a non-linear relationship between λ and frequency which was not expected since λ only changes the distance along a line. This was most notable in the triangle and pentagon case where $\lambda = 0.9$ results in a similar behavior for all frequency values. These overarching observations strongly indicate a fundamental

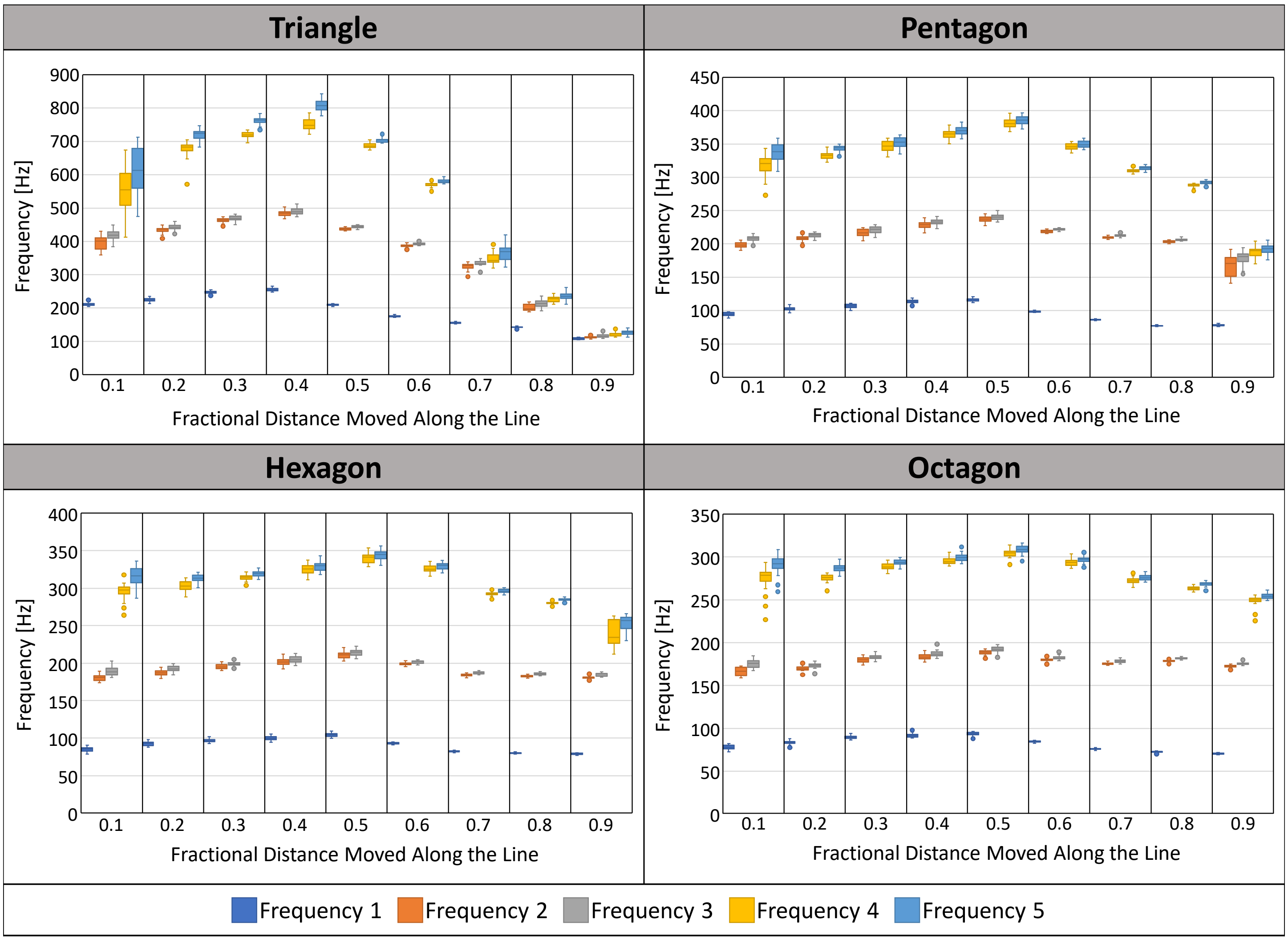


FIGURE 7: THE FIRST FIVE FREQUENCIES FOUND FOR THE STRUCTURES GENERATED WITH THE FRACTIONAL DISTANCE MOVED, λ VARYING FROM 0.1 – 0.9 BY AN INTERVAL OF 0.1. EACH SET OF PARAMETERS WAS RUN 25 TIMES AND THE BOX PLOTS FOR EACH FREQUENCY NUMBER AT EACH DISTANCE ARE SHOWN.

connection between the geometric parameters and mechanical response, at least for natural frequencies. Having said this, similar experiments are needed for other mechanical responses (e.g. deformation under quasi-static compression and tension) to ascertain these relationships.

7.2 Behavioral Patterns

One behavioral pattern we observe is that a value of λ at which the frequencies appear to reach a maximum value. This is related to how sensitive a given polygon (n) is to the λ value. For example, while $\lambda = 0.5, n = 3$ results in a Sierpiński triangle, $\lambda = 0.5, n = 8$ (i.e. octagon) has no discernible fractal (Figure 3). Furthermore, for the triangle there is a decrease in modal frequencies only near $\lambda = 0.5$ (i.e. the Sierpiński triangle case). For an octagon and triangle at $\lambda = 0.75$ there is a clear fractal structure appearing which may relate to the decrease in the modal frequencies observed in octagon before $\lambda = 0.75$. In another example, we find that for $n = 5$ (i.e. a pentagon), the Sierpiński-type fractal occurs only at $\lambda = \frac{2}{3}$. This could be a reason why there is a decrease in the modal frequencies near $\lambda = \frac{2}{3}$ for the pentagon.

Another behavioral pattern that we observe occurred in nearly all of the structures was that of pairing of the results. This can first be observed in our first experiment wherein the second and third natural frequencies have close median values while the fourth, fifth, and sixth frequencies form another cluster of close median values (Table 1). Similarly, we observe clustering of median values for several other cases as well (Figure 7). The only structures where this pairing did not occur was for high values of λ in the triangle and pentagon. In order to validate that each member of a pairing originated from a separate population a t-value comparison was conducted. In every case the $p < 0.05$ meaning that there is reason to believe that members of a pairing originate from different populations. The largest p-values that we found in our selective t-tests was $p = 0.04$ for $(n = 6, \lambda = 0.5)$ between fourth and fifth natural frequencies. Similarly, we get $p = .03$ for $(n = 3, \lambda = 0.9)$ and $(n = 5, \lambda = 0.9)$ between fourth and fifth frequencies. Even here, note that the p-value is safely below the threshold of $p < .05$. While we cannot posit regarding the reason for these relatively close distributions, we do observe visual similarity in terms of the deformation across the first few modes. For example, the second and third natural frequencies generally display high qualitative similarity for each polygonal domain (Figure 5).

7.3 Limitations and Future work

There are several questions that require further exploration in this work. First, there are several variations to the chaos game with extended rules for point generation. One example is to apply a preference model during the random selection of polygonal vertices that leads to completely new types of structures to emerge. Given that this is a rich design space, further expansive investigation is needed to further explore this aspect. Secondly, the application of chaos game to non-regular polygons is certainly worth exploring. The effect of polygonal asymmetry may shed some critical insights regarding behavior similarities. Adding to this, an obvious extension would be to consider arbitrary 2D domains wherein we can apply our method to triangulated domains. Finally, it is important to evaluate our current structures for different multi-physical responses. Another intriguing future direction is to extend the idea to 3D structures. Interestingly, this can be easily done since the chaos game works even for selective 3D polyhedral domain. As a result, it would be interesting to explore sponge-type as well as frame-type structures in 3D domains based on using the edges and faces of 3D Voronoi cells. Overall, we see an immense potential for self-similar structures in generative design.

8. CONCLUSION

In this paper, we introduced a forward-design approach for generative statistically self-similar structures based on fractal geometry. Using a combination of chaos game and Voronoi tessellation, we show that it is possible to generate families of structures whose geometric similarities carry forward in terms of mechanical response. We demonstrated this within the concrete context of natural modes of the structures generated using our method. Our experiments conclusively show that the parameters of the chaos game offer a controlled way to tune the mechanical response and enable the generation of populations of shapes rather than a single optimal shape. We further demonstrated tractable relationship across different parameters (especially fractional distance) of the chaos game. This is an essential requirement for generative design workflow. We believe that this work is merely a starting point to a potentially rich research direction in the domain of generative structural design.

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation (NSF) Awards #2048182 (Engineering Design and Systems Engineering Program). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

REFERENCES

[1] Shea, Kristina, Aish, Robert and Gourtovaia, Marina. "Towards integrated performance-driven generative design tools." *Automation in Construction* Vol. 14 No. 2 (2005): pp. 253–264.

[2] Buonamici, Francesco, Carfagni, Monica, Furferi, Rocco, Volpe, Yary and Governi, Lapo. "Generative design: an explorative study." *Computer-Aided Design and Applications* Vol. 18 No. 1 (2020): pp. 144–155.

[3] Oh, Sangeun, Jung, Yongsu, Kim, Seongsin, Lee, Ikjin and Kang, Namwoo. "Deep generative design: Integration of topology optimization and generative models." *Journal of Mechanical Design* Vol. 141 No. 11 (2019).

[4] Al-Jawfi, Rashad A, Al-Helali, Baligh M and Ahmed, Adil M. "Fractal image compression using self-organizing mapping." *Applied Mathematics* Vol. 5 No. 12 (2014): pp. 1810–1819.

[5] Rian, Iasef Md and Sassone, Mario. "Fractal-based generative design of structural trusses using iterated function system." *International journal of space structures* Vol. 29 No. 4 (2014): pp. 181–203.

[6] Fan, Jonathan A, Yeo, Woon-Hong, Su, Yewang, Hattori, Yoshiaki, Lee, Woosik, Jung, Sung-Young, Zhang, Yihui, Liu, Zhuangjian, Cheng, Huanyu, Falgout, Leo et al. "Fractal design concepts for stretchable electronics." *Nature communications* Vol. 5 No. 1 (2014): p. 3266.

[7] Mishra, Jibitesh and Mishra, Sarojananda. *L-system Fractals*. Elsevier (2007).

[8] Barnsley, Michael F and Demko, Stephen. "Iterated function systems and the global construction of fractals." *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences* Vol. 399 No. 1817 (1985): pp. 243–275.

[9] Schroeder, Manfred. *Fractals, chaos, power laws: Minutes from an infinite paradise*. Courier Corporation (2009).

[10] Deschavanne, Patrick J, Giron, Alain, Vilain, Joseph, Fagot, Guillaume and Fertil, Bernard. "Genomic signature: characterization and classification of species assessed by chaos game representation of sequences." *Molecular biology and evolution* Vol. 16 No. 10 (1999): pp. 1391–1399.

[11] Jampour, Mahdi, Yaghoobi, Mahdi, Ashourzadeh, Maryam and Solemani, Adel. "A new fast technique for fingerprint identification with fractal and chaos game theory." *Fractals-complex Geometry Patterns and Scaling in Nature and Society - FRACTALS* Vol. 18 (2010). DOI 10.1142/S0218348X10005020.

[12] Biswas, Balaka, Ghatak, Rowdra and Poddar, D. R. "A Fern Fractal Leaf Inspired Wideband Antipodal Vivaldi Antenna for Microwave Imaging System." *IEEE Transactions on Antennas and Propagation* Vol. 65 No. 11 (2017): pp. 6126–6129. DOI 10.1109/TAP.2017.2748361.

[13] Prusinkiewicz, Przemyslaw. "Graphical applications of L-systems." *Proceedings of graphics interface*, Vol. 86. 86: pp. 247–253. 1986.

[14] Rosen, David W. "Computer-aided design for additive manufacturing of cellular structures." *Computer-Aided Design and Applications* Vol. 4 No. 5 (2007): pp. 585–594.

[15] Ren, Xin, Das, Raj, Tran, Phuong, Ngo, Tuan Duc and Xie, Yi Min. "Auxetic metamaterials and structures: a review." *Smart materials and structures* Vol. 27 No. 2 (2018): p. 023001.

[16] Wang, Hongqing, Chen, Yong and Rosen, David W. "A hybrid geometric modeling method for large scale conformal cellular structures." *ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*: pp. 421–427. 2005. American Society of Mechanical Engineers Digital Collection.

[17] Wang, Hui, Zhang, Yuxuan, Lin, Wanqing and Qin, Qing-Hua. "A novel two-dimensional mechanical metamaterial with negative Poisson's ratio." *Computational Materials Science* Vol. 171 (2020): p. 109232. DOI https://doi.org/10.1016/j.commatsci.2019.109232. URL https://www.sciencedirect.com/science/article/pii/S0927025619305312.

[18] Ai, L. and Gao, X.-L. "Metamaterials with negative Poisson's ratio and non-positive thermal expansion." *Composite Structures* Vol. 162 (2017): pp. 70–84. DOI https://doi.org/10.1016/j.compstruct.2016.11.056. URL https://www.sciencedirect.com/science/article/pii/S0263822316316038.

[19] Eschenauer, Hans A and Olhoff, Niels. "Topology optimization of continuum structures: a review." *Appl. Mech. Rev.* Vol. 54 No. 4 (2001): pp. 331–390.

[20] Sigmund, Ole and Maute, Kurt. "Topology optimization approaches: A comparative review." *Structural and Multidisciplinary Optimization* Vol. 48 No. 6 (2013): pp. 1031–1055.

[21] Bielefeldt, Brent R, Akleman, Ergun, Reich, Gregory W, Beran, Philip S and Hartl, Darren J. "L-system-generated mechanism topology optimization using graph-based interpretation." *Journal of Mechanisms and Robotics* Vol. 11 No. 2 (2019).

[22] Ward, R Mason, Bielefeldt, Brent R and Hartl, Darren J. "Design of tailorable stiffness structures using L-system topology optimization." *Behavior and Mechanics of Multifunctional Materials IX*, Vol. 11377: pp. 112–122. 2020. SPIE.

[23] Tian, Jie, Tang, Keke, Chen, Xianyan and Wang, Xianqiao. "Machine learning-based prediction and inverse design of 2D metamaterial structures with tunable deformation-dependent Poisson's ratio." *Nanoscale* (2022).

[24] White, Daniel A., Arrighi, William J., Kudo, Jun and Watts, Seth E. "Multiscale topology optimization using neural network surrogate models." *Computer Methods in Applied Mechanics and Engineering* Vol. 346 (2019): pp. 1118–1135. DOI https://doi.org/10.1016/j.cma.2018.09.007. URL https://www.sciencedirect.com/science/article/pii/S004578251830450X.

[25] Garland, Anthony P., White, Benjamin C., Jensen, Scott C. and Boyce, Brad L. "Pragmatic generative optimization of novel structural lattice metamaterials with machine learning." *Materials Design* Vol. 203 (2021): p. 109632. DOI https://doi.org/10.1016/j.matdes.2021.109632. URL https://www.sciencedirect.com/science/article/pii/S0264127521001854.

[26] Podroužek, Jan, Marcon, Marco, Ninčević, Krešimir and Wan-Wendner, Roman. "Bio-Inspired 3D Infill Patterns for Additive Manufacturing and Structural Applications." *Materials* Vol. 12 No. 3 (2019). DOI 10.3390/ma12030499. URL https://www.mdpi.com/1996-1944/12/3/499.

[27] Ebert, Matthew, Subramanian, Sai Ganesh, Akleman, Ergun and Krishnamurthy, Vinayak R. "Generative Infills for Additive Manufacturing Using Space-Filling Polygonal Tiles." *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 84010: p. V11BT11A014. 2020. American Society of Mechanical Engineers.

[28] Icking, Christian, Klein, Rolf, Ma, Lihong, Nickel, Stefan and Weißler, Ansgar. "On bisectors for different distance functions." *Discrete applied mathematics* Vol. 109 No. 1-2 (2001): pp. 139–161.

[29] Howison, Mark and Séquin, Carlo H. "CAD Tools for

Creating Space-filing 3D Escher Tiles." *Computer-Aided Design and Applications* Vol. 6 No. 6 (2009): pp. 737–748.

[30] Subramanian, Sai Ganesh, Eng, Mathew, Krishnamurthy, Vinayak R and Akleman, Ergun. "Delaunay Lofts: A biologically inspired approach for modeling space filling modular structures." *Computers & Graphics* Vol. 82 (2019): pp. 73–83.

[31] Schumacher, Christian, Marschner, Steve, Gross, Markus and Thomaszewski, Bernhard. "Mechanical characterization of structured sheet materials." *ACM Transactions on Graphics (TOG)* Vol. 37 No. 4 (2018): pp. 1–15.

[32] Martínez, Jonàs, Skouras, Mélina, Schumacher, Christian, Hornus, Samuel, Lefebvre, Sylvain and Thomaszewski, Bernhard. "Star-shaped metrics for mechanical metamaterial design." *ACM Transactions on Graphics (TOG)* Vol. 38 No. 4 (2019): pp. 1–13.

[33] Sun, Fangfang, Lai, Changliang and Fan, Hualin. "In-plane compression behavior and energy absorption of hierarchical triangular lattice structures." *Materials & Design* Vol. 100 (2016): pp. 280–290.

[34] Zhikharev, LA. "A Sierpiński triangle geometric algorithm for generating stronger structures." *Journal of Physics: Conference Series*, Vol. 1901. 1: p. 012066. 2021. IOP Publishing.

[35] Cheng, Qian, Yin, Jianfei, Wen, Jihong and Yu, Dianlong. "Mechanical properties of 3D-printed hierarchical structures based on Sierpinski triangles." *International Journal of Mechanical Sciences* Vol. 247 (2023): p. 108172. DOI https://doi.org/10.1016/j.ijmecsci.2023.108172. URL https://www.sciencedirect.com/science/article/pii/S0020740323000747.

[36] Liang, Hongyu, Hao, Wenqian, Xue, Guilian, Liu, Baichuan, Pu, Yongfeng and Ma, Fangwu. "Parametric design strategy of a novel self-similar hierarchical honeycomb for multi-stage energy absorption demand." *International Journal of Mechanical Sciences* Vol. 217 (2022): p. 107029. DOI https://doi.org/10.1016/j.ijmecsci.2021.107029. URL https://www.sciencedirect.com/science/article/pii/S0020740321007050.

[37] Tao, Yong, Li, Weiguo, Wei, Kai, Duan, Shenyu, Wen, Weibin, Chen, Liming, Pei, Yongmao and Fang, Daining. "Mechanical properties and energy absorption of 3D printed square hierarchical honeycombs under in-plane axial compression." *Composites Part B: Engineering* Vol. 176 (2019): p. 107219. DOI https://doi.org/10.1016/j.compositesb.2019.107219. URL https://www.sciencedirect.com/science/article/pii/S1359836819302641.

[38] Chen, Yanyu, Li, Tiantian, Jia, Zian, Scarpa, Fabrizio, Yao, Chun-Wei and Wang, Lifeng. "3D printed hierarchical honeycombs with shape integrity under large compressive deformations." *Materials Design* Vol. 137 (2018): pp. 226–234. DOI https://doi.org/10.1016/j.matdes.2017.10.028. URL https://www.sciencedirect.com/science/article/pii/S0264127517309486.

[39] Jeffrey, H.Joel. "Chaos game visualization of sequences." *Computers Graphics* Vol. 16 No. 1 (1992): pp. 25–33. DOI https://doi.org/10.1016/0097-8493(92)90067-6. URL https://www.sciencedirect.com/science/article/pii/0097849392900676.

[40] Talatahari, Siamak and Azizi, Mahdi. "Optimization of constrained mathematical and engineering design problems using chaos game optimization." *Computers Industrial Engineering* Vol. 145 (2020): p. 106560. DOI https://doi.org/10.1016/j.cie.2020.106560. URL https://www.sciencedirect.com/science/article/pii/S0360835220302941.

[41] Ramadan, Abdelhady, Kamel, Salah, Hussein, Mahmoud M. and Hassan, Mohamed H. "A New Application of Chaos Game Optimization Algorithm for Parameters Extraction of Three Diode Photovoltaic Model." *IEEE Access* Vol. 9 (2021): pp. 51582–51594. DOI 10.1109/ACCESS.2021.3069939.

[42] Barnsley, Michael F. *Fractals everywhere*. Academic press (2014).

[43] Ewins, David J. *Modal testing: theory, practice and application*. John Wiley & Sons (2009).

[44] Avitabile, Peter. "Experimental modal analysis." *Sound and vibration* Vol. 35 No. 1 (2001): pp. 20–31.