

**ALGORITHM DEVELOPMENT FOR PATTERN GENERATION INSPIRED FROM
TREE GROWTH**

by

Mohammad Hosseinpour Miyandasteh

M.Sc., University of Tehran, 2018
B.Sc, Sharif university of technology, 2015

PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR THE DEGREE OF
MASTER
OF
NATURAL RESOURCES AND ENVIRONMENTAL STUDIES

UNIVERSITY OF NORTHERN BRITISH COLUMBIA

February 2020

© Mohammad Hosseinpour Miyandasteh, 2020

Abstract

The nature was the source of inspiration in many designs and products. Learning the algorithms from nature and incorporating them in the product design can be another level of inspiration. Algorithms for generating the pattern of tree-growth and venation in a leaf are nature-based algorithms for various uses in design and modeling.

Development of the plant-pattern generators was initially intolerant of target shape until introduction of the space colonization algorithm (SCA). The SCA had a target area filled with points. The nodes, which create the final shape, start growing the pattern from an initial point to cover the target area. The points have an attraction field, which determines the direction of pattern.

This project consists of two phases. The first phase improved some the features in the SCA including: i) the capability of starting the branching pattern from outside the target area, ii) tolerating symmetric distribution of points in target area, and iii) not canceling the effect of points from each other. The second phase used a branching equation assigned thickness to the members. The parameter in the equation was optimized to achieve the minimum variance of stress/capacity ratio among members.

Table of Contents

Abstract	ii
List of figures	v
List of definitions	viii
Chapter 1: Introduction	1
1.1. Motivation	1
1.2. Nature-inspired designs	2
1.3. Goal	4
Chapter 2: Literature review	5
2.1. Evolution of methods and algorithms.....	5
2.2. Space colonization algorithm	5
2.3. Applications of the space colonization algorithm	8
2.4. Literature summary	11
Chapter 3: Phase one: Improved space colonization algorithm	12
3.1. Objectives	12
3.2. Scope and limitations	12
3.3. Overview of steps in algorithm	14
3.4. Detail explanation of terms and steps in the improved algorithm:.....	16
3.4.1. Fields: Parameters and calculation	16
3.4.2. Node effect scale (NES):	19
3.4.3. Calculating Δr :	19
3.4.4. Location of next node:.....	20
Chapter 4: Phase two: Modifying branching equations from an structural perspective	21
4.1. Objective.....	21
4.2. Explanation of steps in second phase	22
4.2.1. Force and tension analysis:.....	23
Chapter 5: Results	25
5.1. Description of change in parameters:	25
5.1.1. Effect of change in oval factor:	25
5.1.2. Effect of change in node-effective-angle	26
5.1.3. Effect of change in omitting distance:.....	27

5.1.4. Generated patterns	27
Chapter 6: Conclusions	35
6.1. Objective accomplishment	35
6.2. Applications of the algorithm	36
6.3. Further works and suggestions	36
Bibliography.....	38
Appendix A: Method used for structural calculations	41
Appendix B: Other results.....	43

List of figures

Figure 1 Fractal pattern in fern. (Public source https://www.pikrepo.com)	1
Figure 2 Pattern visible in [1] human capillary, [2] tree branches, [3] coral reef, [4] bronchi 1	
Figure 3 kingfisher in the left and fast train on the right which inspired the shape from the bird	2
Figure 4 Open air museum inspired from armadillo and made with bent oak laths (Douglass, 2015)	3
Figure 5 Beijing's 'bird nest' sport stadium. The structure and form was inspired from a bird's nest (Gonzales, 2012).....	3
Figure 6 Taipei 101, located in the Xinyi District in Taiwan. Inspired structurally and architecturally from bamboo tree (Gonzales, 2012).....	4
Figure 7 the red points define the parameter and the black points are nodes which generate the pattern step by step. Points attraction determines the new node's origin and its direction and in (Runions, 2005).....	6
Figure 8 (Runions 2007) (p. 3) The space colonization algorithm. a) 4 points exist b) points exert attraction on the nodes, c) amounts of attraction is calculated, d, e)the ones with highest attraction will branch a new node, f, g) points in the deleting radius of new nodes will be deleted, h) procedure continues with remaining points and new nodes.....	7
Figure 9 (Runions, 2007) (p. 6) A tree generated using continuously added attraction points on the right and a venation system on the left which started from the bottom point and grow to the final shape.	7
Figure 10 (Argudo et al., 2015) (p. 65), rendering procedure using SCA.	9
Figure 11 (Lohan et al., 2017) (p.1075), comparison of different heat conducting methods. 10	
Figure 12 products generated by the algorithm and 3D printed (Rosenkrantz, 2007).....	11
Figure 13 flowchart of pattern generation algorithm	15
Figure 14 flowchart of structure analysis.....	22

Figure 15 An oval shape and a sphere. the change in the oval factors can scale the request and provision regarding the direction which later will cause more alignment of nodes in the direction with higher value.....	26
Figure 16 the name ‘bridge’ is chosen for this shape because of a flat surface that looks similar to a half span of a bridge being supported from bottom. Parameters are 25 degree and 0.05 on horizontal and 5 on vertical oval factor.....	28
Figure 17 parameters are 45 degree for node effective angle and oval factor of 1 in all three directions	29
Figure 18 bridge with 45 degree and 0.05 on horizontal and 5 on vertical oval factor	30
Figure 19 comparison between the 25 and 45 degree approach. In 25 degree the further parts of target area was not covered by nodes in more vertical branches and thus new branches arise with more horizontal inclination.....	31
Figure 20 change of pattern due to change in oval factor Left: tree with oval factor of [1,1,0.1] and 45 degree in Node Effective Angle. Reduction in the vertical factor. Middle: tree with oval factor of [0.1, 0.1, 1] and 45 degree in Node Effective Angle. Branches have more vertical inclination. Right: tree with oval factor of [1,1,1] and 45 degree in Node Effective Angle. Due to equal factors branches are scattered in all directions.....	32
Figure 21 Left: curved wall with omit distance of 2.4 and equal oval factor. Node effective angle is 25 degree. Right: curved wall with omit distance of 2 and equal oval factor node effective angle is 45 degree. The target area portion of a cylindrical shell, which is shown in green color.....	33
Figure 22 Left: curved wall with omit distance of 8 and equal oval factor. Node effective angle is 45 degree. Right: curved wall with omit distance of 10 and equal oval factor. Node effective angle is 25 degree.....	34
Figure 23 structure with 4 different patterns to cover walls and roof.....	43
Figure 24 shell of a tree shape with large member length	44
Figure 25 conical shape.....	45

List of definitions

Target area	ii
R-points	ii
Nodes	ii
Fields	iii

Chapter 1: Introduction

1.1. Motivation

Patterns in nature are regular forms that are visible and found in natural object like trees, venations, bronchi, capillary, and coral reefs (ECstep, 2011). Fractal patterns can be seen in ferns (Figure 1). Human bronchi, capillary system, branches of a tree and some coral reefs exhibit similar patterns. Some of these patterns can be explained mathematically and chemically (Figure 2).



Figure 1 Fractal pattern in fern. (Public source <https://www.pikrepo.com>)

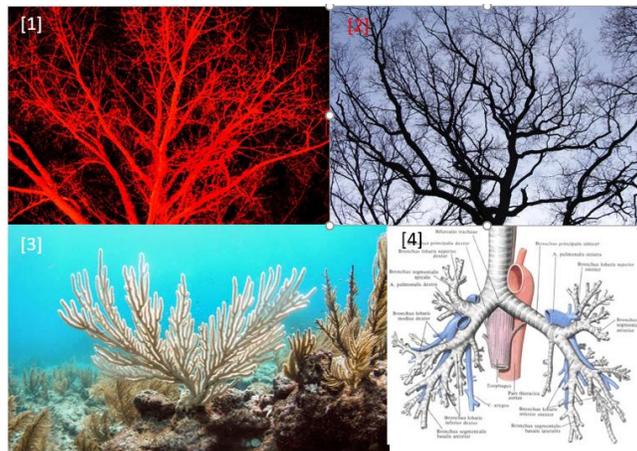


Figure 2 Pattern visible in [1]¹ human capillary, [2]² tree branches, [3]³ coral reef, [4]⁴ bronchi

Inspiring and mimicking the shapes and patterns in nature gave birth to new technologies and products in various areas. Japanese designer solved the problem of huge sonic boom

¹ <https://slideplayer.com/slide/8198381/>

² https://commons.wikimedia.org/wiki/File:Branches_of_a_tree.JPG

³ <https://www.the-scientist.com/daily-news/bleached-corals-sickest-scientists-have-ever-seen-33328>

⁴ http://en.shram.kiev.ua/health/anatomy/page_1739.shtml

coming from fast trains by adopting making it in the shape of a kingfisher that bears great impact diving to water (Weinzettl, 2017) (Figure 3). Termite house inspired architects to build energy sufficient buildings¹. NASA used sharp skin pattern to make coatings that reduce drag and deters microorganisms (such as algae) attaching to the surface².



Figure 3 kingfisher in the left and fast train on the right which inspired the shape from the bird³

Nature has been creating different methods and models of life in billions of years. Living systems have been developing and generating aspects of life like patterns, materials, algorithms, structures, and behaviors. On the other hand, throughout the time human civilizations have also been generating and designing new products and methods. The human innovations have been influenced by nature in different aspects including shape design, algorithm adoption and system design. Biomimicry have been recently introduced in which innovations and design are directly inspired from natural patterns.

1.2. Nature-inspired designs

Designing has been an inseparable part of human civilizations, from simple tools and huts to big stone structures in Roman Empire and early mechanical systems, and then now modern equipment, and artistic buildings. Some designs had elements inspired from nature in them. Including Downland Gridshell Building in Singleton, Chichester, UK, which was described as an armadillo (Douglass, 2015) shown in Figure 4. Another example of nature-inspired

¹ <https://www.youtube.com/watch?v=oPWiiiOe7GM>

² <https://www.sciencefocus.com/future-technology/biomimetic-design-10-examples-of-nature-inspiring-technology/>

³ <https://www.xing.com/news/insiders/articles/biomimicry-learning-from-nature-626416>

structures can be seen in Beijing National Stadium inspired from a bird's nest (Gonzales, 2012) . Some of which are architectural elements and some are structural like the Taipei 101 is located in the Xinyi District in Taiwan's capital city, which got their structural inspiration from bamboo trees to increase lateral resistance (Figure 6).

Learning from the pattern and trying to develop algorithm based on them can be another form of inspiration from nature, which genetic algorithm and bee algorithm are among them¹. In this project the pattern seen in the trees were the source of inspiration to develop an algorithm that can generate similar patterns.



Figure 4 Open air museum inspired from armadillo and made with bent oak laths (Douglass, 2015)



Figure 5 Beijing's 'bird nest' sport stadium. The structure and form was inspired from a bird's nest (Gonzales, 2012)

¹ https://www.researchgate.net/post/What_are_the_latest_Nature_Inspired_Algorithms



Figure 6 Taipei 101, located in the Xinyi District in Taiwan. Inspired structurally and architecturally from bamboo tree (Gonzales, 2012)

1.3. Goal

The purpose of this project is to learn from the nature by trying to mimic some design patterns and develop an algorithm that automatically generates nature-based patterns.

This project works on two aspects, which are the improvement of the current algorithms and an optimization of branching thickness equation. The general steps to are: i) Review existing algorithms and find the most current algorithm; ii) Develop a new or improve the algorithm found in previous step, capable of simulating similar patterns; and iii) Assess the branching equation for member thickness estimation in the algorithm by measuring the stress integrity throughout pattern under one case of loading. The project has two phases and the details of the steps comes in the objectives section of the corresponding phase.

Chapter 2: Literature review

2.1. Evolution of methods and algorithms

Generating patterns like a tree has been going on since 1971 when Honda, proposed a recursive pattern for plant growth based on the geometric characteristics including distance ratio between branches and branching angles (Honda, 1971). His approach was using geometrical equations, which generated the shape regardless of boundary.

Masaki and Tosiyasu produced the algorithm to generate the pattern in which design parameters were the angle between the new branch and the main branch and the space direction angle of new branch relative to the previous branch projected in the plane perpendicular to the main branch (Masaki & Tosiyasu , 1984). Later by modifying the angle parameters based on the location of the branches, Prusinkiewicz et al. developed more changeable and diverse patterns. They integrated the three structural patterns, “posture (manifested in curved stems and elongated leaves), gradual variation of features, and the progression of the drawing process from overall silhouette to local details” (Przemyslaw , Mündermann, Karwowski, & Lane, 2001)In their algorithms the shape (curvature), location, and length of plants were designable.

Sachs described some other factors affecting the shape of a tree or plant canopy, including success ratio of buds becoming branches and further availability of light. These descriptions showed differences between actual growth procedure and simulating patterns (Sachs & Novoplansky, 1995). Later works involved the statistical analysis of success-fail ratio of buds (De Reffye, Edelin, & Françon, 1988), sun light absorption competition (Takenaka, 1994). The effect of gravity was then incorporated by (Jirasek & Prusinkiewicz, 2000).

2.2. Space colonization algorithm:

These algorithms did not intake and tolerate the concept of a target area and were generating patterns based on fixed shape equations that enforced the angle and length relations between members. Runions incorporated equations which did not enforce the shape and instead directed the pattern toward where it was more ‘asked for’ by the points defining the inside and boundary of final pattern. Runions discussed that previous algorithms generally used the

same recursive process and modifications are not causing significant changes and he proposed a new algorithm named ‘space colonization algorithm’ (Runions, 2007).

This algorithm is based on the repetitive growth of the node structure and spanning the assigned area. The assigned area is a 2D or 3D surface that filled with points and the nodes will span it by their gradual growth (Figure 7-). The user assigns the first node location and the rest nodes are placed after that. The algorithm was first used to simulate leaf venation.

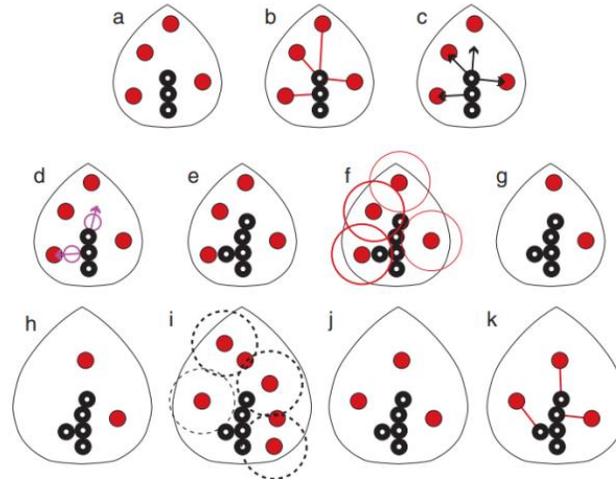


Figure 7 the red points define the parameter and the black points are nodes which generate the pattern step by step. Points attraction determines the new node's origin and its direction and in (Runions, 2005)

The outline shape of a leaf is the perimeter and the points are randomly distributed as auxin source, which should be transferred by veins. The sources direct the veins and as a vein node approached the source with a distance less than a certain amount called ‘kill distance’ the source will be deleted. The placement of new nodes follows steps:

The points (referred to as source in original paper) influence the nodes and the set of points that effect a point $v \in V$ (V in the set of nodes) is called $S(v)$. If $S(v)$ is not empty, then a new point (v') will be assigned. The distance from the initial node is D which is a predefined parameter and the direction of the next node will be calculated using eq.1.

$$v' = v + D \frac{\vec{n}}{\|\vec{n}\|}, \text{ where } \vec{n} = \sum_{s \in S(v)} \frac{s - v}{\|s - v\|} \quad \text{eq.1}$$

After the placement of each node, all the points within a certain radius called ‘kill distance’ will be deleted and the process will continue until no more point remain. The systematic process can be seen in Figure 8 and Figure 7 (Runions, 2007).

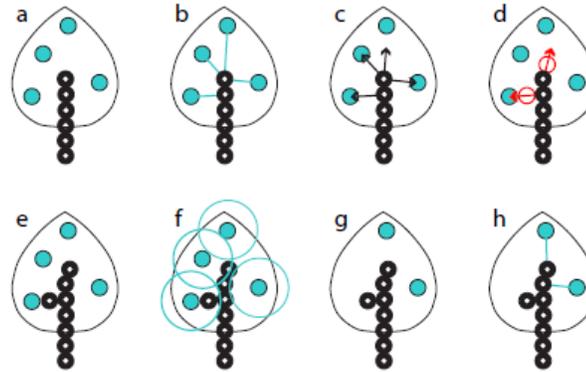


Figure 8 (Runions 2007) (p. 3) The space colonization algorithm. a) 4 points exist b) points exert attraction on the nodes, c) amounts of attraction is calculated, d, e) the ones with highest attraction will branch a new node, f, g) points in the deleting radius of new nodes will be deleted, h) procedure continues with remaining points and new nodes

Some of the generated patterns are presented in Figure 9.

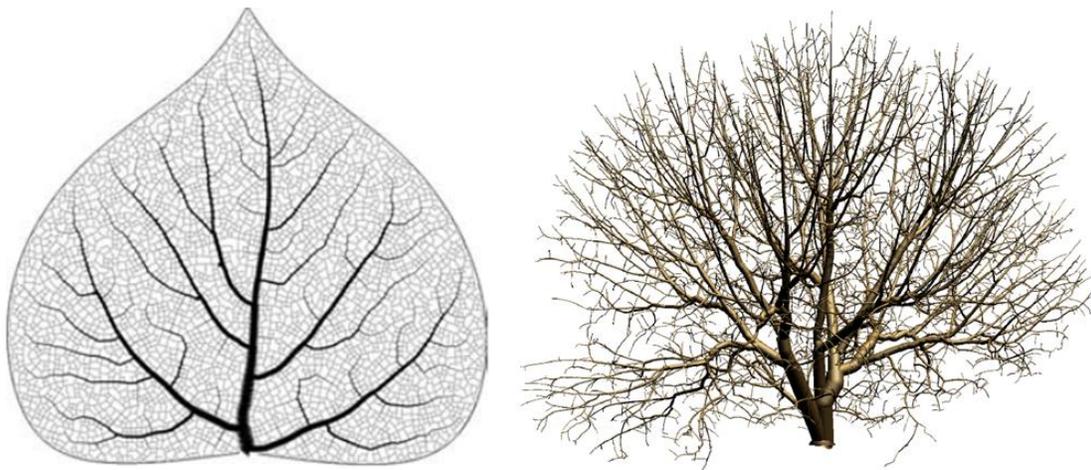


Figure 9 (Runions, 2007) (p. 6) A tree generated using continuously added attraction points on the right and a venation system on the left which started from the bottom point and grow to the final shape.

2.3. Applications of the space colonization algorithm

Fernandes et al. used the SCA (space colonization algorithm) for procedural road layout generation, they used the 2D version of algorithm and modified it to become suitable for urban road layouts by introducing additional features. In the original algorithm turns were allowed to have any degree but in the fixed angle mode only certain degrees (30, 45 or 90) were allowed to happen which is more compatible with city design. In the plants there are no middle branches that connect two other branches and create a loop, so the modified version has the ability to merge the close nodes and create loops. Flow field was another feature added to the algorithm that could direct the patterns in certain directions. Information map reflects the high and low density areas, causing more concentration of major and minor roads (Fernandes & Fernandes, 2018).

Lima Bicho et al. had used the 2D SCA for crowd movement patterns. They added three features to the original algorithm. The main difference goes with the introduction of time dependency. SCA did not account for time variations in the system but the nature of crowd movement required the time variation and speed consideration, so the new algorithm includes the speed analysis and time variations. In the SCA, the points were deleted after nodes had reached them but in the crowd simulation points due to temporary condition of space occupation, points are not deleted and patterns change with time. The SCA was didn't differentiate between branches to reach a point and the provision itself was the target while in the new algorithm only certain people should get to certain locations (Lima Bicho, Araújo Rodrigues, & Raupp Musse, 2012).

Silveria et al. Combined the SCA with D* and generated a new algorithm called Space D*, used for path finding for robots in an unknown environment with multiple robots. The optimized factors were possible speed for robot without collision and time requires for spanning the area (Silveira, Maffei, Drews, & Bicho, 2012).

Li had used the SCA for simulation of rooting system affected by soil moisture heterogeneity. Roots were assumed to grow toward the moisture locations with the pattern described by SCA (Li & Gao, 2015).

Argudo et al. used the SCA to recreate and render the blurred plant photographs. They had the perimeter and an imprecise topography of tree which was fed to the SCA as the initial 3D target area and the SCA developed the tree shape (Argudo, Chica, & Andujar, 2016). Figure 10 shows the application of algorithm in generating tree shapes which is later used in image analysis.

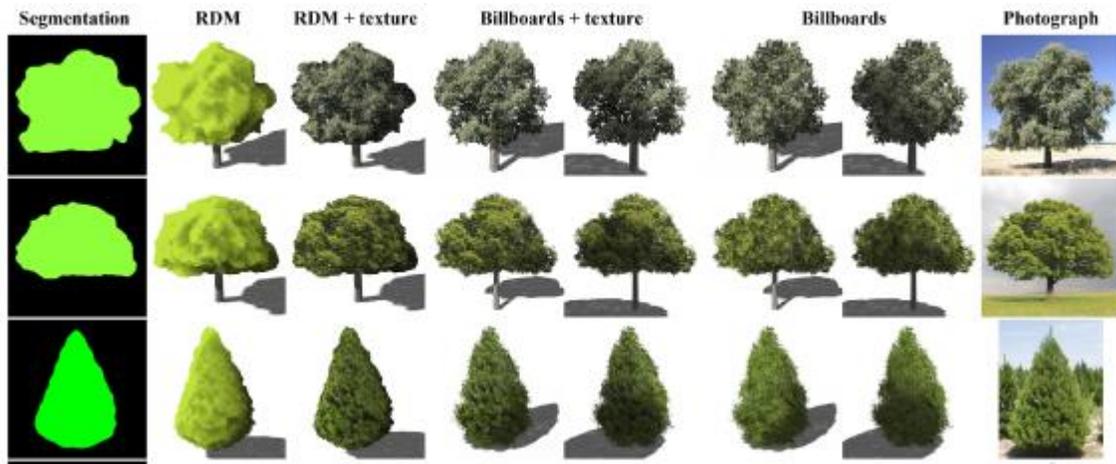


Figure 10 (Argudo et al., 2015) (p. 65), rendering procedure using SCA.

Pirk et al. developed an algorithm for generating tree shapes, which needed the ‘skeleton’ of tree as initial structure and would have done branching on it. His algorithm handled the existence of obstacles and did not need growth perimeter. He also included the shading and pruning in the algorithm (Pirk, Stava, Krat, Abd al Masih, & Neubert, 2012).

Lohan et al. used the two dimensional SCA and genetic algorithm to design heat conducting system (Figure 11). The results were tested with standard finite element software. Generative algorithms have basic parameters that will change the design and due to nature of each problem these parameters can have differ. This research had tested the algorithm with different parameters and compared those (Lohan et al., 2017).

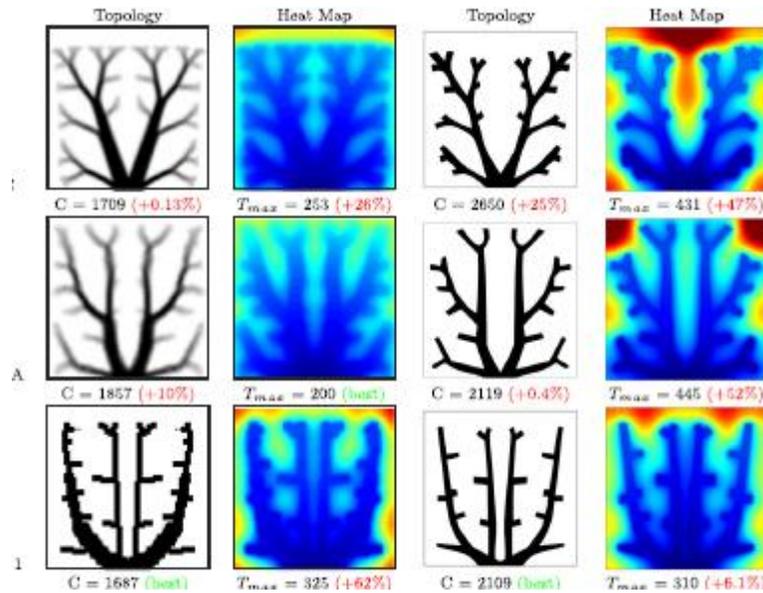


Figure 11 (Lohan et al., 2017) (p.1075), comparison of different heat conducting methods.

Rodrigues et al. Used the SCA for path generating. In the article the process is described as “presented some behaviors to simulate groups of agents, such as group path planning (we called tree paths), pursue behavior, surround behavior, escape behavior, groups formation, and groups alignment. The key innovation is the simple way in which the paths are created, by ‘observing’ and ‘occupying’ free space, which is represented using a set of marker points, which leads to a simple yet computationally effective implementation of the competition for space.” They added the feature for crowd guidance and path assignment to avoid collisions by assigning specified markers to each agent (Rodrigues & Bicho, 2010).

Some other applications of these patterns are seen in products by Nervous company (Rosenkrantz, 2007) using the Runion’s algorithm. Some of the products can be seen in the Figure 12.



Figure 12 products generated by the algorithm and 3D printed (Rosenkrantz, 2007)

2.4. Literature summary

Reviewing the literature illustrates that the space colonization is the latest and most referred algorithm among tree-shaped pattern-generating algorithms. Therefore, this project focuses on improvement of this certain algorithm. Space colonization algorithm was developed from predefined patterns to target dependent patterns. Various applications are possible beyond the initial intentions, including pathfinding in traffics, designing venations systems for heating and cooling, and product designs. As shown in Figure 4 and Figure 5, the algorithm has the potential for implementation in architectural design, which is one of major motivations of using this algorithm in design.

There are various improvement opportunities with the current algorithm. This project focus on the two features. First feature is the ability to start from outside target area while generating multiple branches before reaching to the target area. Second feature is being able to handle symmetric distribution of point. In current algorithm if the points arrange in a symmetrical formation, then the vectors in eq.1 cancel each other and there would not be a second node. Solution for these two limitations is with introduction of a vector field assigned to each point ($R \rightarrow R^n$) detailed explanation of the field is in 3.4.1.

Chapter 3: Phase one: Improved space colonization algorithm

3.1. Objectives

The objective of this section is to develop and improve an algorithm that has following features:

1. Branching from outside target area
2. Not canceling the effects of points on each other
3. Tolerating the symmetric distributions of points

3.2. Scope and limitations

The project will have two phases. The first phase will develop the algorithm to generate a shape in the desired target area (defined in p.**Error! Bookmark not defined.**). The second phase will assign a thickness to the members by using a geometrical equation from the relation between the branches of a tree at the joints, eq.16 (Runions, 2007). After calculating the stresses in the sections, the variance of the ratio of maximum stress to the capacity of member will be an indicator for the integrity in the structure.

From an engineering point of view, an ideal structure should have many characteristics and assessing all of them is out of this project's scope. However, one of the main structural characteristics is the integrity in distribution of forces and stresses, so that every section and member goes under a relatively similar rate of force to strength. After the branching equation is applied to the system of nodes and thickness was assigned to the members, system of nodes and according thicknesses would be analyzed and the stress would be calculated in the members, and the variance of stress is the indicator for optimizing the parameter in the branching equation.

As the main goal of this project is to take the first steps in incorporating natural algorithms in design, it would be expectable that the algorithm have some limitations and does not cover all aspects. Some of the aspects that are not covered are i) adjustment of member length which is set to be at an applicable level; ii) having only one starting point and not starting the pattern from several points. The second phase also includes limitations such as i) the

loading, set to be in one direction and applied to all end nodes; and ii) considering all connections between members of pattern solid and not joint-connection.

3.3. Overview of steps in algorithm

The general steps in the algorithm are:

1. Definition of the target area by user and the starting point
2. Calculating the request from the points on all the nodes
3. Calculating all provisions from the nodes on the nodes
4. Finding the overall field
5. Assigning a new node
6. Deleting covered points if available
7. Repeating steps 2 to 6 until no points remain.

Step 1: The system starts with a target area that consists from a set of r-points (definitions of the terms are mentioned in page **Error! Bookmark not defined.**), and some starting nodes. The purpose would be to have a system of nodes that starts from a starting node and grows toward the target area. After reaching the target area, it expands inside the target area up to a stage that the request to create new nodes from all r-points in the target would be ‘answered’ as the nodes will reach that area.

Step 2 to 4: R-points and nodes both will create fields. R-points will create a request field and nodes will satisfy that request by a provision field. The mathematic definition of the fields is mentioned in eq.4 and eq.3. Both fields from the points and the nodes will be calculated on the nodes and then the two fields will be subtracted from each other. Only the positive amounts will be kept and negative values (which mean the provision is more than request) will be neglected and replaced with zero. The result from the subtraction will be the overall field used for assigning new nodes.

Step 5: The pattern growth starts with the starting node. After calculation of the overall field on the nodes, the highest request in the distribution among all nodes will determine the direction that the next node will be assigned. The distance of the new node from the previous node will be calculated using eq.14.

Step 6: The new node might be the neighboring points that are closer than an omitting distance. Any point within the omitting distance of the new node will be removed from the set of points.

Step 7: After omitting the neighboring points, the iterative procedure begins. The procedure continues until no more request remains or all the points are omitted.

Each step is marked by its number with square brackets in the flowchart shown in Figure 13.

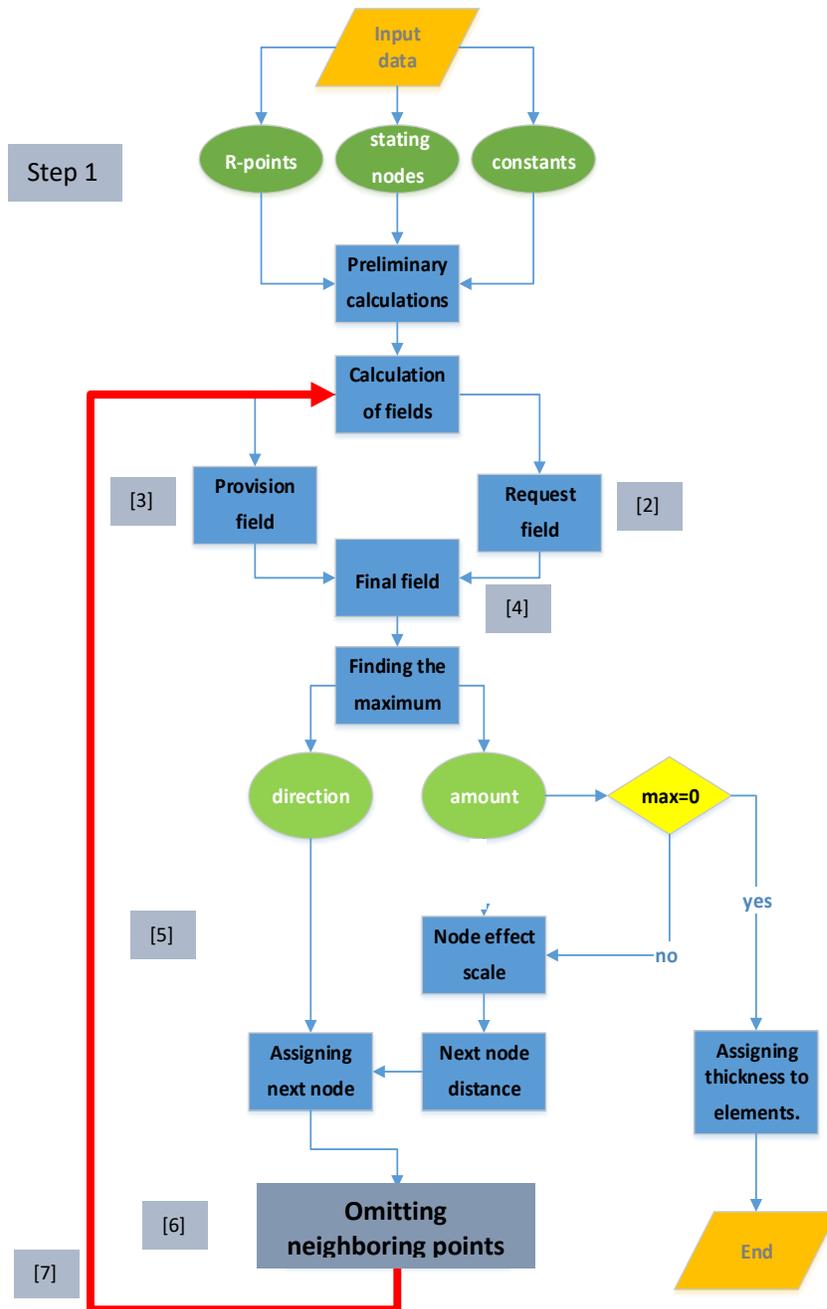


Figure 13 flowchart of pattern generation algorithm

3.4. Detail explanation of terms and steps in the improved algorithm:

3.4.1. Fields: Parameters and calculation

The target area is a two or three dimensional surface volume that needs to be covered, it could be in a cone shape representing outside surface of a tree, a flat surface as roof to be supported, a volume like human lung that needs to be covered by bronchus.

The target area will be introduced by a set of location points that span it. The density of r-points will be determined by the operator. The r-points in the target area can be either randomly distributed or have a distribution pattern, which could be an even distribution or have different densities in different areas.

Nodes are locations that will make the final structure of the system. Each node refers to one location that will have a turning or a branch may add to the structure. By attaching line between them the final structure will be formed.

Starting nodes are the locations that our structure begins to grow from. Starting nodes could either be one or more than one node.

Nodes could either be in the target area or between the starting node(s) and the target area. Nodes in the target area are supposed to provide the desired feature to the target area. One example of the features can be a tree branch absorbing light from an area in which the nodes will represent the branches and the feature will be absorbing sunlight or air from the area. Another example is nutrient support from vessels and capillary to cells in a body organ, the organ would be the target area and the capillaries will be shown by nodes and line between nodes.

Each r-point or node will spread information about its existence through a 'field'. The fields from r-points in target area would be called 'Request Field' because they attract the structure toward themselves. The field from nodes would be called 'Provision Field' since the nodes provide the need that the r-points requested.

In this distribution result on a location in a three dimensional space is not a single vector, but is a distribution over a sphere with radius of one unit. An amount would be assigned to each r-point on the sphere which will indicate the amount of field in that direction.

R-points' location is a $N_p \times 3$ matrix containing the locations of r-points in which the N_p represents number of the r-points. After the nodes reach the target area and start erasing the nearby points, the N_p will be reduced accordingly.

Nodes' location is a $N_n \times 3$ matrix containing the locations of the nodes in which N_n represents number of the nodes. Number of nodes starts with the initial node and increases during the process.

The field from one r-point on one node shown in eq.2:

$$F(n, p) = \frac{1}{|r_n - r_p|^{c1}} * \emptyset(\overrightarrow{(r_n - r_p)}) \quad \text{eq.2}$$

In addition, the total field from all the r-points effecting on one single node is shown in eq.3:

$$Frqs_n = \sum_i \frac{1}{|r_n - r_{p_i}|^{c1}} * \emptyset(\overrightarrow{(r_n - r_{p_i})}) \quad \text{eq.3}$$

$Frqs_n$, total field from all r-points on node number n

r_n , the location of the node.

r_{p_i} , the location of r-point number.

In addition, total field from all the nodes on one single node is shown in eq.4:

$$Fprv_n = \sum_i^{\hat{k}} \frac{1}{|r_{n_i} - r_{n_k}|^{c1}} * NES(i, k) * \emptyset(\overrightarrow{(r_{n_i} - r_{n_k})}) \quad \text{eq.4}$$

$Fprv_n$, the total field from all the nodes on the node number k

$\emptyset(\vec{r})$, the function for the field distribution is defined in eq.5, eq.6 and eq.7:

$$D: \{(x, y, z) | \frac{x^2}{a} + \frac{y^2}{b} + \frac{z^2}{c} = 1\} \quad \text{eq.5}$$

$$\phi(\vec{r}) = f : D \rightarrow R \quad \text{eq.6}$$

$$f(\vec{\rho}) = \begin{cases} \left(\frac{\vec{\rho} \cdot \vec{r}}{|\vec{r}|} \right)^{c2} & \theta < \theta_0 \\ 0 & \theta > \theta_0 \end{cases} \quad \text{eq.7}$$

θ_0 Is the effective angle range the fields.

Since the algorithm performs numerical calculations, having infinite numbers for the distribution function is not possible. The distribution function requires considering all the directions with the same level of importance. Therefore, a set of evenly distributed points on the sphere surface will be used for performing the field calculations. The domain is shown in eq.8. In some cases, the effects in one particular direction is more important than the other directions. In this case the values of a, b, c in eq.8 will be adjusted by assigning a higher value to the more important direction. Due to similarity to the oval equation, the effect of this change is called oval factor and described in 5.1.1.

$$D = \{r_i | \frac{x_{r_i}^2}{a} + \frac{y_{r_i}^2}{b} + \frac{z_{r_i}^2}{c} = 1, i = 1:n_s\}, n_s = \text{number of points on sphere} \quad \text{eq.8}$$

Researching on the effects of changes in the $c1$ is not in the scope of this research. In addition, choosing a $c1$ for the project requires some kind of preference. Therefore, in this project $c1 = 2$ will be chosen since the eq.3 and eq.4 have the same formation to the equation for the electric field from a group of charged particles.

$c2$ Will determine the sharpness of field spreading over the sphere in a way that higher amount of $c2$ will cause sharper distribution. Based on the scope of this project, it is not intend to analyze the effect and usages of various amounts of $c2$. Other values for the constants can be subject of further studies same as optimizing the $c1$ and $c2$.

In order to find the maximum field, the two fields from points and nodes will be subtracted. The negative amount on a direction means that the provision is more than the request. The negative number is not required so only the positive amounts will be kept, as shown in eq.9.

$$FF_n = \frac{(Freq_n - Fprv_n) + |Frqs_n - Fprv_n|}{2} \quad \text{eq.9}$$

$$Max_{req} = \text{maximun}(FF)$$

$Num_{max} = \text{number of node with maximum request}$

$\overrightarrow{Dir_{max}} = \text{the direction of maximum of in node \# } Num_{max}$

3.4.2. Node effect scale (NES):

Each new node will produce provision in the direction of the emerging node. In some cases, the provision would not be equal to the request on the direction between two adjacent nodes, and an extra node might be assigned to the same location. In order to prevent this, the concept of NES (calculated using eq.13, eq.12, eq.11, and eq.10) will be defined. The NES is a correcting factor that is multiplied to the provision to ensure that after assigning the new node, the provision on the previous node in the same as the request.

$$NES = \max\left(\frac{Max_{req}}{request_{basic}}, NES_{basic}\right) \quad \text{eq.10}$$

$$request_{basic} = \max\left(\frac{1}{|distance_{min}|} * \phi\left(distance_{min} * \overrightarrow{(1,0,0)}\right)\right) \quad \text{eq.11}$$

$$NES_{basic} = rp_{density} * distance_{min}^3 * \frac{4\pi}{3} \quad \text{eq.12}$$

$$rp_{density} = \frac{N_p}{V_p} \quad \text{eq.13}$$

$distance_{min}$ is the minimum distance, that the nodes will have from with each other. The amount for $distance_{min}$ is among the input data.

V_p is the volume of the target area.

3.4.3. Calculating Δr :

The distance between nodes would be calculated as shown in eq.14 and will not be less than the ' $distance_{min}$ '.

$$\Delta r = \begin{cases} distance_{min}, & \frac{Max_{req}}{request_{basic}} \leq 1 \\ distance_{min} * \left(\frac{Max_{req}}{request_{basic}}\right)^{\frac{1}{c1}}, & \frac{Max_{req}}{request_{basic}} > 1 \end{cases}, \quad \text{eq.14}$$

The same procedure was used for developing a 2D algorithm and the equations were used in 2D form.

3.4.4. Location of next node:

The location for assigning the next node will be found using eq.15, in which the node $\#Node_{max}$ is the number of node with maximum final field and Δr is the distance from previous node.

$$New_{location} = location(\#Node_{max}) + \Delta r * \overrightarrow{Dir_{max}} \quad \text{eq.15}$$

Chapter 4: Phase two: Modifying branching equations from an structural perspective

4.1. Objective

The objective of this phase is to find the optimal parameter in the branching equation (eq.16).The optimal parameter will cause minimum variance in the stress/capacity ratio of the pattern members. The following steps shall be taken to achieve the objective.

Step1, Member thickness: Thickness of the thinnest members, which are the end members, is given and using eq.16. Thickness of other members in the pattern will be calculated.

Step 2: Structural analysis: A point load similar in three directions will be applied to the end members and the stress in the members of the pattern will be calculated. The stiffness matrix of the system of nodes and their thicknesses will be assembled (assuming the structure to be in the range of linear behavior) and solved for the external loads. All the forces and moments in the elements will be calculated. These values will be used to find the maximum of stress in each member. The maximum stresses will be divided by the capacity of that member (for simplicity, the material is assumed to be isometric). The variance of the ratio of stress/capacity from all members would be used for optimizing in the next step.

Step3, Iteration: different values of the parameter in the branching equation (eq.16) will be used to find the optimal value that gives the minimum variance for pattern members.

Each step is marked by its number in square brackets in the flowchart shown in Figure 14. Inputs in phase two consist of i) structural data which is the location of members and joints in the pattern from phase one, ii) constants for stress calculations, and iii) the initial parameter for branching equation (eq.16). The next step is the structural calculation to find the forces and stresses. After that would be analysis of optimality through finding the variance of the stress/capacity ratio in the system of nodes. The previous steps will iterate until an optimal value for the branching equation is found.

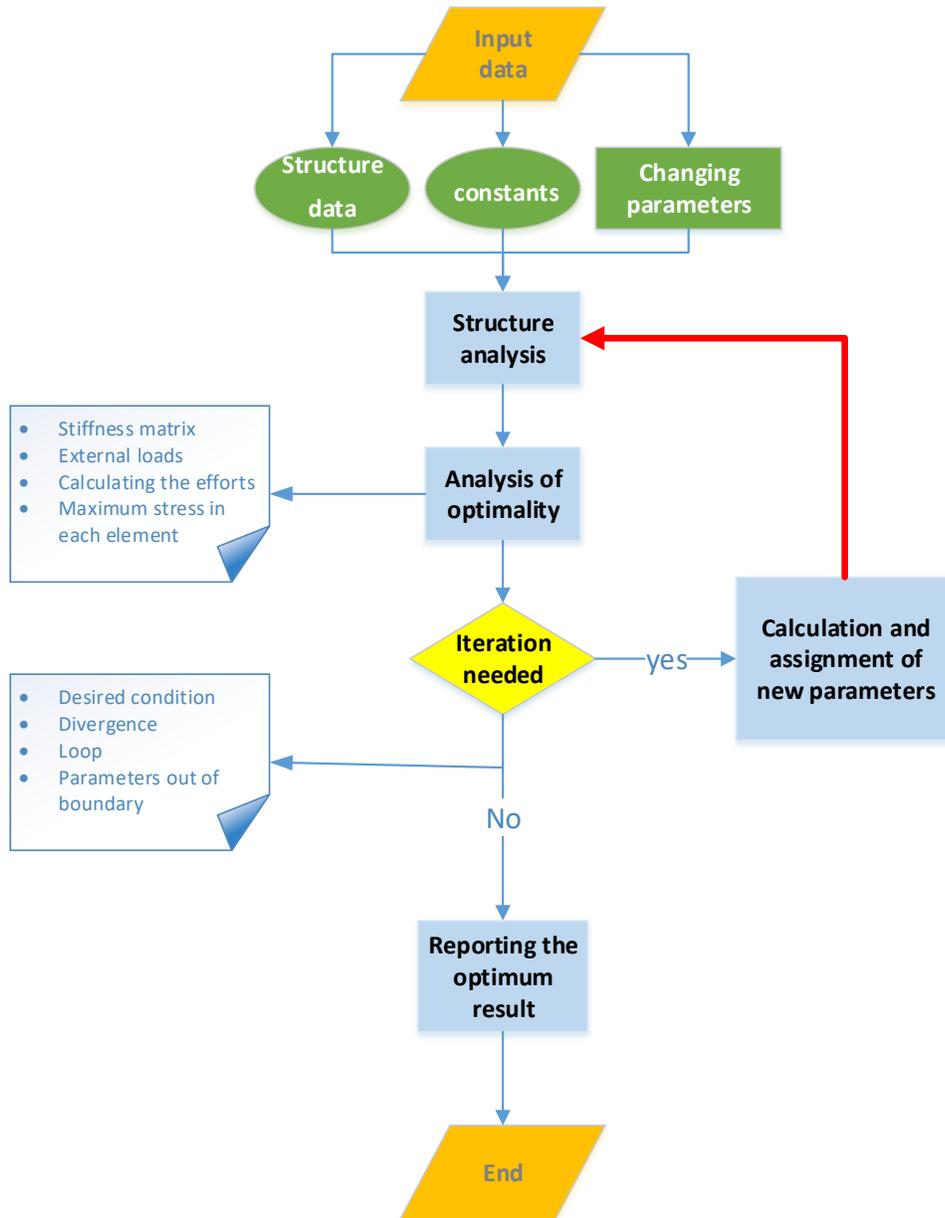


Figure 14 flowchart of structure analysis

4.2. Explanation of steps in second phase

The branching equation is a relation between members in one node. The member toward the start of graph is noted with the u suffix and the members toward the end with the l suffix.

$$t_u^{c1} = \sum_l t_l^{c1}, l \text{ is for all members that goes out from the node} \quad \text{eq.16}$$

Structure data, constants, and changing parameters are taken as inputs.

In this project the structure will be a set of beam elements connected to each other between one starting point and a number of end-points. All connections are assumed to be fully attached and moment transferring. The starting point will have a fixed rest that will support the structure. The end points are assumed to have two conditions regarding their supports. The first case is that all the end points have joint connections with the target area (the definition of target area is mentioned in the previous section). In the second case all joints are free and have no attachments. In both cases all of the external forces are applied to the end points.

Structural input data will consist of the location of joints and initial thickness of the end members. Constants include material properties like elasticity and shear module which is considered the same in all the results. The changing parameter is $c1$ used in branching equation which will determine the thickness of each element.

4.2.1. Force and tension analysis:

The purpose of using structural analysis is only to use its results to optimize the parameter in the branching equation and the result does not have any application in further analysis. The results remain in the program and will be used in other steps of program. Therefore, the result of analysis itself will not subject of discussion, but the analysis procedure will be explained as a method for optimizing.

Conventional structure analysis software may not suitable for doing numerous of iterations and analysis over the same geometry. They are usually designed for designated purposes. In this project geometry and section properties of elements may change every loop so transferring data between two MATLAB and any other software would have either been impossible or so much time consuming that writing a code for analyzing our typical form of structure was more efficient.

Structure analysis has different methods. In this project the Timoshenko beam theory is used to analyze the efforts on the elements. Forces and moments on the two ends are related to deflections in the ends using the local stiffness matrix which is presented in the appendix 1.

By using the local stiffness matrix for each element and deflections at each end, local forces and moments at each end will be found. Maximum stress in each element will be found through following equation:

$$\sigma_{moment} = \frac{Mr}{\frac{1}{4}\pi r^4}, \sigma_{axisl} = \frac{F_{axial}}{\pi r^2} \quad \text{eq.14}$$

$$\sigma_{shear} = \frac{4}{3} * \frac{F_{shear}}{A}, \sigma_{torsion} = \frac{\tau}{\frac{1}{2}\pi r^3} \quad \text{eq.15}$$

$$\sigma_{max} = \sqrt{(\sigma_{moment} + \sigma_{axial})^2 + \sigma_{torsion}^2 + \sigma_{shear}^2} \quad \text{eq.16}$$

One use of maximum stress will be determining whether it exceeds the maximum allowed stress or not. In another use, maximum tensions in all elements will make an array. Standard deviation of the array and the average will be used for make a unitless number that will show how uniform the stress is distributed in structure.

$$SDF = \frac{std(\bar{\sigma}_{max})}{average(\bar{\sigma}_{max})}, \bar{\sigma}_{max} = \text{array of } \sigma_{max} \quad \text{eq.17}$$

Changing parameters will be changed so that minimum amount of *SDF* will be obtained.

Chapter 5: Results

The first phase's objective was improving the existing algorithms by introducing new features. The second phase intended to assign the best parameter to the branching equation (eq.16) from a structural perspective. Based on the objectives, the results will consist of two sections: samples of generated patterns and the optimal branching parameter for each sample. The project had more focus on the first phase (pattern generation) and thus the results will as well.

The results of the first phase are presented through various sample patterns demonstrating the algorithm's ability to handle different formations of points. These patterns include open and closed surfaces like a cone and tree shell, basic full shapes like oval spheres and rectangular cubes, and mixed shapes. Each pattern is produced several times by changing the parameters to illustrate the effect of change in the parameters.

The result of the second phase is the final amount for the optimal value in the branching equation (eq.16), which will be mentioned at the bottom of each pattern. The intention of structural analysis in this project is to gain the variance of the stress/capacity in members thus displaying the stress distribution and force diagrams is not within the scope of the project.

In the following sections, the effect of initial parameters will be explained and the corresponding images will be presented in the next section.

5.1. Description of change in parameters:

5.1.1. Effect of change in oval factor:

This subsection shows the effect of the changing factors in eq.8. The name 'oval factor' comes from the similarity between eq.8 and the equation of an oval shape. The lower/higher amount of a, b, c in eq.8 will decrease/increase the corresponding request in that direction, so the new node will have less/more tendency to be assigned in the direction of the changing factor. Figure 18 and Figure 17 show the difference caused by the oval factor assigned to the requests and provision spatial spreading.

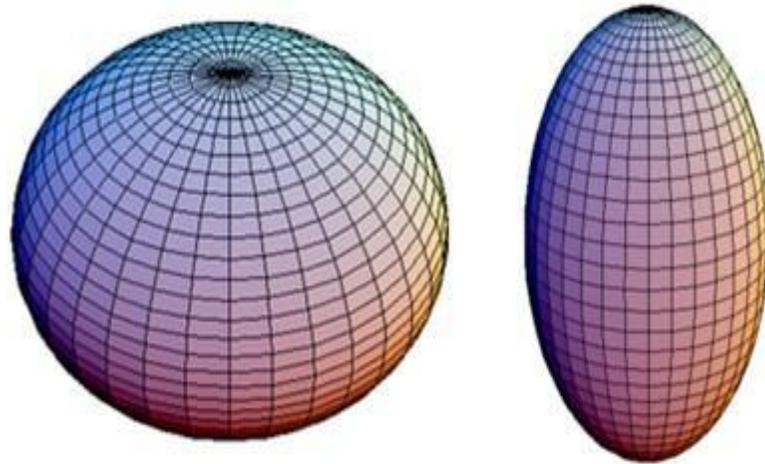


Figure 15 An oval shape and a sphere. the change in the oval factors can scale the request and provision regarding the direction which later will cause more alignment of nodes in the direction with higher value.

In Figure 18 the oval factors in eq.8 were set to be (0.05, 0.05,1), Which caused more attraction in the Z direction than the X and Y directions. In comparison with Figure 17 the distribution of nodes is more vertically directed than horizontally. In Figure 17 all factors were equal to ‘one’ thus, the structure reached the plane of points and from there expanded horizontally. In Figure 18 with 0.05 on the X and Y direction, much less horizontal expansion is observed and the inclined vertical lines are extended more than the case with equal oval factors.

The application of oval factor can be in situations where directional preferences exist. In the design of structures where tension or compression is preferred over bending having the member more inclined to the direction of gravity can give preference to the perpendicular directions.

5.1.2. Effect of change in node-effective-angle

This factor determines the angle range nodes effect on each other. The effect of the angle can be observed in Figure 16 and Figure 18. In the target area, the further points fall in a smaller angle and the nodes with large effective-angle-range cover the request. Thus, the new nodes will not be toward the end of the bridge. In Figure 16 with a smaller effective-angle-range, the nodes will direct more towards the end than the case with larger effective-angle-range.

5.1.3. Effect of change in omitting distance:

This parameter effects the density of nodes, and the distance between them. In the Figure 21 and Figure 22, the patterns were generated with different omitting-distance of 2.4, 2, 8 and, 10. Two factors influence the omitting-distance, the omitting-factor and the member distance. Since the omitting-distance is the multiplication of the two parameters, any change in either of them can cause changes in the density of nodes. In Figure 22-left and Figure 21-right, omitting-factor is the same while in Figure 21-right, the member length inside the boundary (minimum member length), is half the member length in Figure 22-left.

5.1.4. Generated patterns

In this section generated pattern will be demonstrated with explanation for each pattern.

The patterns are proof that the algorithm can generate the required results.

In the Figure 16 to Figure 18, the target is a rectangular cube shown in green and the starting point is below the area at middle of the side. The pattern starts from the starting point and grows toward the target area. The generated pattern is shown in brown color.

The difference between the patterns in the Figure 16 to Figure 18 comes from different the initial parameters, the oval factor and the node effective angle. When nodes are close to the starting point, there is one branch in Figure 17 and Figure 16, which have the same angle of 45 degree. The reason is that as the target area is far from the nodes, it would fit in a smaller angle range from the node's point of view. As the nodes get close to the target area the angle spanning the target area would increase, and the nodes will emerge from the node in more than one direction. Emerging two branches happens earlier in Figure 16 since the spanning angle gets larger than the node effective angle (which is 25 degree) sooner.

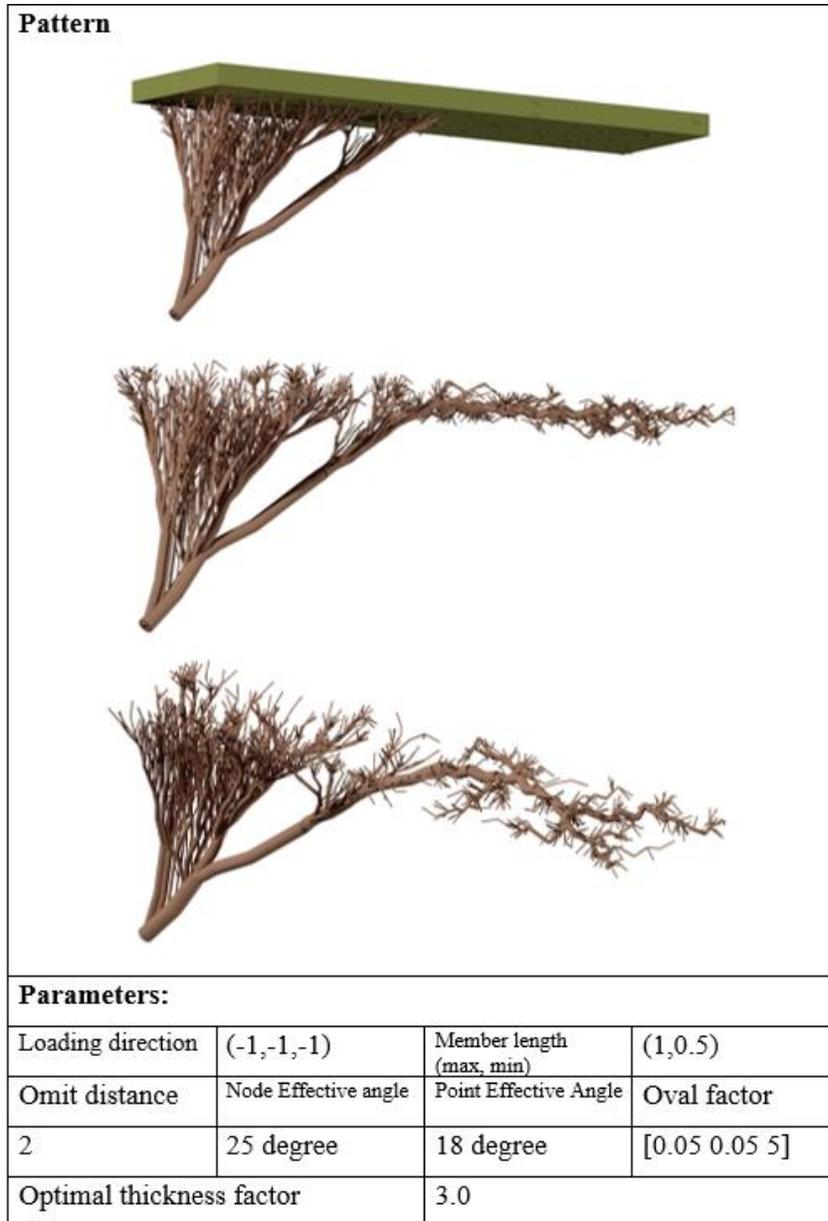


Figure 16 the name 'bridge' is chosen for this shape because of a flat surface that looks similar to a half span of a bridge being supported from bottom. Parameters are 25 degree and 0.05 on horizontal and 5 on vertical oval factor

In Figure 17 the target area and the starting point are the same to Figure 16 (a rectangular cube shown in green). The oval factor is set to one in all directions, which causes more attraction in X and Y direction. The thicker branches are more incline toward the end of target area than the in Figure 18

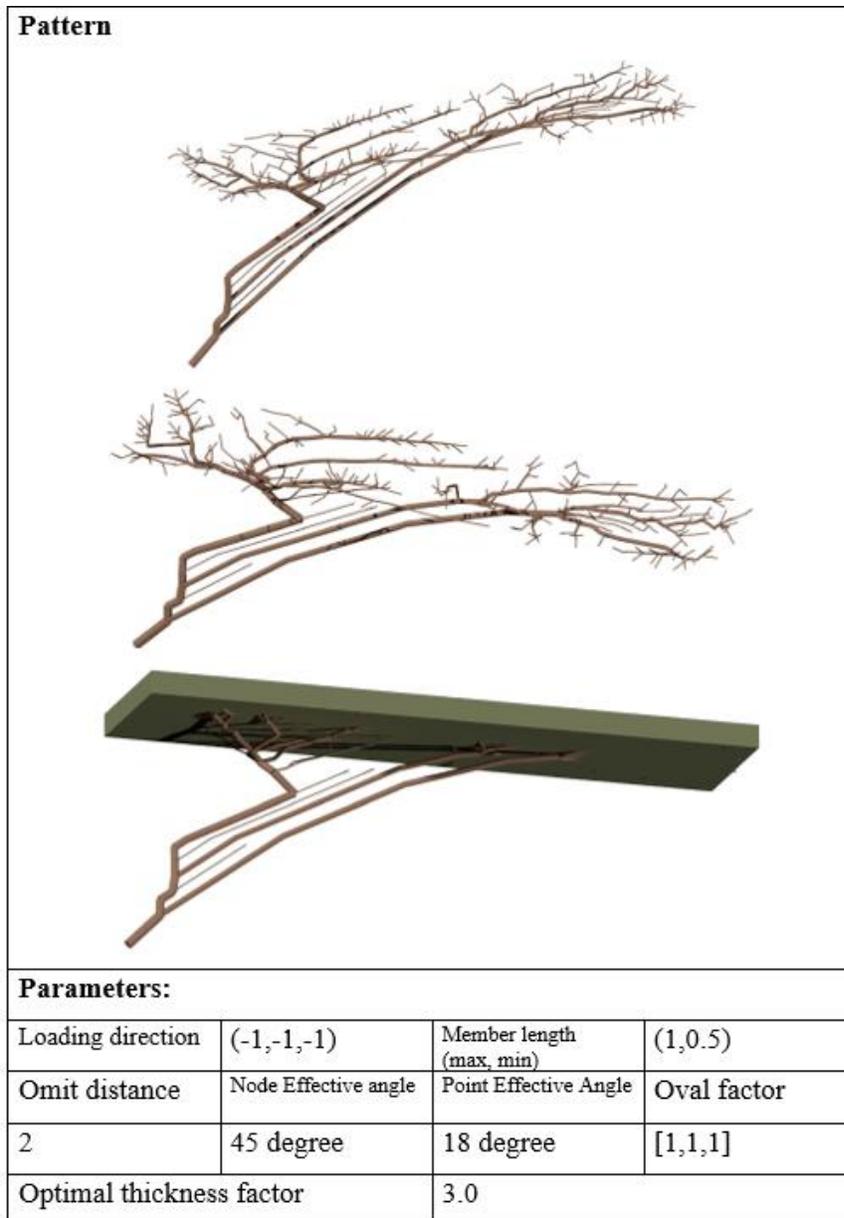


Figure 17 parameters are 45 degree for node effective angle and oval factor of 1 in all three directions

In Figure 18, the target area and the starting point are the same and an oval factor of 0.05 was applied on the horizontal surface. The change in oval factor reduces the attraction from far sections of the target area. In this case, the branches are mostly vertical and after reaching the boundary they spread horizontally.

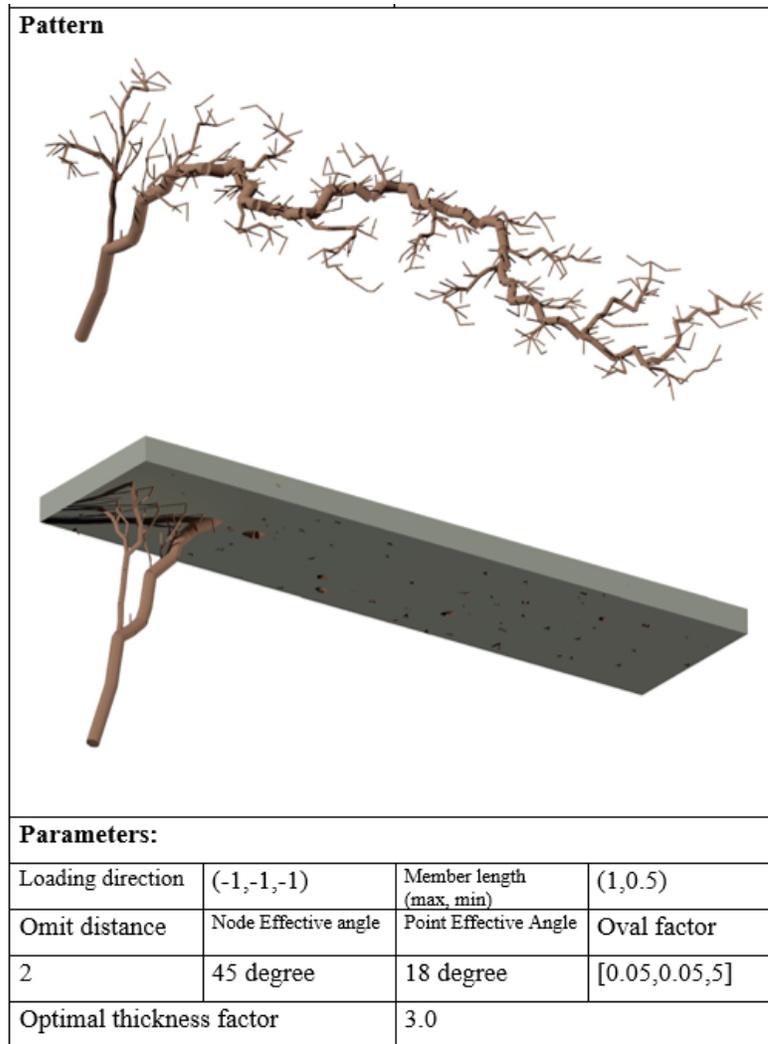


Figure 18 bridge with 45 degree and 0.05 on horizontal and 5 on vertical oval factor



Figure 19 comparison between the 25 and 45 degree approach. In 25 degree the further parts of target area was not covered by nodes in more vertical branches and thus new branches arise with more horizontal inclination

In Figure 20 the target area was in the shape of a tree which is shown with green spheres in the figures. The changing parameter is the oval factor. By decreasing the oval factor the attraction will be weekend. As the oval factor changes in the vertical and horizontal directions, the shape of branches changes from relatively vertical to more horizontal in the corresponding cases. The spreading will remain neutral if all factors are equal. In Figure 20-left, the oval factors were one for X and Y directions, and 0.1 for Z direction, which resulted horizontal branching. Figure 20-middle had 0.1 for horizontal direction, and one for vertical, thus the branching is more vertically directed. In Figure 20-right oval factors were equal to ‘one’ which created a normal distribution on branches.

Name:		Tree	
Pattern			
			
Parameters:			
Loading direction	(-1,-1,-1)	Member length (max, min)	(1,0.5)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor
2	45 degree	18 degree	[1,1,0.1]
Optimal thickness factor		3.0	

Name:		Tree	
Pattern			
			
Parameters:			
Loading direction	(-1,-1,-1)	Member length (max, min)	(1,0.5)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor
2	45 degree	18 degree	[0.1,0.1,1]
Optimal thickness factor		3.0	

Name:		Tree	
Pattern			
			
Parameters:			
Loading direction	(-1,-1,-1)	Member length (max, min)	(1,0.5)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor
2	45 degree	18 degree	[1,1,1]
Optimal thickness factor		3.0	

Figure 20 change of pattern due to change in oval factor Left: tree with oval factor of [1,1,0.1] and 45 degree in Node Effective Angle. Reduction in the vertical factor. Middle: tree with oval factor of [0.1, 0.1, 1] and 45 degree in Node Effective Angle. Branches have more vertical inclination. Right: tree with oval factor of [1,1,1] and 45 degree in Node Effective Angle. Due to equal factors branches are scattered in all directions.

In the Figure 21 and Figure 22, the target area is a portion of a cylindrical shell. The changing parameters are the node-effective-angle, omit-distance and oval factor. A large omit-distance generates fewer and more separated members as in Figure 22. Longer member length decreases the precision as in Figure 21.

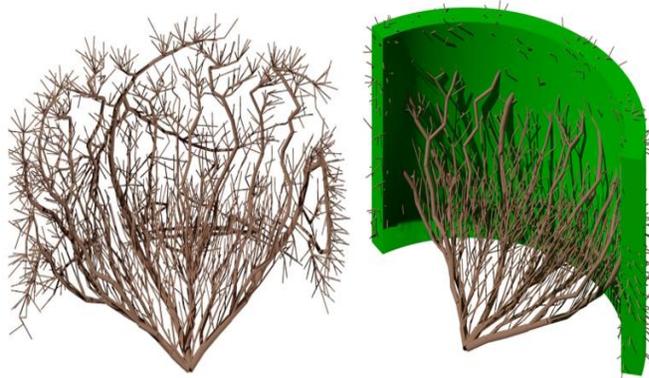
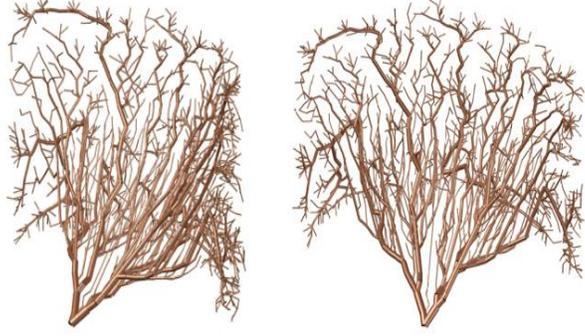
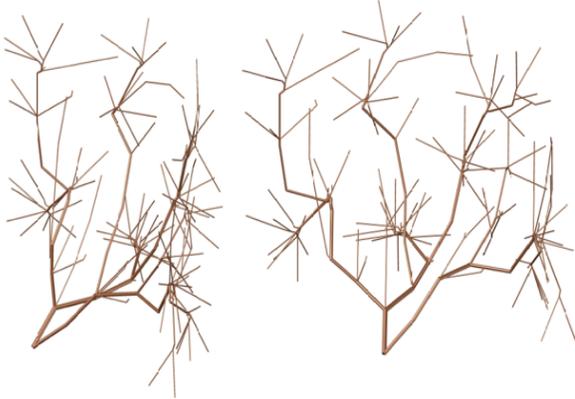
Name:		Curved wall		Name:		Curved wall	
Pattern				Pattern			
							
Parameters:				Parameters:			
Loading direction	(-1,-1,-1)	Member length (max, min)	(2,1)	Loading direction	(-1,-1,-1)	Member length (max, min)	(1,0.5)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor	Omit distance	Node Effective angle	Point Effective Angle	Oval factor
2.4	25 degree	18 degree	[1,1,1]	2	45 degree	18 degree	[1,1,1]
Optimal thickness factor		3.0		Optimal thickness factor		3.0	

Figure 21 Left: curved wall with omit distance of 2.4 and equal oval factor. Node effective angle is 25 degree. Right: curved wall with omit distance of 2 and equal oval factor node effective angle is 45 degree. The target area portion of a cylindrical shell, which is shown in green color

Name:		Curved wall 2	
Pattern			
			
Parameters:			
Loading direction	(-1,-1,-1)	Member length (max. min)	(4,1)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor
8	45 degree	18 degree	[1,1,1]
Optimal thickness factor	3		

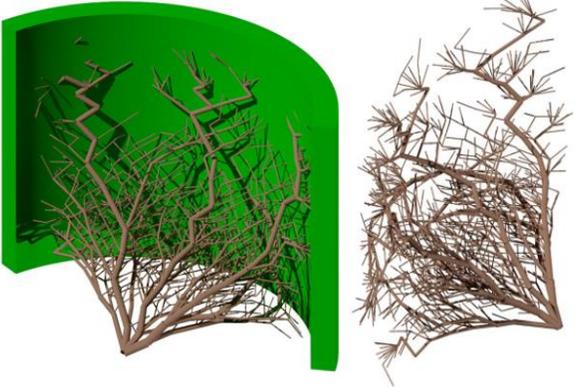
Name:		Curved wall	
Pattern			
			
Parameters:			
Loading direction	(-1,-1,-1)	Member length (max. min)	(2,1)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor
10	25 degree	18 degree	[1,1,0.1]
Optimal thickness factor	3.0		

Figure 22 Left: curved wall with omit distance of 8 and equal oval factor. Node effective angle is 45 degree. Right: curved wall with omit distance of 10 and equal oval factor. Node effective angle is 25 degree

Some other results are mentioned in the appendix B.

Chapter 6: Conclusions

6.1. Objective accomplishment

All the three objectives of the first phase were achieved by the introduction of the multi-vector distribution field. The first objective was to start from outside the target area and create multiple lines while approaching the target area which is demonstrated in Figure 16. Assigning one R^3 vector to each node and determining the direction based on that, gives only one straight line before getting to the boundary of target area and omitting some points. As long as the points structure is unchanged, there would be no change in the direction of the created lines, while having a multi-vector distribution field generates new direction as the request on that direction reaches a certain level.

Second accomplished objective is tolerating the symmetric formation of points in target area. The corresponding result can be observed in Figure 20 where the distribution of points are symmetric around the axis. If the points distribute evenly on a symmetrical shape like a circle or a sphere, and the starting node would be at the center, then, in conventional methods all the vectors cancel each other and there would be no initial direction. If the shape of the target would be the surface of a cone, then the direction for assigning the nodes will be the axis, and the nodes do not enter or cover the surface. Therefore, the previous algorithms used random points to prevent total cancelation of vectors. Multi-vector distribution field can generate shapes in all directions if located at the center of a sphere and does not depend on random points.

Third objective was to keep the effect of each point and not canceling the effect of a point in vector-summation. The multi-vector distribution field allows keeping the effect of each point in its corresponding direction and the point-effective-angle narrows the effect of points to their desired direction and minimizes the interaction effect of the points. Directional preference is another feature not existing in the previous algorithms. If the nature of the problem requires more spreading in one direction than the other directions, then the algorithm assigns an oval factor to the spatial distribution of the point requests.

Objective in the second phase was finding the optimal value for the parameter in branching equation (eq.16). The range for test was from one to six with a 0.5 step and the result in almost

all samples was three. The result may change due to other forms of loading which can be a topic for further researches.

6.2. Applications of the algorithm

This algorithm and similar one are attempts toward the idea of learning from nature and the nature-based designs. Besides that, several applications for this algorithm can be mentioned which doesn't cover all possible usages. The architectural design was the initial intention for this project, which would enable architects to suggest a branching pattern for covering any target area.

Another significant application is in the fluid (liquid and gas) distribution in the areas. Cooling/heating systems and distribution/collection of the gas and liquid can be designed by using this algorithm. The areas with more density of material/heat can be modeled by higher density of nodes.

Another application would be path finding. This algorithm suggests a path that starts from one location and covers the target areas which is capable of considering various densities for locations. Various required densities for a certain locations can be demonstrated by allocating more points to that location.

By looking at the human lung bronchi's pattern, another possible application would be in the design of fluid injection and drainage systems for any desired shapes. Due to the complexity of construction of such structures, this usage can be more applicable in accordance with 3D printing technology.

6.3. Further works and suggestions

All projects have limitations, which can be completed in the further studies. Some of the suggestions regarding this project would include:

1. Developing the capability to start the pattern generation from more than one initial point.
2. Using other equations for modeling.
3. Assessing the branch thickness equation under various load conditions.
4. Using other equations for branch thickness.

Bibliography

- De Reffye, P., Edelin, C., & Françon, J. (1988). Plant Models Faithful to Botanical Structure and Development. *ACM SIGGRAPH Computer Graphics*, 151-158.
- Rodrigues, R. A., & Bicho, A. (2010). An interactive model for steering behaviors of groups of characters. *Applied Artificial Intelligence*, 594–616.
- Argudo, O., Chica, A., & Andujar, c. (2016). Single-picture reconstruction and rendering of trees for plausible vegetation synthesis. *Computers and Graphics*, 55-67.
- Douglass, M. (2015, September 16). *BBC*. Retrieved from Nine incredible Building designs: <http://www.bbc.com/earth/story/20150913-nine-incredible-buildings-inspired-by-nature>
- ECstep. (2011). *ecstep.com*. Retrieved from NATURAL PATTERNS: <https://ecstep.com/natural-patterns/>
- Fernandes, G. D., & Fernandes, A. R. (2018). Space Colonisation for Procedural Road Generation. *2018 International Conference on Graphics and Interaction (ICGI)*. Lisbon: IEEE.
- Gonzales, Z. (2012, August 17). *ucreative.com*. Retrieved from you! Be Inspired! – 10 Nature Inspired Architectural Designs: <https://www.ucreative.com/inspiration/you-be-inspired-10-nature-inspired-architectural-designs/>
- Honda, H. (1971). Description of the form of trees by the parameters of the tree-like body: effects of the branching angle and the branch length on the sample of the tree-like body. *Journal of theoretical Biology*, 331-338.
- Jirasek, C., & Prusinkiewicz, P. (2000). Integrating biomechanics into developmental plant models expressed using L-systems. *Proceedings of the 3rd Plant Biomechanics Conference* (pp. 615-624). Stutgard: Georg Thieme Verlag.
- Li, S., & Gao, J. (2015). A dynamic root simulation model in response to soil moisture heterogeneity. *Mathematics and Computers in Simulation*, 40-50.

- Lima Bicho, A., Araújo Rodrigues, R., & Raupp Musse, S. (2012). Virtual Reality in Brazil 2011: Simulating crowds based on a space colonization algorithm. *Computers and Graphics*.
- Masaki, A., & Tosiyaasu, L. K. (1984). Botanical Tree Image Generation. *IEEE Computer Graphics and Applications*, 10-34.
- Pałubick, W., & Horel, K. (2009). Self-organizing tree models for image synthesis. *ACM Transactions on Graphics*.
- Pirk, S., Stava, O., Krat, J., Abd al Masih, M. S., & Neubert, B. (2012). Plastic trees: interactive self-adapting botanical tree models. *ACM Transactions on Graphics*.
- Przemyslaw, P., Mündermann, L., Karwowski, R., & Lane, B. (2001). The use of positional information in the modeling of plants. *annual conference on Computer graphics and interactive techniques*, pp. 289-300.
- Rosenkrantz, J. (2007). *Nervous System*. Retrieved from Nervous System: https://n-e-r-v-o-u-s.com/about_us.php
- Runions, A. (2005). *Modeling and visualization of leaf venation patterns*. Calgary: Department of Computer Science, University of Calgary.
- Runions, A. (2007). Modeling biological patterns using the space colonization algorithm. *university of Calgary*.
- Sachs, T., & Novoplansky, A. (1995). TREE FORM: ARCHITECTURAL MODELS DO NOT SUFFICE. *Israel Journal of Plant Sciences*, 203-212.
- Silveira, L., Maffei, R., Drews, P., & Bicho, A. (2012). A path-planning algorithm for multiple robots in unknown environments. *Journal of the Brazilian Computer Society*, 363-373.
- Takenaka, A. (1994). A simulation model of tree architecture development based on growth response to local light environment. *Journal of Plant Research*, 321-330.
- Weinzettl, J. (2017, February 3). *Biomimicry - learning from nature*. Retrieved from <https://www.xing.com>: <https://www.xing.com/news/insiders/articles/biomimicry-learning-from-nature-626416>

Appendix A: Method used for structural calculations

The elements in a joint have different directions, so local forces and moments should be converted to the global coordination, which will be performed by multiplying in the axis rotation matrix.

Local coordination:

At each joint the local X axis is always toward the element centerline. The Y and Z axis are in the plane perpendicular to the X axis. Two axis can have many combinations but adding one condition will reduce the degree of freedom and will determine the axis coordination. The condition, set for Y axis, is that the local Y axis wouldn't have any component in the global Z direction. By assuming the right hand rule local Z axis will be determined.

$$\vec{v} = (x_b - x_a), (y_b - y_a), (z_b - z_a) \quad \text{eq. A1}$$

$$\hat{x}_{local} = \frac{\vec{v}}{|\vec{v}|} = (x_1, x_2, x_3) \quad \text{eq. A2}$$

$$\hat{y}_{local} = \left(\frac{-x_2}{\sqrt{x_1^2 + x_2^2}}, \frac{x_1}{\sqrt{x_1^2 + x_2^2}}, 0 \right) = (y_1, y_2, y_3) \quad \text{eq. A3}$$

$$\hat{z}_{local} = \left(-\frac{x_1 * x_3}{\sqrt{x_1^2 + x_2^2}}, -\frac{x_2 * x_3}{\sqrt{x_1^2 + x_2^2}}, \sqrt{x_1^2 + x_2^2} \right) = (z_1, z_2, z_3) \quad \text{eq. A4}$$

$$[R]^T = \begin{bmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{bmatrix} \quad \text{eq. A5}$$

By adding all local rotated matrixes and combining them in a general matrix and applying the restrains, the structure stiffness matrix will be produced. Next step is assigning the external loads. As mentioned before **forces are only applied on** the end points, so that the force vector will be made. By solving the displacement equation, the deflections in all joints will be found.

$$\vec{F} = [M_{stiffness}] * \vec{\Delta} \quad \text{eq. A6}$$

$$\vec{\Delta} = [M_{stiffness}]^{-1} * \vec{F} \quad \text{eq. A7}$$

Appendix B: Other results

In Figure 23 a structure was separated into four sections and for each section a pattern was generated. The structure consists of two sections for roof, one section for side walls and one section for the curved wall. This pattern shows the use of algorithm in architecture as a visible heating system for building, or structural support.

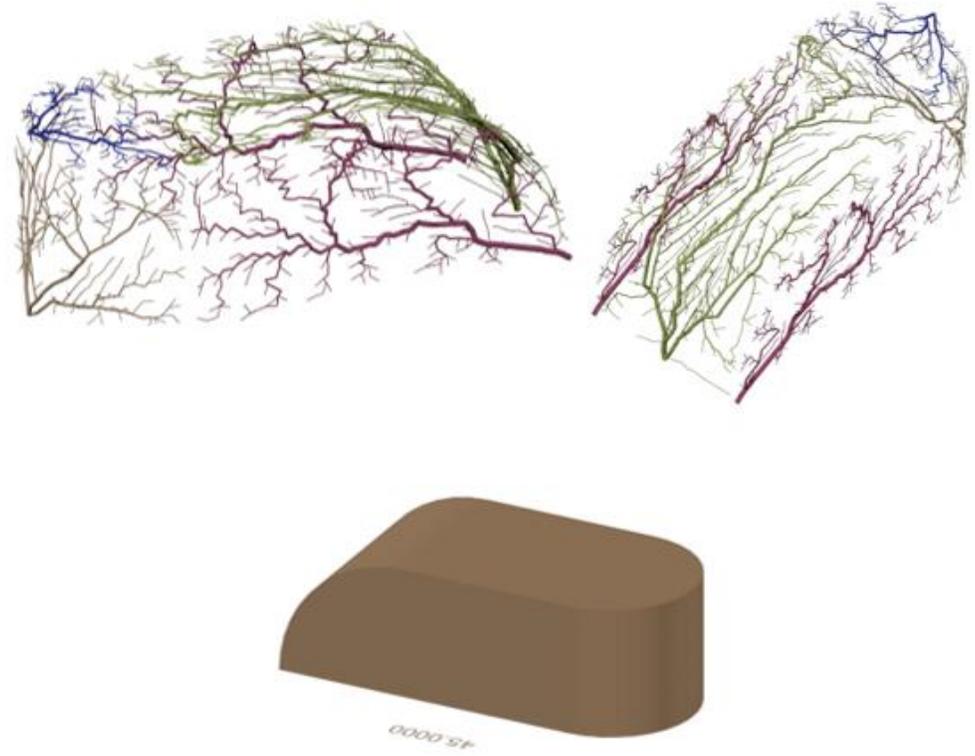
Name:		structure	
Pattern			
			
Parameters:			
Loading direction	(-1,-1,-1)	Member length (max, min)	(1,0.5)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor
2	45 degree	18 degree	[1,1,1]
Optimal thickness factor		3.0	

Figure 23 structure with 4 different patterns to cover walls and roof

Name:		Tree shell	
Pattern			
			
Parameters:			
Loading direction	(-1,-1,-1)	Member length (max, min)	(4,1)
Omit distance	Node Effective angle	Point Effective Angle	Oval factor
5	45 degree	18 degree	[1,1,1]
Optimal thickness factor		2.9	

Figure 24 shell of a tree shape with large member length

A canonical shape with symmetric distribution of points was given to program and the result is shown in Figure 25. In previous algorithms the pattern would have become a single line on the axis of the cone but the surface distribution field enabled the program to distinguish between requests from different directions.

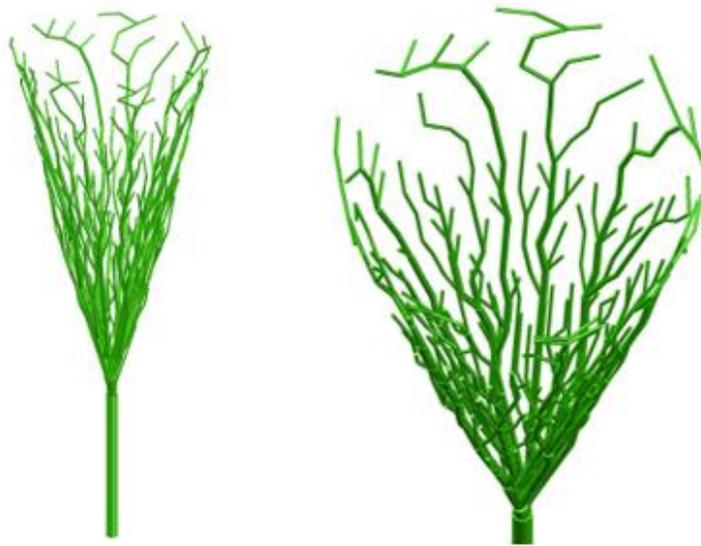


Figure 25 conical shape