

Modeling Plant Branching Structure with L-Systems

CSCI 4314/5314 Final Project

Allison Liu and Anna McTigue

Spring 2022

Abstract

Plants exhibit diverse branching structures, and generating computational models of these structures is useful to understand plant growth, improve plant identification, and create better computer renderings of plants. It is difficult to model branching structure with traditional statistical methods, so a different approach has been developed that is compatible with the self-similar fractal structure displayed by many plant species. Lindenmeyer Systems (L-Systems) were developed to model plant-like structures, and have been improved to address particular biological conditions. In our project, we used L-Systems to model the branching structure of specific plant species. We first generated specific structures using a series of L-System instructions, then we characterized how several parameters alter monopodial tree structure, and lastly we generated three dimensional models of monopodial and ternary trees of specific species.

1 Introduction

From nature, we are able to observe the incredible diversity of structures that plants take on. Understanding and quantifying the complex branching structure of plants could help us to develop better computer renderings of trees [1], improve models for plant growth, more accurately identify plants, and even calculate available biomass [2]. Additionally, modeling the morphology plants is potentially useful for plant identification and could yield key insights into how and why plants develop into the fractal-like shapes. However, the organic structure of plants makes them rather difficult and cumbersome to model using traditional geometric or statistical techniques.



Figure 1: Natural fractal tree branching

Due to the self-similar structure of plants, the property that they look roughly the same at any scale, it makes sense to use a recursive or iterative computational model to model the branching structures observed in the natural world. We use Lindenmayer systems (L-systems), which are a formal system to provide a mathematical description of plant growth using strings of characters that represent instructions and are

then recursively transformed into new instructions using a set of rules. L-systems were developed by a theoretical biologist named Aristid Lindenmayer in 1968 to describe the behavior of plant cells and model the growth processes of many biological organisms [3]. In this project, we use L-systems to characterize and quantitatively describe the branching structure of specific species of plants. We first investigate and detail the effects of each of the parameters. Then, we identify some diverse plant structures and modify parameters and instructions in the L-system model to generate outputs with similar branching structures.

2 Background

Because many trees appear very similar to human eye, we require quantitative methods for characterizing the chaotic appearance of tree structures in order to identify them. Recognizing the growth pattern and the branching patterns of trees can be very helpful for identifying different species of trees, modeling the amount of carbon dioxide that a tree is able to exchange with its environment, and modeling the process of water transpiring through a tree [2]. Leonardo DaVinci wrote that "all the branches of a tree at every stage of its height when put together are equal in thickness to the trunk" in the 15th century, which turned out to be true for many branching patterns found in nature as well as many self-similar structures [4] and the reasons for why have been investigated [5].

Since Aristid Lindenmayer first used L-systems to model the growth of plants and published *The Algorithmic Beauty of Plants* [6] a lot of progress towards creating a functional model has been made. Notably, Honda investigated branching patterns of real trees by tuning numerical parameters [7]. Aono and Kunii were the first to apply L-systems to the generation of botanical plants [8]. Murray et al. have used fractal analysis to monitor tree health by measuring the structural condition of trees [9]. In this project, we will gain a deep understanding of these previous works and build on them by creating models of trees and plant species of our own.

3 Methods

3.1 L-Systems

Structural patterns can be plotted in several ways. Most commonly, we use a series of points connected by lines to create a visual representation of a pattern. For L-Systems, we consider a "turtle" whose path is described as a starting point and a series of instructions like "Move forward, left, turn *d*degrees, stop" and have it trace a line behind the path that it traveled. This series of instructions can be represented symbolically using a series of strings. A very abbreviated list of example commands used in L-Systems are listed in Table 1. There are many more rather standardized commands including for moving half a step forward, adjusting the pitch, dividing the angle change scale, etc [10].

String Representation	Description of Action
F	move forward one unit and trace the path with a line
f	move forward but don't draw anything
-	rotate counter-clockwise but don't move
+	rotate clockwise but don't move
[stay still but remember it's current location and angle
]	return to the last remembered location, then forget about it

Table 1: Simple commands used in an L-System

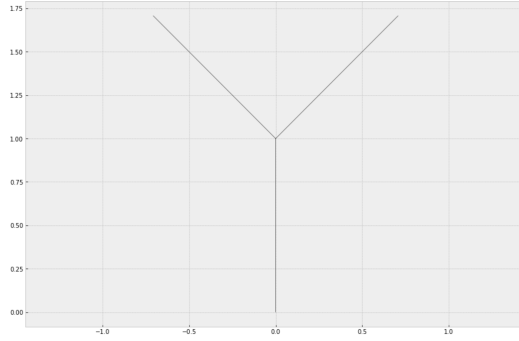


Figure 2: A simple branching pattern

For example, Fig 2 shows a very simple branching pattern, which appears if we give the turtle the instructions $F \rightarrow F[-F][+F]$.

Next, we can iterate this pattern to create more complex fractals. After iterating 2 times using the same string of instructions as before, we get the following pattern:

Iteration number n	Instruction String
0	F
1	$F[-F][+F]$
2	$F[-F][+F][-F[-F][+F]][+F[-F][+F]]$
\vdots	\vdots

We get the following visual in Fig 3 if we simply iterate this $\rightarrow F[-F][+F]$ mapping $n = 4$ times.

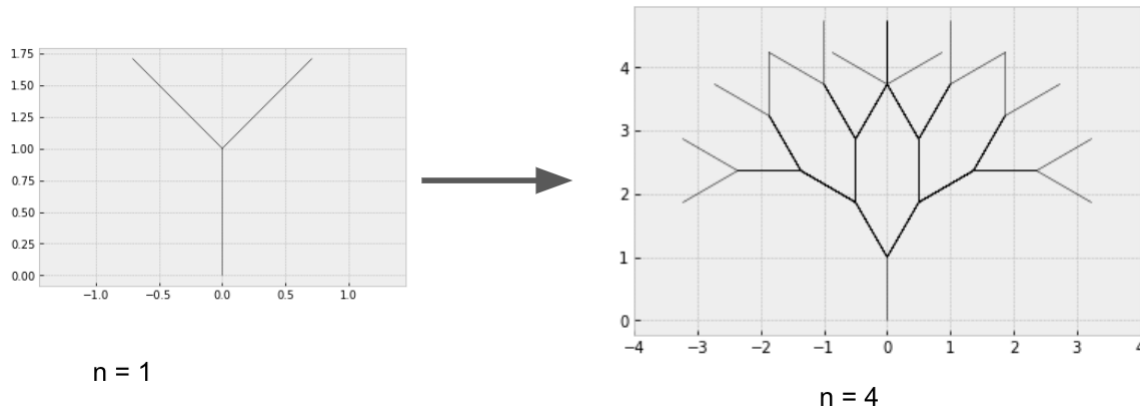


Figure 3: Simple branching pattern iterated 4 times

The tree in Fig 3 looks pretty boring. Now, we introduce some asymmetry to string of instructions and iterate it a few more times (Fig 4(a)). To make it appear less bush-like, we introduce an additional mapping $F \rightarrow FF$, which has the effect of making lines closer to the root appear longer (4(b)).

As an example, we have

Iteration number n	Instruction String
0	A
1	$F + A$
2	$F + F + A$
3	$F + F + F + A$
4	$F + F + F + F + A$

without the $F \rightarrow FF$ transformation. And we have

Iteration number n	Instruction String
0	A
1	$F + A$
2	$FF + F + A$
3	$FFFF + FF + F + A$
4	$FFFFFFFF + FFFF + FF + F + A$

with the $F \rightarrow FF$ transformation.

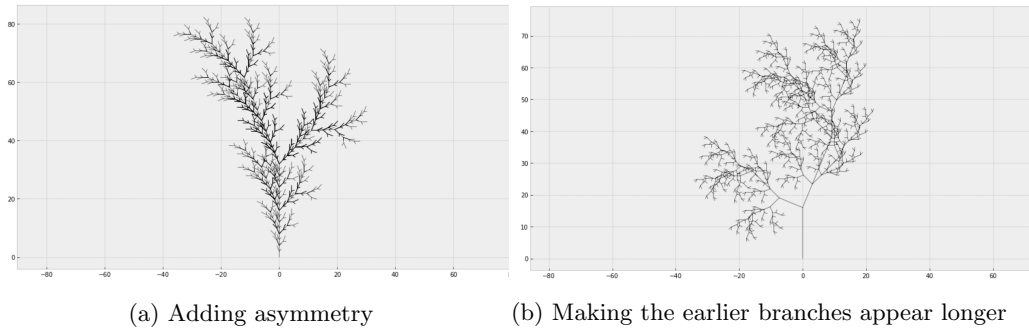


Figure 4: The effect of changing instruction strings and adding various transformations to our turtle's path

There are infinitely more modifications that can be made to these L-System commands that can result in plant like structures.

3.2 Three Dimensional Models and Parameterization

While the simpler L-system models described above provide useful exhibitions of branching structure, it is useful to expand these models to three dimensions. Introducing the Z axis allows for more complex structures that are more biologically relevant.

Additionally, it is useful to incorporate parameters that alter specific aspects of the L-system models to make them more manipulable. We aimed to include parameters to manipulate a variety of structural aspects of the model including contraction ratio of the trunk and branches, branching angle of the trunk and lateral axes, the divergence angle, and the width decrease rate. This allows for more precise manipulation of trunk/branch thickness, angle of branching, and overall structure. [10]

3.3 Monopodial vs Ternary Trees

We also aimed to manipulate a few different types of tree structures. We investigated the structure of monopodial and ternary trees as described in Honda's original paper [7].

Monopodial trees are composed of a single, central axis (trunk) and branches that project from that central point. Ternary trees are composed of a central axis split into three sections, each of which project their own branches. These two structures can be represented by the following set of rules as described in Houdini Kitchen [10].

Monopodial Tree Rules

- Premise: $A(1, 10)$
- Rule 1: $A(l, w) = F(l, w)[(c)B(l * e, w * h)]/(m)A(l * b, w * h)$
- Rule 2: $B(l, w) = F(l, w)[-(d)C(l * e, w * h)]C(l * b, w * h)$
- Rule 3: $C(l, w) = F(l, w)[+(d)B(l * e, w * h)]B(l * b, w * h)$

Ternary Tree Rules

- Premise: $F(0.5, 1)A$
- Rule 1: $A = TF(0.5, 1)[(c)F(0.5, 1)A]/(b)[(c)F(0.5, 1)A]/(e)[(c)F(0.5, 1)A]$
- Rule 2: $F(l, w) = F(l * d, w * h)$

4 Results

4.1 Modeling Specific Plants with Simple L-Systems

To begin investigating plant branching structures, we used a turtle to graph L-systems and capture particular aspects of a few specific plant structures.

4.1.1 Cow Parsley

The first plant that we modeled was cow parsley. This plant is characterized by a long stem that branches about a third of the way up the plant [11], and branches more frequently toward the tip of the plant. We were able to capture the location and frequency of branches in our model (Fig. 5). Our model is represented by the instructions: $F \rightarrow FF, A \rightarrow FF[+A]F[-A]+A$.

Cow Parsley

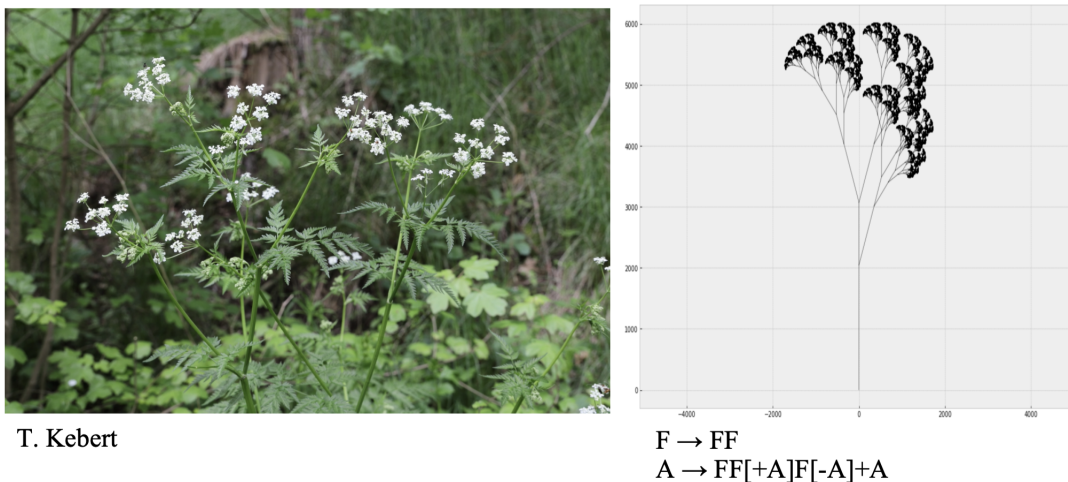


Figure 5: L-System model of cow parsley.

4.1.2 Indian Grass

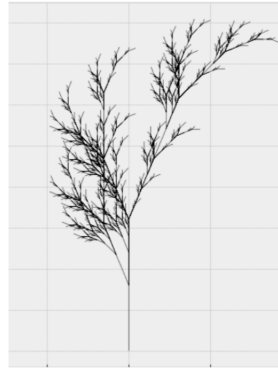
Next, we looked at Indian Grass, a species with a very different branching behavior [12]. In this model, we were able to capture the general asymmetrical shape of the grass (Fig. 6), and the long stem that ascends

into a more highly branched structure at the tips of each blade. This model is represented by the L-system instructions: $F \rightarrow FF, A \rightarrow F-[[A]+A]+F[+FA]-A$.

Indian Grass



High Country Gardens



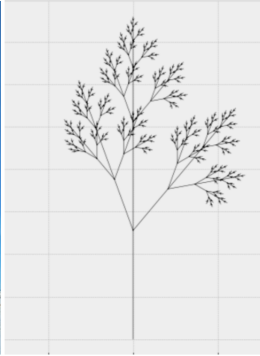
$F \rightarrow FF$
 $A \rightarrow F-[[A]+A]+F[+FA]-A$

Figure 6: L-System model of Indian grass.

4.1.3 Birch Tree

Lastly, we aimed to create a model for a birch tree. This tree is characterized by one long central trunk, and asymmetrical branches coming off that center trunk. Each of the branches then divide into smaller branches, with the smallest and most complex branches occurring at the ends. We were able to capture this general structure (Fig. 7), and our model can be represented by the L-system instructions: $F \rightarrow FF, A \rightarrow FF[++A][-A]FA$.

Birch Tree



$F \rightarrow FF$
 $A \rightarrow FF[++A][-A]FA$

Figure 7: L-System model of a birch tree.

4.2 Parameter Characterization

To better understand how the parameters implemented affect overall tree structure, we manipulated each parameter individually. Figure 8 includes detailed descriptions of how each parameter impacts tree structure. Once these parameters had been characterized, they were useful to manipulate the resulting structure to be more similar to specific plant species (section 4.3: Modeling Specific Plant Species).

Parameter Characterization of Monopodial Trees

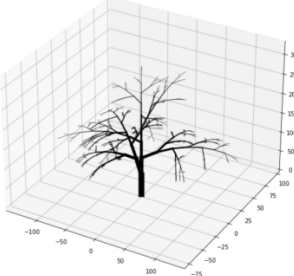
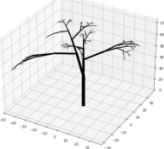
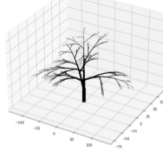
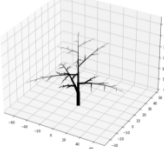
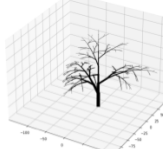
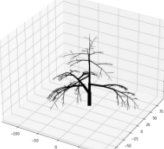
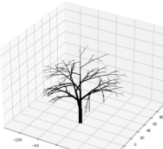
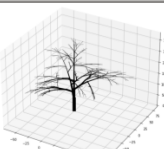
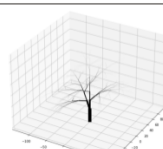
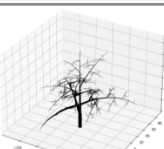
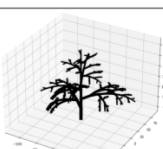
Parameters		Result	Observations		
r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/4 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.707 # width decrease rate			<ul style="list-style-type: none"> Fairly consistent branching structure Thickness decreases toward the top/ends of the tree 		
Parameters	Result	Observations	Parameters	Result	Observations
r1=0.6 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/4 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A lower contraction trunk ratio appears to reduce branching a bit and increase the relative thickness and length of the branches 	r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/6 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A smaller lateral axis branching angle means that the branches separate from each other less severely. It appears as though there is low repulsion between branches, and they therefore stay closer together.
r1=0.9 # contraction ratio trunk r2=0.3 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/4 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A lower contraction branch ratio appears to make the branches less spread out, and more straight compared to the example tree. 	r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/4 # branching angle lateral axes d=110*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A lower divergence angle appears to decrease the branch directional divergence from the trunk. For instance, the lower right branch is closer to the rest of the tree than when under the example conditions.
r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/2 # branching angle trunk a2=pi/4 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A larger branching trunk angle results in branches that look heavier, and come away from the trunk at a more severe angle. 	r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/4 # branching angle lateral axes d=165*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A higher divergence angle appears to increase the branch directional divergence from the trunk. For instance, the lower right branch is further from the rest of the tree than when under the example conditions.
r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/6 # branching angle trunk a2=pi/4 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A smaller branching trunk angle results in branches that look more springy and upward facing. They come away from the trunk at a less severe angle. 	r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/4 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.5 # width decrease rate		<ul style="list-style-type: none"> Decreasing the width decrease rate results in a faster decrease in thickness of the branches as the ends of the tree are approached.
r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/2 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.707 # width decrease rate		<ul style="list-style-type: none"> A larger lateral axis branching angle means that the branches separate from each other at more severe angles. It appears as though there is strong repulsion between the branches and they are optimizing distance. 	r1=0.9 # contraction ratio trunk r2=0.6 # contraction ratio branches a0=pi/4 # branching angle trunk a2=pi/4 # branching angle lateral axes d=137.5*pi/180 # divergence angle wr=0.9 # width decrease rate		<ul style="list-style-type: none"> Increasing the width decrease rate results in slower decrease in thickness of the branches as the ends of the tree are approached.

Figure 8: Parameters of the monopodial tree model were characterized. These consisted of the contraction ratio of the trunk and branches, the branching angle of the trunk and lateral axes, the divergence angle, and the width decrease rate.

The above example is done for monopodial trees, but a similar strategy could be employed to investigate parameter effects on structure for other tree types, including ternary trees.

4.3 Modeling Specific Plant Species

Now that we have established how different parameters effect the overall branching structure of the tree, we were equipped to model specific species of trees in three dimensions. To expand the types of structures that we can model, we used both monopodial and ternary trees (see section 3.3: Monopodial vs Ternary Trees).

4.3.1 Acacia Tree

The first tree that we modeled was the Acacia Tree, which has a particularly distinctive shape. It branches fairly early down the trunk, and forms this evenly spaced umbrella pattern of branches. We were able to capture the dispersed branching features in a monopodial model (Fig. 9), with the parameters listed below:

- Contraction ratio trunk: $r_1 = 0.7$
- Contraction ratio branches: $r_2 = 0.7$
- Branching angle trunk: $a_0 = \pi/6$
- Branching angle lateral axes: $a_2 = \pi/4$
- Divergence angle: $d = 137.5 * \pi/180$
- Width decrease rate: $w_r = 0.707$
- Iterations: $N = 7$

Acacia



Charles Sharp

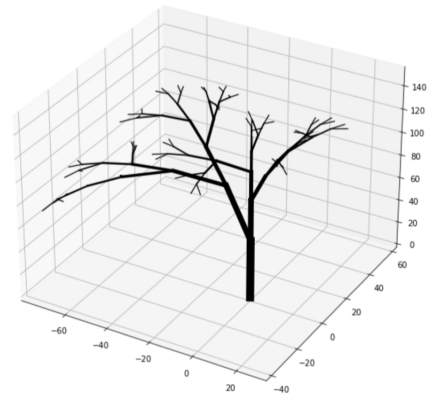


Figure 9: Acacia tree model using monopodial L-systems.

One feature of the tree we were not able to capture in our model is the roundness of the branches. In the image by Sharp 2004 (Fig. 9) [13], one can observe the curvature of the branches. It would be interesting to try to build a model in the future that can include this feature.

4.3.2 Weeping Willow

We next created a model for the weeping willow using a ternary tree. This tree is characterized by a split around halfway up the tree, and long droopy branches sourced from the middle and upper end of the trunk [14]. We were able to represent these features in our model (Fig. 10). We included two different trees that result from the same set of parameters, one with 6 iterations and the other with 7 iterations. This shows that more iterations result in a more highly branched structure, and this can be altered to represent the respective tree being modeled. The parameters used to create this model are as listed:

- Divergence angle 1: $d1 = 94.74 * \pi/180$
- Divergence angle 2: $d2 = 132.63 * \pi/180$
- Branching angle: $a = 18.95 * \pi/180$
- Elongation rate: $lr = 1.15$

- Width increase rate: $vr = 1$
- Iterations: $N = 6, N = 7$

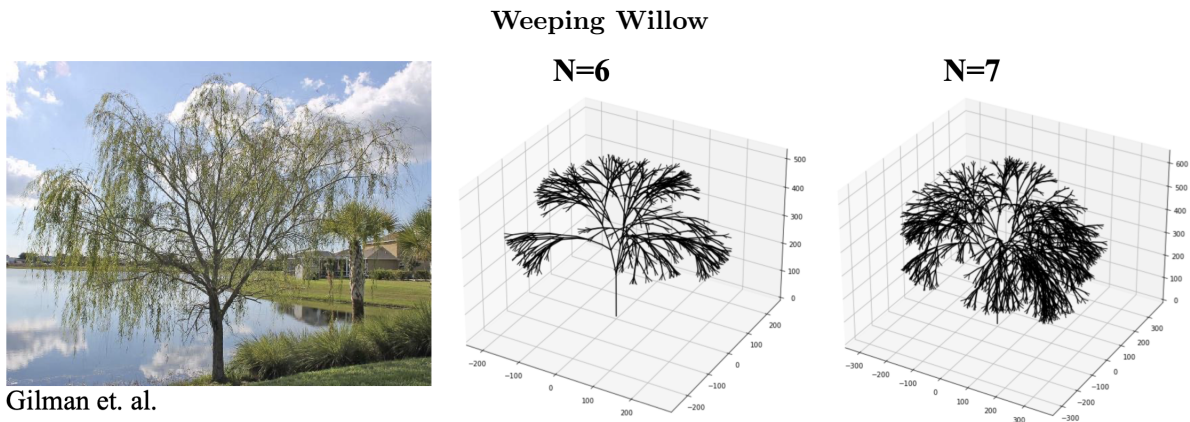


Figure 10: Willow tree model using ternary L-systems.

5 Conclusion

In this project, we used parameterized L-systems to model specific species of plants. We have shown how simple L-system instructions can be used to generate complex structures, and capture particular branching features. Additionally, we have characterized the parameters that can alter monopodial tree structures with the same set of rules. We also altered parameters on monopodial and ternary trees to produce three dimensional models of specific tree species. The results in this project as a whole lead to a better understanding of how to computationally represent particular features of plant branching structures. This is potentially useful to improve our understanding of plant growth and development, to develop better computer renderings of plants, and to improve plant identification. L-systems are an excellent strategy to model plant structures, and there is much that can still be done to improve their accuracy and usefulness in the future.

6 Future Work

To continue this project, we would be interested in integrating artificial neural networks to evaluate the generation of the plants we have created. This would be a good way to determine the accuracy and realism of the plants that have been generated. Furthermore, we would be excited to implement a General Adversarial Network (GAN) to create mappings from our generated structure models into realistic photos of their respective plants. This could be a good way to create more evaluation data for plant identification tasks and for training plant identification models.

Code

The code we used for this project can be found here: <https://github.com/allisonliu/tree-modeling>. It was based on models from these repositories:

- <https://github.com/ThomasLENNE/L-system>
- <https://github.com/paulgb/bitaesthetics/blob/master/notebooks/Fractal%20Generation%20with%20L-Systems.ipynb>
- <https://github.com/ambron60/l-system-drawing>

Contributions

Anna:

- characterizing parameters of monopodial trees
- using the code to generate models of specific tree species
- report: abstract, results, background, conclusion

Allison:

- gathering and implementing Python code
- background research on L-systems
- report: introduction, background, methods, future work

References

- [1] R. Viruchpintu and N. Khiripet, “Real-time 3 d plant structure modeling by l-system with actual measurement parameters,” 2006.
- [2] L. P. Bentley, J. C. Stegen, V. M. Savage, *et al.*, “An empirical assessment of tree branching networks and implications for plant allometric scaling models,” *Ecology Letters*, vol. 16, no. 8, pp. 1069–1078, 2013, eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/ele.12127>. DOI: <https://doi.org/10.1111/ele.12127>. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1111/ele.12127>.
- [3] A. Lindenmayer, “Mathematical models for cellular interactions in development i. filaments with one-sided inputs,” *Journal of Theoretical Biology*, vol. 18, no. 3, pp. 280–299, 1968, ISSN: 0022-5193. DOI: [https://doi.org/10.1016/0022-5193\(68\)90079-9](https://doi.org/10.1016/0022-5193(68)90079-9). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0022519368900799>.
- [4] R. Minamino and M. Tateno, “Tree Branching: Leonardo da Vinci’s Rule versus Biomechanical Models,” *PLOS ONE*, vol. 9, no. 4, pp. 1–12, Apr. 2014, Publisher: Public Library of Science. DOI: [10.1371/journal.pone.0093535](https://doi.org/10.1371/journal.pone.0093535). [Online]. Available: <https://doi.org/10.1371/journal.pone.0093535>.
- [5] C. Eloy, “Leonardo’s rule, self-similarity, and wind-induced stresses in trees,” *Physical review letters*, vol. 107, p. 258 101, Dec. 2011. DOI: [10.1103/PhysRevLett.107.258101](https://doi.org/10.1103/PhysRevLett.107.258101).
- [6] P. Prusinkiewicz and A. Lindenmayer, “The algorithmic beauty of plants,” in *The Virtual Laboratory*, 1990.
- [7] H. Honda, “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 shape of the tree-like body,” *Journal of Theoretical Biology*, vol. 31, no. 2, pp. 331–338, 1971, ISSN: 0022-5193.
- [8] M. Aono and T. L. Kunii, “Botanical tree image generation,” *IEEE Computer Graphics and Applications*, vol. 4, no. 5, pp. 10–34, 1984. DOI: [10.1109/MCG.1984.276141](https://doi.org/10.1109/MCG.1984.276141).
- [9] J. Murray, G. A. Blackburn, J. D. Whyatt, and C. Edwards, “Using fractal analysis of crown images to measure the structural condition of trees,” *Forestry: An International Journal of Forest Research*, vol. 91, no. 4, pp. 480–491, Mar. 2018, eprint: <https://academic.oup.com/forestry/article-pdf/91/4/480/25432212/cpy008.pdf>, ISSN: 0015-752X. DOI: [10.1093/forestry/cpy008](https://doi.org/10.1093/forestry/cpy008). [Online]. Available: <https://doi.org/10.1093/forestry/cpy008>.
- [10] *How to create L-Systems – Houdini Kitchen*, en-US. [Online]. Available: <https://www.houdinikitchen.net/2019/12/21/how-to-create-l-systems/> (visited on 05/03/2022).
- [11] T. Kebert, *File:anthriscus sylvestris tk 2021-05-16 2.jpg — wikimedia commons, the free media repository*, [Online; accessed 29-April-2022], 2021. [Online]. Available: https://commons.wikimedia.org/w/index.php?title=File:Anthriscus_sylvestris_TK_2021-05-16_2.jpg&oldid=560915744.
- [12] *Thin man indian grass*, [Online; accessed 29-April-2022]. [Online]. Available: <https://www.highcountrygardens.com/perennial-plants/ornamental-grass/sorghastrum-nutans-thin-man>.
- [13] C. J. Sharp, *Acacia tree in the serengeti, tanzania*, 2004. [Online]. Available: <https://commons.wikimedia.org/wiki/File:Acacia-tree.jpg>.
- [14] “Salix babylonica: Weeping willow,” 2019, [Online; accessed 29-April-2022]. [Online]. Available: <https://edis.ifas.ufl.edu/publication/ST576>.