

L-Systems: The Good, the Bad, and the Beautiful

---

A Thesis  
Presented to  
The Division of Mathematics and Natural Sciences  
Reed College

---

In Partial Fulfillment  
of the Requirements for the Degree  
Bachelor of Arts

---

R Jacob Hoopes

May 2023



Approved for the Division  
(Computer Science)

---

Dylan McNamee



# Acknowledgements

Thanks to my roommates, Henry Wilson, Brent Ellis, and especially Gabe Fish for the late night conversations that led to some of the more interesting choices made and questions raised throughout this document.



# Table of Contents

<b>Introduction</b> . . . . .	<b>1</b>
0.1 Motivations . . . . .	1
<b>Chapter 1: What is an L-System?</b> . . . . .	<b>3</b>
1.1 Understanding through Definitions . . . . .	3
1.2 Understanding through Images . . . . .	4
1.2.1 Turtle . . . . .	4
1.2.2 Not Changing Position . . . . .	4
1.3 Understanding through Big Words . . . . .	5
<b>Chapter 2: Explaining the Program</b> . . . . .	<b>7</b>
<b>Chapter 3: Contribution</b> . . . . .	<b>9</b>
3.1 Related Work . . . . .	9
3.2 Generative Art . . . . .	10
3.2.1 My Explorations . . . . .	10
3.3 Philosophical Quandaries . . . . .	10
3.3.1 Is Math Art? . . . . .	10
3.3.2 The Times We Live In . . . . .	10
<b>Chapter 4: Next Steps</b> . . . . .	<b>11</b>
4.1 Extending the Program . . . . .	11
4.2 The work of Przemyslaw Prusinkiewicz and Aristid Lindenmayer . . . . .	11
4.3 Generalizing L-Systems . . . . .	11
4.3.1 What does it <i>Mean</i> to Change Dimensions . . . . .	11
4.3.2 Alternative Dimensions . . . . .	12
4.3.3 The Problem of Depiction . . . . .	12
<b>Conclusion</b> . . . . .	<b>13</b>
<b>Appendix A: Extra Images</b> . . . . .	<b>15</b>
<b>Appendix B: Generative Art Resources</b> . . . . .	<b>17</b>
<b>References</b> . . . . .	<b>19</b>





# Abstract

The world is a mess of systems. I hope to establish the context for one system that interests me particularly, and build it up for you so that you may be as interested in it as I have been.



# Dedication

This thesis is dedicated to the worms in the ground and the trees waving in the wind.



# Introduction:

[\[Illustration of a simple L-System, perhaps just a fractal tree\]](#)

The world of Computer Science can seem like a wilderness, teeming with frightening phrases like Zero-knowledge proofs, NP-Completeness, Finite State Machines, and the Halting Problem, but I hope to open up this world as one worthy of exploration. Specifically exploration at the hands of those from distant disciplines, like Anthropology, Economics, and Psychology, as well as the locals, pure Math, Bio, Chem, and Physics. The tool I will show you and give to you will demonstrate the accessibility of one of the most powerful, subtle, and yet strangely familiar disciplines: Computer Science. These tools have been given the name L-Systems, and their capabilities and beauty can astound.

Before we jump in, there are some things that must be said. This is a Computer Science Thesis, and so (essentially by necessity,) there will be code. I would like to keep this Thesis as accessible as possible to those who don't code, and so I have tried to design it so that anyone with the ability to feel wonder and joy can find something of value between these covers. The code will be accessible on my github page ([github.com/JacobHoopes/Thesis](https://github.com/JacobHoopes/Thesis)) and hosted at at least one other location if possible. The code of the central project will also be accessible in the last pages of the Thesis document itself. However! I have included pictures on every page I could, and I'd like to imagine that they tell enough of a story by themselves that the words are only a lovingly crafted garnish. I suppose we'll see if I'm right.

Lindenmayer Systems, usually called L-Systems, are a type of "rewriting" system that have a distinct visual and

[\[Illustration of an L-System, something that parallels the first one, and yet does something dramatically different \(maybe the same image, just upside down? Inverted somehow?\) Maybe just another cool L-System.\]](#)

## 0.1 Motivations

There are several motivations for this thesis, each which I hope will appeal to people from different disciplines with radically different interests. I hope that this thesis will be able to pull on interesting threads in enough of a variety of disciplines that everyone will find something interesting to look at in this short body of work. The primary motivation, and the reason that I am personally so invested in this topic, is the idea of *pattern*. Patterns that I demonstrate with L-Systems and other generative arts appear at different scales throughout life, the world, and our universe. My friends

and acquaintances are often much more versed than I am in many of the structures that I reference throughout this thesis, in areas such as biology and chemistry, but also astronomy and physics. I hope to draw in Economists and Anthropologists as well with discussion of patterns that appear in the structures of society and social organization, and which can be illustrated to some extent with the methods that I outline here.

The second main motivation, one that I try to emphasize at every opportunity, is the pure artistry behind so much of the patterns and methods that I explore. This is the motivation that I imagine might be the most interesting to artists, be they visual artists or otherwise. This may be the reason I was drawn into discussion of L-Systems and their relatives in the first place, so it is as essential a part of this story as the first motivation.

The third motivation is interested in a philosophical perception of these structures. I am not a trained philosopher, and I would not be surprised to learn if many of the questions that I ask as a part of my exploration are in well-tread ground. It's possible the questions I ask in this thesis will contribute to the conversations where I imagine they might find a home. Some of the questions I ask address larger issues of the place of machines in the creation of art and the creative process. These questions feel to me to be inseparable from the rest of the work.

I hope that there will be future interest in L-Systems, either as a result of the work that I've done here or as part of some other popular exploration of their capabilities. This came to be a motivating project for me for more reasons beyond the three outlined above, and I imagine that someone who sets new eyes on this work could see it in a way I'd never could have foreseen. I go into more detail near the end of the paper about what I believe I have contributed to the conversation around L-Systems and their relatives, as well as what I believe this tool to be capable of with the proper aid.

# Chapter 1

## What is an L-System?

(What are we talking about? What have I dedicated months of my time and energy to communicating about as best I can? The answer feels complicated after having delved into the depths for a while, but an introduction should be shallow, and so I'll do that now. L-Systems are recursive replacement system with a unique visual identity. What is a recursive replacement systems? [Perhaps explain in detail, perhaps not - remember, I want to keep this accessible])

---

This is an L-System:

[\[Picture of impressive looking L-system\]](#)

Well, it would be more right to say that this is the *output* of an L-System. L-Systems are a *technique* with which one can describe a complex, changing system with just a few letters and symbols. Some Biologists may take offence, but it may be compared to DNA in how it compresses a potentially infinite complexity into a handful of characters. This chapter will cover a few main ideas. The first, and most central, is a discussion of the mechanics of how they work and how to build them. After that, I'll go on to describe some of their features, some of their other traits and patterns, and their applications. As part of that last point, I'll illustrate particularly effective ways these systems can be used, how I've used them, (including some early experimentations before the central project,) and end with a lead-in to the central coding portion of this thesis, the L-System Drawer.

There are a couple effective ways I've found to introduce L-Systems to people. I'll go through each of these different strategies to give you some solid footing before we start to start to delve into the more complicated material.

### 1.1 Understanding through Definitions

L-Systems have three main parts. The first part is the "start" variable. It is usually just a character, we'll call it "A" here. The second part is a set of rules that describe how the L-System changes. These are usually written like " $A \rightarrow AB$ " or " $B \rightarrow AA$ ", where each of these is a different rule. There are different names for what "A" and "B" are, but in this thesis I will call them "Productions". In a similar

fashion, each of these rules is called a “Production Rule”. (Any character without a corresponding production rule is therefore not a production. This will be relevant later.) The third part of an L-System is a list of other information that is used when it is being constructed. For most of the L-Systems discussed in this thesis, there are two pieces of information in this list, a value  $\theta$  which is read as the “angle” and a value “Iterations” which is the number of times the system is generated. Alternatives are discussed in Chapter 4.

## 1.2 Understanding through Images

By using this sort of image to describe these structures, I define the dimensions in which they operate and therefore lose access to some of their possible complexity. I suggest alternative visualization strategies in Chapter 4. [\[A series of images that show the progression of an L-System through multiple generations. Each of the images is labelled with a short description that explains the process shown in that image.\]](#)

This set of images is generated with the parameters, [start: A; rules:  $A \rightarrow AB$ ,  $B \rightarrow AA$ ;  $\theta : 45^\circ$ ]. Different values for iterations are displayed in each image. “A” is drawn as a line going upward and “B” is drawn as a line going to the right. Each new production starts from where the last one left off.

[\[Image A0\]](#): Iterations = 0, current string: A

[\[Image A1\]](#): Iterations = 1, current string: AB

[\[Image A2\]](#): Iterations = 2, current string: ABAA

[\[Image A3\]](#): Iterations = 3, current string: ABAAABAB

Here is a different set of images with the same start variable and extra information, but alternative production rules. These rules are:  $A \rightarrow BAB$ ,  $B \rightarrow A$ .

[\[Image B0\]](#): Iterations = 0, current string: A

[\[Image B1\]](#): Iterations = 1, current string: BAB

[\[Image B2\]](#): Iterations = 2, current string: ABABA

[\[Image B3\]](#): Iterations = 3, current string: BABABABABAB

### 1.2.1 Turtle

Introduce Turtle. Demonstrate its abilities and ways in which it is especially powerful at drawing L-Systems. Show how it takes advantage of the idea of “state” to

### 1.2.2 Not Changing Position

Introduce the idea of the “state” of the turtle. Introduce the idea that a production is able to change parts of the state



## 1.3 Understanding through Big Words

L-Systems are what we in CS would call a *Recursive Replacement structure*. This describes the way in which an L-System makes more of itself.



## Chapter 2

# Explaining the Program

The program that I've created for this thesis demonstrates the structure of an L-System and allows the user to interact with the parameters to dynamically adjust the displayed content. This "L-System Illustrator"

L-Systems are a way to see every stage of the life-cycle of a tree simultaneously. There are limitations to this metaphor, but the freedom of understanding that is gained with the metaphor more than makes up for the rough patches. It's also accurate especially by the use of the "tree" metaphor.



# Chapter 3

## Contribution

A primary goal of this work is to make L-Systems, and by extension, generative art and even Computer Science as a whole, less frightening. I want people who had no intention of getting involved in the intricacies of 1s and 0s begin to see beauty in the geometry. I want folks who have never used computers become able to create things on their own initiative to be amazed and inspired by the possibilities. I even want my peers, who have been immersed in the depths of CS for years, to be able to approach the discipline which they've sunk their whole minds into with a fresh awe and understanding that the possibilities are as boundless as they've dreamed and perhaps even more so. These are grand dreams, to be sure, but computers take the largest of tasks with little more difficulty than the smallest, and art can inspire us to do great things. I hope that this will spark something like that in my readers, though even just using the resources I've collected and attach to pursue your own interests is a dreamy enough prospect.

I am fully aware that L-Systems have been around for 55 years, since they were proposed back in 1968 by Lindenmayer, and for a time I was worried that with 55 years to develop these ideas I wouldn't be able to tread on any new ground. This is true in many respects, and I re-emphasize that another major goal of this work is to raise awareness of L-Systems and their capabilities as well as to bring together as many sources as possible that discuss these ideas. However, this work goes beyond that. There are

A central contribution of my own has been the development of the interactive L-System Illustrator

### 3.1 Related Work

Go into depth about different implementations of this work that I've found. Advertise resources that I found useful in the development of my understanding about these topics.

## **3.2 Generative Art**

Include references to some of the numerous types of generative art that I've come across during my research.

### **3.2.1 My Explorations**

Show some of the different projects that I worked on in the build-up to this thesis. Explain how they led into L-Systems or any other part of my work that eventually coalesced into this thesis.

## **3.3 Philosophical Quandaries**

Here is where I feel it's appropriate to ask some of the big questions. Start with an intro explaining and defending the existence of this section,

### **3.3.1 Is Math Art?**

Is this art? Where do we draw the line between art and math?

### **3.3.2 The Times We Live In**

What does it mean for a machine to make art?

# Chapter 4

## Next Steps

### 4.1 Extending the Program

Talk briefly about different ways in which the program could be extended to address what I perceive to be flaws or oversights. Mention things that I wished I had time to change, but go beyond this to discuss features that might deserve their own program. Include a lead-in to the next section

### 4.2 The work of Przemyslaw Prusinkiewicz and Aristid Lindenmayer

Use this section to talk about the wide variety of other forms of L-Systems that I have not investigated in this Thesis. I will make mention of parametric L-Systems, stochastic L-Systems, context-dependent L-Systems, and L-Systems that are used for a wide variety of other uses. From tree-modeling at Pixar to demonstrating cell-growth in biology labs. Say essentially that *The Algorithmic Beauty of Plants* has an abundance of thoughtfully demonstrated resources and examples, as well as many compelling pictures.

### 4.3 Generalizing L-Systems

So far, we've only been looking at L-Systems that fall into a very narrow category. In addition to the types of L-Systems explored by Prusinkiewicz and Lindenmayer, there is an opportunity that I feel is extremely untapped to explore what it would mean if we change the dimensions of an L-System.

#### 4.3.1 What does it *Mean* to Change Dimensions

When the state at a particular point in the construction of an L-System changes, it alters the way that the L-System is displayed. This can be especially obvious and dramatic if the part of the state that is changed is the angle. By changing the angle

by only a small amount, let's say a degree, we affect the position and rotation of every following node. Changing the position part of the state will only change the position of later nodes. This seems to suggest that somehow, changing the angle is a *more powerful* alteration than changing the position. Looking more closely at what actually separates changing position from changing angle, we can see that the angle never actually draws anything. The value that draws things is always F. We can take our simplified L-System and see that there is a depth of nuance to the values we implicitly chose to use at the beginning of this process. The following table shows the differences between these two operations, the rotate operation and the forward operation.

Table 4.1: Rotate vs Forward Atomization

Attribute	Following Node has the Same Position	Following Node has the Same Orientation	Draws
Rotate	Y	N	N
Forward	N	Y	Y

Every time the Forward operation is read, two things happen, while when the Rotate operation is read, only one thing happens. It seems like it might be possible to create a more specific operation for forward that breaks it up into its two parts. The way that people have found to get around this problem is by introducing a second type of Forward operation that changes position but doesn't draw. Usually this is notated by a lowercase f instead of the usual capital F for the regular Forward operation. We could approach the forward operation in a different way - perhaps it would be more right to call it a "draw" command, and the way it draws is governed by either the state of the L-System at the moment when the operation is read or the parameters of the L-System at the outset, or both. If we begin to think about the forward command as a draw command instead, a new world of possibilities opens to us.

### 4.3.2 Alternative Dimensions

What if we choose to change our chosen dimensions to display the same L-System in a different way? There are a few ways to do this *I think*). First, we must figure out possible alternative dimensions. Beyond the basic three spatial dimensions, a classic dimension is time. This suggests that there might be a way to display L-Systems with variations in time, perhaps in the form of an animation. Some other classic dimensions that people have used with L-Systems are width and length, determining how wide or long a line is drawn when the Forward operation is read. We could change the color of the line, so that it changes hue, saturation, or opacity according to some color space. We could alter the curvature of the line, perhaps in such a way that the Forward operation changes ...



### 4.3.3 The Problem of Depiction

Some dimensions lend themselves more easily to being displayed. For example, if we try to display an L-System in which the dimension that



## Conclusion



# Appendix A

## Extra Images



## Appendix B

### Generative Art Resources





# References

- Angel, E. (2000). *Interactive Computer Graphics : A Top-Down Approach with OpenGL*. Boston, MA: Addison Wesley Longman.
- Angel, E. (2001a). *Batch-file Computer Graphics : A Bottom-Up Approach with QuickTime*. Boston, MA: Wesley Addison Longman.
- Angel, E. (2001b). *test second book by angel*. Boston, MA: Wesley Addison Longman.
- Deussen, O., & Strothotte, T. (2000). Computer-generated pen-and-ink illustration of trees. *“Proceedings of” SIGGRAPH 2000*, (pp. 13–18).
- Fisher, R., Perkins, S., Walker, A., & Wolfart, E. (1997). *Hypermedia Image Processing Reference*. New York, NY: John Wiley & Sons.
- Gooch, B., & Gooch, A. (2001a). *Non-Photorealistic Rendering*. Natick, Massachusetts: A K Peters.
- Gooch, B., & Gooch, A. (2001b). *Test second book by gooches*. Natick, Massachusetts: A K Peters.
- Hertzmann, A., & Zorin, D. (2000). Illustrating smooth surfaces. *Proceedings of SIGGRAPH 2000*, 5(17), 517–526.
- Jain, A. K. (1989). *Fundamentals of Digital Image Processing*. Englewood Cliffs, New Jersey: Prentice-Hall.
- Molina, S. T., & Borkovec, T. D. (1994). The Penn State worry questionnaire: Psychometric properties and associated characteristics. In G. C. L. Davey, & F. Tallis (Eds.), *Worrying: Perspectives on theory, assessment and treatment*, (pp. 265–283). New York: Wiley.
- Noble, S. G. (2002). *Turning images into simple line-art*. Undergraduate thesis, Reed College.
- Reed College (2007). Latex your document. <http://web.reed.edu/cis/help/LaTeX/index.html>
- Russ, J. C. (1995). *The Image Processing Handbook, Second Edition*. Boca Raton, Florida: CRC Press.

- Salisbury, M. P., Wong, M. T., Hughes, J. F., & Salesin, D. H. (1997). Orientable textures for image-based pen-and-ink illustration. *“Proceedings of” SIGGRAPH 97*, (pp. 401–406).
- Savitch, W. (2001). *JAVA: An Introduction to Computer Science & Programming*. Upper Saddle River, New Jersey: Prentice Hall.
- Wong, E. (1999). *Artistic Rendering of Portrait Photographs*. Master’s thesis, Cornell University.