Generating and visualising realistic plants

Yu Zhu

September 9, 2020

School of Computer Science and Informatics

Cardiff University

Supervisor: Dr Yukun Lai

A dissertation submitted in partial fulfilment of requirement for MSc (Computing) by taught program

Abstract

Virtual plant is based on computer technology, and involves with botany, ecology, applied mathematics, information science and visualization technique. Virtual plant is a 3D model which is the basis and key of virtual plant research and application. On the one hand, it abstracts the morphological structure, physiological and ecological processes of plants from the perspective of experience. On the other hand, researchers used mathematical methods to analyze the progress of the growth of real plants quantitatively. The topological structure of a plant, occurrence of geometry, rules of growth, feedback mechanism between structure of plants and function are elements which are needed to be applied to visual virtual plants by computer. Using advanced computer technology and theory to build mathematical model provides support for computational models of systems biology and visualization, also predicts and guides the production of agriculture. Therefore, it provides essential research value and profound significance by rebuilding virtual plants in natural and abundant modeling methods.

From the point of botanical classification, plants are divided into herbaceous and woody plants. Whether herbaceous or woody, they have four main ways of branching: monopodial branching, sympodial branching, false dichotomous branching, and dichotomous branching. This dissertation will focus on monopodial branching and sympodial branching of woody plants. Because the difference between monopodial branching and sympodial branching is the topological structure of the trees, the author needs to find a method which can describe the topology of plants clearly. L-system uses strings to describe the topology and geometry of plants. Interpretation of turtle acquire the strings and describe the strings, so L-system can simulate plants topology through controlling strings. In this dissertation, L-system will be applied as the main method.

The specific work is as follows:

1. Use L-system to simulate trees with sympodial branching

By observing plants in nature, the sympodial branching characteristics of trees are summarized. Extract the parameters that control the form of trees, which let the users change topology of trees.

2. Control string to simulate trees with monopodial branching

By observing plants in nature, summarize the characteristics of monopodial branching. Control the production rules to generate monopodial branching trees by L-system.

3. Make a simulation program

A program to simulate two ways of branching will be developed. This program can choose which branching type of tree and can control the topology of tree by inputting different production rules.

Acknowledgement

I would like to thank my supervisor Dr Yukun Lai for his invaluable guidance.

Table of content

Declaration and statement.....1

Abstract.....2

Acknowledgement.....4

***Chapter 1: Introduction*.....9**

1.1 Introduction.....9

1.2 Motivation 10

1.3 Aims and objectives 11

***Chapter 2: Literature review*..... 12**

2.1 Research status.....12

2.2 Research on model of plants branching.....13

2.3 Methods of simulation of plants.....13

2.3.1 L-system.....14

2.3.2 Iterated Function System.....15

2.3.3 Diffusion-limited aggregation.....15

2.3.4 Reference axis technology.....16

2.4 Typical plant model software.....16

2.4.1 AMAP.....17

2.4.2 Xfrog.....17

2.4.3 Other software.....18

2.4.4 Conclusion of existing simulation of software.....18

2.5 Literature review conclusions.....19

***Chapter 3: Problem statement and approach*.....19**

3.1 Problem statement.....19

3.2 Plant topological structure.....19

3.2.1 Branching of plants in natural.....20

3.2.2 Model of branching.....21

3.3 Technology.....21

3.3.1 Visualization technology.....22

3.3.2 OpenGL.....22
3.3.3 DirectX.....22
3.3.4 Conclusion of technology selection.....23

Chapter 4: Main method and implementation.....23

4.1 The definition of L-system.....23
4.1.1 Explanation about plane geometry of L-system.....25
4.1.2 Explanation about 3D geometry of L-system.....26

4.2 Classify of L-system.....27
4.2.1 DOL-System.....28
4.2.2 Random L-system.....29
4.2.3 Context-sensitive L-system.....31
4.2.4 Parameter L-system.....32

4.3 The construction of sympodial branching trees.....34
4.3.1 The concept of sympodial branching trees.....34
4.3.2 L-system construction of sympodial trees.....36

4.4 The construction of monopodial branching trees.....40
4.4.1 The symbol of monopodial trees.....40
4.4.2 Construction of monopodial trees by L-system.....41

4.5 Experimental simulation of random L-system.....42
4.5.1 Choice of development tools.....42
4.5.2 The design idea.....43

4.6 Implementation of the system.....45
4.6.1 An overview of the software.....45
4.6.2 Perspective.....47
4.6.3 Growth progress.....48
4.6.4 Details of control.....49
4.6.5 Generate different trees.....51

4.7 The summary of this chapter.....53

Chapter 5: Conclusion and further work.....53

5.1 Conclusion.....53

5.2 Further work.....55

5.3 Future trend..... 55
Chapter 6: Reflection..... 56
Reference..... 59

List of Figures

Figure 1: Virtual plants generated by L-system..... 14
Figure 2: Virtual plants generated by IFS..... 15
Figure 3: Virtual plants generated by DLA..... 16
Figure 4: Axial tree..... 20
Figure 5: Shapes generated by L-system..... 24
Figure 6: Turtle interpretation..... 26
Figure 7: Control of turtle in three dimensions..... 27
Figure 8: Virtual plant generated by DOL..... 29
Figure 9: Virtual plant generated by random L-system..... 30
Figure 10: Interpretation of parameter L-system..... 33
Figure 11: Axial tree..... 34
Figure 12: Branching angle..... 35
Figure 13: Three branches on the internodes..... 38
Figure 14: Two or three branches on the internodes..... 39
Figure 15: Comparison of tree with different angles..... 40
Figure 16: Monopodial tree..... 42
Figure 17: Flow chart..... 44
Figure 18: The user interface..... 46
Figure 19: The page for inputting parameters..... 47
Figure 20: Observe the tree from the front..... 47

Figure 21: Observe the tree from the top 48
Figure 22: The tree iterates once 48
Figure 23: The tree iterates twice 48
Figure 24: The tree iterates three times..... 49
Figure 25: The tree iterates four times 49
Figure 26: Comparison between same tree with different angles 50
Figure 27: Comparison between same tree with different diameters 50
Figure 28: Comparison between same tree with different rates 50
Figure 29: Comparison between same tree with different iteration times 51
Figure 30: Comparison between two trees by different rules 52
Figure 31: Comparison between two trees generated by Random L-system..... 52

List of tables

Table 1: Different characteristics of branches..... 13
Table 2: The progress of iteration of DOL-system..... 28
Table 3: The progress of iteration of context-sensitive L-system 31
Table 4: The proportion of samples by branch series..... 35

Chapter1. Introduction

1.1 Introduction

With the rapid development of information technology, information science has provided strong technical support for the scientific research of various natural disciplines and the application of scientific and technological achievements, especially in the aspects of information acquisition, storage, transmission, processing and control. The integration of information science and biological science have become an effective way of scientific and technological innovation. With the powerful information processing power of computer, bioinformatics research has developed rapidly, which makes it possible to reveal the essential structure of life from the molecular level. With the further integration of information science, biological science and agricultural science, computer virtual technology and simulation technology have been widely applied in other areas.

There are numbers of plants in natural which are closely related to people. With the development of computer technology and the widespread use of 3D technology, the topological structure and geometrical morphology of plants are becoming hot research spot. Lots of scientists are committed to research different plant models and plant theories (Prusinkiewicz, Hanan and Měch, 2000).

Virtual plants cover various areas, such as computer science, crop culture, microbiology and botany. Using computer science to rebuild morphological structure and growth process of plants can help researchers to study the different factors affecting plant growth, reducing growth time, and saving costs.

In general, Virtual plant morphology is studied from two aspects, one is topological structure of a plant(Godin, Costes and Caraglio, 1997), and another is geometry (Zapata-Grajales, Cano-Velásquez and Villa-Ochoa, 2018).

Topological structure is an important symbol for plants, and every plant has its own structure. Topological structure is the morphological structure of plants in three-dimensional space from different angles such as self-similar

topology and morphological structure. The geometry of plants is the structure and morphology of plant organs, for example, blade shape and fruit size. The research about physiological ecology of plants and Internal growth rules make a big difference to geometric structure of plants. The aim of the research is committed to agricultural. There are some methods to simulate structure of plants, such as L-system(Lindenmayer, 1968), Fractal method (Tatsumi, Yamauchi and Kono, 1989), Iterated Function System [IFS] (Demko, Hodges and Naylor, 1985), Diffusion Limited Aggregation [DLA] (T.A.Witten, 1983) and others.

L-system is a system of parallel rewriting of characters which was proposed by Lindenmayer in 1968. It uses turtle geometry to get grammar and describe grammar to draw plants. Fractal method is based on fractal theory to simulate structure of plants which is suitable to construct plants with self-similar feature. IFS (Iterated Function System) is a method to construct structure of plants which need to be compressed, affined, transformed and iterated for several time. However, it is a big problem for IFS how to get IFS code for Iterative function which stop this method cannot be applied by some researchers. Moreover, DLA (Diffusion Limited Aggregation) is used to simulate fractal growth and agglomerate phenomenon. In a word, all of them are useful to construct virtual plants and inspire researchers.

As mentioned earlier, each method has own advantages and field. The simplest L-system is called DOL system, because there is just one rule in any set of characters which makes the virtual plants have no change. Furthermore, the random L-system takes the probability factor of the rule into consideration when it chooses to rewrite the rule which make virtual plants different. Thus, this dissertation chooses L-system to model virtual plants.

1.2 Motivation

The growth of plants is a complex process, because of region, season and biology. It is hard to model and predict for growth of plants. In recent years, researchers use computer technology to build the model of products or models of progress before production of products and conducted analysis of products to find problems in advance. In this way, developers can reduce the time of actual product development and increase the probability of

one-time success of products. Moreover, using computer technology can shorten the cycle of product development and reduce production costs, so it is important and meaningful to model virtual plants which can be used in various disciplines.

Relying on computer to simulate the growth of plants is modeling the structure of plants, progress of growth and progress of production. Developer can use computer design plants, plan plants, predict products and model life cycle of plants. Thus, it is crucial to simulate systems of plant growth which are based on computer technology.

1.3 Aims and objectives

This dissertation is based on L-system to do the research. Considering the characteristics of L-system modeling plants and combining other methods to build a model of L-system. The following objectives are identified:

(1) In-depth research and analysis of normal plant modeling methods

Firstly, summarize several modeling methods and theories of virtual plant morphology. Secondly, compare several methods and get the advantages and disadvantages of different methods. Thirdly, make conclusions of the advantages of L-system.

(2) Research about plants branching model

There are many different patterns of plant branching. Plants such as poplars have tall and straight trunks, and their lateral branches grow less than the trunks. The branching pattern of these plants is monopodial branching. Additionally, when the trunk of a pear tree grows to a certain height, it is replaced by lateral branches, and the crown of the tree is unfolded. The branching pattern of this kind of plant is the sympodial branching. By observing these plants, this dissertation summarizes the branching patterns of several typical plants.

(3) Simulate two types of trees by L-system

In this dissertation, several types of L-systems are studied. Considering the morphological structure of various branching patterns of plants, random L-systems are used to express different branching patterns of trees. The basic properties of branch structure are parameterized to extract several

important parameters to control the branch shape of trees.

(4) Experimental simulation
Develop a user GUI through Tkinter to show the results.

Chapter 2: Literature review

2.1 Research status

Ulam (1962) started to do research about branching models of plants which is the origin of virtual plants. After that, Lindenmayer (1968) put forward L-system which became one of methods to learn about simulation of plant morphology. Then, based on the progress and application for L-system by Prusinkiewicz (1986), L-system had become the most popular method for simulation of plant morphology. Additionally, according to Self-similar characteristic of plants, Mandelbrot (1982) put forward the idea of classification geometry which was been developed and applied in the world not after long. In 1981, Witten (1983) put forward the DLA. The particles move in random Brownian motion in a certain boundary region and become to a certain form of plant. Besides, Reeves (1983) put forward particle systems which is used to simulate grass and woods. The simulation of the system is used to show fuzzy plant morphology. In 1984, Aono and Kunii (1984) put forward A-system. This method is based on three-dimensional structure of a tree and can set some characteristic parameters of plant branch structure. So, this method can simulate numbers of plants vividly. Moreover, a method called Language-restricted IFS (Prusinkiewicz and Hammel, 1991) had been put forward by Prusinkewiz and Hammel which can add constrains on the order of change to generalize different IFS methods. In 1988, reference axis technique had been put forward (Jean, de Reffye and Daniel, 1997). De Reffye and other authors also developed software called AMAP which is used to construct different structure of plants.

Until the end of the 20th century, it appeared some models to simulate the structure of plants, such as LIGNUUM (Perttunen and Sieva, 1998). Enter the 21st century, L-system was developing rapidly and recognized. The University of Queensland developed a software which was based on L-

system called Virtual Plants and used the software to simulate the growth of soybean, cotton, and other kinds. With the rapid growth and widespread of Internet, VRML (Taubin *et al.*, 1998) appeared as a computer virtual language. This language can be used to reconstruct model of three-dimension of plant and can be used in designing of three-dimension of Internet.

2.2 Research on model of plants branching

Branching structures are important factors to determine the structure of plants. The structure of a branch depends to a large extent on the relationship between the terminative vertex and the continued vertex. Continued vertexes grow new branches, while terminative vertexes grow fruit, flower and another organ. In addition, there are four pattens which include terminative mode, multiple mode, monopodium mode and sympodium mode.

Table 1: Different characteristics of branches

Branch			
Main vertexes stop growing		Main vertexes continue to grow	
All side vertexes stop growing	Some side vertexes continue to grow	All side vertexes stop growing	Some side vertexes continue to grow
Terminative mode	Sympodium mode	Monopodium mode	Multiple mode

The branching modes mentioned above can be described by L-system easily. A, B and C represent continued vertexes. X is a terminative vertex, I is a internode. [and] represent the appearing of branch. Terminative mode and sympodium mode can be represented by the rule below:

$$A \rightarrow I[B]^n[X]^m \mathrm{X}$$

For the terminative mode, $n = 0, m > 0$; For the sympodium mode, $n > 0, m > 0$. Monopodium mode and multiple mode can be represented by the rule below:

$$A \rightarrow I[B]^n[X]^m C$$

For the monopodium mode, $n = 0, m > 0$; For the multiple mode, $n > 0, m > 0$.

2.3 Methods of simulation of plants

There are various methods to simulate plants and lots of factors to infect

the topological structures of plants. It is impossible to set a model to simulate the progress of plants growth, but researchers used large amount of data to simulate plants as real as possible. This part will introduce several different methods which were put forward by some researchers and compare several methods of simulating plants.

2.3.1 L-system

L-system which is the most successful method to simulate plants that was put forward by a biologist named Lindenmayer. Also, it is a progress of iteration whose core concept is rewriting (Lindenmayer, 1968). Moreover, L-system uses strings to describe the topology and geometry of plants. Interpretation of turtle acquire and describe the strings. It is useful for researchers, because L-system can directly describe topological structure of plants and has fractal characteristics.

Prusinkiewicz, Hammel and Mjolsness (1993) raised differential L-system(dL-system) which uses differential equation to simulate the continuous progress of plants growth. In 1996, Měch and Prusinkiewicz (1996) put forward OPEN L-system. There are several characteristics about the system, such as controlling, adjusting the exchanges of information between plants and environment. Prusinkiewicz (1997) put forward a timed L-system which was used to simulate the dynamic progress of plants growth. Further, Przemyslaw Prusinkiewicz, Radoslaw Karwowski, Radomír Měch (1999) based on L-system to extend and improve it. They developed a software called L-Studio which was to simulate plants and ran under windows system. They have made outstanding contribution to L-system.

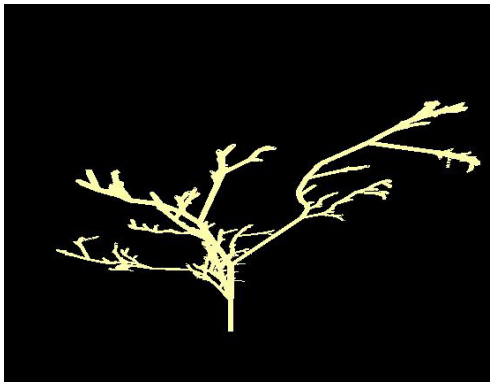


Figure1: Virtual plant generated by L-system

2.3.2 Iterated Function System

Iterated Function System (IFS) is important for fractal geometry, and it was put forward by Hutchinson (Hutchinson, 1981). Michael F Barnsley (1985) brought forward a complete set theory of Iterated Function System. The core concept of IFS is affine and transformation. Large amounts of objects in natural have self-similar characteristics, so structures of objects are affined and transformed until getting fractal graphics. The core concept of using IFS to simulate plants is that pictures of original plants choose functions to be iterated whose probability have been got.

The definition of IFS: $IFS = \{(w_i, p_i) | i = 1,2,3.....,n\}$. The w_i is a set of functions which are satisfied some conditions and p_i is a set of probability. IFS use probability of transformation to choose function to generate graphics. Using simple data to simulate graphics is advantage for IFS.

It is hard to get IFS which is fatal flaw for IFS. If researchers want to get IFS codes, they need to use graphic interaction technology to try to get IFS code. Plants simulation of IFS is concentrating on the gray graphics, so how to use color is also disadvantage for IFS. The plants simulated by IFS is similar and have less variability because IFS is lack of controlling of topology of plants.

Figure 2: Virtual plants generated by IFS (Laug, 2019)

2.3.3 Diffusion-limited aggregation

Diffusion-limited aggregation (DLA) was put forward by T.A.Witten (1983).

The main idea of DLA is as follows. At the first stage, choose an agglomerate point in plane grid as initial point, then release a point in the plane grid where is far from initial point. When the point moves to the agglomerate point nearly, change the state of the point to be agglomerate point. If the point moves to the border of the grid, the point must be disappeared. By repeating the steps above, we can get a set of agglomerate points. DLA is normally used to simulate fractal growth and agglomerate phenomenon. DLA is a model of random growth of plants and is used to describe the progress of dynamic plants growth.

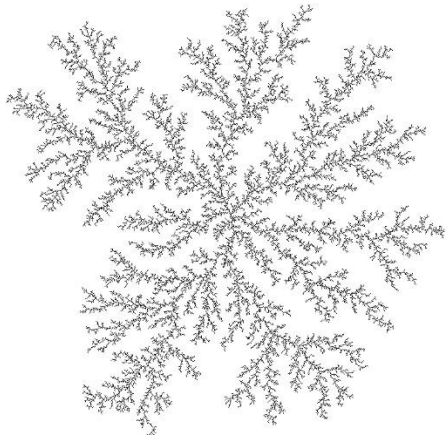


Figure 3: Virtual plants generated by DLA (Bourke, 1991)

2.3.4 Reference axis technology

Reference axis technique is a typical stochastic process approach which was put forward by Philippe de Reffye and other researchers. In this method, the structure of plants is firstly summarized into more than twenty basic models, and the basic model suitable for the simulated plant structure is determined through qualitative analysis. Then, markov process and state transition diagram are used to dynamically simulate the meristematic tissues of plants, such as the occurrence of buds, lateral branches, death, dormancy, and other processes. CIRAD develops AMAP software to simulate plant growth, analyzes the evolution rules of plant topological structure based on the actual biological data measured, extracts the growth rules through pattern recognition, and uses geometric methods to express the morphological rules of plants, which is widely used in the construction of various plant models.

2.4 Typical plant model software

Due to the different fields, the functions and implementation methods of plants simulation software are also quite different. The simulation software can be divided into two types: graphics software and agricultural research software. Graphic software focuses on computer graphics, however, this software uses as little botanical knowledge as possible to simulate virtual plants quickly, such as: Xfrong, Plant studio, Virtual Gardening, and others. Comparing to the graphics software, another kind of software is characterized by combining the dynamic structure model of plants and simulating the natural growth law of plants quantitatively according to the actual change law in the process of plant growth, for instance, AMAP, GOSSYM, CottonPlus and Virtual Plants.

2.4.1 AMAP

AMAP is called Arctic Monitoring and Assessment Programme (Jaeger and De Reffye, 1992) which is used to simulate plant structure and progress. This software combining plant growth model with visual model is suitable to simulate tall plants.

AMAP includes several subsystems: AMAPara, AMAPhydro, AMAPmod and AMAsim. The AMAPara is used to simulate the influence of environment on plant growth. It uses reference axis to describe plant topology and can simulate the collision of branches as a plant growth. And AMAPhydro construct Hydraulic model of plants which simulates the production and distribution of nutrients in plants. AMAPmod is used to establish the growth and development model of specific plants based on actual measurement data. Additionally, AMAPsim is used to simulate the plant growth progress. The AMAP is mainly used in landscape design.

2.4.2 Xfrog

Xfrog (Lintermann and Deussen, 1999) was developed by Lintermann from University of Karlsruhe. This software uses interactive modeling of plants based on function graph which is modified to construct the plant as a whole branch unit. Xfrog defines four graphs and each of graph represents different parts of plant. Here, the first graph is to generate plant organs, such as leaf and petal. The second graph is to generate the geometric distribution of branches. And the third graph represents global variable and can define the environmental factors influencing plant simulation, such as

lighting and gravity field. The fourth graph represents functions which can decide that the next step is to grow branches or petals. The users just need to click the corresponding graph and the system can construct real virtual plants. The biggest advantage of Xfrog is that users can generate virtual plants through human-computer interaction. It is simpler than L-system because there are no complex strings, however, the Xfrog has disadvantages. Although this software can simulate the appearance and structure of plants, it is not suitable to simulate the plant growth progress.

2.4.3 Other software

Plant Studio (2000) was developed by Kurtz-Fernhout. It applies graphics to represent branching structure, internode, phyllotaxis, inflorescence, flower, leaf and others. The users who are not equipped with knowledges about botany can input parameters into computer to get virtual plants easily. There are several file formats for Plant Studio, such as 3DS, DFX, WRL, OBJ, POV and so on. It is convenient for users to construct virtual plants. Besides, Tree Professional (1993) was developed by Onyx Computing. It is a plant simulation software which can generate plants quickly. The users just input the plant species, plant ages and other random parameters to generate plants through plant graphics library and leaf library. Compared with Plant Studio, Tree Professional is more real and convenient. The University of Calgary develop CPEG, L-Studio and Virtual Laboratory which are based on L-system. This software was used to simulate plant structure, botany teaching and the study of the reproduction of extinct tree species.

2.4.4 Conclusion of existing simulation of software

The simulation of virtual plants has developed significantly over past three decades but the impact of virtual plants on humans is just beginning. The plants shape simulations are becoming mature but the software which need to be applied to agroforestry remain to be further studied, such as progress of plant growth.

The simulation of plant roots is the weakest link in the study of virtual plants. The root system is an essential medium for the interaction of matter and energy between plants and the environment, so it is important to strengthen root system research. It can be considered that the research of root system will become a crucial research direction of virtual plants in the

future. In addition, the traditional agricultural expert system is based on the experience of agricultural experts and simulates the process of reasoning, judgment and decision measurement by means of knowledge base and decision reasoning system. However, it is difficult to make accurate and comprehensive decision reasoning by relying solely on the empirical knowledge of agricultural experts. Hence, it is of great significance to study how to improve the decision-making ability of agricultural expert system by using the dynamic model of virtual plants, and to show the decision-making results of the expert system visually.

2.5 Literature review conclusions

As mentioned above, there are several methods to simulate virtual plants, but it is hard to tell which method can construct plants more real and whose methods are easier. For the IFS, it is lack of controlling plants topology and the structure of plants is similar. For the DLA system, it cannot describe plants topology very well. In this project, it is needed to construct real and clearly structured plants, so L-system is the modest choice to use.

Chapter 3: Problem statement and approach

3.1 Problem statement

The growth of virtual plants is a progress of simulation of plants in computer which is an important tool for researchers. The simulation of plants growth and visual simulation have wide application prospect to other disciplines, such as agroforestry, landscape design, virtual reality, computer animation and others (De Reffye *et al.*, 1997). The existing research fields will be strengthened and new ones will be opened up(Room, Hanan and Prusinkiewicz, 1996). However, it is a problem that how to generate virtual plants as real as possible. There are three steps to simulate virtual plants.

The first step is simulating plant topological structure. The second step is the structure of plant organs and organs spatial orientation. And the final step is the realization of visual system.

3.2 Plant topological structure

The branch is a symbol of plants, and the growth of branches decides the

structure of plants. The simulation of plants is the simulation of basic skeleton of plants and is an essential of simulation of plant growth. Plants can be divided into herbaceous and woody plants and their structures of branches are diverse. Whether herbaceous or woody, there are four ways of branching: monopodial branching, sympodial branching, false dichotomous branching and dichotomous branching (Wang *et al.,* 2013). This dissertation focuses on monopodial branching and sympodial branching.

3.2.1 Branching of plants in natural

Herbaceous plants are the plants whose stems are less woody; woody plants are the plants whose stem are with large wood. Moreover, woody plants can be divided into arbor and bush. The arbors are woody plants whose bodies are tall. Bushes are woody plants with no obvious trunk and a clumpy appearance. Weather woody plants and herbaceous, their branches are different. This dissertation is focused on arbors (tree).

Diverse trees have different branches. Even if the same kind of tree, their branches diverse from each other. Most trees stop branching when they grow to a certain point, but a few species of tree have many changes when they branch to the end, such as willow. The simulation of willow is complex, this dissertation does not do research to this kind trees.

The method of arranging branches for an axial tree is used to describe branches of tree (Gravelius, 1914), and the definition of the axial tree is as follows:

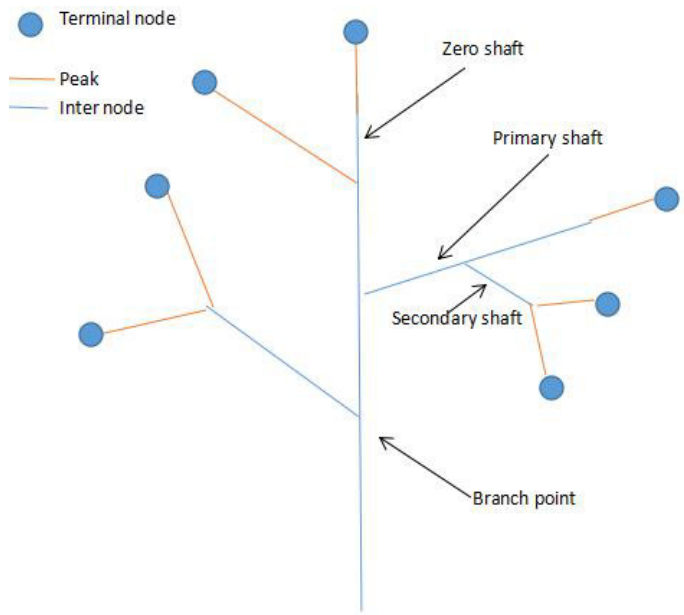


Figure 4: Axial tree

As can be seen from Figure 4, the main trunk is called zero shaft; the lateral branch grows from main trunk is called primary shaft; and the lateral branch grows from primary shaft is named secondary shaft.

In addition, most branches of trees can be divided into small units. Zero shaft and primary shaft can be seen as a unit; primary shaft and secondary shaft can be seen as another unit. Each unit is similar with each other and is similar with integral shape of tree, which has laid the foundation for the theory.

3.2.2 Model of branching

It is needed to understand two concepts: the apex of tree which grows main trunk; lateral apex of tree which grows lateral branch.

Monopodial branching represents that main apex grows to the main trunk and the lateral apexes grow to thinner lateral branches. Besides, the plants with monopodial branching whose main branches growth ability is stronger than lateral branches. Most of these plants have obvious main branches and are tall and straight. There are several trees with monopodial branching, such as pine tree, aspen, spruce and other kinds.

Compared with monopodial branching, sympodial branching is that the main apex grows to a certain degree and dies or stops growing, then lateral

branches replace the main branch. The plants with sympodial branching that main branches grow to a certain height and are replaced by lateral branches, the lateral branches growing to certain degree will be replaced by next level branches. Thus, this kind of trees can form curved main branches, which enhances the support ability of plants. There are several plants with sympodial branching, for instance, apple trees and jujube.

3.3 Technology

In order to research plants growth rules, numbers of researchers started to simulate plants growth since 1960s. Through the model of simulation of plants growth, they could predict synthesized indicators about plants in different environments, such as crop yield. Reffye (1998) simplified the influence of plants structure on plants growth, which was significant for agricultural production, but this simulation could not satisfy the requirement of other disciplines.

The technology is important for the research about plants structure. In the past, there were no technology for researchers to use. Because simulations of plants need powerful ability of computer and graphic, which had not appeared. Through the growth of computer technology, computer hardware and 3D digital technology can satisfy the requirements of plants simulation. The visualization technology and object-oriented programming are becoming more and more mature. In a word, the visualization technology is the most important technology to simulate virtual plants.

3.3.1 Visualization technology

In real life, the researchers could get large amounts of data through technology of data collection. It is hard for researchers to analyze the data. In the 1980s, the visualization technology had been put forward which could translate the experimental data and calculate data into graphs. Due to the help of visualization technology, researchers could find the scientific laws in data directly. So far, there are several tools for researchers to use. It is a problem to choose which tools to visualize virtual plants, however, the widely used technologies are OpenGL and DirectX.

3.3.2 OpenGL

The OpenGL (1992) is a program interface and is a powerful and convenient

to call its underlying 3D graphics library. The precursor of the OpenGL is IRIS GL which was developed by SGI. Here, the IRIS GL is an industry standard 3D graphics interface with powerful function, but its portability is poor. So, SGI developed OpenGL based on IRIS GL.

In addition, the OpenGL is not related to hardware. It can be transplanted to different platforms, such as Windows 95, Windows NT, Unix, Linux, MacOS and others. The software which can be transplanted by OpenGL can be applied widely.

3.3.3 DirectX

The DirectX (1996) is an application programmers' interface. It can make Windows-based games and multimedia programs for more efficient execution. Moreover, the DirectX is not only a graphic API because it has several modules: Direct Graphics, Direct Input, Direct Play, Direct Sound, Direct show, Direct Setup and Direct Media. It is convenient for developer to make games. And the Direct Graphics is responsible for the graphics.

3.3.4 Conclusions of technology selection

The OpenGL is the most popular program interface since 1992, developer can use it to develop several application programs. Additionally, the OpenGL has functions of texture mapping and render effect, which is useful to generate virtual plants. Here, taking generating virtual plants as the starting point, the OpenGL has the advantages as below:

1: The quality of graphic is high, which makes virtual plants more real.
2: The stability of OpenGL ensures that the developed programs will not lose efficacy due to compatibility between the higher version and the lower version.
3: The graphic software based on OpenGL is not related to hardware, which means that application developed by OpenGL can be ran in any hardware that meets with OpenGL API.

Chapter 4. Main method and implementation

4.1. The definition of L-system

Lindenmayer system referred to as L-system was put forward by a biologist whose name was Lindenmayer in 1968. L-system is a progress of iteration, and its core concept is rewriting. Besides, L-system uses rules of rewriting or a set of production to replace part of simple initial target whose aim is to define complex target. In L-system, production replace all characters in character strings. The following diagrams (see Figure 5) are simple living example of application of L-system.

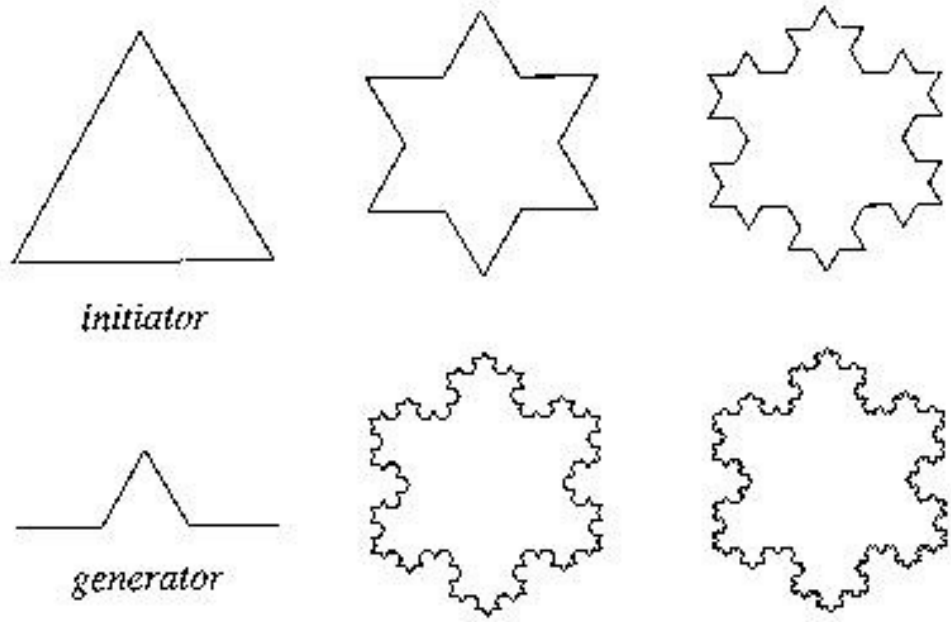


Figure 5: Shapes generated by L-system (Prusinkiewicz P. and Lindenmayer, A, 2004)

The first step is that we need to get initial graphics and rule of rewriting. The rule is formed by broken lines which is an equilateral graphic, and the length of the rule is r^n. Further, the equilateral graphic called model of rule is used to replace each side of initial graphic. After that, the rule of rewriting has the same endpoint with interval. Finally, after four-times iteration, the initial graphic becomes to the shape of snow.

The L-system formal language is based on characters. Assuming the initial string is A and the production rule is as follows:

$$A \rightarrow aB$$
$$B \rightarrow bA$$

After the first derivation, the first rule of generation is satisfied with the

condition, A is replaced by aB and the result is aB. Then, after the second derivation, the character B is satisfied with the second rule, and the result turns out abA. By the analogy above, we can get $abaB$ from abA. If we continue to infer, we can get $ababA$, $ababaB$ and goes on and on.

4.1.1 Explanation about plane geometry of L-system

To simulate more realistic virtual plants, we need a mature method of graphics to explain the string which are obtained previously. Prusinkiewicz put forward a method called turtle interpretation (Prusinkiewicz and Hanan, 1997) to explain. In the plane, the state of turtle is defined by ternary array (x, y, z), the cartesian coordinate (x, y) represents the position of the turtle, z represents turtle' direction of advance. Given the stride length d and the increment angle δ, the turtle shape corresponds to the following order:

F: The turtle moves one step and the stride length is d, then the position of turtle turn to (x, y, z), where $x_1 = x + dcos * z,\ y_1 = y + dsin * z$. Draw line between (x, y) and (x_1, y_1)

f: Move forward d. Do not need to draw line.

$+$: Rotate δ left, turn the current direction δ counterclockwise, take it as the new current direction, the turtle shape turns to $(x, y, z + \delta)$

$-$: Rotate δ right, the turtle shape turns to $(x, y, z - \delta)$

[: put current state of turtle onto stack.

]: pop current state of turtle out of stack.

There is a character string showed below: F[-F[-F-F]FF-F-FF]FF-F-FF

The initial direction of turtle is 90°, the increment angle is 90°, stride length is d (length of grid). The route of the turtle is showed below:

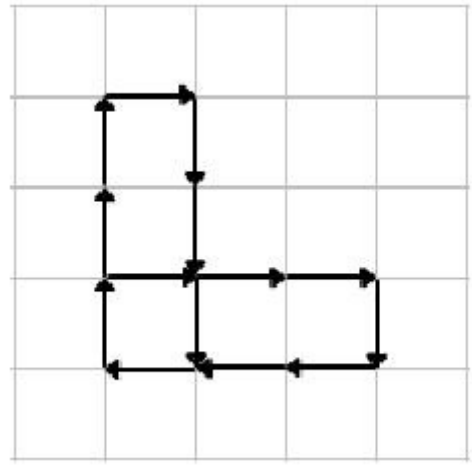


Figure 6: Turtle interpretation (Prusinkiewicz P. and Lindenmayer, A, 2004)

4.1.2 Explanation about 3D geometry of L-system

Abelson and DiSessa (1992) extended explanation of geometry of L-system to three-dimension. The main concept is that the position of turtle uses three vectors $[X, Y, Z]$ to represent the path. X represents the direction of movement; Y represents the direction of turning left and Z represents the direction of turning up. Three vectors are orthogonal for each other and they are all unit vectors which are satisfied the function: $X * Y = Z$. The function below can represent the rotation of turtle:

$$[X1, Y1, Z1] \ = \ [X, Y, Z]\, R$$

The R is 3*3 rotation matrix. When the rotation angle is α, the chart about rotation matrix revolving X, Y, Z is showed below.

$$R_x\,(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$R_y\,(\alpha) = \begin{bmatrix} \cos\alpha & 0 & -\sin\alpha \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos\alpha \end{bmatrix}$$

$$R_z\,(\alpha) = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Additionally, the direction of rotation about turtle is showed below:

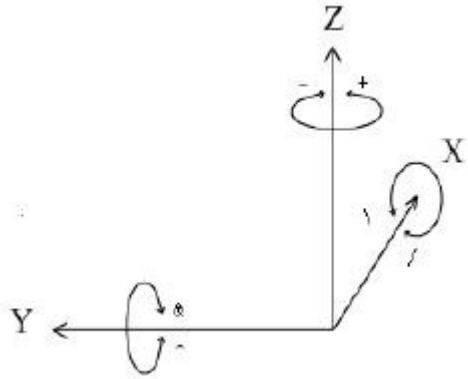


Figure 7: Control of turtles in three dimensions (Prusinkiewicz P. and Lindenmayer, A, 2004)

$+$: Rotate α degrees to left and apply rotation matrix $R_z\ (\alpha)$;

-: Rotate α degrees to right and apply rotation matrix $R_z\ (\alpha)$;

&: Rotate α degrees to down and apply rotation matrix $R_y\ (\alpha)$;

^: Rotate α degrees to up and apply rotation matrix $R_y\ (\alpha)$;

\: Rotate α degrees to left and apply rotation matrix $R_x\ (\alpha)$;

/: Rotate α degrees to right and apply rotation matrix $R_x\ (\alpha)$;

Through the methods mentioned above, the L-system can draw graphics in three-dimensional space.

4.2 Classify of L-system

There are several L-systems which are based on the traditional L-system. The common L-system is DOL-system, Random L-system, Context-sensitive L-system and parametric L-system. In the following, each L-system will be introduced respectively.

4.2.1 DOL-System

The DOL-system is the simplest L-system. Each of production rule is decided which is corresponding to the characters in string and the progress of rewriting characters is irrelevant to the semantic environment. The DOL-system can be represented by ordered triples:

$$G =< v, w, p >$$

In the triple, v represents system alphabet, w represents the initial string which cannot be empty, and p is a set containing several production rules. A rule of rewriting $(a, b) \epsilon P$, we record $a \rightarrow b$, character a represents the front of rule of rewriting, character b represents the end of rule of rewriting. When a meet the production rule, b replaces a. The example can be seen in the following:

$$\omega: A_i$$

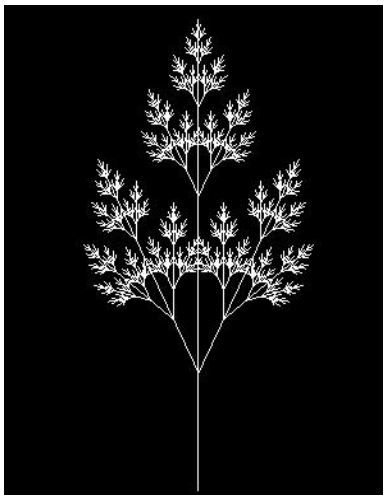
$$P1: A_i \rightarrow A_j B_i$$

$$P2: A_j \rightarrow B_i A_i$$

$$P3: B_i \rightarrow A_i A_j$$

Table 2: The progress of iteration of DOL-system

Initial character	A_i
Iterate once	$A_j B_i$
Iterate twice	$B_i A_i A_i A_j$
Iterate there-times	$A_i A_j A_j B_i A_j B_i B_i A_i$
……	……

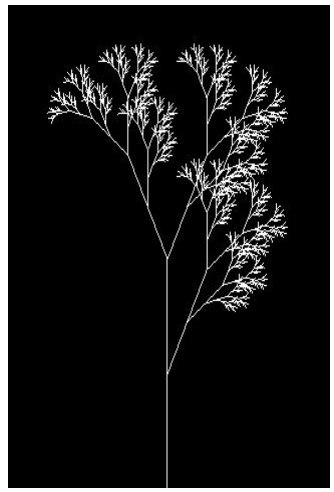


$n = 6, \$ = 25°, l = 1$

X

$X \rightarrow F[+X][-X]FX$

$F \rightarrow FF$

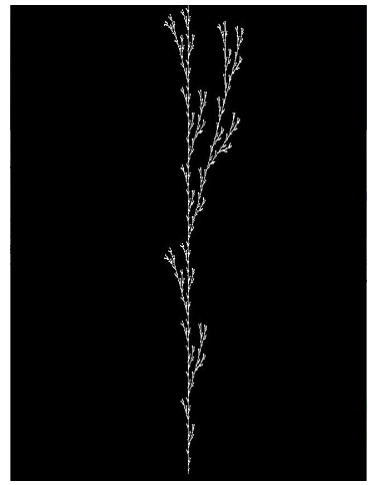


$n = 6, \$ = 25°, l = 1$

X

$X \rightarrow F[+X]F[-X]+X$

$F \rightarrow FF$



$n = 6, \$ = 25°, l = 1$

F

$F \rightarrow F[+F]F[-F]F$

Figure 8: Virtual plants generated by DOL

In the graphics above, character n represents the times of iteration, $ represents the angel and l represents the length. The DOL-system is simple and practical, and the structure of rules is convenient. However, the DOL-system is decided. Thus, the graphics which is formed by DOL-system is simplex and stiff. The virtual plants formed by DOL-system cannot be real, but the rules are simple and easily understood which are applied in various domain.

4.2.2 Random L-system

The virtual plants constructed by DOL is decided and the structure of plant is same. For changing details of virtual plants, random L-system has more advantages than DOL. The random L-system can be described by ordered quad $G\pi = < V, W, P, \pi >$. Here, V represents system alphabet, W represents the initial string which cannot be empty, and P represents a set

containing several production rules. Comparing with DOL-system, here the π represents a mapping from production rules set to production probability set. Function π: $P \rightarrow (0,1]$ is probability distribution. Rule of random L-system can be $a \rightarrow b$: π. Character a represents the front of production rule, and character b represents the end of production rule. In addition, the π is the probability of application.

There is an example for random L-system in the following:

ω: F

$p1$: $F \quad F[+F]F[-F]F$:1/3

$p2$: $F \quad F[+F]F$:1/3

$p3$: $F \quad F[-F]F$:1/3

The initial string is F which can be replaced by $p1$ or $p2$ or $p3$ and the probability of application of three rules is same. If we use $p2$ in the first iteration, the result is $F[+F]F$. Also, if we use $p3$ in the second iteration, the result is $F[-F]F[+F][-F]F]F[-F]F$.

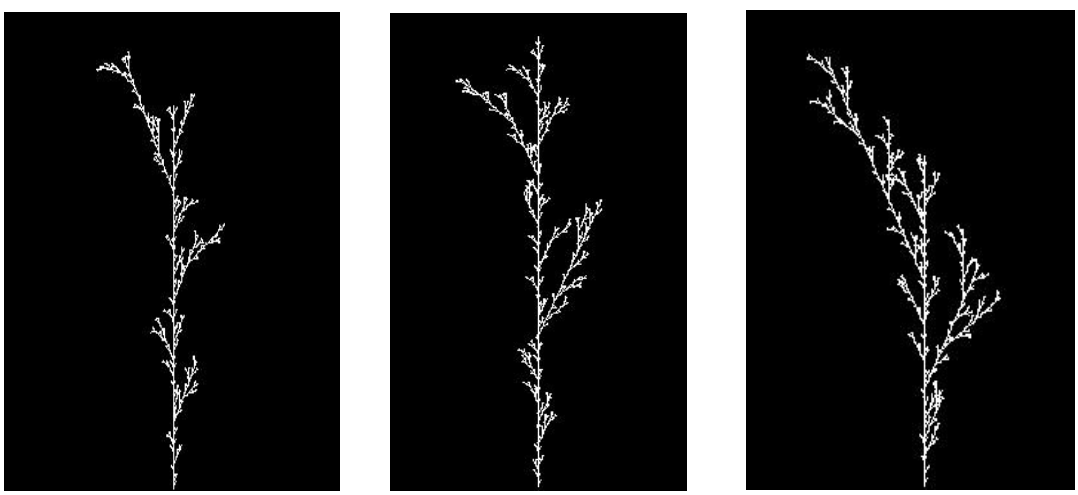


Figure 9: Virtual plants generated by random L-system

The figure above is generated by random L-system iterating for five times, the length is 4 and the angle is 25°. The random L-system is based on DOL

to change probability of production rule to generate virtual plants. Using random L-system can generate plants with same variety but different samples. Furthermore, the Random L-system is useful to calculate the probability of where can grow branches.

4.2.3 Context-sensitive L-system

The Context-sensitive L-system can be divided to 1L-system and 2L-system. 1L-system can be described as $a_i < a \rightarrow b$ or $a_j > a \rightarrow b$. Here, character a can produce b, but the production condition is that a_i is front of a or a_j is behind a. The a_i is on the left of a and a_j is on the right of a. In addition, 2L-system can be described as $a_i < a > a_j \rightarrow b$. Here, character a can produce b, but the production condition is that a_i is front of a and a_j is behind a. Character a_i and a_j is on the left of a and on the right of b respectively.

There is an example for context-sensitive L-system in the following:

ω:$baaaaaaa$

$p1$:$b < a \rightarrow b$

$p2$:$b \rightarrow a$

Context-sensitive L-system iterate the initial characters for several times and the result is below:

Table 3: The progress of iteration of context-sensitive L-system

Initial string	$Baaaaaaaa$
Iterate once	$Abaaaaaaa$
Iterate twice	$Aabaaaaaa$
Iterate three-times	$Aaabaaaaa$
Iterate four-times	$Aaaabaaaa$
……	……

The application of context-sensitive L-system is relevant with front context and back context which affects the simulation of virtual plants. Besides, the persistence of plant growth is difficult to describe, as are some related parameters.

4.2.4 Parameter L-system

Although DOL system, random L-system and context-sensitive L-system can use turtle interpretation to simulate numbers of structure of plants, their simulation ability is restricted, such as plant continue growth, passing single in plants and others. To solve the problem, Lindenmayer put forward a method which combines parameter with the characters in L-system. This method is called parameter L-system.

The parameter L-system is composed with 0L-system and 2L-system. The 0L-system combines digital parameters with decided characters in L-system and can be defined as ordered quad $G =< V, \sum, W, P >$. Here, V is the system alphabet; P is a set of production rules and W is the initial string. They have same meanings with DOL-system. It is worth noting that the $\sum$ is a set of form parameters for parameter L-system which is important to combine DOL. The production rules of 0L-system can be described as $A(x)$:$\{x|x\epsilon X\} > B(f(x))$. The $A(x)$ is precursor of production rule and x is form parameter which needs to be satisfied $x\epsilon X$. And $B(f(x))$ is succeeding whose form parameters are corresponding to form parameter of precursor.

The example for 0L-system is shown below:

$$W: B(2)A(4,4)$$

$$P1: A(x,y):y <= 3 \quad A(x * 2, x + y)$$

$$P2: A(x,y):y > 3 \quad B(x)A(x/y,0)$$

$$P3: B(x):x < 1 \quad C$$

$$P4: B(x):x >= 1 \quad B(x - 1)$$

The progress of iteration can be seen in the following figure:

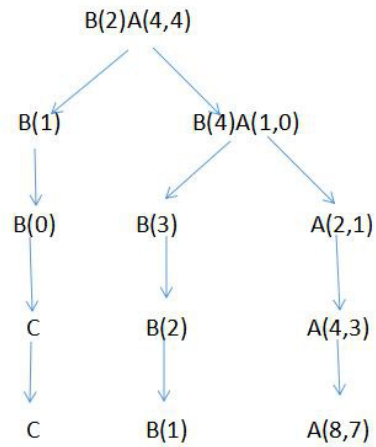


Figure 10: Interpretation of parameter L-system

In comparison, the 2L-system combines digital parameters with context-sensitive. The production rules can be described as: $B(y) < A(x) > C(z)$:$\{x,y,z|x\epsilon X, y\epsilon Y, z\epsilon Z\} \rightarrow E(f(xy,z))$. Each part of precursor of 2L-system is combined string containing parameters from system alphabet with $\sum$ containing formed parameters. Hence, the 2L-system can simulate the information exchanges between modules which are near.

The example for 2L-system is as follows:

$$W: B(3)A(4)C(5)$$

$$P1: B(x) < A(y) > C(z): x + y + z > 8 \quad E(2x + y)F(x * z)$$

Here, A is B's precursor because W is satisfied with production rule $P1$, and the succeeding is C. After calculation, A (4) is replaced by $E(10)F(15)$, and the result is $B(3)E(10)F(15)C(5)$.

4.3 The construction of sympodial branching trees

4.3.1 The concept of sympodial branching trees

To explain feature of plants branching, there are three concepts of tree branching.

(1) Level of branching

The level of branching is an important parameter for simulation of iteration, which is significant to simulate plants.

Due to the description above, a branch that grows directly from the trunk is the primary shaft; a branch that grows from a primary branch is the secondary branch, and so on. The largest level of branch is defined as the branching level of a tree. As the figure below, it can be known that the largest branching level is 3.

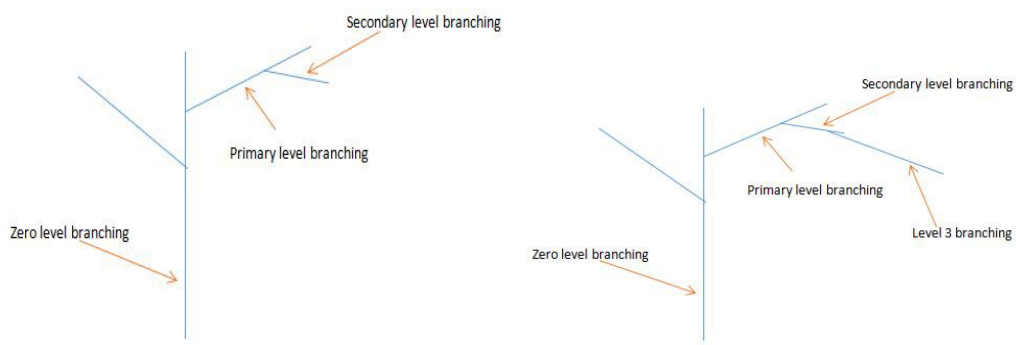


Figure 11: Axial tree

In order to generate a more real tree, this dissertation statistics and analyses one hundred branching level of trees. The largest branching level

is 7 and the level 6 and level 7 are majority of samples. Besides, the level 5 and level 4 are less. The details are as follows:

Table 4: The proportion of samples by branch series

Branching level	Number of trees	The proportion of samples
4	9	9%
5	15	15%
6	30	30%
7	46	46%

(2) Branching angle

In addition to the effect of branching level on branching characteristics, the branching angle is also an important factor to trees.

The branching angle refers to the size of the branching angle between the branch and the mother branch, as the Figure 13 shows below. It is a crucial index to measure the spatial distribution ability of trees and plays a decisive role in the formation of canopy. Moreover, the larger branching angle, the vertical projection of the tree to the ground is larger. The angles of most sympodial branching trees are 30°, 45°,60°and 90°, and several branching angles can occur on the same tree.

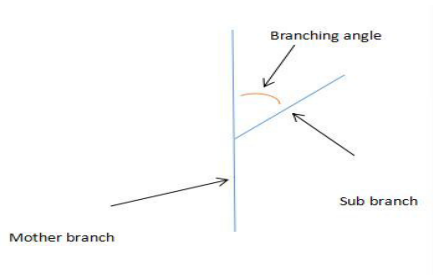


Figure 12: Branching angle

(3) Number of branches in internode

The number of branches of sympodial trees is different for each stage. In terms of trees in general, the number of branches in the first internode is

two, the number of branches in secondary internode is four and the number of branches in third internode is eight and so on. Usually, the number of branches increases exponentially, but it is not absolutely.

4.3.2 L-system construction of sympodial trees

(1) Framework of sympodial trees

In most cases, the crown of a tree with sympodial branching pattern is unfolded and the trees have no main trunk. The trunk of sympodial trees is made up of many branches. Additionally, the length of branches is gradually shortened, and the diameter of the branches is also gradually reduced. Due to the salient features, the spreading shape of the crown is related to branching level and branching angle. The larger branching level and branching angle, the greater the canopy expansion. As a result, this dissertation will set the branching angel and branching level as parameters to control iteration and growth directions. According to the characteristics of the length of the branches, the length of branches and diameter of a branch need to be set as parameters.

For tree branching, introduce a bunch of bracketed strings: [and]

‘[’: Push the turtle's current state onto the stack. The information into stack is mainly the current location of the turtle.

‘]’: Pop the turtle’s state as current state.

According to the samples collected, the internodes of sympodial trees have three or two branches. It is needed that a model in which two and three branches appear randomly in each internode. Therefore, the author sets up two production rules: one for producing three branches, and another for producing two branches.

Axiom(w): ‘X’

Production rules(P) $p1$: ‘$F - [[@X@]\# + \#X\$\$] + F[\% + FX\%] - X$’, $p2$: ‘$F + [\%X\%\$]\$ - X - f\%[\% - FX] + X$’.

$X \rightarrow p1$ or $X \rightarrow p2$; $F \rightarrow FF$

The meaning of the L-system above is that the tree could has two or three branches. It can use turtle interpretation to describe the meaning of each characters.

'F': Turtle move one steps

'[': and ']' push the turtle's current state into stack. The information into stack is mainly the current location of the turtle. It is important to describe the branches in internode. If the turtle meets the '[', the position of turtle will be recorded, then the turtle moves to the different direction. The new path is the new branch, so if the tree needs three branches in the internode, the production rules need three '[' to record the position. When the turtle meets the end points, the turtle needs to go back to the internode to generate new branches, so ']' can lead the turtle to the original internode. From the perspective of stack, it is out of stack. The turtle moves to another direction whose path is another branch.

'+': Rotate α degrees to left and apply rotation matrix R_Z (α).

'-': Rotate α degrees to right and apply rotation matrix R_Z(α).

'@': Rotate α degrees to down and apply rotation matrix R_y(α).

'#': Rotate α degrees to up and apply rotation matrix R_y(α).

'$': Rotate α degrees to left and apply rotation matrix R_x(α).

'%': Rotate α degrees to right and apply rotation matrix R_x(α).

(2) Branching attribute setting

In the natural, the subbranches are shorter than mother branches. It is meaningful to simulate the gradual shortening of the branches, which can make the tree more real. For the gradual shortening of the branches, the length of branches needs a factor 'L' to control the length of branches. Because mother branch is longer than subbranch, so it is reasonable to set

‘L’ as 0.8. When the turtle meets the ‘[’, the length of turtle’s step needs to multiply 0.8.

Like length of branches, it is also essential to simulate the diameter of branches. The diameter of branches needs a factor ‘d’ to control. By measuring the sample which collected before, it is reasonable to set ‘d’ as 0.8.

(3) The experimental results

To verify the effect of ‘[’ on the number of branches of the tree, delete p2 production rule first, which means that all internodes have three branches. Then, set the length of the tree to 1, the initial diameter of the tree is 0.5, the angle is 25°, the shortening rate of diameter is 0.9 and the shortening rate of length is 0.9. To make the graph more vivid, the number of iterations is set to 4. The experimental results generated by this set of data are shown in the following figure.

Figure 13: Three branches on the internodes

It can be seen from the figure, the branches on each internode are three, which proves the ‘[’ can control the branches on the internode. If the

system put back the p2 production rule, the branches on the internodes would be three or two. This theory will be checked in the following.

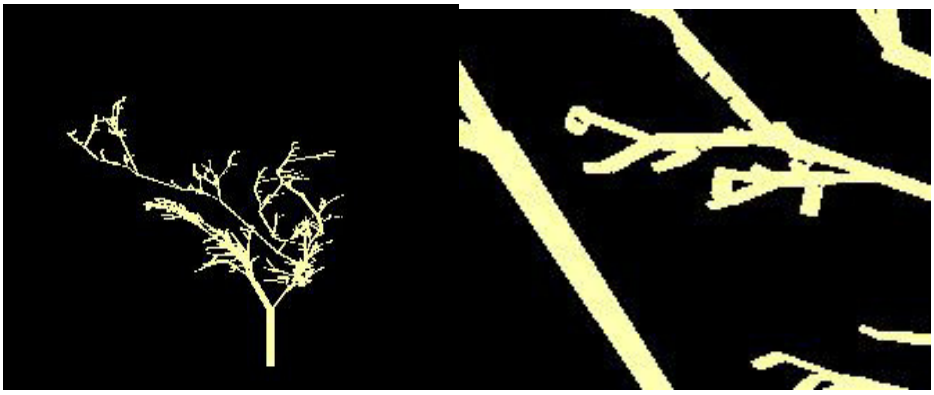


Figure 14: Two or three branches on the internodes

As mentioned in previous chapter, there are two primary shafts in the tree obviously, which represents the p2 production rule has been applied to the tree. Because the virtual tree is 3D, trees can be viewed from different angles.

On the other hand, the end of the tree can be seen clearly. The internode in the graphic has three branches, which represents the p1 production rule has been applied. Production rules applied on each node are randomly selected by the computer, so the tree is more in line with natural of growth.

From the graphic, the mother branch is thicker than the subbranch, which represents that the diameter of trees makes a difference. However, from the first graphic, the diameters of same level of branches are different. The reason of this problem is that since the tree is 3D, the diameters of the tree are different when viewed from different angles, which can make the users observe the tree more realistically.

In addition to the experiments done above, the angle of tree has not been changed. Therefore, change the angle to 15°. The Figure 16 shows different angles of the same tree. And the angle is 15° and 25° respectively.

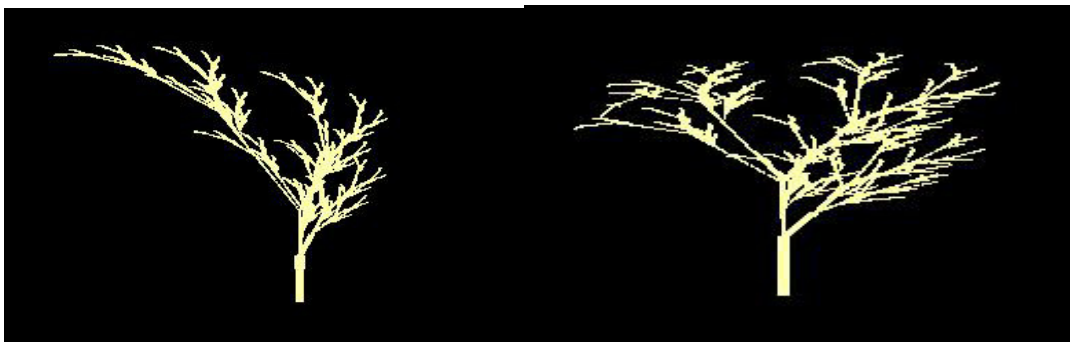


Figure 15: Comparison of tree with different angles

As can be known from the figure, the larger angle, the larger the projection surface of the crown, which vitrifies the above view.

4.4 The construction of monopodial trees

4.4.1 The symbol of monopodial trees

To generate more real monopodial branching trees, there are two main symbols of monopodial branching tree in daily life.

(1) Level of branching

The same as sympodial trees, the branching level is an essential parameter for iteration. Differently, the sympodial trees have fewer branching levels. In order to clearly see the branches of the tree, the dead tree without leaves is the best sample. From the picture of the sample tree (Figure 17), the largest branching level of sympodial tree is two or three.

(2) Angel of branching

From the sample of picture (see Figure 17), it can be known that the angle of the tree is 30°,45° and 60°.

4.4.2 Construction of monopodial trees by L-system

To construct sympodial trees, the first step is to change the production rules. No matter monopodial tree or sympodial tree, production rules control main forms of trees. What needed to do is to construct the main trunk and the branches in the truck.

Axiom(w): 'X'

Production rules(P) $p1$: "$F[+F]F[-F]F[@F]F[\#F]F[\$F]F[\%F]$"

$X \rightarrow p1$; $F \rightarrow p1$

'F': Turtle move one step.

'[': and ']' Push the turtle's current state into stack.

'+': Rotate α degrees to left and apply rotation matrix R_Z (α).

'-': Rotate α degrees to right and apply rotation matrix R_Z(α).

'@': Rotate α degrees to down and apply rotation matrix R_y(α).

'#': Rotate α degrees to up and apply rotation matrix R_y(α).

'$': Rotate α degrees to left and apply rotation matrix R_x(α).

'%': Rotate α degrees to right and apply rotation matrix R_x(α).

From the simple production rule, each '[' has ']' to correspond, which means that when the turtle moves to the endpoint of branches, it will go back to the main truck and move along the main truck. No matter how many times of iterations, the turtle will not change initial direction of the main truck. Through changing the production rules, a tree with distinct trunk could be constructed.

The tree below is made up of the following experimental data. The diameter of the tree is 0.5, the length of the tree to 1, the initial diameter of the tree is 0.5, the angle is 25°, the shortening rate of diameter is 0.9 and the shortening rate of length is 0.9.

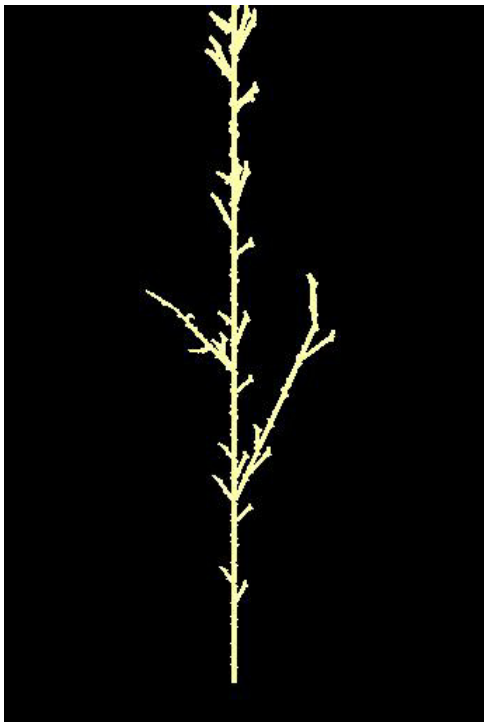


Figure 16: Monopodial tree

4.5 Experimental simulation of random L-system

4.5.1 Choice of development tools

Python is the programming language that implements the project. It is succinct and easy to read and write, allowing users to focus on solving problems rather than figuring out the language itself. Furthermore, Python is an open source software, which means users can copy, read, and change it without spending a penny. It is created and constantly improved by a group of people who want to see a better Python. In addition, Python is compatible with many platforms, so developers do not have to deal with the hassles that often come with using other languages. Tkinter is a built-in

GUI library in Python, and one of the best things about Tkinter is that it does not have to be loaded. The library can be used, and the whole library is very small, simple to get started. It is convenient for developers to set a GUI. In the program, several parameters can be put into software, such as branch length, diameter of a branch and angel of branch.

4.5.2 The design idea

This dissertation mainly simulates monopodial tree and sympodial tree, due to the characteristics of branching patterns. This dissertation also develops L-system to simulate trees.

(1) Simulation of sympodial trees

For the model of sympodial trees, the previous researchers could express the branch structure of plants in nature by using L-system, and also simulated the branch shape through experiments. However, the number of branches at each branch point was basically the same, which was obviously inconsistent with the branch of trees in nature. In order to solve this problem, this dissertation constructs a random L-system that can randomly express different number of branches, so that the branches on each node are generated randomly. This experiment has the following three steps:

Step 1:

Firstly, it is needed to design the user interface which includes control parameters for tree branching attributes, such as the diameter of tree, the angle of branching and the initial length of the branch. Additionally, user can set the parameter value through the interface, and the system will automatically get the value from the interface.

Step 2:

Axioms generate corresponding strings by production rules.

Step 3:

The system reads the generated string. From the initial growth point, the system calculates the position of the next growth point according to the turtle interpretation method, and then plots the string generated in the step 2 according to the turtle interpretation method.

Step 4:

The string which was red by the step 3 is generated again by the production rules. Then repeat the step 3.

The specific flow chart is shown below:

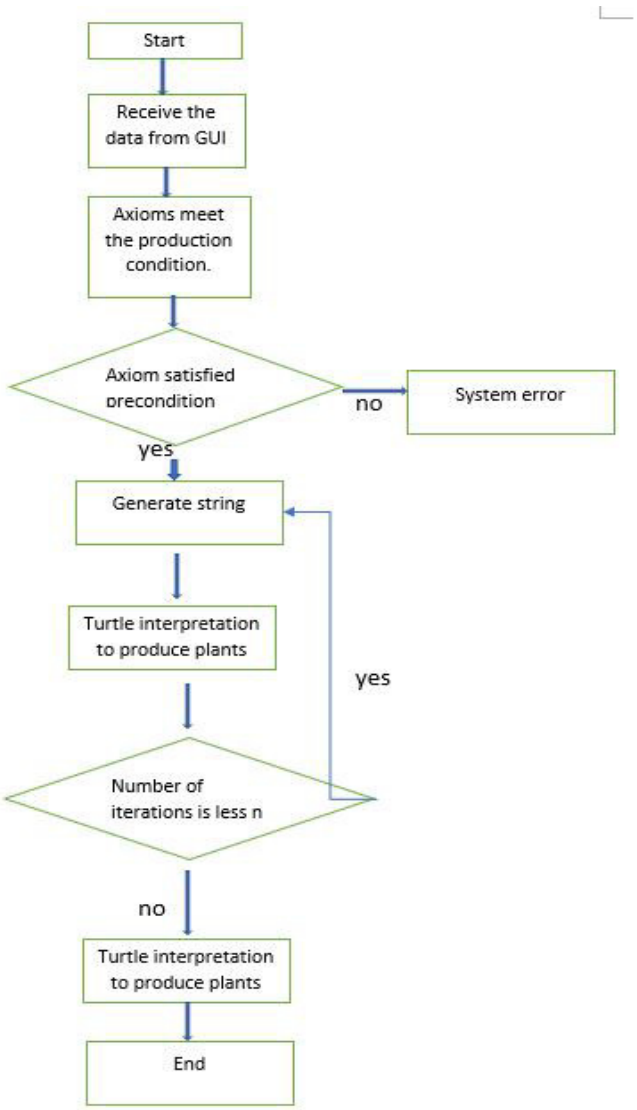


Figure 17: Flow chart

(2) Simulation of monopodial trees

Step 1: Same with sympodial trees, design the user interface in the first place. The user interface includes control parameters for tree branching attributes such as the diameter of tree, the angle of branching and the initial length of the branch. Besides, the users are able to set the parameter value through the interface, and the system will automatically get the value from the interface.

Step 2: Input the production rules. The input production rules need to be made sure about the grammar which can generate monopodial tree.

Step 3: The system reads the generated string. From the initial growth point, the system calculates the position of the next growth point according to the turtle interpretation method, and then plots the string generated in the step 2 according to the turtle interpretation method.

Step 4: The string which was red by the step 3 is generated again by production rules. Then repeat the step 3.

The flow chart is same with sympodial trees (Figure 18), because the flows are the same. In the aim of generating monopodial trees, the grammar is important to control the structure of trees. The users must put the production rules to ensure the turtle moves to the main directions to form the main trunk.

4.6 Implementation of the system

4.6.1 An overview of the software

This software is developed by OpenGL, and the language is Python. To better reflect the system of human-computer interaction, the author adds an interactive interface by Tkinter.

The user interface is mainly composed of three parts: Axioms, production rules parameter control area, parameter control area for trees. The user can change some parameters to control the tree through this interface. To get the user's desired tree shape, the main interface displays two kinds of tree patterns.

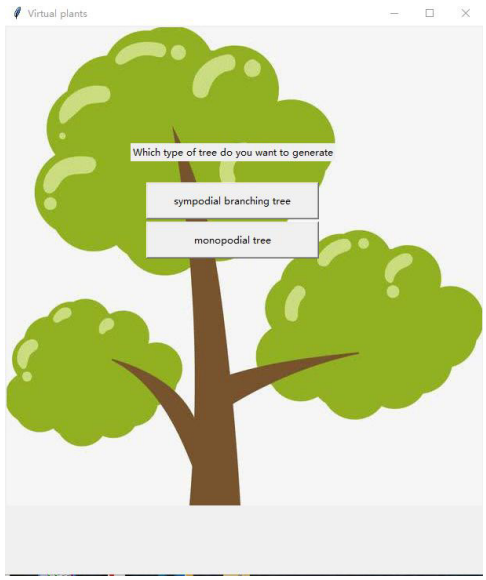


Figure 18: The user interface

When users finish selecting the tree mode, the interface will jump to the parameters page. The production rules are already defined, and if users want to change the production mode, he or she can enter the desired production rule in the input box. The users then enter diameter of the branches and the number of iterations. The system will generate the corresponding trees according to the user input parameters and the number of iterations.

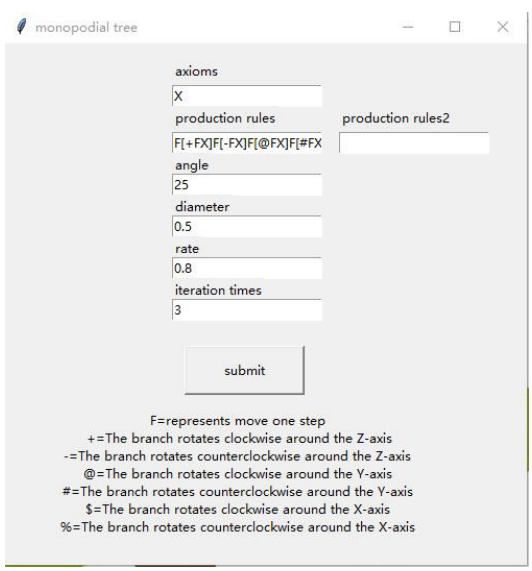


Figure 19: The page for inputting parameters

4.6.2 Perspective

To allow users to view the tree generated by the software from all angles, the author added a 3D drag function. In the following, the Figure 20 and Figure 21 show the same tree generated by the software but observed from different angles.

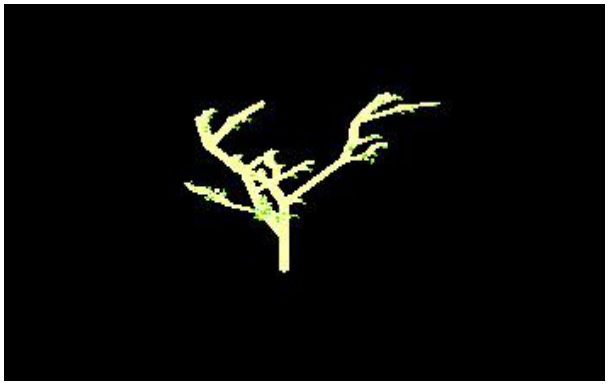


Figure 20: Observe the tree from the front

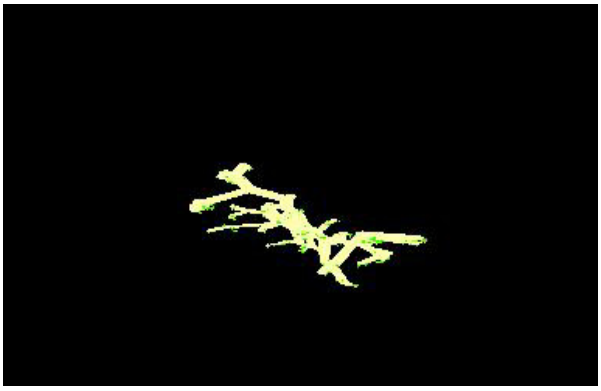


Figure 21: Observe the tree from the top

4.6.3 Growth process

To allow the user to see how the tree is growing, the user can drag the mouse to observe the tree from 3D perspective. When the user drags the tree by mouse to observe, the tree will grow automatically until the number of iterations is reached. Figure 22-25 show the growth process of the tree.

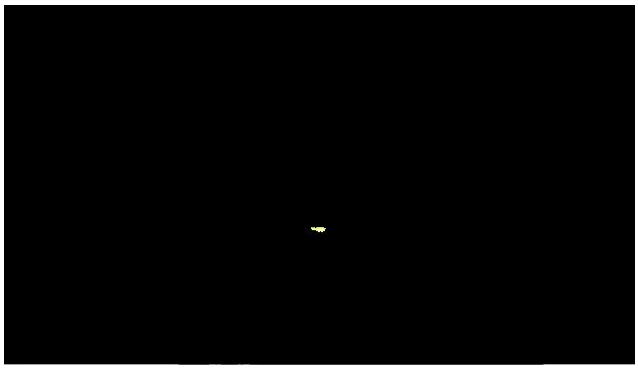


Figure22: The tree iterates once

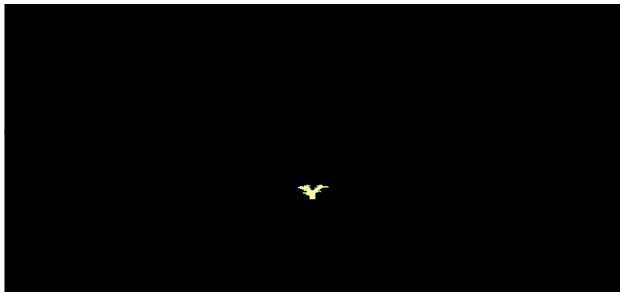


Figure 23: The tree iterates twice

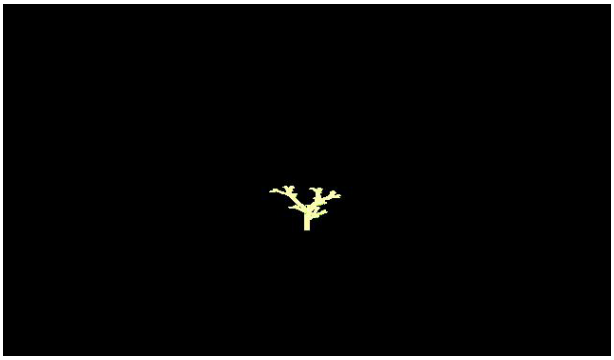


Figure 24: The tree iterates three times

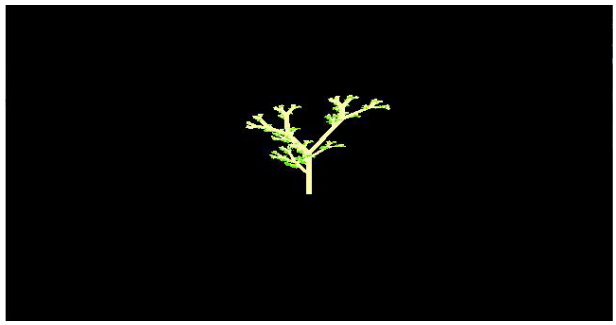


Figure 25: The tree iterates four times

4.6.4 Details of tree control

Using the system can control details of trees, such as changing diameter of the tree, branching angle and number of iterations. Therefore, Changes in each parameter will also lead to changes in the form of trees. Below is the comparison between same shape trees by different parameters. Below is the sample tree which is used to compare a tree with a changed parameter.

Axiom:X

Production rules: $F - [[@X@]\# + \#X\$\$] + F[\% + FX\%] - X$

Angle 25°; Diameter 0.5; Rate: 0.9; Iteration times: 5.

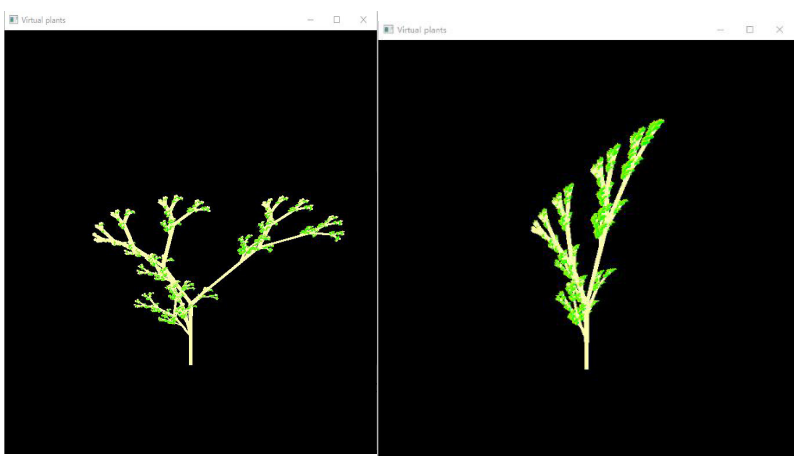


Figure 26: Comparison between same tree with different angles

Figure 27: Comparison between same tree with different diameters

Figure 28: Comparison between same tree with different rates

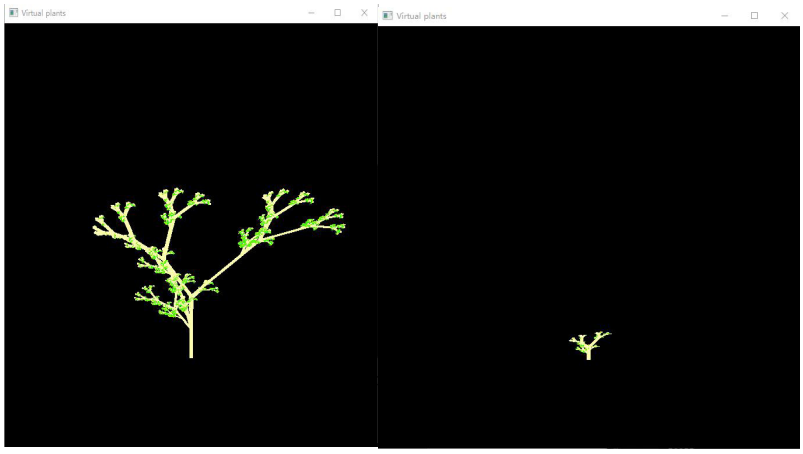


Figure 29: Comparison between same tree with different iteration times

4.6.5 Generate different trees

The software enables the users enter almost parameters about the tree including the production rules. If users input one production rules into the input box, the system can be DOL-system. If users input two production rules, the system can be random L-system. The system will apply two production rules randomly which can generate different trees.

Figure 26 is two trees generated by the different production rules:

Axiom:X

Production rules: $F - [[@X@]\# + \#X\$\$] + F[\% + FX\%] - X$

Angle 25°; Diameter 0.5; Rate: 0.9; Iteration times: 5

Axiom:X

Production rules: $F + [\%X\%\$]\$ - X - F\%[\% - FX] + X$

Angle 25°; Diameter 0.5; Rate: 0.9; Iteration times: 5

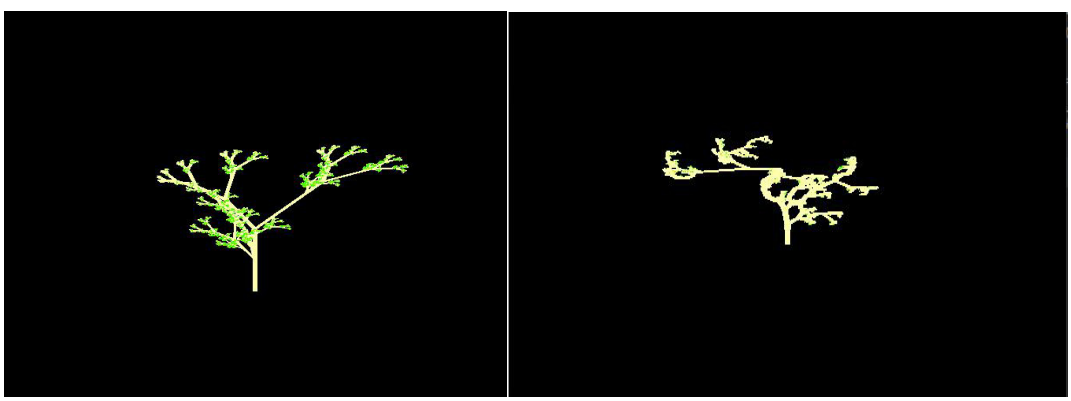


Figure 30: Comparison between two trees by different rules

If two of production rules above are been put, the tree will grow differently every time. The Figure 28 is the comparison between trees by same production rules.

Axiom:X

Production rules1: $F - [[@X@]\# + \#X\$\$] + F[\% + FX\%] - X$

Production rules2: $F + [\%X\%\$]\$ - X - F\%[\% - FX] + X$

Angle 25°; Diameter 0.5; Rate: 0.9; Iteration times: 5

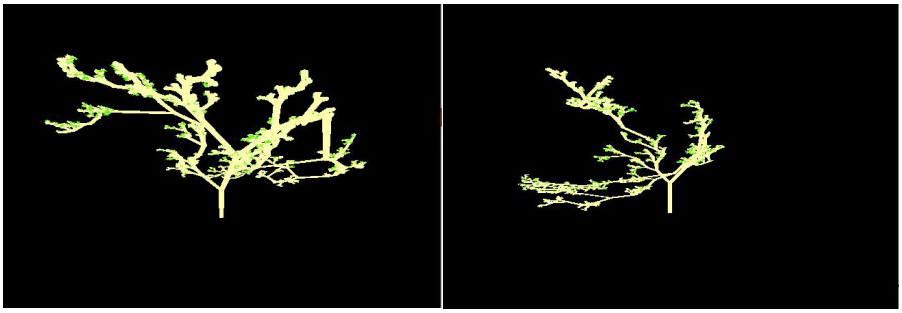


Figure 31: Comparison between two trees generated by Random L-system

From the figure 20-31, all functions of the software have been showed and users can generate different trees by this system.

However, from the figure 28, the trees generated is a little weird. If users input production rules which are disorganized, the tree generated will be strange. It is important to generate trees by inputting axioms and production rules which control topology of trees as 2.3.1 mentioned. This system can only promise to generated trees by ordered production rules and cannot ensure that every production rule can generate trees.

4.7 The summary of this chapter

This chapter introduces several mainstream L-systems through instances and do research to monopodial trees and sympodial trees. The branching characteristics of monopodial trees and sympodial trees are analyzed and summarized, and two different L-systems have been constructed to simulate monopodial trees and sympodial trees. Two types of L-system have been tested to simulate trees. This dissertation uses random L-system

to simulate trees and explains how to control number of branches on the internodes. By changing the production rules, L-system simulates a monopodial tree. Besides, using Thinkter to establish the user interface to implement human-computer interaction and flow chart is used to introduce the operation of the whole system.

Chapter 5: Conclusion and further work

5.1 Conclusion

Plants are the most important parts of nature. They are various in variety and are closely related to human life. The plant growth system is a complex ecosystem with multi-object and multi-factor interaction, which includes both the natural growth status of individual plants and the influence of environment on individual plant growth. Therefore, it is a difficult task to model and simulate the whole ecosystem of plants, but this dissertation is only a preliminary exploration. If a complete plant ecological simulation is to be built, it will require more people to invest in research.

With the development of computer virtual technology and a variety of emerging new technology, so that a variety of previously impossible ideas have been realized. Virtual plants are the ones that benefit the most and becoming more and more popular. This dissertation studies the branching pattern of plants and analyzes the principles and ways of plant branching. For the production rules of L-system, L-system is constructed to simulate virtual plants. This dissertation mainly does the following several aspects of work. Firstly, several methods of generating virtual plants are analyzed and their advantages and disadvantages are analyzed. Then, study monopodial trees and sympodial trees and summarize the structural characteristics of each tree. Thirdly, control the parameters of the branching attribute to change the shape of the tree. The 3D technology not only allows the tree to be viewed from different angles, but also shows the simple growth process of the tree. In this dissertation, random variables are introduced into the L-

system to control the number of branches on the tree nodes, to make trees more realistic. Based on the above research, Pycharm was finally used as a development tool in this dissertation to develop a simulation system of trees and obtain a relatively real virtual plant. The results obtained in this dissertation mainly include the following aspects:

1. Based on a large amount of previous research results, the application of the L-system in plant simulation has a certain understanding. Furthermore, the two-dimensional and three-dimensional drawing principles of L-system are studied. In addition, have a deep understanding of several basic L- system.

2. Based on reading a large amount of literatures, an L-system is constructed to express sympodial trees. This system introduces the parameters used to express the branching characteristics of trees which can control the structure of trees. Using two production rules to control the number of branches on the internodes, which makes the generated trees more in line with the branching characteristics of trees in nature.

3. The methods of generating monopodial trees, namely the production rules, has been studied.

4. A simple system for generating trees has been developed. Users can perform simple operations on the interface and change parameters to control the structure of trees.

5.2 Further work

In this dissertation, the traditional L-system is built which can construct natural lifelike plant forms. However, this dissertation has some shortcomings that needs to be further improved, to this, the future of the main work can be carried out in the following aspects:

1-The branches of the generative plants are rigid. There are several ways to generate curves which can simulate trees more real.

2-The factors infecting plants structure have not been considered. This dissertation simply simulates plant morphology and appearance without considering the influence of internal and external factors in plants growth process. The following research needs to combine the physiological and ecological models of plants to make the simulation effect more realistic.

3-This dissertation needs to improve the program. The structure of this program is simple, the content is less, and the plant structure has certain limitations. The next step is to develop a set of mature software to simulate plant morphology based on the plant physiological and ecological model, to enrich program functions and make it convenient for users to use.

4-Add plant organs to the tree, such as flowers and leaves.

5.3 Future trend

In the future, research about virtual plants which are based on computer technology will develop in high speed. The trend of research contains following aspects:

(1) The modeling approach will get simpler.
The existing modeling approaches require researchers to be equipped with computer technology, but for the researchers who are not equipped with it, the approaches are hard for them, thus future modeling software need to be simpler and more generalized. More mature graphical interface, the formation of powerful modeling software, is the future trend of development.

(2) Elaboration of virtual plants
Simulation of the structure of plants should be more fine than original foundation. Future models can show real natural beauty of plants, for example: curl of leaves, curvature of trunk and trunk texture.

(3) Plant morphogenesis model and physical model need to be more combined. Plants structure is the external performance and can

reflect plant states intuitively. The physiological and ecological characteristics of plants are the results of co-action of lots of factors, such as photosynthesis, respiration and light, so doing the research about physical model of plants and environment model of affecting plants is very meaningful.

(4) Virtual plants need to be more combined with other disciplines. Virtual plants involve with lots of academic disciplines, for example, computer graphics, agriculture, forestry science, applied mathematics, botany and ecology. The development of virtual plants promotes other related disciplines.

Chapter 6 Reflection

This chapter provides the author's reflection of the project. At the beginning of the project, the aim was to generate virtual plants. There are several methods can be chosen to finish the project. The author consulted several relative materials and books and had a general understanding of the relevant methods. In the process, there was a problem that how to generate a virtual plant as real as possible. The author compared this method with the chapter 2 mentioned and found that L-system is the best way. Then the author red different materials about L-system and had in-depth understanding of L-system.

When the author decided to use the L-system to generate visual plants, another problem arose that how to choose tools to draw the graphics. By comparing several tools and got the conclusion that OpenGL could draw the graphics easily because OpenGL is platform-independent and programming language-independent. It is convenient because any computer system and programming language could be used.

Despite the ease, however, there were several disadvantages, for instance the image was in two-dimensional and the branches were all same thickness. To overcome the first problem, a three-dimensional space is required to build to show the plants. Due to lack of knowledge of computer graphics, the author constructed a simple three-dimensional space which included coordinate system. To enable objects in the visual field can be clearly seen, it is needed to understand some of the concepts in OpenGL: projection, frustum and viewport. Through several attempts, a three-dimensional space had been set. The author tried to draw several shapes in the space to be familiar with the OpenGL. In order to finish the object in the space could be seen from different angles, the author set mouse functions to drag the objects in the space. When the object in the space had been dragged, the distance between the eye and the observation point, elevation and azimuth need to be changed through calculating.

After the three-dimensional space had been set, it is acquired to draw the tree in the space. The first problem was that the line could not been used to describe branches of tree, because line was a concept in two-dimensional. What was needed to do was to find a substitute for lines. The cylinder was the best substitute to replace the line in three-dimensional space, because cylinder could describe the trunk in 3D. If the author wanted to change the thickness of the trunk, the author just needed to control the diameter of the cylinder. However, the ends of the cylinder were not closed, which caused the graphics to look ugly when magnified.

The next problem was that how to control the branches' nodes. The author was familiar with the turtle interpretation via reading previous literatures. Each pair of '[]' represents a branch and the reason have been explained above, so the length and diameter of tree could be controlled by '[]'. However, there was a better way to record the internodes, which was using the stack to record the internodes and the endpoint of the branches. The positions of internodes and the endpoints of branches recorded could control the length and diameter of each level more easily.

References:

Abelson, H. and DiSessa, A., (1992). ‘Turtle Geometry’. Cambridge: MIT Press.

Aono, M. and Kunii, T. L. (1984) ‘Botanical Tree Image Generation’, *IEEE Computer Graphics and Applications*, 4(5), pp. 10–34.

Bouchon Jean, De Reffye Philippe and Barthélémy Daniel. (1997). ‘Modélisation et simulation de l'architecture des végétaux’. Paris: INRA, 1997.

Bourke (1991), ‘DLA - Diffusion Limited Aggregation’. Available at: http://paulbourke.net/fractals/dla/

Demko, S., Hodges, L. and Naylor, B. (1985) ‘Construction of Fractal Objects

with Iterated Function Systems.', *Computer Graphics (ACM)*, 19(3), pp. 271–278.

Gravelius Harry (1914), 'Flusskunde'. Berlin: G.J. göschen

Godin, C., Costes, E. and Caraglio, Y. (1997) 'Exploring plant topological structure with the AMAPmod software: An outline', *Silva Fennica*, 31(3), pp. 357–368.

Greene, N. (1989) 'Voxel space automata: Modeling with stochastic growth processes in voxel space', *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1989*, 23(3), pp. 175–184.

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 shape of the tree-like body', Journal of Theoretical Biology, pp:331=338

Honda, H. and Tomlinson, P. B. (1982) 'Two geometrical models of branching of botanical trees', *Annals of Botany*, pp. 1–12.

Hutchinson, J. E. (1981) 'Fractals and Self Similarity', *Indiana University Mathematics Department*, 30(5), pp. 713–747.

Jaeger, M. and De Reffye, P. H. (1992) 'Basic concepts of computer simulation of plant growth', *Journal of Biosciences*, 17(3), pp. 275–291.

Kurtz-Fernhout (2000), Available at: http://www. kurtz-fernhout. com/PlantStudio, Onyx Computing Inc (1993), Available at: http://onyxtree.com

Laug (2019), Available at : https://commons.wikimedia.org/wiki/File:Barnsley_fern_2000x2000.png

Lindenmayer A. (1968). 'Mathematical models for cellular interactions in development'. *Journal of Theoretical Biology*

Lintermann, B. and Deussen, O. (1999) ‘Interactive Modeling of Plants’, *IEEE Computer Graphics and Applications*, 19(February), pp. 56–65.

Mandelbrot, B. B. (1982) ‘Fractal Geometry’, *Nature*.

Měch, R. and Prusinkiewicz, P. (1996) ‘Visual models of plants interacting with their environment’, *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996*, pp. 397–410.

Michael F Barnsley, S. D. (1985) ‘Iterated Function Systems and the Global Construction of Fractals’, *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 399(1817), pp. 243–275.

Microsoft (1996), Available at: https://directx.en.softonic.com/

Perttunen, J. and Sieva, R. (1998) ‘Modelo Lignum’, *1998 Elsevier Science B.V.*, 108, pp. 189–198.

Prusinkiewicz, P. (1986) ‘Graphical applications of L − systems’, *Proceedings of graphics interface*.

Prusinkiewicz, P. (1997) ‘A look at the visual modeling of plants using L-systems’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1278, pp. 11–29.

Prusinkiewicz, P. and Hammel, M. (1991) ‘and " Fractal Modeling in 3D Computer Graphics and Imagery "’, pp. 1–29.

Prusinkiewicz, P., Hammel, M. S. and Mjolsness, E. (1993) ‘Animation of plant development’, *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1993*, pp. 351–360.

Prusinkiewicz, P. and Lindenmayer, A., (2004). ‘The Algorithmic Beauty of Plants’. New York: Springer-Verlag.

Prusinkiewicz, P. and Hanan, J. (1997) ‘Extensions to the graphical interpretation of L-systems based on turtle geometry’, *Science Research & Publications*, pp. 1–56.

Prusinkiewicz, P., Hanan, J. and Měch, R. (2000) ‘An L-system-based plant modeling language’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1779, pp. 395–410.

Przemyslaw Prusinkiewicz, Radoslaw Karwowski, Radomír Měch, J. H. (1999) *L-Studio/cpfg: A Software System for Modeling Plants*, *Applications of Graph Transformations with Industrial Relevance*.
Reeves, W. T. (1983) ‘Particle SystemsmA Technique for Modeling a Class of Fuzzy Objects’, *ACM Transactions on Graphic*, 2(2), pp. 91–108.

de Reffye, P. *et al.* (1988) ‘Plant models faithful to botanical structure and development’, *Computer Graphics (ACM)*, 22(4), pp. 151–158.

De Reffye, P. *et al.* (1997) ‘A functional model of tree growth and tree architecture’, *Silva Fennica*, 31(3), pp. 297–311.

Room, P., Hanan, J. and Prusinkiewicz, P. (1996) ‘Virtual plants: New perspectives for ecologists, pathologists and agricultural scientists’, *Trends in Plant Science*, 1(1), pp. 33–38.

Schwartz, J. T., von Neumann, J. and Burks, A. W. (1967) ‘Theory of Self-Reproducing Automata’, *Mathematics of Computation*, p. 745.

Silicon Graphics Inc (1992), Available at: https://www.opengl.org/

T.A.Witten, L. M. S. (1983) ‘Diffusion-limited aggregation’, *1983 The American Physical Society*, 27(9), p. 240.

Tatsumi, J., Yamauchi, A. and Kono, Y. (1989) ‘Fractal analysis of plant root systems’, *Annals of Botany*, 64(5), pp. 499–503.

Taubin, G. *et al.* (1998) 'Geometry coding and VRML', *Proceedings of the IEEE*, 86(6), pp. 1228–1243.

Ulam S. (1962) 'On some mathematical problems connected with patterns of growth of figures', Proceedings of Symposia in Applied Mathematics. Providence: Am. Math. Soc. Vol. 14, 1962, 14: 215-224.

Viennot, X. G. *et al.* (1989) 'Combinatorial analysis of ramified patterns and computer imagery of trees', *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1989*, 23(3), pp. 31–40.

Wang, C. et al. (2013). 'Recursion for Capturing Leaf Mass'. Applied Mechanics and Materials, vol. 295–298, pp. 2300–2307.

Zapata-Grajales, F. N., Cano-Velásquez, N. A. and Villa-Ochoa, J. A. (2018) 'Art and geometry of plants: Experience in mathematical modelling through projects', *Eurasia Journal of Mathematics, Science and Technology Education*, 14(2), pp. 585–603.