

二维 L-系统的推广及在植物模拟中的应用

袁修久, 刘欣, 王胜勇, 赵学军

YUAN Xiu-jiu, LIU Xin, WANG Sheng-yong, ZHAO Xue-jun

空军工程大学 理学院 应用数学物理系, 西安 710051

Science Institute, Air Force Engineering University, Xi'an 710051, China

YUAN Xiu-jiu, LIU Xin, WANG Sheng-yong, et al. Improvement of two-dimensional L-system and its application to plant modeling. Computer Engineering and Applications, 2010, 46(1): 143-146.

Abstract: Simulating bending branches is an important part of plant simulation and L-system is one of the significant methods of it. But it requires plenty of productions if L-system is applied to simulate bending branches. In this paper, functional L-system is presented, through adding the bending symbols and curvature functions to the productions. When modeling shapes of plants using functional L-system, without changing the number of productions, various shapes of trees with the same topological structure will be produced through adjusting the curvature functions. Functional L-system is a very concise description in modeling bending branches of plants, and the length of the branches can be easily controlled in programming.

Key words: simulation of shapes of plants; L-systems; curvature functions; bending branches

摘要: 弯曲枝条的模拟是植物模拟的重要组成部分, L-系统是植物模拟的重要方法之一。如果用 L-系统模拟弯曲的枝条, 则需要大量的产生式。通过在产生式中增加弯曲符号以及描述弯曲程度的曲率函数得到了函数 L-系统。利用函数 L-系统模拟植物, 在不改变产生式个数的情况下, 通过调整曲率函数能够绘制出相同拓扑结构, 但整体形态差异较大的植物; 利用函数 L-系统模拟具有弯曲枝条的植物, 产生式简洁, 同时在编程时, 容易控制枝条的长度。

关键词: 植物形态结构模拟; L-系统; 曲率函数; 弯曲枝条

DOI: 10.3778/j.issn.1002-8331.2010.01.044 **文章编号:** 1002-8331(2010)01-143-04 **文献标识码:** A **中图分类号:** TP391

1 引言

植物模拟是计算机图形学的重要内容之一。逼真地绘制植物图形离不开弯曲的枝条的模拟。文献[1]提出了一种计算枝条弯曲的力学模型, 实现了基于双尺度自动机模型^[2]的枝条弯曲算法, 该算法能简洁直观地模拟植物在生长过程中形态结构的动态发展。文献[3]利用参数曲线拟合实际测量的数据得到弯曲枝条的形状, 该方法需要对植物做大量的观测。文献[4]将枝条定义为一个可以弯曲变形, 但不可以伸长的弹性杆, 建立了枝条生长的生物力学模型, 模拟了由内力所造成的枝条弯曲。影响枝条弯曲的因素很多, 直接从影响枝条弯曲的机理上建立弯曲枝条的模型是非常困难的。

A. Lindenmayer 于 1968 年提出了 L-系统^[5]。L-系统算法能够绘制出多种漂亮分形曲线和植物图像, 特别是能够模拟植物形态结构的演化过程。L-系统对于弯曲枝条的绘制, 从数学上看, 使用的是“以直代曲”的思想, 即用折线代替曲线, 要逼真地绘制一个弯曲的枝条, 则需要大量的产生式, 从而使得 L-系统的表达变得繁琐, 失去了数学美。另外在绘制植物的时候, 常常需要考虑主杆和侧枝的比, 上一级枝条和本级枝条的比, 这样

在编程时就需要计算枝条的长度, 有时还需要考虑枝条的光滑联接, 而使用三次参数曲线绘制弯曲枝条虽然能够容易实现光滑联接, 但是其计算枝条的长度很麻烦, 同样使用样条绘制弯曲枝条计算枝条长度也不太方便。由于各种数学软件的发展, 使得求微分方程组的速度明显加快, 精度明显提高。在该文, 将使用微分方程来描述曲线, 由此绘制弯曲的枝条。

曲线论的基本定理指出, 平面曲线的曲率完全决定了曲线形状。曲率函数不同, 曲线的形状不同。从这一思想出发利用曲线线段代替直线段改进了二维 L-系统, 得到了函数 L-系统。由于微分方程组的参数变化范围就是枝条的长度, 因此在编程时枝条的长度容易控制, 同时使用枝条每一点的弯曲程度来控制枝条的形状实际意义明显。

2 经典 L-系统

L-系统是一个字符串改写系统。它首先定义字符集合, 设置初始字符串和字符串替代规则(或产生式), 然后根据产生式对原始字符串不断进行替代。每一步替代过程中字符的替换都是平行的, 即所有字符使用时都进行替代操作。使用二维 L-系

基金项目: 国家自然科学基金(the National Natural Science Foundation of China under Grant No.60663003); 陕西省电子信息系统综合集成重点实验室基金。

作者简介: 袁修久(1966-), 男, 教授, 研究方向为虚拟现实, 粗糙集; 刘欣(1985-), 女, 硕士研究生, 研究方向为虚拟现实; 王胜勇(1983-), 硕士研究生, 研究方向为虚拟现实; 赵学军(1972-), 男, 讲师, 研究方向为软件理论。

收稿日期: 2009-07-21 **修回日期:** 2009-09-07

统生成植物图形包括两个步骤:首先根据初始字符和产生式生成一个字符串,然后从左至右依次将每个字符解释为动作指令,操控一只“海龟”在二维平面上绘制出相应的植物。“海龟”所处的状态定义为一个三元素集合 (x, y, α) ,其中坐标 (x, y) 表示“海龟”的位置,方向角 α 表示龟的方向,文献[6]对最简单的OL型L-系统中的字符作了如下的龟解释。

- $F(d)$:沿当前方向向前移动一步,步长为 d ;
- $f(d)$:沿当前方向向前移动一步,步长为 d ,但不画线;
- $+(\delta)$:逆时针旋转 δ 角度;
- $-(\delta)$:顺时针旋转 δ 角度;
- $[$:将“海龟”的当前状态压入堆栈;
- $]$:从堆栈中弹出一个状态作为当前状态。

3 二维曲线理论及在植物形态绘制中的解释

由微分几何理论,存在唯一一条平面光滑曲线 $r=r(s)$,该曲线以 $r(0)=r_0$ 为起点,以 $\varphi(s)$ 为相对曲率函数,以 α_0 为初始切向量,以 β_0 为初始法向量。该曲线的方程是下面微分方程组(1)的解。

$$\begin{cases} \frac{dr}{ds} = \alpha \\ \frac{d\alpha}{ds} = \varphi(s)\beta \\ \frac{d\beta}{ds} = -\varphi(s)\alpha \\ r(0)=r_0; \alpha(0)=\alpha_0; \beta(0)=\beta_0 \end{cases} \quad (1)$$

其中 α 是曲线的切向量, β 是曲线的主法向量, s 是弧长。用微分方程组(1)解出的曲线段表示植物的枝条,而 α 是树枝生长的方向, β 是枝条弯曲的方向,弧长 s 为枝条的长度。在下文也经常把曲线段说成枝条。

设曲线 $r=r(s)$ 的单位切向量 $\alpha(s)=(\alpha^1(s), \alpha^2(s))$,则其单位法向量为 $\beta(s)=(-\alpha^2(s), \alpha^1(s))$ 或 $\beta(s)=(\alpha^2(s), -\alpha^1(s))$ 。 $\alpha(s), \beta(s)$ 取定后,则 $\varphi(s)$ 和 $-\varphi(s)$ 对应枝条弯曲的方向正好相反。枝条的弯曲方向由切向量的方向、法向量的方向,曲率函数的符号组合确定。为了能够清楚地看出这种组合同枝条弯曲方向的关系,取定 $\varphi(s)>0$,并且取 $\beta(s)=(-\alpha^2(s), \alpha^1(s))$,这时枝条的弯曲方向同 $\alpha(s)$ 各分量符号之间的关系如表1所示。如果取 $\varphi(s)>0$,则当 $\beta(s)=(\alpha^2(s), -\alpha^1(s))$ 时,表1中的枝条弯曲方向刚好相反。

表1 切向量各分量的符号同枝条弯曲方向的关系

$$(\varphi(s)>0, \beta(s)=(-\alpha^2(s), \alpha^1(s)))$$

切向量各分量的符号	枝条弯曲方向
$\alpha^1(s)>0, \alpha^2(s)>0$	弯曲方向向上(或弯曲方向向右)
$\alpha^1(s)>0, \alpha^2(s)<0$	弯曲方向向上(或弯曲方向向右)
$\alpha^1(s)<0, \alpha^2(s)<0$	弯曲方向向下(或弯曲方向向左)
$\alpha^1(s)<0, \alpha^2(s)>0$	弯曲方向向下(或弯曲方向向左)

如果枝条的初始方向 α_0 和弯曲方向 β_0 给定,曲率函数的符号恒定,则枝条的弯曲方向就不会发生变化。但是实际的一段枝条可能既有向上弯曲的部分,也有向下弯曲的部分,并且向上弯曲和向下弯曲的枝条的连接是光滑的。要绘制这种枝条,开始的时刻可以取 $\varphi(s)>0$,并且随后 $\varphi(s)$ 的符号一直是正的,如果经过长度 l 后,枝条要改变弯曲方向,再将 $\varphi(s)$ 取为负值,在不同弯曲方向的分界点令 $\varphi(s)=0$ 。

下面的例子描述了曲率 $\varphi(s)$ 的两种典型形式下,曲线的

形状。

$$\text{例1 曲率 } \varphi(s) = \begin{cases} \frac{h}{s_1-s_0}(s-s_0), & s \in [s_0, s_1] \\ \frac{h}{s_2-s_1}(s-s_2), & s \in [s_1, s_2] \\ 0 & \text{其他} \end{cases}$$

其中 $s_0, s_1, s_2, h>0$ 是已知参数。 $\varphi(s)$ 的示意图如图1(a)所示。当 $s_0=0.1, s_1=0.7, s_2=1, h=2, r_0=(0, 0), \alpha_0=(1/\sqrt{2}, 1/\sqrt{2}), \beta_0=(1/\sqrt{2}, -1/\sqrt{2})$ 时,由微分方程组(1)解出的曲线如图1(b)所示。这是一段弯曲方向向下,在 s_1 处弯曲达到最大的曲线弧。

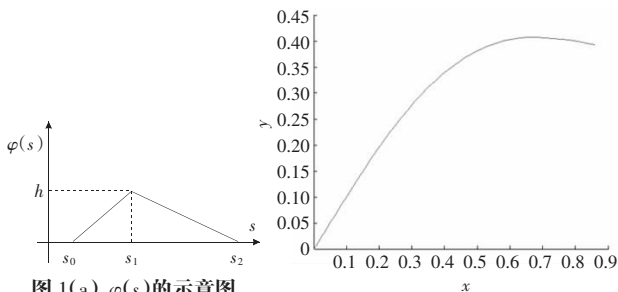


图1(a) $\varphi(s)$ 的示意图

图1(b) 由微分方程组(1)解出的曲线

例2 曲率

$$\varphi(s) = \begin{cases} c_1 & s_0 \leq s < s_1 \\ c_1 + \frac{c_2-c_1}{s_2-s_1}(s-s_1) & s_1 \leq s < s_2 \\ c_2 & s_2 \leq s < s_3 \\ c_2 + \frac{c_3-c_2}{s_4-s_3}(s-s_3) & s_3 \leq s < s_4 \\ c_3 & s_4 \leq s < s_5 \\ c_3 + \frac{c_4-c_3}{s_6-s_5}(s-s_5) & s_5 \leq s < s_6 \\ c_4 & s_6 \leq s < s_7 \\ c_4 + \frac{c_5-c_4}{s_8-s_7}(s-s_7) & s_7 \leq s \leq s_8 \end{cases}$$

$\varphi(s)$ 的示意图如图2(a)。当 $s_0=0.1, s_1=0.3, s_2=0.5, s_3=0.7, s_4=0.9, s_5=1.1, s_6=1.3, s_7=1.5, s_8=1.7, c_1=3, c_2=-5, c_3=7, c_4=-9, c_5=11$,

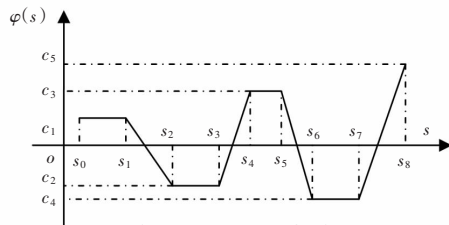


图2(a) $\varphi(s)$ 的示意图

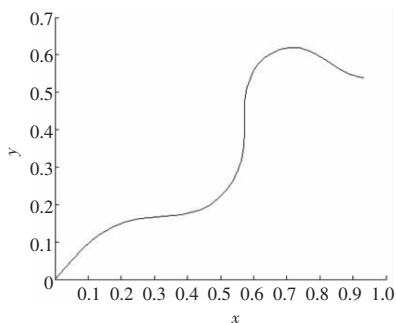


图2(b) 由微分方程(1)解出的曲线

$\alpha_0=(1/\sqrt{2}, 1/\sqrt{2}), \beta_0=(1/\sqrt{2}, -1/\sqrt{2})$ 时, 由微分方程(1)解出的曲线如图 2(b), 这是一段向不同方向弯曲的曲线。

4 推广的二维 L-系统

对于一段各点弯曲程度不同的曲线, 如果用 L-系统进行逼真地绘制, 就需要多次重写, 在每次重写时, 由于龟前进时逆时针或顺时针旋转角度可能不相同, 因此需要大量的产生式, 而且角度很难计算。虽然对 L-系统进行龟解释以后, 可以生成各种植物图形, 但这些植物图形的枝条都是由一个一个的直线段联接而成, 或者直接由直线段构成, 生成的植物, 特别是带有弯曲枝条的植物, 如果产生式少就有可能出现一段向不同方向弯曲的曲线段用直线段替代的可能, 生成的植物生硬、呆板。下面以 OL-系统为例把 L-系统推广成函数 L-系统。

定义 1 二维函数 OL-系统是一个有序的四元组 $G=\langle V, \Sigma, \omega, P \rangle$, 其中 V 是字符表, 它包括表示弯曲方向的字符 $\cap, \cup, \supset, \subset$; Σ 是曲率函数的集合; $\omega \in (V \times \Sigma)$ 是联系着曲率函数的一个字符串, 称为公理; $P \subseteq (V \times \Sigma) \times (V \times \Sigma)$ 是一个有限产生式的集合。

4.1 龟解释

同经典的 L-系统相比, 二维函数 L-系统的龟形解释中应增加对表示弯曲方向的符号的解释, 而其他符号的意义也应该重新解释。二维函数 L-系统“海龟”所处的状态定义为一个四元素集合 $(x, y, \alpha_0, \beta_0)$, 其中坐标 (x, y) 表示“海龟”的位置, 向量 α_0 表示“海龟”头部的指向, 向量 β_0 表示“海龟”将要前进的曲线的弯曲方向。二维函数 L-系统中的字符龟解释如下:

$F(d, \varphi(s))$: 以当前方向沿曲线向前移动一步, 步长为 d , 且曲线各点的弯曲程度为 $\varphi(s)$, 曲线段的方程由微分方程组(1)解出。如果 $\varphi(s)=0$, 此时, 曲线段为直线段, 将 $F(d, \varphi(s))$ 简称为 $F(d)$ 或 F ;

$f(d, \varphi(s))$: 以当前方向沿曲线向前移动一步, 步长为 d , 且曲线各点的弯曲程度为 $\varphi(s)$, 但不画线;

\cup : 曲线弯曲方向向上;

\cap : 曲线弯曲方向向下;

\subset : 曲线弯曲方向向左;

\supset : 曲线弯曲方向向右;

$+\delta$: 初始切向量 α_0 逆时针旋转 δ ;

$-\delta$: 初始切向量 α_0 顺时针旋转 δ ;

函数 L-系统同经典 L-系统最大的不同主要是在字符表中增加了表示弯曲方向的符号 $\cap, \cup, \supset, \subset$ 以及画线的字符联系了描述曲线段弯曲程度的曲率函数 $\varphi(s)$ 。

4.2 龟解释程序设计步骤

下面以有主干的植物为例, 叙述龟解释程序设计步骤。

字符 F 解释为主干图形, 字符 B 解释为枝条图形。 k_1 是下一级主干长度同本级主干长度的比或下节枝条与本节枝条的长度比, k_2 是主干的一个内节点生成的侧枝的第一个内节点的长度与该主干内节点长度的比, k_3 是下一级枝条长度同本级枝条长度的比。

步骤 1 给出初始主干长: $alength$; 初始枝长: $blength$;

步骤 2 字符串解释开始, 当遇到字符“F”时转步骤 3; 当遇到字符“B”时转步骤 4; 当遇到“+”时, 转步骤 5, 当遇到“-”时转步骤 6, 当遇到字符 $\cap, \cup, \supset, \subset$ 中的一个时转步骤 7, 当遇“[”转步骤 8, 当遇“]”转步骤 9; 字符串解释完毕, 转步骤 11;

步骤 3 调用描述主干弯曲程度的曲率函数 $\varphi_1(s)$, 以及主干的初始参数 $r_0, \alpha_0, \beta_0, alength$, 求微分方程组(1)的解, 绘出该解表示的曲线。特别地, 记录曲线末端的坐标 r_0' , 切向量 α_0' , 法向量 β_0' 。 $r_0=r_0', \alpha_0=\alpha_0', \beta_0=\beta_0', alength=k_1 \cdot alength, blength=k_2 \cdot alength$, 转步骤 10;

步骤 4 调用描述枝条弯曲程度的曲率函数 $\varphi_2(s)$, 以及枝条的初始参数 $r_0, \alpha_0, \beta_0, blength$, 求方程组(1)的解, 绘出该解表示的曲线。特别地, 记录曲线末端的坐标 r_0' , 切向量 α_0' , 法向量 β_0' 。 $r_0=r_0', \alpha_0=\alpha_0', \beta_0=\beta_0', blength=k_1 \cdot blength$, 转步骤 10;

步骤 5 将 α_0 顺时针旋转角 θ , 得到 α_0' 。 $\alpha_0=\alpha_0'$, 转步骤 10;

步骤 6 将 α_0 逆时针旋转角 θ , 得到 α_0' 。 $\alpha_0=\alpha_0'$, 转步骤 10;

步骤 7 根据表 1 确定 β_0' 。 $\beta_0=\beta_0'$, 转步骤 10;

步骤 8 将当前节点的 r_0, α_0, β_0 压入堆栈, $blength=k_3 \cdot blength$, 转步骤 10;

步骤 9 将栈顶的节点参数弹出, 赋给当前节点, $blength=blength/k_3$, 转步骤 10;

步骤 10 $i=i+1$, 转步骤 2;

步骤 11 退出程序。

由于曲率为 0 的曲线段就是直线段, 因此经典的 L-系统可以看成是函数 L-系统的特殊情况。

4.3 实验结果分析

实验 1 文献[6]中给出了一个产生嫩枝的 L-系统:

$\omega: F$

$P_1: F \rightarrow [B]F, 1 \leq i \leq 3$

$P_4: F \rightarrow [B]A$

$P_5: A \rightarrow K$

其龟解释 F, B 和 A 都是直线段, 但是画出来的草本植物, B 却是曲线段, 如图 3(a), 这让人很费解。要写出图 3(a)所示植物的经典 L-系统, 则需要大量的产生式。可以把图 3(a)的 L-系统用函数 L-系统表示为函数 L-系统式(3)。函数 L-系统式(3)在字符 B 中增加了曲率函数, 同 L-系统式(2)相比产生式的个数没有发生变化。

$\omega: F$

$P_1: F \rightarrow [(\cap B(1, \varphi(s)))]F, 1 \leq i \leq 3$

$P_4: F \rightarrow [(\cap B(1, \varphi(s)))]A$

$P_5: A \rightarrow K$

$\omega: F$

$P_1: \begin{cases} F \xrightarrow{0.5} [(\cap B(1, \varphi(s)))]F \\ F \xrightarrow{0.5} [(\cup B(1, \varphi(s)))]F \end{cases}, 1 \leq i \leq 3$

$P_4: \begin{cases} F \xrightarrow{0.5} [(\cup B(1, \varphi(s)))]A \\ F \xrightarrow{0.5} [(\cap B(1, \varphi(s)))]A \end{cases}$

$P_5: A \rightarrow K$

其中 $\varphi(s)=10s$ $\alpha_0=(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$ 。

函数 L-系统式(4)的图形如图 3(b), 它是在函数 L-系统式(3)的基础上, 增加了随机性后的函数 L-系统。下面是允许主干弯曲的二个函数 L-系统。

$\omega: F(\varphi_1(s))$

$P_1: F(\varphi_1(s)) \rightarrow [(\cap B(\varphi_2(s)))]C F(\varphi_1(s)) \quad i=1, 3$

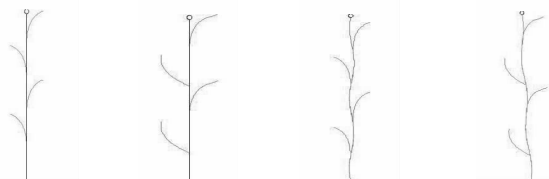
$P_2: F(\varphi_1(s)) \rightarrow [(\cap B(\varphi_2(s)))]C F(\varphi_1(s))$

$P_4: F(\varphi_1(s)) \rightarrow [(\cap B(\varphi_2(s)))]C A$

$P_5: A \rightarrow K$

$$\begin{aligned} \omega: & F(\varphi_1(s)) \\ P_1: & \begin{cases} F(\varphi_1(s)) \xrightarrow{0.5} [\cup B(\varphi_2(s))] \subset F(\varphi_1(s)) \\ F(\varphi_1(s)) \xrightarrow{0.5} [\cup B(\varphi_2(s))] \supset F(\varphi_1(s)) \end{cases}, 1 \leq i \leq 3 \\ P_4: & \begin{cases} F(\varphi_1(s)) \xrightarrow{0.5} [\cup B(\varphi_2(s))] \subset A \\ F(\varphi_1(s)) \xrightarrow{0.5} [\cap B(\varphi_2(s))] \subset A \end{cases} \\ P_5: & A \rightarrow K \end{aligned} \quad (6)$$

其中 $\varphi_1(s) = \sin(2\pi s)$, $\varphi_2(s) = 10s$, $\alpha_0 = (1/\sqrt{2}, 1/\sqrt{2})$ 。函数 L-系统式(5)、(6)所画草本植物如图 4(a), (b)所示。



(a)改进的确定的 L-系统绘制的枝为弯曲的树 (b)改进的随机的 L-系统绘制的枝为弯曲的树 (a)改进的确定的 L-系统绘制的枝为弯曲的树 (b)改进的随机的 L-系统绘制的枝和干都为弯曲的树

图 3 用改进的 L-系统绘制的枝为弯曲的树 图 4 用改进的 L-系统绘制的枝和干都为弯曲的树

图 3, 图 4 是用 matlab 绘制的植物。看到侧枝是弯曲的, 不同的函数 L-系统实现了不同方向的弯曲。如果使用经典的 L-系统, 要表示侧枝的弯曲性必须再增加产生式。因此, 函数 L-系统通过在字符中增加曲率函数产生了弯曲的树枝, 而不需要再增加产生式。

实验 2 经典的 L-系统:

$$\begin{aligned} \omega: & X \\ P_1: & X \rightarrow F[-X][+X]FX \\ P_2: & F \rightarrow FF \end{aligned} \quad (7)$$

所画出的树如图 5(a)所示。旋转角为 $\delta = 25.7^\circ$ 。如果把 L-系统式(7)改为函数 L-系统式(8):

$$\begin{aligned} \omega: & X(\varphi(s)) \\ P_1: & X(\varphi(s)) \rightarrow F[-\cap X(\varphi(s))][+\cup X(1, \varphi(s))]FX \\ P_2: & F \rightarrow FF \end{aligned} \quad (8)$$

其中 $\delta = 22.5^\circ$, 右枝: $\alpha_0 = (\sin(\delta), \cos(\delta))$, $\beta_0 = (-\cos(\delta), \sin(\delta))$, 左枝: $\alpha_0 = (\sin(-\delta), \cos(\delta))$, $\beta_0 = (\cos(\delta), \sin(\delta))$, $\varphi(s) =$

$$\begin{cases} \frac{5}{2}s, & 0 < s \leq 0.8 \\ -\frac{5}{3}(s-2), & 0.8 < s \leq 2, \text{ 主干中下一级内节点的长度与本级内} \\ 0, & \text{其他} \end{cases}$$

节点的长度的比为 4:5, 主干的一个内节点顶端生成的侧枝的第一个内节点的长度与该主干内节点长度的比为 2:3, 同一级侧枝的下一个内节点的长度与本级内节点的长度为 2:3。函数 L-系统式(8)所绘制的树如图 5(b)。

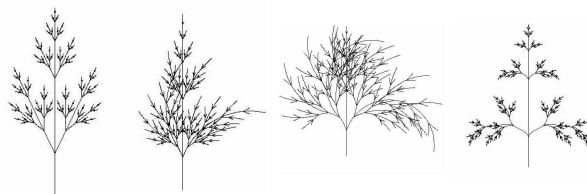
函数 L-系统:

$$\begin{aligned} \omega: & X(\varphi(s)) \\ P_1: & X(\varphi(s)) \rightarrow F[-\cap X(\varphi(s))][+\cap X(1, \varphi(s))]FX \\ P_2: & F \rightarrow FF \end{aligned} \quad (9)$$

如果 $\delta = 22.5^\circ$, 右枝: $\alpha_0 = (\sin(\delta), \cos(\delta))$, $\beta_0 = (-\cos(\delta), \sin(\delta))$, 左枝: $\alpha_0 = (\sin(-\delta), \cos(\delta))$, $\beta_0 = (-\cos(\delta), \sin(-\delta))$,

$$\varphi(s) = \begin{cases} \frac{5}{2}s, & 0 < s \leq 0.8 \\ -\frac{5}{3}(s-2), & 0.8 < s \leq 2, \text{ 主干中下一级内节点的长度与本} \\ 0, & \text{其他} \end{cases}$$

级内节点的长度的比为 4:5, 主干的一个内节点顶端生成的侧枝的第一个内节点的长度与该主干内节点长度的比为 4:5, 同一级侧枝的下一个内节点的长度与本级内节点的长度的比为 4:5, 则函数 L-系统式(9)所绘制的树如图 5(c); 如果 $\delta = 22.5^\circ$, 右枝: $\alpha_0 = (\sin(\delta), \cos(\delta))$, $\beta_0 = (-\cos(\delta), \sin(\delta))$ 左枝: $\alpha_0 = (\sin(-\delta), \cos(\delta))$, $\beta_0 = (-\cos(\delta), \sin(-\delta))$, 第 i 级侧枝的曲率函数 $\varphi(s) = \frac{10}{i} \cdot C_{7s}^i (1-s)^{7-i}$, 主干中下一级内节点的长度与本级内节点的长度的比, 主干的一个内节点顶端生成的侧枝的第一个内节点的长度与该主干内节点长度的比, 同一级侧枝的下一个内节点的长度与本级内节点的长度的比, 按函数 L-系统重写后各内节点形成的自然比计算, 则函数 L-系统式(9)所绘制的树如图 5(d)。



(a)经典 L-系统绘制的树 (b)树枝的曲率函数是分段函数, 下级主干和上级主干的比不同的树 (c)树枝的曲率函数是分段函数, 下级主干和上级主干的比相同的树 (c)树枝曲率函数是多项式的树

图 5 用改进的 L-系统绘制的拓扑结构相同, 形态差异较大的树

使用经典 L-系统绘制出的树较呆板, 初始元给定后, 树的形状只能靠调整龟前进的方向和距离来改变树的形状。而函数 L-系统, 不仅可以调整龟前进时的方向角和距离, 而且可以使用不同的曲率函数和使用不同的弯曲方向来改变树的形状, 同一个树可以使用同一个曲率函数, 也可以使用不同的曲率函数。图 5(a), 图 5(b), 图 5(c)是同一个树的所有枝使用的是同一个曲率函数, 图 5(d)是同一个树不同的枝使用不同的曲率函数。由图 5 可以看出, 尽管植物的拓扑结构相同, 但树的整体形状有很大的不同。如果再考虑随机性, 将会得到更多的形式自然的植物。

5 结论

通过把二维 L-系统理论和曲线理论相结合, 在 L-系统的产生式中增加描述弯曲程度的曲率函数, 在字符表中增加表示弯曲方向的字符, 改进了 L-系统, 得到了函数 L-系统。函数 L-系统的各个参数解释自然, 在编程中参数容易控制, 从而能够方便地绘制弯曲枝条的植物。通过实验发现使用函数 L-系统能够绘制出拓扑结构相同, 但整体形状有很大差异的植物。

参考文献:

[1] 赵星, 熊范纶, Reffye P. 一种新的植物枝条弯曲生成算法[J]. 中国科学技术大学学报, 2001, 31(6): 714-720.