# 第7讲 分形实验

## 实验目的

1.了解分形几何的基本理论；

2.了解通过迭代方式，产生分形图的方法；

3.了解分形几何的简单应用。

## 基本概念

### 迭代：

（1）图形迭代

给定初始图形，以及一个替换规则，将反复作用在初始图形上，产生一个图形序列：



（2）函数迭代

给定初始值，以及一个函数，将反复作用在初始值上，产生一个数列：



### 分形几何

1. 起源：

分形几何对象一般具有无限精细的自相似的层次结构，即局部与整体的相似性，早在19世纪就陆续出现了一些具有这种特性的图例，比如：瑞典数学家科赫（von Koch）设计的类似雪花和岛屿边缘的一类曲线，波兰数学家谢尔宾斯基（Sierpinski）设计的类似地毯和海绵一样的几何图形，英国植物学家布朗通过观察悬浮在水中的花粉的运动轨迹，提出来的布朗运动轨迹，德国数学家维尔斯特拉斯（Weierestrass）构造的处处连续但处处不可微的函数，等等。

1. 形成：

上面提到的一些几何对象，都具有极不规则的形态，客观世界中的一些自然现象，比如：云朵、烟雾等，也具有类似的极不规则的形态，Mandelbrot 将这类几何形体称为分形(fractal)，意思就是不规则的、分数的、支离破碎的，并对它们进行了系统的研究，创立了分形几何这一新的数学分支。Mandelbrot认为海岸、山峦、云彩和其他很多自然现象都具有分形的特性，因此可以说：分形是大自然的几何学。

### 分形几何题的维数

通常的几何体具有整数维，比如：一维的线段、二维的正方形、三维的立方体，维数就是 几何体在“尺度”上的特征。对于分形中的几何对象，通常意义下的维数已经没有意义，比如Koch曲线（长度是无穷大，面积是零），用一维的“线段”去量，得数无穷大，“尺子” 太小；用二维的“正方形”去量，得数为零，“尺子”又太大，因此需要定义分形自己的维数(分数维)。分形的维数目前有多种定义，我们这里介绍相似维数。

设分形F 是自相似的，即F由m个子集构成，每个子集放大c倍后同F一样，则定义 F 的维数为：



对于通常的几何对象，采用这种方式计算出来的维数，与传统的维数是一致的，比如对正方形，将它边长 k 等份，则相似形个数，每边长放大 k 倍后与原长相同，即，显然d = 2。

人类肺的构造，从气管尖端成倍地反复分叉，是一种典型的分形，其分维数大约是 2.17。

## 实验内容

### 问题1

针对函数，取初始值 x0 = 1，做迭代，并作图观察

#### 问题分析

书上基础例题，创建一个循环函数，反复迭代即可。

#### 实验程序

function ex31(n,x0) %n-迭代次数, x0-初值

fn = [x0];

for i = 2:n

fn = [fn, 2\*sin(fn(i-1)) ] %将第 i 项添加到数组fn 中

end

plot(fn) %画曲线

#### 实验结果



**Figure 1 例题结果**

#### 结果分析

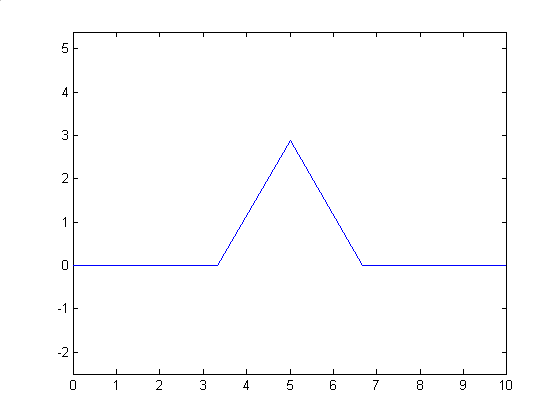
较好地反映出的迭代结果，检测前几个数，结果正确。

### 问题2

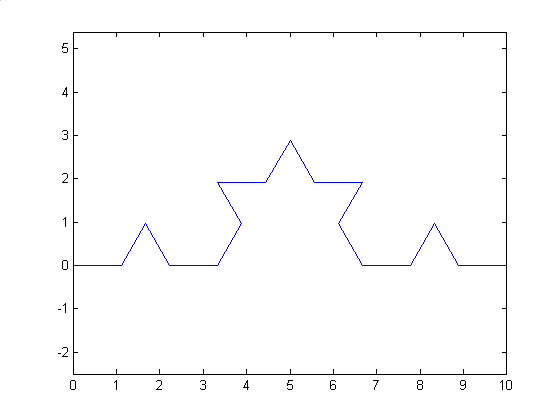
绘制科赫（Koch）曲线。

Koch 曲线是通过图形迭代的方式产生的，其迭代规则是：

对一条线段，先将它分成三等份，然后将中间的一份替换成以此为底边的等边三角形的 另外两条边。无限次迭代下去，最终形成的曲线就是 Koch 曲线。第一次迭代得Fig-2，第二次迭代得Fig-3，具体图形如下：



**Figure 2 科赫曲线 1 次迭代**



**Figure 3 科赫曲线 2 次迭代**

对Koch曲线迭代方式进行修改，由线段改为三角形，并且只画三角形。

#### 问题分析

需要一个for循环负责1: k的迭代，同时需要内嵌一个计算各个点坐标的1: n的for循环。由于每次迭代需要更新的边数和规则都不一样，因此每次执行完内嵌for循环都应该重新计算下一次循环的n值同时更新新的节点的坐标。

#### 实验程序

%%Koch曲线

function frat1(k) %显示迭代 k 次后的 Koch 曲线图

location = [0,0;10,0];

n = 1;

A = [cos(pi/3),-sin(pi/3);sin(pi/3),cos(pi/3)]; %用于计算新的结点

for a = 1:k

j = 0;

for i = 1:n %每条边计算一次

s1 = location(i,:);

s2 = location(i+1,:);

d = (s2-s1)/3;

j = j+1;

r(j,:) = s1; %原起点存入 r

j = j+1;

r(j,:) = s1+d; %新 1 点存入 r

j = j+1;

r(j,:) = s1+d+d\*A'; %新 2 点存入 r

j = j+1;

r(j,:) = s1+2\*d; %新 3 点存入 r

end

n=4\*n; %全部线段迭代一次后，线段数量乘 4

clear p

location=[r;s2]; %重新装载本次迭代后的全部结点

end

plot(location(:,1),location(:,2)) %显示各结点的连线图

axis equal %各坐标轴同比例

%%Koch曲线变形

function frat4(k) %显示等边三角形迭代k 次后的图形

A=[cos(pi/3) -sin(pi/3);sin(pi/3) cos(pi/3)]; %用于计算新的结点

p=[0 0;10 0]; %存放结点坐标

n=1;

for s=1:k

j=0;

for i=1:n

q1=p(i,:); %目前线段的起点坐标

q2=p(i+1,:); %目前线段的终点坐标

d=(q2-q1)/3;

j=j+1;a(j,:)=q1;

j=j+1;a(j,:)=q1+d;

j=j+1;a(j,:)=q1+d+d\*A';

j=j+1;a(j,:)=q1+2\*d;

end

n=4\*n;

clear p %清空 p

p=[a;q2]; %重新装载本次迭代后的全部结点

end

plot(p(:,1),p(:,2)) ;%显示各结点的连线图

fill(p(:,1),p(:,2),'k') %填充颜色

set(findobj(gcf,'type','patch'),'edgecolor','none')

axis off

axis equal

#### 实验结果

**Figure 4 科赫曲线3次迭代**

**Figure 5 科赫曲线5次迭代**

#### 

**Figure 6 三角形3次迭代**

**Figure 7 三角形5次迭代**

#### 结果分析

三次迭代和五次迭代的图像展现了Koch曲线的迭代特性，代码和注释完整地体现了一步步迭代的原理和过程。

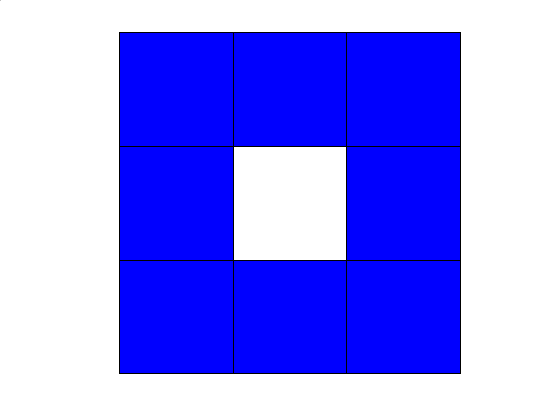
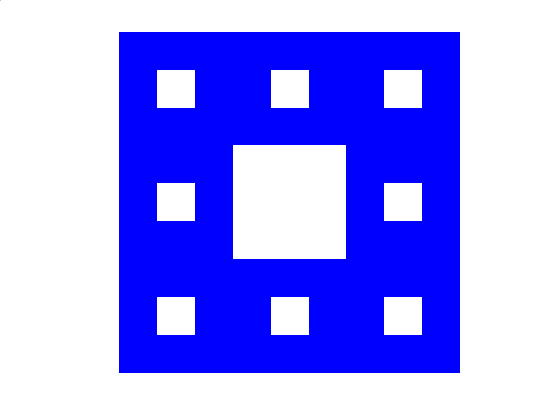
按照题目要求计算Koch曲线的长度，假设起始一条直线的长度为1，第1次迭代后，增加长度，总长度变为；第二次迭代后，增加长度为，总长度变为。一般地，n次迭代后总长度为。

按照题目要求计算Koch曲线的维度：

### 问题3

Sierpinski 地毯也是通过图形迭代的方式产生的，其迭代规则是：

对一个正方形，首先将它分成九个小正方形，然后挖掉中间的一个。无限次迭代下去，最终形成的图形就是 Sierpinski 地毯。第一次迭代得Fig-8，第二次迭代得Fig-9，具体图形如下：



**Figure 8 谢尔宾斯基地毯 2 次迭代**

**Figure 9 谢尔宾斯基地毯 1 次迭代**

#### 实验程序

function frat2(x,y,d,n)

% x 为正方形的顶点的横坐标

% y 为正方形的顶点的纵坐标

% d 为初始正方形边长

% n 为迭代次数

for p=1:n

location\_x=[];

location\_y=[];

for q=1:length(x) %每个小正方形计算一次

x1=x(q)+[0,d/3,2\*d/3,0,2\*d/3,0,d/3,2\*d/3];

y1=y(q)+[0,0,0,d/3,d/3,2\*d/3,2\*d/3,2\*d/3];

location\_x=[location\_x,x1];

location\_y=[location\_y,y1];

end

d=d/3;

x=location\_x;

y=location\_y;

end

hold on

for q=1:length(x) %用蓝色注满多边形区域

fill(x(q)+[0,d,d,0,0],y(q)+[0,0,d,d,0],'b')

end

hold off

axis off

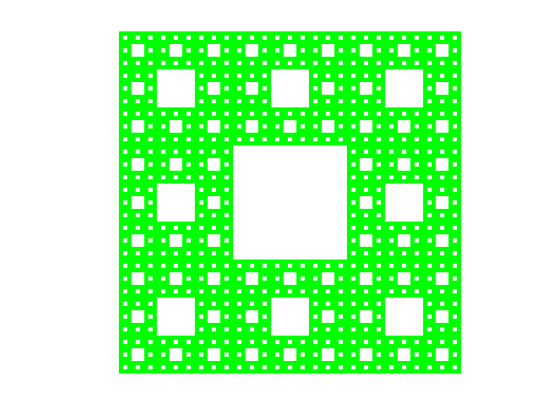
axis equal %各坐标轴同比例

set(findobj(gcf,'type','patch'),'edgecolor','none')

#### 实验结果



**Figure 10 迭代3次后图像**



**Figure 11 迭代4次后图像**

#### 结果分析

三次迭代和五次迭代的图像展现了Sierpinski图形的迭代特性，代码和注释完整地体现了一步步迭代的原理和过程。

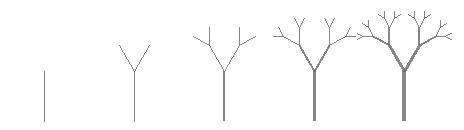
按照题目要求计算Sierpinski图形的面积，假设起始面积为1，第1次迭代后，减少，剩余；第二次迭代后，减少为，剩余面积变为。一般地，n次迭代后面积为。

按照题目要求计算Sierpinski图形的维度：

### 问题4

分形树木是为了模拟自然界中树木花草的形状，其迭代方式与前面两个问题不同，主要原理是设定基本的绘图规则，然后让计算机根据这些规则进行反复跌代，最终生成分形图。

给定迭代规则：对一条线段，在它顶端左右两侧各画一条小线段。无限次迭代下去，最终形成的图形就像一棵树。具体的实现如下Fig-12：



**Figure 12 实验4图**

#### 实验程序

%%分形树木

function [z,A]= frat3(z,A,n);

N=6; % 最大迭代次数

s=0.7;

Br=[pi/10;-pi/10]; % 偏转角

if n>N % 超出迭代次数

return

end

k=1;

if n==1; % 第 1 次，画主干

plot([z,z+exp(i\*A)],'linewidth',3/N\*(N-n+1));

pause(1)

hold on

z=z+exp(i\*A);

[z,A]= frat3(z,A,n+1);

else

while k<3

A=A+Br(k);

leng=s^(n-1);

plot([z,z+leng\*exp(i\*A)],'linewidth',3/N\*(N-n+1)); %[z,z+leng\*exp(i\*A)]画图

pause(1)

z=z+leng\*exp(i\*A);

[z,A]= frat3(z,A,n+1);

A=A+pi;

z=z+leng\*exp(i\*A); A=A+pi-Br(k); k=k+1;

end

end

#### 实验结果



**Figure 13 一层主干**



**Figure 14 两层主干**

#### 结果分析

分形树木是最常见的迭代绘图案例，通过设置while循环并赋予一定的规则（先画左枝再画右枝）可以画出合适的分形图，利用pause函数可以清晰看出迭代规则。

### 问题5

考虑如下的复函数迭代：



其中，Z 为复数，C 为复常数，迭代的初始值为Z0 。

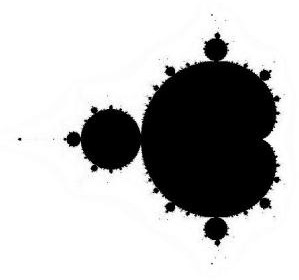
这个迭代产生的序列{Zk }可能有界，也可能无界。它的有界性既与复常数 C 有关，也与初始值 Z0 有关。

Mandelbrot 集：

固定 Z0，考虑使得迭代序列{Zk } 有界的所有 C 的集合。



为了分析满足上述条件的 C 在复平面上的分布，必须借助计算机绘图的手段，在复平面绘制此集合的图形，观察发现它具有典型的分形结构。这个集合被称为 Mandelbrot 集，图形如下：



**Figure 15 实验五图**

#### 实验程序

function frat5(res,iter,xc,yc,xoom)

% res 是目标分辨率，iter 是循环次数，

%（xc,yc）是图像中心，xoom 是放大倍数

x0=xc-2/xoom;

x1=xc+2/xoom;

y0=yc-2/xoom;

y1=yc+2/xoom;

x=linspace(x0,x1,res);

y=linspace(y0,y1,res);

[xx,yy]=meshgrid(x,y);

z=xx+yy\*1i;

C=z;

N=zeros(res,res); %初始化N，最终根据N，对各点进行染色

tic %启用秒表计时器

for k=1:iter

z=z.^2+C;

N(abs(z)>4)=k; %逃逸半径为 4，逃逸时间 k,未逃逸为 0

z(abs(z)>4)=0;

C(abs(z)>4)=0;

end

imshow(N,[]);

toc %读取已用时间

#### 实验结果



**Figure 16 实验五实验结果**

时间已过 2.161621 秒。

#### 结果分析

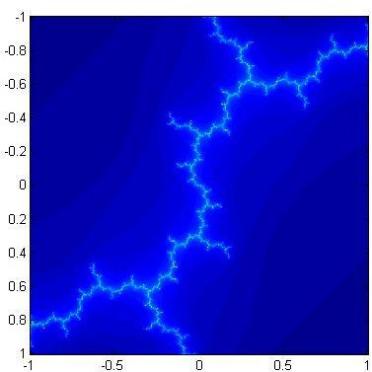
较好的展现了Julia 集的迭代情况，通过设置不同的迭代次数会有不一样的结果。

### 问题6

固定 C，考虑使得迭代序列有界的所有的集合。



为了分析满足上述条件的Z0在复平面上的分布，必须借助计算机绘图的手段，在复平面绘制此集合的图形，观察发现它也具有典型的分形结构。这个集合被称为 Julia 集，图形如下：



**Figure 17 实验六题图**

#### 实验程序

function frat6(c, res, iter, xc, yc, xoom)

%Julia 集

x0=xc-2/xoom;

x1=xc+2/xoom;

y0=yc-2/xoom;

y1=yc+2/xoom;

x=linspace(x0, x1, res);

y=linspace(y0, y1, res);

[xx, yy]=meshgrid(x, y);

z=xx + yy\*1i;

N=zeros(res, res);

C=c\*ones(res, res);

for k=1:iter

z=z.^2+C;

N(abs(z)>2)=k;

C(abs(z)>2)=0;

z(abs(z)>2)=0;

end

colormap jet %colormap map将当前图窗的颜色图设置为预定义的颜色图之一，jet以一个三列数组的形式返回 Jet 颜色图

image(x, y, N); %定坐标，确定像素颜色

axis square %axis square 当前坐标系图形设置为方形axis equal 将横轴纵轴的定标系数设成相同值

#### 实验结果



**Figure 18 循环次数10 放大倍数2图像**



**Figure 19 循环次数200 放大倍数2图像**



**Figure 20 循环次数200 放大倍数5图像**

#### 结果分析

较好的展现了Julia 集的迭代情况，通过设置不同的颜色图Jet、调整迭代次数会有不一样的结果。

## 实验感想

本次案例学习中，我按时上线接收文件，细致地观看了PPT和电子课本。通过本次对PPT、附件和电子课件附录的学习，我大致理解了分形的原理及一些典型案例

我认为这次实验是非常有意义有价值的，通过这次案例学习，我对初步掌握了几种迭代绘图的方法。虽然这次案例较难，部分代码依然存在问题，但经过一下午的学习，确实收获不少。在这次实验中涉及到的疑难知识点、用到的新函数（tic toc colormap jet image imshow meshgird set等等）我都通过记笔记或者录屏的方式认真记了下来，丰富了我的MATLAB知识储备。在本次实验中，所有的实验均由我独立完成，相关代码和图片结果也都整理到位，代码中存在疑惑的地方以及需要注意的地方均已注释好，以备下次复习时使用。在这次实验里，我认真完成了相关实验任务，颇有所获，剩下存在的问题我会在课外作业里面继续解决。

另外，这次案例的代码理解起来需要花费较多时间，自学也带来很多不便。特殊时期，在家自学确实会有许多不方便的地方，也会花更多的时间，希望老师能适当增加平时分占比，期末实验也请手下留情，谢谢老师！

6 许柏城 62号 课外实验7

2020-04-23 19:00