

# Mandelbrot Set 的生成和探索

洪艺中

数学与应用数学 3190105490

2022 年 7 月 1 日

## 摘要

本文介绍 Mandelbrot 集的定义以及其与有理函数迭代中收缩循环的关系. 随后给出其图像的一种基于迭代法的生成方式, 并简要分析迭代次数对所得图像性质的影响, 以及其中不同区域的数学含义.

## 1 引言

Mandelbrot 集是最著名的分形图形之一. 在 1978 年, Robert W. Brooks 和 Peter Matelski 首次定义并画出了这个集合, 这也是他们克莱因群研究的一部分. Mandelbrot 集有一个无限复杂的分形边界, 在放大的过程中会逐渐显现更多自相似的细节. 在数学研究之外, Mandelbrot 集靠它无限递归的美感获得盛名, 它也是数学可视化和数学美学的最著名的例子之一 [5].

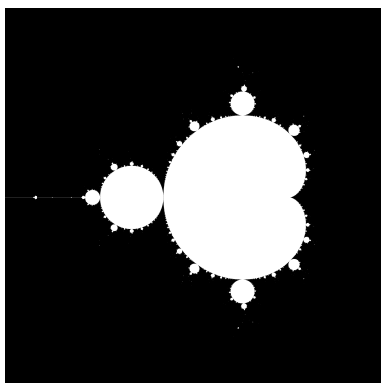


图 1: Mandelbrot 集图

## 2 问题的背景介绍

Mandelbrot 集的来源可以联系到有关有理函数的迭代问题上. 有理函数的迭代是许多迭代函数中十分常见的, 例如牛顿迭代法求解多项式  $f(x)$  的零点时, 会用到

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (1)$$

式. 其中右端项即是一个有理函数  $g(x) = x - f(x)/f'(x)$ . 每次使用牛顿法时, 我们会选取一个初值来迭代, 因此就会有这样一个问题: 牛顿法对任何初值都收敛吗? 如果有不收敛的点, 选到这些点的概率是多

少？这两个问题的答案决定了我们能否自由地使用牛顿法 — 如果牛顿法由很高的概率选到一个不收敛的初值，那么它就不好用了。所以研究有理函数的迭代性质是很有意义的。

上述的问题可以推广到一般的有理函数  $g(x)$ ，研究的初值空间也可以推广到复平面上：我们希望研究迭代方程  $z_{n+1} = g(z_n)$  的变化特点。实际上有一些特殊点的迭代性质是容易掌握的，例如方程  $z = g(z)$  的根（因为是有理函数，所以必然存在）是这个迭代方程的不动点，那么自然是收敛的。因此我们就可以研究，不动点附近的点是会随着迭代趋近这个不动点，还是会发散开呢？实际上，这可以通过计算  $g'(z)$  的值来判定 [1]。

除了不动点，我们也可以考虑经过  $n$  次变化回到原点的“循环点列”，即方程  $f(f(\cdots(f(z))\cdots)) = z$  的零点。如果初值选在这些点上，那么显然迭代会出现循环因而不收敛。如果只有这些点会导致不收敛其实是安全的，因为选到它们的概率为 0。但是我们还会担心，这些点附近的点也会随着循环而动吗？还是很快也会收敛到其他点呢？

如果一个循环点列附近的点也会一直随着循环而动（更准确地说，他们也会逐渐逼近那些循环点），那么可以称之为一个**收缩循环**。收缩循环意味着平面上有一些开集，其中点作为初值是不收敛的，也就说明算法可能会不收敛。因此我们希望找到一些方法，在算法陷入无限的循环之前就判定出函数  $g(x)$  是否存在收缩循环。

幸运的是，Fatou[3, 4] 证明了：**如果有理函数  $g(z)$  存在上述的收缩循环，那么其稳定点，即  $g'(z) = 0$  的零点，一定会收敛到这些收缩循环上。**也就是说，我们只需要检测稳定点在迭代中的表现，就可以判定函数是否存在收缩循环。

因此如果我们希望知道  $g(z) = z^2 + c$  这个最简单的有理函数（之一）是否存在收缩循环，只需要检测 0 迭代的表现即可 — 这也就引出了本文中的 Mandelbrot 集，它正是在研究  $g(z) = z^2 + c$  从 0 开始迭代的表现。它实际上也给出了关于  $g(z)$  迭代性质的信息。

### 3 数学理论

本文着眼于研究 Mandelbrot 集图像的绘制而非不动点问题，因此这一部分主要列出 Mandelbrot 自身的性质 [5]。

Mandelbrot 集的定义为  $\{c \in \mathbb{C} : \text{以 } 0 \text{ 为初值的迭代 } z_{n+1} = g(z_n) \text{ 始终有界}\}$ 。当  $|c| > 2$  时，迭代一定会发散，因此 Mandelbrot 集包含在原点为圆心，半径为 2 的圆中。

此外还有一些有趣的性质，例如 Mandelbrot 实际上是一个连通集（虽然在计算机模拟中常常会因为精度问题而变得不连通），并且其中中心的“最大的”部分的边界确实是一条心形线。它对应了  $g(x)$  的不动点满足第 2 节中所说的“收缩”性质的点集的边界 [1]。

### 4 算法

我们模拟迭代过程来判断点是否属于 Mandelbrot 集。迭代次数越多，图像越准确（但连通性可能因为像素限制而破坏）。这个算法来自 [2]。

```

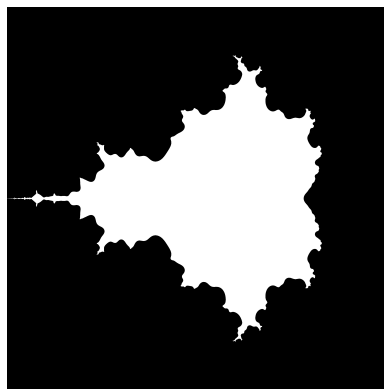
for each pixel p of the image do
   $c \leftarrow$  Complex Represented By  $p$ ;
   $z \leftarrow 0$ ;
   $\text{flag} \leftarrow \text{true}$ ;
  while  $t < \text{MAX ITERATION TIME}$  do
    if  $|z| > 2$  then
       $\text{flag} \leftarrow \text{false}$ ;
      Break;
    else
       $z \leftarrow z^2 + c$ ;
       $t \leftarrow t + 1$ ;
    end
  end
  Set color black if flag is true, white if false;
end

```

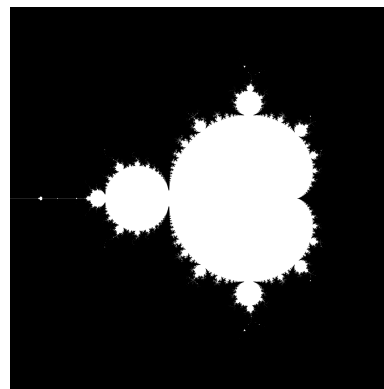
**Algorithm 1:** 判断点是否属于 Mandelbrot 集

## 5 数值算例

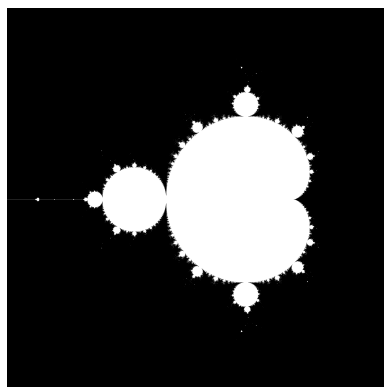
使用 c++ 实现上述的算法, 生成了不同迭代上限下的图片:



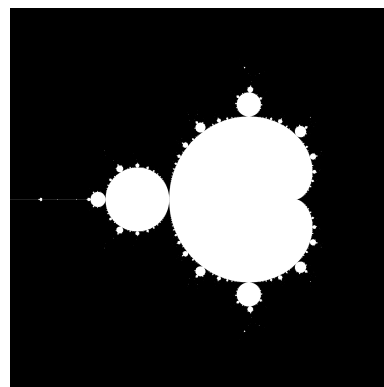
(a) 迭代 10 次



(b) 迭代 50 次



(c) 迭代 100 次



(d) 迭代 10000 次

图 2: 不同迭代次数得到的图像

可以看到, 随着迭代次数的增加, 图像边界的细节也逐渐增多. 但是迭代次数过多, 由于精度问题, 图像会变得不连通.

我们也记录其迭代次数, 并按照这个数目为 Mandelbrot 集外的点染色, 以期看出随着迭代整体的变化过程. 这样可以得到一幅有艺术气息的图片, 如图 3.

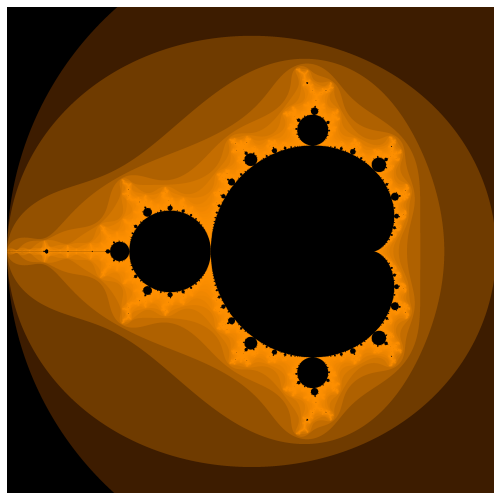


图 3: 按迭代次数染色

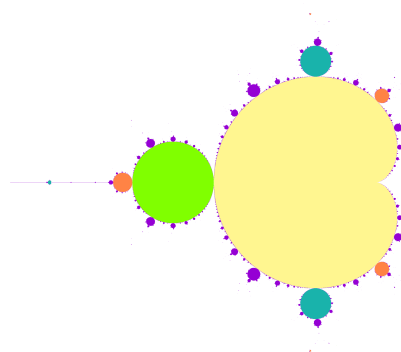


图 4: 按循环特征染色

最后, 我们也对 Mandelbrot 的点做了循环测试, 即判定迭代有界的点趋向于怎样的循环. 结果如图 4.

不同的颜色表示对应区域的循环为1-循环, 2-循环, 3-循环, 4-循环或是5-循环及以上. 这也呼应了我们第 2 节中介绍的关于迭代中收缩循环的判定, 通过这个图就可以判断对应的  $z^2 + c$  的迭代有怎样的收缩循环.

## 6 结论

生成 Mandelbrot 集图的方法较为简单, 但是精确的边界依赖大量的迭代, 因此寻求边界的解析解对绘图和研究都更为重要. 此外没有其他结论了.

## 参考文献

- [1] 3Blue1Brown. Beyond the mandelbrot set, an intro to holomorphic dynamics - youtube. <https://youtu.be/LqbZpur38nw>, 2021.
- [2] Arnaud Cheritat. Mandelbrot set - techniques for computer generated pictures in complex dynamics. [http://www.math.univ-toulouse.fr/~cheritat/wiki-draw/index.php/Mandelbrot\\_set](http://www.math.univ-toulouse.fr/~cheritat/wiki-draw/index.php/Mandelbrot_set), 2016. [Online; accessed 30-June-2022].
- [3] James Curry, Lucy Garnett, and Dennis Sullivan. On the iteration of a rational function: Computer experiments with newton's method. *Communications in Mathematical Physics*, 91, 06 1983.
- [4] Pierre Fatou. Sur les équations fonctionnelles. *Bulletin de la Société Mathématique de France*, 48:161–271, 1919.
- [5] Wikipedia contributors. Mandelbrot set — Wikipedia, the free encyclopedia. [https://en.wikipedia.org/wiki/Mandelbrot\\_set](https://en.wikipedia.org/wiki/Mandelbrot_set), 2022. [Online; accessed 30-June-2022].