МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

ФРАКТАЛЫ

άγεωμέτρητοζ μηδείζ είσιτω

© В. И. Зенкин, 2008 г.

К понятию *фрактала* Б. Мандельброт (Benuit Mandelbrot) пришёл, исходя из найденных им решений вполне реальных практических задач.

К одной из таких задач относился на первый взгляд довольно простой вопрос: какова длина береговой линии Великобритании? Ясно, что практические измерения зависят от величины мерки и всегда будут приблизительны, тем точнее, чем меньше мерка. Например, если мерить береговую линию циркулем, разведя его ножки на один метр, то результат будет меньше, чем при измерении тем же циркулем, разведённым на 10 см, т. к. в первом случае будут «перешагиваться» повороты и изгибы, меньшие метра.Чем меньше раствор циркуля, тем больше будет учтено деталей и тем точнее измерение. Можно предположить, что результаты последовательности измерений с применением все меньшей и меньшей мерки будут сходиться к некоторому конечному значению, которое и следует считать ответом на поставленный в задаче вопрос.

Если бы береговая линиия имела «правильную» форму, вроде окружности или треугольника, то так бы и было. Однако, Мандельброт показал, что при бесконечном уменьшении мерки длина береговой линии неограниченно растёт. Хорошей математической моделью береговой линии является *кривая Коха*.

Рис. 1. Кривая Коха

Термин *фрактал* Мандельброт образовал от латинского слова fractus — сломанный, кстати, созвучному английскому fraction — частица, фракция, дробь. Фрактальные множества были введены и применяются с целью описания формы таких природных объектов, как облако, гора, дерево, береговая линия, система кровеносных сосудов, турбулентный водный поток и т. п. — объектов, для моделирования которых классическая геометрия Евклида оказалась малопригодна. Характерными чертами фрактальных множеств является их, как правило, *дробная* размерность — понятие, впервые сформулированное Ф. Хаусдорфом, и *самоподобность*. Определение, данное самим Мандельбротом [1, стр. 31], таково:

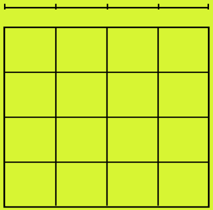
> Фракталом *называется множество, размерность Хаусдорфа–Безиковича для которого строго больше его топологической размерности.*

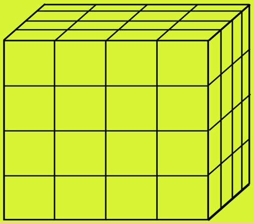
Позднее им же было сформулировано более узкое определение:

> Фракталом *называется структура, состоящая из частей, которые в каком-то смысле подобны целому.*

Строгого общепринятого определения фракталов в настоящее время не существует, оба приведённых выше определения во многих задачах и физических приложениях оказываются либо слишком ограничительными, либо не совсем полными. Точные определения размерностей и всех связанных с ними понятий см., например, в [1], [3].

Для лучшего понимания введённых понятий рассмотрим следующие элементарные построения [2]. Вначале разделим единичный отрезок прямой на N равных частей. Пусть при этом каждый из полученных отрезков меньше исходного в $1/r$ раз, тогда $Nr = 1$. Далее, квадрат со стороной, равной единице, разобъём на N равных квадратов с площадью в $1/r^2$ меньшей площади исходного, при этом $Nr^2 = 1$. Наконец, такую же процедуру проделаем с единичным кубом, получив кубы объёмом в $1/r^3$ меньше исходного. Тогда, очевидно, что





$$Nr^d = 1, \tag{1}$$

где d — размерность объекта, в рассмотренных случаях это — целое положительное число, $d = 1, 2, 3$. Тем не менее, можно построить объект, размерность которого будет выражаться *дробным* числом. Такое множесто и называют фракталом. Из (1) следует

$$d = \frac{\ln N}{\ln \dfrac{1}{r}}. \tag{2}$$

В качестве примера рассмотрим кривую Коха. Из исходного отрезка K_0 на первом этапе построения удалим среднюю треть и вместо неё построим две стороны равностороннего треугольника. Полученное множество обозначим K_1. На каждом отрезке K_1 проделаем такое же построение, получив т. о. множество K_2, и т. д., как на рис. 1.

Предел последовательности K_n при $n \to \infty$ называется кривой Коха (такой предел действительно существует, но на доказательстве данного факта мы останавливаться не будем). Заметим, что любое множество K_n *самоподобно*, т. к. составлено из уменьшенных в три раза четырёх множеств K_{n-1} (см. рис. 1), следовательно, для него $r = 1/3$, $N = 4$ и размерность кривой в соответствии с формулой (2) есть дробное число

$$d = \frac{\ln 4}{\ln 3} \approx 1{,}2618.$$

Интересно также отметить, что длина кривой Коха равна бесконечности, хотя сама она занимает ограниченный участок плоскости. Действительно, очевидно, что если принять $|K_0| = 1$, то

$$|K_n| = \left(\frac{4}{3}\right)^n \Rightarrow \lim_{n\to\infty} |K_n| = \infty.$$

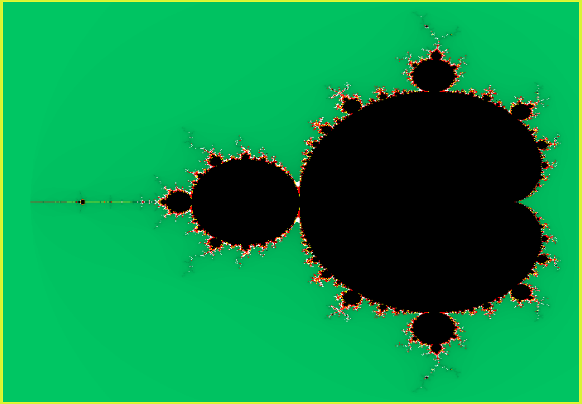
Рассмотрим другой пример — *множество Мандельброта* M, определяемое как множество точек c комплексной плоскости, не уходящих в бесконечность при любом количестве итераций

$$z_{k+1} = z_k^2 + c, \quad z_0 = (0, 0). \tag{3}$$

Таким образом,

$$M = \{c \in \mathbb{C} : z_k \nrightarrow \infty, \text{ при } k \to \infty\}.$$

Конечно, реально можно построить только приближение к множеству Мандельброта, тем более точное, чем больше итераций в (3) будет сделано.



Множество Мандельброта, несмотря на очень простое определение, имеет необычайно сложную самоподобную структуру. На рис. точки, принадлежащие множеству, окрашены в чёрный цвет, а все другие имеют различные оттенки, зависящие от количества итераций, сделанных для этих точек. Можно показать [2, стр. 232–235], что достаточно проверить только точки, для которых $|z_k| \leqslant 2$. Отметим, что визуализация множества Мандельброта, как и многих других фрактальных объектов, требует довольно большого объёма вычислений и была практически невозможна до появления достаточно мощных компьютеров.

L-системы. L-системы были впервые введены в рассмотрение А. Линденмейером (A. Lindenmayer) в 1968 году для моделирования развития живых организмов. С их помощью легко построить многие известные самоподобные фракталы, они часто применяются в компьютерной графике для получения фрактальных деревьев, растений и т. п. [2]. L-система представляет собой набор символов и правил, задаваемых текстовой записью на формальном языке, для графического представления объекта на плоскости или в пространстве. Далее будут рассмотрены т. н. *детерминированные* L-системы на плоскости.

Удобно считать, что графические построения реализуются при помощи специального *исполнителя*, который в каждый момент времени «знает» свои координаты, направление движения и длину отрезка (длина отрезка постоянна), на величину которого он может переместиться. Управляется исполнитель следующими командами:

F	—	рисовать заданный отрезок в заданном направлении;
f	—	передвинуться в заданном направлении на заданный отрезок расстояния без рисования;
+	—	изменить текущее направление против часовой стрелки на заданный угол;
−	—	изменить текущее направление по часовой стрелке на заданный угол;
$\vert$	—	реверс (поворот на 180°);
[	—	поместить в стек текущие координаты и направление;
]	—	вытолкнуть из стека координаты и направление и сделать их текущими;

{	—	начало записи L-системы;
}	—	конец записи L-системы;
;	—	комментарий до конца строки.

Спецификация входных данных для описания L-систем представляет собой простой текстовый файл. В начале, до символа `{`, принято задавать имя объекта. После символа начала записи задаётся текущий угол — `Angle n`, угол равен $360^\circ/n$. Затем т. н. *инициатор* `Axiom S`, где `S` — начальное значение строки команд, содержащей один или несколько текстовых символов, которая далее будет содержать команды исполнителю. Символ `=` означает подстановку правой части вместо левой. Сама операция подстановки выполняется столько раз, сколько задаст пользователь (параметр Order). Начальное направление по умолчанию равно 0°.

Пример 1. Кривая Коха (рис. 1).

```
koch {
        Angle  3          ; угол = PI/3
        Axiom  F          ; стартовый символ
        F = F+F--F+F
     }
```

При Order $= 1$ строка команд имеет вид `F+F--F+F`, при Order $= 2$

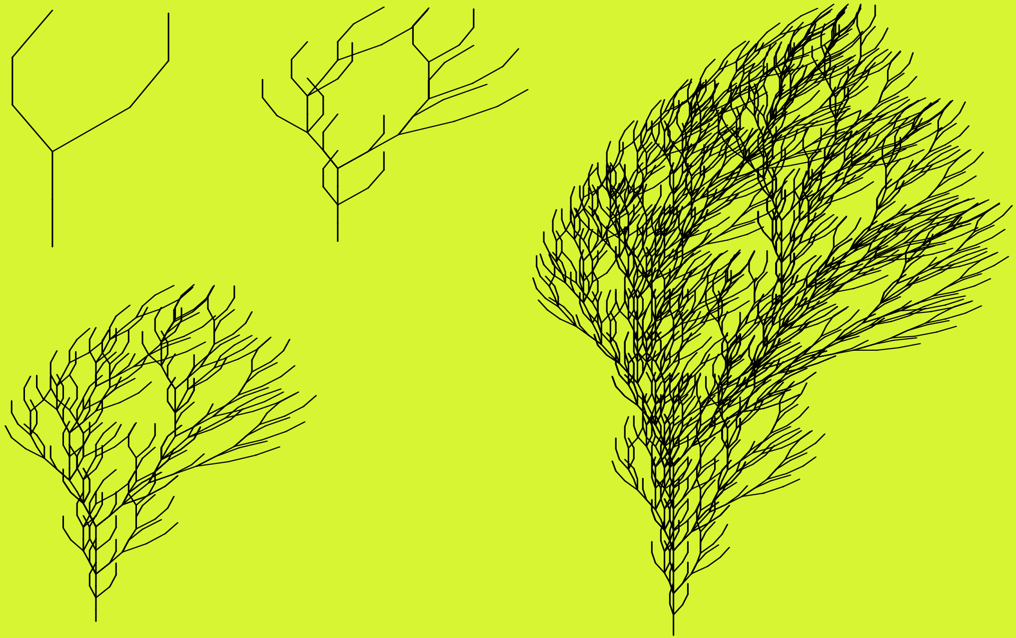
```
F+F--F+F + F+F--F+F -- F+F--F+F + F+F--F+F
```

и т. д. Соответственно будет меняться изображение кривой Коха, как показано на рис. 1.

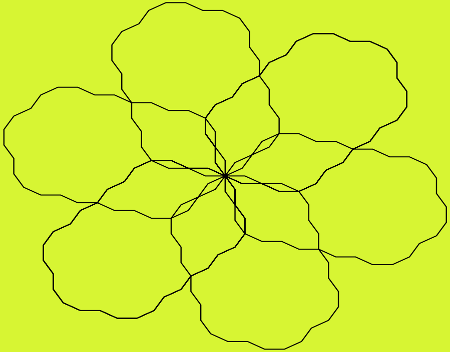
назад
закр.

Пример 2. «Дерево» (Order = 1, 2, 3, 4).

```
tree{
        Angle  16                         ; угол = PI/16
        Axiom  F                          ; стартовый символ
        F = FF-[-F+F+F]+[+F-F-F]
    }
```



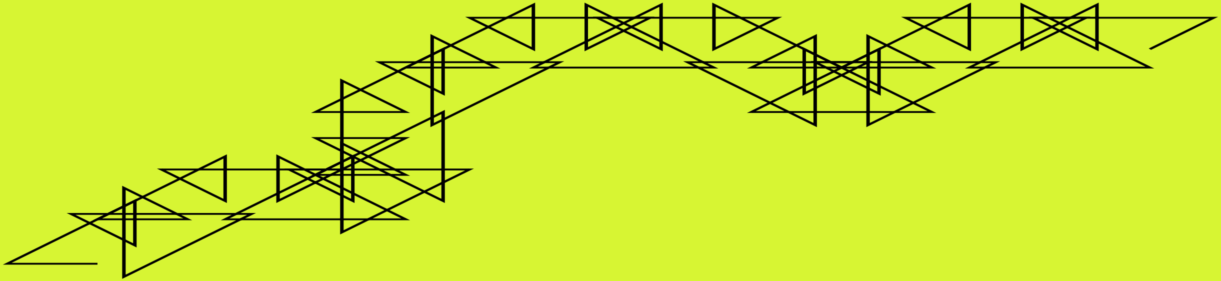
Пример 3. «Цветок» (Order = 4).



```
flower{
          Angle  6
          Axiom  F
          F = F+F-F+F
      }
```

Пример 4. «Орнамент» (Order = 4).

```
ornament{
          Angle  4
          Axiom  F
          F = -FF---FF+
      }
```



Система итерирующих функций (IFS — Iterated Function System) является совокупностью аффинных преобразований, задающихся системой уравнений

$$x_{n+1} = ax_n + by_n + c;$$
$$y_{n+1} = dx_n + ey_n + f,$$

Рис. 2. Папортник

причём если коэффициенты a, b, c, d, e, f выбираются на каждой итерации случайным образом, с определёнными вероятностями, то такую систему называют *Random Iterated Function System* (RIFS). При помощи RIFS, в частности, моделируются изображения различных природных живых растительных форм. Впервые такой подход к моделированию применил М. Барнсли (M. Barnsley). Классический пример — лист папортника — получается при следующих значениях величин (p — вероятность)

a	0,0	0,2	-0,15	0,75
b	0,0	-0,26	0,28	0,04
c	0,0	0,23	0,26	-0,04
d	0,16	0,22	0,24	0,85
e	0,0	0,0	0,0	0,0
f	0,0	1,6	0,44	1,6
p	0,1	0,08	0,08	0,74

Задачи

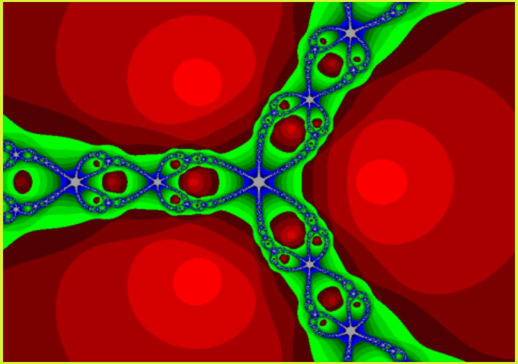
1. Фрактал Ньютона. Метод Ньютона для решения уравнения вида

$$f(z) = 0,$$

где z — комплексная переменная, $f(z) = a_0 + a_1 z + \cdots + a_k z^k$, заключается в нахождении последовательных приближений к решению по формуле

$$z_{n+1} = z_n - \frac{f(z_n)}{f'(z_n)}. \qquad \textbf{(4)}$$

Начальное приближение z_0 может быть произвольным. Написать программу, которая закрашивает каждую точку заданной прямоугольной области комплексной плоскости цветом, зависящим от количества итераций в (4), нужных для достижения решения с заданной точностью, если эту точку выбрать в качестве начального приближения. Цвета подобрать по своему усмотрению. Входные параметры: x_0, x_m, y_0, y_m (определяют прямоугольную область), максимальное количество итераций и точность решения уравнения (задаются пользователем). Если взять $f(z) = z^3 - 1$, то получится приблизительно такая картинка, как на рисунке.



2. Написать программу для генерации изображений фракталов, которые определяются как множество точек c комплексной плоскости, не уходящих в бесконечность при итерациях

$$z_{k+1} = p(z_k) + c,$$

где $p(z_k) = a_4 z_k^4 + a_3 z_k^3 + a_2 z_k^2 + a_1 z_k$, величины z_k, c комплексные, $a_i, i = 1, \ldots, 4$ — вещественны. При этом нужно будет решить следующие задачи:

1. оценить границы фрактального множества;
2. определить критические точки множества, решив уравнение $p'(z) = 0$;
3. фрактал строится расчётом для всех точек c, лежащих в найденных границах, начиная с z_0, совпадающей с какой-либо критической точкой, при $k = 0, \ldots, N$, где N — большое число (чем оно больше, тем чётче будет видна структура фрактала). Если при некотором k точка z_k выходит за границы множества, то счёт прекращается и точка окрашивается в зависимости от количества сделанных итераций (цвета можно устанавливать по своему усмотрению), в противном случае её цвет — чёрный, она принадлежит фракталу.

Например, для множества Мандельброта $p(z_k) = z_k^2$ и не трудно показать, что в этом случае: 1) $|c| \leqslant 2$; 2) $(0, 0)$ — критическая точка (в данном случае она единственна).

Входные параметры модели: $a_i (i = 1, \ldots, 4)$, N. Предусмотреть возможность масштабирования изображения фрактала, увеличения выделенных пользователем частей картинки, получения координат всех его точек, сохранения построенных изображений в графическом файле. Расчёты оптимизировать.

3. Фрактал Ляпунова. Рекуррентное уравнение

$$x_{n+1} = rx_n(1 - x_n), \qquad x_0 = 0{,}5 \tag{5}$$

называют *логистическим*, оно описывает, в частности, процесс изменения некоторой популяции животных. Параметр r представляет собой последовательность вещественных чисел a, b (т. е. $r =$ либо a, либо b). Показатель Ляпунова:

$$\lambda = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \log_2 \left| \frac{dx_{n+1}}{dx_n} \right|$$

Если экспонента Ляпунова < 0, то это указывает на стабильное периодическое поведение, если она > 0, то это указывает на *хаос* (или на расходящуюся модель). Составить программу для изображения фрактала Ляпунова. По осям координат плоскости откладываются значения a, b и для каждой пары (a, b) вычисляется показатель Ляпунова (вместо предела вычисляется выражение, стоящее под знаком lim, при большом значении N). Если экспонента Ляпунова < 0, то точка окрашивается в синий цвет, если она > 0 — то в красный, а если $= 0$ — в жёлтый (для большей красоты можно использовать несколько оттенков этих цветов). При расчётах по формуле (5) делать не менеее 600 итераций (для каждого из значений a, b), N можно взять равным 4000, другие два параметра a, b: $3{,}8360 < a < 3{,}8405, \;\; 3{,}845 < b < 3{,}851$.

назад
закр.

Список литературы

1. Б. Мандельброт. Фрактальная геометрия природы. М., 2002.
2. Р. М. Кроновер. Фракталы и хаос в динамических системах. М.: Постмаркет, 2000.
3. Е. Федер. Фракталы. М.: Мир, 1991.
4. `http://spanky.triumf.ca/www/fractint`
5. M. Barnsley. Fractals Everywhere, Academic Press, 1988.
6. И. Пригожин, И. Стенгерс. Порядок из хаоса, М, 1986.
7. Дж. Глейк. Хаос. Создание новой науки, СПб, 2001.
8. М. Шредер. Фракталы, хаос, степенные законы. Ижевск, 2001.
9. P. Prusinkiewicz and J. Hanan. Lindenmayer Systems, Fractals, and Plants. Springer — Verlag, New York, 1989.