Chapter 8

Fractals

8.1 Introduction

Nested patterns with some degree of self-similarity are not only found in decorative arts, but in many natural patterns as well (see Figure 8.1).

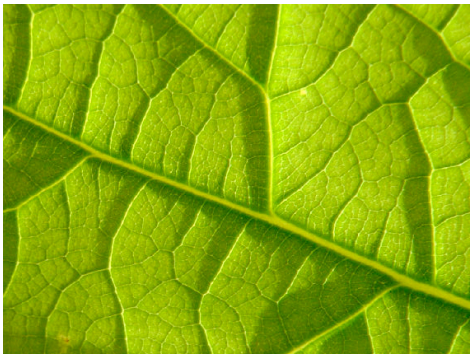
In mathematics, nested shapes began to be used at the end of the 1800's (Riemann in 1861, von Koch in 1872, Cantor in 1874, Peano in 1890) and extended in the mid-1900's (Sierpinski in 1916). In the late 1960's, the mathematician Benoît Mandelbrot showed that nested shapes can not only be identified in many natural systems, but also in several branches of mathematics. This induced increased interest in fractals to model many chaotic phenomenae with applications ranging from computer graphics, fluid dynamics, financial analysis, to data and image compression.

The term "fractals", coined in 1975 by Mandelbrot to refer to objects displaying self-similarity on different scales, has since then become very famous – probably also because of their fascinating beauty.

We have seen at different places in the previous chapters how fractals are closely related to recursion and chaos. In the present chapter, we will not only explore further systems with fractal nature and learn how to characterize some of their properties, but also draw a connection to formal languages as we encounter the so-called L-systems.

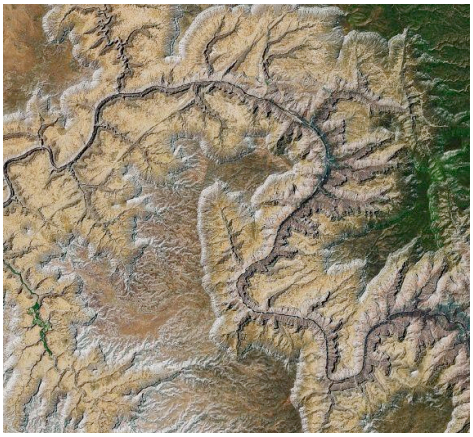
(a)

(b)



(c)

(d)



(e)

(f)

Figure 8.1: Various nested patterns with some degree of self-similarity. (a) A ceiling of the Alhambra. (b) Romanesco broccoli. (c) Leaf. (d) Fern. (e) Satellite view of the Grand Canyon. (f) Snowflake.

(a) (b) (c)

Figure 8.2: Measuring the length of the coast of Britain with different units of measurement. Notice that the smaller the ruler, the bigger the result.

8.2 Measuring the Length of Coastlines

How long is the coast of Britain?

While studying the causes of war between two countries, the mathematician Lewis Fry Richardson decided to search for a relation between the probability of two countries going to war and the length of their common border. While collecting data, he realised that there was considerable variation in the various gazetted lengths of international borders. For example, that between Spain and Portugal was variously quoted as 987 or 1214 km while that between the Netherlands and Belgium as 380 or 449 km.

What Richardson realized in his research, is that the border of a country does not necessarily have a "true" length, but that the measured length of a border depends on the unit of measurement!

For instance, the coast length of a country grows with increasing level of details, as shown in Figures 8.2 and 8.3. One could have supposed that measured lengths would converge to a finite number representing the "true" length of the coastline. However, Richardson demonstrated that the measured length of coastlines and other natural features appears to increase without limit as the unit of measurement is made smaller. This is known as the *Richardson effect*.

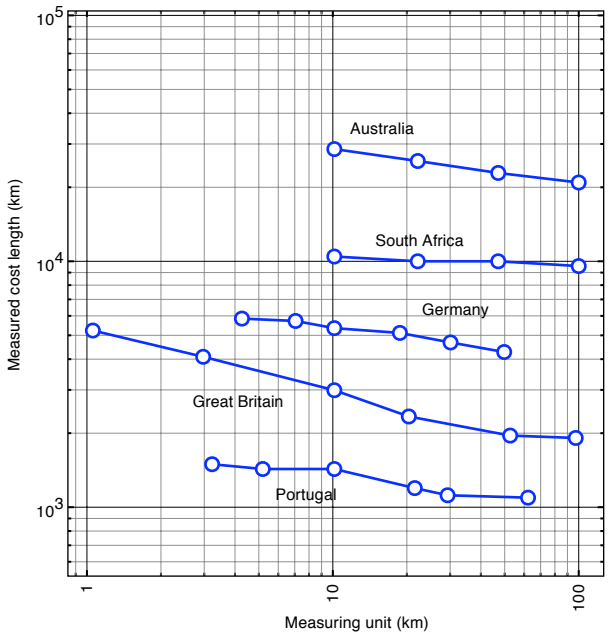


Figure 8.3: Measured length of the coastlines of different countries as function of the measuring unit. As the measurement unit is made smaller, the measured lengths do not converge to a "true" length of the coastline, but appear to increase without limit.

8.3 Fractional Dimension

We just have seen that measuring a "complex" object such as the coastline of a country leads to surprising results. But let us first consider how one can measure "normal objects".

One-dimensional object

Let us start with a one-dimensional object, such as the line shown in Figure 8.4(a). We choose the measuring unit as a square, and count how many such squares are needed to cover the line. Obviously, reducing the size of the measuring unit by a linear factor of 2 – i.e. using a measuring square with half the side length – requires approximately twice as many units to cover the line.

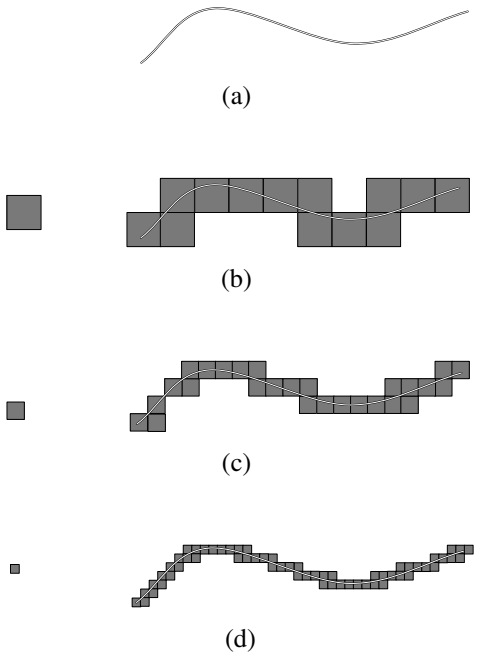
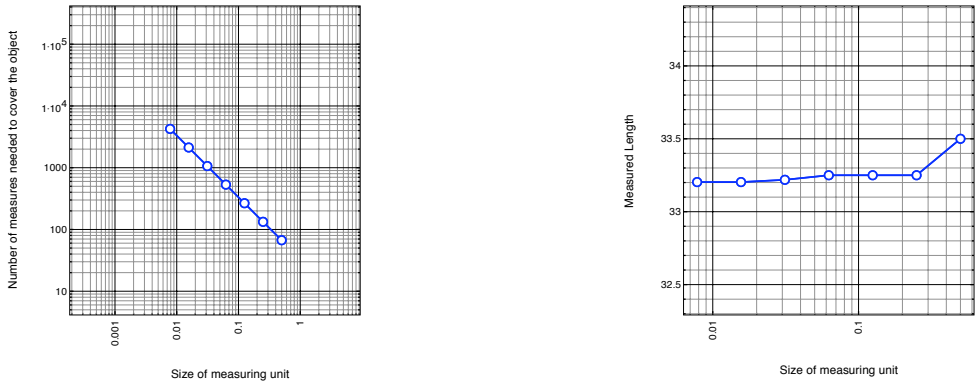


Figure 8.4: Measuring the size of a one-dimensional object, the line shown in (a), using measuring units of different sizes. Reducing the size of the measuring unit by a (linear) factor of 2 requires approximately twice as many units to cover the line.



(a) (b)

Figure 8.5: (a) Number of measuring units needed to cover the line as function of the size of the measuring unit. (b) Measured length of the line.

If a represents the size of the measuring unit (i.e. the side length of the square) and N represents the number of measuring units needed to cover the line, we have the relation:

$$N \approx c_1 \cdot \frac{1}{a} \tag{8.1}$$

with a particular constant c_1.

The plot of this relation in log-log scales in shown in Figure 8.5(a). In particular, the measured length L of the line, defined as $L = N \cdot a$ converges to a fixed value as a gets smaller, since we have:

$$L = N \cdot a \approx \left(c_1 \cdot \frac{1}{a} \right) \cdot a = c_1$$

Two-dimensional object

Let us now measure the size of a two-dimensional object, the surface shown in the top left corner of Figure 8.6, using again the same square measuring unit. This time, reducing the size of the measuring unit by a linear factor of 2 requires approximately four times as many units to cover the surface. We thus have now the relation:

$$N \approx c_2 \cdot \left(\frac{1}{a} \right)^2 \tag{8.2}$$

with a particular constant c_2.

The plot of this relation in log-log scales in shown in Figure 8.7(a). The measured area A of the surface, defined as $A = N \cdot a^2$ converges again to a fixed value as a gets smaller:

$$A = N \cdot a^2 \approx \left[c_2 \cdot \left(\frac{1}{a} \right)^2 \right] \cdot a^2 = c_2$$

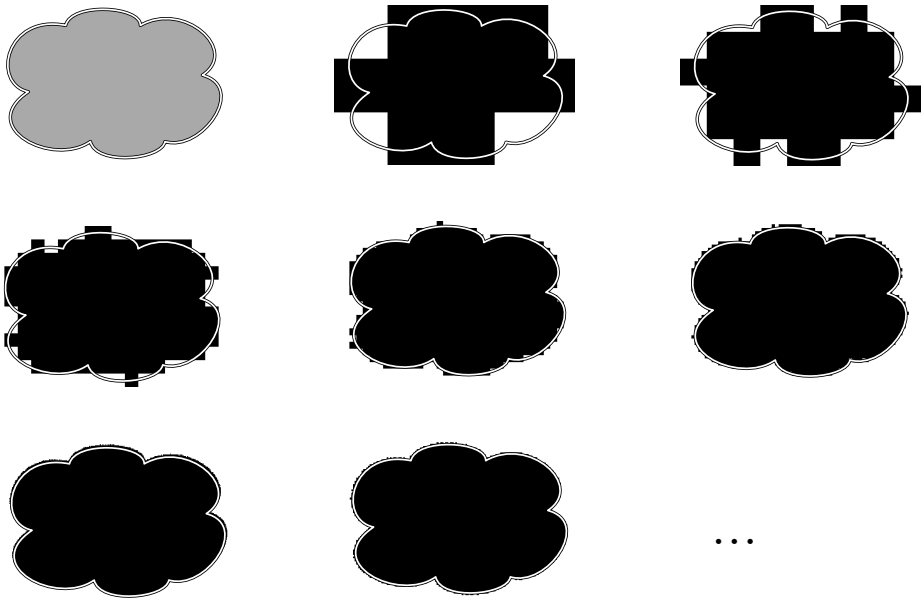
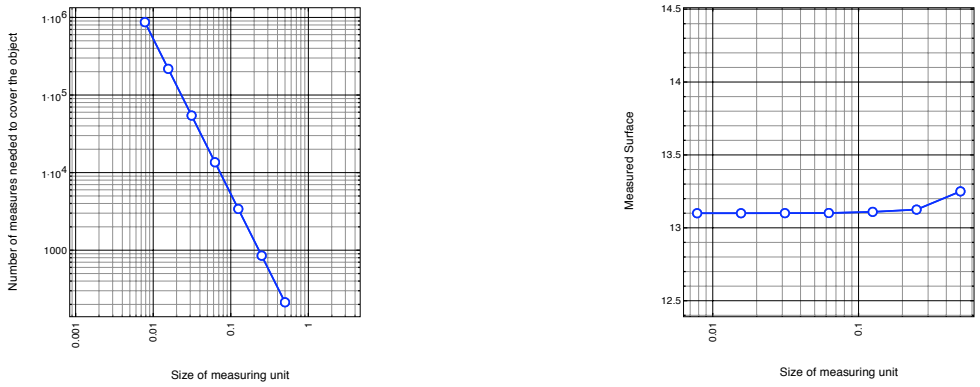


Figure 8.6: Measuring the size of a two-dimensional object, the surface shown in the top left corner, using measuring units of different sizes. Reducing the size of the measuring unit by a (linear) factor of 2 requires approximately four times as many units to cover the surface.



(a) (b)

Figure 8.7: (a) Number of measuring units needed to cover the surface as function of the size of the measuring unit. (b) Measured area of the surface.

D-dimensional object

Equations 8.1 and 8.2 can be summarized, in the general case of a D-dimensional object, as:

$$N \approx c \cdot \left(\frac{1}{a}\right)^D \tag{8.3}$$

Applying the $\log$ to both sides and taking the limit where a get infinitesimally small, we can rewrite this equation as:

$$D = \lim_{a \to 0} \frac{\log(N)}{\log(\frac{1}{a})} \tag{8.4}$$

Another way of looking at this equation is that the slope of N as function of a in a log-log plot gives us $-D$. This relation is illustrated in Figure 8.8, which combines together the two plots of Figures 8.5(a) and 8.7(a).

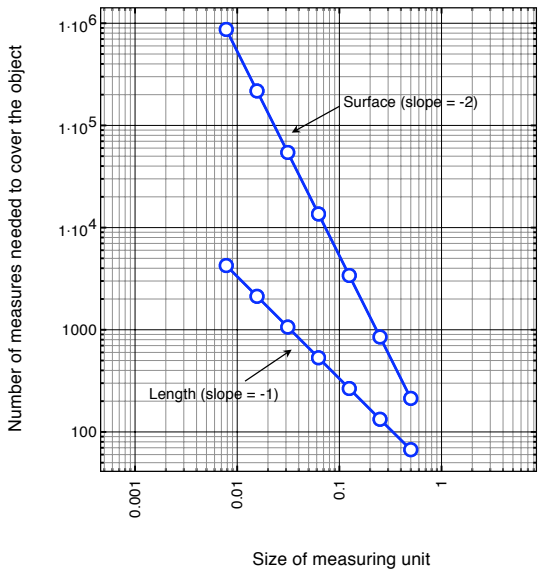
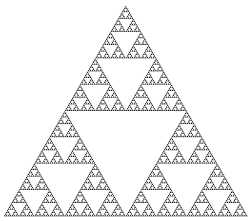


Figure 8.8: Number of measuring units needed to cover objects as function of the size of the measuring unit. The slope (with the reverse sign) gives us the dimension of the object being measured.

Fractal

Let us consider a fractal figure that we've already encountered in the previous chapter – the Sierpinsky gasket:



Let us now measure its surface, using again squares as measuring units. Figure 8.9 shows how the number of measuring units required to cover the fractal pattern increases as the size of the measuring unit decreases.

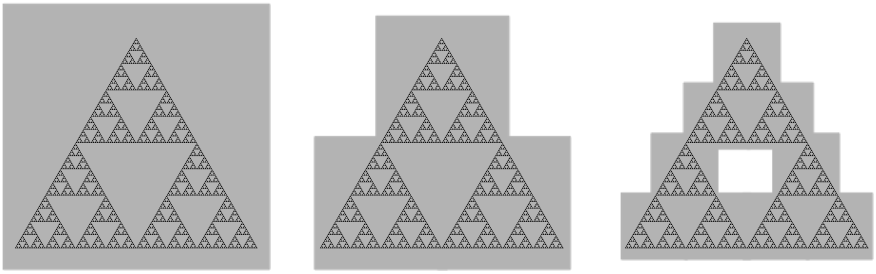


Figure 8.9: Number of measuring units needed to cover objects as function of the size of the measuring unit. The slope (with the reverse sign) gives us the dimension of the object being measured.

Let us consider a serie of measuring units whose size are each time divided by two: $a_n = \left(\frac{1}{2}\right)^n$. From Figure 8.9, we see that the number of measuring units needed to cover the pattern will each time increase by three: $N = 3^n$.

The dimension of this pattern is thus:

$$\begin{aligned} D &= \lim_{a\to 0} \frac{\log(N)}{\log(\frac{1}{a})} = \lim_{n\to\infty} \frac{\log(3^n)}{\log(2^n)} \\ &= \lim_{n\to\infty} \frac{n\log(3)}{n\log(2)} = \lim_{n\to\infty} \frac{\log(3)}{\log(2)} \\ &= \frac{\log(3)}{\log(2)} \approx 1.585 \end{aligned}$$

The dimension is not an integer anymore, but is *fractional*. In a sense, the Sierpinsky gasket is neither one-dimensional (as a line) nor two-dimensional (as a surface), but somewhere in between.

Note also that the actual measures (such as length or surface) of patterns with fractional dimenions do not converge to fixed (positive) value anymore. For instance, the Sierpinsky gasket has in fact no area, since:

$$\begin{aligned} A &= N \cdot a^2 \\ &= \lim_{n\to\infty} N_n \cdot a_n^2 \\ &= \lim_{n\to\infty} 3^n \cdot \left(\frac{1}{2^n}\right)^2 \\ &= \lim_{n\to\infty} \left(\frac{3}{4}\right)^n = 0 \end{aligned}$$

Furthermore, the measured length of the border of the Sierpinsky gasket diverges to infinity – similar to the lengths of the coastlines measured by Richardson!

$$\begin{aligned} L &= N \cdot a \\ &= \lim_{n\to\infty} N_n \cdot a_n \\ &= \lim_{n\to\infty} \left(\frac{3}{2}\right)^n \longrightarrow \infty \end{aligned}$$

8.4 Example of Fractals

8.4.1 The Cantor Set

The so-called *Cantor set* is one of the simplest fractals. It can be constructed as follows (see Figure 8.10):

1. Start with the interval $[0, 1]$ and color it black.
2. Take out the middle third of it, yet leaving the endpoints. You end up with two intervals: $[0, \frac{1}{3}] \cup [\frac{2}{3}, 1]$.
3. From each remaining interval take out the (open) middle third and repeat this procedure for ever.

This pattern has a number of remarkable and deep properties.

Figure 8.10: The first 7 iterations of the Cantor set.

Properties

What is the length of the Cantor set? After n steps, we have 2^n segments, each of which has a length of $\frac{1}{3^n}$. The length of the Cantor set is thus zero:

$$L = \lim_{n\to\infty} \frac{2^n}{3^n} = 0$$

Even though the Cantor set has a length of zero, it still contains some points. The "endpoints" of each interval are never removed, so the Cantor set contains an infinite number of points.

It may appear that only the endpoints are left, but that is not the case either. The number $\frac{1}{4}$, for example, is in the bottom third, so it is not removed at the first step. It is in the top third of the bottom third, and is in the bottom third of that, and in the top third of that, and in the bottom third of that, and so on ad infinitum – alternating between top third and bottom third. Since it is never in one of the middle thirds, it is never removed, and yet it is also not one of the endpoints of any middle third.

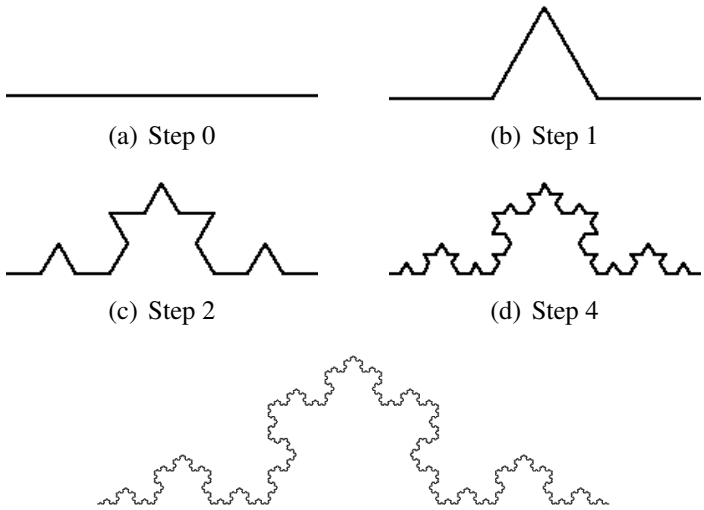
In fact, it can be shown that the Cantor set is *uncountable* – there are as many points in the Cantor set as there are in the original interval [0, 1]!

8.4.2 The Koch Curve

The Koch curve is one of the earliest fractal curves to have been described. Each stage is constructed in 3 steps by starting with a line segment and then recursively altering each line segment as follows (see Figure 8.11):

1. Divide the line segment into three segments of equal length.
2. Draw an equilateral triangle that has the middle segment from step one as its base.
3. Remove the line segment that is the base of the triangle from step 2.

The Koch snowflake, shown in Figure 8.12, is the same as the Koch curve, except that it starts with an equilateral triangle.



(a) Step 0 (b) Step 1

(c) Step 2 (d) Step 4

(e) The Koch curve

Figure 8.11: Recursive construction of the Koch curve.

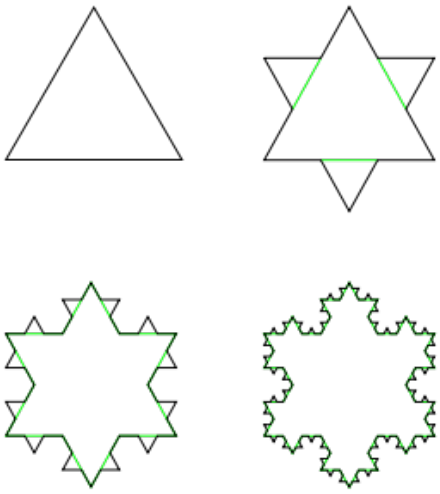


Figure 8.12: The Koch snowflake applies the Koch curve to an equilateral triangle.

8.4.3 The Peano and Hilbert Curves

We just saw that the Cantor set, despite having a length of *zero* (such as a point), could nevertheless contain as many points as a *line*.

The Peano curve is a similar object, but one dimension higher. On the one hand, it is a *curve* – a continuous mapping from the interval $[0, 1]$. On the other hand, it passes through *all* the points contained in the two-dimensional square $[0, 1] \times [0, 1]$! Because of these properties, it is called a *space-filling curve*.

The Peano curve can be constructed using the recursive iteration shown in Figure 8.13(a). The first steps of the construction of the Peano curve are shown in Figure 8.13(b).

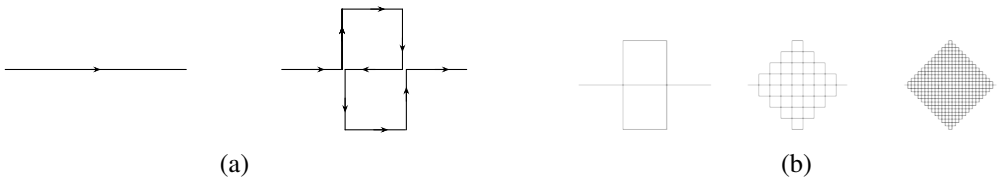


Figure 8.13: (a) Iteration of the Peano curve: each line segment is replaced by the pattern shown on the right. (b) First three steps of the construction leading to the Peano curve.

Another example of space-filling curve is the Hilbert curve:

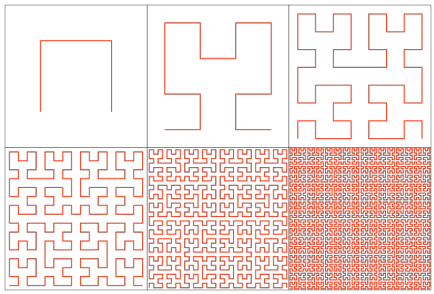


Figure 8.14: The first six iterations of the Hilbert curve.

8.5 L-Systems

In 1968, the biologist Aristid Lindenmayer invented a mathematical formalism to model the growth of simple multicellular organisms, such as bacteria, algae or plant cells. This formalism, known as *Lindenmayer system* or *L-system*, is essentially a formal grammar – such as the ones we encountered in Chapter 2.

With the advent of informatics, L-systems have not only become popular to model the growth processes of plant development, but also to graphically generate the morphology of complex organisms (see Figure 8.15).

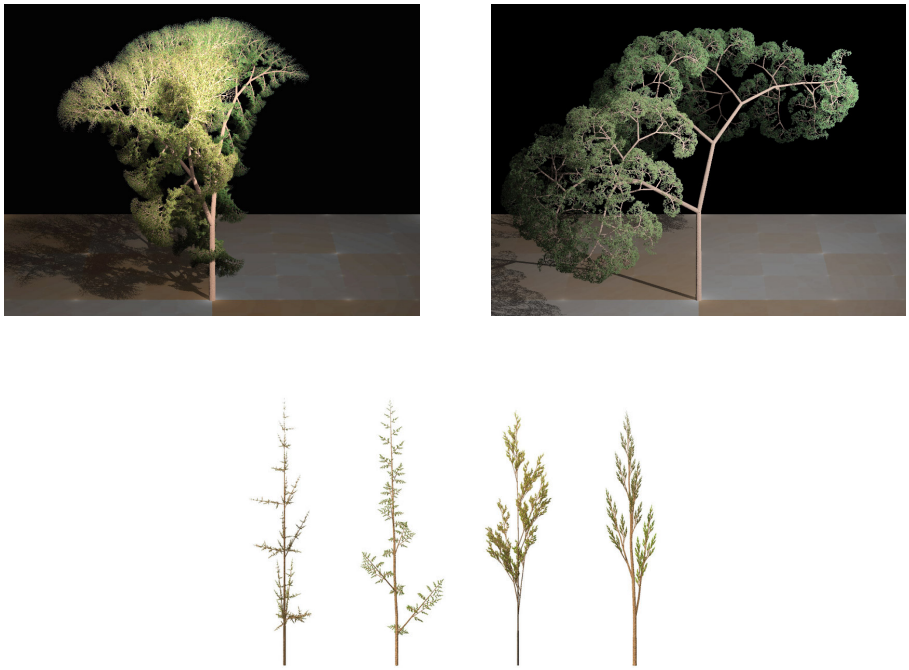


Figure 8.15: Fractal trees and plants created using a Lindenmayer system.

L-systems consist essentially of rewrite rules that are applied iteratively to some initial string of symbols. The *recursive* nature of the L-system rules leads to *self-similarity* and thereby to fractal-like forms, which are easy to describe with an L-system.

Note that the rules of the L-system grammar are applied iteratively starting from the initial state. During each iteration, as many rules as possible are applied *simultaneously*. This is the distinguishing feature between an L-system and the

formal language generated by a grammar.

Example 8.1. The following grammar was Lindenmayer's original L-system for modelling the growth of algae:

- **Variables:** A, B
- **Constants:** (none)
- **Axiom:** A
- **Production rules:**

$$\begin{aligned} A &\rightarrow AB \\ B &\rightarrow A \end{aligned}$$

This produces:

Step 0: A

Step 1: AB

Step 2: ABA

Step 3: $ABAAB$

Step 4: $ABAABABA$

Step 5: $ABAABABAABAAB$

Step 6: $ABAABABAABAABABAABABA$

Step 7: $ABAABABAABAABABAABABAABAABABAABAAB$

8.5.1 Turtle Graphics

L-systems in and of themselves do not generate any image – they merely produce large sequences of symbols. In order to obtain a picture, these strings have to be interpreted in some way. Very often, L-systems are interpreted by turtle graphics.

The concept of turtle graphics originated with Seymour Papert. Intended originally as a simple computer language for children to draw pictures (LOGO), a modified turtle graphics language is well suited for drawing L-systems. Plotting

Command	Turtle Action
F	draw forward (for a fixed length)
|	draw forward (for a length computed from the execution depth)
+	turn right (by a fixed angle)
−	turn left (by a fixed angle)
[	save the turtle's current position and angle for later use
]	restore the turtle's position and angle saved during the corresponding] command

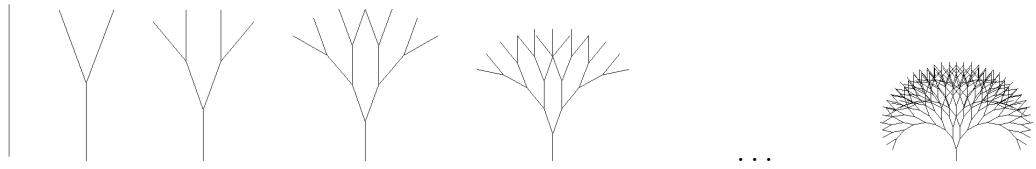
Table 8.1: Turtle graphics commands.

is performed by a virtual turtle. The turtle sits at some position looking in some direction on the computer screen and can move forward, either with or without drawing a line, and turn left or right. A brief summary of commands used for drawing L-systems is given in Table 8.1. Note that without the brackets, the drawing of branching structures would be impossible.

Example 8.2. The following L-system:

- **Axiom:** F
- **Production rule:** F $\rightarrow$ F [−F][+F]
- **Angle:** 20

produces the following stages of the draw process:

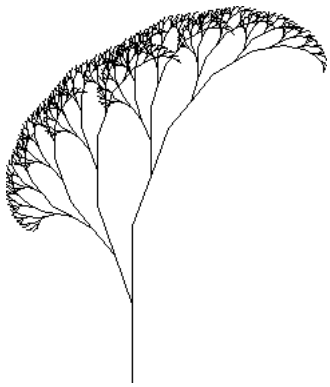


Example 8.3.

Axiom: F
Production rule: F $\rightarrow$ F [−F] F [+F] F
Angle: 20

Example 8.4.

Axiom: F
Production rule: F $\rightarrow$ | [+F] | [−F] + F
Angle: 20



8.5.2 Development Models

Naturally, the interpretation of the L-system can be extended to three dimensions by adding a third dimension to the orientation of the turtle. Furthermore, in order to simulate more complex models of plant development, additional information can be included into the production rules, including delay mechanisms, influence of environmental factors or stochastic elements – so that not all the plants look the same (see Figure 8.16).

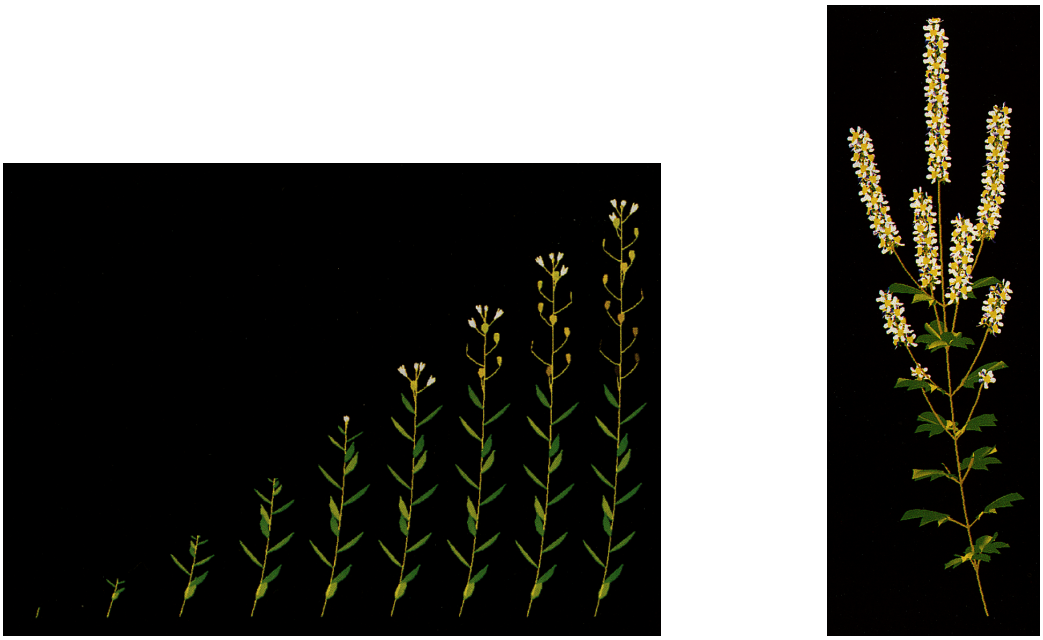


Figure 8.16: Simulated development of plants grown with more complex L-systems.

8.6 The Mandelbrot Set

We have seen so far how fractal patterns can be generated with a developmental process, using for instance rewrite rules that are applied recursively. In some sense, the self-simililarity of fractals seems closely related to the recursive nature of the production rules.

Remember however when we first encountered a fractal pattern. These were cellular automata (Chapter 6) – very simple finite state machine with no recursivity whatsoever!

In this section, wee will see once again how fractals are not necessarily related to recursive property of the system, but rather surprisingly closely related to decidability and chaos.

"Will it diverge?"

Let us consider the sequence of numbers defined by the following equation:

$$x_{i+1} = x_i^2 + c \tag{8.5}$$

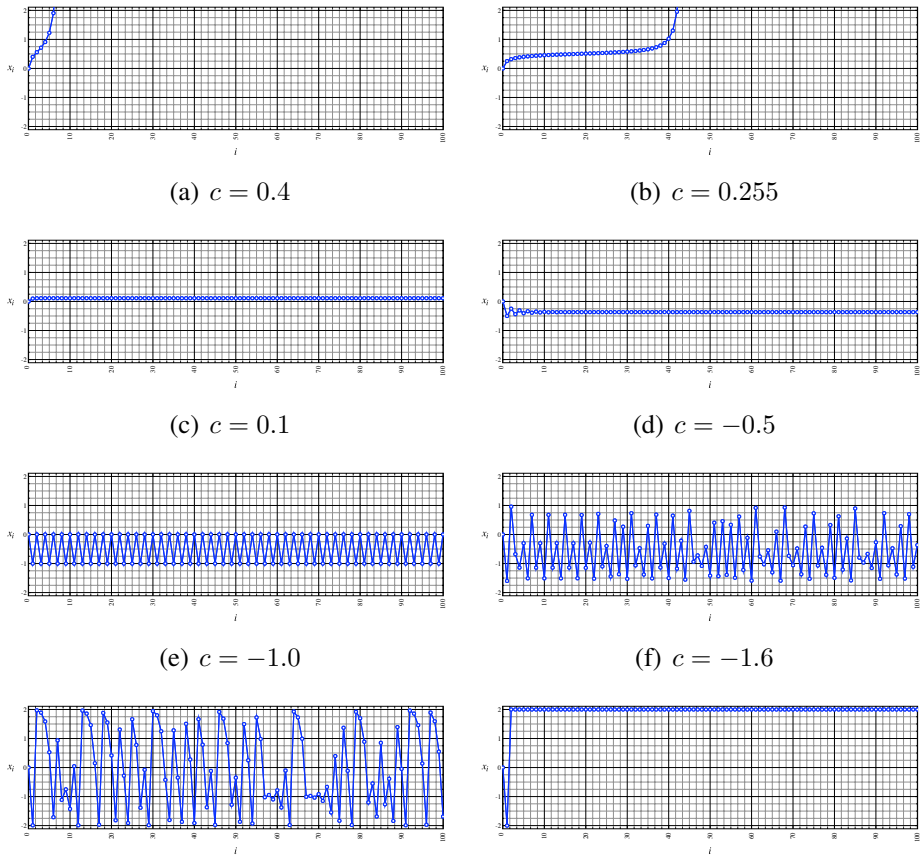
with $x_0 = 0$ and a given constant c.

The question is: for what c will the serie diverge to infinity?

With real numbers

For $|c|$ sufficiently large, the serie will obviously grow without limits. However, the general answer to the question is far from obvious – remember what we've learned from the logistic map of Chapter 7! In fact, the serie will display all kinds of behaviors with point, periodic or even strange attractors, as shown in Figure 8.17.

In summary, even though the answer to the question can be given – the serie seems not to diverge for $c \in [-2, \frac{1}{4}]$ – we see that a simple squaring and an addition are sufficient to bring us very close to chaos.



(a) $c = 0.4$ (b) $c = 0.255$

(c) $c = 0.1$ (d) $c = -0.5$

(e) $c = -1.0$ (f) $c = -1.6$

(g) $c = -1.99$ (h) $c = -2.0$

Figure 8.17: Evolution of the serie defined by Eq. 8.5 for different values of c.

With complex numbers

Whereas an answer seems to exist for $c \in \mathbb{R}$, the results become baffling when considering complex number, i.e. $c = a + ib \in \mathbb{C}$.

The following figure shows the so-called *Mandelbrot set*, which consists of all $c \in \mathbb{C}$ for which the serie of Equation 8.5 does not diverge:

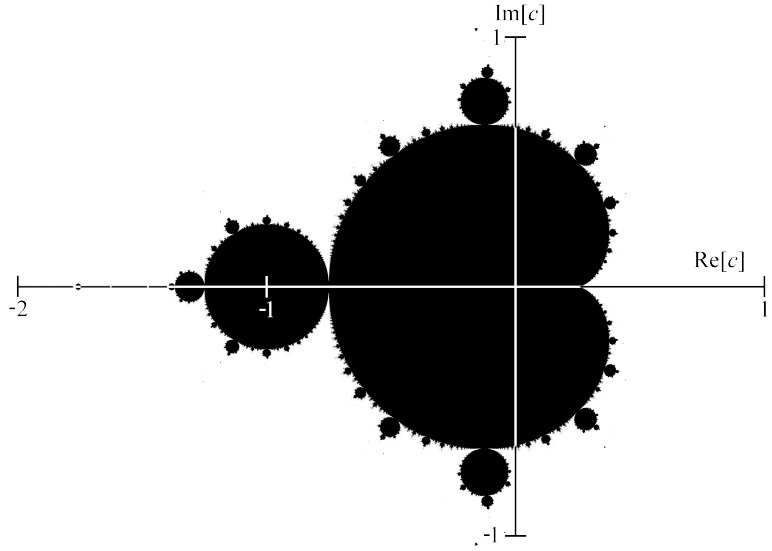


Figure 8.18: The Mandelbrot set.

In this figure, the points belonging to the Mandelbrot set are displayed in black, and the other points are displayed in white.

Usually, some other coloring scheme are used to produce more spectacular graphics (see Figure 8.19). Typically, the color of the points outside the Mandelbrot set are chosen as a function of the number of steps needed for the serie to exceed a given threshold (i.e. the minimum i for which $|x_i| > \theta$, with typically $\theta = 2$).

Also, since it is impossible to know whether the serie will diverge or not, the serie is usually assumed not to diverge if it hasn't exceeded the threshold after a maximum number of steps (i.e. if $|x_{i_{\max}}| \leq \theta$).

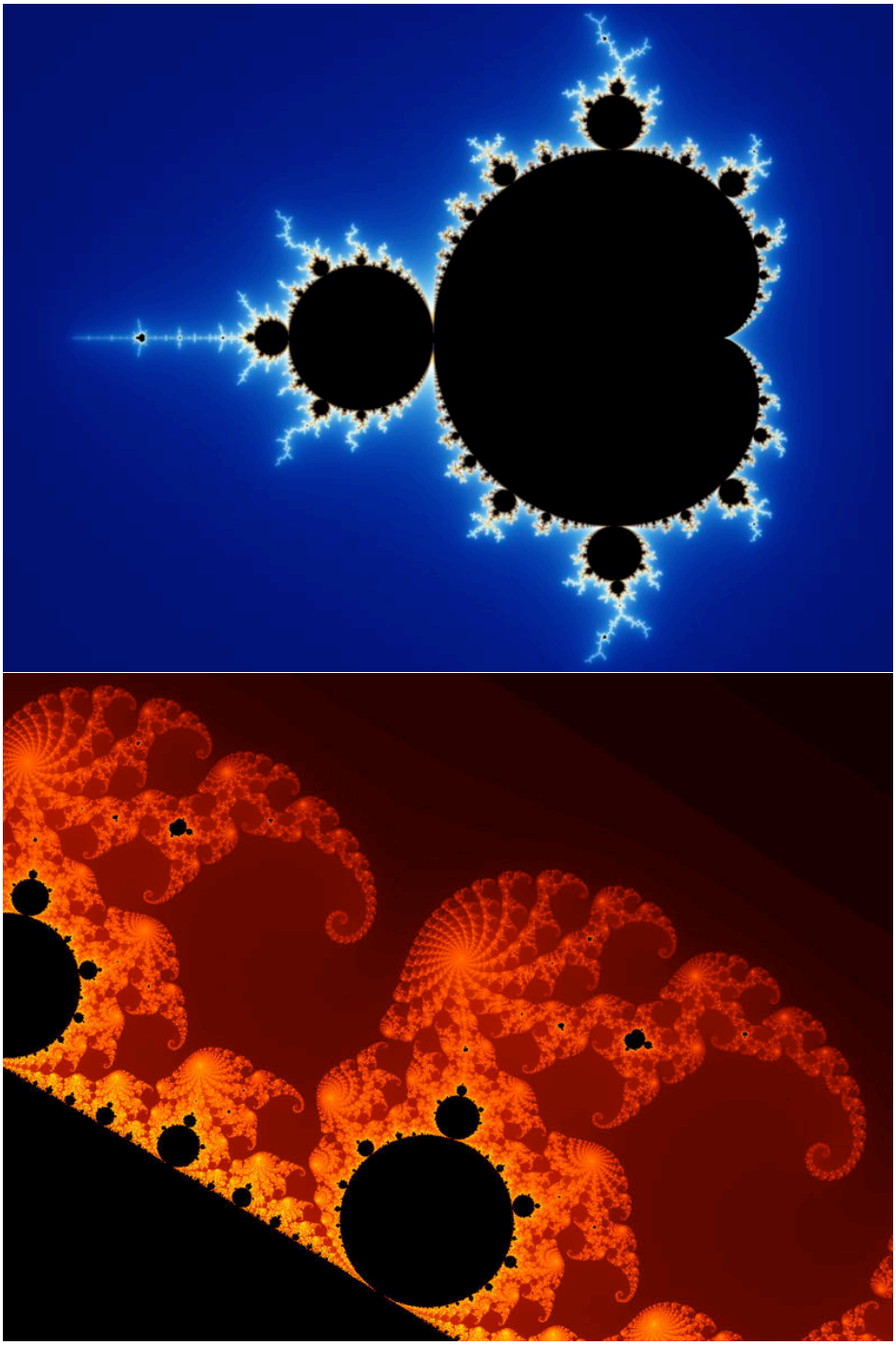


Figure 8.19: Top: A colored version of the Mandelbrot set. Bottom: A zoom on the edge of the Mandelbrot set showing the fractal nature of the structure.

8.7 Chapter Summary

- *Fractals* refer to structures displaying self-similarity on different scales.
- Coastlines do not have a "real" length, but their measured length depend on the length of the measuring unit.
- Fractals have *fractional dimensions*.
- The *Cantor set* has a length of zero, but contains as many points as the interval $[0, 1]$. Similarly, *space-filling curves* (such as the Peano curve or the Hilbert curve) are curves that traverse all points of the two-dimensional interval $[0, 1] \times [0, 1]$.
- *L-systems* are a particular kind of formal grammars used to model growth of multicellular organisms and to display, using *turtle graphics*, patterns with self-similiarity such as plants or fractals.
- The *Mandelbrot set* is another case study that illustrates the close relationship between simple non-linear systems, chaos and fractals.