Recursion : Fractal Curves Explained

Alvin Rizqi Alfisyahrin / 13519126
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13519126@std.stei.itb.ac.id

Abstract*—Fractal curve is one of many things that applies recursive pattern in a mathematical way. There's a lot of curve that represents recursivity, but here author would like to explain 2 curves that is Fibonacci word fractal and Koch snowflake. Authors decided to bring this topic because those curves are one of the most common curve out there that implements recursive functions. Fractal curves are related to other fields such as economics, fluid mechanics, geomorphology, human physiology, and linguistics. Fractal curves also can be found in nature such as broccoli, snowflakes, lightning bolts, and frost crystals. Fractal curves are a self-similar object, which is in mathematics means that an object that is similar to a part of itself. For better understanding, the whole object has similar shape as on or more parts of the object.

***Keywords*— curve, fibonacci, fractal, koch, recursive**

I. Introduction

Before we start discussing fractal curves, we must know what is a fractal in mathematics. Fractal is a self-similar subset of *Euclidean space* whose fractal dimension strictly exceeds its topological dimension. What is fractal dimension? Fractal dimension is a ratio providing a statistical index of complexity comparing how detail in a pattern changes with the scale at which it is measured. It has also been characterized as a measure of the space-filling capacity of a pattern that tells how a fractal scales differently from the space it is embedded in; a fractal dimension does not have to be an integer.

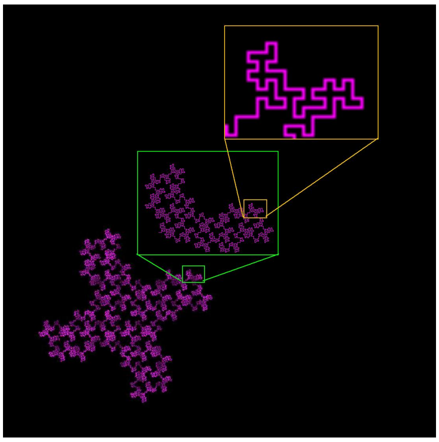


Image 1 : Image representation of fractal dimension
Source : Audrey Karperien

Topological dimension explains how the dimension in geometry works, for example, a cube has topological dimension of three, its boundary which is a square has topological dimension of two, and the square's boundari which is a line has topological dimension of 1 and so on. We do not discuss topological deeper than this because we concern the recursivity in fractal curves.

Back to fractal curves, a fractal has the same shape no matter how much we magnify it. This explains the irregularity in fractals for its behavior to maintain its shape even though it consumes a super-small space for us humans.

Early in the 17th century with notions of recursion, fractals have moved through increasingly rigorous mathematical treatment of the concept to the study of continuous but not differentiable functions in the 19th century by the seminal work of Bernard Bolzano, Bernhard Riemann, and Karl Weierstrass and on to the coining of the word fractal in the 20th century with a subsequent burgeoning of interest in fractals and computer-based modelling in the 20th century. The term "fractal" was first used by mathematician Benoit Mandelbrot in 1975. Mandelbrot based it on the Latin frāctus, meaning "broken" or "fractured", and used it to extend the concept of theoretical fractional dimensions to geometric patterns in nature.

There are common techniques to generate fractals using programs. One technique is called Iterated Function Systems, L-systems, and Escape-time fractals.

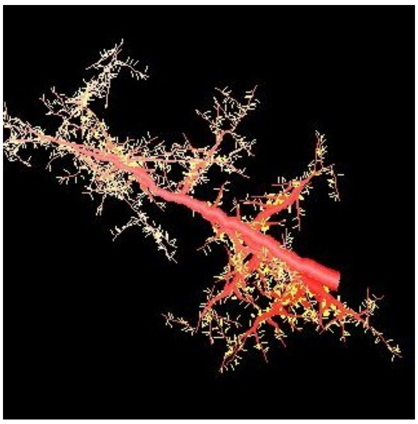


Image 2 : Image representation of L-systems
Source : Arkape

Iterated function systems uses fixed geometry replacement rules, the examples are Koch snowflake, Peano curve, and Menger sponge.

L-systems uses string rewriting that resembles branching patterns for example in cells and plants.

Escape-time fractals uses formula of recursive relations at each point in space, the examples are Mandelbrot set, Julia Set, etc.

We may found fractals on natural phenomenon such as Earthquakes, DNA, Snowflakes, Trees, Clouds, Algae, Broccoli, Rings of Saturn, and so many more. Fractals help to build a strong basic to the shape and helps a lot in perfecting the shape of the object.

II. Basic Theory

Recursion is a method of solving a problem where the solution depends on solutions to smaller instances of the same problem. Such problems can generally be solved by iterations, but this needs to identify and index the smaller instances at programming time. Recursion solves such recursive problems by using that call themselves from within their own code. The approach can be applied to many types of problems, and recursion is one of the central ideas of computer science.

Recursive functions defined by two parts:

i. Basis

A part where the value of functions defined explicitly. It is also a part that stops the iteration of recursive functions. It can be one or more basis to stop the recursive function

ii. Recurrence

A part where the function defines itself (self-similarity), also counts the output using its own smaller parts.

Examples of recursive functions :

We can define factorials in recursive function,

$$n! = \begin{cases} 1 & , n = 0 \\ n \cdot (n-1)! & , n > 0 \end{cases}$$

We can also define Fibonacci sequence using recursive function,

$$f_n = \begin{cases} 0 & , n = 0 \\ 1 & , n = 1 \\ f_{n-1} + f_{n-2} & , n > 1 \end{cases}$$

There's a lot more functions that can be explained using recursion.

Here is the recursive pseudocode of Factorial

```
function factorial is:

input: integer n such that n >= 0

output: [n × (n-1) × (n-2) × … × 1]

    1. if n is 0, return 1

    2. otherwise, return [ n × factorial(n-1) ]

end factorial
```

Recurrence relation is an equation that recursively defines a sequence or multidimensional array of values, once one or more initial terms are given; each further term of the sequence or array is defined as a function of the preceding terms.

Example: Fibonacci numbers,

f(n) = f(n-1) + f(n-2), with initial conditions of f(0) = 0 and f(1) = 1.

Solving homogeneous linear recurrence relations with constant coefficients.

an order-d **homogeneous linear recurrence with constant coefficients** is an equation of the form

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_d a_{n-d},$$

where the d coefficients c_i (for all i) are constants, and c_d is not 0.

A constant-recursive sequence is a sequence satisfying a recurrence of this form. There are d degrees of freedom for solutions to this recurrence, i.e., the initial values can be taken to be any values but then the recurrence determines the sequence uniquely.

The same coefficients yield the characteristic polynomial (also "auxiliary polynomial")

$$p(t) = t^d - c_1 t^{d-1} - c_2 t^{d-2} - \cdots - c_d$$

whose d roots play a crucial role in finding and understanding the sequences satisfying the recurrence. If the roots r_1, r_2, ... are all distinct, then each solution to the recurrence takes the form

$$a_n = k_1 r_1^n + k_2 r_2^n + \cdots + k_d r_d^n,$$

on order 1, the recurrence

$$a(n) = r\, a(n-1)$$

has the solution $a_n = r^n$ with $a_0 = 1$ and the most general solution is $a_n = kr^n$ with $a_0 = k$. The characteristic polynomial equated to zero (the characteristic equation) is simply $t - r = 0$.

Solutions to such recurrence relations of higher order are found by systematic means, often using the fact that $a_n = r^n$ is a solution for the recurrence exactly when $t = r$ is a root of the characteristic polynomial. This can be approached directly or using generating functions (formal power series) or matrices.

Consider, for example, a recurrence relation of the form

$$a_n = Aa_{n-1} + Ba_{n-2}.$$

When does it have a solution of the same general form as $a_n = r^n$? Substituting this guess in the recurrence relation, we find that

$$r^n = Ar^{n-1} + Br^{n-2}$$

must be true for **all** $n > 1$.

Dividing through by r^{n-2}, we get that all these equations reduce to the same thing:

$$r^2 = Ar + B,$$
$$r^2 - Ar - B = 0,$$

which is the characteristic equation of the recurrence relation. Solve for r to obtain the two roots λ_1, λ_2: these roots are known as the characteristic roots or **eigenvalues** of the characteristic equation. Different solutions are obtained depending on the nature of the roots: If these roots are distinct, we have the general solution

$$a_n = C\lambda_1^n + D\lambda_2^n$$

while if they are identical (when $A^2 + 4B = 0$), we have

$$a_n = C\lambda^n + Dn\lambda^n$$

This is the most general solution; the two constants C and D can be chosen based on two given initial conditions a_0 and a_1 to produce a specific solution.

Recursive data structures (structural recursion)

An important application of recursion in computer science is in defining dynamic data structures such as lists and trees. Recursive data structures can dynamically grow to a theoretically infinite size in response to runtime requirements; in contrast, the size of a static array must be set at compile time.

Linked Lists
Here is the data structure of linked list that can be used for recursive functions

```
struct node
{
  int data;          // some integer data
  struct node *next;  // pointer to another struct node
};
```

Here is the example of printing list by calling its own function;

```
void list_print(struct node *list)
{
   if (list != NULL)             // base case
   {
      printf ("%d ", list->data);  // print integer data followed by a space
      list_print (list->next);     // recursive call on the next node
   }}
```

III. Fibonacci Word Fractal

Before we begin, let me explain what a fibonacci word is. **Fibonacci word** is a specific sequence of binary digits (or symbols from any two-letter alphabet). The Fibonacci word is formed by repeated concatenation in the same way that the Fibonacci numbers are formed by repeated addition.

Fibonacci word could also be categorized as a formal language that has zeros and ones, but never two consecutive ones. There are many kinds of that formal language though, but Fibonacci word is one of the many kinds.

Let S0 be "0" and S1 be "01". Now Sn = Sn-1 . Sn-2 (the concatenation of the previous sequence and the one before that).

The infinite Fibonacci word is the limit , that is, the (unique) infinite sequence that contains each , for finite , as a prefix.

Enumerating items from the above definition produces:

S0	0
S1	01
S2	010
S3	01001
S4	01001010
S5	0100101001001
	...

The first few elements of the infinite Fibonacci word are:

0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, ...

Now that we know what Fibonacci word is, we shall continue to explain Fibonacci word fractal. **Fibonacci word fractal** is a fractal curve that is defined by fibonacci word.

This curve is built iteratively by applying, to the Fibonacci word 0100101001001...etc., the Odd–Even Drawing rule:

For each digit at position k :

1. Draw a segment forward
2. If the digit is 0:
    - Turn left if k is even
    - Turn right if k is odd

A curve made up of segments is associated with a Fibonacci ter m of length (the nth Fibonacci number). Three different aspects are shown in the curve, if n is in the 3k, 3k + 1 or 3k + 2 form.

Here are the properties of Fibonacci word fractal:

- The curve , contains segments, right angles and flat angles.
- The curves never intersects itself. On its limit, all the lines will be so close that we can say its asymptotic

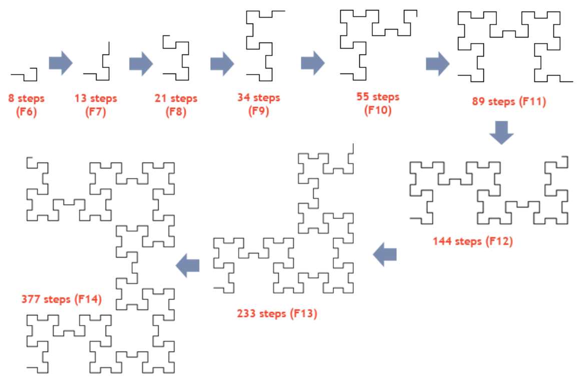


Image 3 : Steps of generating fibonacci word fractal

Source : Prokofiev

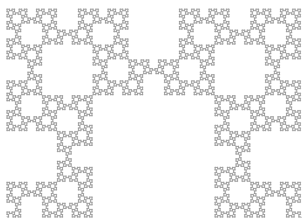


Image 4 : Fibonacci word fractal at F23

Source : Alexis Monnerot-Dumaine

Extra : The Fibonacci tile

The juxtaposition of four curves allows the construction of a closed curve enclosing a surface whose area is not null. This curve is called a "Fibonacci Tile".

- Fibonacci tiles almost tiles the plane. The 4 tiles' juxta position leaves a free square at the middle whose area tends to zero as k tends to infinity.At the limit, the infinite Fibonacci tile tiles the plane.
- If the tile is enclosed a square of side 1, then its area tends to 0.5857.

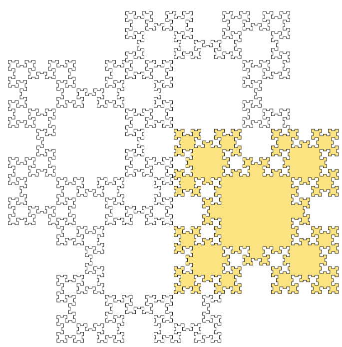


Image 5 : Fibonacci tile
Source : Prokofiev

Juxtaposition is an act or an instance of bringing two elements close together or side by side. This is also done to compare/ contrast the two, to demonstrate similarities or differences, etc.

IV. Koch Snowflake

Koch Snowflake is a fractal curve and one of the earliest fractals to have been described.

The Koch snowflake can be built iteratively, in a series of ste ps. The first stage is the equilateral triangle, and each successiv e stage is created by adding outward bends to each side of the p revious stage, resulting in smaller equilateral triangles. The are as surrounded by successive stages of snowflake construction c onverge to 8/5 times the area of the initial triangle, while the pe rimeters of the successive stages increase without a boundary.. As a consequence, the snowflake encloses a finite field, but has an infinite perimeter.

The Koch snowflake can be built starting with the equilateral triangle, and then recursively altering each line section as follows.:

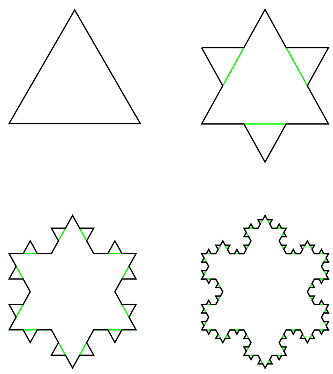
1. divide the line segment into three segments of equal length.
2. draw an equilateral triangle that has the middle segment from step 1 as its base and points outward.
3. remove the line segment that is the base of the triangle from step 2.

The first iteration of this process produces the outline of a hexagram.

The Koch snowflake is the limit approached since the above steps are followed forever. The Koch curve originally defined by Helge von Koch is designed using only one of the three sides of the original triangle. In other words, there are three Koch curves that create a Koch snowflake.

Similarly, a Koch curve-based representation of a nominally flat surface can be generated by repeatedly segmenting each line in a sawtooth pattern of segments with a given angle..

Image 6 : First steps of generating Koch snowflake



Source : https://commons.wikimedia.org/wiki/ File:KochFlake.svg

Number of sides (n)

For each iteration, one side of the figure in the previous stage becomes four sides in the next stage. Since we start with three sides, there is a formula for the number of sides in the Koch Snowflake.

$$N_n = N_{n-1} \cdot 4 = 3 \cdot 4^n \, .$$

in the nth iteration.

For iterations 0, 1, 2 and 3, the number of sides are 3, 12, 48 and 192, respectively.

Length of a side (length)

IN every iteration, the length of the side is 1/3 of the length of the side from the preceding point. If we begin with an equilateral triangle with a side length of x, then the length of the side is n.

$$S_n = \frac{S_{n-1}}{3} = \frac{s}{3^n} \, ,$$

For iterations 0 to 3, length = s, s/3, s/9 and s/27.

Perimeter (p)

Since all the sides in every iteration of the Koch Snowflake is the same the perimeter is simply the number of sides multiplied by the length of n side

$$P_n = N_n \cdot S_n = 3 \cdot s \cdot \left(\frac{4}{3}\right)^n \, .$$

for the n-th iteration.

Again, for the first 4 iterations (0 to 3) the perimeter is 3s, 4s, 16s/3, and 64s/9.

As we can see, the perimeter increases by 4/3 times each iteration so as n goes to infinity the perimeter continues to increase with no bound.

Often when n goes to infinity, the snowflake consists of sharp corners with no straight lines linking them. Thus, although the perimeter of the snowflake, which is an infinite sequence, is continuous since there are no breaks in the perimeter, it is not distinguishable because there are no smooth lines.

V. Menger Sponge

The Menger sponge (also known as the Menger cube, Menger universal curve, Sierpinski cube, or Sierpinski sponge) is a fractal curve in mathematics. It is a three-dimensional generalization of the one-dimensional set of Cantor and the two-dimensional set of Sierpinski Carpets. In 1926, Karl Menger first defined the definition of the topological dimension in his research.

The construction of a Menger sponge can be described as follows:

1. Begin with a cube.
2. Divide every face of the cube into nine squares, like Rubik's Cube. This sub-divides the cube into 27 smaller cubes.
3. Remove the smaller cube in the middle of each face, and remove the smaller cube in the very center of the larger cube, leaving 20 smaller cubes. This is a level-1 Menger sponge (resembling a void cube).
4. Repeat steps two and three for each of the remaining smaller cubes, and continue to iterate *ad infinitum*.

The second iteration is a level-2 sponge, the third iteration is a level-3 sponge, and so on. The Menger sponge itself is the li mit of this process to an infinite number of iterations.

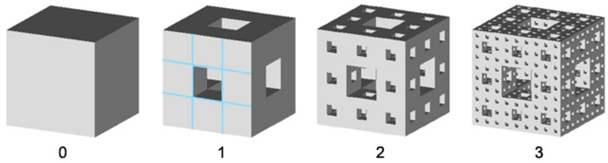


Image 7 : Image representation of menger sponge until 3rd iteration
Source : http://fractalfoundation.org/OFC/OFC-10-3.html

The nth phase of the Menger sponge, Mn, consists of 20 n s maller cubes, each with a length equal of 1/3 n. Thus the total volume of Mn is (20/27)n. The total surface area of Mn is given by the expression 2(20/9)n + 4(8/9)n.[6]. Therefore the volume of the construction reaches zero while its surface area increases without bounds. However any preferred surface in the construction will be extensively lacerated as the construction continues, so that the limit is neither a solid nor a surface; it has a topological dimension of 1 and is therefore ide ntified as a curve.

Cantor Set

The set of Cantor is a set of dots lying on a single line section with a number of extraordinary and profound properties. It was discovered in 1874 by Henry John Stephen Smith and introduced in 1883 by the German mathematician George Cantor.

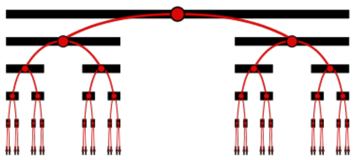


Image 8 : Image representation of Cantor set
Source : Sam Derbyshire

The Cantor ternary set is generated by removing the open middle third from a set of line segments. One begins by eliminating the middle third (1/3, 2/3) from the range[0, 1], leaving two line segments: [0, 1/3] [2/3, 1]. Next, the middle third of both of these remaining segments is removed, leaving four line segments: [0, 1/9] [2/9, 1/3] [2/3, 7/9] [8/9, 1]. This method is repeated ad infinitum, where the nth set is set.

$$C_n = \frac{C_{n-1}}{3} \cup \left(\frac{2}{3} + \frac{C_{n-1}}{3}\right) \text{ for } n \geq 1, \text{ and } C_0 = [0,1].$$

Sierpiński carpet

The Sierpiński carpet is a plane fractal, first described by Waclaw Sierpinski in 1916. The carpet is a two-dimensional generalization of the Cantor; another is the Cantor tile.

The technique of subdividing the form into a smaller one, eliminating one or two copies, and proceeding recursively, can be generalized to other forms. For eg, subdividing the equilateral triangle into four equilateral triangles, eliminating the middle triangle, and recursing leads to the Sierpinski triangle.

The construction of the Sierpiński carpet starts with a rectangle. The square is cut into 9 congruent sub-squares in a 3-by-3 grid and the central sub-square is omitted. The same method is then extended recursively to the remaining 8 sub-squares, ad infinitum. It can be realized as a group of points in a unit square whose coordinates, written in base three, do not have a digit '1' in the same place..

The process of recursively removing squares is an example of a finite subdivision rule.

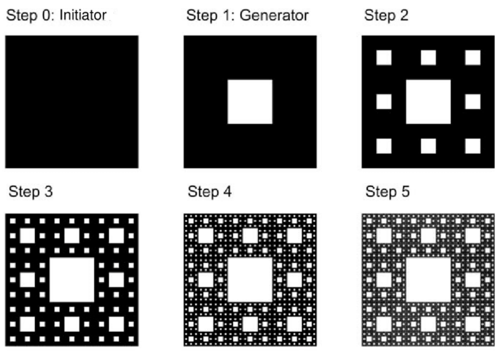


Image 7 : Image representation Sierpinski carpet to step 5
Source : https://www.researchgate.net/figure/Sierpinski-carpet_fig15_254559207

Let Nn be the number of black boxes, Ln the length of a side of a white box, and An the fractional area of black boxes after the n-th iteration. Then

$$Nn = 8^n$$
$$Ln = 3^{-n}$$
$$An = Ln^2 . Nn = (8/9)^n$$

VI. Conclusion

From this paper author could explain the implementation of recursion in a mathematical way. Author explained the basics of Fibonacci word fractal, Koch snowflake, and Menger sponge. These implementation are the example of fractal curves, where fractal curves are familiar with its recursivity. From this paper we could also tell that there is a lot of things constructed by recursion, for example a snowflake. The construction of snowflake could be explained by Koch snowflake.

There are many more patterns in our world that could be explained using recursion, while the author could only explain three patterns because the lack of knowledge and time. This paper is made for scientific purposes, to enrich the knowledge of reader and author regarding the implementation of recursion.

VII. Acknowledgment

Author would like to thank Allah SWT. for His guidance on writing this paper, because without Him I wouldn't be able to make it. Author would like to also thank my lecturer Dra. Harlili, M.Sc for teaching me basics of recursion so that the author could understand the theory better. Also, big thanks to Mr. Rinaldi Munir for encouraging the students to finish the paper and supports the students by creating a website that store all the theories there.

Author would also like to thank his family and friends that gives me encouraging words on finishing this paper. Without you, author would not be able to finish this.

References

1. Ramírez, José L.; Rubiano, Gustavo N. (2014). "Properties and Generalizations of the Fibonacci Word Fractal", *The Mathematical Journal*, Vol. 16.
2. Dickau, R. M. "Menger (Sierpinski) Sponge." https://mathforum.org/advanced/robertd/sponge.html.
3. HLauwerier, Hans (1991). Fractals: Endlessly Repeated Geometrical Figures. Translated by Gill-Hoffstädt, Sophia. Princeton University Press. p. 36. ISBN 0-691-02445-6.
4. Allouche, Jean-Paul; Shallit, Jeffrey (2003), Automatic Sequences: Theory, Applications, Generalizations, Cambridge University Press, ISBN 978-0-521-82332-6.
5. Falconer, Kenneth (2003). Fractal Geometry: Mathematical Foundations and Applications. John Wiley & Sons. xxv. ISBN 978-0-470-84862-3.
6. JEdgar, Gerald (2008). Measure, topology, and fractal geometry. New York: Springer-Verlag. p. 1. ISBN 978-0-387-74748-4.
7. Mandelbrot, B. B.: The Fractal Geometry of Nature. W. H. Freeman and Company, New York (1982); p. 15.

Pernyataan

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 3 Desember 2020

Alvin Rizqi Alfisyahrin | 13519126