# Fractal Snowflakes

# Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk

October 10th, 2018

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk

Fractal Snowflakes

# Contents

## Counting

One reason why this is helpful

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk Fractal Snowflakes

- We often use numbers to count things.
- ▶ When we do it step by step, we use a scale of symbols 1, 2, ... to order the items we count one after the other.
- This process the application of a scale of numerals by a one-to-one correspondence.

- ▶ We handle cases where our count ends. Thus, we only use an initial segment of our scale, e.g. 1, 2, ..., n − 1, n.
- An application of an initial segment endows the collection of objects we have ordered with a numerical determination.
- This process makes use <u>both</u> of an ordered disposition of things and of a terminating point on the scale.

イロト 不得 とくほと くほとう ほ

- If we decide to generalise numbers, it seems reasonable to rely on more terminating points and keep applying initial segments.
- ▶ The notation 1, 2, 3, ... identifies a limitation.
- We can apply initial segments only as long as they have a finite terminating point. We need more terminating points.

◆□▶ ◆□▶ ◆臣▶ ◆臣▶ 臣 のへで

• We extend our notation as follows:

 $1,2,3,\ldots, \textcircled{1}-1, \textcircled{1}, \textcircled{1}+1,\ldots$ 

- ► Any term that contains ① (grossone) comes after all finite terms. In particular, the term ① marks off N.
- $\blacktriangleright$  We now think of  $\mathbb N$  as the canonical completed sequence

 $1,2,3,\ldots,\textcircled{1}-2,\textcircled{1}-1,\textcircled{1}$ 

Fractal Snowflakes

# Contents

### Counting

One reason why this is helpful

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk Fractal Snowflakes ▲□▶ ▲□▶ ▲目▶ ▲目▶ 三日 - のへで

- We can classify sequential processes that involve infinitely many stages.
- If a process has features that vary at every stage, we can keep track of them after infinitely many, not only finitely many stages.
- We can study fractals that evolve from different starting points to different points at infinity, compute their area and perimeter.

・ロト ・回ト ・ヨト ・ヨト

3

# The Koch Snowflake (Helge von Koch 1904)



# The Koch Snowflake (Helge von Koch 1904)



Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk Fractal Snowflakes ▲ロ ▶ ▲ 聞 ▶ ▲ 臣 ▶ ▲ 臣 ▶ ─ 臣 ─ の Q @

・ロン ・回 と ・ ヨ と ・ ヨ と …

3

# The Koch Snowflake (Helge von Koch 1904)



・ロン ・回 と ・ ヨ と ・ ヨ と …

3

# The Koch Snowflake (Helge von Koch 1904)



・ロン ・回 と ・ ヨ と ・ ヨ と …

3

# The Koch Snowflake (Helge von Koch 1904)



- Each figure is a stage in a construction with an equilateral triangle as an initiator. We may choose a different initiator.
- If two initiators are different, we may evolve them through the same number of stages. What happens?
- If this number is finite, we can tell that the resulting stages will be different. What happens after infinitely many stages?

- Instead of talking about differences, let us focus on one feature: the perimeter of a stage in fractal evolution.
- If we look at the result of finitely many stages from some initiator, we can compute its perimeter.
- Let us call P<sub>n</sub> the perimeter of the snowflake after n stages. If n only takes finite values, we have:

 $\lim_{n\to\infty}P_n=\infty$ 

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk

Fractal Snowflakes

イロン 不同 とくほう イロン

-

# Counting to (some) infinity

- ▶ If we adopt ①, we are in a position to look at distant stages.
- Because we are operating with a computational generalisation of finite numbers, we can in particular compute P<sub>0</sub>.
- ► If we know the perimeters of two initiators, we can find out the perimeters of their ①-results and compare them.

- If the initiator is an equilateral triangle with sides of length *l*, its perimeter is 3*l*.
- ► At the first iteration, each side is replaced by four sides. The length of each is one third of *I*.
- When we reach the second stage, there are  $3 \cdot 4 = 12$  sides.
- At the next stage, there are  $(3 \cdot 4) \cdot 4 = 48$  sides.

# Some finite evolutions

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk Fractal Snowflakes

# Some finite evolutions

► 
$$P_0 = 3I$$
.

# Some finite evolutions

► 
$$P_0 = 3I.$$
  
►  $P_1 = 4 \cdot 3 \cdot \frac{l}{3} = 4I.$ 

# Some finite evolutions

• 
$$P_0 = 3l$$
.  
•  $P_1 = 4 \cdot 3 \cdot \frac{l}{3} = 4l$ .  
•  $P_2 = 4 \cdot 4 \cdot 3 \cdot \frac{l}{3^2} \sim 5.3$ .

3

# Some finite evolutions

▶ In these examples the perimeter increases. In general:

$$P_n=\frac{4^n}{3^{n-1}}I$$

► Note that this formula describes the evolution of the initiator at a terminating point *n*. We may set *n* = ①.

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk

Fractal Snowflakes

# Some infinite evolutions

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk Fractal Snowflakes

# Some infinite evolutions

$$\blacktriangleright P_{\odot} = \frac{4^{\odot}}{3^{\odot-1}} I.$$

# Some infinite evolutions

• 
$$P_{\odot} = \frac{4^{\odot}}{3^{\odot-1}} I.$$
  
•  $P_{\odot-1} = \frac{4^{\odot-1}}{3^{\odot-2}} I.$ 

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk Fractal Snowflakes

# Some infinite evolutions

$$P_{0} = \frac{4^{0}}{3^{0}-1}I.$$

$$P_{0-1} = \frac{4^{0}-1}{3^{0}-2}I.$$

$$P_{0} - P_{0-1} = \frac{4^{0}-1}{3^{0}-1}I.$$

◆□▶ ◆□▶ ◆三▶ ◆三▶ 三三 うの()

# Some infinite evolutions

► 
$$P_{\odot} = \frac{4^{\odot}}{3^{\odot-1}} I.$$
  
►  $P_{\odot-1} = \frac{4^{\odot-1}}{3^{\odot-2}} I.$   
►  $P_{\odot} - P_{\odot-1} = \frac{4^{\odot-1}}{3^{\odot-1}} I.$ 

- The difference between the perimeter of the ①-th and the preceding stage is infinitely large.
- Initiators may be infinitely significant. What about areas?

Davide Rizza, University of East Anglia, Norwich UK d.rizza@uea.ac.uk

Fractal Snowflakes

∃ nar

Read this paper: Sergeyev, Ya. D. 'The exact (up to infinitesimals) infinite perimeter of the Koch snowflake and its finite area'