

Koch Snowflake and Fractals

Douglas Meade, Ronda Sanders, and Xian Wu

Department of Mathematics

Overview

The word “fractal” is often used in referring any object that is recursively constructed so that it appears similar at all scales of magnification. There are many examples of complex real-life phenomena, such as chaos, ferns, mountains, river networks, biological growth, that can be described and studied using fractals. In this lab and project, we will analyze and generate a classic fractal, the Koch snowflake, and its variations. While it is natural to use a computer to do recursive constructions, we will focus on applications of sequences and series in our study.

A Variation of Koch Snowflake

Basic Construction: While Koch Snowflake starts with an equilateral triangle, this variation starts with a square. A smaller square is then added to each of the four sides. It is done in such a way that a side of each new square is the middle one-third of each side of the original figure. This process is repeated again and again in each successive iteration. The final flake is the limit of this construction.

Finding Area and Perimeter: First notice that we are getting a sequence of flakes with more and more sides by the same basic construction: each side is replaced by five sides of one-third of its length at the next stage. Therefore, as we started at stage 0 with 4 sides, there are $4 * 5 = 20$ sides at stage 1, $4 * 5 * 5 = 100$ sides at stage 2, and, more general, $4 * 5^n$ sides at stage n . Moreover, if we started with a side of length 1 at stage 0, then each side will have a length of $(1/3)^n$ at stage n . That gives a formula

$$TotPerim_n = 4 * 5^n * (1/3)^n = 4 * (5/3)^n$$

for the perimeter of the flake at stage n . We all knew that this sequence diverges and the perimeter of the final flake is hence infinite. To get a formula for the area, notice that the new flake at stage $n \geq 1$ is obtained by adding a square of the side length $(1/3)^n$ to each side of the previous flake, that is, we are adding $4 * 5^{n-1}$ such squares at stage n . Therefore, the total new area added at stage $n \geq 1$ is:

$$AreaAdd_n = 4 * 5^{n-1} * (1/3)^n * (1/3)^n = (4/5) * (5/9)^n.$$

Since the area at stage 0 is 1 and the above area is added at each stage thereafter, we hence obtain the following series for the area of the flake at stage $n \geq 1$:

$$TotArea_n = TotArea_{n-1} + (4/5) * (5/9)^n = 1 + \sum_{k=1}^n (4/5) * (5/9)^k.$$

Apply our knowledge in geometric series, we obtain that the square snowflake has a finite area of

$$1 + (4/9)/(1 - (5/9)) = 1 + 1 = 2.$$

Assignment:

Repeat the same discussion and analysis of at least two different variations of the Koch snowflake. Use your imagination but one of them should be the classic Koch snowflake, that is, instead of using squares, start with an equilateral triangle and recursively add smaller triangles. Your focus should be on how to use sequences and series to get formulas and limits of perimeters and areas. The given Maple worksheets on the course website can be used as a tool to help you to verify and visualize. Have a discussion on what you have learned, such as some interesting properties of the flake. For example, as we have computed, the square flake has a finite area but infinite perimeter. Now, imagining that you have a container with the flake as its base and fill it up with some paint. That means one could paint an infinite area (the interior surface of the container) with a finite amount of paint! Such flakes are also examples of curves that are everywhere continuous but nowhere differentiable. If needed, your TA will help you to modify the given worksheets to generate the first few stages of your constructions.

The Square flake Construction in Maple:

1. **Note:** You should always **restart** from the beginning after any modification.

```
> restart;
> with(plots):
```
2. **Initial data and results (stage 0):** Each side of the initial square is determined by two of its endpoints (vertices) so we need a sequence of five points with the first and the last being the same vertex. Notice that `plot` automatically connects a given sequence of points.

```
> A[1]:=[0,0]; A[2]:=[0,1]; A[3]:=[1,1]; A[4]:=[1,0]; A[5]:=A[1];
> LenSide[0]:=1;
> TotArea[0]:=LenSide[0]*LenSide[0];
> NumSide[0]:=4;
> TotPerim[0]:=NumSide[0]*LenSide[0];
> plot([A[1],A[2],A[3],A[4],A[5]],axes=NONE,scaling=constrained, title=sprintf("My Flake
(stage 0) \n area = % 8.5f, perimeter=%8.5f,sides=%d",TotArea[0],TotPerim[0],NumSide[0]));
```
3. **Results up to stage n=10:** Please pay close attention as your TA works out recursive and general formulas for `TotPerim[n]` (a sequence) and `TotArea[n]` (a series) by hand.

```
> n:=10:
> for k from 1 to n do
>   stage:=k;
>   LenSide[k]:=LenSide[k-1]/3.;
>   TotArea[k]:=TotArea[k-1]+NumSide[k-1]*TotArea[0]/9.^k;
>   NumSide[k]:=5*NumSide[k-1];
>   TotPerim[k]:=NumSide[k]*LenSide[k];
> end do;
```
4. **Construct sequences of snowflake vertices and plots:** Between each pair of adjacent vertices $A[m]$ and $A[m+1]$, we need to add four new vertices $B[m]$, $C[m]$, $E[m]$, and $F[m]$. (Note: Letter D is reserved in Maple.) We then have to re-index them to form a new sequence of vertices $A[m]$ at the next stage. Computations for stages above 5 may require too much time and memory for your computer to handle.

```
> n:=5: # For program testing, use n=2 or 3.
> for k from 1 to n do
>   for m from 1 to 4*5^(k-1) do
>     B[m]:=A[m]+(A[m+1]-A[m])/3;
Note that if let A[m+1]-A[m]=[x,y], then [-y,x]=[A[m][2]-A[m+1][2],A[m+1][1]-A[m][1]]
>     E[m]:=B[m]+[A[m][2]-A[m+1][2],A[m+1][1]-A[m][1]]/3;
>     C[m]:=A[m]+2*(A[m+1]-A[m])/3;
>     F[m]:=C[m]+[A[m][2]-A[m+1][2],A[m+1][1]-A[m][1]]/3;
>   end do;
>   for m from 1 to 4*5^(k-1) do
>     Temp[5*(m-1)+1]:=simplify(A[m]);
>     Temp[5*(m-1)+2]:=simplify(B[m]);
>     Temp[5*(m-1)+3]:=simplify(E[m]);
>     Temp[5*(m-1)+4]:=simplify(F[m]);
>     Temp[5*(m-1)+5]:=simplify(C[m]);
>   end do;
>   for m from 1 to 4*5^k do
>     A[m]:=Temp[m];
>   end do;
>   A[4*5^k+1]:=A[1];
>   SFplots[k] := seq(plot([A[m],A[m+1]]),m=1..4*5^k);
> end do;
> for k from 1 to n do
>   display([SFplots[k]],axes=NONE,scaling=constrained,title=sprintf("My flake(stage %a)
\n area =%8.5f,perimeter=%8.5f,sides =%d",k,TotArea[k],TotPerim[k],NumSide[k]));
> end do;
```