JACM **3**, No. 4, 1-15 (2018)
Journal of Abstract and Computational Mathematics
http://www.ntmsci.com/jacm

A different construction of the classical fractals via the escape time algorithm

Nisa Aslan[1], *Mustafa Saltan*[2], *Bünyamin Demir*[3]

[1,2,3]Department of Mathematics, Eskişehir Technical University, Eskişehir 26470, Turkey

Received: 19 Sep 2018, Accepted: 06 Nov 2018
Published online: 23 Dec 2018

Abstract: Fractals are fascinating shapes that have many examples in the nature and have the self-similarity property. In recent years, many studies have been made to obtain different fractal sets. There are several methods to generate these sets such as the iterated function systems (IFSs), L-systems and the escape-time algorithms. In this paper, we use the escape-time algorithm to get the classical fractals by some specific folding and expanding mappings. Finally, we clearly give the maple codes that these fractals are obtained.

Keywords: Classical fractals, folding mappings, expanding mappings, the escape-time algorithm.

1 Introduction

In recent years, there have been many studies on fractals since there is a close relationship between many shapes in the nature and models of fractals ([1,2,3,4,5,6,7]). So, we encounter various techniques to generate fractals in the literature. One of the most important techniques is the iterated function systems (IFS). The IFS theory was first created by Hutchinson and then supported by Barnsley's works. Simultaneously, many scientists contributed to the construction of the fractal theory and to generate various fractal models. For example, the classical fractals such as Cantor set, Sierpinski Gasket, Koch curve, Sierpinski carpet, Vicsek fractal (or box fractal) can be generated by IFSs. Thus they have exact self-similarity property. That is, the whole shape can be seen in every smaller pieces of these fractals. The Newton fractal, the Mandelbrot set and the Julia sets can be generated by the escape time algorithm. From these sets, the Julia sets of f_c ($c \in \mathbb{C}$) are defined as the boundary of the filled-in set

$$\{z \in \mathbb{C} \mid \{f_c^n(z)\} \text{ is bounded}\}$$

where $f_c : \mathbb{C} \to \mathbb{C}$, $f_c(z) = z^2 + c$ (for details see [6]). Furthermore, the branching patterns such as in plants, biological cells, blood vessels, pulmonary structure or turtle graphics patterns such as space-filling curves and tilings are generated by L-systems (for details see [8]).

In this paper, by using the escape time algorithm, we construct classical fractals such as the Cantor set, the Cantor dust, the classical Sierpinski gasket, the right-angled Sierpinski gasket, the Koch curve, the Sierpinski carpet and the Vicsek fractal. To this end, for every fractals we use special folding, expanding, affine, translation and rotation mappings. Note that, to generate these fractals we apply the escape-time algorithm for continuous functions different from Barnsley's method below:

* Corresponding author e-mail: mustafasaltan@anadolu.edu.tr

To obtain the right Sierpinski triangle (S) via the escape time algorithm, Barnsley uses the IFS $\{\mathbb{R}^2, w_1, w_2, w_3\}$ where $w_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3)$

$$w_1(x,y) = \left(\frac{x}{2}, \frac{y}{2} + \frac{1}{2}\right),$$
$$w_2(x,y) = \left(\frac{x}{2} + \frac{1}{2}, \frac{y}{2}\right),$$
$$w_3(x,y) = \left(\frac{x}{2}, \frac{y}{2}\right).$$

There is a relationship between the function

$$f(x,y) = \begin{cases} (2x, 2y-1), & \text{if } y > 0.5 \\ (2x-1, 2y), & \text{if } x > 0.5,\ y \le 0.5 \\ (2x, 2y), & \text{otherwise.} \end{cases} \tag{1}$$

and the IFS $\{w_1, w_2, w_3\}$ since f restricted to S satisfies

$$f(x,y) = \begin{cases} w_1^{-1}(x,y), & \text{if } (x,y) \in S - \{(0,0.5),(0.5,0.5)\} \\ w_2^{-1}(x,y), & \text{if } (x,y) \in S - \{(0.5,0)\} \\ w_3^{-1}(x,y), & \text{if } (x,y) \in S \end{cases}$$

and $f|_S$ is surjective (for details see [2]). Moreover, if $(x,y) \notin S$, then $\{f^n(x,y)\}$ diverges to infinity. So, this fractal can be generated by the escape time algorithm and consequently, Figure 1 is obtained. Note that f is not continuous. Similarly,

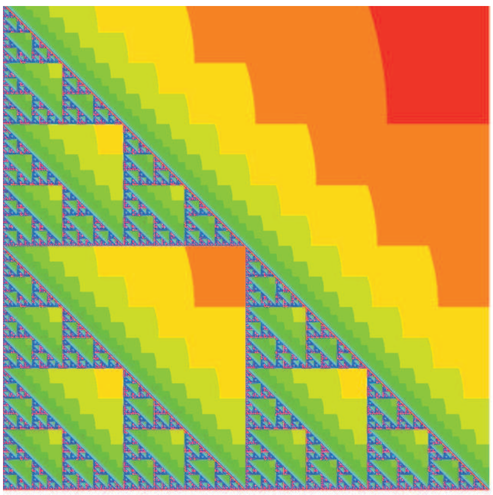

Fig. 1: The construction of the right Sierpinski triangle by using the function defined in (1) via the escape time algorithm

this method can be applied any fractals which can be expressed by the attractor of an IFS. In the following, in order to obtain the image of the classical fractals we define different functions compatible with the structure of each set. Although we again use the escape time algorithm, the functions which we defined in Examples 1, 2, 3, 4 5, 6, and 7 do not depend on their IFSs and are continuous.

2 The construction of the classical fractals by using different mappings via the escape time algorithm

In this section, we will show that each classical fractal can be obtained by the compositions of some functions such as expanding, folding, translations, rotations and affine mappings. Figures 3, 5, 7, 9, 11, 13, 15, and 16 are generated by the mathematical program Maple 18. Finally, we merely give the codes of the right Sierpinski triangle. To get another figures substitute with the corresponding functions.

The Cantor set: It is well-known that the Cantor set (C) is the attractor of the IFS $\{\mathbb{R}, w_1, w_2\}$ where $w_i : \mathbb{R} \to \mathbb{R}$ $(i = 1,2)$

$$w_1(x) = \frac{x}{3},$$
$$w_2(x) = \frac{x}{3} + \frac{2}{3}.$$

Example 1. By the escape time algorithm, the set which is obtained by using the composition function $F : \mathbb{R} \to \mathbb{R}$

$$F = f_2 \circ f_1$$

where $f_i : \mathbb{R} \to \mathbb{R}$ $(i = 1,2)$

$$f_1(x) = 3x,$$
$$f_2(x) = \frac{3}{2} - \left| x - \frac{3}{2} \right|$$

is the Cantor set.

We first define the composition function $F = f_2 \circ f_1$ where f_1 is an expanding and f_2 is a folding mapping that takes the points from the right hand side of the line $x = \frac{3}{2}$ to the the left hand side of the line (see Figure 2).

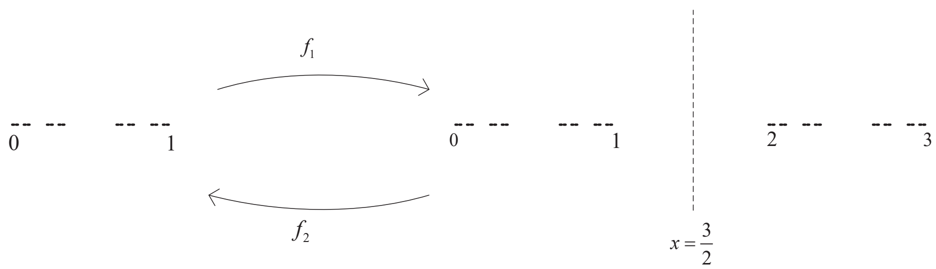

Fig. 2: An expanding and a folding mapping on the Cantor set

Note that F restricted to Cantor set is surjective. Since $\{F^n(x)\}$ diverges to infinity for $x \notin C$, the Cantor set can be generated by the escape time algorithm (for the images see Figure 3).

Fig. 3: The construction of the Cantor set via the escape time algorithm

The Cantor dust: The attractor of the IFS $\{\mathbb{R}^2, w_1, w_2, w_3, w_4\}$ where $w_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3,4)$

$$w_1(x,y) = \left(\frac{x}{3}, \frac{y}{3}\right),$$
$$w_2(x,y) = \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3}\right),$$
$$w_3(x,y) = \left(\frac{x}{3}, \frac{y}{3} + \frac{2}{3}\right),$$
$$w_4(x,y) = \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3} + \frac{2}{3}\right)$$

is called the Cantor dust (D).

Example 2. By the escape time algorithm, the set which is obtained by using the composition function $F : \mathbb{R}^2 \to \mathbb{R}^2$

$$F = f_3 \circ f_2 \circ f_1$$

where $f_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3)$

$$f_1(x,y) = (3x, 3y),$$
$$f_2(x,y) = \left(\frac{3}{2} - \left|x - \frac{3}{2}\right|, y\right),$$
$$f_3(x,y) = \left(x, \frac{3}{2} - \left|y - \frac{3}{2}\right|\right)$$

is the Cantor dust.

Firstly, define the composition function $F = f_3 \circ f_2 \circ f_1$ such that the function f_1 is an expanding mapping, f_2 is a folding mapping that takes the points from the right hand side of the line $x = \frac{3}{2}$ to the the left hand side and f_3 is a folding mapping that takes the points from the upper hand side of the line $y = \frac{3}{2}$ to the the lower hand side (see Figure 4). Figure

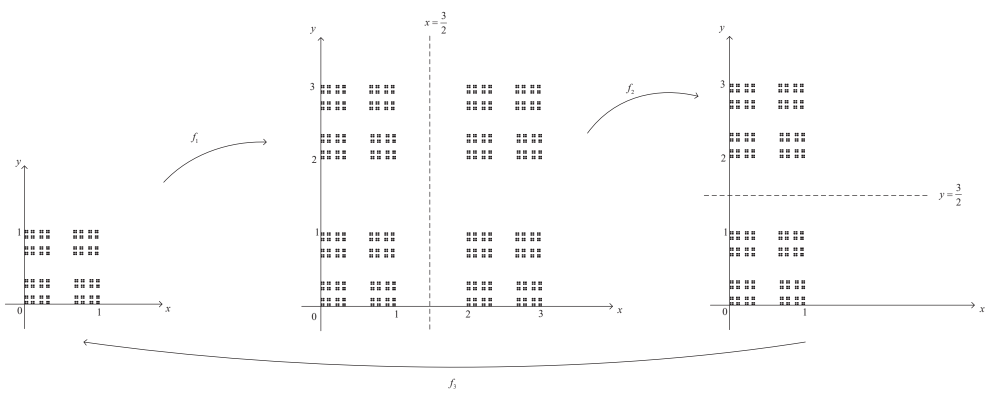

Fig. 4: An expanding and two folding mappings on the Cantor dust

4 shows that F restricted to Cantor dust is surjective. The Cantor dust can be generated by the escape time algorithm due to the fact that $\{F^n(x,y)\}$ diverges to infinity for $(x,y) \notin D$ (for the images see Figure 5).

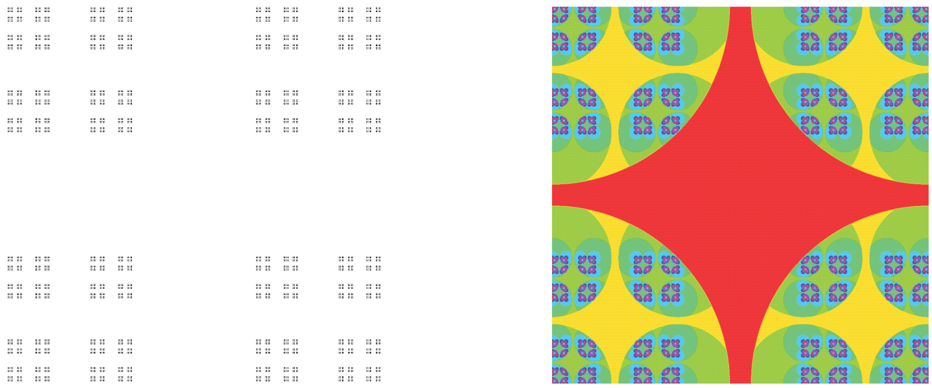

Fig. 5: The construction of the Cantor dust via the escape time algorithm

The Sierpinski triangle: The classical Sierpinski triangle (S) is the attractor of the IFS $\{\mathbb{R}^2, w_1, w_2, w_3\}$ where $w_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3)$

$$w_1(x,y) = \left(\frac{x}{2}, \frac{y}{2}\right),$$
$$w_2(x,y) = \left(\frac{x}{2} + \frac{1}{4}, \frac{y}{2} + \frac{\sqrt{3}}{4}\right),$$
$$w_3(x,y) = \left(\frac{x}{2} + \frac{1}{2}, \frac{y}{2}\right).$$

Example 3. By the escape time algorithm, the set which is obtained by using the composition function $F : \mathbb{R}^2 \to \mathbb{R}^2$

$$F = f_3 \circ f_2 \circ f_1$$

where $f_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3)$

$$f_1(x,y) = (2x, 2y),$$
$$f_2(x,y) = \left(-\frac{1}{2}\left|\frac{1}{2}(x-2+y\sqrt{3})\right| - \frac{\sqrt{3}}{2}\left(\frac{y-\sqrt{3}(x-2)}{2}\right) + 2, \frac{y-\sqrt{3}(x-2)}{4} - \frac{\sqrt{3}}{2}\left|\frac{x-2+y\sqrt{3}}{2}\right|\right),$$
$$f_3(x,y) = (1-|x-1|, y)$$

is the Sierpinski triangle.

Consider the composition function $F = f_3 \circ f_2 \circ f_1$ such that the function f_1 is an expanding mapping, f_2 is a folding mapping that takes the points from the upper hand side of the line

$$y = -\frac{\sqrt{3}x}{3} + \frac{2\sqrt{3}}{3}$$

to the the lower hand side and f_3 is a folding mapping that takes the points from the right hand side of the line $x = 1$ to the the left hand side. $F|_S : S \to S$ is a surjective function (see Figure 6). In particular, $\{F^n(x,y)\}$ diverges to infinity for

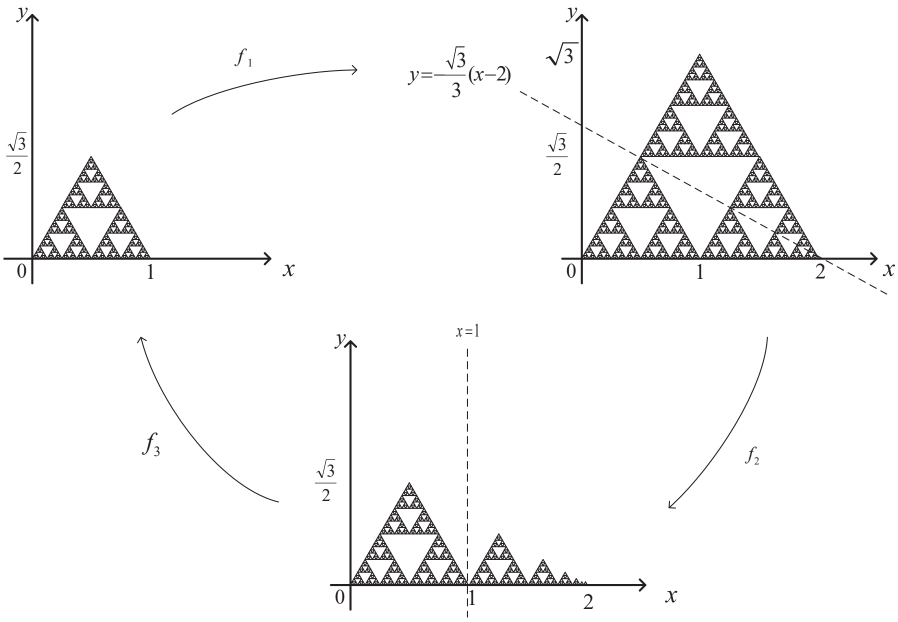

Fig. 6: An expanding and two folding mappings on the Sierpinski triangle

$(x,y) \notin S$. So, the Sierpinski gasket can be generated by the escape time algorithm (for the images see Figure 7).

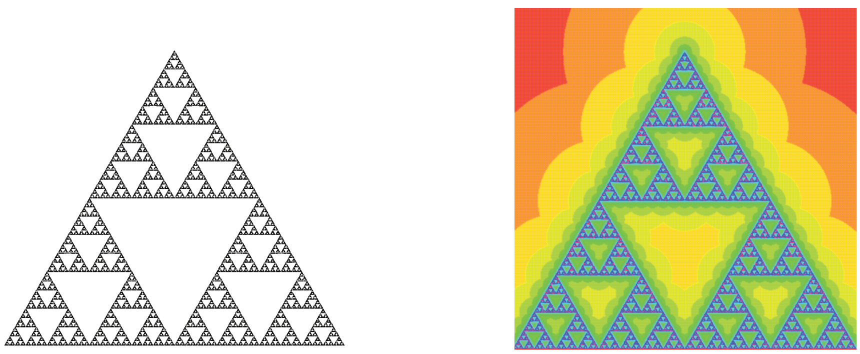

Fig. 7: The construction of the Sierpinski triangle by the escape time algorithm

The Koch curve: The Koch curve (K) is the attractor of the IFS $\{\mathbb{R}^2, w_1, w_2, w_3, w_4\}$ where $w_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3,4)$

$$w_1(x,y) = \left(\frac{x}{3}, \frac{y}{3}\right),$$

$$w_2(x,y) = \left(\frac{x}{6} - \frac{\sqrt{3}}{6}y + \frac{1}{3}, \frac{\sqrt{3}}{6}x + \frac{y}{6}\right),$$

$$w_3(x,y) = \left(\frac{x}{6} + \frac{\sqrt{3}}{6}y + \frac{1}{2}, -\frac{\sqrt{3}}{6}x + \frac{y}{6} + \frac{\sqrt{3}}{6}\right),$$

$$w_4(x,y) = \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3}\right).$$

Example 4. By the escape time algorithm, the set which is obtained by using the composition function $F:\mathbb{R}^2\to\mathbb{R}^2$

$$F = f_4\circ f_3\circ f_2\circ f_1$$

where $f_i:\mathbb{R}^2\to\mathbb{R}^2$ $(i=1,2,3,4)$

$$f_1(x,y) = \left(\sqrt{3}x,\sqrt{3}y\right),$$
$$f_2(x,y) = \left(\frac{\sqrt{3}}{2}-\left|x-\frac{\sqrt{3}}{2}\right|,y\right),$$
$$f_3(x,y) = \left(-\frac{\sqrt{3}}{2}x-\frac{y}{2},-\frac{\sqrt{3}}{2}y+\frac{x}{2}\right),$$
$$f_4(x,y) = (x+1,y)$$

is the Koch curve.

Let us consider the composition function $F = f_4\circ f_3\circ f_2\circ f_1$ where f_1 is an expanding mapping and f_2 is a folding mapping that takes the points from the right hand side of the line $x=\frac{\sqrt{3}}{2}$ to the the left hand side. The function f_3 is a rotation mapping which rotates the curve 150° as counter clockwise and the function f_4 is a translation mapping (see Figure 8).

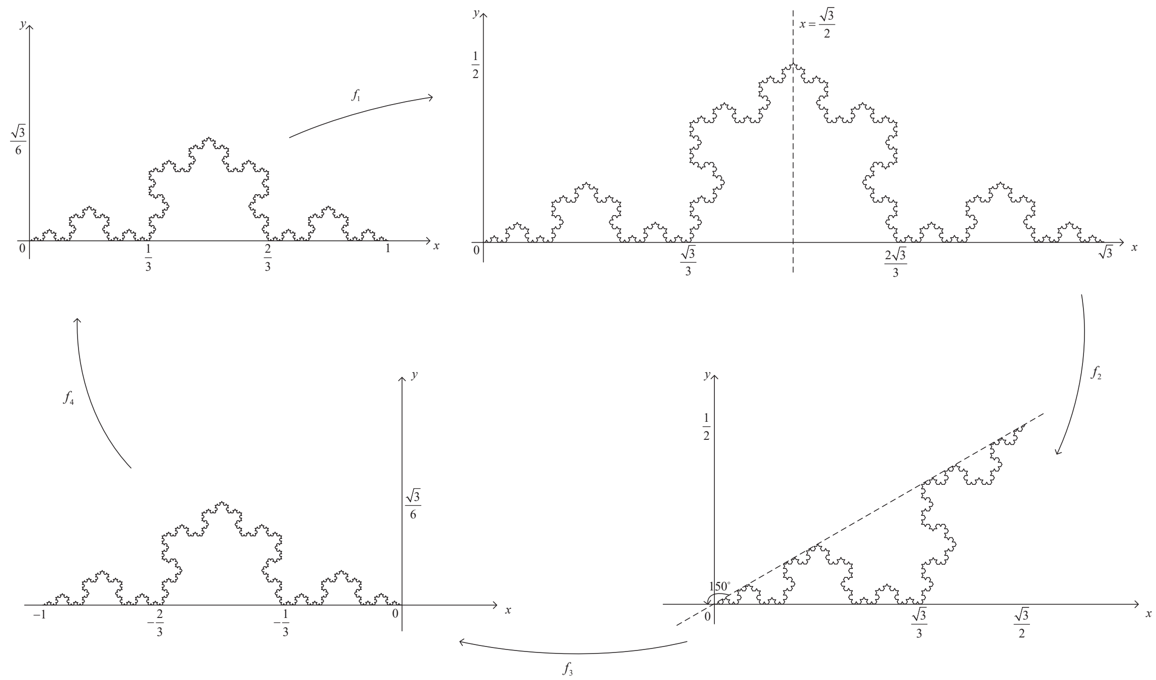

Fig. 8: An expanding, a folding, a rotation and a translation mappings on the Koch curve

Note that F restricted to the the Koch curve is surjective. Since $\{F^n(x,y)\}$ diverges to infinity for $(x,y)\notin K$, the Koch curve can be generated by the escape time algorithm (for the images see Figure 9).

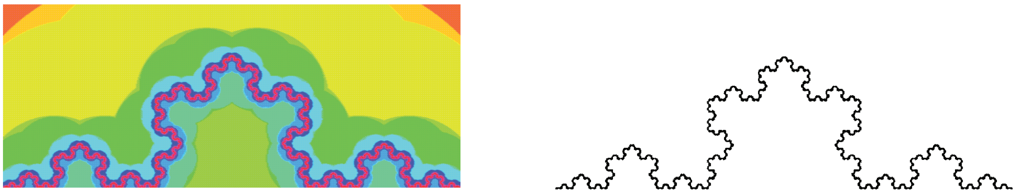

Fig. 9: The construction of the Koch curve via the escape time algorithm

The Vicsek fractal: The attractor of the IFS $\{\mathbb{R}^2, w_1, w_2, w_3, w_4, w_5\}$ where $w_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3,4,5)$

$$w_1(x,y) = \left(\frac{x}{3}, \frac{y}{3}\right),$$
$$w_2(x,y) = \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3}\right),$$
$$w_3(x,y) = \left(\frac{x}{3} + \frac{1}{3}, \frac{y}{3} + \frac{1}{3}\right),$$
$$w_4(x,y) = \left(\frac{x}{3}, \frac{y}{3} + \frac{2}{3}\right),$$
$$w_5(x,y) = \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3} + \frac{2}{3}\right)$$

is the Vicsek fractal (V).

Example 5. By the escape time algorithm, the set which is obtained by using the composition function $F : \mathbb{R}^2 \to \mathbb{R}^2$

$$F = f_4 \circ f_3 \circ f_2 \circ f_1$$

where $f_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3,4)$

$$f_1(x,y) = (3x, 3y),$$
$$f_2(x,y) = \left(x, \frac{3}{2} - \left|y - \frac{3}{2}\right|\right),$$
$$f_3(x,y) = \left(\frac{3}{2} - \left|x - \frac{3}{2}\right|, y\right),$$
$$f_4(x,y) = \left(-\frac{1}{2}|x+y-2| - \frac{1}{2}(y-x-2), \frac{1}{2}(y-x+2) - \frac{1}{2}|x+y-2|\right)$$

is the Vicsek fractal.

The function f_1 is an expanding mapping. The function f_2 is a folding mapping that takes the points from the upper hand side of the line $y = \frac{3}{2}$ to the the lower hand side, f_3 is a folding mapping takes the points from the right hand side of the line $x = \frac{3}{2}$ to the the left hand side and f_4 is a folding mapping that takes the points from the upper hand side of the line $y = 2 - x$ to the the lower hand side. We now define the composition function $F = f_4 \circ f_3 \circ f_2 \circ f_1$ (see Figure 10).

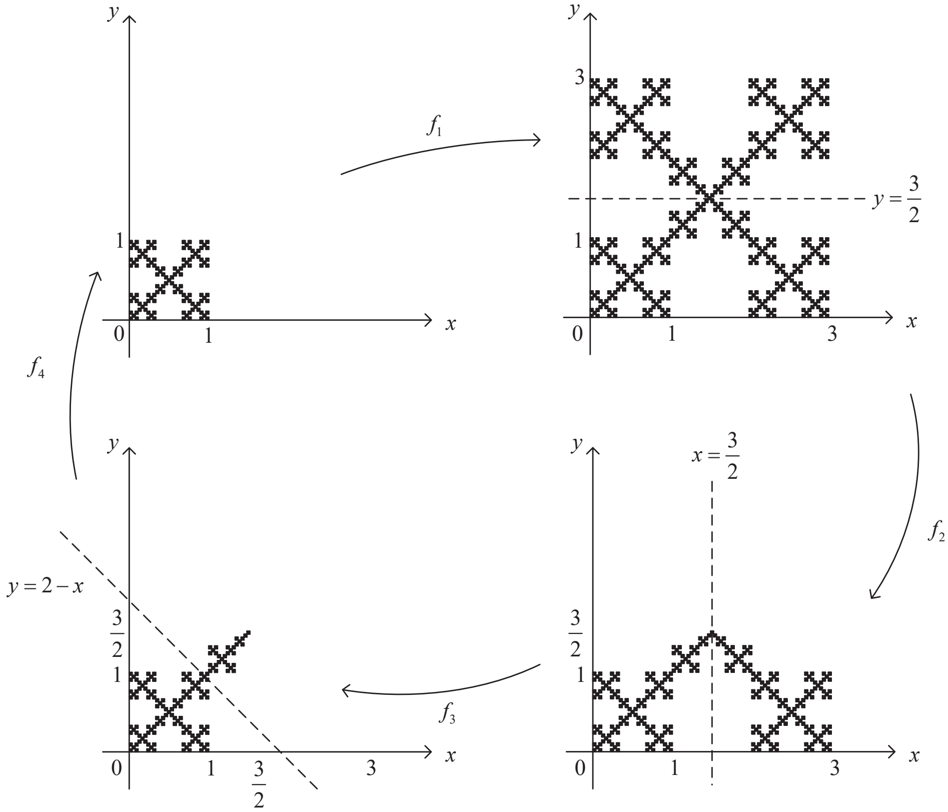

Fig. 10: An expanding mapping and three folding mappings on the Vicsek fractal

Due to the fact that F restricted to the Vicsek fractal is surjective and $\{F^n(x,y)\}$ diverges to infinity for $(x,y) \notin V$, the Vicsek fractal can be generated by the escape time algorithm (for the images see Figure 11).

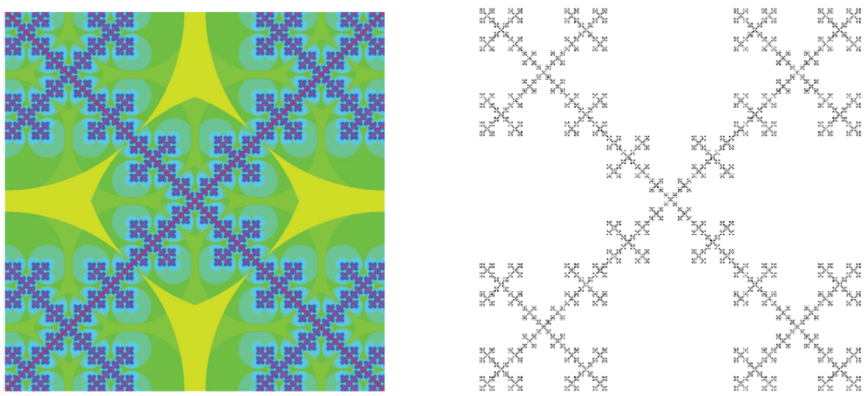

Fig. 11: The construction of the Vicsek fractal via the escape time algorithm

The Sierpinski Carpet: The Sierpinski carpet (SC) is the attractor of the IFS $\{\mathbb{R}^2, w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8\}$ where $w_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3,4,5,6,7,8)$

$$\begin{aligned}
w_1(x,y) &= \left(\frac{x}{3}, \frac{y}{3}\right), \\
w_2(x,y) &= \left(\frac{x}{3} + \frac{1}{3}, \frac{y}{3}\right), \\
w_3(x,y) &= \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3}\right), \\
w_4(x,y) &= \left(\frac{x}{3}, \frac{y}{3} + \frac{1}{3}\right), \\
w_5(x,y) &= \left(\frac{x}{3}, \frac{y}{3} + \frac{2}{3}\right), \\
w_6(x,y) &= \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3} + \frac{1}{3}\right), \\
w_7(x,y) &= \left(\frac{x}{3} + \frac{1}{3}, \frac{y}{3} + \frac{2}{3}\right), \\
w_8(x,y) &= \left(\frac{x}{3} + \frac{2}{3}, \frac{y}{3} + \frac{2}{3}\right).
\end{aligned}$$

Example 6. By the escape time algorithm, the set which is obtained by using the composition function $F : \mathbb{R}^2 \to \mathbb{R}^2$

$$F = f_5 \circ f_4 \circ f_3 \circ f_2 \circ f_1$$

where $f_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3,4,5)$

$$\begin{aligned}
f_1(x,y) &= (3x, 3y), \\
f_2(x,y) &= \left(\frac{1}{2}|x-y| + \frac{1}{2}(x+y), \frac{1}{2}(x+y) - \frac{1}{2}|x-y|\right), \\
f_3(x,y) &= \left(-\frac{1}{2}|x+y-3| - \frac{1}{2}(y-x-3), \frac{1}{2}(y-x+3) - \frac{1}{2}|x+y-3|\right), \\
f_4(x,y) &= (1 + |1-x|, y), \\
f_5(x,y) &= (2 - |x-2|, y), \\
f_6(x,y) &= (x-1, y)
\end{aligned}$$

is the Sierpinski carpet.

The function f_1 be an expanding mapping, the function f_2 is a folding mapping that takes the points from the upper hand side of the line $y = x$ to the the lower hand side, f_3 is a folding mapping that takes the points from the upper hand side of the line $y = 3 - x$ to the the lower hand side, f_4 is a folding mapping that takes the points from the left hand side of the line $x = 1$ to the the right hand side, f_5 is a folding mapping that takes the points from the right hand side of the line $x = 2$ to the the left hand side and f_6 is a translation. Let us consider the composition function $F = f_6 \circ f_5 \circ f_4 \circ f_3 \circ f_2 \circ f_1$ (see Figure 12).

Note that F restricted to the Sierpinski carpet is surjective. Since $\{F^n(x,y)\}$ diverges to infinity for $(x,y) \notin SC$, the Sierpinski carpet can be generated by the escape time algorithm (for the images see Figure 13).

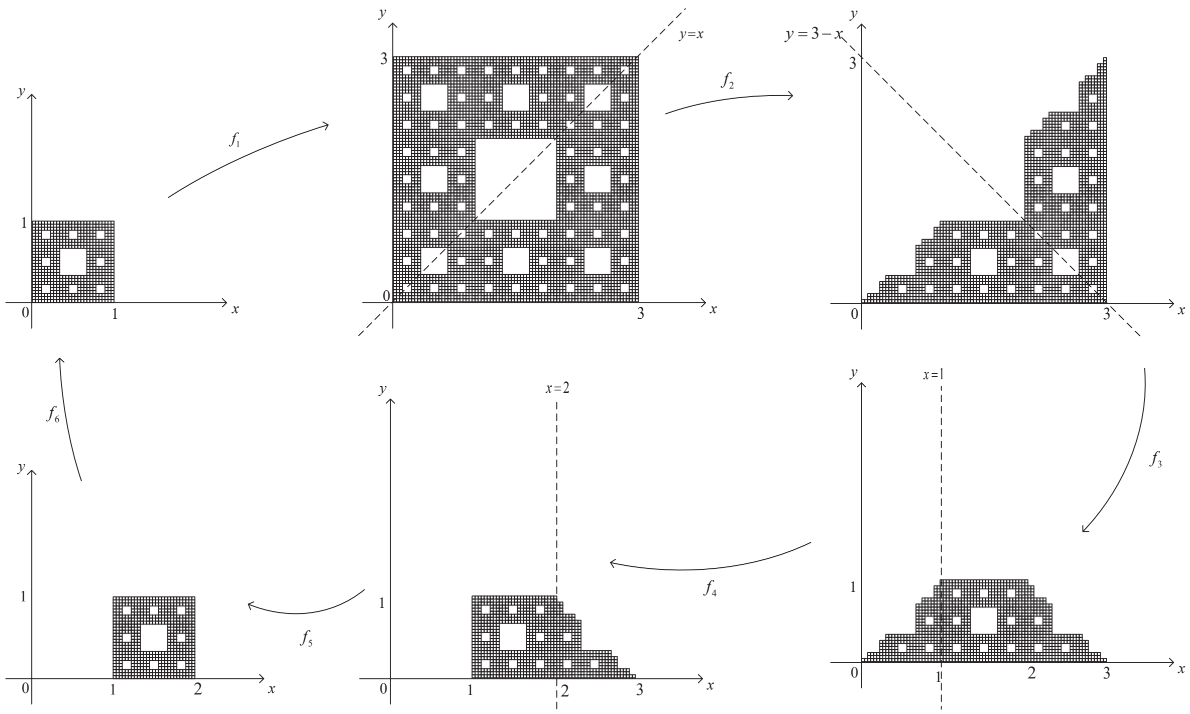

Fig. 12: An expanding mapping, a translation and four folding mappings on the Sierpinski carpet

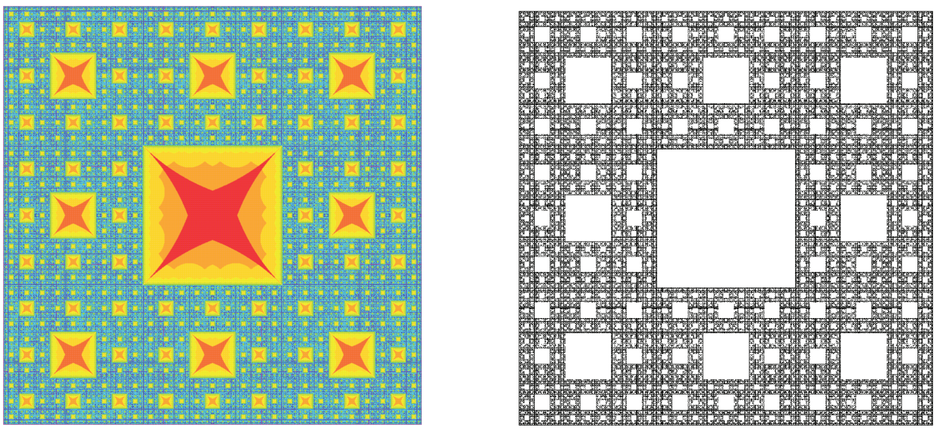

Fig. 13: The construction of the Sierpinski carpet via the escape time algorithm

The right Sierpinski triangle:

Example 7. In the escape time algorithm, the set which is obtained by using the composition function $F : \mathbb{R}^2 \to \mathbb{R}^2$

$$F = f_5 \circ f_4 \circ f_3 \circ f_2 \circ f_1 \tag{2}$$

where $f_i : \mathbb{R}^2 \to \mathbb{R}^2$ $(i = 1,2,3,4,5)$

$$\begin{aligned}
f_1(x,y) &= (2x,2y),\\
f_2(x,y) &= \left(x+\frac{y}{2},\frac{y\sqrt{3}}{2}\right),\\
f_3(x,y) &= \left(-\frac{1}{2}\left|\frac{1}{2}(x-2+y\sqrt{3})\right|-\frac{\sqrt{3}}{2}\left(\frac{y-\sqrt{3}(x-2)}{2}\right)+2,\frac{y-\sqrt{3}(x-2)}{4}-\frac{\sqrt{3}}{2}\left|\frac{x-2+y\sqrt{3}}{2}\right|\right),\\
f_4(x,y) &= (1-|x-1|,y),\\
f_5(x,y) &= \left(x-\frac{\sqrt{3}y}{3},\frac{2\sqrt{3}y}{3}\right)
\end{aligned}$$

is the right Sierpinski triangle.

Consider the composition function $F = f_5 \circ f_4 \circ f_3 \circ f_2 \circ f_1$ such that the function f_1 is an affine mapping from the equilateral Sierpinski triangle to the right Sierpinski triangle, the function f_2 is an expanding mapping, f_3 is a folding mapping that takes the points from the right hand side of the line

$$y = -\frac{\sqrt{3}x}{3}+\frac{2\sqrt{3}}{3}$$

to the the left hand side, f_4 is a folding mapping that takes the points from the right hand side of the line $x = 1$ to the the left hand side and the function f_5 is an affine mapping from the right Sierpinski triangle to the equilateral Sierpinski triangle (see Figure 14).

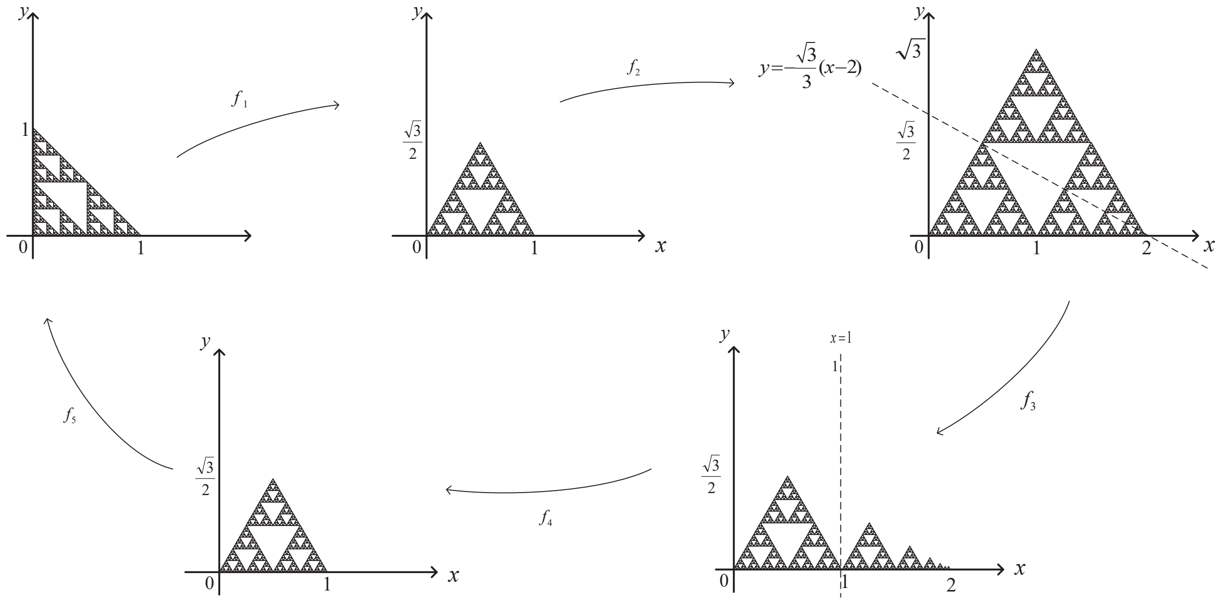

Fig. 14: An expanding, two folding and two affine mappings on the right Sierpinski triangle

Note that F restricted to the right Sierpinski triangle is surjective and $\{F^n(x,y)\}$ diverges to infinity if (x,y) is not an element of the right Sierpinski triangle. Hence, the right Sierpinski triangle can be generated by the escape time algorithm (for the images see Figures 15 and 16).

```
> with(plots):
  f[1]:=(x,y)->(evalf(2*x),evalf(2*y)):
  f[2]:=(x,y)->(evalf(x+y/2),evalf(y*sqrt(3)/2)):
  f[3]:=(x,y)->(-1/2*abs((x-2)/2+y*evalf(sqrt(3)/2))+evalf(sqrt(3)
  /2)*(evalf(sqrt(3)/2)*(x-2)-y/2)+2,
  -evalf(sqrt(3)/2)*abs((x-2)/2+y*evalf(sqrt(3)/2))+1/2*(y/2-evalf
  (sqrt(3)/2)*(x-2))):
  f[4]:=(x,y)->(evalf(1-abs(x-1)),evalf(y)):
  f[5]:=(x,y)->(evalf(x-y/sqrt(3)),evalf(2*y/sqrt(3))):
> F:=(x,y)->f[5](f[4](f[3](f[2](f[1](x,y))))):
> rightsierpinski := proc(a, b)
   local A,A1,A2, m;
   A[1]:=a:
   A[2]:=b:
  A1:=A[1]:
  A2:=A[2]:
  for m to 50 while A1^2+A2^2 < 2 do
      A := F(A[1], A[2]):
  A1:=A[1]:
  A2:=A[2]:
      end do;
   m;
  end proc:
  densityplot(rightsierpinski, 0..1, 0..1, colorstyle=HUE,
  grid=[1000, 1000], style=patchnogrid, axes=none);
```

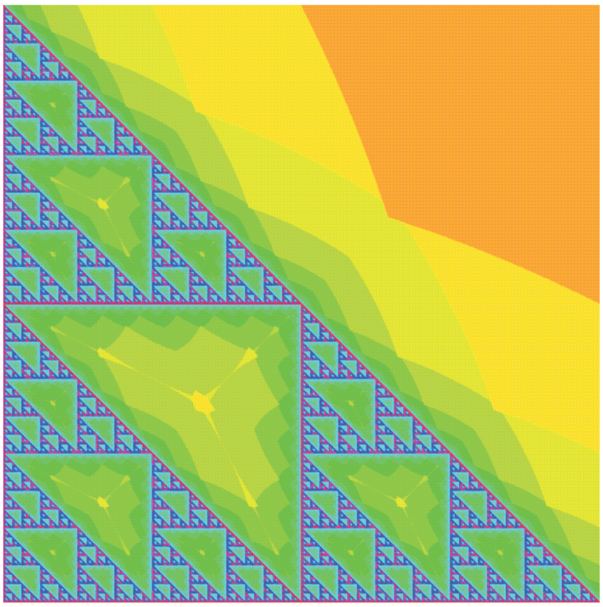

Fig. 15: The construction of the right Sierpinski triangle via the escape time algorithm

```
> restart;
> with(plots):
> f[1]:=(x,y)->(evalf(2*x),evalf(2*y)):
  f[2]:=(x,y)->(evalf(x+y/2),evalf(y*sqrt(3)/2)):
  f[3]:=(x,y)->(-1/2*abs((x-2)/2+y*evalf(sqrt(3)/2))+evalf(sqrt(3)
  /2)*(evalf(sqrt(3)/2)*(x-2)-y/2)+2,-evalf(sqrt(3)/2)*abs((x-2)/2
  +y*evalf(sqrt(3)/2))+1/2*(y/2-evalf(sqrt(3)/2)*(x-2))):
  f[4]:=(x,y)->(evalf(1-abs(x-1)),evalf(y)):
  f[5]:=(x,y)->(evalf(x-y/sqrt(3)),evalf(2*y/sqrt(3))):
  F:=(x,y)->f[5](f[4](f[3](f[2](f[1](x,y))))):
> step:=0.001:
> say:=0:
  for i from 0 to 1 by step do
  for j from 0 to 1 by step do
  a:=evalf(F(i,j)):
  for k from 1 to 8 do
  a:=evalf(F(a[1],a[2])):
  od:
  if evalf(a[1]^2)+evalf(a[2]^2)<=1 then
  say:=say+1:
  depo[say]:=(i,j);
  fi:
  od:
  od:
> pointplot({seq([depo[t][1],depo[t][2]],t=1..say)},axes=none,symb
  ol=point);
  say;
```

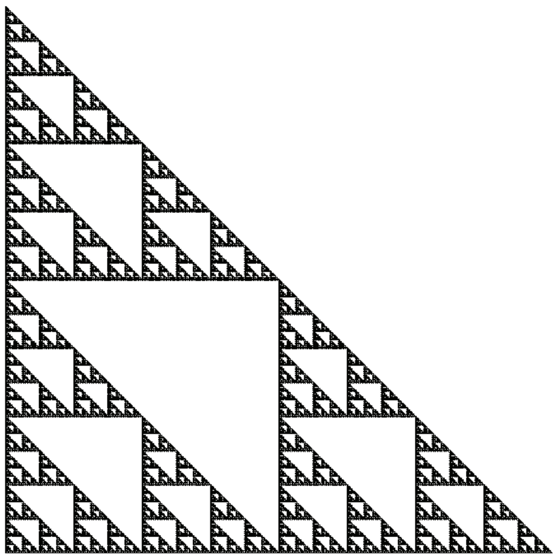

Fig. 16: The construction of the right Sierpinski triangle without coloring

3 Conclusions

Fractals can be generated by different ways. This paper presents a different perspective to construct the classical fractals via the escape time algorithm. This method can be also applied to generate different fractal sets.

References

[1] Barnsley, M.F. *Superfractals*; Cambridge University Press: New York, NY, USA, 2006.
[2] Barnsley, M.F. *Fractals Everywhere*; Dover Publications, Inc: Mineola, New York, USA, 2012.
[3] Edgar, G. *Measure, topology, and fractal geometry*; Springer Undergraduate Texts in Mathematics, 2008.
[4] Falconer, K.J. *Fractal Geometry*; Mathematical Foundations and Application, John Wiley: UK, 2014.
[5] Hutchinson, J.E. Fractals and self-similarity. *Indiana Univ. Math. J.* **1981**, *30*, 713–747.
[6] Mandelbrot, B.B. *The fractal geometry of nature*; W. H. Freeman, San Francisco, 1982.
[7] Peitgen, H.O.; Jürgens, H.; Saupe, D. *Chaos and fractals*; New frontiers of science, Springer-Verlag, 2004.
[8] Prusinkiewicz, P. Graphical applications of L-systems. *Proceedings of graphics interface* **1986**, 247–253.