Numerical Algorithms (2024) 96:211–236
https://doi.org/10.1007/s11075-023-01644-4

ORIGINAL PAPER

On the viscosity approximation type iterative method and its non-linear behaviour in the generation of Mandelbrot and Julia sets

Sudesh Kumari[1] · Krzysztof Gdawiec[2] · Ashish Nandal[3] · Naresh Kumar[4] · Renu Chugh[5]

Received: 25 November 2022 / Accepted: 4 August 2023 / Published online: 17 August 2023

Abstract

In this paper, we visualise and analyse the dynamics of fractals (Julia and Mandelbrot sets) for complex polynomials of the form $T(z) = z^n + mz + r$, where $n \geq 2$ and $m, r \in \mathbb{C}$, by adopting the viscosity approximation type iteration process which is most widely used iterative method for finding fixed points of non-linear operators. We establish a convergence condition in the form of escape criterion which allows to adapt the escape-time algorithm to the considered iteration scheme. We also present some graphical examples of the Mandelbrot and Julia fractals showing the dependency of Julia and Mandelbrot sets on complex polynomials, contraction mappings, and iteration parameters. Moreover, we propose two numerical measures that allow the study of the dependency of the set shape change on the values of the iteration parameters. Using these two measures, we show that the dependency for the considered iteration method is non-linear.

Keywords Viscosity approximation type method · Escape criterion · Julia set · Mandelbrot set

1 Introduction

For many years, people thought that objects surrounding us in nature, such as mountains and ferns, could be described using classical Euclidean geometry. However, we cannot do this because many natural objects do not follow simple rules. Most of them share the self-similarity property, i.e., every small part contains a copy of the whole shape, so classical Euclidean geometry cannot model such a property [17]. Fractal geometry is a tool that provides a general framework to analyse such objects. Fractals

✉ Krzysztof Gdawiec
krzysztof.gdawiec@us.edu.pl

Extended author information available on the last page of the article

found applications in various practical problems, for example, in the analysis of river networks [33], image encryption [53] or compression [2], pattern recognition [13], generating art [40], cryptography [1]. Fractal antennas have attracted increasing attention in recent years due to their compact design and superior wideband performance [7, 23]. In biology and medicine, fractals are used to study the nerve system, the culture of micro organs, etc. [10].

One of the fractal types extensively studied in the literature is the Mandelbrot and Julia sets. Mandelbrot introduced the images of these fractals in the late 1970 s. The Julia sets were introduced much earlier by French mathematicians Pierre Fatou [11], and Gaston Julia [21], but they were unable to sketch them manually. The invention of computers brought a breakthrough in the visualisation of those sets. Mandelbrot was the first who plotted the Julia set of the quadratic function $z^2 + c$, where $c \in \mathbb{C}$. Since then, Mandelbrot and Julia's sets have been extended in various directions. One such direction is the replacement of complex numbers with other types of numbers such as quaternions [8], bicomplex numbers [52], tricomplex numbers [42], and trinition numbers [4]. Another direction was the use of various types of functions instead of the quadratic polynomial function, e.g., trigonometric [22], exponential [34], rational [5], and Möbius transformations [35]. Moreover, many methods of visualisation of those sets were introduced, see [20, 30, 51].

The most significant contribution in the generation of fractals (especially Julia and Mandelbrot sets) is presented by various fixed-point iterative methods. Fixed point iterative methods provide ways for finding fixed points of non-linear mappings. Most fixed-point iterative methods can be classified into two types: (1) the Mann-type iterative method and (2) the Halpern-type iterative method. The Mann iterative method is an averaged iterative method. It was introduced by Mann in 1953 [32]. However, the Mann iteration does not converge strongly in general [14, 47]. Many attempts have been made to modify the Mann iteration method to guarantee strong convergence.

Halpern [16] in 1967 proposed a new iterative method for approximate finding of fixed points of nonexpansive mappings. One of the important generalisations of the Halpern method is the viscosity approximation method. This method was introduced by Moudafi [36], and since its introduction it has been widely used for finding fixed points of a nonexpansive mappings and other classes of non-linear mappings (see [18, 37–39, 43] and references therein). There is extensive literature regarding the convergence analysis of the viscosity approximation method with several types of operators. Maingé [31] discussed the convergence of this method with respect to the computation of fixed points of operators in the broad class of quasi-nonexpansive mappings.

The literature review shows that previous studies have almost exclusively focused on Mann and its similar type fixed-point iterative procedures for generating fractals (especially Julia and Mandelbrot sets). In this group of methods, we can distinguish two types of iterations, i.e., explicit and implicit iterations. For each of the types, we can find many examples of iterations used in the generation of fractals, e.g., for the explicit type, we find: the Mann iteration [45, 46], the Picard-Mann iteration [55], the Ishikawa iteration [6], the Noor iteration [3], the CR iteration [29], the SP iteration with s-convexity [26], the K-iteration [54], the F-iteration [49], the Fibonacci-Mann iteration [41] and for the implicit ones: the Jungck-SP iteration [28], the Jungck-SP iteration with s-convexity [27], the Jungck-CR iteration [48], the Jungck-DK iteration [15], the Jungck-Mann iteration [50]. Recently, two papers on the use of viscosity type approximation methods appeared. In [25], Kumari et al. studied the use of the Moudafi viscosity approximation method in the generation of Mandelbrot and Julia sets and biomorphs. Then, in [24], Kumari et al. studied a generalised viscosity approximation type method proposed by Nandal et al. in [39] for the generation of Mandelbrot and Julia sets. Iteration methods are not only used in the generation of Mandelbrot and Julia sets. We can find their use in the generation of other types of fractals, e.g., iterated function system fractals [44], inversion fractals [12], biomorphs [19, 25].

Due to the explosive number of research articles in the last two decades on the viscosity approximation method in the field of fixed point theory, we feel that this method has great potential for other applications, such as the generation of fractals. Motivated by this fact, our paper proposes a new approach to generate fractals (Julia and Mandelbrot sets) based on the viscosity approximation method presented by Maingé [31]. We adapt the escape-time algorithm to the considered viscosity approximation method. Moreover, we propose two numerical measures that allow for the study of the dependence of the set shape change on the values of the iteration parameters.

The remainder of the paper is organised as follows. Section 2 is dedicated to some basic definitions. In Section 3, we define the viscosity Mandelbrot and Julia sets. Moreover, we obtain the escape criterion, which is the main key to drawing viscosity Julia and Mandelbrot sets. In Section 4, we introduce the adapted escape-time algorithms to generate viscosity Mandelbrot and Julia sets and present some graphical examples of sets generated with those algorithms. In Section 5, we introduce two numerical measures that can be used to study the dependence of the change in shape of the set on the values of the iteration method. Moreover, we present this dependency for the considered iteration method. Finally, in Section 6, we conclude our findings.

2 Preliminaries

In this section, we recollect some basic definitions which are taken into account throughout the paper.

Definition 1 **(Julia set** [9]**)** The *filled Julia set* F_{f_r} of a complex-valued function $f_r : \mathbb{C} \to \mathbb{C}$, where $r \in \mathbb{C}$ is a parameter, is defined as

$$F_{f_r} = \{z \in \mathbb{C} : \{|f_r^j(z)|\}_{j=0}^{\infty} \text{ is bounded}\}, \tag{1}$$

where f_r^j denotes the j times composition of the function f_r. The *Julia set* J_{f_r} of f_r consists of the boundary points of the filled Julia set F_{f_r}, i.e., $J_{f_r} = \partial F_{f_r}$.

Definition 2 **(Mandelbrot set** [9]**)** Let $f_r : \mathbb{C} \to \mathbb{C}$ be a complex-valued function, where $r \in \mathbb{C}$ is a parameter. The *Mandelbrot set* M is defined as the the set of all numbers $r \in \mathbb{C}$ for which the filled Julia set F_{f_r} is connected, i.e.,

$$M = \{r \in \mathbb{C} : F_{f_r} \text{ is connected}\}. \tag{2}$$

Equivalently,

$$M = \{r \in \mathbb{C} : |f_r^j(0)| \nrightarrow \infty \text{ as } j \to \infty\}. \tag{3}$$

The first viscosity approximation method was investigated by Moudafi in 2000 [36]. This method – in the complex plane – is defined as

Definition 3 (Viscosity approximation method) Let $T : \mathbb{C} \to \mathbb{C}$ be a complex map. Consider a sequence z_j given by

$$z_{j+1} = \alpha_j g(z_j) + (1 - \alpha_j)T(z_j), \quad j \geq 0, \tag{4}$$

where $z_0 \in \mathbb{C}$ is a starting point, $\alpha_j \in (0, 1)$ and $g : \mathbb{C} \to \mathbb{C}$ is a contraction mapping. The z_j sequence is called *viscosity approximation method.*

Note that if in (4), we take a constant mapping $g(z) = b$, where $b \in \mathbb{C}$, then the sequence z_j is known as the Halpern iteration [16].

Maingé [31] considered the following viscosity approximation method which is a special variant of (4): starting with an arbitrary initial point $z_0 \in \mathbb{C}$, z_j generated by

$$z_{j+1} = \alpha_j g(z_j) + (1 - \alpha_j)T_\lambda z_j, \quad j \geq 0, \tag{5}$$

where $T_\lambda = \lambda I + (1 - \lambda)T$, with T quasi-nonexpansive mapping, I identity mapping and $\alpha_j, \lambda \in (0, 1)$.

For simplicity, throughout the article we consider $\alpha_j = \alpha$, where $\alpha \in (0, 1)$.

3 Viscosity Mandelbrot and Julia sets

In complex space, we consider the following viscosity approximation-type orbit z_j given by Maingé [31]:

$$\begin{aligned} z_{j+1} &= \alpha g(z_j) + (1 - \alpha)y_j \\ y_j &= \lambda z_j + (1 - \lambda)T(z_j), \quad j \geq 0, \end{aligned} \tag{6}$$

where $z_0 \in \mathbb{C}$ is a starting point, T is a complex polynomial, $g : \mathbb{C} \to \mathbb{C}$ is a contraction mapping and $\alpha, \lambda \in (0, 1)$ are parameters.

Let us consider the following complex polynomial of degree n:

$$T_r(z) = z^n + mz + r, \tag{7}$$

where $n \geq 2$ and $m, r \in \mathbb{C}$. Moreover, let $g(z) = az + b$ be a complex contraction with $a, b \in \mathbb{C}$ and $|a| < 1$.

Definition 4 (Viscosity Julia set) The *viscosity filled Julia set* F'_{T_r} of T_r given in (7), is defined as

$$F'_{T_r} = \{z_0 \in \mathbb{C} : \{z_j\}_{j=0}^{\infty} \text{ is bounded}\}, \tag{8}$$

where z_j is defined by (6). The *viscosity Julia set* J'_{T_r} of T_r consists of the boundary points of the viscosity filled Julia set F'_{T_r}, i.e., $J'_{T_r} = \partial F'_{T_r}$.

Definition 5 (Viscosity Mandelbrot set) Let T_r be a function given by (7), where $r \in \mathbb{C}$ is a parameter. The *viscosity Mandelbrot set* M' is defined as

$$M' = \{r \in \mathbb{C} : |z_j| \nrightarrow \infty \text{ as } j \to \infty\}, \tag{9}$$

where $z_0 = 0$ and z_j for $j > 0$ is defined by (6) with T_r as the operator.

In the literature, various methods such as escape time, potential function, iterated function systems, and distance estimator algorithms have been considered to construct and analyse fractals. Among the methods the escape time algorithm is one of the most important methods used in the generation of fractals. The escape criterion is a condition which allows us to test whether the orbit of an initial point escapes to infinity or not.

Now, we derive the general escape criterion which can be used to generate the viscosity Julia and Mandelbrot sets.

Theorem 1 *Assume that*

$$|z_0| \geq \max\{|r|, |b|\} > \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}, \tag{10}$$

where $\alpha, \lambda \in (0, 1)$. *Let the sequence* z_j *be defined as*

$$\begin{aligned} z_{j+1} &= \alpha g(z_j) + (1-\alpha) y_j \\ y_j &= \lambda z_j + (1-\lambda) T_r(z_j), \quad j \geq 0. \end{aligned} \tag{11}$$

Then $\lim_{j\to\infty} |z_j| = \infty$.

Proof Consider from (11)

$$\begin{aligned} |y_0| &= |\lambda z_0 + (1-\lambda) T_r(z_0)| \\ &= |\lambda z_0 + (1-\lambda)(z_0^n + m z_0 + r)| \\ &= |(1-\lambda) z_0^n + (1-\lambda) m z_0 + (1-\lambda) r + \lambda z_0| \\ &\geq |(1-\lambda) z_0^n + (1-\lambda) m z_0 + \lambda z_0| - (1-\lambda)|r|. \end{aligned}$$

Our assumption $|z_0| \geq \max\{|r|, |b|\}$ yields that $-|r| \geq -|z_0|$, therefore, we have

$$\begin{aligned} |y_0| &\geq |(1-\lambda) z_0^n + (1-\lambda) m z_0 + \lambda z_0| - (1-\lambda)|z_0| \\ &\geq |(1-\lambda) z_0^n + (1-\lambda) m z_0| - |\lambda z_0| - (1-\lambda)|z_0| \\ &= |(1-\lambda) z_0^n + (1-\lambda) m z_0| - |z_0| \\ &\geq |(1-\lambda) z_0^n| - |(1-\lambda) m z_0| - |z_0| \\ &= (1-\lambda)|z_0|^n - (1-\lambda)|m||z_0| - |z_0| \\ &= |z_0| \left((1-\lambda)|z_0|^{n-1} - (1-\lambda)|m| - 1\right). \end{aligned}$$

Thus, we have

$$|y_0| \geq |z_0| \left((1-\lambda)|z_0|^{n-1} - (1-\lambda)|m| - 1\right). \tag{12}$$

From (11), consider

$$\begin{aligned} |z_1| &= |\alpha g(z_0) + (1-\alpha)y_0| \\ &= |\alpha(az_0 + b) + (1-\alpha)y_0| \\ &\geq |(1-\alpha)y_0| - |\alpha(az_0 + b)| \\ &\geq (1-\alpha)|y_0| - \alpha|az_0| - \alpha|b|. \end{aligned}$$

Our assumption $|z_0| \geq \max\{|r|, |b|\}$ yields that $-|b| \geq -|z_0|$, therefore, we have

$$\begin{aligned} |z_1| &\geq (1-\alpha)|y_0| - \alpha|a||z_0| - \alpha|z_0| \\ &> (1-\alpha)|y_0| - \alpha|z_0| - \alpha|z_0|. \end{aligned}$$

Using (12), we have

$$\begin{aligned} |z_1| &> (1-\alpha)(|z_0| \left((1-\lambda)|z_0|^{n-1} - (1-\lambda)|m| - 1\right)) \\ &- 2\alpha|z_0| = |z_0| \left((1-\alpha)(1-\lambda)|z_0|^{n-1}\right. \\ &\left.-(1-\alpha)(1-\lambda)|m| - \alpha - 1\right). \end{aligned} \tag{13}$$

From the assumption $|z_0| > \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}$, we get

$$(1-\alpha)(1-\lambda)|z_0|^{n-1} - (1-\alpha)(1-\lambda)|m| - \alpha - 1 > 1. \tag{14}$$

Thus, there exists a real number $\sigma > 0$ such that

$$(1-\alpha)(1-\lambda)|z_0|^{n-1} - (1-\alpha)(1-\lambda)|m| - \alpha - 1 > 1 + \sigma > 1.$$

From (13), we have

$$|z_1| > (\sigma + 1)\ |z_0|.$$

In particular, $|z_1| > |z_0|$, so we may apply the same argument repeatedly to obtain

$$|z_j| > (\sigma + 1)^j |z_0|.$$

Hence, $|z_j| \to \infty$ as $j \to \infty$. □

In the proof of Theorem 1, we have used only the fact that $|z_0| > \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}$ and $|z_0| \geq \max\{|r|, |b|\}$. So, we can refine it and obtain the following corollary.

Corollary 1 *Let* $|z_0| > \max\left\{|r|, |b|, \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}\right\}$, *where* $a, b, m, r \in \mathbb{C}$ *with* $|a| < 1$, *and* $\alpha, \lambda \in (0, 1)$. *Then* $\lim_{j\to\infty} |z_j| = \infty$.

Corollary 2 *Suppose that:*

$$|z_k| > \max\left\{|r|, |b|, \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}\right\} \tag{15}$$

for some $k \geq 0$, where $a, b, m, r \in \mathbb{C}$ with $|a| < 1$, and $\alpha, \lambda \in (0, 1)$. Then, there exists $\sigma > 0$ such that $|z_{k+j}| > (1+\sigma)^j |z_k|$ and we have $\lim_{j\to\infty} |z_j| = \infty$.

4 Graphical examples of the viscosity Mandelbrot and Julia sets

With the help of Corollaries 1 and 2, we generate viscosity filled Julia sets of nth degree complex polynomials $T_r(z) = z^n + mz + r$ where $n \geq 2$ and $m, r \in \mathbb{C}$ using the escape-time algorithm as follows. If any point of the orbit z_j lies outside the circle of radius:

$$R = \max\left\{|r|, |b|, \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}\right\}, \tag{16}$$

then we know that the orbit of $|z_0|$ escapes to infinity, and in consequence z_0 does not belong to the viscosity filled Julia set. If z_j does not exceed the circle with radius R, then the point z_0 stays in a bounded area and in consequence it belongs to the viscosity filled Julia set. Algorithm 1 presents the pseudocode of the described escape-time algorithm for generating the viscosity Julia set. In the algorithm, we generate the viscosity Julia set in the given area of the complex space $A \subset \mathbb{C}$ using some colour map. If a point is a non-escaping point, then we need to perform an infinite number of iterations, which leads to an infinite loop. Thus, to avoid the infinite loop, we limit the maximal number of iterations to K iterations.

Algorithm 1 Viscosity Julia set generation.

```
Input: $T_r(z) = z^n + mz + r$, where $m, r \in \mathbb{C}$ and $n \geq 2$; $A \subset \mathbb{C}$ – area in which we draw the set;
       $K$ – the maximum number of iterations; $\alpha, \lambda \in (0, 1)$ – parameters for the viscosity
       approximation type iterative method; $g(z) = az + b$, where $a, b \in \mathbb{C}$ and $|a| < 1$;
       $colourmap[0..K]$ – colour map with $K + 1$ colours.
Output: Viscosity Julia set for area $A$.
$R = \max\left\{|r|, |b|, \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}\right\}$
for $z_0 \in A$ do
    $j = 0$
    while $|z_j| < R$ and $j < K$ do
        $y_j = \lambda z_j + (1-\lambda)T_r(z_j)$
        $z_{j+1} = \alpha g(z_j) + (1-\alpha)y_j$
        $j = j + 1$
    colour $z_0$ with $colourmap[j]$
```

The pseudocode for the escape-time algorithm generating the viscosity Mandelbrot set is presented in Algorithm 2. Like in the case of Algorithm 1, it uses the criterion derived in Section 3. The set is generated in the area $A \subset \mathbb{C}$ using the maximal number of iterations equal to K.

Algorithm 2 Viscosity Mandelbrot set generation

Input: $T_r(z) = z^n + mz + r$, where $m, r \in \mathbb{C}$ and $n \geq 2$; $A \subset \mathbb{C}$ – area in which we draw the set; K – the maximum number of iterations; $\alpha, \lambda \in (0, 1)$ – parameters for the viscosity approximation type iterative method; $g(z) = az + b$, where $a, b \in \mathbb{C}$ and $|a| < 1$; $colourmap[0..K]$ – colour map with $K + 1$ colours.
Output: Viscosity Mandelbrot set for area A.

```
for $r \in A$ do
    $R = \max\left\{|r|, |b|, \left(|m| + \frac{2+\alpha}{(1-\alpha)(1-\lambda)}\right)^{\frac{1}{n-1}}\right\}$
    $j = 0$
    $z_0 = r$
    while $|z_j| < R$ and $j < K$ do
        $y_j = \lambda z_j + (1-\lambda)T_r(z_j)$
        $z_{j+1} = \alpha g(z_j) + (1-\alpha)y_j$
        $j = j + 1$
    colour $r$ with $colourmap[j]$
```

To generate the graphical examples presented in this section, we implemented Algorithms 1 and 2 in Mathematica 12. The examples presented in this section were generated using the same colour map (Fig. 1), image resolution 800 × 800 pixels, and $K = 50$.

4.1 Examples of viscosity Julia sets

In the first example, we generated a quadratic viscosity Julia set using Algorithm 1. The parameters used to generate this set were the following: $n = 2$, $m = 1.92 + 2.09i$, $r = 0.6 - 1.02i$, $A = [-10, 5] \times [-9, 6]$, $K = 50$, $a = 0.3 + 0.6i$, $b = 0.07 + 0.5i$. In the example, we divided the images into two groups. In Fig. 2, we see the first group in which we fixed the value of λ to 0.33, and changed the value of α: (a) 0.25, (b) 0.5, (c) 0.65, (d) 0.75, (e) 0.8, and (f) 0.85. From the images, we see that the set becomes larger with the increase of the α parameter. Thus, the α parameter has a significant impact on the shape and size of the set. In Fig. 2(c) and (d), we can see some spiral structures but they disappear when we take higher values of α, see Fig. 2(e) and (f). Moreover, we can observe that the set has a 2-fold symmetry.

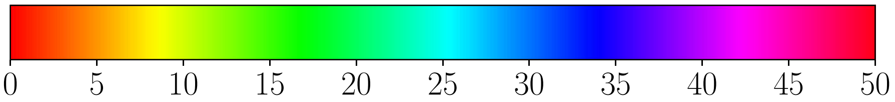
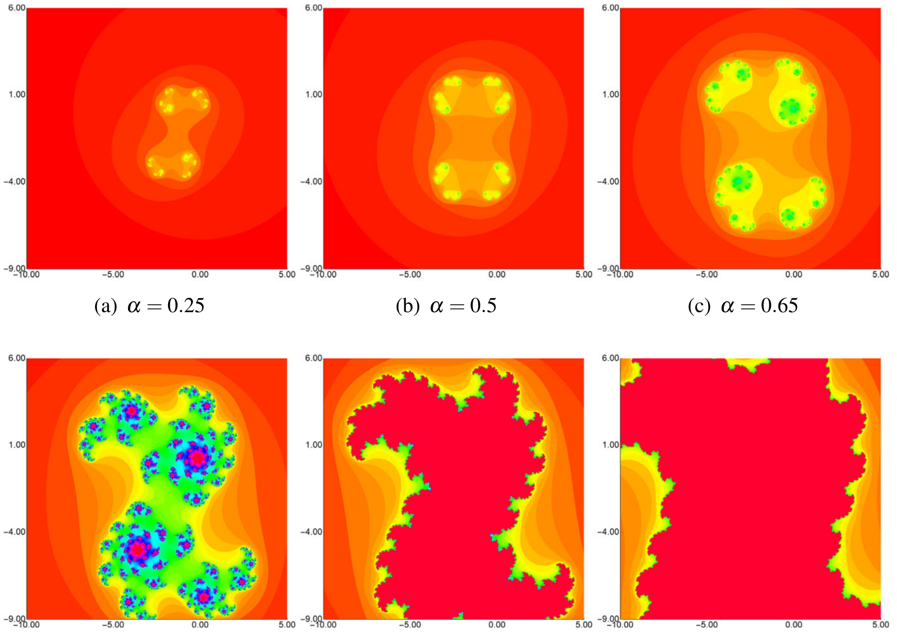


Fig. 1 Colour map used in the generation of viscosity Mandelbrot and Julia sets examples



(a) $\alpha = 0.25$ (b) $\alpha = 0.5$ (c) $\alpha = 0.65$

(d) $\alpha = 0.75$ (e) $\alpha = 0.8$ (f) $\alpha = 0.85$

Fig. 2 Julia set for $n = 2$ generated via the viscosity approximation type iteration with $\lambda = 0.33$ and varying α

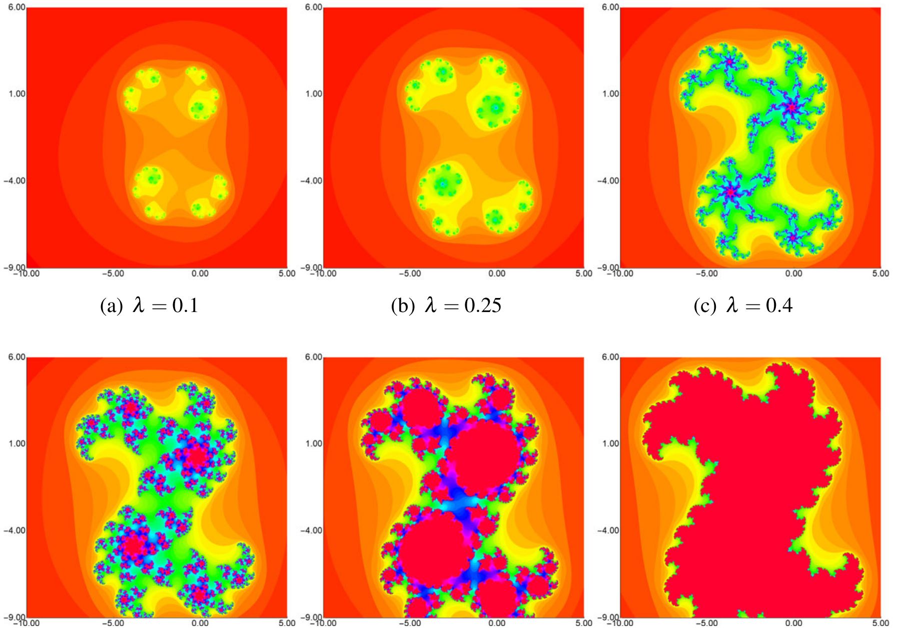
In the second group, presented in Fig. 3, we fixed α to 0.7, and changed the value of λ: (a) 0.1, (b) 0.25, (c) 0.4, (d) 0.45, (e) 0.5, and (f) 0.55. In this case, we also see that the λ parameter has a great impact on the shape and size of the set. Moreover, we can observe that for lower values of λ the shape change is smaller than in the case of higher λ values.

In the next example, we generated viscosity Julia set for the fifth degree polynomial. The parameters used to generate the set in this example were the following: $n = 5$, $m = 3.91 + i$, $r = 0.002 + 0.008i$, $A = [-2, 2]^2$, $K = 50$, $a = 0.35$, $b = 0.001$. Similar to the first example, we divided the images into two groups:

- Figure 4 in which we fixed $\lambda = 0.25$ and changed the value of α: (a) 0.3, (b) 0.4, (c) 0.55, (d) 0.65, (e) 0.75, and (f) 0.8,
- Figure 5 in which we fixed $\alpha = 0.6$ and changed the value of λ: (a) 0.2, (b) 0.35, (c) 0.5, (d) 0.65, (e) 0.67, and (f) 0.85.

From Figs. 4 and 5, we see that both the parameters (α, λ) have an impact on the set's shape, and even a slight change of the parameter's value can considerably change the set. Moreover, at first sight, it may seem that all sets have a 4-fold symmetry. However, when we look closely, then we notice that, for instance, the spiral arm in the top-right corner in Fig. 4(e) or the two upper spiral arms in Fig. 5(b) are different from the other arms, so these sets do not have a 4-fold symmetry.

To show the variety of viscosity Julia sets that can be obtained by the proposed method, in the last graphical example of viscosity Julia sets, we present some various



(a) $\lambda = 0.1$ (b) $\lambda = 0.25$ (c) $\lambda = 0.4$

(d) $\lambda = 0.45$ (e) $\lambda = 0.5$ (f) $\lambda = 0.55$

Fig. 3 Julia set for $n = 2$ generated via the viscosity approximation type iteration with $\alpha = 0.7$ and varying λ

Julia sets generated via the viscosity approximation type iteration (Fig. 6). The values of the parameters used to generate them were the following:

(a) $n = 2$, $m = 1.92 + 2.09i$, $r = 0.6 - 1.02i$, $A = [-10, 4] \times [-8, 6]$, $\alpha = 0.33$, $\lambda = 0.75$, $a = 0.03$, $b = 0.07 + 0.13i$,

(b) $n = 2$, $m = 0.0$, $r = -1.2 + 0.2i$, $A = [-2, 2]^2$, $\alpha = 0.07$, $\lambda = 0.09$, $a = 0.3$, $b = 0.9i$,

(c) $n = 3$, $m = 3.98 + 1.2i$, $r = 0.008 + 0.001i$, $A = [-3.5, 3.5]^2$, $\alpha = 0.7$, $\lambda = 0.4$, $a = 0.35$, $b = 0.001$,

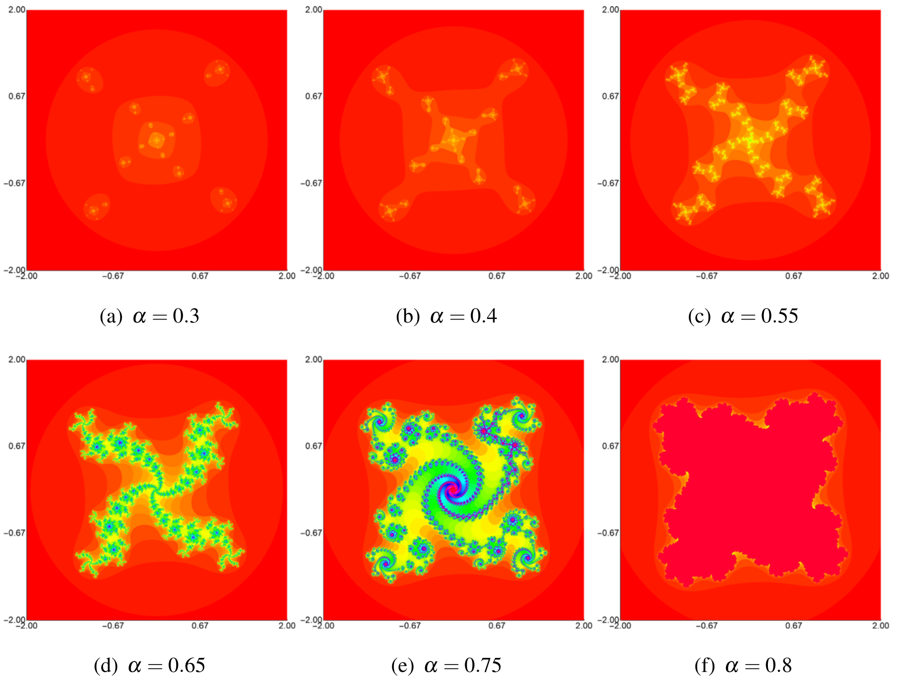
(d) $n = 3$, $m = -2.1 + 0.09i$, $r = 0.01 - 1.2i$, $A = [-1.9, 1.9]^2$, $\alpha = 0.05$, $\lambda = 0.34$, $a = 0.5$, $b = -0.5 + 0.04i$,

(e) $n = 5$, $m = 3.91 + 1.2i$, $r = 0.002 + 0.008i$, $A = [-2, 2]^2$, $\alpha = 0.69$, $\lambda = 0.4$, $a = 0.35$, $b = 0.001$,

(f) $n = 10$, $m = 3.91 + 1.2i$, $r = 0.008 + 0.002i$, $A = [-1.5, 1.5]^2$, $\alpha = 0.7$, $\lambda = 0.4$, $a = 0.35$, $b = 0.001$.

4.2 Examples of viscosity Mandelbrot sets

Now, let us see analogous examples of viscosity Mandelbrot set generated using Algorithm 2. We start with viscosity Mandelbrot set of quadratic polynomial. The



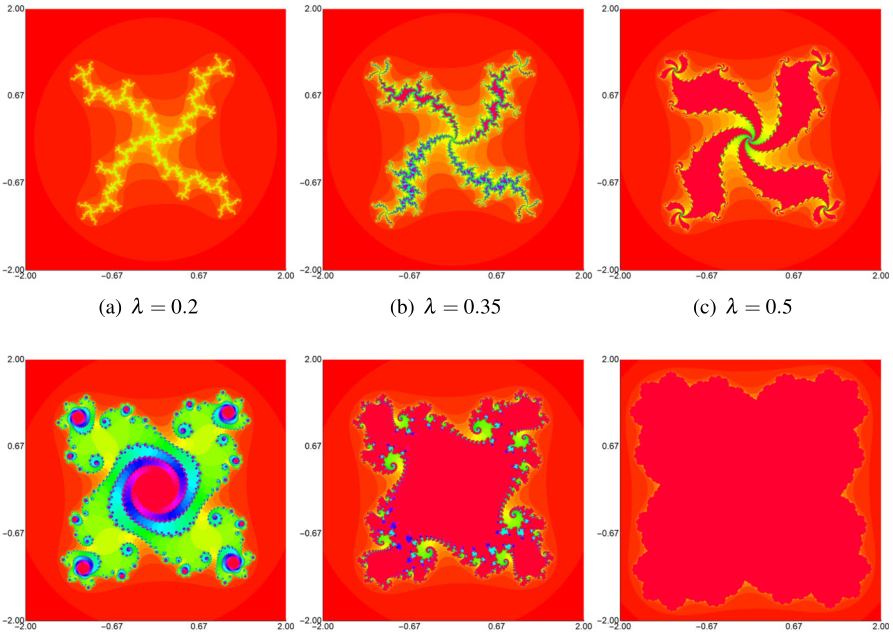
(a) $\alpha = 0.3$ (b) $\alpha = 0.4$ (c) $\alpha = 0.55$

(d) $\alpha = 0.65$ (e) $\alpha = 0.75$ (f) $\alpha = 0.8$

Fig. 4 Julia set for $n = 5$ generated via the viscosity approximation type iteration with $\lambda = 0.25$ and varying α

parameters used to generate this set were the following: $n = 2$, $m = 4.29 + 0.009i$, $K = 50$, $A = [-10, 1.5] \times [-5.75, 5.75]$, $a = 0.25$, $b = 0.09i$. Similar to the viscosity Julia set case, we divide the images into two groups. In Fig. 7, we see the first group in which we fixed the value of λ to 0.45, and changed the value of α: (a) 0.05, (b) 0.15, (c) 0.25, (d) 0.4, (e) 0.5, and (f) 0.7. The second group is presented in Fig. 8. Images were generated for a fixed value of α equal to 0.2, and varying the parameter λ: (a) 0.1, (b) 0.3, (c) 0.4, (d) 0.5, (e) 0.65, and (f) 0.8. From Figs. 7 and 8, we see that both parameters have a large impact on the shape of the set. The set grows and changes shape with increasing values of any of the two parameters. We can also notice that the resulting sets have axial symmetry.

In the second example, we consider a quartic polynomial and generate the viscosity Mandelbrot set with the following parameters: $n = 4$, $m = 4.9 + 0.9i$, $K = 50$, $A = [-3.5, 3.5]^2$, $a = 0.7 + 0.35i$, $b = 0.3 + 0.3i$. Figure 9 presents images obtained with a fixed value of λ equal to 0.85, and varying the parameter α: (a) 0.1, (b) 0.2, (c) 0.3, (d) 0.5, (e) 0.7, (f) 0.9, whereas Fig. 10 presents images generated for a fixed $\alpha = 0.6$, and varying λ: (a) 0.3, (b) 0.55, (c) 0.7, (d) 0.85, (e) 0.9, and (f) 0.95. In this example, we can also observe a high dependency between the parameters of the iteration and the shape of the resulting set. For low values of parameters, the sets are small and get bigger with the increase of the parameters. We can notice that the shape change is not uniform in the three arms. For higher values of the parameters,



(a) $\lambda = 0.2$ (b) $\lambda = 0.35$ (c) $\lambda = 0.5$

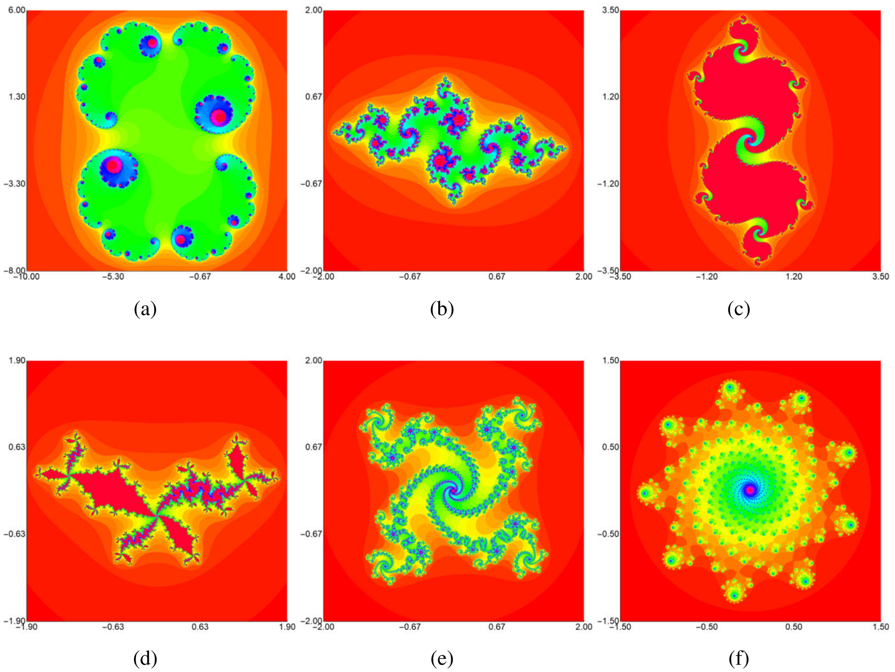
(d) $\lambda = 0.65$ (e) $\lambda = 0.67$ (f) $\lambda = 0.85$

Fig. 5 Julia set for $n = 5$ generated via the viscosity approximation type iteration with $\alpha = 0.6$ and varying λ

one of the arms starts dominating the shape. Moreover, we can observe that the sets do not have any symmetry.

In the last graphical example, we show some various viscosity Mandelbrot sets generated by the proposed method. The images obtained are presented in Fig. 11. The values of the parameters used to generate them were the following:

(a) $n = 2$, $m = 0.8 + 0.09i$, $A = [-3, 0.4] \times [-1.7, 1.7]$, $\alpha = 0.09$, $\lambda = 0.2$, $a = 0.38$, $b = 0.9 + 0.09i$,
(b) $n = 2$, $m = 0$, $A = [-2.3, 0.7] \times [-1.5, 1.5]$, $\alpha = 0.2$, $\lambda = 0.01$, $a = 0.05$, $b = 0.005$,
(c) $n = 4$, $m = 1.01 + 0.004i$, $A = [-3.2, 2.8] \times [-3, 3]$, $\alpha = 0.05$, $\lambda = 0.91$, $a = 0.05$, $b = 0.001 + 0.05i$,
(d) $n = 5$, $m = 4.69 + 0.003i$, $A = [-1.6, 1.6]^2$, $\alpha = 0.4$, $\lambda = 0.6$, $a = 0.05$, $b = 0.005$,
(e) $n = 5$, $m = 0.3$, $A = [-1.1, 1.1]^2$, $\alpha = 0.004$, $\lambda = 0.005$, $a = 0.008$, $b = 0.9 + 0.1i$,
(f) $n = 10$, $m = 4.79 + 0.003i$, $A = [-1.5, 1.5]^2$, $\alpha = 0.63$, $\lambda = 0.53$, $a = 0.05$, $b = 0.005$.



(a) (b) (c)

(d) (e) (f)

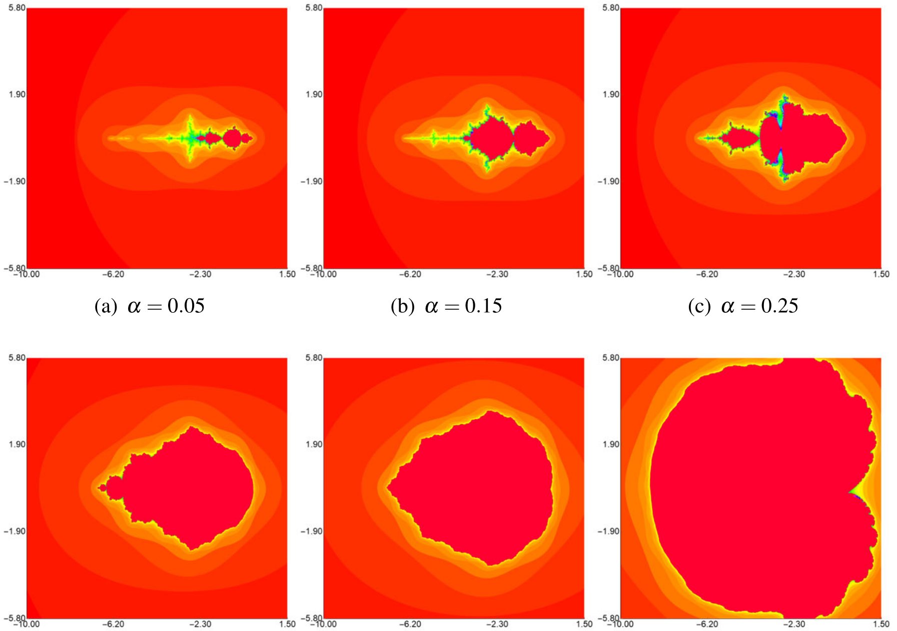
Fig. 6 Examples of various viscosity Julia sets

5 Dependency between the iteration parameters and new numerical measures

The graphical examples presented in Figs. 2–5 and 7–10 showed that the shape of the viscosity Mandelbrot and Julia sets is highly dependent on the parameters used in the considered iteration method. Therefore, we may ask whether this dependency is linear or non-linear? In the literature regarding the use of various iteration processes in the generation of Mandelbrot and Julia sets (see, for example, [3, 6, 26–29, 45, 46, 48, 55]), there are no methods that allow to study this dependency. The authors of those papers showed only some images of the sets obtained with the derived escape criterion and did not study – in any way – this dependency. In this section, we will propose two numerical measures that will allow studying – to some extent – the impact of the iteration used in the generation algorithm on the obtained Mandelbrot and Julia sets. The proposed measures can be used to study any of the iteration methods introduced in the literature.

Assume that we generate an image of viscosity Mandelbrot or viscosity Julia set in the area $A \subset \mathbb{C}$ using the maximal number of iterations equal to K. The first introduced measure will refer to the escaping points. We call it the *average escape time* (AET) E_{ave}, and define it as follows. Let

$$E = \{z \in A : I(z) < K\}, \tag{17}$$



(a) $\alpha = 0.05$ (b) $\alpha = 0.15$ (c) $\alpha = 0.25$

(d) $\alpha = 0.4$ (e) $\alpha = 0.5$ (f) $\alpha = 0.7$

Fig. 7 Mandelbrot set for $n = 2$ generated via the viscosity approximation type iteration with $\lambda = 0.45$ and varying α

where $I(z)$ is the number of performed iterations with the starting point z. Thus, E is the set of escaping points in A. Now, if $E \neq \emptyset$, then

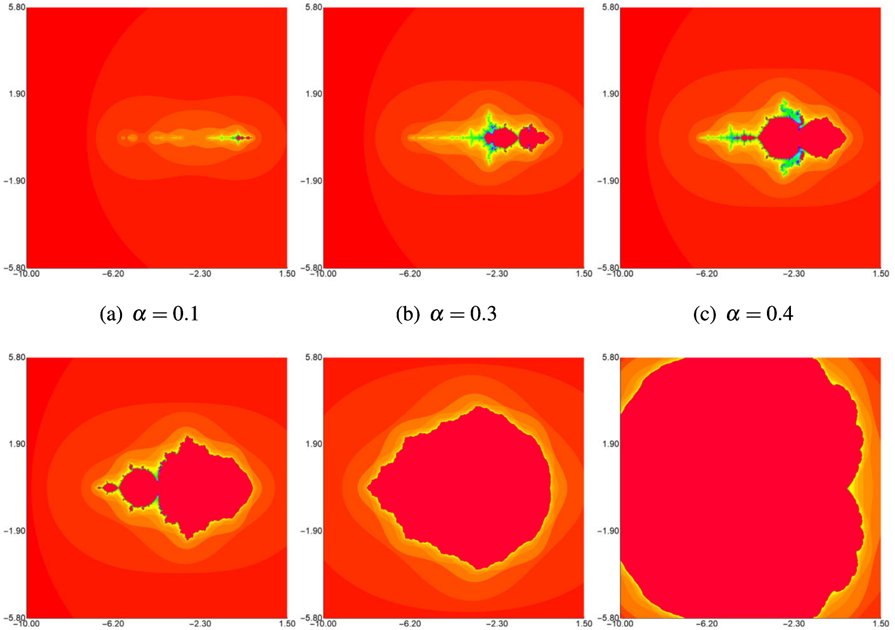
$$E_{ave} = \frac{1}{|E|} \sum_{z \in E} I(z), \tag{18}$$

else E_{ave} is undefined. The second measure refers to the non-escaping points, i.e., points z for which $I(z) = K$. We define it as

$$P = \frac{|A \setminus E|}{|A|}, \tag{19}$$

and call it the *non-escaping area index* (NAI). From the definitions, we see that $E_{ave} \in [0, K)$ and $P \in [0, 1]$. The average escape time, as the name suggests, tells us how fast, in the considered area, the escaping points escape to infinity on average. The non-escaping area index tells us about the percentage of the considered area covered by the non-escaping points, giving us information about a relative set size in A.

Using the average escape time and non-escaping area index, we can study the dependency of these measures on the iteration parameters. We do so by generating Mandelbrot or Julia set images in the same area using various values of the parameters, calculating the measures, and then plotting the data. Let us see how such plots look



(a) $\alpha = 0.1$ (b) $\alpha = 0.3$ (c) $\alpha = 0.4$

(d) $\alpha = 0.5$ (e) $\alpha = 0.65$ (f) $\alpha = 0.8$

Fig. 8 Mandelbrot set for $n = 2$ generated via the viscosity approximation type iteration with $\alpha = 0.2$ and varying λ

for the viscosity approximation iteration that is considered in the paper. Except for the plots of AET and NAI, we will also present plots showing the generation times.

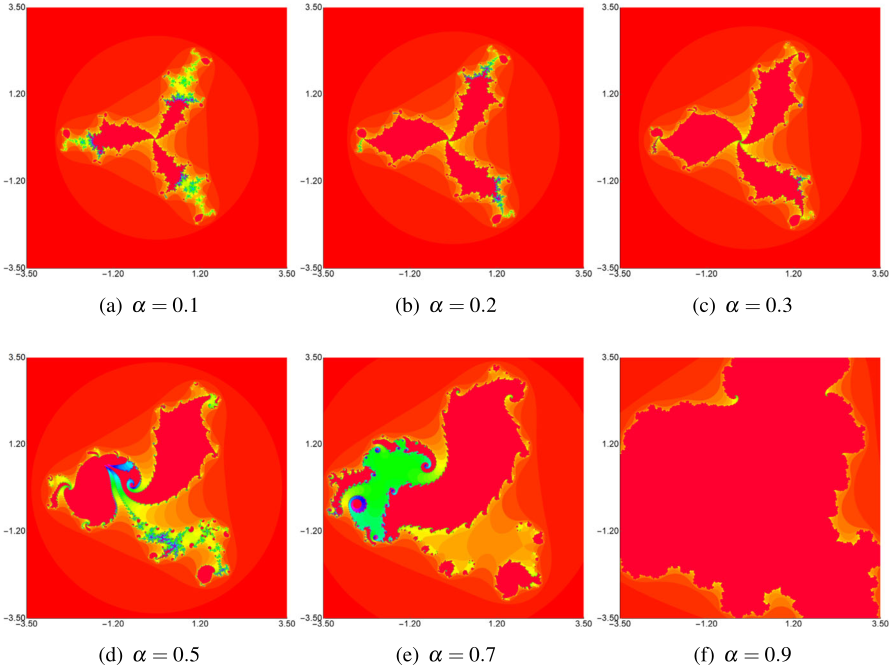
For all the presented plots, we will use $K = 50$, the area A will be divided into grid of 800 × 800 points, and for the iteration parameters (α, λ) we will use 99 values in $(0, 1)$, giving the value step equal to 0.01 and $99^2 = 9801$ generated images for a single plot. The numerical experiments were performed on a computer with the following specifications: Intel i5-9600K (@3.70 GHz), 32 GB DDR4 RAM, and Windows 10 (64 bit). The algorithms for generating the sets (Algorithms 1 and 2) were implemented in Mathematica 12 using the parallelization option of the `Compile` command.

5.1 Viscosity Julia sets

In the first example, we will present plots of AET and NAI for a quadratic viscosity Julia set generated with $n = 2$, $m = 1.92+2.09i$, $r = 0.6-1.02i$, $A = [-10, 5] \times [-9, 6]$, and two contractive mappings:

- $g_1(z) = 0.03z + 0.07 + 0.13i$,
- $g_2(z) = (0.3 + 0.6i)z + 0.07 + 0.5i$.

We saw examples of viscosity Julia sets obtained with g_2 in Figs. 2 and 3. In Figs. 12 and 13, we present the plots of AET, NAI and generation times for g_1 and g_2,

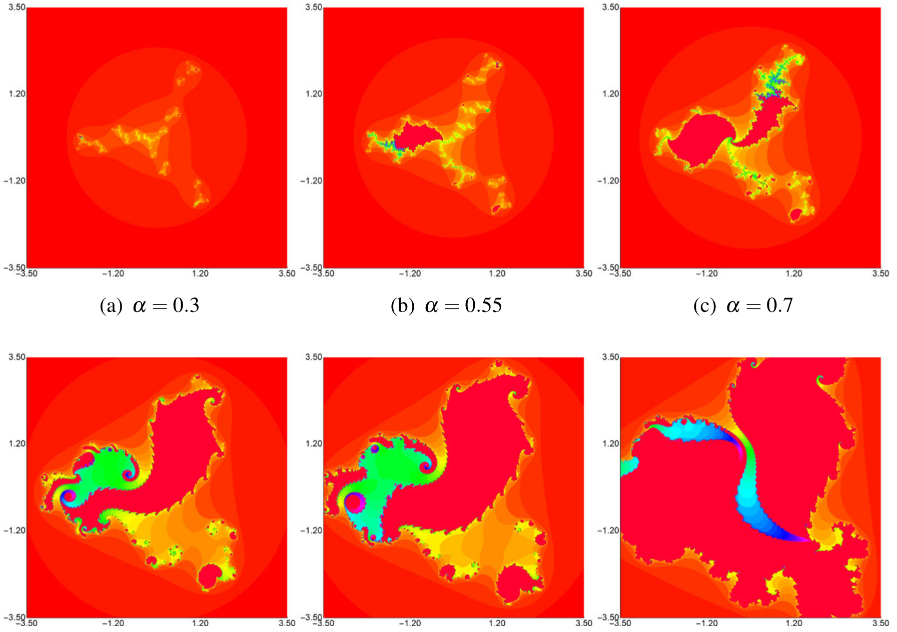


(a) $\alpha = 0.1$ (b) $\alpha = 0.2$ (c) $\alpha = 0.3$

(d) $\alpha = 0.5$ (e) $\alpha = 0.7$ (f) $\alpha = 0.9$

Fig. 9 Mandelbrot set for $n = 4$ generated via the viscosity approximation type iteration with $\lambda = 0.85$ and varying α

respectively. The white areas in the AET plots indicate areas in which we were not able to calculate the measure, i.e., all points in the considered area are non-escaping ones. When we look at the AET plots, we see that the overall shape is very similar. They only differ in some details, and the white area in case of g_2 is smaller. The values of this measure vary greatly from 0.409 (attained at $\alpha = 0.01$, $\lambda = 0.01$) to 46.961 (attained at $\alpha = 0.05$, $\lambda = 0.99$) for g_1, and from 0.409 (attained at $\alpha = 0.01$, $\lambda = 0.01$) to 45.727 (attained at $\alpha = 0.08$, $\lambda = 0.99$) for g_2. However, the number of points with a high value of AET is small compared to the number of points with low values. Now, by looking at the NAI plots, we see that the dependency is non-linear, and the overall shape is correlated with the shape of the AET plot. Moreover, from the plots, we can observe that the filled viscosity Julia sets in the considered area grow with the increase of both the parameters. For high values of α and λ the filled viscosity Julia set occupies the whole area, for example, for $\alpha = 0.06$, $\lambda = 0.99$ for g_1, and $\alpha = 0.09$, $\lambda = 0.99$ for g_2 the NAI measure equals to 1.0. From the time plots in Figs. 12(c) and 13(c), we see that the times are highly correlated with the NAI measure. The only difference between these two plots is that the time plots look noisy. This is due to the fact that the operating system services and processes are working in the background and can take away some CPU time, depending on the needs of the operating system. The minimal times in both considered cases are obtained for low values of the parameters: 0.397 s for $\alpha = 0.02$, $\lambda = 0.18$ in the case of Fig. 12(c), and 0.404 s for $\alpha = 0.02$, $\lambda = 0.08$ in the case of Fig. 13(c). On the other hand, the longest times were obtained for high



(a) $\alpha = 0.3$ (b) $\alpha = 0.55$ (c) $\alpha = 0.7$

(d) $\alpha = 0.85$ (e) $\alpha = 0.9$ (f) $\alpha = 0.95$

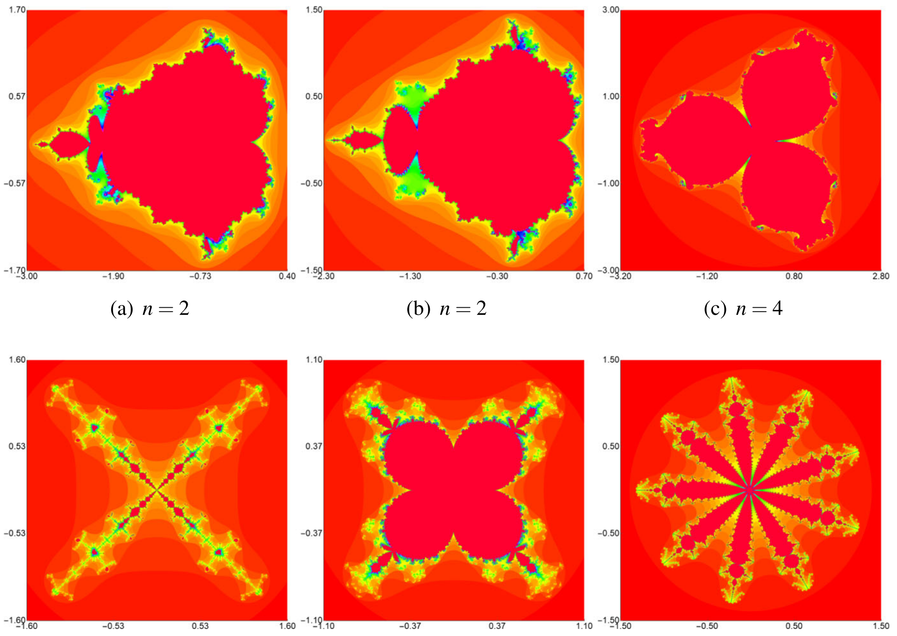
Fig. 10 Mandelbrot set for $n = 4$ generated via the viscosity approximation type iteration with $\alpha = 0.6$ and varying λ

values of the parameters: 3.376 s for $\alpha = 0.8$, $\lambda = 0.58$ in Fig. 12(c), and 3.149 s for $\alpha = 0.65$, $\lambda = 0.93$ in Fig. 13(c).

In the next example, we will present similar plots, but this time for viscosity Julia sets with the following parameters $n = 5$, $m = 3.91 + i$, $r = 0.002 + 0.008i$, $A = [-2, 2]^2$, and two contractive mappings:

- $g_3(z) = 0.35z + 0.001$,
- $g_4(z) = (0.7 + 0.35i)z + 0.3 + 0.3i$.

Examples of viscosity Julia sets generated with the g_3 mapping are presented in Figs. 4 and 5. The plots with the numeric measures for the two contraction mappings are presented in Figs. 14 and 15. From the AET plots, we see that the average speed of escaping is fast, and only a small number of points – in the parameter space – obtain higher values of AET. Moreover, we can observe that the dependency is non-linear, and it can be different for various contraction mappings used in the iteration process, i.e., in Fig. 14(a) the dependency can be seen as regular, whereas in Fig. 15(a) we see irregular areas. The minimal values of AET in both plots were attained at $\alpha = 0.01$, $\lambda = 0.01$, and they were equal to 0.525. The maximal values of the measure are different for both plots. In plot from Fig. 14(a) the maximal value was equal 25.67 and it was attained at $\alpha = 0.15$, $\lambda = 0.96$, whereas for plot from Fig. 15(a) the maximal value 34.77 was attained at $\alpha = 0.21$, $\lambda = 0.99$. Now, by looking at the NAI plots in Figs. 14(b) and 15(b), we see that in the lower-left corner, there is a large uniform area in which NAI equals to 0.0, so for the parameters' values in this area there were no

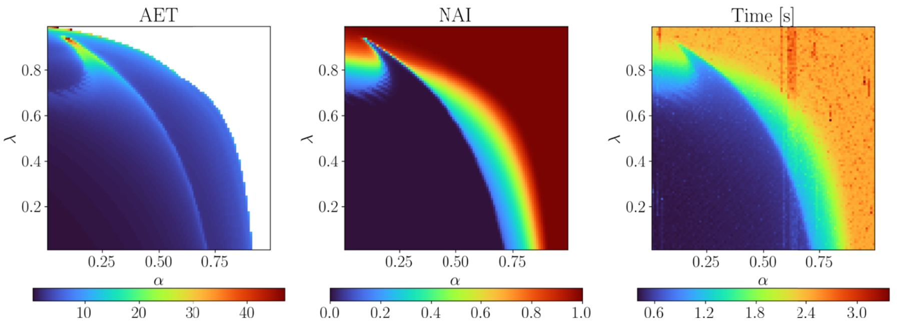


(a) $n = 2$ (b) $n = 2$ (c) $n = 4$

(d) $n = 5$ (e) $n = 5$ (f) $n = 10$

Fig. 11 Examples of various viscosity Mandelbrot sets

non-escaping points. In the opposite corner (upper-right), we see points in which NAI equals to 1.0, thus the whole considered area is covered by the non-escaping points meaning that the filled viscosity Julia set occupies this area. In the rest of the parameters space, we see that we obtain various values of NAI. Additionally, they change in a non-linear way, and the change is dependent on the contraction mapping used. Like in the case of the quadratic viscosity Julia sets, the time plots in Figs. 14(c) and 15(c) are correlated with the plots of the NAI numerical measure and the plots are noisy



(a) (b) (c)

Fig. 12 AET, NAI, and time (in seconds) plots for viscosity Julia set for $n = 2$ and $g_1(z) = 0.03z + (0.07 + 0.13i)$

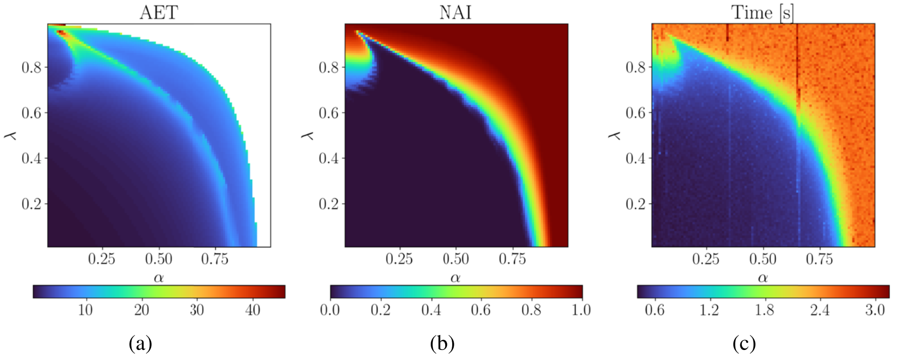


Fig. 13 AET, NAI, and time (in seconds) plots for viscosity Julia set for $n = 2$ and $g_2(z) = (0.3+0.6i)z+(0.07+0.5i)$

due to the processes and services that are running in the background. From the plots, we see that the shortest times are obtained for low values of the parameters, whereas the longest times for high values of the parameters. More precisely, the minimal times 0.399 s (Fig. 14(c)) and 0.401 s (Fig. 15(c)) were attained at $\alpha = 0.01$, $\lambda = 0.19$ and $\alpha = 0.01$, $\lambda = 0.22$, respectively. The maximal times 3.161 s (Fig. 14(c)) and 2.935 s (Fig. 15(c)) were attained at $\alpha = 0.78$, $\lambda = 0.99$ and $\alpha = 0.9$, $\lambda = 0.96$, respectively.

5.2 Viscosity Mandelbrot sets

In the first example with the viscosity Mandelbrot set, we will present plots of AET and NAI for a quadratic viscosity Mandelbrot set generated with the following parameters: $n = 2$, $m = 4.29 + 0.009i$, $A = [-10, 1.5] \times [-5.75, 5.75]$, and two contractive mappings:

- $g_5(z) = 0.25z + 0.09$,
- $g_6(z) = (0.5 - 0.3i)z + 0.2 + 0.7i$.

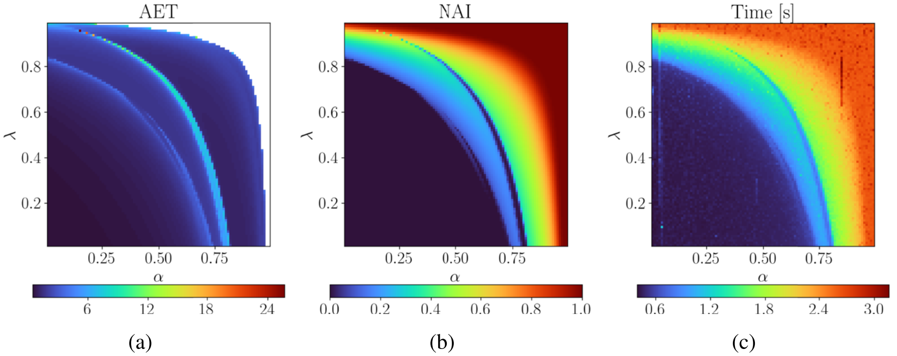


Fig. 14 AET, NAI, and time (in seconds) plots for viscosity Julia set for $n = 5$ and $g_3(z) = 0.35z + 0.001$

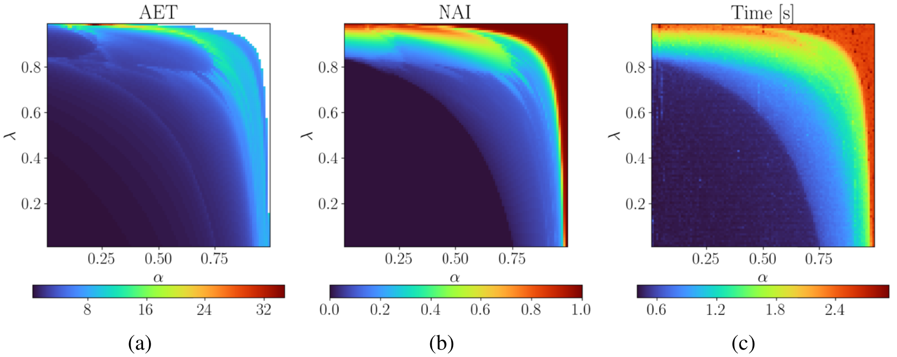


Fig. 15 AET, NAI, and time (in seconds) plots for viscosity Julia set for $n = 5$ and $g_4(z) = (0.7+0.35i)z+(0.3+0.3i)$

The obtained plots are presented in Figs. 16 and 17. For the AET plots, we see very similar behaviour. The lowest values of this measure are attained for low values of both parameters—attaining the minimal value of 0.691 at $\alpha = \lambda = 0.01$, in both cases. Moreover, we see that the areas with high AET values depend on the contraction mapping. For g_5 the area is very narrow, with the maximal value equal to 48.667 attained at $\alpha = 0.42$, $\lambda = 0.91$, whereas for g_6 the area is wider and the variation of the values is bigger, attaining the maximum (49.0) at $\alpha = 0.2$, $\lambda = 0.98$. When we look at the NAI plots, we can observe that for g_5, there is a large area in which NAI is equal to 1.0. For g_5 in a similar area, we see a much smaller area with NAI equal to 1.0, but in the remaining area, the values are high, i.e., about 0.8 and higher. This means that the size change of the viscosity Mandelbrot set for g_6 was more diverse than for the g_5 mapping. In both cases, we clearly see that the size of the viscosity Mandelbrot set is a non-linear function of the two parameters (α, λ). When we look at the time plots, we can observe a similar situation as in the case of the viscosity Julia sets. Namely, the time plots are correlated with the NAI plots, and they are noisy for the same reasons. The shortest times for both the contraction mappings are attained at

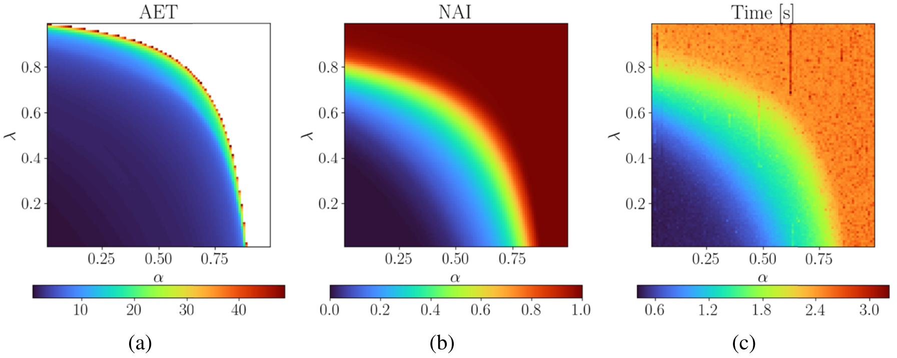
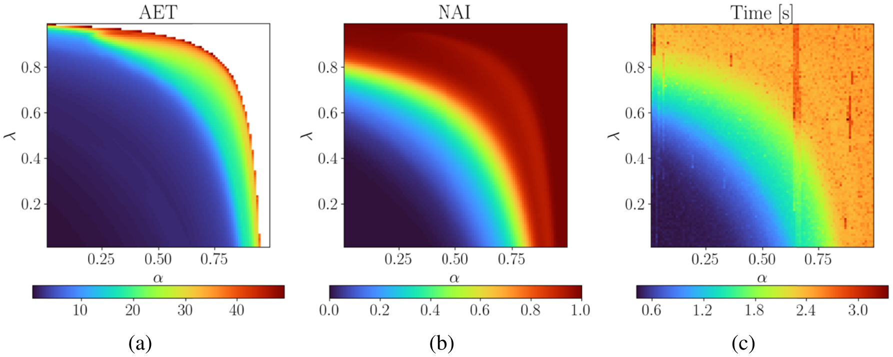


Fig. 16 AET, NAI, and time (in seconds) plots for viscosity Mandelbrot set for $n = 2$ and $g_5(z) = 0.25z + 0.09i$



(a) (b) (c)

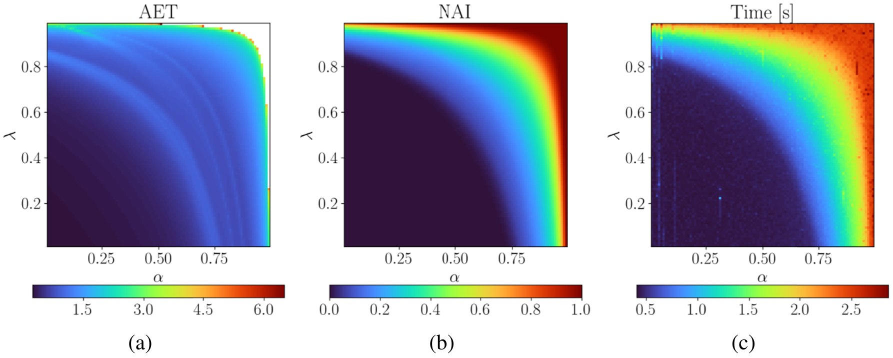
Fig. 17 AET, NAI, and time (in seconds) plots for viscosity Mandelbrot set for $n = 2$ and $g_6(z) = (0.5 - 0.3i)z + (0.2 + 0.7i)$

low values of the parameters: 0.397 s at $\alpha = 0.01$, $\lambda = 0.1$ and 0.421 s at $\alpha = 0.08$, $\lambda = 0.15$ for g_5 and g_6, respectively. The longest times, on the other hand, are attained for high values of the parameters: 3.216 s at $\alpha = 0.62$, $\lambda = 0.69$ and 3.344 s at $\alpha = 0.88$, $\lambda = 0.57$ for g_5 and g_6, respectively.

In the last example, we present plots for the quartic viscosity Mandelbrot set with the following parameters: $n = 4$, $m = 4.9 + 0.9i$, $A = [-3.5, 3.5]^2$, and two contractive mappings:

- $g_7(z) = 0.25z + 0.001$,
- $g_8(z) = (0.7 + 0.35i)z + 0.3 + 0.3i$.

The obtained AET, NAI and time plots are presented in Figs. 18 and 19. For the quartic viscosity Mandelbrot set, the overall behaviour is very similar to the quadratic case. When looking at the AET plots, we see that for low values of the parameters (α, λ), the escaping speed is fast, attaining the minimal value of 0.244 at $\alpha = \lambda = 0.01$ in both cases. The speed changes in a non-linear way depending on the parameters. The slowest escape speed for g_7 is visible at the boundary, where the maximum value of 6.5 is attained at $\alpha = 0.51$, $\lambda = 0.99$. In the case of the g_8 mapping, we see that the



(a) (b) (c)

Fig. 18 AET, NAI, and time (in seconds) plots for viscosity Mandelbrot set for $n = 4$ and $g_7(z) = 0.25z + 0.001$

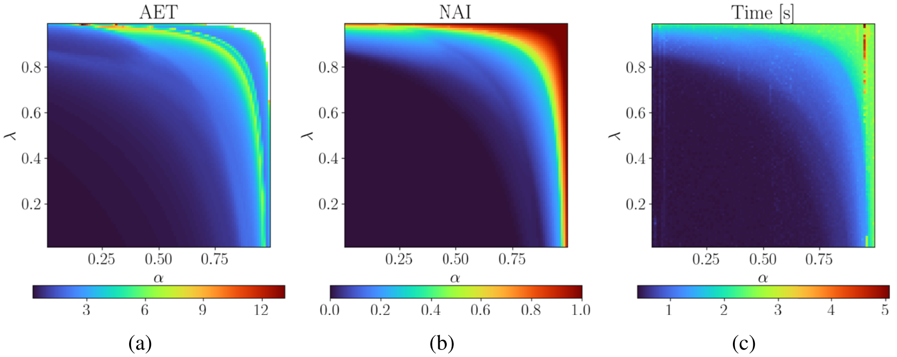


Fig. 19 AET, NAI, and time (in seconds) plots for viscosity Mandelbrot set for $n = 4$ and $g_8(z) = (0.7 + 0.35i)z + (0.3 + 0.3i)$

high values of AET are obtained at the boundary, but also, there is some strip with high values inside the parameters space. The maximal value of 13.154 is attained in this strip at $\alpha = 0.16$, $\lambda = 0.99$. When looking at the NAI plots, there is a similar behaviour in both cases, i.e., the change of viscosity Mandelbrot set size. In the lower-left corner of the parameters space, there are no non-escaping points, so there is no visible viscosity Mandelbrot set for these parameters. However, in the opposite corner (upper-right), we see points in the parameter space for which the set occupies the considered area. In the rest of the parameters space, we can observe a viscosity Mandelbrot set of varying size. Therefore, from the NAI plots, we clearly see that the parameters of the viscosity iteration change the size of the Mandelbrot set in a non-linear way. The generation times for the considered quartic viscosity Mandelbrot set behave similarly to all the other examples. Namely, they are highly correlated with the NAI numerical measure. Therefore, the shortest times are obtained for low values of the parameters (0.408 s at $\alpha = 0.08$, $\lambda = 0.27$ in Fig. 18(c), and 0.391 s at $\alpha = 0.01$, $\lambda = 0.14$ in Fig. 19(c)), whereas the longest times for high values of α and λ (2.855 s at $\alpha = 0.95$, $\lambda = 0.95$ in Fig. 18(c), and 5.065 s at $\alpha = 0.95$, $\lambda = 0.99$ in Fig. 19(c)).

6 Conclusions

In this paper, we have analysed the viscosity approximation type method in the generation of Julia and Mandelbrot sets. For this, we have established some convergence conditions to visualise Julia and Mandelbrot sets via viscosity approximation type orbit using the adapted versions of the escape-time algorithms. Various artistic patterns of fascinating viscosity Julia and Mandelbrot sets with different shapes and colours have been generated using different complex polynomials, contraction mappings and parameters α, λ. We have examined that a very small change in any input parameter can change the image drastically. Moreover, we proposed two numerical measures that allow the study of the dependence of the change in set shape on the values of the iteration parameters. Using these two measures, we showed that the dependency for the considered iteration method is highly non-linear.

As fractals have attracted people's attention in the field of design, we believe that the results of this research can be interesting for those whose works are related to the automatic creation of nicely looking graphics and designing printing patterns. In our future work, we will extend the result from the paper by using other viscosity approximation type orbits. Moreover, we will study the set shape change for other iterations introduced in the literature using the proposed numerical measures. Another interesting direction for future work is the use of various iteration methods for Mandelbrot and Julia sets generated using the trinition numbers introduced in [4].

Acknowledgements Not applicable.

Author contribution Conceptualization: S.K., K.G., A.N.; formal analysis: S.K., K.G.; investigation: S.K., K.G., A.N.; validation: S.K., K.G., A.N.; software: S.K., K.G., A.N., N.K.; visualisation: S.K., K.G., A.N., N.K.; data curation: K.G.; writing—original draft: K.G.; writing—review and editing: S.K., K.G., A.N.; supervision: K.G., R.C.

Data availability The data and code that support the findings of this study are available from the corresponding author, upon reasonable request.

Declarations

Ethical approval Not applicable.

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Agarwal, S.: Cryptographic key generation using burning ship fractal. In: Proceedings of the 2nd International Conference on Vision, Image and Signal Processing, ICVISP 2018, p. Article 51. Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3271553.3271577
2. Al Sideiri, A., Alzeidi, N., Al Hammoshi, M., Chauhan, M., Al Farsi, G.: CUDA implementation of fractal image compression. J. Real Time Image Process. **17**(5), 1375–1387 (2020). https://doi.org/10.1007/s11554-019-00894-7
3. Ashish, Rani, M., Chugn, R.: Julia sets and Mandelbrot sets in Noor orbit. Appl. Math. Comput. **228**, 615–631 (2014). https://doi.org/10.1016/j.amc.2013.11.077
4. Atangana, A., Mekkaoui, T.: Trinition the complex number with two imaginary parts: fractal, chaos and fractional calculus. Chaos Solit. Fractals. **128**, 366–381 (2019). https://doi.org/10.1016/j.chaos.2019.08.018
5. Beardon, A.: Iteration of rational functions: complex analytic dynamical systems. Springer-Verlag, New York (1991)
6. Chauhan, Y., Rana, R., Negi, A.: New Julia sets of Ishikawa iterates. Int. J. Comput. Appl. **7**(13), 34–42 (2010)

7. Costanzo, S., Venneri, F.: Polarization-insensitive fractal metamaterial surface for energy harvesting in IoT applications. Electronics **9**(6), 959 (2020). https://doi.org/10.3390/electronics9060959
8. Dang, Y., Kauffman, L., Sandin, D.: Hypercomplex iterations: distance estimation and higher dimensional fractals. World Scientific, Singapore (2002)
9. Devaney, R.: A first course in chaotic dynamical systems: theory and experiment, 2nd edn. CRC Press, Boca Raton (2020)
10. Di Ieva, A., Grizzi, F., Jelinek, H., Pellionisz, A., Losa, G.: Fractals in the neurosciences, part I: general principles and basic neurosciences. Neuroscientist **20**(4), 403–417 (2014). https://doi.org/10.1177/1073858413513927
11. Fatou, P.: Sur les substitutions rationnelles. C. R. Acad. Sci. Paris **164**, 806–808 (1917)
12. Gdawiec, K.: Inversion fractals and iteration processes in the generation of aesthetic patterns. Comput. Graph. Forum. **36**(1), 35–45 (2017). https://doi.org/10.1111/cgf.12783
13. Gdawiec, K., Domańska, D.: Partitioned iterated function systems with division and a fractal dependence graph in recognition of 2D shapes. Int. J. Appl. Math. Comput. Sci. **21**(4), 757–767 (2011). https://doi.org/10.2478/v10006-011-0060-8
14. Genel, A., Lindenstrauss, J.: An example concerning fixed points. Isr. J. Math. **22**(1), 81–86 (1975). https://doi.org/10.1007/BF02757276
15. Guran, L., Shabbir, K., Ahmad, K., Bota, M.F.: Stability, data dependence, and convergence results with computational engendering of fractals via Jungck-DK iterative scheme. Fractal Fract. **7**(6), 418 (2023). https://doi.org/10.3390/fractalfract7060418
16. Halpern, B.: Fixed points of nonexpanding maps. Bull. Am. Math. Soc. **73**, 957–961 (1967). https://doi.org/10.1090/S0002-9904-1967-11864-0
17. Husain, A., Nanda, M., Chowdary, M., Sajid, M.: Fractals: an eclectic survey, part-I. Fractal Fract. **6**(2), 89 (2022). https://doi.org/10.3390/fractalfract6020089
18. Hussain, N., Nandal, A., Kumar, V., Chugh, R.: Multistep generalized viscosity iterative algorithm for solving convex feasibility problems in Banach spaces. J. Nonlinear Convex Anal. **21**(3), 587–603 (2020)
19. Jolaoso, L., Khan, S.: Some escape time results for general complex polynomials and biomorphs generation by a new iteration process. Mathematics **8**(12), 2172 (2020). https://doi.org/10.3390/math8122172
20. Jovanovic, R., Tuba, M.: A visual analysis of calculations-paths of the Mandelbrot set. Wseas Trans. Comput. **8**(7), 1205–1214 (2009)
21. Julia, G.: Mémoire sur l'itération des fonctions rationnelles. J. Math. Pures Appl. **8**(1), 47–246 (1918)
22. Keen, L., Kotus, J.: Dynamics of the family $\lambda \tan z$. Conform. Geom. Dyn. **1**, 28–57 (1997). https://doi.org/10.1090/S1088-4173-97-00017-9
23. Krzysztofik, W.: Fractals in antennas and metamaterials applications. In: F. Brambila (ed.) Fractal Analysis – Applications in Physics, Engineering and Technology, pp. 953–978. IntechOpen (2017). https://doi.org/10.5772/intechopen.68188
24. Kumari, S., Gdawiec, K., Nandal, A., Kumar, N., Chugh, R.: An application of viscosity approximation type iterative method in the generation of Mandelbrot and Julia fractals. Aequ. Math. **97**(2), 257–278 (2023). https://doi.org/10.1007/s00010-022-00908-z
25. Kumari, S., Gdawiec, K., Nandal, A., Postolache, M., Chugh, R.: A novel approach to generate Mandelbrot sets, Julia sets and biomorphs via viscosity approximation method. Chaos Solit. Fractals. **163**, 112540 (2022). https://doi.org/10.1016/j.chaos.2022.112540
26. Kumari, S., Kumari, M., Chugh, R.: Generation of new fractals via SP orbit with s-convexity. Int. J. Eng. Technol. **9**(3), 2491–2504 (2017). https://doi.org/10.21817/IJET/2017/V9I3/1709030282
27. Kumari, S., Kumari, M., Chugh, R.: Dynamics of superior fractals via Jungck SP orbit with s-convexity. Ann. Univ. Craiova Math. Comput. Sci. Ser. **46**(2), 344–365 (2019)
28. Kumari, S., Kumari, M., Chugh, R.: Graphics for complex polynomials in Jungck-SP orbit. IAENG Int. J. Appl. Math. **49**(4), 568–576 (2019)
29. Kwun, Y., Tanveer, M., Nazeer, W., Gdawiec, K., Kang, S.: Mandelbrot and Julia sets via Jungck-CR iteration with s-convexity. IEEE Access **7**, 12167–12176 (2019). https://doi.org/10.1109/ACCESS.2019.2892013
30. Liu, S., Pan, Z., Fu, W., Cheng, X.: Fractal generation method based on asymptote family of generalized Mandelbrot set and its application. J. Nonlinear Sci. Appl. **10**(3), 1148–1161 (2017). https://doi.org/10.22436/jnsa.010.03.24
31. Maingé, P.: The viscosity approximation process for quasi-nonexpansive mappings in Hilbert spaces. Comput. Math. with Appl. **59**(1), 74–79 (2010). https://doi.org/10.1016/j.camwa.2009.09.003

32. Mann, W.: Mean value methods in iteration. Proc. Am. Math. Soc. **4**(3), 506–510 (1953). https://doi.org/10.1090/S0002-9939-1953-0054846-3
33. Martinez, F., Manriquez, H., Ojeda, A., Olea, G.: Organization patterns of complex river networks in Chile: a fractal morphology. Mathematics **10**(11), 1806 (2022). https://doi.org/10.3390/math10111806
34. McMullen, C.: Area and Hausdorff dimension of Julia sets of entire functions. Trans. Am. Math. Soc. **300**, 329–342 (1987). https://doi.org/10.1090/S0002-9947-1987-0871679-3
35. Mork, L., Ulness, D.: Visualization of Mandelbrot and Julia sets of Möbius transformations. Fractal Fract. **5**(3), 73 (2021). https://doi.org/10.3390/fractalfract5030073
36. Moudafi, A.: Viscosity approximation methods for fixed-points problems. J. Math. Anal. Appl. **241**(1), 46–55 (2000). https://doi.org/10.1006/jmaa.1999.6615
37. Nandal, A., Chugh, R.: On zeros of accretive operators with application to the convex feasibility problem. Sci. Bull. Ser. A: Appl. Math. Phys. **81**(3), 95–106 (2019)
38. Nandal, A., Chugh, R., Kumari, S.: Convergence analysis of algorithms for variational inequalities involving strictly pseudo-contractive operators. Poincare J. Anal. Appl. **2019**(2), 123–136 (2019). https://doi.org/10.46753/pjaa.2019.v06i02.006
39. Nandal, A., Chugh, R., Postolache, M.: Iteration process for fixed point problems and zeros of maximal monotone operators. Symmetry **11**(5), 655 (2019). https://doi.org/10.3390/sym11050655
40. Ouyang, P., Chung, K., Nicolas, A., Gdawiec, K.: Self-similar fractal drawings inspired by M.C. Escher's print Square Limit. ACM Trans. Graph. **40**(3), 31 (2021)
41. Özgür, N., Antal, S., Tomar, A.: Julia and Mandelbrot sets of transcendental function via Fibonacci–Mann iteration. J. Funct. Spaces **2022**, Article ID 2592573 (2022). https://doi.org/10.1155/2022/2592573
42. Parise, P., Rochon, D.: A study of dynamics of the tricomplex polynomial $\eta^h + c$. Nonlinear Dyn. **82**(1–2), 157–171 (2015). https://doi.org/10.1007/s11071-015-2146-6
43. Postolache, M., Nandal, A., Chugh, R.: Strong convergence of a new generalized viscosity implicit rule and some applications in Hilbert space. Mathematics **7**(9), 773 (2019). https://doi.org/10.3390/math7090773
44. Prasad, B., Katiyar, B.: Fractals via Ishikawa iteration. In: P. Balasubramaniam (ed.) Control, Computation and Information Systems, Communications in Computer and Information Science, vol. 140, pp. 197–203. Springer (2011). https://doi.org/10.1007/978-3-642-19263-0_24
45. Rani, M., Kumar, V.: Superior Julia sets. J. Korean Soc. Math. Educ. Ser. D: Res. Math. Educ. **8**(4), 261–277 (2004)
46. Rani, M., Kumar, V.: Superior Mandelbrot set. J. Korean Soc. Math. Educ. Ser. D: Res. Math. Educ. **8**(4), 279–291 (2004)
47. Reich, S.: Strong convergence theorems for resolvents of accretive operators in Banach spaces. J. Math. Anal. Appl. **75**(1), 287–292 (1980). https://doi.org/10.1016/0022-247X(80)90323-6
48. Tanveer, M., Nazeer, W., Gdawiec, K.: New escape criteria for complex fractals generation in Jungck-CR orbit. Indian J. Pure Appl. Math. **51**(4), 1285–1303 (2020). https://doi.org/10.1007/s13226-020-0466-9
49. Tassaddiq, A., Tanveer, M., Azhar, M., Nazeer, W., Qureshi, S.: A four step feedback iteration and its applications in fractals. Fractal Fract. **6**(11), 662 (2022). https://doi.org/10.3390/fractalfract6110662
50. Tomar, A., Prajapati, D., Antal, S., Rawat, S.: Variants of Mandelbrot and Julia fractals for higher-order complex polynomials. Math. Methods Appl. Sci. (2022). https://doi.org/10.1002/mma.8262. (in press)
51. Wang, X., Jia, R.: Rendering of the inside structure of the generalized M set period bulbs based on the pre-period. Fractals **16**(4), 351–359 (2008). https://doi.org/10.1142/S0218348X08004125
52. Wang, X., Song, W.: The generalized M-J sets for bicomplex numbers. Nonlinear Dyn. **72**(1–2), 17–26 (2013). https://doi.org/10.1007/s11071-012-0686-6
53. Zhang, X., Wang, L., Zhou, Z., Niu, Y.: A chaos-based image encryption technique utilizing Hilbert curves and H-fractals. IEEE Access **7**, 74734–74746 (2019). https://doi.org/10.1109/ACCESS.2019.2921309
54. Zhou, H., Tanveer, M., Li, J.: Comparative study of some fixed-point methods in the generation of Julia and Mandelbrot sets. J. Math. **2020**, Article ID 7020921 (2020). https://doi.org/10.1155/2020/7020921
55. Zou, C., Shahid, A., Tassaddiq, A., Khan, A., Ahmad, M.: Mandelbrot sets and Julia sets in Picard-Mann orbit. IEEE Access **8**, 64411–64421 (2020). https://doi.org/10.1109/ACCESS.2020.2984689

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Sudesh Kumari[1] · Krzysztof Gdawiec[2] · Ashish Nandal[3] · Naresh Kumar[4] · Renu Chugh[5]

Sudesh Kumari
tanwarsudesh10@gmail.com

Ashish Nandal
ashishnandalmdu@gmail.com

Naresh Kumar
nks280@gmail.com

Renu Chugh
chughrenu1988@mdurohtak.ac.in

[1] Government College for Girls Sector 14, Gurugram 122001, India

[2] Institute of Computer Science, University of Silesia, Bedzinska 39, 41-200 Sosnowiec, Poland

[3] Government College Baund Kalan, Charkhidadri 127306, India

[4] Department of Mathematics, Guru Jambheshwar University of Science & Technology, Hisar 125001, India

[5] Department of Mathematics, Gurugram University, Gurugram 122001, India