An Application of Viscosity Approximation Type Iterative Method in the Generation of Mandelbrot and Julia Fractals

Sudesh Kumari, Krzysztof Gdawiec, Ashish Nandal,
Naresh Kumar, and Renu Chugh

Abstract. In this paper, we present an application of the viscosity approximation type iterative method introduced by Nandal et al. (Iteration Process for Fixed Point Problems and Zeros of Maximal Monotone Operators, Symmetry, 2019) to visualize and analyse the Julia and Mandelbrot sets for a complex polynomial of the type $T(z) = z^n + pz + r$, where $p, r \in \mathbb{C}$, and $n \geq 2$. This iterative method has many applications in solving various fixed point problems. We derive an escape criterion to visualize Julia and Mandelbrot sets via the proposed viscosity approximation type method. Moreover, we present several graphical examples of the fractals generated with the proposed iteration method.

Mathematics Subject Classification. 37F10, 39B12, 28A80.

Keywords. Viscosity approximation type iterative method, Escape criterion, Julia set, Mandelbrot set.

1. Introduction

Fractal geometry provides a general framework to study those natural objects which can not be easily described using Euclidean geometry. Usually, fractals are used for self-similar complex objects [39]. Nowadays, fractals have been applied for various purposes, for example, in cryptography [1], image encryption [44] or compression [2], art and design [33], pattern recognition [10]. The industry of security control systems, capacitors, radar systems, radio and antennas for wireless systems [6,18] were revolutionized with the applications of fractal theory. Fractals were also used in biology and medicine to study the culture of micro-organisms, the nervous system, etc. [9]. Moreover, architects and engineers applied fractal theory to sketch and design the maps of different projects [13].

Before the invention of computers, the figures of attractive objects, patterns and geometries had been sketched manually. Initially, the investigators of classical fractals like the Cantor set, the Koch curve, Sierpinski's triangle, and the Koch snowflake sketched graphs of these fractals manually. The French mathematician Gaston Julia [16] was the first to use an iteration process to define a new fractal named Julia set for the complex map $T_r(z) = z^2 + r$ where z is a complex variable, and r is a complex parameter. Thereafter, in 1975, Mandelbrot [27] used a computer for the first time to extend the work of Julia. He generated beautiful graphics for complex polynomials known as Mandelbrot sets. He defined a fractal as "a fragmented geometric shape that can be subdivided into congruent pieces, each of which is a reduced-size copy of the original one". Julia and Mandelbrot sets were extended from the complex numbers to quaternions [7], bicomplex numbers [43], tricomplex numbers [34] etc. Fractals have been analysed and visualized with various techniques (see, [4,16,19–22,27]). The escape criterion plays a prominent role in the generation of fractals (especially, Julia and Mandelbrot sets) which is a stopping criterion depending on the number of iterations required to determine whether the orbit of an initial point escapes to infinity or not. This criterion has been proved to be an appropriate mechanism to demonstrate the features of dynamical systems using various iterative procedures.

Fixed point iterative methods have been used as a milestone to generate and visualize fractals (especially Julia and Mandelbrot sets). These methods provide a unified treatment for finding the fixed points of non-linear operators. Generally, there are two main types of these fixed point iterative methods – one is the Mann type iterative method and the other is the Halpern type iterative method. The Mann iterative method was introduced by Mann [28] which is an averaged iterative method. However, this method does not converge strongly in general (see [11,38]). Many modified forms of the Mann iteration method have been investigated to achieve strong convergence.

In 1967, Halpern [12] introduced one of the most important iterative methods for finding a fixed point of nonexpansive type mappings. In 2000, Moudafi [29] introduced a famous generalization of the Halpern method, that is, the viscosity approximation method which is widely used to approximate a fixed point of a nonexpansive mapping and other classes of non-linear mappings (see [15,30–32,35] and the references therein).

In the literature, the Mann and similar types of fixed point iterative methods have been used so far for generating these fractals. For example, the explicit type of iterations: Mann iteration [36,37], Picard-Mann iteration [45], Picard-Mann iteration with s-convexity [40], Ishikawa iteration [5], Noor iteration [3], SP iteration with s-convexity [23], and the implicit ones: Jungck-CR iteration [41], Jungck-CR iteration with s-convexity [26], Jungck-SP iteration [25], Jungck-SP iteration with s-convexity [24].

In [32], Nandal et al. introduced a generalized form of viscosity approximation type iterative methods in the framework of a Hilbert space. They used their method to solve various problems, including a general system of variational inequalities, convex feasibility problems, zero point problems of inverse strongly monotone and maximal monotone mappings, split common null point problems, split feasibility problems, split monotone variational inclusion problems and split variational inequality problems. Due to the large number of applications of the proposed iterative method in the field of fixed point theory, we realized that this method also has a potential to generate fractals. Motivated by this fact, our paper used this new type of viscosity approximation method for generating fractals (Julia and Mandelbrot sets).

The rest of the paper is organized as follows: Sect. 2 deals with the basic definitions, facts and notations. In Sect. 3, we derive the escape criterion which is used to draw Julia and Mandelbrot sets. Next, in Sect. 4, we present pseudocodes of escape time algorithms for generating Mandelbrot and Julia sets via the proposed iteration method. Moreover, we present some graphical examples of the sets obtained with those algorithms. Finally, we conclude our findings in Sect. 5.

2. Preliminaries

In this section, we give some basic definitions, notations and facts from the literature for the completeness of the paper.

Definition 2.1. (*Julia set* [8]) The *filled Julia set* F_{T_r} of a function $T_r : \mathbb{C} \to \mathbb{C}$, where $r \in \mathbb{C}$ is a parameter, is the set of points in the complex plane whose orbits are bounded, i.e.,

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

where T_r^j denotes the jth iteration of the function T_r. The boundary of the filled Julia set F_{T_r} is said to be the *Julia set* J_{T_r} of the function T_r, i.e., $J_{T_r} = \partial F_{T_r}$.

In 1975, Mandelbrot [27] defined the Mandelbrot set as follows:

Definition 2.2. (*Mandelbrot set* [8]) The collection of all complex numbers $r \in \mathbb{C}$ for which the filled Julia set F_{T_r} remains connected is known as *Mandelbrot set* M, i.e.,

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

Equivalently,

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

Initially, in 2000 Moudafi [29] investigated the viscosity approximation method. In the complex plane, this method can be defined as

Definition 2.3. (*Viscosity approximation method* [29]) Let $T \colon \mathbb{C} \to \mathbb{C}$ be a complex map. For an initial point $z_0 \in \mathbb{C}$, consider the following sequence $\{z_j\}$ of iterates

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

where $\alpha_j \in (0,1)$ and $g \colon \mathbb{C} \to \mathbb{C}$ is a contraction mapping. The iterative method given in (4) is called the *viscosity approximation method*.

It is remarkable to note that if in (4), we consider the mapping g as a constant mapping, i.e., $g(z) = b$, where $b \in \mathbb{C}$, then the sequence $\{z_j\}$ reduces to the Halpern iteration [12].

In [32], Nandal et al. considered a new generalized viscosity approximation type method. In the complex plane, this method can be defined as follows: starting with an arbitrary initial point $z_0 \in \mathbb{C}$, the sequence $\{z_j\}$ generated by

$$z_{j+1} = J_{\rho_j}^{B_1} T_k^j T_{k-1}^j \cdots T_1^j y_j,$$
$$y_j = \alpha_j g(z_j) + (1 - \alpha_j) J_{\mu_j}^{B_2} V_j z_j, \quad j \geq 0, \tag{5}$$

where g is a contraction, $V_j = (1 - \beta_j)I + \beta_j V$, $T_i^j = (1 - \gamma_j^i)I + \gamma_j^i T_i$ for $i = 1, 2, \cdots, k$ with $\alpha_j, \beta_j, \gamma_j^i \in (0,1)$, the resolvent operators $J_{\rho_j}^{B_1} = (I + \rho_j B_1)^{-1}$ and $J_{\mu_j}^{B_2} = (I + \mu_j B_2)^{-1}$ are associated with monotone operators B_1 and B_2, respectively, with $\rho_j, \mu_j \in (0, \infty)$.

3. Main result

In the literature, usually, the authors study the escape criterion for the function of the form $z^n + r$. To gain more control over the shape of the generated set, we will consider a function that has a parameter controlling the linear part, i.e.,

$$T(z) = z^n + pz + r, \tag{6}$$

where $n \geq 2$ and $p, r \in \mathbb{C}$. For this function, we prove a general escape criterion by using a form of the viscosity approximation type method given in (5).

Let us assume that $k = 1$, and that we use constant sequences $\alpha_j = \alpha$, $\beta_j = \beta$, $\gamma_j^1 = \gamma$, $\rho_j = \rho$, $\mu_j = \mu$, where $\alpha, \beta, \gamma \in (0,1)$ and $\rho, \mu \in (0, \infty)$. Moreover, let us assume that $g(z) = az + b$ is a complex contraction with $a, b \in \mathbb{C}$ and $|a| < 1$, and that $B_1(z) = mz$ and $B_2(z) = qz$, where $m, q \in \mathbb{R}$. Thus, $J_\rho^{B_1}(z) = \frac{z}{1+m\rho}$ and $J_\mu^{B_2}(z) = \frac{z}{1+q\mu}$. Also, let $T_1 = V = T$ where T is given in (6). For such parameters the iteration (5) takes the following form:

$$z_{j+1} = J_\rho^{B_1} T_\gamma y_j,$$

$$y_j = \alpha g(z_j) + (1-\alpha)J_\mu^{B_2}V_\beta z_j, \tag{7}$$

where

$$T_\gamma = (1-\gamma)I + \gamma T,$$
$$V_\beta = (1-\beta)I + \beta T. \tag{8}$$

Theorem 3.1. *Let* $|z_0| \geq \max\{|r|,|b|\} > \max$ $\left\{ \left(\frac{(1+\alpha(1+|a|))|1+q\mu|+(1-\alpha)(1+\beta|p|)}{\beta(1-\alpha)} \right)^{\frac{1}{n-1}}, \left(\frac{|1+m\rho|+1+\gamma|p|}{\gamma} \right)^{\frac{1}{n-1}} \right\}$. *Then* $|z_j| \to \infty$ *as* $j \to \infty$ *where* $\{z_j\}$ *is defined in* (7).

Proof. From the construction of V_β, we have

$$|V_\beta z_j| = |((1-\beta)I + \beta T)z_j|, \ \ j \geq 0.$$

For $j = 0$, we have

$$
\begin{aligned}
|V_\beta z_0| &= |((1-\beta)I + \beta T)z_0| \\
&= |(1-\beta)z_0 + \beta T z_0| \\
&= |(1-\beta)z_0 + \beta(z_0{}^n + pz_0 + r)| \\
&\geq |\beta z_0^n + \beta p z_0 + \beta r| - |(1-\beta)z_0| \\
&\geq \beta|z_0^n| - \beta|p||z_0| - \beta|r| - (1-\beta)|z_0|.
\end{aligned}
$$

The assumption $|z_0| \geq \max\{|r|,|b|\}$ implies $-|r| \geq -|z_0|$, therefore, we obtain

$$
\begin{aligned}
|V_\beta z_0| &\geq \beta|z_0^n| - \beta|p||z_0| - \beta|z_0| - (1-\beta)|z_0| \\
&= \beta|z_0^n| - \beta|p||z_0| - |z_0| \\
&= |z_0|(\beta|z_0|^{n-1} - (1+\beta|p|)).
\end{aligned}
$$

Thus,

$$|V_\beta z_0| \geq |z_0|(\beta|z_0|^{n-1} - (1+\beta|p|)). \tag{9}$$

From (7), consider

$$
\begin{aligned}
|y_0| &= |\alpha g(z_0) + (1-\alpha)J_\mu^{B_2}(V_\beta z_0)| \\
&= \left| \alpha(az_0 + b) + (1-\alpha)\frac{V_\beta z_0}{1+q\mu} \right| \\
&\geq (1-\alpha)\left| \frac{V_\beta z_0}{1+q\mu} \right| - \alpha|az_0| - \alpha|b|.
\end{aligned}
$$

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

$$|y_0| \geq (1-\alpha)\frac{|V_\beta z_0|}{|1+q\mu|} - \alpha|a||z_0| - \alpha|z_0|.$$

Using (9), we get

$$|y_0| \geq (1-\alpha)\frac{|z_0|(\beta|z_0|^{n-1}-(1+\beta|p|))}{|1+q\mu|} - \alpha(1+|a|)|z_0|$$

$$= |z_0|\left(\frac{(1-\alpha)\beta|z_0|^{n-1}-(1-\alpha)(1+\beta|p|)}{|1+q\mu|} - \alpha(1+|a|)\right).$$

Thus, we have

$$|y_0| \geq |z_0|\left(\frac{(1-\alpha)\beta|z_0|^{n-1}-(1-\alpha)(1+\beta|p|)}{|1+q\mu|} - \alpha(1+|a|)\right). \qquad (10)$$

Our assumption $|z_0| > \left(\frac{(1+\alpha(1+|a|))|1+q\mu|+(1-\alpha)(1+\beta|p|)}{\beta(1-\alpha)}\right)^{\frac{1}{n-1}}$ gives

$$\frac{(1-\alpha)\beta|z_0|^{n-1}-(1-\alpha)(1+\beta|p|)}{|1+q\mu|} - \alpha(|a|+1) > 1. \qquad (11)$$

Using (11) in (10), we obtain

$$|y_0| > |z_0|. \qquad (12)$$

Now, from the construction of T_γ, we have

$$\begin{aligned} T_\gamma(y_j) &= ((1-\gamma)I + \gamma T)y_j \\ &= (1-\gamma)y_j + \gamma T(y_j) \\ &= (1-\gamma)y_j + \gamma(y_j^n + py_j + r). \end{aligned}$$

Therefore,

$$T_\gamma(y_j) = (1-\gamma)y_j + \gamma(y_j^n + py_j + r). \qquad (13)$$

Further, from (7), consider

$$|z_1| = |J_\rho^{B_1}T_\gamma(y_0)|.$$

From (13), we have

$$\begin{aligned} |z_1| &= |J_\rho^{B_1}((1-\gamma)y_0 + \gamma(y_0^n + py_0 + r))| \\ &= \left|\frac{(1-\gamma)y_0 + \gamma(y_0^n + py_0 + r)}{1+m\rho}\right| \\ &\geq \frac{|\gamma(y_0^n + py_0 + r)| - (1-\gamma)|y_0|}{|1+m\rho|} \\ &\geq \frac{\gamma|y_0^n| - \gamma|p||y_0| - \gamma|r| - (1-\gamma)|y_0|}{|1+m\rho|} \\ &\geq \frac{\gamma|y_0^n| - \gamma|p||y_0| - \gamma|z_0| - (1-\gamma)|y_0|}{|1+m\rho|} \\ &= \frac{|y_0|(\gamma|y_0^{n-1}| - \gamma|p| - (1-\gamma)) - \gamma|z_0|}{|1+m\rho|}. \end{aligned}$$

Now, from (12), we obtain

$$|z_1| > \frac{|z_0|(\gamma|z_0^{n-1}| - \gamma|p| - 1 + \gamma) - \gamma|z_0|}{|1 + m\rho|}$$

$$= |z_0|\frac{(\gamma|z_0^{n-1}| - \gamma|p| - 1)}{|1 + m\rho|}.$$

This gives

$$|z_1| > |z_0|\frac{(\gamma|z_0^{n-1}| - \gamma|p| - 1)}{|1 + m\rho|}. \tag{14}$$

Our assumption $|z_0| > \left(\frac{|1+m\rho|+1+\gamma|p|}{\gamma}\right)^{\frac{1}{n-1}}$ gives

$$\frac{\gamma|z_0^{n-1}| - \gamma|p| - 1}{|1 + m\rho|} > 1. \tag{15}$$

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

$$\frac{\gamma|z_0^{n-1}| - \gamma|p| - 1}{|1 + m\rho|} > \lambda + 1 > 1. \tag{16}$$

Using (16) in (14), we have

$$|z_1| > (\lambda + 1)\,|z_0|.$$

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

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

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

We obtain the following corollary as a refinement of Theorem 3.1:

Corollary 3.2. *Let*

$$|z_0| > \max\left\{|r|, |b|, \left(\frac{(1 + \alpha(1 + |a|))|1 + q\mu| + (1 - \alpha)(1 + \beta|p|)}{\beta(1 - \alpha)}\right)^{\frac{1}{n-1}}, \right.$$

$$\left.\left(\frac{|1 + m\rho| + 1 + \gamma|p|}{\gamma}\right)^{\frac{1}{n-1}}\right\}, \tag{17}$$

then $|z_j| \to \infty$ *as* $j \to \infty$.

Corollary 3.3. *Suppose that:*

$$|z_k| > \max\left\{|r|, |b|, \left(\frac{(1 + \alpha(1 + |a|))|1 + q\mu| + (1 - \alpha)(1 + \beta|p|)}{\beta(1 - \alpha)}\right)^{\frac{1}{n-1}}, \right.$$

$$\left.\left(\frac{|1 + m\rho| + 1 + \gamma|p|}{\gamma}\right)^{\frac{1}{n-1}}\right\}, \tag{18}$$

for some $k \geq 0$. Then, there exists $\lambda > 0$ such that $|z_{k+j}| > (1 + \lambda)^j |z_k|$ and we have $|z_j| \to \infty$ as $j \to \infty$.

4. Graphical examples of mandelbrot and julia sets via the proposed iteration method

Corollaries 3.2 and 3.3 enable us to generate the Julia and Mandelbrot sets of the nth degree complex polynomial $T_1(z) = z^n + pz + r$, where $n \geq 2$ and $p, r \in \mathbb{C}$, via the iteration method given in (7) using the escape time algorithm. Namely, if for some j, the element z_j lies outside the circle of radius:

$$R = \max \left\{ |r|, |b|, \left(\frac{(1 + \alpha(1 + |a|))(1 + q\mu|) + (1 - \alpha)(1 + \beta|p|)}{\beta(1 - \alpha)} \right)^{\frac{1}{n-1}}, \right.$$
$$\left. \left(\frac{|1 + m\rho| + 1 + \gamma|p|}{\gamma} \right)^{\frac{1}{n-1}} \right\}, \tag{19}$$

then the orbit of $|z_0|$ escapes to infinity. Which implies that the point z_0 does not lie in the filled Julia set. If z_j does not exceed this bound, then by definition, z_0 lies in the filled Julia set. In Algorithm 1, we present the pseudocode of a method for generating Julia sets via the viscosity approximation type method given in (7). We generate Julia sets in the given area $A \subset \mathbb{C}$ and the given colour map. Since an infinite number of iterations cannot be performed, we fix the maximum number of iterations at K iterations.

Using a very similar escape time algorithm, we can generate a Mandelbrot set via the viscosity approximation type method given in (7). The pseudocode of this algorithm is presented in Algorithm 2. The set is generated in the area $A \subset \mathbb{C}$ using the maximal K iterations and a given colour map.

The graphical examples in this section were obtained by a program written in Mathematica 12. In all the examples presented in this section, we used the colour map presented in Fig. 1. The resolution of the images was set to 800×800 pixels, and we used $K = 50$.

4.1. Examples of Julia sets

In the first example, we generated Julia sets for a quadratic function. The parameters used to generate these sets were the following: $n = 2$, $p = 0.09$
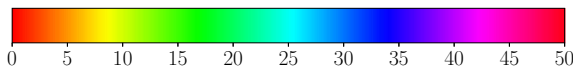

FIGURE 1. Colour map used in the graphical examples

Algorithm 1: Julia set generation

Input: $T(z) = z^n + pz + r$, where $p, r \in \mathbb{C}$ and $n \geq 2$; $A \subset \mathbb{C}$ – area; K – the maximum number of iterations; $\alpha, \beta, \gamma \in (0, 1)$, $m, q \in \mathbb{R}$ and $\rho, \mu \in (0, \infty)$ – parameters for the viscosity approximation 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: Julia set for area A.

1   $R =$
$$\max\left\{|r|, |b|, \left(\frac{(1+\alpha(1+|a|))|1+q\mu|+(1-\alpha)(1+\beta|p|)}{\beta(1-\alpha)}\right)^{\frac{1}{n-1}}, \left(\frac{|1+m\rho|+1+\gamma|p|}{\gamma}\right)^{\frac{1}{n-1}}\right\}$$

2   **for** $z_0 \in A$ **do**

3      $j = 0$

4   **while** $|z_j| < R$ **and** $j < K$ **do**

5         $v = (1 - \beta)z_j + \beta T(z_j)$

6         $y_j = \alpha g(z_j) + (1 - \alpha)\frac{v}{1+q\mu}$

7         $z_{j+1} = \frac{(1-\gamma)y_j + \gamma T(y_j)}{1+m\rho}$

8         $j = j + 1$

9   colour z_0 with $colourmap[j]$

Algorithm 2: Mandelbrot set generation

Input: $T_r(z) = z^n + pz + r$, where $p \in \mathbb{C}$, $n \geq 2$, and $r \in \mathbb{C}$ is a parameter; $A \subset \mathbb{C}$ – area; K – the maximum number of iterations; $\alpha, \beta, \gamma \in (0, 1)$, $m, q \in \mathbb{R}$ and $\rho, \mu \in (0, \infty)$– parameters for the viscosity approximation 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: Mandelbrot set for area A.

1   **for** $r \in A$ **do**

2      $R =$
$$\max\left\{|r|, |b|, \left(\frac{(1+\alpha(1+|a|))|1+q\mu|+(1-\alpha)(1+\beta|p|)}{\beta(1-\alpha)}\right)^{\frac{1}{n-1}}, \left(\frac{|1+m\rho|+1+\gamma|p|}{\gamma}\right)^{\frac{1}{n-1}}\right\}$$

3      $j = 0$

4      $z_0 = r$

5   **while** $|z_j| < R$ **and** $j < K$ **do**

6         $v = (1 - \beta)z_j + \beta T_r(z_j)$

7         $y_j = \alpha g(z_j) + (1 - \alpha)\frac{v}{1+q\mu}$

8         $z_{j+1} = \frac{(1-\gamma)y_j + \gamma T_r(y_j)}{1+m\rho}$

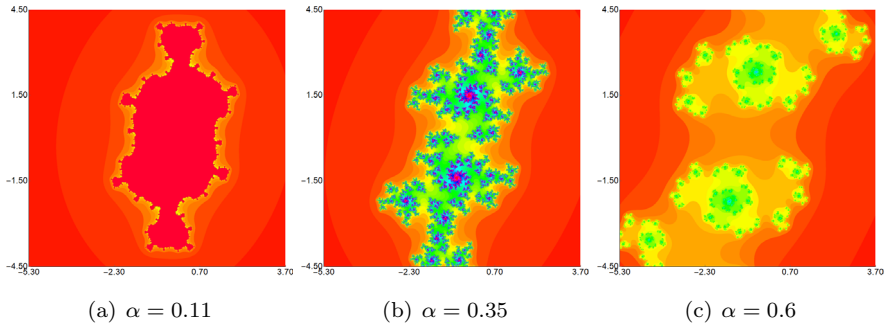9 $j = j + 1$

10   colour r with $colourmap[j]$

(a) $\alpha = 0.11$ (b) $\alpha = 0.35$ (c) $\alpha = 0.6$

FIGURE 2. Julia set for $n = 2$, $p = 0.09 - 0.09i$, $r = 0.02 + 0.02i$ generated via (7) with $\beta = 0.89$, $\gamma = 0.11$ and varying α

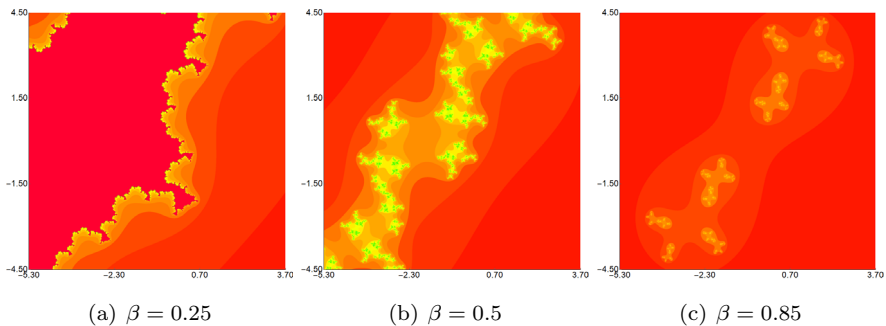(a) $\beta = 0.25$ (b) $\beta = 0.5$ (c) $\beta = 0.85$

FIGURE 3. Julia set for $n = 2$, $p = 0.09 - 0.09i$, $r = 0.02 + 0.02i$ generated via (7) with $\alpha = 0.56$, $\gamma = 0.44$ and varying β

$- 0.09i$, $r = 0.02 + 0.02i$, $A = [-5.3, 3.7] \times [-4.5, 4.5]$, $K = 50$, $g(z) = 0.85z + 2.07 - 6.92i$, $m = 0.52$, $q = 0.75$, $\rho = 0.45$, $\mu = 0.27$. In the example, we divided the images into three groups. In each group, we fix two parameters from α, β, γ and vary the remaining one. In Fig. 2, we see images generated for fixed $\beta = 0.89$, $\gamma = 0.11$, and varying α: (a) 0.11, (b) 0.35, (c) 0.6. In Fig. 3, we fixed $\alpha = 0.56$, $\gamma = 0.44$ and varied β: (a) 0.25, (b) 0.5, (c) 0.85. Finally, in Fig. 4, the images were generated for fixed $\alpha = 0.33$, $\beta = 0.67$ and varying γ: (a) 0.2, (b) 0.55, (c) 0.85. From the three figures, we clearly see that each of the three parameters has a great impact on the shape of the Julia set and its size. We also notice that when the value of the varying parameter increases, then the set loses its connectivity and tends to a dust-like set. Moreover, we see a great variety of shapes for the fixed function.

In the next example, we generated Julia sets for a fourth order function. The parameters used to generate these sets were the following: $n = 4$, $p =$
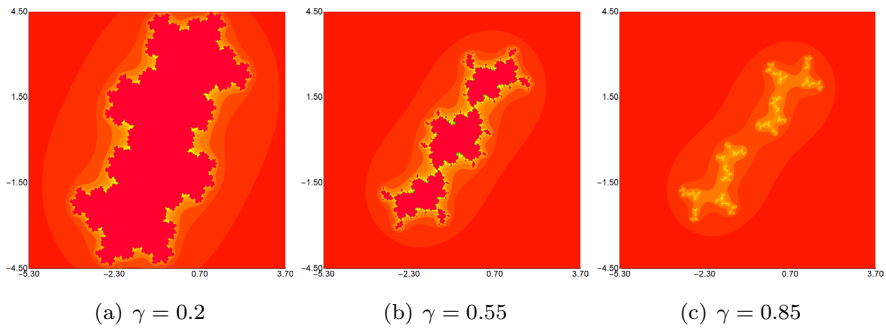
(a) $\gamma = 0.2$ (b) $\gamma = 0.55$ (c) $\gamma = 0.85$

FIGURE 4. Julia set for $n = 2$, $p = 0.09 - 0.09i$, $r = 0.02 + 0.02i$ generated via (7) with $\alpha = 0.33$, $\beta = 0.67$ and varying γ

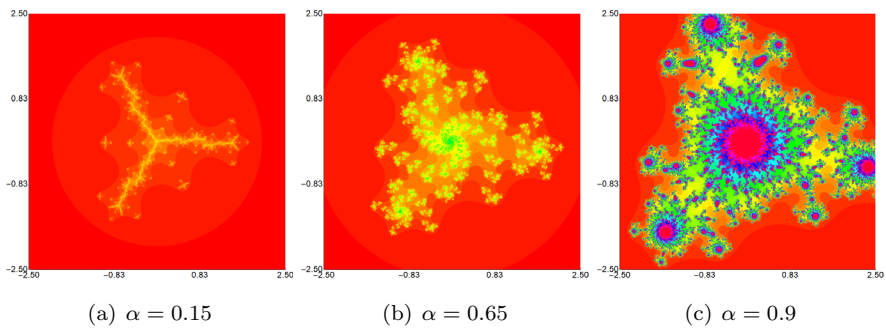(a) $\alpha = 0.15$ (b) $\alpha = 0.65$ (c) $\alpha = 0.9$

FIGURE 5. Julia set for $n = 4$, $p = -3.45 - 0.01i$, $r = -0.01 - 0.01i$ generated via (7) with $\beta = 0.67$, $\gamma = 0.67$ and varying α

$-3.45 - 0.01i$, $r = -0.01 - 0.01i$, $A = [-2.5, 2.5]^2$, $K = 50$, $g(z) = (-0.37 + 0.37i)z - 0.01 + 0.01i$, $m = 0.13$, $q = 0.61$, $\rho = 0.54$, $\mu = 1.12$. Like in the first example, the images are divided into three groups according to the varying parameter. In Fig. 5, we fixed $\beta = 0.67$, $\gamma = 0.67$ and vary the value of α: (a) 0.14, (b) 0.65, (c) 0.9. The values of α and γ were fixed in Fig. 6, where $\alpha = 0.33$, $\gamma = 0.44$, and the values of β were the following: (a) 0.5, (b) 0.79, (c) 0.9. In the last figure in this example – Fig. 7 – we fixed $\alpha = 0.89$, $\beta = 0.67$ and varied γ: (a) 0.45, (b) 0.67, (c) 0.95. In Fig. 5, we see that the generated Julia set becomes bigger and more complex with the increase in the value of α, whereas in Figs. 6, 7, we see an opposite tendency. The parameters also have influence on the symmetry of the sets. In most of the cases the sets have a rotational symmetry, but for some values of the parameters the symmetry is broken, see, for instance, Figs. 5c or 6b.
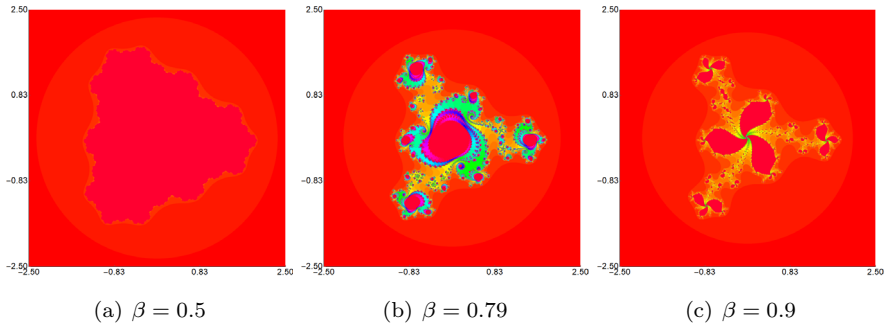
(a) $\beta = 0.5$ (b) $\beta = 0.79$ (c) $\beta = 0.9$

Figure 6. Julia set for $n = 4$, $p = -3.45 - 0.01i$, $r = -0.01 -$
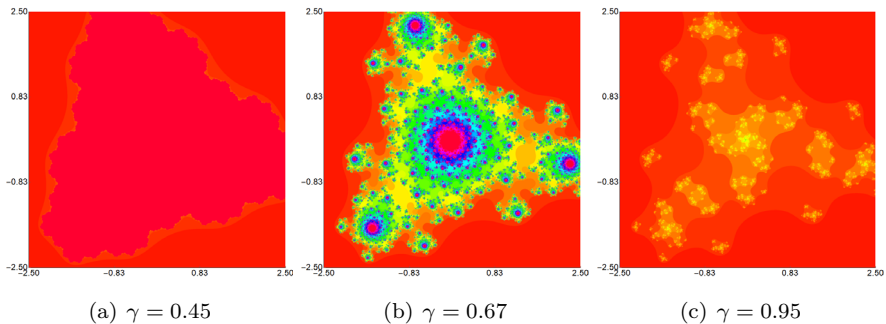$0.01i$ generated via (7) with $\alpha = 0.33$, $\gamma = 0.44$ and varying β

(a) $\gamma = 0.45$ (b) $\gamma = 0.67$ (c) $\gamma = 0.95$

Figure 7. Julia set for $n = 4$, $p = -3.45 - 0.01i$, $r = -0.01 -$
$0.01i$ generated via (7) with $\alpha = 0.89$, $\beta = 0.67$ and varying γ

The shape of the set, except for the iteration parameters (α, β, γ), can
be changed using the p parameter of T_r. To show this, for both the examples
used in this section, we generated Julia sets for fixed α, β, γ, but with various
values of p. For the quadratic Julia set, we used $\alpha = 0.33$, $\beta = 0.67$, $\gamma = 0.35$,
whereas for the fourth order function we used $\alpha = 0.45$, $\beta = 0.65$, $\gamma = 0.56$.
Figures 8, 9 present the obtained images of Julia sets in the quadratic and the
fourth order case, respectively. From the images, we see that the imaginary
part of p plays an important role in obtaining the swirls in the pattern. In
the quadratic case, the negative imaginary part of p caused that the swirls are
smaller, whereas the positive value of the same magnitude caused that many
swirls appeared, and that the set lost its connectivity. In the quartic case, we
see a similar behaviour. However, this time for the negative values, we get
more smaller swirls, and for the positive ones, the swirls disappear.
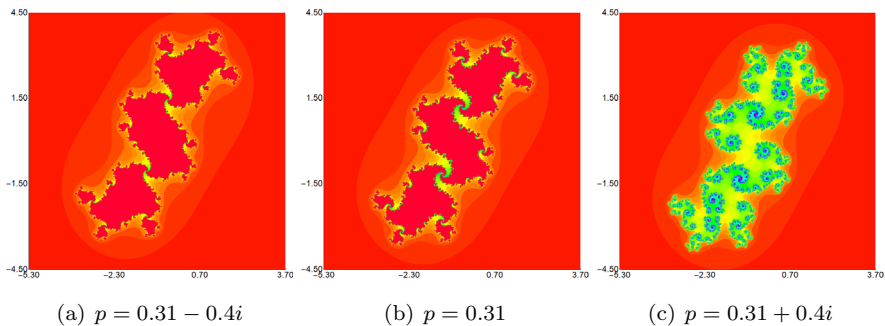
(a) $p = 0.31 - 0.4i$ (b) $p = 0.31$ (c) $p = 0.31 + 0.4i$

FIGURE 8. Julia set for $n = 2$, $r = 0.02 + 0.02i$, $\alpha = 0.33$, $\beta = 0.67$, $\gamma = 0.35$ generated via (7) with varying p

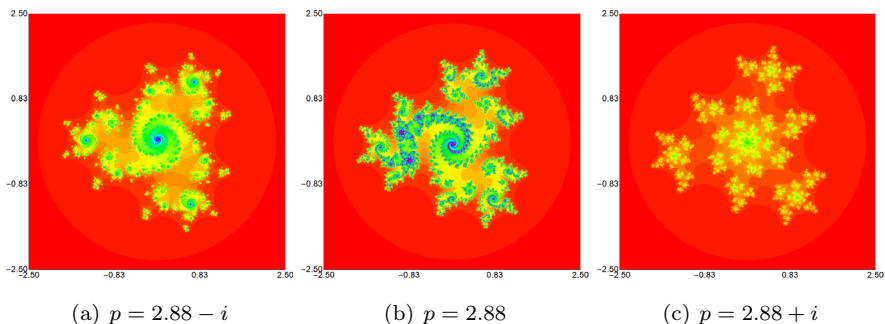(a) $p = 2.88 - i$ (b) $p = 2.88$ (c) $p = 2.88 + i$

FIGURE 9. Julia set for $n = 4$, $r = -0.01 - 0.01i$, $\alpha = 0.45$, $\beta = 0.65$, $\gamma = 0.56$ generated via (7) with varying p

To show the variety of Julia sets that can be generated by the proposed method, in the last example we present various Julia sets. They are presented in Fig. 10, and the parameters used to generate them were the following:

(a) $n = 3$, $p = 4.89 - 0.2i$, $r = 0.09 - 2.91i$, $A = [-3,3]^2$, $K = 50$, $g(z) = 0.58z - 0.02 + 2.4i$, $m = 0.65$, $q = 0.75$, $\rho = 0.45$, $\mu = 2.05$, $\alpha = 0.01$, $\beta = 0.5$, $\gamma = 0.36$,

(b) $n = 5$, $p = 4.9 - 0.2i$, $r = 0.09 - 0.09i$, $A = [-2,2]^2$, $K = 50$, $g(z) = (0.3 + 0.01i)z - 0.02 + 0.04i$, $m = 0.95$, $q = 0.95$, $\rho = 0.45$, $\mu = 1.05$, $\alpha = 0.76$, $\beta = 0.45$, $\gamma = 0.67$,

(c) $n = 10$, $p = -5.54 - 0.02i$, $r = 0.02 - 0.09i$, $A = [-1.25, 1.25]^2$, $K = 50$, $g(z) = (0.04 - 0.2i)z - 0.01 + 0.02i$, $m = 0.41$, $q = 0.86$, $\rho = 1.85$, $\mu = 0.62$, $\alpha = 0.45$, $\beta = 0.75$, $\gamma = 0.36$.
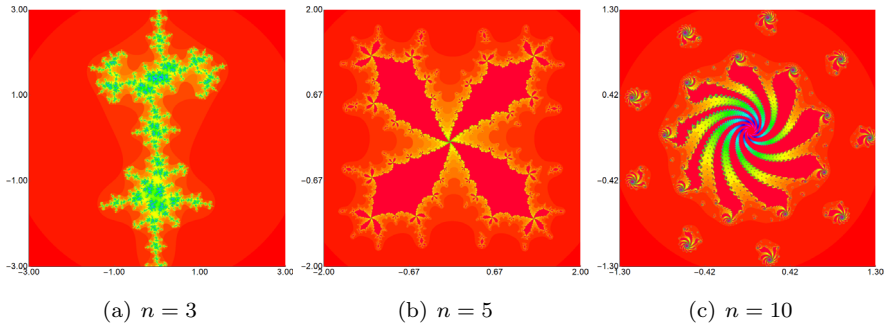
(a) $n = 3$ (b) $n = 5$ (c) $n = 10$

Figure 10. Various Julia sets generated via (7)

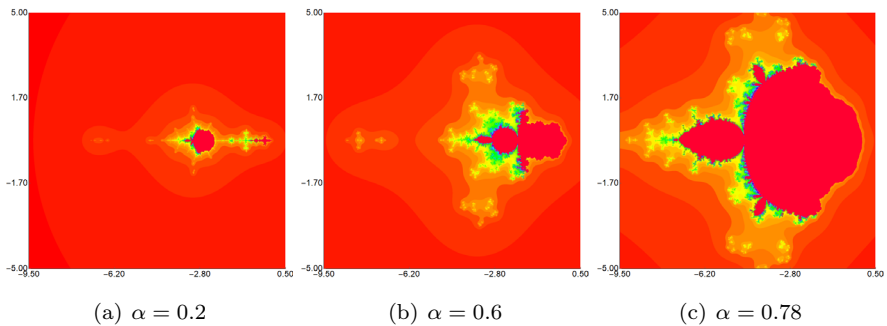(a) $\alpha = 0.2$ (b) $\alpha = 0.6$ (c) $\alpha = 0.78$

Figure 11. Mandelbrot set for $n = 2$, $p = 4.62 - 0.002i$ generated via (7) with $\beta = 0.56$, $\gamma = 0.78$ and varying α

4.2. Examples of mandelbrot sets

In the first example in this section, we generated Mandelbrot sets for a quadratic function. The parameters used to generate these sets were the following: $n = 2$, $p = 4.62 - 0.002i$, $A = [-9.5, 0.5] \times [-5, 5]$, $K = 50$, $g(z) = 0.5z + 0.01 + 0.02i$, $m = 0.7$, $q = 0.01$, $\rho = 0.05$, $\mu = 0.62$. In the example, we divided the images into three groups. In each group, we fix two parameters from α, β, γ and vary the remaining one. In Fig. 11, we see images generated for fixed $\beta = 0.56$, $\gamma = 0.78$, and varying α: (a) 0.2, (b) 0.6, (c) 0.78. In Fig. 12, we fixed $\alpha = 0.33$, $\gamma = 0.11$ and varied β: (a) 0.4, (b) 0.6, (c) 0.9. Finally, in Fig. 13, the images were generated for fixed $\alpha = 0.78$, $\beta = 0.67$ and varying γ: (a) 0.4, (b) 0.65, (c) 0.9. From the figures, we see that the three parameters have a great impact on the size of the generated Mandelbrot set. In Fig. 11,
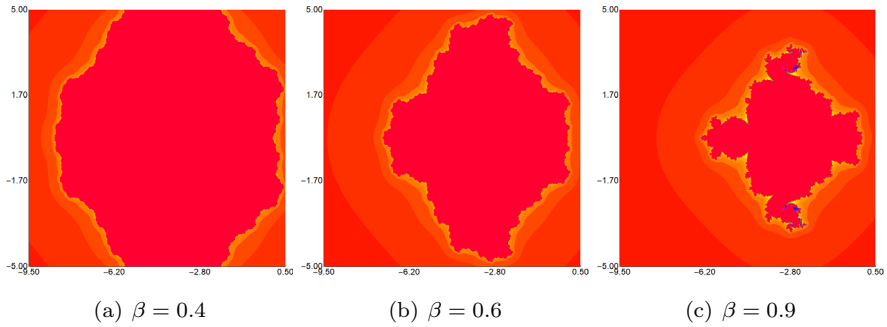
(a) $\beta = 0.4$ (b) $\beta = 0.6$ (c) $\beta = 0.9$

FIGURE 12. Mandelbrot set for $n = 2$, $p = 4.62 - 0.002i$ generated via (7) with $\alpha = 0.33$, $\gamma = 0.11$ and varying β

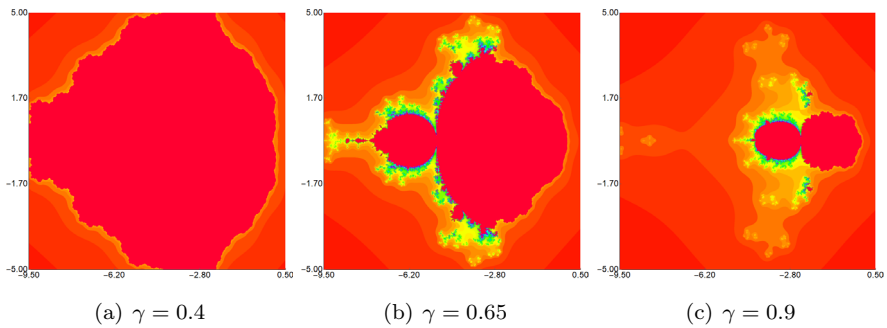(a) $\gamma = 0.4$ (b) $\gamma = 0.65$ (c) $\gamma = 0.9$

FIGURE 13. Mandelbrot set for $n = 2$, $p = 4.62 - 0.002i$ generated via (7) with $\alpha = 0.78$, $\beta = 0.67$ and varying γ

the set grows when α increases, whereas in Figs. 12 and 13, the set grows when β or γ decreases. We also see that in each case the set has axial symmetry, and that using the viscosity approximation iteration (7) we can generate sets of various shapes for a fixed function.

    In the next example, we generated Mandelbrot sets for a fifth order function. The parameters used to generate these sets were the following: $n = 5$, $p = 3.4 + 0.01i$, $A = [-2, 2]^2$, $K = 50$, $g(z) = (0.3 + 0.37i)z + 0.1 + 0.1i$, $m = 0.21$, $q = 0.91$, $\rho = 1.44$, $\mu = 0.72$. Like in the first example, the images are divided into three groups according to the varying parameter. In Fig. 14, we fixed $\beta = 0.56$, $\gamma = 0.44$ and varied the value of α: (a) 0.25, (b) 0.5, (c) 0.75. The values of α and γ were fixed in Fig. 15, where $\alpha = 0.11$, $\gamma = 0.33$, and the values of β were the following: (a) 0.2, (b) 0.5, (c) 0.8. In the last
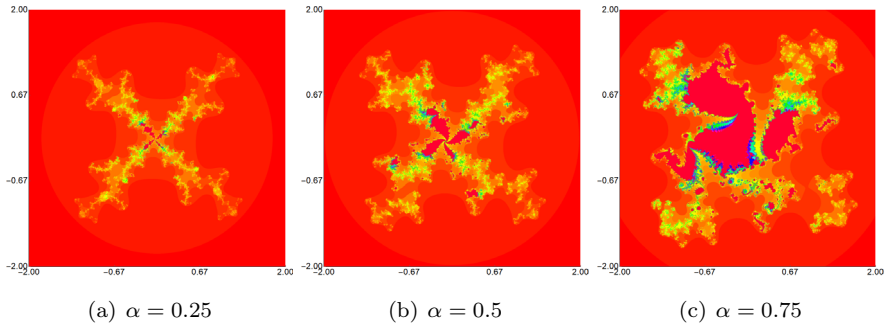
(a) $\alpha = 0.25$ (b) $\alpha = 0.5$ (c) $\alpha = 0.75$

FIGURE 14. Mandelbrot set for $n = 5$, $p = 3.4 + 0.01i$ generated via (7) with $\beta = 0.56$, $\gamma = 0.44$ and varying α

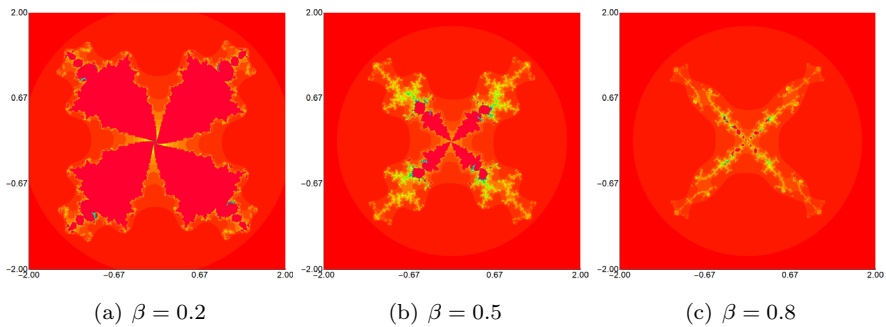(a) $\beta = 0.2$ (b) $\beta = 0.5$ (c) $\beta = 0.8$

FIGURE 15. Mandelbrot set for $n = 5$, $p = 3.4 + 0.01i$ generated via (7) with $\alpha = 0.11$, $\gamma = 0.33$ and varying β

figure in this example – Fig. 16 – we fixed $\alpha = 0.56$, $\beta = 0.33$ and varied γ: (a) 0.25, (b) 0.45, (c) 0.85. In the three figures, we see that each of the three parameters has a great impact not only on the shape, like in all the previous examples, but also on the symmetry. Most of the presented sets do not possess symmetry, but there is one example of a set with rotational symmetry, see Fig. 15a. Moreover, we can notice that the shape changes in a very irregular way.

Similarly, as in the case of the Julia sets, we can change the shape of the Mandelbrot set by changing the value of p in T_r. To show this effect, we generated Mandelbrot sets for fixed values of α, β and γ and various values of p. For the quadratic Mandelbrot set, we used $\alpha = 0.78$, $\beta = 0.56$, $\gamma = 0.78$, whereas for the fifth order function $\alpha = 0.15$, $\beta = 0.91$ and $\gamma = 0.01$. Figures 17, 18 present the obtained images of Mandelbrot sets in the quadratic
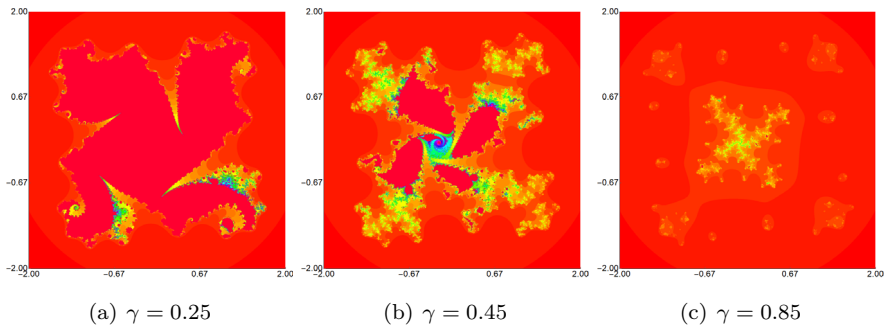
(a) $\gamma = 0.25$ (b) $\gamma = 0.45$ (c) $\gamma = 0.85$

FIGURE 16. Mandelbrot set for $n = 5$, $p = 3.4 + 0.01i$ generated via (7) with $\alpha = 0.56$, $\beta = 0.33$ and varying γ

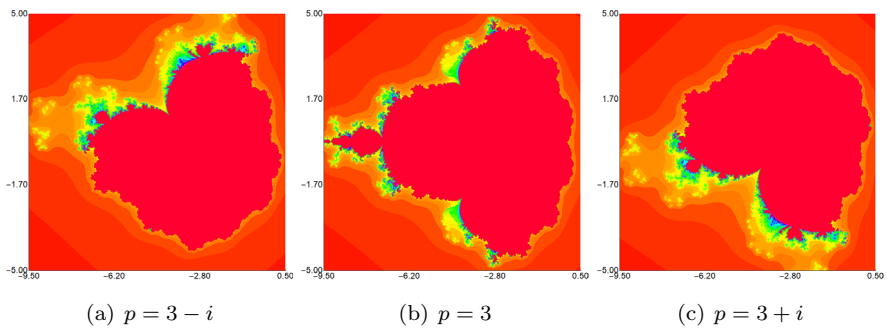(a) $p = 3 - i$ (b) $p = 3$ (c) $p = 3 + i$

FIGURE 17. Mandelbrot set for $n = 2$, $\alpha = 0.78$, $\beta = 0.56$, $\gamma = 0.78$ generated via (7) with varying p

and the fifth order cases, respectively. For the quadratic Mandelbrot set, we see that the introduction of the imaginary part into p causes the set to rotate, and the direction of rotation depends on the sign of the imaginary part of p. In the quintic case, the imaginary part of p causes that the set loses its connectivity, and some rotation appears. For the negative value of the imaginary part, the rotation is in a clockwise direction, whereas for the positive one, the rotation is in a counter-clockwise direction.

To show the variety of Mandelbrot sets that can be generated by the proposed method, in the last example we present various Mandelbrot sets. They are presented in Fig. 19, and the parameters used to generate them were the following:

(a) $n = 4$, $p = -3.04 - 0.01i$, $A = [-1.6, 1.6]^2$, $K = 50$, $g(z) = (0.3 - 0.17i)z + 0.01 - 0.09i$, $m = 0.05$, $q = 0.51$, $\rho = 0.64$, $\mu = 1.42$, $\alpha = 0.15$, $\beta = 0.65$, $\gamma = 0.56$,
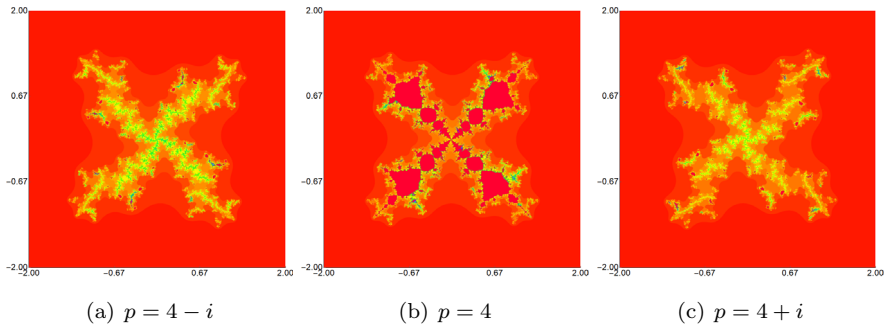
(a) $p = 4 - i$ (b) $p = 4$ (c) $p = 4 + i$

FIGURE 18. Mandelbrot set for $n = 5$, $\alpha = 0.15$, $\beta = 0.91$, $\gamma = 0.01$ generated via (7) with varying p

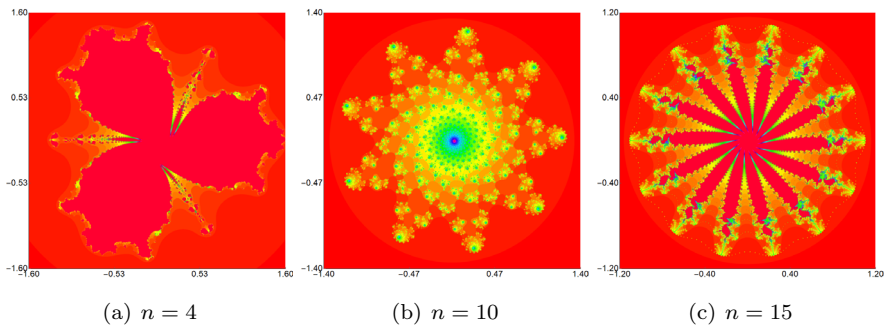(a) $n = 4$ (b) $n = 10$ (c) $n = 15$

FIGURE 19. Various Mandelbrot sets generated via (7)

(b) $n = 10$, $p = -5.54 - 0.02i$, $A = [-1.4, 1.4]^2$, $K = 50$, $g(z) = (0.02 - 0.8i)z + 0.01 + 0.02i$, $m = 0.41$, $q = 0.86$, $\rho = 1.85$, $\mu = 0.62$, $\alpha = 0.45$, $\beta = 0.75$, $\gamma = 0.36$,

(c) $n = 15$, $p = -2.97 - 0.01i$, $A = [-1.2, 1.2]^2$, $K = 50$, $g(z) = (0.05 - 0.05i)z + 0.01 - 0.01i$, $m = 0.05$, $q = 0.51$, $\rho = 0.54$, $\mu = 1.42$, $\alpha = 0.15$, $\beta = 0.65$, $\gamma = 0.59$.

5. Conclusions

We conclude from our work that the viscosity approximation type method considered by Nandal et al. [32] has the capability of generating fascinating and attracting graphics of fractals (Julia and Mandelbrot sets), which proves

the applicability of the proposed method. We have derived a result to obtain an escape criterion for the generation of these fractals using the proposed iterative method. Some tempting graphics of fractals have been generated by choosing different values of polynomials, contraction mappings, the resolvent operators associated with monotone operators and parameters α, β, γ. We have noticed that the parameters have a great impact not only on the shape, but also on the symmetry of the generated set.

Due to their attractive nature in the field of design [17,42], we believe that the results of this research might be very useful for those who are interested in creating nice looking graphics and designer printing patterns. Moreover, the results of this paper might be used to expand the space for the initial keys used in image encryption [14].

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

Declarations

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

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

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, page Article 51, New York, NY, USA, (2018). Association for Computing Machinery

[2] Al Sideiri, A., Alzeidi, N., Al Hammoshi, M., Chauhan, M.S., Al, Farsi G.: CUDA implementation of fractal image compression. J. Real-Time Image Process. **17**(5), 1375–1387 (2020)

[3] Ashish, M., Rani, R.: Chugn: Julia sets and Mandelbrot sets in Noor orbit. Appl. Math. Comput. **228**, 615–631 (2014)

[4] Barnsley, M.F.: Fractals Everywhere, 3rd edn. Dover Publications, New York (2012)

[5] Chauhan, Y.S., Rana, R., Negi, A.: New Julia sets of Ishikawa iterates. Int. J. Comput. Appl. **7**(13), 34–42 (2010)

[6] Costanzo, S., Venneri, F.: Polarization-insensitive fractal metamaterial surface for energy harvesting in IoT applications. Electronics **9**(6), 959 (2020)

[7] Dang, Y., Kauffman, L.H., Sandin, D.: Hypercomplex Iterations: Distance Estimation and Higher Dimensional Fractals. World Scientific, Singapore (2002)

[8] Devaney, R.L.: A First Course in Chaotic Dynamical Systems: Theory and Experiment, 2nd edn. CRC Press, Boca Raton (2020)

[9] Di Ieva, A., Grizzi, F., Jelinek, H., Pellionisz, A.J., Losa, G.A.: Fractals in the neurosciences, part I: General principles and basic neurosciences. Neuroscientist **20**(4), 403–417 (2014)

[10] 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)

[11] Genel, A., Lindenstrauss, J.: An example concerning fixed points. Israel J. Math. **22**(1), 81–86 (1975)

[12] Halpern, B.: Fixed points of nonexpanding maps. Bull. Am. Math. Soc. **73**, 957–961 (1967)

[13] Harris, J.: Fractal Architecture: Organic Design Philosophy in Theory and Practice. University of New Mexico Press, Albuquerque, New Mexico, USA (2012)

[14] Hasanzadeh, E., Yaghoobi, M.: A novel color image encryption algorithm based on substitution box and hyper-chaotic system with fractal keys. Multimedia Tools Appl. **79**(11–12), 7279–7297 (2020)

[15] 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)

[16] Julia, G.: Mémoire sur l'itération des fonctions rationnelles. Journal de Mathématiques Pures et Appliquées **8**(1), 47–246 (1918)

[17] Kharbanda, M., Bajaj, N.: An exploration of fractal art in fashion design. In: 2013 International Conference on Communication and Signal Processing, pp. 226–230. IEEE (2013)

[18] Krzysztofik, W.J.: Fractals in antennas and metamaterials applications. In: F. Brambila (ed) Fractal Analysis – Applications in Physics, Engineering and Technology, 953–978. IntechOpen (2017)

[19] Kumari, S., Chugh, R.: A novel four-step feedback procedure for rapid control of chaotic behavior of the logistic map and unstable traffic on the road. Chaos **30**, 123115 (2020)

[20] Kumari, S., Chugh, R.: Novel fractals of Hutchinson Barnsley operator in Hausdorff g-metric spaces. Poincare J. Anal. Appl. **7**(1), 99–117 (2020)

[21] Kumari, S., Chugh, R., Cao, J., Huang, C.: Multi fractals of generalized multivalued iterated function systems in b-metric spaces with applications. Mathematics **7**(10), 967 (2019)

[22] Kumari, S., Chugh, R., Cao, J., Huang, C.: On the construction, properties and Hausdorff dimension of random Cantor one p-th set. AIMS Math. **5**(4), 3138–3155 (2020)

[23] 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)

[24] 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)

[25] Kumari, S., Kumari, M., Chugh, R.: Graphics for complex polynomials in Jungck-SP orbit. IAENG Int. J. Appl. Math. **49**(4), 568–576 (2019)

[26] Kwun, Y.C., Tanveer, M., Nazeer, W., Gdawiec, K., Kang, S.M.: Mandelbrot and Julia sets via Jungck-CR iteration with s-convexity. IEEE Access **7**, 12167–12176 (2019)

[27] Mandelbrot, B.B.: The Fractal Geometry of Nature. W.H. Freeman and Company, San Francisco (1982)

[28] Mann, W.R.: Mean value methods in iteration. Proc. Am. Math. Soc. **4**(3), 506–510 (1953)

[29] Moudafi, A.: Viscosity approximation methods for fixed-points problems. J. Math. Anal. Appl. **241**(1), 46–55 (2000)

[30] Nandal, A., Chugh, R.: On zeros of accretive operators with application to the convex feasibility problem. Scientific Bull. Ser. A Appl. Math. Phys. **81**(3), 95–106 (2019)

[31] 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)

[32] Nandal, A., Chugh, R., Postolache, M.: Iteration process for fixed point problems and zeros of maximal monotone operators. Symmetry **11**(5), 655 (2019)

[33] Ouyang, P., Chung, K.W., Nicolas, A., Gdawiec, K.: Self-similar fractal drawings inspired by M.C. Escher's print Square Limit. ACM Trans. Gr. **40**(3):Article 31 (2021)

[34] Parise, P.O., Rochon, D.: A study of dynamics of the tricomplex polynomial $\eta^h + c$. Nonlinear Dyn. **82**(1–2), 157–171 (2015)

[35] 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)

[36] Rani, M., Kumar, V.: Superior Julia sets. J. Korea Soc. Math. Educ. Ser. D Res. Math. Educ. **8**(4), 261–277 (2004)

[37] Rani, M., Kumar, V.: Superior Mandelbrot set. J. Korea Soc. Math. Educ. Ser. D Res. Math. Educ. **8**(4), 279–291 (2004)

[38] Reich, S.: Strong convergence theorems for resolvents of accretive operators in Banach spaces. J. Math. Anal. Appl. **75**(1), 287–292 (1980)

[39] Salawu, S.A., Sobamowo, M.G., Sadiq, O.M.: Dynamic analysis of non-homogenous varying thickness rectangular plates resting on Pasternak and Winkler foundations. Eng. Appl. Sci. Lett. **3**(1), 1–20 (2020)

[40] Shahid, A.A., Nazeer, W., Gdawiec, K.: The Picard-Mann iteration with s-convexity in the generation of Mandelbrot and Julia sets. Monatshefte für Mathematik **195**(4), 565–584 (2021)

[41] 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)

[42] Toeters, M., Feijs, L., van Loenhout, D., Tieleman, C., Virtala, N., Jaakson, G.K.: Algorithmic fashion aesthetics: Mandelbrot. In: Proceedings of the 23rd International Symposium on Wearable Computers, ISWC '19, pp. 322–328, New York, NY, USA, (2019). Association for Computing Machinery

[43] Wang, X.Y., Song, W.J.: The generalized M-J sets for bicomplex numbers. Nonlinear Dyn. **72**(1–2), 17–26 (2013)

[44] 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)

[45] Zou, C., Shahid, A.A., Tassaddiq, A., Khan, A., Ahmad, M.: Mandelbrot sets and Julia sets in Picard-Mann orbit. IEEE Access **8**, 64411–64421 (2020)

Sudesh Kumari
Government College for Girls Sector 14
Gurugram 122001
India
e-mail: tanwarsudesh10@gmail.com

Krzysztof Gdawiec
Institute of Computer Science
University of Silesia
Bedzinska 39
41-200 Sosnowiec
Poland
e-mail: krzysztof.gdawiec@us.edu.pl

Ashish Nandal
Government College Baund Kalan
Charkhidadri 127025
India
e-mail: ashishnandalmdu@gmail.com

Naresh Kumar
Department of Mathematics
Guru Jambheshwar University of Science & Technology
Hisar 125001
India
e-mail: nks280@gmail.com

Renu Chugh
Department of Mathematics
Maharshi Dayanand University
Rohtak 124001
India
e-mail: chugh.r1@gmail.com