



Geometric application of a viscosity approximation-type iterative method to the generation the fractals as Julia and Mandelbrot Sets for complex function

Iqbal Ahmad^{a*}, Mohammad Sajid^b, Osama Abdullah Al-Bosaili^c, Mohammed Dakhilallah Alharbi^d

^{a,b}Department of Mechanical Engineering, College of Engineering, Qassim University, Saudi Arabia; ^cDepartment of Civil Engineering, College of Engineering, Qassim University, Saudi Arabia; ^dDepartment of Electrical Engineering, College of Engineering, Qassim University, Saudi Arabia

Abstract

This work explores an application of novel fractal patterns, specifically Julia and Mandelbrot sets, generated by a modified class of complex function in which the traditional constant term is replaced with a logarithmic function. Utilizing a viscosity approximation-type iterative method, we develop escape criteria that enhance existing algorithms, thereby enabling the precise visualization of intricate fractal structures as Julia and Mandelbrot sets. Numerical experiments in MATLAB reveal that varying the input parameters induces significant dynamic transformations in the fractals' morphology. We believe that the insights gained from this study will inspire and motivate researchers and enthusiasts with a deep interest in fractal geometry.

Mathematics Subject Classification. 28A80, 37F10, 39B12, 47H10

Key words and phrases. Algorithms, Escape criteria, Iterative methods, Julia sets, Mandelbrot sets, AMS Subject.

1. Introduction

Fractals are infinitely complex mathematical patterns that exhibit self-similarity, meaning their intricate shapes repeat at different scales. These mesmerizing structures appear throughout nature,

Email addresses: i.ahmad@qu.edu.sa (Iqbal Ahmad); msajid@qu.edu.sa (Mohammad Sajid); 431108028@qu.edu.sa (Osama Abdullah Al-Bosaili); 431107901@qu.edu.sa (Mohammed Dakhilallah Alharbi)

from the branching patterns of trees and river networks to the delicate formations of snowflakes and the spiraling arrangements of fern leaves. Mathematically, fractals are generated through iterative methods, where a simple equation or set of rules is applied repeatedly, with each iteration building upon the last to produce increasingly detailed and complex patterns. For more details, refer to [13, 14, 16]. This process of repeated refinement transforms basic mathematical operations into stunningly elaborate forms that mirror the organic beauty found in the natural world.

This comprehensive approach deepens our understanding of the fundamental principles that govern natural phenomena through fractal mathematics. Fractals, especially the Julia and Mandelbrot sets, have fascinated mathematicians for nearly a century, beginning with Gaston Julia's early 20th-century work on Julia sets [7]. These intricate structures are generated using fixed-point iterative methods like Mann and Picard iteration, along with other advanced techniques, to explore their complex dynamics (see [3, 5, 10, 17–19]). These iterative approaches have proven particularly valuable in analyzing the behavior of polynomials, complex trigonometric functions, and transcendental functions, demonstrating how the choice of the iteration method can dramatically influence a fractal's visual characteristics—affecting its shape, coloration, and structural complexity even when applied to identical base functions (see [20, 21]). Beyond classical Julia sets, these iterative frameworks serve as the mathematical foundation for creating various other fractal types, including intricate biomorphs, iterated function system fractals, inversion fractals, and root-finding fractals (see [6, 8]), showcasing the remarkable versatility of iterative methods in fractal geometry.

The study of iterative techniques in fractal generation and analysis has seen significant theoretical advancements through key developments in viscosity approximation methods. In 2000, Moudafi's seminal work [12] established crucial convergence properties of the viscosity method for semi-non-expansive mappings, providing a rigorous mathematical framework that has profoundly influenced iterative approaches to fractal construction. These theoretical breakthroughs have enabled deeper analysis of fractal structures and their convergence behavior. Building upon this foundation, Adhikari et al. [1], Babar et al. [4], Nandal et al. [15] and Kumari et al. [9] recently extended these concepts by developing generalized viscosity approximation-type iterative methods within Hilbert spaces, marking an important expansion of both the theoretical underpinnings and practical implementations of iterative techniques. These methodological innovations have substantially enriched fractal mathematics, offering more sophisticated tools for investigating the complex dynamics of fractal generation while simultaneously broadening their potential applications across mathematical and computational domains. The progressive refinement of these iterative approaches continues to enhance our capacity to analyze and generate increasingly complex fractal structures with greater precision and mathematical rigor.

Recent studies by Tanveer et al. [20] and Iqbal et al. [2] have advanced fractal generation by using generalized viscosity approximation methods to produce Mandelbrot and Julia sets, improving the visualization of their complex dynamics. Building on this, our research extends these techniques to derive escape criteria for the complex function $W(z) = \Psi z^n + \log \alpha^p$, where $n \geq 2$, $\Psi \in \mathbb{C}$, $\alpha \in \mathbb{C} \setminus \{0\}$ and $p \in \mathbb{R}$, $p \geq 1$.

This paper is systematically organized into five sections to present a comprehensive study of fractal generation using viscosity approximation methods. Section 2 establishes the fundamental mathematical framework by introducing essential definitions, concepts, and preliminary results necessary for our analysis. In Section 3, we develop the theoretical core of our work, presenting key theorems that establish a generalized escape criterion a critical component for constructing both Julia and Mandelbrot sets through our viscosity approximation-type iterative approach. The practical implementation is presented in Section 4, where we detail the computational algorithms and showcase corresponding visualizations of Julia and Mandelbrot sets for varying parameters, all generated using MATLAB R2024a (version 24.1.0.2537033, 64-bit) on a standard HP laptop featuring an Intel(R) Core(TM) i7-14700HX (2.10 GHz) processor and 32 GB of RAM. Section 5 presents numerical results that illustrate a clear variation in the shapes and sizes of the fractal sets generated by the

proposed iterative method. Finally, Section 6 concludes the study by summarizing the main findings and contributions.

2. Preliminaries

Definition 2.1. (Julia set [7]). For a complex function $\mathcal{U} : \mathbb{C} \rightarrow \mathbb{C}$ the filled Julia set $F_{\mathcal{U}}$ is defined as:

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

where $\mathcal{U}^j$ denotes the j^{th} iterate of $\mathcal{U}$. The boundary of $F_{\mathcal{U}}$ is called the Julia set.

Definition 2.2. (Mandelbrot set [7]). For a complex function $\mathcal{U} : \mathbb{C} \rightarrow \mathbb{C}$, the Mandelbrot set M is defined as:

$$M = \{\alpha \in \mathbb{C} : F_{\alpha} \text{ is connected}\},$$

where F_{α} denotes the filled Julia set associated with E_{α} equivalently,

$$M = \{\alpha \in \mathbb{C} : |F_{\alpha}^j(0)| \rightarrow \infty \text{ as } j \rightarrow \infty\}.$$

Definition 2.3. ([12]). A sequence $\{z_j\} \subseteq \mathbb{C}$ with initial point $z_0 \in \mathbb{C}$ is referred to as the viscosity approximation method if

$$z_{j+1} = \vartheta_j h(z_j) + (1 - \vartheta_j) \mathcal{U}(z_j), \quad j \in \mathbb{N} \cup \{0\}, \quad (2.1)$$

where $\vartheta_j \in (0, 1)$, $h, \mathcal{U} : \mathbb{C} \rightarrow \mathbb{C}$ and h is a contraction mapping.

Mainge [11] proposed the following viscosity approximation method which is a special variant of (2.1): starting with an arbitrary initial point $z_0 \in \mathbb{C}$, z_j generated by

$$z_{j+1} = \vartheta_j h(z_j) + (1 - \vartheta_j) \mathcal{U}_{\varsigma_j}(z_j), \quad j \in \mathbb{N} \cup \{0\}, \quad (2.2)$$

where $\mathcal{U}_{\varsigma_j} = \varsigma_j I + (1 - \varsigma_j) \mathcal{U}$, with quasi-nonexpansive mapping $\mathcal{U}$ ([15]), and $\vartheta_j, \varsigma_j \in (0, 1)$. For simplicity, throughout the manuscript we consider $\vartheta_j = \vartheta$ and $\varsigma_j = \varsigma$, where $\vartheta, \varsigma \in (0, 1)$. Consider the complex function $\mathcal{U} : \mathbb{C} \rightarrow \mathbb{C}$ defined by:

$$\mathcal{U}(z) = \Psi z^n + \log \alpha^p, \quad (2.3)$$

where $n \geq 2$, $\Psi \in \mathbb{C}$, $\alpha \in \mathbb{C} \setminus \{0\}$ and $p \in \mathbb{R}$, $p \geq 1$. Additionally, assume $h(z) = az + b$ is a contraction mapping ($a, b \in \mathbb{C}$, $|a| < 1$).

Throughout the article, we assume $\psi = \frac{\log(\alpha^p)}{\alpha}$, giving the relation $\log(\alpha^p) = \psi\alpha$.

3. Escape criteria for Viscosity Mandelbrot and Julia sets

In the complex space, we study the viscosity approximation-type orbit z_j given by Mainge [11]:

$$\begin{cases} z_{j+1} = \vartheta h(z_j) + (1 - \vartheta) y_j, \\ y_j = \varsigma z_j + (1 - \varsigma) \mathcal{U}(z_j), \end{cases} \quad (3.1)$$

where z_0 is starting point, $\mathcal{U}$ is complex valued function, h is a contraction mapping and $\vartheta, \varsigma \in (0, 1)$ are parameters.

Definition 3.1. A mapping $h : \mathbb{C} \rightarrow \mathbb{C}$ is called a Banach contraction mapping if there exists $\tau \in (0, 1)$ such that

$$|h(z_1) - h(z_2)| \leq \tau |z_1 - z_2|, \text{ for all } z_1, z_2 \in \mathbb{C}.$$

Definition 3.2. (Viscosity Julia set [7]). *For the operator $\mathcal{U}$ in (2.1), the viscosity filled Julia set $F_{\mathcal{U}}$ consists of all $z \in \mathbb{C}$ where the orbit $\{z_j\}$ from (2.2) remains bounded. The boundary of the filled Julia set, $F_{\mathcal{U}}$ is referred to as the viscosity Julia set.*

Definition 3.3. (Viscosity Mandelbrot set [7]). *For the complex operator $\mathcal{U}$ in (2.2), the viscosity Mandelbrot set M is*

$$M = \{\alpha \in \mathbb{C} : |z_j| \nrightarrow \infty \text{ as } j \rightarrow \infty\},$$

where $z_0 = 0$ and z_j for $j > 0$ is defined by (3.1) with $\mathcal{U}$ as the complex operator.

In the literature, several methods have been employed to construct and analyze fractals, including the escape time algorithm, potential function, iterated function systems, and distance estimator algorithms. Among these, the escape time algorithm is one of the most significant techniques for fractal generation. The escape criterion serves as a condition to determine whether the orbit of an initial point diverges to infinity. In this work, we establish a general escape criterion applicable to the generation of viscosity Julia and Mandelbrot sets.

Theorem 3.1. *Assume that $|z_0| \geq \max\{|\alpha|, |b|\} > \left(\frac{(2+|\alpha|+(1-\vartheta)(1-\varsigma|\psi|)}{\varsigma(1-\vartheta)|\Psi|} \right)^{\frac{1}{n-1}}$, where $n \geq 2$, $\Psi \in \mathbb{C}$, $\alpha \in \mathbb{C} \setminus \{0\}$ and $p \in \mathbb{R}$, $p \geq 1$, and $h(z) = az + b$ with $a, b \in \mathbb{C}$ and $|\alpha| < 1$. Let the sequence $\{z_j\}$ be defined as*

$$\begin{cases} Z_{j+1} = \vartheta h(z_j) + (1-\vartheta)y_j; \\ y_j = \varsigma z_j + (1-\varsigma)\mathcal{U}(z_j). \end{cases} \quad (3.2)$$

Then $|z_j| \rightarrow \infty$ as $j \rightarrow \infty$.

Proof. From (3.2), we have

$$|y_j| = |\varsigma z_j + (1-\varsigma)\mathcal{U}(z_j)|, j \geq 0.$$

For $j = 0$, we have

$$\begin{aligned} |y_0| &= |(1-\varsigma)z_0 + \varsigma(\Psi z_0^n + \log \alpha^p)| \\ &= |(1-\varsigma)z_0 + \varsigma(\Psi z_0^n + \psi \alpha)| \\ &\geq |\varsigma(\Psi z_0^n + \psi \alpha)| - |(1-\varsigma)z_0| \\ &\geq \varsigma|\Psi z_0^n| - \varsigma|\psi \alpha| - (1-\varsigma)|z_0| \\ &\geq \varsigma|\Psi||z_0^n| - \varsigma|\psi||\alpha| - |z_0| + \varsigma|z_0| \\ &\geq \varsigma|\Psi||z_0^n| - \varsigma|\psi||\alpha| - |z_0|. \end{aligned}$$

Our assumption $|z_0| \geq \max\{|\alpha|, |b|\}$ yields that $-\alpha \geq -|z_0|$, we obtain

$$\begin{aligned} |y_0| &\geq \varsigma|\Psi||z_0^n| - \varsigma|\psi||z_0| - |z_0| \\ &\geq |z_0| \varsigma|\Psi||z_0^{n-1}| - (1+\varsigma|\psi|) \end{aligned}$$

thus,

$$|y_0| \geq |z_0| \varsigma|\Psi||z_0^{n-1}| - (1+\varsigma|\psi|). \quad (3.3)$$

From (3.2), consider

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

Our assumption $|z_0| \geq \max\{|\alpha|, |b|\}$ yields that $-|b| \geq -|z_0|$, and using (3.3), we have

$$\begin{aligned} |z_1| &\geq (1 - \vartheta) |z_0| (\varsigma |\Psi| |z_0^{n-1}| - (1 + \varsigma |\psi|)) - \vartheta |\alpha| |z_0| - \vartheta |z_0| \\ &\geq |z_0| (\varsigma (1 - \vartheta) |\Psi| |z_0^{n-1}| - (1 - \vartheta)(1 + \varsigma |\psi|)) - \vartheta (1 + |\alpha|) |z_0|, \quad \because \vartheta \in (0, 1) \\ &\geq |z_0| (\varsigma (1 - \vartheta) |\Psi| |z_0^{n-1}| - (1 - \vartheta)(1 + \varsigma |\psi|)) - (1 + |\alpha|) |z_0| \\ &\geq |z_0| (\varsigma (1 - \vartheta) |\Psi| |z_0^{n-1}| - (1 - \vartheta)(1 + \varsigma |\psi|) - (1 + |\alpha|)). \end{aligned} \quad (3.4)$$

Our assumption $|z_0| > \left(\frac{(2 + |\alpha| + (1 - \vartheta)(1 - \varsigma |\psi|))}{\varsigma (1 - \vartheta) |\Psi|} \right)^{\frac{1}{n-1}}$ gives

$$\varsigma (1 - \vartheta) |\Psi| |z_0^{n-1}| - (1 - \vartheta)(1 + \varsigma |\psi|) - (1 + |\alpha|) > 1. \quad (3.5)$$

Thus, there exists $\Omega > 0$ such that

$$\varsigma (1 - \vartheta) |\Psi| |z_0^{n-1}| - (1 - \vartheta)(1 + \varsigma |\psi|) - (1 + |\alpha|) > 1 + \Omega > 1.$$

From (3.4), we obtain

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

In particular $|z_1| > |z_0|$. Continuing this procedure, we obtain

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

Hence, $|z_j| \rightarrow \infty$, as $j \rightarrow \infty$.

In the proof of Theorem 3.1, We just made use of the fact that $|z_0| \geq \max\{|\alpha|, |b|\}$ and

$|z_0| \geq \left(\frac{(2 + |\alpha| + (1 - \vartheta)(1 - \varsigma |\psi|))}{\varsigma (1 - \vartheta) |\Psi|} \right)^{\frac{1}{n-1}}$. So, we can refine it and get the following result.

Corollary 3.1. Assume $|z_0| \geq \max \left\{ |\alpha|, |b|, \left(\frac{(2 + |\alpha| + (1 - \vartheta)(1 - \varsigma |\psi|))}{\varsigma (1 - \vartheta) \Psi} \right)^{\frac{1}{n-1}} \right\}$, where $\alpha, b, \Psi, \alpha \in \mathbb{C}$ ($\alpha \neq 0$, $|\alpha| < 1$), $p \geq 1$, then $\lim_{j \rightarrow \infty} |z_j| = \infty$.

Corollary 3.2. Let $|z_k| \geq \max \left\{ |\alpha|, |b|, \left(\frac{(2 + |\alpha| + (1 - \vartheta)(1 - \varsigma |\psi|))}{\varsigma (1 - \vartheta) \Psi} \right)^{\frac{1}{n-1}} \right\}$, for some $j \geq 0$, $\alpha, b, \Psi, \alpha \in \mathbb{C}$

($\alpha \neq 0$, $|\alpha| < 1$), $p \geq 1$, and $\vartheta, \varsigma \in (0, 1)$. Then there exists $\Omega > 0$ such that $|z_{j+k}| > (1 + \Omega)^j |z_k|$, and we

have $\lim_{j \rightarrow \infty} |z_j| = \infty$.

4. Graphical Examples

By leveraging Corollaries 3.1 and 3.2, we generate viscosity-filled Julia and Mandelbrot sets for the function $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$, where $n \geq 2$, $a, b, \Psi, \alpha \in \mathbb{C}$ ($\alpha \neq 0$, $|\alpha| < 1$), $p \geq 1$, using the proposed viscosity approximation iterative method. Based on these corollaries, if for some $j \geq 0$, the point z_j lies outside the circle of radius

$$R \geq \max \left\{ |\alpha|, |b|, \left(\frac{(2 + |\alpha| + (1 - \vartheta)(1 - \varsigma |\psi|))}{\varsigma (1 - \vartheta) \Psi} \right)^{\frac{1}{n-1}} \right\},$$

then the orbit of $|z_0|$ escapes to infinity, and consequently, z_0 does not belong to the viscosity-filled Julia set. Conversely, if z_j remains within the circle of radius R , the point z_0 stays in a bounded region and thus belongs to the viscosity-filled Julia set. Algorithm 1 outlines the pseudocode for the escape-time algorithm used to generate the viscosity-filled Julia set. In this algorithm, the viscosity-filled Julia set are generated within a specified region $A \subset \mathbb{C}$ of the complex plane, utilizing a chosen color map. Algorithm 2 summarizes the escape-time algorithm for the function $\mathcal{U}(z)$ and its iterations. Here, each point α in the region A is treated as the constant in $\mathcal{U}(z)$. The function $\mathcal{U}(z)$ is iterated using the selected iterative method, and it is checked whether the point computed in the current iteration exceeds the escape threshold R . Upon completing the iteration process, a color from the selected color map is assigned to α . These algorithms provide a systematic approach to visualizing the viscosity-filled Julia and Mandelbrot sets for the given function, offering insights into the behavior of the iterative process within the complex plane. We generate and visualize viscosity-filled Julia and Mandelbrot sets for various parameter values, exploring the effects of modifying these parameters. To avoid infinite loops, a maximum iteration limit K is imposed. Utilizing MATLAB R2024a (version 24.1.0.2537033, 64-bit) on a standard HP laptop featuring an Intel(R) Core(TM) i7-14700HX (2.10 GHz) processor and 32 GB of RAM, we visualize non-classical Julia and Mandelbrot sets, analyzing their structural variations under different parameters to efficiently capture fractal patterns dynamics while maintaining computational feasibility. Throughout the paper, a maximum number of iterations $K = 70$ is consistently applied.

4.1. Viscosity Julia sets generation for $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$.

This section presents viscosity Julia sets generated from the complex function (2.3) under different parameter configurations.

In the first example, we generate Julia sets for $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$ using Algorithm 1. The resulting images are organized into two distinct categories based on the values of p : the first explores integer values of p , i.e., $t = 1, 7, 10$ while the second examines noninteger values i.e., $t = 1.75, 4.5, 11.5$, and $A = [-1.75, 1.75]^2$. This reveals distinct structural patterns, emphasizing the critical role of p in shaping Julia set morphology. Note that the last column in all the tables displays the image execution time (in short, IET) in seconds.

Figure 2A–F illustrate the morphology of Julia sets for a spectrum of values of the parameter p , with subfigures (A–C) and (D–F) dedicated to integer and non-integer values, respectively. The profound influence of p on the set's geometry is immediately evident, as even minor variations precipitate significant structural changes. For instance, the fractal in Figure 2A displays central symmetry, composed of two primary lobes connected by a narrow neck and encapsulated within a smoothed, rounded rectangle. Its pattern is further characterized by six distinct clusters of points, arranged symmetrically with three around each lobe. In contrast, Figure 2A,D initially suggest a fourfold symmetry, while the sets in panels (b) and (c) exhibit axial symmetry exclusively along the real axis. These fractals, such as those in Figure 2B–F, often feature a red central shape with four primary extensions, each culminating in intricate, self-similar patterns, all set against a gradient blue background. A consistent trend emerges upon closer inspection: as the value of p increases, the scale of the bulb-like structures within the Julia sets progressively diminishes. The image generation time for each iteration is also recorded.



Figure 1: A color map is used in the examples.

Algorithm 1. Viscosity Julia set generation for $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$.

Input: $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$, where $n \geq 2$, $\alpha, b, \Psi \in \mathbb{C}$, $\alpha \in \mathbb{C} \setminus \{0\}$, $p \geq 1$,
 $A \subset \mathbb{C}$ -area in which we draw the set; K -maximal number of iterations;
 $\vartheta, \varsigma \in (0, 1)$ and $h(z) = az + b$, colourmap $[0..C-1]$ -color with C colors.

Output: Julia set for area A

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

$$\psi = \frac{\log(\alpha^p)}{\alpha}$$

$$R = \max \left\{ |\alpha|, |b|, \left(\frac{(2 + |a| + (1 - \vartheta)(1 - \varsigma|\psi|))}{\varsigma(1 - \vartheta)\Psi} \right)^{\frac{1}{n-1}} \right\},$$

$j = 0$

while $j \leq K$ **do**

$$z_{j+1} = \vartheta h(z_j) + (1 - \vartheta)y_j$$

$$y_j = \varsigma z_j + (1 - \varsigma)\mathcal{U}(z_j), \text{ where } 0 < \vartheta, \varsigma < 1$$

if $|z_{j+1}| \geq R$ then break end if

$j = j + 1$

$$J = \left\lfloor (C-1) \frac{j}{K} \right\rfloor$$

colour z_0 with colourmap $[J]$

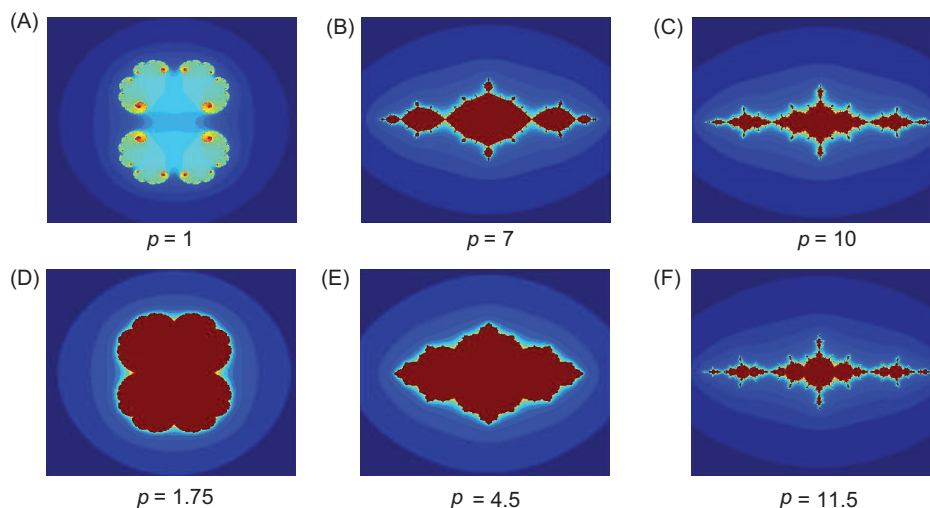
Table 1: Effect of the parameter p while all other parameters are fixed.

	n	p	Ψ	α	ϑ	ς	a	b	IET
(a)	2	1	1.01i	0.5	0.5	0.6	0.4	1	0.68s
(b)	2	7	1.01i	0.5	0.5	0.6	0.4	1	0.76s
(c)	2	10	1.01i	0.5	0.5	0.6	0.4	1	0.88s
(d)	2	1.75	1.01i	0.5	0.5	0.6	0.4	1	0.94s
(e)	2	4.5	1.01i	0.5	0.5	0.6	0.4	1	1.02s
(f)	2	11.5	1.01i	0.5	0.5	0.6	0.4	1	1.17s

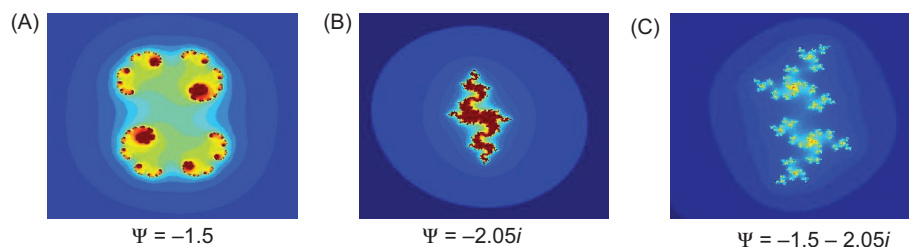
In the second example, we generate Julia sets for $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$ using Algorithm 1. The resulting images are organized into three distinct categories based on the values of Ψ : the first explores real value of Ψ , i.e., $\Psi = -1.5$ the second explores purely imaginary value of Ψ , i.e., $\Psi = -2.05i$ while the third examines complex value Ψ i.e., $\Psi = -1.05 - 2.05i$, and $A = [-1.75, 1.75]^2$. This reveals distinct structural patterns, emphasizing the critical role of Ψ in shaping Julia set morphology. Each figure isolates one varying parameter while holding others constant.

The viscosity Julia sets in Figures 3 & 4A–C demonstrate the influence of individual parameter variations Ψ and α on fractal geometry when one parameter is altered while others remain fixed. In these figures, subplots (a), (b), and (c) correspond to distinct parameter configurations: purely real, purely imaginary, and complex values, respectively. For instance, Figure 3 highlights variations in Ψ , while Figures 3 and 4 focus on Ψ or α , in turn. The results reveal that shifts in these parameters significantly affect the shape, size, and color gradients of the sets, particularly near the edges of their leaf-like structures. Notably, the arms of the fractals exhibit non-uniform morphological changes, with asymmetry in local features, yet the overall sets retain axial symmetry.

Figures 5 and 6 A–C illustrate the impact of individual parameter variations on the geometry of viscosity Julia sets. By altering either parameter a or b while keeping others fixed, distinct morphological changes are observed. Each row of subplots—(a) for purely real, (b) for purely imaginary, and

Figure 2: Viscosity Julia sets for the values of p generated via Algorithm 1.Table 2: Effect of the parameter Ψ while all other parameters are fixed.

	n	p	Ψ	α	ϑ	s	a	b	IET
(i)	2	4	-1.5	$0.07 + 0.87i$	0.5	0.5	$-0.85 - 0.05i$	$-1.05 - 1.5i$	1.08s
(ii)	2	4	$-2.05i$	$0.07 + 0.87i$	0.5	0.5	$-0.85 - 0.05i$	$-1.05 - 1.5i$	1.19s
(iii)	2	4	$-1.05 - 2.05i$	$0.07 + 0.87i$	0.5	0.5	$-0.85 - 0.05i$	$-1.05 - 1.5i$	1.37s

Figure 3: Viscosity Julia sets generated for different Ψ values via Algorithm 1.Table 3: Effect of the parameter α while all other parameters are fixed.

	n	p	Ψ	α	ϑ	s	a	b	IET
(i)	2	4	1	0.75	0.5	0.5	$0.5 + 0.75i$	$1.05 + 1.5i$	1.28s
(ii)	2	4	1	0.75i	0.5	0.5	$-0.85 - 0.05i$	$-1.05 - 1.5i$	1.41s
(iii)	2	4	1	$0.07 + 0.87i$	0.5	0.5	$-0.85 - 0.05i$	$-1.05 - 1.5i$	1.52s

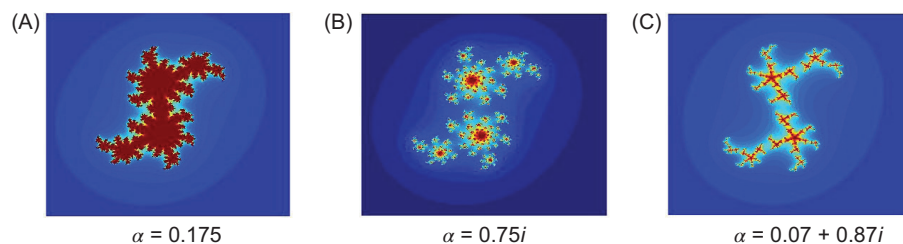
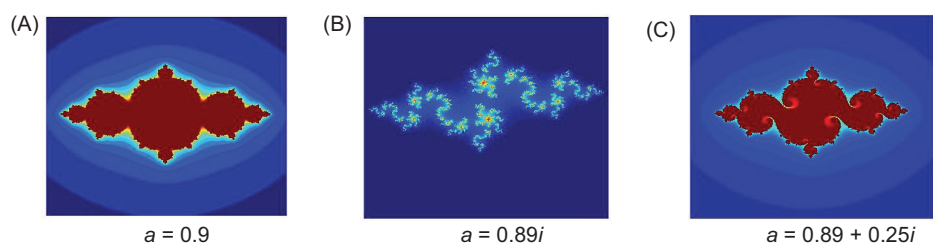
Figure 4: Viscosity Julia sets generated for different α values via Algorithm 1.

Table 4: Effect of the parameter a while all other parameters are fixed.

	n	p	Ψ	α	ϑ	ς	a	b	IET
(i)	2	4	1.01	0.3	0.5	0.5	0.89	2.4	1.38s
(ii)	2	4	1.01	0.3	0.5	0.5	$0.89i$	2.4	1.59s
(iii)	2	4	1.01	0.3	0.5	0.5	$0.89 + 0.25i$	2.4	1.71s

Figure 5: Viscosity Julia sets generated for different a values via Algorithm 1.

(c) for complex values—showcases these effects. The analysis reveals that parameter shifts notably alter the shape, size, and color gradients, especially near the delicate edges of the structures. A key finding is that although the fractal arms exhibit asymmetric, non-uniform changes at a local level, the global structure of the sets preserves axial symmetry.

4.2. Mandelbrot sets generation for $U(z) = \Psi z^n + \log \alpha^p$.

This subsection showcases the viscosity Julia sets corresponding to complex function (2.3) for various input values.

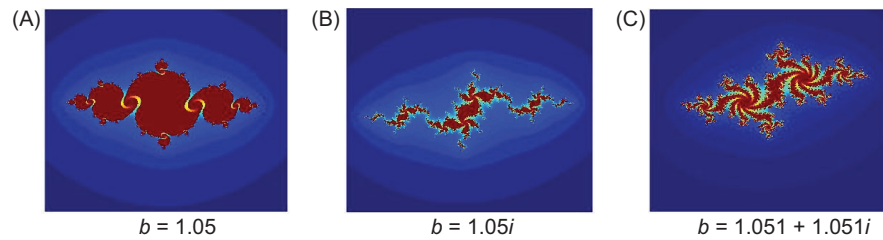
In the first example, using Algorithm 2 with fixed parameters $n = 2$, $\Psi = -1.1i$, $\alpha = 0.5$, $a = 0.41$, $b = 1$, $\vartheta = 0.5$, $\varsigma = 0.6$ and $A = [-1.5, 1.5]^2$. The images are divided into two groups, corresponding to two different cases considered in this example. In the first case, we explore the behavior of the Mandelbrot sets using integer values of p , while in the second case, we examine the impact of non-integer values of p on the structure and dynamics of the sets.

In Figures 7A–C, we observe multiple copies of the same Mandelbrot set, where the number of copies corresponds to the value of p . For instance, when $p = 1$, there is one copy; for, $p = 3$, three copies, for $p = 6$, six copies, for $p = 11$, eleven copies, and so on. Each pattern exhibits a p -fold symmetry, creating an intricate rosette-like design. In each image, we observe multiple copies of the same Mandelbrot set, with the number of sets corresponding to the value of p . Additionally, each pattern exhibits a p -fold symmetry, resembling a rosette-like structure.

Figures 8A–C illustrate more complex behavior for non-integer values of 8 (a , b , c), where the set comprises smaller copies of Mandelbrot sets. As p increases, these sets diminish in size but occupy more space, eventually leading to overlapping among the smaller copies. With further reduction in the decimal value of p , the overlapping intensifies, and the shapes of the smaller copies begin to

Table 5: Effect of the parameter b while all other parameters are fixed.

	n	p	Ψ	α	ϑ	ς	a	b	IET
(i)	2	4	1.01	0.3	0.5	0.5	$0.75 + 0.25i$	1.05	1.43s
(ii)	2	4	1.01	0.3	0.5	0.5	$0.75 + 0.25i$	$1.05i$	1.57s
(iii)	2	4	1.01	0.3	0.5	0.5	$0.75 + 0.25i$	$1.05 + 1.5i$	1.67s

Figure 6: Viscosity Julia sets generated for different b values via Algorithm 1.

Algorithm 2. Viscosity Mandelbrot set generation for $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$.

Input: $\mathcal{U}(z) = \Psi z^n + \log \alpha^p$, where $n \geq 2$, $\alpha, b, \Psi \in \mathbb{C}$, $\alpha \in \mathbb{C} \setminus \{0\}$, $p \geq 1$,
 $A \subset \mathbb{C}$ -area in which we draw the set; K -maximal number of iterations;
 $\vartheta, \varsigma \in (0, 1)$ and $h(z) = az + b$, colourmap $[0..C-1]$ -color with C colors.

Output: Viscosity Mandelbrot set for area A
 if $\alpha = 0$ then discard the point

$$\psi = \frac{\log(\alpha^p)}{\alpha}$$

$$R = \max \left\{ |\alpha|, |b|, \left(\frac{(2 + |\alpha| + (1 - \vartheta)(1 - \varsigma)|\psi|)}{\varsigma(1 - \vartheta)\Psi} \right)^{\frac{1}{n-1}} \right\},$$

$j = 0$

$z_0 = 0$

while $j \leq K$ **do**

$$z_{j+1} = \vartheta h(z_j) + (1 - \vartheta)y_j,$$

$$y_j = \varsigma z_j + (1 - \varsigma)\mathcal{U}(z_j), \text{ where } 0 < \vartheta, \varsigma < 1$$

if $|z_{j+1}| \geq R$ then break end if $j = j + 1$

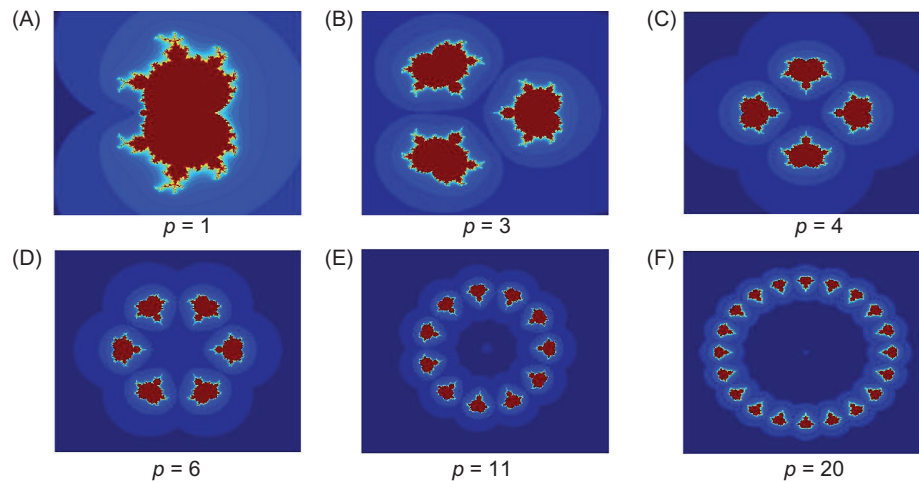
$$J = \left\lfloor (C-1) \frac{j}{K} \right\rfloor$$

colour z_0 with colourmap $[J]$

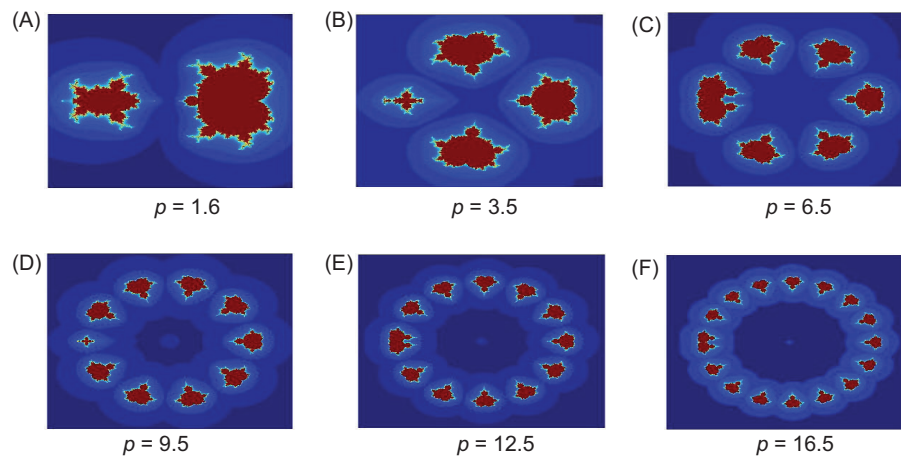
Table 6: Effect of the integer values for p while all other parameters are fixed.

	n	p	Ψ	ϑ	ς	a	b	IET
(a)	2	1	$-1.1i$	0.7	0.6	0.4	1	0.88s
(b)	2	3	$-1.1i$	0.7	0.6	0.4	1	0.92s
(c)	2	4	$-1.1i$	0.7	0.6	0.4	1	1.09s
(d)	2	6	$-1.1i$	0.7	0.6	0.4	1	1.21s
(e)	2	11	$-1.1i$	0.7	0.6	0.4	1	1.32s
(f)	2	20	$-1.1i$	0.7	0.6	0.4	1	1.47s

deviate from the characteristic forms seen at higher p values. Figures 8A–C, demonstrate more complex behavior for non-integer values of p . The set consists of small copies of a Mandelbrot sets. When the value of p increase the sets become smaller and occupy bigger space. We notice that the smaller sets start to overlap. With the further decrease in decimal value p , the overlapping becomes larger and the shapes of the smaller copies change because we do not see the characteristic shapes of the sets visible for the higher values of p . Despite this, the patterns retain their fascinating rosette-like symmetry. The image generation time for each iteration is also recorded. In the second example,

Figure 7: Viscosity Mandelbrot set for integer values of p generated via Algorithm 2.Table 7: Effect of non-integer value of p while all other parameters are fixed.

	n	p	Ψ	ϑ	ς	a	b	IET
(a)	2	1.6	$-1.1i$	0.7	0.6	0.4	1	0.88s
(b)	2	3.5	$-1.1i$	0.7	0.6	0.4	1	0.92s
(c)	2	6.5	$-1.1i$	0.7	0.6	0.4	1	1.09s
(d)	2	9.5	$-1.1i$	0.7	0.6	0.4	1	1.21s
(e)	2	12.5	$-1.1i$	0.7	0.6	0.4	1	1.32s
(f)	2	16.4	$-1.1i$	0.7	0.6	0.4	1	1.47s

Figure 8: Viscosity Mandelbrot set for non-integer values of p generated via Algorithm 2.

we see the effect on ϑ and ς to generate viscosity Mandelbrot sets via Algorithm 2 for $\mathcal{U}(z)$. The Mandelbrot sets were produced with the following fixed parameters.

In this example, the images were categorized into three groups. In each group, one parameter from ϑ or ς was held constant while the other was systematically varied to observe its effects. In Figure 9A–C,G,H,J,K, the Mandelbrot set structures exhibit strong rotational and axial symmetry, with the first three, seventh, and eighth images showing four-fold symmetry. Each image features intricate, self-similar boundaries and bulb-like red regions that merge toward the center, increasing density. A vivid color gradient illustrates escape time, with deep red for non-escaping points,

Table 8: Effect of the parameters ϑ and ς while all other parameters are fixed.

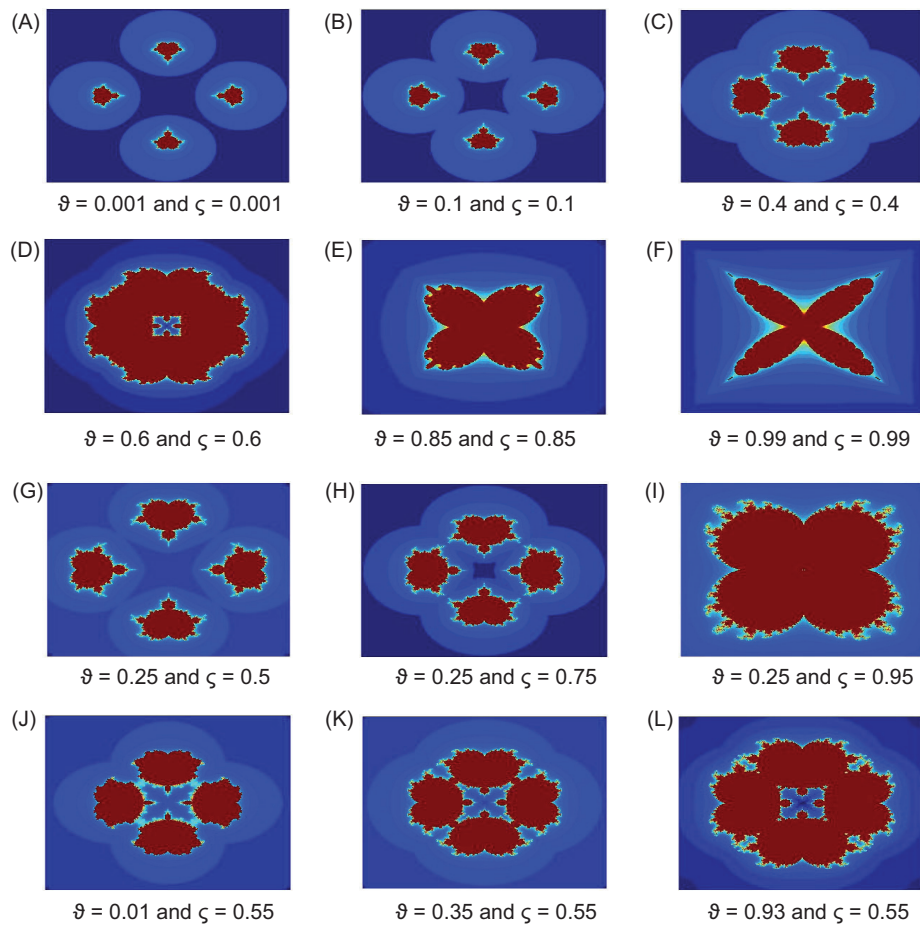
	n	p	Ψ	ϑ	ς	a	b	IET
(a)	2	4	1.05	0.001	0.001	$0.5i$	0.65	0.78s
(b)	2	4	1.05	0.01	0.01	$0.5i$	0.65	0.86s
(c)	2	4	1.05	0.4	0.4	$0.5i$	0.65	0.93s
(d)	2	4	1.05	0.6	0.6	$0.5i$	0.65	0.99s
(e)	2	4	1.05	0.85	0.85	$0.5i$	0.65	1.08s
(f)	2	4	1.05	0.99	0.99	$0.5i$	0.65	1.17s
(g)	2	4	1.05	0.25	0.55	$0.5i$	0.65	1.27s
(h)	2	4	1.05	0.25	0.75	$0.5i$	0.65	1.36s
(i)	2	4	1.05	0.25	0.95	$0.5i$	0.65	1.47s
(j)	2	4	1.05	0.01	0.55	$0.5i$	0.65	1.55s
(k)	2	4	1.05	0.35	0.55	$0.5i$	0.65	1.62s
(l)	2	4	1.05	0.93	0.55	$0.5i$	0.65	1.74s

and blue to cyan for faster escapes. Bright outlines mark the boundary regions. Each lobe contains intricate, self-similar patterns, and rotational dynamics. Variations in lobe size and density reflect changes in parameters and escape radius, while the central region remains stable. Moreover, when ϑ and ς exceed 0.5, the smaller sets begin to overlap, and as these parameters increase further, the overlapping becomes more pronounced while the shapes of the smaller copies deviate from their characteristic structures observed at higher values and intricate rosette patterns (see, Figures 9D–F,I,L). In contrast, at lower values of ϑ and ς , the sets retain more classical structures without overlapping effects.

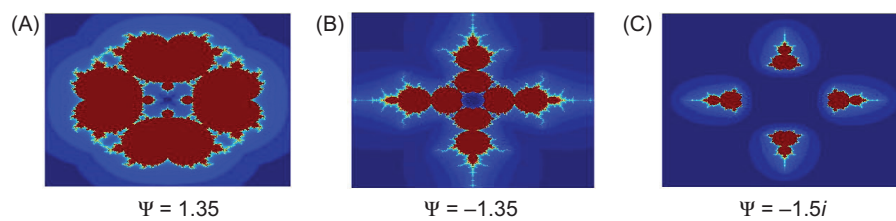
In the third example, viscosity Mandelbrot sets are generated via Algorithm 2 for $\mathcal{U}(z)$.

This example systematically investigates the influence of parameters Ψ , a , b by organizing the results into three groups (Figures 10–12), each focusing on one parameter while keeping the others constant. The fractal displays 4-fold radial symmetry, forming four petal-like lobes around a central point. Bright outlines mark the boundary regions. Each lobe contains intricate, self-similar patterns. Moreover, when a and b are purely imaginary, the smaller sets begin to overlap, and the overlapping becomes more pronounced while the complex values of a and b (see, Figures 11,12B,C). Visually, the structure resembles floral or Rangoli designs, combining mathematical complexity with artistic beauty. The image generation time for each iteration is also recorded.

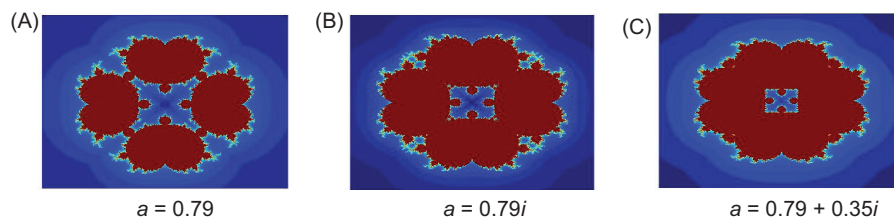
This example illustrates the influence of the exponent n on the generation of viscosity Mandelbrot sets using Algorithm 2 for $\mathcal{U}(z)$. The Mandelbrot sets were produced with the following parameter values: $p = 8$, $\Psi = 1.1$, $a = 0.35i$, $b = 0.85i$, $\vartheta = 0.5$, and $\varsigma = 0.65$. In Figures 13A–L, the fractal structures, presented as Mandelbrot sets, exhibit strong radial symmetry and are reminiscent of floral, flower-like, or star-shaped patterns. Specifically, Figure 13B features a 7-lobed structure symmetrically arranged around a large, empty central circle, with each lobe evenly spaced. The background transitions from deep blue, indicating rapid escape, to lighter blues and bright reds near the fractal boundaries, representing slower escape times. Each red lobe displays intricate, self-similar patterns with finely jagged and detailed edges, characteristic of fractal structures. The large central void and clearly separated lobes suggest minimal overlap among the structures. In other images, multiple smaller red “flower-like” fractal copies are symmetrically arranged around a central opening. The remaining images in the series (Figures 13A–L) consistently maintain the same core fractal characteristics, with variations arising from parameter adjustments. Each figure preserves the hallmark radial symmetry, floral or star-like morphology, and gradient color schemes transitioning from deep blue to vibrant red, reflecting differences in escape dynamics. The smaller size and tighter packing of the lobes indicate parameter changes that increase the number of fractal copies while preserving overall symmetry. It is observed that this parametric manipulation serves as a powerful tool for controlling the visual and structural properties of the fractals. It is seen that Our analysis demonstrates

Figure 9: Viscosity Mandelbrot sets generated for different ϑ and ς values via Algorithm 2.Table 9: Effect of the parameter Ψ while all other parameters are fixed.

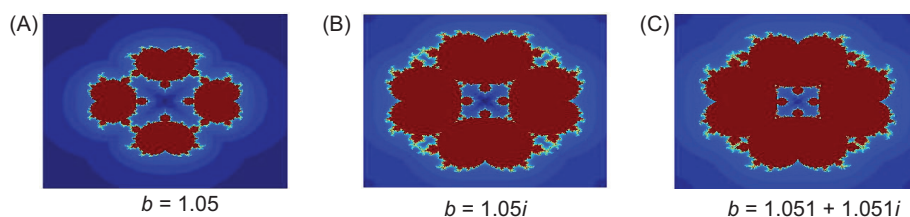
	n	p	Ψ	ϑ	ς	a	b	IET
(a)	2	4	1.05	0.65	0.45	$-0.65i$	0.55	0.98s
(b)	2	4	1.05	0.65	0.45	$-0.65i$	0.55	1.12s
(c)	2	4	1.05	0.65	0.45	$-0.65i$	0.55	1.21s

Figure 10: Viscosity Mandelbrot sets generated for different Ψ value via Algorithm 2.Table 10: Effect of the parameter a while all other parameters are fixed.

	n	p	Ψ	ϑ	ς	a	b	IET
(a)	2	4	1.25	0.65	0.45	0.79	-0.55	0.95s
(b)	2	4	1.25	0.65	0.45	$0.79i$	-0.55	1.14s
(c)	2	4	1.25	0.65	0.45	$0.79 + 0.35i$	-0.55	1.29s

Figure 11: Viscosity Mandelbrot sets generated for different a value via Algorithm 2.Table 11: Effect of the parameter b while all other parameters are fixed.

	n	p	Ψ	ϑ	ς	a	b	IET
(a)	2	4	1.85	0.65	0.45	0.35	1.05	1.05s
(b)	2	4	1.85	0.65	0.45	0.35	1.05i	1.19s
(c)	2	4	1.85	0.65	0.45	0.35	1.05 + 1.15i	1.29s

Figure 12: Viscosity Mandelbrot sets generated for different b value via Algorithm 2.Table 12: Effect of the parameter n while all other parameters are fixed.

	n	p	Ψ	ϑ	ς	a	b	IET
(a)	2	8	1.1	0.5	0.65	0.35i	0.85i	0.88s
(b)	3	8	1.1	0.5	0.65	0.35i	0.85i	0.96s
(c)	4	8	1.1	0.5	0.65	0.35i	0.85i	1.11s
(d)	5	8	1.1	0.5	0.65	0.35i	0.85i	1.21s
(e)	6	8	1.1	0.5	0.65	0.35i	0.85i	1.32s
(f)	7	8	1.1	0.5	0.65	0.35i	0.85i	1.42s
(g)	8	8	1.1	0.5	0.65	0.35i	0.85i	1.49s
(h)	9	8	1.1	0.5	0.65	0.35i	0.85i	1.54s
(i)	10	8	1.1	0.5	0.65	0.35i	0.85i	1.61s
(j)	15	8	1.1	0.5	0.65	0.35i	0.85i	1.72s
(k)	25	8	1.1	0.5	0.65	0.35i	0.85i	1.84s
(l)	35	8	1.1	0.5	0.65	0.35i	0.85i	1.97s

that parameters Ψ , α , p , a , b , ϑ , ς and the exponent n fundamentally control fractal morphology, while convergence criteria govern image resolution. These results not only highlight the mathematical depth of the study but also underscore the powerful fusion of computational techniques and creative expression.

- the parameters Ψ , α , p , a , b , ϑ , ς and the exponent n play a very important role in giving shape, size, and colour to the fractals.
- the convergence criteria derived for the fractals are also playing a very crucial role in determining the resolution and richness of the pixels in the fractals.
- all the fractals developed in this paper are very novel, aesthetic, and pleasing as the complex function $\mathcal{U}(z)$.

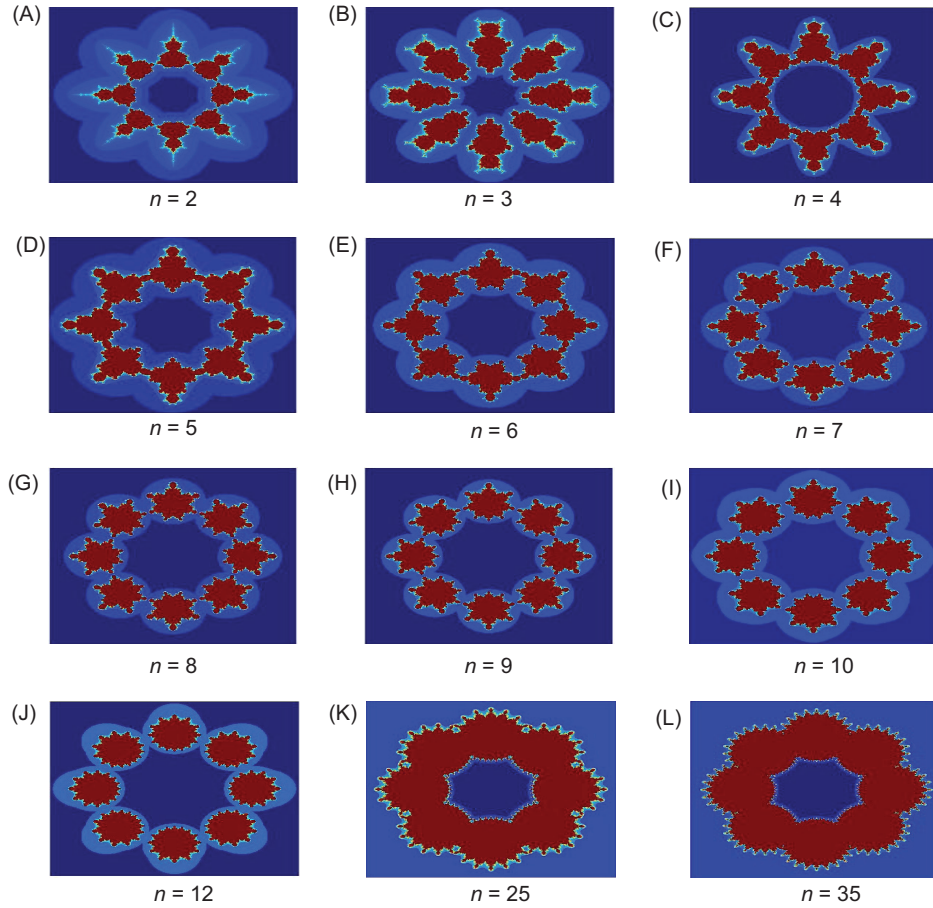


Figure 13: Viscosity Mandelbrot sets generated for different n value via Algorithm 2.

5. Numerical Result

The examples in Section 4 demonstrate a clear variation in the shapes and sizes of the fractal sets generated using the viscosity approximation-type iterative method, revealing a complex, non-trivial dependency on the parameters ϑ and ς . To investigate this relationship, we analyze two quantitative measures—the Average Escape Time (AET) and the Non-Escaping Area Index (NAI)—along with computation time. These measures provide insight into the relative size of the sets and the speed of the computations; they are directly influenced by the number of non-escaping points, as a higher concentration of these points results in higher AET and NAI values. Consequently, a significant difference in these measures, particularly the NAI, between two parameter sets indicates a substantial difference in the sizes of the corresponding Mandelbrot or Julia sets. For our numerical experiments, we generated a total of 12,00 fractals by varying ϑ and ς across 100 equally spaced values each, producing 200 X 200 pixel images using Algorithms 1 and 2 from Section 4. All computations were performed in MATLAB R2024a (version 24.1.0.2537033, 64-bit) on a standard HP laptop featuring an Intel(R) Core(TM) i7-14700HX (2.10 GHz) processor and 32 GB of RAM and Microsoft Windows 11.

In the first example, we generated quadratic Julia sets using the viscosity approximation-type iterative method for the following parameters: $K = 25$, $n = 2$, $h(z) = 0.89iz + 2.4A = [-1.75, 1.75]^2$ with two different value of $\alpha = 0.5$ and $\alpha = 0.07 + 0.87i$. The obtained results for the viscosity approximation-type iterative method are presented in Figure 14. From the plot, we see that both measures (AET, ANI, time) are non-trivial and that the function of the parameters is non-monotonic. Both plots have a similar shape, but the plot for the time is noisier, whereas the plot for the AET and ANI are

smooth. Minimal time, equal to 0.2s, is attained at $\vartheta = 0.5$ and $\varsigma = 0.5$, whereas the maximal value (0.97s) at $\vartheta, \varsigma = 0.5$. For the AET and ANI measure, the dispersion of the values is wide. Comparing these results with the results for Picard's iteration, we see that the generated Julia sets in the given area can be generated using a lower number of iterations. Moreover, from both plots, we can observe that the highest values of the measures are obtained when the ϑ and ς values varies.

In the next example, Mandelbrot sets for $n = 4$ were generated using the viscosity approximation-type iterative method. The parameters used to generate the images of the sets were the following: $K = 25$, $n = 4$, $A = [-1.5, 1.5]^2$ with $h(z) = -0.65iz + 0.55$ and $h(z) = -0.35iz + 0.85i$. The results for the viscosity approximation-type iterative method in Figure 15.

The plots reveal that the dependency of both measures on the parameters is non-trivial and non-monotonic. While the overall shapes of the AET, ANI, and time graphs are similar, the computation time plot is notably noisy compared to the smooth profiles of the AET and ANI. The generation time varies significantly, with a minimum of 0.016s observed at $\vartheta = 0.001$ and $\varsigma = 0.001$, and a maximum of 0.95s at $\vartheta = 0.99$ and $\varsigma = 0.99$. This demonstrates that the parameters ϑ and ς have a substantial impact on the generation time, AET, and ANI. Crucially, the shortest times achieved with the viscosity approximation-type iterative method are faster than those obtained using the standard Picard iteration.

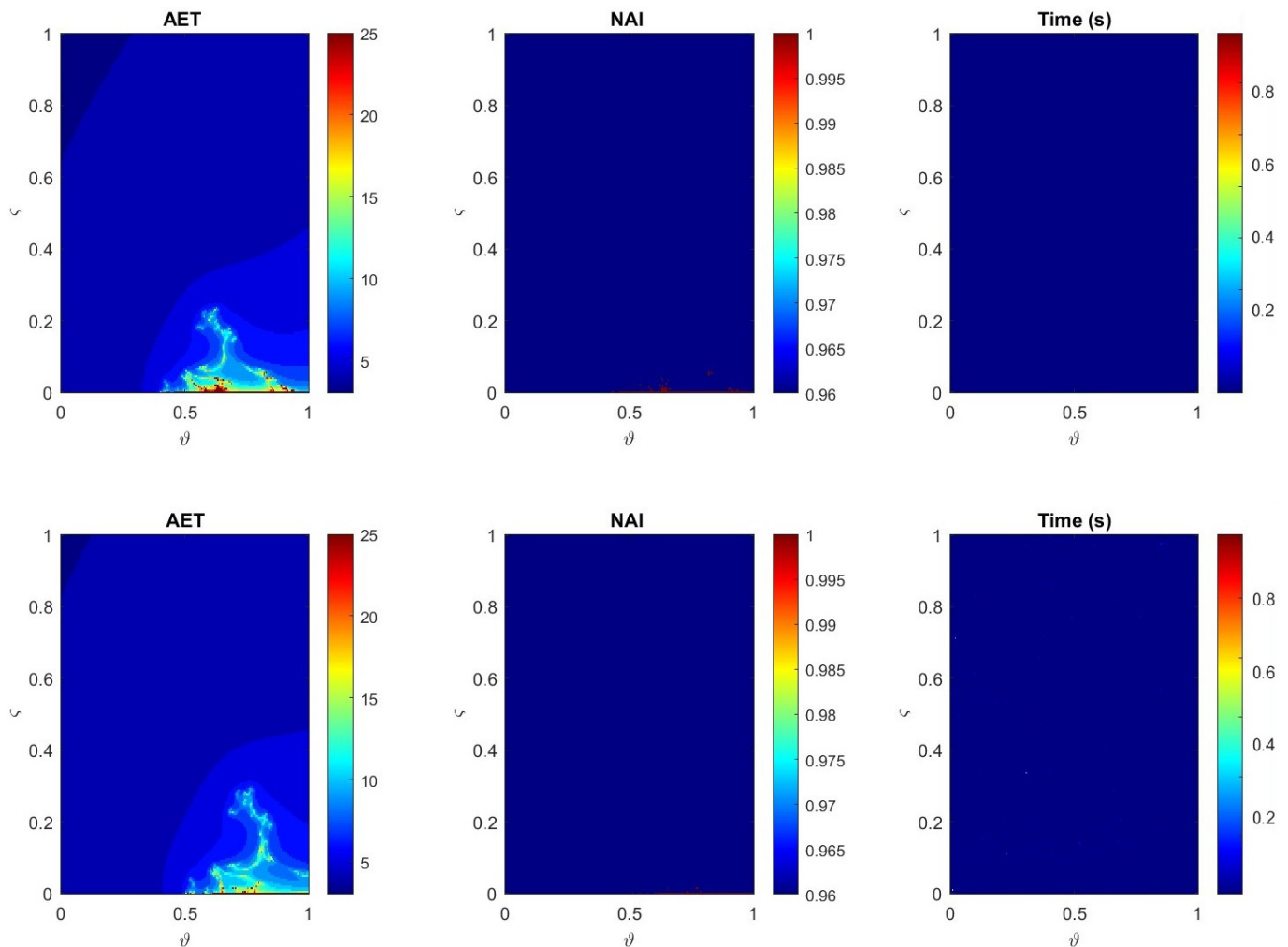


Figure 14: AET, NAI, and time (s) plots of iterations in the parameters' space for the Julia set with $\alpha = 0.5$ and $\alpha = 0.07 + 0.87i$.

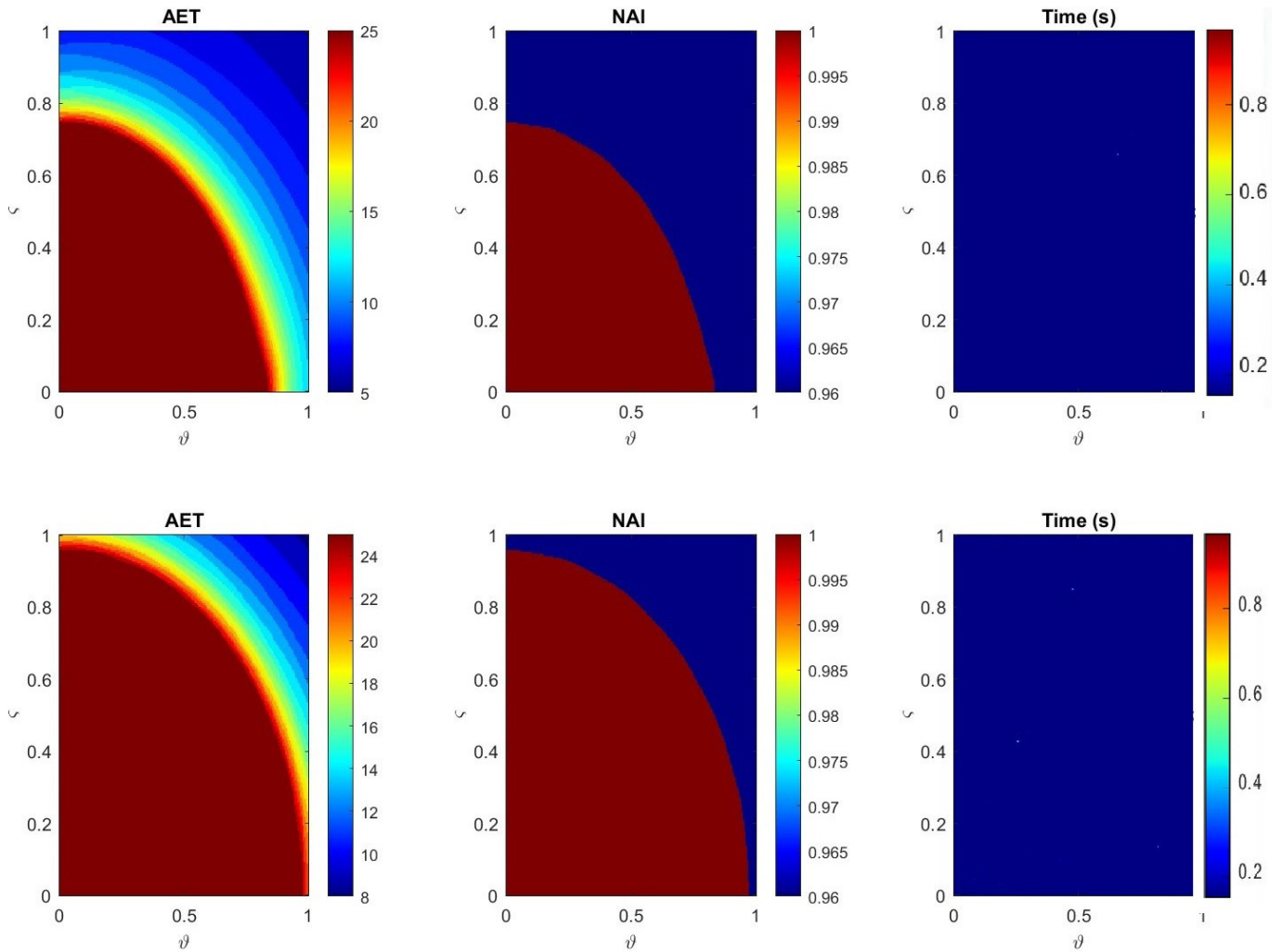


Figure 15: AET, NAI, and time (s) plots of iterations in the parameters' space for the Mandelbrot sets with $h(z) = -0.65iz + 0.55$ and $h(z) = -0.35iz + 0.85i$.

6. Conclusion

We developed an escape criterion based on the viscosity approximation-type iterative method for the considered complex function. Using this framework, we generated viscosity Julia and Mandelbrot sets and visualized them through Algorithms 1 and 2, respectively. MATLAB software was employed to analyze the behavior of these fractals under varying parameter values, revealing intricate and non-classical structures. Our findings show that parameters such as Ψ , α , p , a , b , ϑ , ς and the exponent n significantly influence the shape, color, and complexity of the fractals—even small variations lead to noticeable changes. The graphical examples showed that the proposed iteration scheme has the capacity to generate new fractal forms. The numerical examples showed that the dependence of the considered measures (AET, ANI, times (s)) on the iteration's parameters is a non-trivial and non-monotonic function. The results also showed that the ϑ parameter has greater impact on the measures than the ς parameter. In future work, we aim to explore further generalizations using modified rational exponential and sine functions, and to include additional metrics such as generation time and ANI in our analysis. The visual richness and diversity of the generated fractals also point to potential applications in the textile industry, especially in pattern design and printing.

7. Acknowledgments

The authors gratefully acknowledge Qassim University, represented by the Deanship of Graduate Studies and Scientific Research, on the financial support for this research under the number (QU-J-UG-2-2025-56981) during the academic year 1446 AH/2024 AD.

8. Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

- [1] N. Adhikari and W. Sintunavarat, *A novel investigation of quaternion Julia and Mandelbrot sets using the viscosity iterative approach*, Results in Control and Optimization 18 (2025), 100525, <https://doi.org/10.1016/j.rico.2025.100525>.
- [2] I. Ahmad, M. Sajid and R. Ahmad, *Julia sets of transcendental functions via a viscosity approximation-type iterative method with s-convexity*, Stat., Optim. Inf. Comput., Vol. 12, September 2024, pp. 1553–1572, <https://doi.org/10.19139/soic-2310-5070-1918>.
- [3] S. Antal, A. Tomar, D.J. Parjapati and M. Sajid, *Fractals as Julia sets of complex sine function via fixed point iterations*, Fractal Fract., **5**, (2021), 1-17, <https://doi.org/10.3390/fractalfract5040272>.
- [4] R. Babar and W. Sintunavarat, *On generation of Julia sets, Mandelbrot sets and biomorphs via a modification of the viscosity approximation method*, Results in Control and Optimization 18 (2025), 100516, <https://doi.org/10.1016/j.rico.2025.100516>.
- [5] A. Husain, M. N. Nanda, M. S. Chowdary, M. Sajid, *Fractals: An Eclectic Survey, Part-I*, Fractal Fract., **6** (89), (2022), <https://doi.org/10.3390/fractalfract6020089>.
- [6] L. Jolaoso, S. Khan and K. Aremu, *Dynamics of RK iteration and basic family of iterations for polynomiography*, Mathematics, **10** (18), (2022), 3324, <http://dx.doi.org/10.3390/math10183324>.
- [7] G. Julia, *Memoire sur l'iteration des fonctions rationnelles*, J. Math. Pures Appl., **8**, (1918), 47-245.
- [8] S. Kumari, K. Gdawiec, A. Nandal, M. Postolache and R. Chugh, *A novel approach to generate Mandelbrot sets, Julia sets and biomorphs via viscosity approximation method*, Chaos Solit. Fractals, **163**, (2022), 112-140, <http://dx.doi.org/10.1016/j.chaos.2022.112540>.
- [9] S. Kumari, K. Gdawiec, A. Nandal, N. Kumar and R. Chugh, *On the viscosity approximation type iterative method and its non-linear behavior in the generation of Mandelbrot and Julia sets*, Numer. Algorithms, vol. 2023, 2023.
- [10] Y. C. Kwun, M. Tanveer, W. Nazeer, K. Gdawiec, and S. M. Kang, *Mandelbrot and Julia sets via Jungck-CR iteration with s-convexity*, IEEE Access, **7**, (2019), 12167–12176.
- [11] P. Maingé, *The viscosity approximation process for quasi-nonexpansive mappings in Hilbert spaces*, Comput. Math. with Appl., **59**(1), (2010), 74–79, <https://doi.org/10.1016/j.camwa.2009.09.003>.
- [12] A. Maudafi, *Viscosity approximation methods for fixed-points problems*, J. Math. Anal. Appl. **2000**, 241, 46–55.
- [13] P. Muthukumar, P. Balasubramaniam, *Feedback synchronization of the fractional order reverse butterfly-shaped chaotic system and its application to digital cryptography*, Nonlinear Dyn., **74**, (2013), 1169-1181, <https://doi.org/10.1007/s11071-013-1032-3>.
- [14] K. Nakamura, *Iterated inversion system: An algorithm for efficiently visualizing Kleinian groups and extending the possibilities of fractal art*, J. Math. Arts, **15**, (2021), 106-136, <https://doi.org/10.1080/17513472.2021.1943998>.
- [15] A. Nandal, R. Chugh and M. Postolache, *Iteration process for fixed point problems and zero of maximal monotone operators*, Symmetry, **11** (5), pp. 655, 2019.
- [16] P. C. Ouyang, K. W. Chung, A. Nicolas and K. Gdawiec, *Self-similar fractal drawings inspired by M. C. Escher's print square limit*, ACM Trans. Graphic., **40**, (2021), 1-34, <https://doi.org/10.1145/3456298>.
- [17] W. Phuengrattana and S. Suantai, *On the rate of convergence of Mann, Ishikawa, Noor and SP-iterations for continuous functions on an arbitrary interval*, J. Comput. Appl. Math., **235** (9), (2011), 3006-3014.
- [18] S. Rawat, D.J. Prajapati, A. Tomar and K. Gdawiec, *Generation of Mandelbrot and Julia sets for generalized rational maps using SP-iteration process equipped with s-convexity*, Math. Comput. Simul., **220**, (2024), 148-169.
- [19] A. A. Shahid, W. Nazeer and K. Gdawiec, *The Picard-Mann iteration with s-convexity in the generation of Mandelbrot and Julia sets*, Monatsh. Math., **195**, (2021), 565-584, <https://doi.org/10.1007/s00605-021-01591-z>.
- [20] M. Tanveer, W. Nazeer and K. Gdawiec, *New escape criteria for complex fractals generation in Jungck-CR orbit*, Indian J. Pure Appl. Math., **51**, (2020), 1285-1303, <https://doi.org/10.1007/s13226-020-0466-9>.
- [21] G. I. Usurelu, A. Bejenaru and M. Postolache, *Newton-like methods and polynomiographic visualization of modified Thakur processes*, Int. J. Comput. Math., **98**, (2021), 1049-1068, <https://doi.org/10.1080/00207160.2020.1802017>.