



Article

Escape Criteria Using Hybrid Picard S-Iteration Leading to a Comparative Analysis of Fractal Mandelbrot Sets Generated with S-Iteration

Rekha Srivastava ^{1,*} , Asifa Tassaddiq ² and Ruhaila Md Kasmani ³ ¹ Department of Mathematics and Statistics, University of Victoria, Victoria, BC V8W 3R4, Canada² Department of Basic Sciences and Humanities, College of Computer and Information Sciences Majmaah University, Al-Majmaah 11952, Saudi Arabia; a.tassaddiq@mu.edu.sa³ Institute of Mathematical Sciences, Universiti Malaya, Kuala Lumpur 50603, Malaysia; ruhaila@um.edu.my

* Correspondence: rekhas@uvic.ca

Abstract: Fractals are a common characteristic of many artificial and natural networks having topological patterns of a self-similar nature. For example, the Mandelbrot set has been investigated and extended in several ways since it was first introduced, whereas some authors characterized it using various complex functions or polynomials, others generalized it using iterations from fixed-point theory. In this paper, we generate Mandelbrot sets using the hybrid Picard S-iterations. Therefore, an escape criterion involving complex functions is proved and used to provide numerical and graphical examples. We produce a wide range of intriguing fractal patterns with the suggested method, and we compare our findings with the classical S-iteration. It became evident that the newly proposed iteration method produces novel images that are more spontaneous and fascinating than those produced by the S-iteration. Therefore, the generated sets behave differently based on the parameters involved in different iteration schemes.

Keywords: picard iteration; S-iteration; Mandelbrot sets; hybrid Picard S-iteration



Citation: Srivastava, R.; Tassaddiq, A.; Kasmani, R.M. Escape Criteria Using Hybrid Picard S-Iteration Leading to a Comparative Analysis of Fractal Mandelbrot Sets Generated with S-Iteration. *Fractal Fract.* **2024**, *8*, 116. <https://doi.org/10.3390/fractalfract8020116>

Academic Editors: Sze-Man Ngai and Palle Jorgensen

Received: 9 December 2023

Revised: 14 January 2024

Accepted: 23 January 2024

Published: 15 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Fractals are frequently found in nature because they are good descriptions of things like rivers, clouds, crystals, lightning, electricity, tree branches, and leaf patterns. The study of many natural or living frameworks, such as the fractality in water distribution networks, benefits greatly from the use of fractals [1]. To understand and predict violent streams, fractals are also used in liquid mechanics. Fractal geometries play a crucial role in determining where water quality sensors should be placed in the water delivery network [2] and fractal river network [3]. Additionally, fractal theory is frequently applied in several fields, including engineering models, video compression [4], and computational architectural design [5]. Ring-shaped quantum dots play an important role in confining the electrons along a circular orbit. Remarkably, the effect of Mandelbrot fractality is explored by considering different types of Mandelbrot rings [6]. According to research by cognitive neuroscientists, computer-generated fractals can reduce stress in viewers in precisely the same way as fractals seen in nature [7]. These facts serve as the authors' inspiration to use the concepts from fixed-point theory to build a new escape criterion. Then, using the escape radius, we present a comparative analysis of the fractals generated by two different iterations. It turned out that the recently suggested hybrid iteration method creates new, more interesting, and spontaneous visuals than the S-iteration does. Therefore, we first examine and comprehend several related concepts in the following paragraphs before presenting the main results.

In 1980, Mandelbrot introduced the Mandelbrot set when he was studying complex quadratic function $z_{n+1} = z_n^2 + c$. He utilizes c as a complex parameter and invented the

word “Fractal” for such best-known illustrations generated by a very basic computational method [8]. The Mandelbrot set can be identified using a fractal artistic design [9]. Fractal art is commonly settled with the help of fractal-creating software. In certain examples, graphics tools are utilized to additionally adjust the patterns produced. It is well known that fractal images cannot be created without computer fractal art because a computer has the extraordinary ability to calculate [10]. Almost certainly, fractal art is recognized from other digital activities [11,12]. Fractals are generated using iterative techniques to solve polynomial equations or non-linear equations. Generating fractals can be an artistic endeavour, a mathematical model, or just a soothing diversion.

Various properties and extensions of the Mandelbrot set have been studied extensively in the literature. The initial and most evident generalization included the application of the function $z^p + c$ rather than the second-degree polynomial [13,14]. Other categories of functions were also examined in the literature such as anti-polynomials [15], transcendental [16], rational [17], elliptic [18], etc. Another extension in the study of Mandelbrot sets is from complex number system [19] to bicomplex numbers [20], quaternions [21], octonions [22], etc. Some fixed-point techniques were also used for the extension and generalization of Mandelbrot sets. For the creation of fractals using fixed-point theory, different cyclical methods for locating fixed points in an identified map are used; for example, inversion fractals [23], v -variable fractals [24], iterated function systems [25], and biomorphs [26]. A new and novel iteration scheme is applied to solve image deblurring and signal recovery problems [27] using polynomiography. The Mann iteration was used by Rani and Kumar [28,29] to visualize superior Julia and Mandelbrot sets. Afterwards, the Ishikawa iteration was used in [30,31] for the visualization of relative superior Julia and Mandelbrot sets. The authors in [32] proved that the convergence rate of S -iteration is higher than that of the Ishikawa iteration and presented relatively superior Mandelbrot sets through the S -iteration scheme. In [33], the Noor orbit was used by Rani et al. to visualize Mandelbrot sets. Li et al. utilize the Jungck–Mann orbit creation of Mandelbrot sets in [34]. Different iterative schemes were used by different researchers, such as in [35,36] Jungck–CR iterative formulas having a specific convexity. Similarly, in [37], S -iteration orbit with s -convexity was used. Then in [38], Jungck–Mann and Jungck–Ishikawa iterations with s -convexity were used. Noor orbit and s -convexity were also used in [39]. Recently, Zou et al. [40] introduced the Mandelbrot and Julia sets via hybrid Picard–Mann iteration. For more recent and updated studies on related iterations and techniques, the interested reader is referred to [41–43].

From this review of the literature, one can observe that most fractals have escape requirements that dictate their dynamical behaviour, and they do so through a variety of iterative strategies. One key idea is escape when figuring out if a point in the complex plane is contained in a Mandelbrot set. Thus, it is believed that accurately creating a high-level “Mandelbrot” is challenging and intricate [6]. One especially unique feature of fractals that is lacking in other systems is scaling invariance. They are very suitable for real-world situations where researchers may conduct studies on multiple dimensions because of this feature. Since creating fractals is crucial from several perspectives indicated above, a lot of focus has been placed on doing it in recent years, utilizing a variety of methods. Multiple iterative procedures discussed above have also been used to create some fractals with generic properties [44–47]. In these situations, a decision must be made between a few different iteration techniques while considering crucial factors. For instance, an iteration method is more effective than the others based on two primary criteria: simplicity and convergence speed. Under such circumstances, the following issues inevitably surface: Which of these iteration techniques is accelerating convergence? Hence, it was demonstrated that the Picard S -iteration method converges more quickly than the CR iteration method and, consequently, more quickly than any other known iteration method, as well as all of the Picard, Mann, Ishikawa, Noor, SP, and S methods [48]. To the best of our knowledge, this recently developed three-step iteration in the literature has not been applied to fractals. Hence this research article fills this gap, which is significant because many

iteration systems [49–51] provide variants for the same function in terms of shape, size, colour, and additional attributes. Furthermore, the results of research conducted at various scales might paint somewhat distinct images even though the geometrical features of the structure itself are the same. This is true for systems with diverse structures. Conversely, a mathematical fractal is a kind of feedback that depends on recursion; it is constructed around an iterated equation. Iteration is essential to fully appreciating the aesthetic value and beauty of fractals. It is simpler to visualize self-similar behaviour as consistently carrying out a task. Each time a step is finished, we replace each initiator copy with a smaller generator copy, rotating as necessary. Taking motivation from these facts, we use this newly developed iteration scheme to prove an escape criterion. It proved useful for generating fractals by analyzing their dynamic patterns in the presence of complex polynomials. The fact that a new iteration strategy was successfully implemented emphasizes how significant this research is compared to the previous ones. The dependence of time on variations in the involved parameters of the iteration scheme is analyzed numerically and graphically. The three-stage, fast-convergence Picard S-iterative technique is therefore more notable and unique than the corpus of existing work.

The plan of this article is as follows: We discuss some necessary definitions and preliminaries related to different iterations in Section 2. Then, we prove a general escape criterion to generate the fractals in Section 3. This criterion is proved using a new hybrid Picard S-iteration method for general complex polynomials. Further refinement of these criteria is also presented in the form of corollaries. The visualization of Mandelbrot sets is provided in Section 4. The pseudocode to create the Mandelbrot sets in Picard S-orbit is also exhibited in this section. Based upon that Sections 4.1 and 4.2 contain a variety of images for Mandelbrot sets by considering different parameter values in the proposed iteration technique. A comparison of fractal images produced by using the new proposed iteration and the classical S-iteration is also provided in this section. Time analysis is performed for seconds using tabular and graphical illustrations. This comparison proved that the Mandelbrot set images produced by the hybrid Picard S-iteration are far better than those produced by S-iteration. This comparison is discussed and concluded in Section 5, leading to some future directions of this work.

2. Basic Definitions and Preliminaries

This section contains some basic definitions and theorems that are essential to proving our new results.

Definition 1. The Mandelbrot set denoted by M for the given polynomial $q_c(s) = s^2 + c$ consists of all $c \in \mathbb{C}$ having a bounded orbit of the point 0 given as [44]

$$M = \{ \{q_c^N(0)\}; c \in \mathbb{C}; N = 0, 1, 2, \dots \text{ is bounded} \}.$$

Here and what follows, \mathbb{C} denotes the set of complex numbers. The beginning point has been taken as 0 because this is the only critical point of q_c .

Definition 2. Consider a map $Z : \mathbb{C} \rightarrow \mathbb{C}$, on the set of complex numbers denoted by \mathbb{C} then the subsequent iteration is a Mann iteration process [45]:

$$\begin{cases} u_0 \in \mathbb{C}, \\ u_{N+1} = (1 - \zeta)u_N + \zeta Z(u_N), \zeta \in (0, 1]; N \geq 0. \end{cases} \quad (1)$$

Definition 3. Consider a map $Z : \mathbb{C} \rightarrow \mathbb{C}$, on the set of complex numbers denoted by \mathbb{C} then the subsequent iteration is an Ishikawa iteration process [46]:

$$\begin{cases} u_0 \in \mathbb{C}, \\ u_{N+1} = (1 - \zeta)u_N + \zeta Z(v_N), \\ v_N = (1 - \delta)u_N + \delta Z(u_N), N \geq 0, \end{cases} \quad (2)$$

where $\zeta, \delta \in (0, 1]$.

Definition 4. Consider a map $Z : C \rightarrow C$, on the set of complex numbers denoted by C then the subsequent iteration is an S -iteration process [47]:

$$\begin{cases} u_0 \in C, \\ u_{N+1} = (1 - \alpha_1)Z(u_N) + \alpha_1 Z(v_N), \\ v_N = (1 - \alpha_2)u_N + \alpha_2 Z(u_N), N \geq 0, \end{cases} \quad (3)$$

where $\alpha_1, \alpha_2 \in (0, 1]$.

Definition 5. Consider a map $Z : C \rightarrow C$, on the set of complex numbers denoted by C and $u_0 \in C$ then the subsequent iteration is a Picard S -iteration process [48]

$$\begin{cases} u_{N+1} = Z(v_N), \\ v_N = (1 - \alpha_1)Z(u_N) + \alpha_1 Z(w_N), \\ w_N = (1 - \alpha_2)u_N + \alpha_2 Z(u_N); N \geq 0, \end{cases} \quad (4)$$

where $\alpha_1, \alpha_2 \in (0, 1]$. This iteration process, named as Picard S -orbit, depends on four variables $(Z, u_0, \alpha_1, \alpha_2)$ which can be written as "Picard S -orbit $(Z, u_0, \alpha_1, \alpha_2)$ ". The first term in (4) represents the Picard iteration whereas the second and third come from S -iteration. That is why it is named as the hybrid Picard S -iteration method or procedure.

Usually, the escape criterion for Mandelbrot sets is followed as stated in the following theorem [44].

Theorem 1. Consider a quadratic polynomial, $q_c(w) = w^2 + c$, $c \in C \geq 0$, then q satisfies the following such that

$$|q_c^N(w)| > \max\{2, |c|\},$$

and $|q_c^N(w)| \rightarrow \infty$ as $N \rightarrow \infty$.

The expression $\max\{2, |c|\}$ is called the threshold of the escape radius. This escape radius is different during every iteration. A widely used escape criterion for polynomials of the form $q_c(w) = w^k + c$ where $k \geq 2$, with S -iteration is given in the subsequent theorem [32].

Theorem 2. Suppose that $|w| \geq |c| > (\frac{2}{\alpha_1})^{\frac{1}{k-1}}$ and $|w| \geq |c| > (\frac{2}{\alpha_2})^{\frac{1}{k-1}}$, where $\alpha_1, \alpha_2 \in (0, 1], k \geq 2$ where c is a complex number. Suppose $z_o = z$ and $w_o = w$. Therefore, we have

$$\begin{cases} w_{N+1} = (1 - \alpha_1)q_c(w_N) + \alpha_1 q_c(z_N), \\ z_N = (1 - \alpha_2)w_N + \alpha_2 q_c(w_N), N \geq 0, \end{cases}$$

then $|w_N| \rightarrow \infty$ as $N \rightarrow \infty$.

Corollary 1. Suppose $|w| > \max\{|c|, (\frac{2}{\alpha_1})^{\frac{1}{k-1}}, (\frac{2}{\alpha_2})^{\frac{1}{k-1}}\}$ then $|w_N| \rightarrow \infty$ as $N \rightarrow \infty$.

3. Escape Criterion

The fixed-point theory is a fundamental concept in fractals because it offers a framework for comprehending the iterative schemes that produce fractals. The points in the fractal that do not change regardless of the number of times the scheme is iterated are actually the fixed points of that iterative scheme in the context of fractals. These fixed points "attract" neighbouring points lying in that fractal towards themselves and therefore are frequently referred to as attractors. The concept of "escape" is essential for colouring and visualizing fractals. A point that escapes to infinity quickly is often assigned one colour,

whereas other points that escape slower or cannot escape are assigned other colours. The maximum number of iterations needed to ascertain whether the orbit sequence tends to infinity determines the escape time algorithm. This algorithm provides a useful element that is significant to visualize the important properties of a dynamic system iteratively. We now prove the general escape criterion required to build the Mandelbrot sets in the Picard S-orbit. Consider a polynomial q_c as well as $w_0 \in \mathbb{C}$, then define the Picard S-iteration procedure as follows:

$$\begin{cases} w_{N+1} = q_c(x_N) \\ x_N = (1 - \alpha_1)q_c(w_N) + \alpha_1q_c(z_N), \\ z_N = (1 - \alpha_2)w_N + \alpha_2q_c(w_N), \end{cases} \quad (5)$$

where $N = 0, 1, 2, \dots$ and $\alpha_1, \alpha_2 \in (0, 1]$. The orbit of this newly defined iteration scheme may denoted as Picard S-orbit($q_c, w_0, \alpha_1, \alpha_2$). Hence, the subsequent result is the proof of the required escape criteria for generalized polynomial $q_c(w) = w^d + c$ where $d \geq 2$, using the hybrid Picard S-iteration procedure.

Theorem 3. Suppose that $|w| \geq |c| > \left(\frac{2}{\alpha_1}\right)^{\frac{1}{d-1}}$ and $|w| \geq |c| > \left(\frac{2}{\alpha_2}\right)^{\frac{1}{d-1}}$, where $\alpha_1, \alpha_2 \in (0, 1]$ and $c \in \mathbb{C}$. Let $w = w_0, x = x_0$ and $z = z_0$. Then, for $q_c(w) = w^d + c$, $|w_N| \rightarrow \infty$ as $N \rightarrow \infty$ using the proposed iteration (5).

Proof. We first consider the following expression

$$|z| = |(1 - \alpha_2)w + \alpha_2q_c(w)|$$

then we replace $q_c(w) = w^d + c$ and obtain the following expression,

$$\begin{aligned} |z| &= \left| (1 - \alpha_2)w + \alpha_2(w^d + c) \right| \\ |z| &\geq \left| \alpha_2w^d + (1 - \alpha_2)w \right| - |\alpha_2c| \\ &\geq \left| \alpha_2w^d + (1 - \alpha_2)w \right| - |\alpha_2w| \quad (\because |w| > |c|) \\ &\geq \left| \alpha_2w^d \right| - |(1 - \alpha_2)w| - |\alpha_2w| \\ &\geq \left| \alpha_2w^d \right| - |w| + |\alpha_2w| - |\alpha_2w| \\ &= |w|(\alpha_2|w|^{d-1} - 1). \end{aligned} \quad (6)$$

Furthermore, we proceed with the hybrid Picard S-iteration process, taking into account the following:

$$\begin{aligned} |x| &= |(1 - \alpha_1)q_c(w) + \alpha_1q_c(z)| \\ &= \left| (1 - \alpha_1)(w^d + c) + \alpha_1(z^d + c) \right|. \end{aligned} \quad (7)$$

Next, combining this with (6) leads to the following

$$|x| \geq \left| (1 - \alpha_1)(w^d + c) + \alpha_1(|w|(\alpha_2|w|^{d-1} - 1))^d + c \right|. \quad (8)$$

Since $|w| > \left(\frac{2}{\alpha_2}\right)^{\frac{1}{d-1}}$, we obtain $\alpha_2|w|^{d-1} > 2$ and $(\alpha_2|w|^{d-1} - 1)^d > 1$, which leads to the following

$$|w|^d(\alpha_2|w|^{d-1} - 1)^d > |w|^d. \quad (9)$$

By making use of (9) in (8), we obtain

$$\begin{aligned}
 |x| &\geq \left| (1 - \alpha_1)(w^d + c) + \alpha_1(|w|^d + c) \right| \\
 &= \left| \alpha_1|w|^d + (1 - \alpha_1)w^d + c - \alpha_1c + \alpha_1c \right| \\
 &= \left| \alpha_1|w|^d + (1 - \alpha_1)w^d + c \right| \\
 &\geq \left| \alpha_1|w|^d - (\alpha_1 - 1)w^d \right| - |c| \\
 &\geq \left| \alpha_1|z|^d - (\alpha_1 - 1)z^d \right| - |z| (\because |z| > |c|) \\
 &\geq \alpha_1|w|^d - \left| (\alpha_1 - 1)w^d \right| - |w| \\
 &= |w|^d - |w| \\
 &= |w| \left(|w|^{d-1} - 1 \right).
 \end{aligned} \tag{10}$$

The last or the third step of the hybrid Picard S-iteration procedure implies that

$$\begin{aligned}
 |w_1| &= |q_c(x)| \\
 &= \left| x^d + c \right|.
 \end{aligned} \tag{11}$$

Then, by making use of (10) in (11), we obtain the following

$$|w_1| \geq \left| (|w| \left(|w|^{d-1} - 1 \right))^d + c \right|. \tag{12}$$

Since $|w| > \left(\frac{2}{\alpha_1} \right)^{\frac{1}{d-1}} \geq (2)^{d-1}$; $\alpha_1 \in (0, 1]$, using the following relation in (12),

$$|w|^{d-1} - 1 \geq 1 \text{ or } |w|^d \left(|w|^{d-1} - 1 \right) \geq |w|^d$$

we obtain

$$\begin{aligned}
 |w_1| &\geq \left| |w|^d + c \right| \\
 &\geq |w|^d - |c| \\
 &\geq |w|^d - |w| (\because |w| > |c|) \\
 &\geq |w| \left(|w|^{d-1} - 1 \right).
 \end{aligned}$$

Because $|w| \geq \left(2 \right)^{\frac{1}{d-1}}$ gives that $|w|^{d-1} - 1 > 1$. Therefore, there exists $\delta > 0$, such that $|w|^{d-1} - 1 > 1 + \delta > 1$. Consequently, we obtain the following

$$|w_1| > (1 + \delta)|w|.$$

The same reasoning can be used again to obtain:

$$\begin{aligned}
 |w_2| &> (1 + \delta)^2|w|, \\
 &\vdots \\
 |w_N| &> (1 + \delta)^N|w|.
 \end{aligned}$$

Hence, $|w_N| \rightarrow \infty$ as $n \rightarrow \infty$, which completes the proof. \square

Corollary 2. Let us take

$$|w| \geq |c| > \left(\frac{2}{\alpha_1}\right)^{\frac{1}{d-1}} \text{ and } |w| \geq |c| > \left(\frac{2}{\alpha_2}\right)^{\frac{1}{d-1}}, \quad (13)$$

then, the Picard S-orbit $(q_c, 0, \alpha_1, \alpha_2)$ escapes to infinity.

The refinement of the escape criterion is provided in the form of the next corollary.

Corollary 3. Let $\alpha_1, \alpha_2 \in (0, 1]$ and suppose that

$$|w| > \max \left\{ |c|, \left(\frac{2}{\alpha_1}\right)^{\frac{1}{d-1}}, \left(\frac{2}{\alpha_2}\right)^{\frac{1}{d-1}} \right\}, \quad (14)$$

then, there exists $\delta > 0$ such that $|w_N| > (1 + \delta)^N |w|$ and $|w_N| \rightarrow \infty$ as $N \rightarrow \infty$.

4. Rich and Exquisite Patterns of the Fractal Mandelbrot Sets

This section is concerned with the rich and exquisite patterns of the Mandelbrot set produced using the escape time algorithm. This depends on the greatest number of iterations required to determine whether the orbit sequence gets closer to infinity. When employing an iterative process, this method offers a useful framework that can be utilized to demonstrate specific features of dynamic systems. New Mandelbrot sets are presented here in this section for quadratic and cubic polynomials using the hybrid Picard S-iteration, as well as compared with Mandelbrot sets created by the S-iteration. The graphics were produced using the Mathematica 10 software and the escape specifications with the escape time algorithm. Observations were analyzed using a computer having specifications such as of Intel i5-2400 (@3.1 GHz) processor, 8 GB DDR3 RAM, and Microsoft Windows 10 (64-bit) operating system. The pseudocode of the Mandelbrot set creation algorithm in Picard S-orbit is exhibited in Algorithm 1 and for S-Orbit in Algorithm 2.

Algorithm 1: Creation of the Mandelbrot sets using the hybrid Picard S-orbit

Input: $q_c(w) = w^d + c$; $d \geq 2$; $R \subset \mathbb{C}$ – area, K – iterations, $\alpha_1, \alpha_2 \in (0, 1]$ – parameters for Picard S-iteration procedure, $colourmap[0..n-1]$ – with n colours.

Output: Mandelbrot set for area R .

```

1 for  $c \in A$  do
2    $R = \max\{|c|, (\frac{2}{\alpha_1})^{\frac{1}{d-1}}, (\frac{2}{\alpha_2})^{\frac{1}{d-1}}\}$ 
    $N = 0$ 
    $w_0 = 0$ 
   while  $N \leq K$  do
3      $y_N = (1 - \alpha_2)w_N + \alpha_2 q_c(w_N)$ ,
        $x_N = (1 - \alpha_1)q_c(w_N) + \alpha_1 q_c(y_N)$ ,
4      $w_{N+1} = q_c(x_N)$ 
       if  $|w_{N+1}| > R$  then
5       break
6      $N = N + 1$ 
7    $i = \lfloor (n-1) \frac{N}{K} \rfloor$ 
     colour  $c$  with  $colourmap[i]$ 

```

Algorithm 2: Creation of the Mandelbrot sets using the S-orbit

Input: $q_c(w) = w^d + c$; $d \geq 2$; $R \subset \mathbb{C}$ – area, K – iterations, $\alpha_1, \alpha_2 \in (0, 1]$ – parameters for Picard S-iteration procedure, $colourmap[0..n-1]$ – with n colours.

Output: Mandelbrot set for area R .

```

1 for  $c \in A$  do
2    $R = \max\{|c|, (\frac{2}{\alpha_1})^{\frac{1}{d-1}}, (\frac{2}{\alpha_2})^{\frac{1}{d-1}}\}$ 
    $N = 0$ 
    $w_0 = 0$ 
   while  $N \leq K$  do
3      $y_N = (1 - \alpha_2)w_N + \alpha_2 q_c(w_N)$ ,
        $x_N = (1 - \alpha_1)q_c(w_N) + \alpha_1 q_c(y_N)$ ,
       if  $|w_{N+1}| > R$  then
4       break
5      $N = N + 1$ 
6    $i = \lfloor (n-1) \frac{N}{K} \rfloor$ 
   colour  $c$  with  $colourmap[i]$ 

```

4.1. Rich and Exquisite Patterns of the Mandelbrot Sets Using Hybrid Picard S-Iteration vs. S-Iteration and Quadratic Functions

In Figures 1–10, Mandelbrot sets using quadratic functions are visualized in the hybrid Picard S-orbit and S-orbit by choosing the greatest number of iterations, 20, as well as fixing parameter $\alpha_2 = 0.4$ and varying the values of the parameter α_1 . The time comparison for quadratic functions using Picard S-orbit and S-orbit is also analyzed in Table 1 and Figure 11 by varying the value of α_1 .

- In Figures 1 and 2 using $A = [-3.3, 0.9] \times [-2.6, 2.6]$ Mandelbrot sets (MSs) are created in the hybrid Picard S-orbit and S-orbit for the fixed value of $\alpha_1 = 0.5$.
- In Figures 3 and 4 using $A = [-3.6, 1.2] \times [-2.5, 2.5]$ Mandelbrot sets (MSs) created in the hybrid Picard S-orbit and S-orbit for the fixed value of $\alpha_1 = 0.6$.
- In Figures 5 and 6 using $A = [-3.8, 1.2] \times [-2.5, 2.5]$ Mandelbrot sets (MSs) created in the hybrid Picard S-orbit and S-orbit for the fixed value of $\alpha_1 = 0.7$.
- In Figures 7 and 8 using $A = [-4.0, 1.2] \times [-2.5, 2.5]$ Mandelbrot sets (MSs) created in the hybrid Picard S-orbit and S-orbit for the fixed value of $\alpha_1 = 0.8$.
- In Figures 9 and 10 using $A = [-4.1, 1.2] \times [-2.5, 2.5]$ Mandelbrot sets (MSs) created in the hybrid Picard S-orbit and S-orbit for the fixed value of $\alpha_1 = 0.9$.

Table 1. Time comparison of iterative schemes in seconds using quadratic functions.

α_1	Time for the Hybrid Picard S-Iteration	Time for S-Iteration
0.5	6.81250	7.42188
0.6	5.78125	5.96875
0.7	5.09375	5.37500
0.8	4.37500	5.00000
0.9	3.78125	5.06250

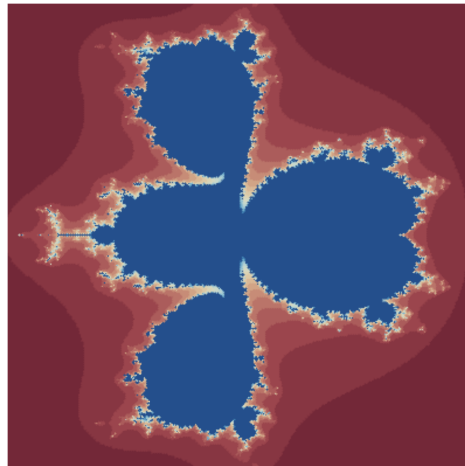


Figure 1. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.5$.

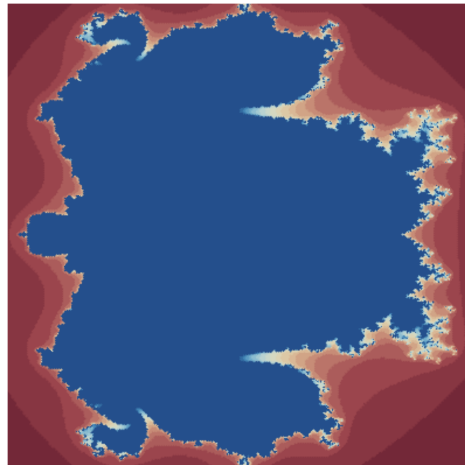


Figure 2. Created in S-orbit for $\alpha_1 = 0.5$.

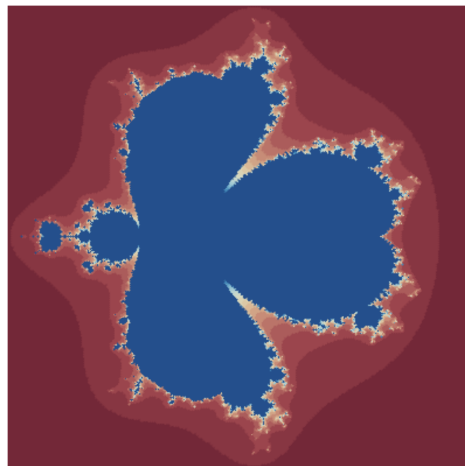


Figure 3. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.6$.

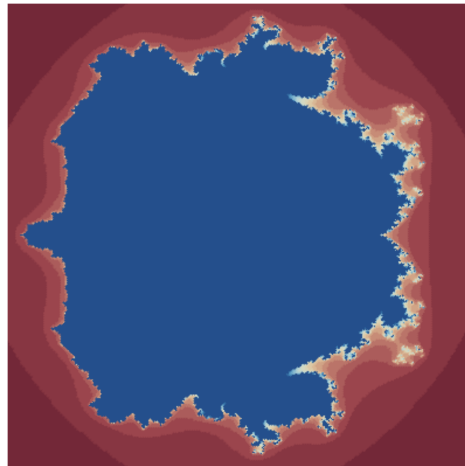


Figure 4. Created in S-orbit for $\alpha_1 = 0.6$.

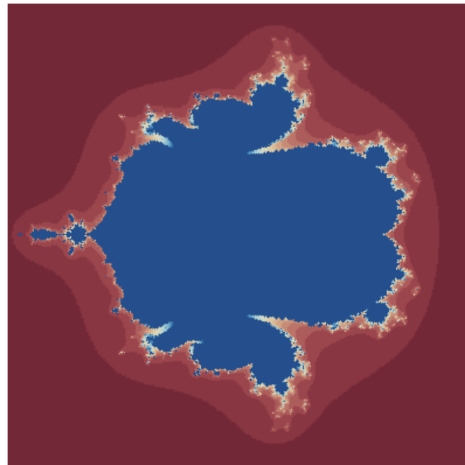


Figure 5. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.7$.

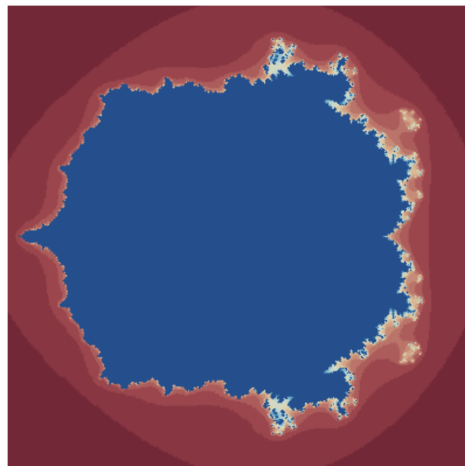


Figure 6. Created in S-orbit for $\alpha_1 = 0.7$.

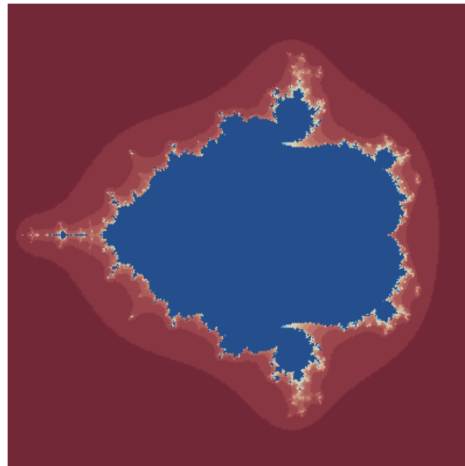


Figure 7. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.8$.

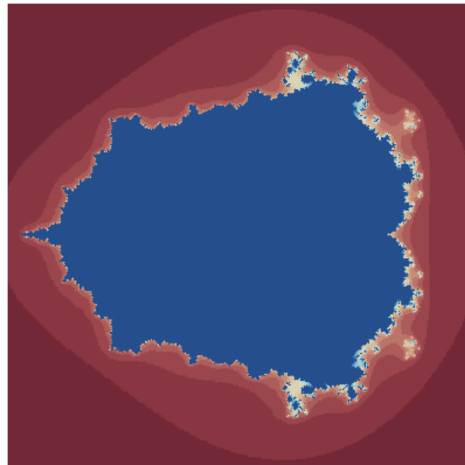


Figure 8. Created in S-orbit for $\alpha_1 = 0.8$.

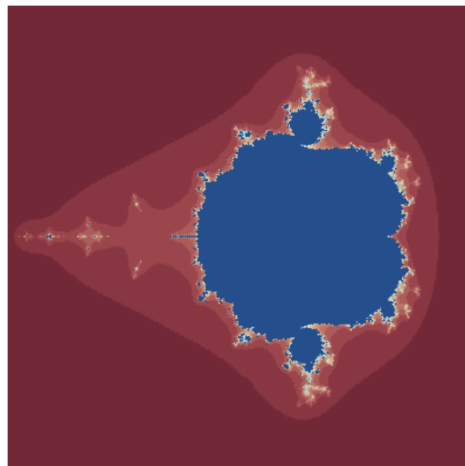


Figure 9. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.9$.

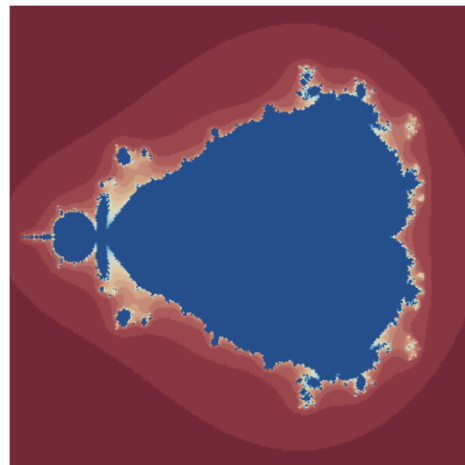


Figure 10. Created in S-orbit for $\alpha_1 = 0.9$.

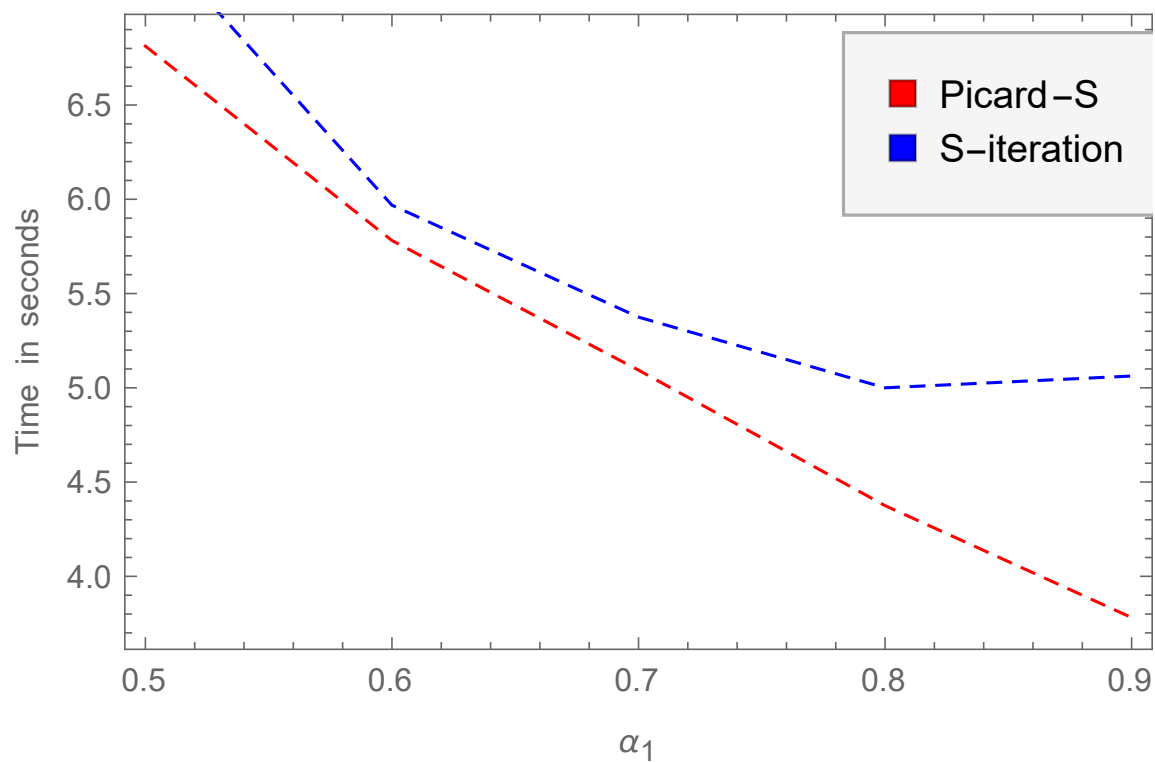


Figure 11. Dependence of the Mandelbrot set creation time (in seconds) in the hybrid Picard S-orbit and S-orbit on α_1 .

In Figures 12–23, quadratic Mandelbrot sets are presented in Picard S-orbit and S-orbit by fixing parameter $\alpha_1 = 0.7$, choosing the greatest number of iterations (20), and varying parameter α_2 . The time comparison for quadratic functions using Picard S-orbit and S-orbit is also analyzed in Table 2 and Figure 24 by varying the value of α_2 .

- In Figures 12 and 13 with $A = [-5.3, 1.4] \times [-2.8, 2.8]$ Mandelbrot set created in the hybrid Picard S-orbit and S-orbit for the same value of $\alpha_2 = 0.2$.
- In Figures 14 and 15 with $A = [-5.3, 1.4] \times [-2.8, 2.8]$ Mandelbrot set created in the hybrid Picard S-orbit and S-orbit for the same value of $\alpha_2 = 0.3$.
- In Figures 16 and 17 with $A = [-4.7, 1.4] \times [-2.8, 2.8]$ Mandelbrot set created in the hybrid Picard S-orbit and S-orbit for the same value of $\alpha_2 = 0.4$.
- In Figures 18 and 19 with $A = [-4.0, 1.4] \times [-2.8, 2.8]$ Mandelbrot set created in the hybrid Picard S-orbit and S-orbit for the same value of $\alpha_2 = 0.5$.

- In Figures 20 and 21 with $A = [-2.8, 1.0] \times [-2.5, 2.5]$ Mandelbrot set created in the hybrid Picard S-orbit and S-orbit for the same value of $\alpha_2 = 0.7$.
- In Figures 22 and 23 with $A = [-2.6, 1.0] \times [-2.2, 2.2]$ Mandelbrot set created in the hybrid Picard S-orbit and S-orbit for the same value of $\alpha_2 = 0.8$.

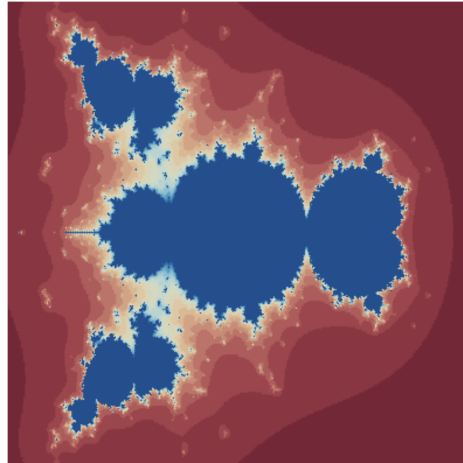


Figure 12. Created in the hybrid Picard S-orbit for $\alpha_2 = 0.2$.

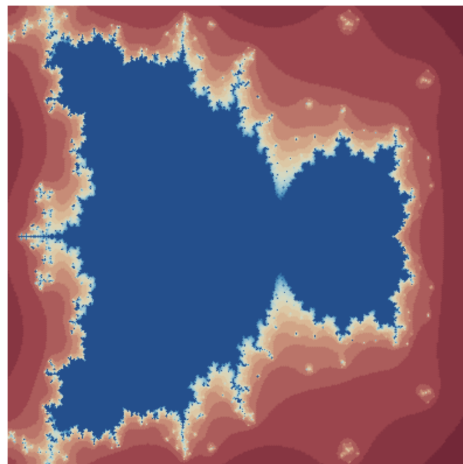


Figure 13. Created in the hybrid S-orbit for $\alpha_2 = 0.2$.

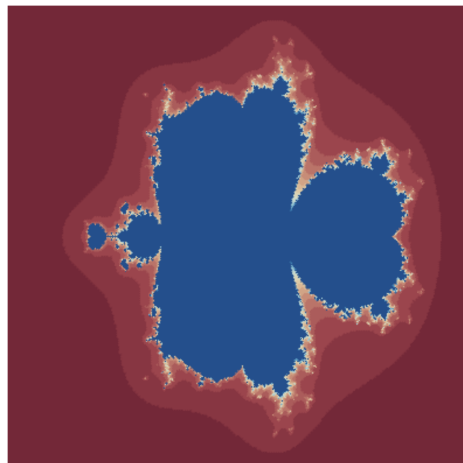


Figure 14. Created in the hybrid Picard S-orbit for $\alpha_2 = 0.3$.

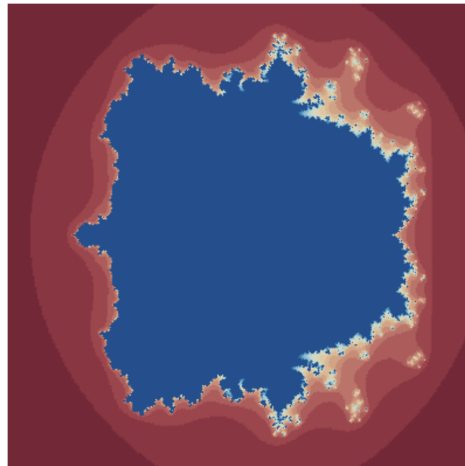


Figure 15. Created in S-orbit for $\alpha_2 = 0.3$.

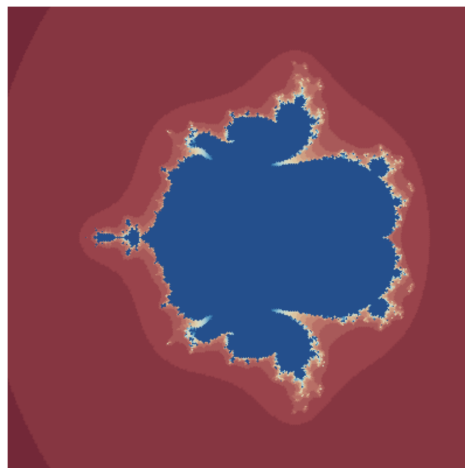


Figure 16. Created in the hybrid Picard S-orbit for $\alpha_2 = 0.4$.

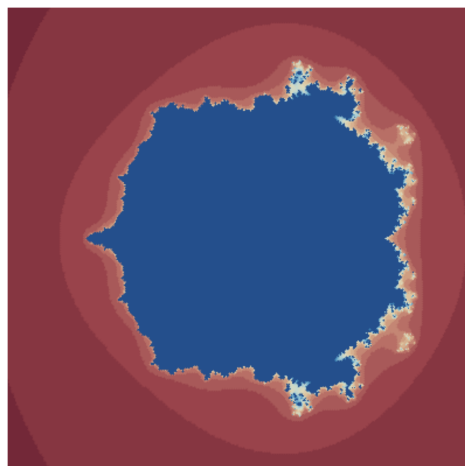


Figure 17. Created in S-orbit for $\alpha_2 = 0.4$.

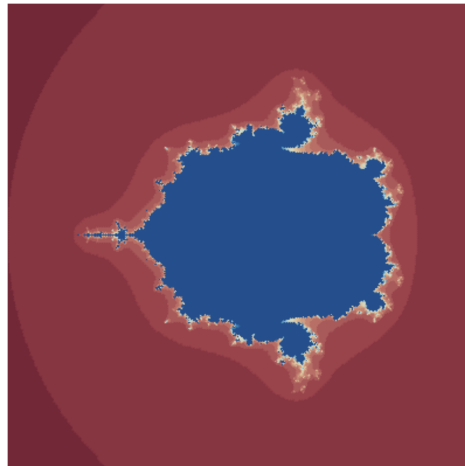


Figure 18. Created in the hybrid Picard S-orbit for $\alpha_2 = 0.5$.

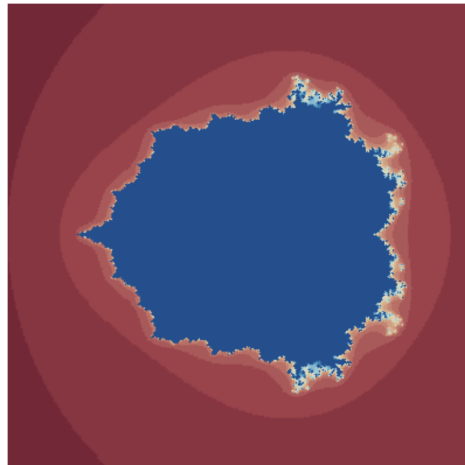


Figure 19. Created in S-orbit for $\alpha_2 = 0.5$.

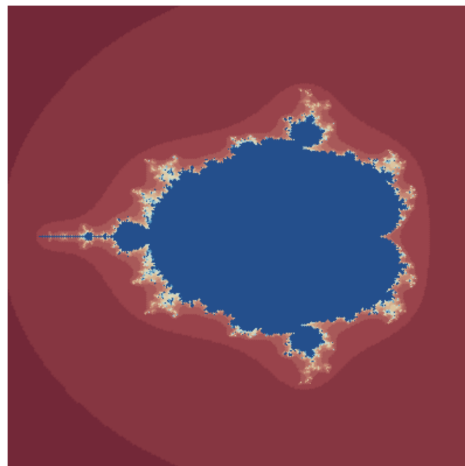


Figure 20. Created in the hybrid Picard S-orbit for $\alpha_2 = 0.7$.

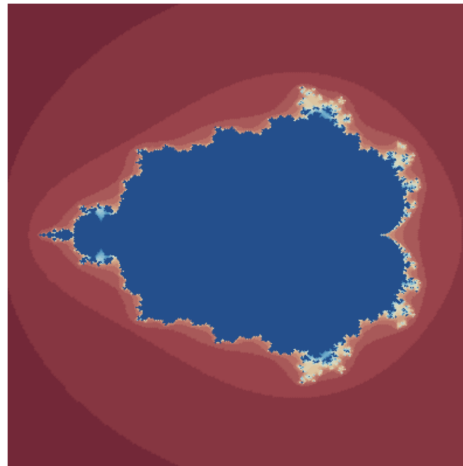


Figure 21. Created in S-orbit for $\alpha_2 = 0.7$.

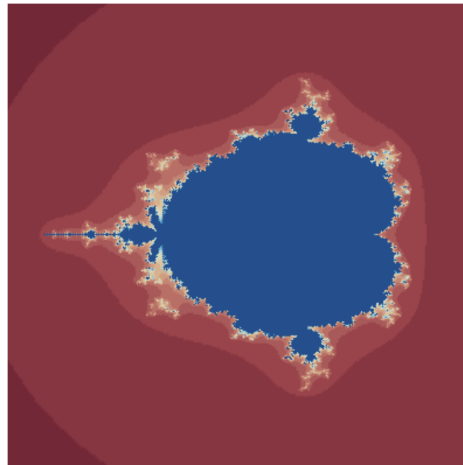


Figure 22. Created in the hybrid Picard S-orbit for $\alpha_2 = 0.8$.

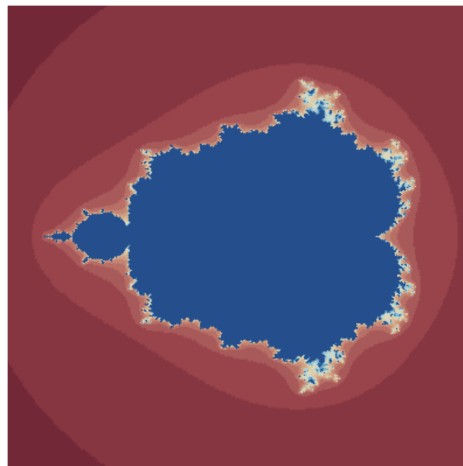
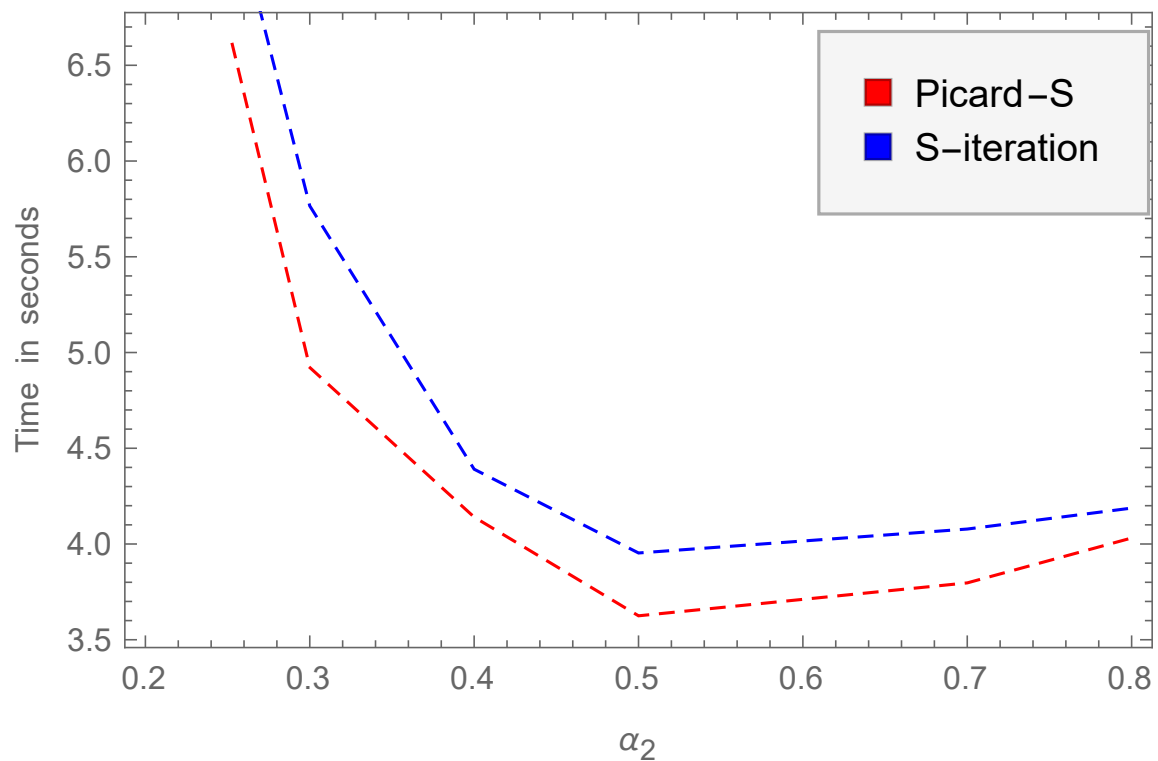


Figure 23. Created in S-orbit for $\alpha_2 = 0.8$.

Table 2. Time comparison of iterative schemes in seconds using quadratic functions.

α_2	Time for the Hybrid Picard S-Iteration	Time for S-Iteration
0.2	8.50000	9.14063
0.3	4.92188	5.76563
0.4	4.14063	4.39063
0.5	3.62500	3.95313
0.7	3.79688	4.07813
0.8	4.03125	4.18750

**Figure 24.** Dependence on time (in seconds) in the hybrid Picard S-orbit and S-orbit for α_2 .

4.2. Rich and Exquisite Patterns of the Mandelbrot Sets Using Hybrid Picard S-Iteration vs. S-Iteration and Cubic Functions

In fractal theory, the Mandelbrot set is arguably the most well-known object. It is thought to be the most complex object that has been made visible, in addition to being the most beautiful. The cubic polynomial analogue of the Mandelbrot set has two critical orbits. Therefore, their analysis using cubic polynomials is significantly more complex as compared to quadratic polynomials.

In Figures 25–38, Mandelbrot sets are presented using the cubic function along with the hybrid Picard S-iteration and S-iteration by choosing $A = [-1.4, 1.4] \times [-2.2, 2.2]$ maximum number of iterations 20 and fixing parameter $\alpha_2 = 0.5$. The time comparison for cubic functions using hybrid Picard S-orbit and S-orbit is also analyzed in Table 3 and Figure 39 by varying the value of α_1 .

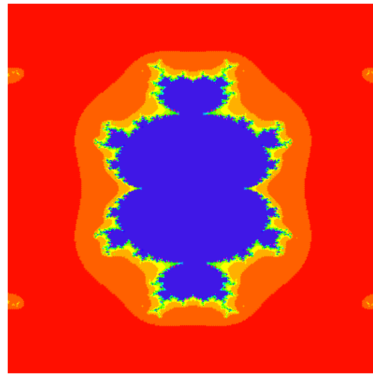


Figure 25. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.1$.

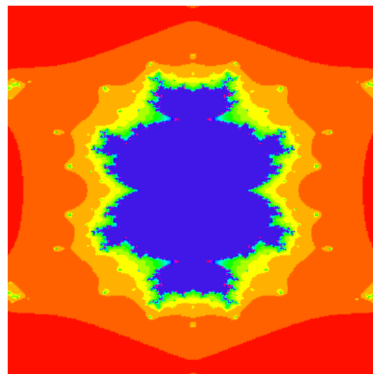


Figure 26. Created in S-orbit for $\alpha_1 = 0.1$.

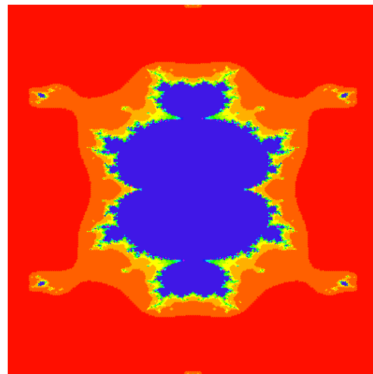


Figure 27. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.2$.

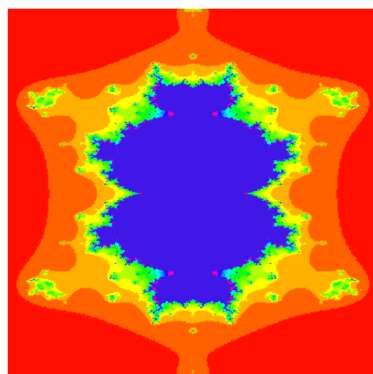


Figure 28. Created in S-orbit for $\alpha_1 = 0.2$.

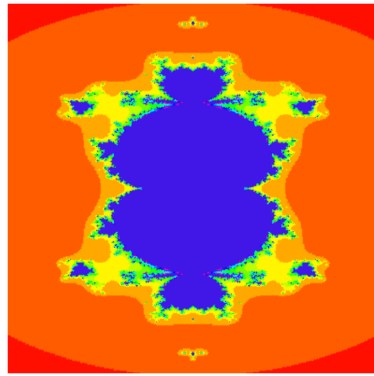


Figure 29. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.4$.

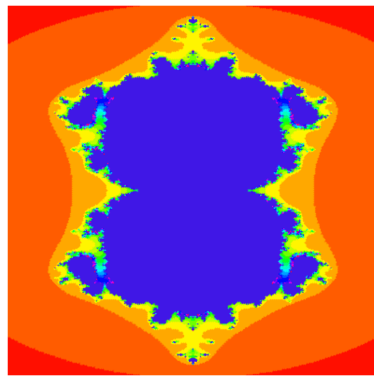


Figure 30. Created in S-orbit for $\alpha_1 = 0.4$.

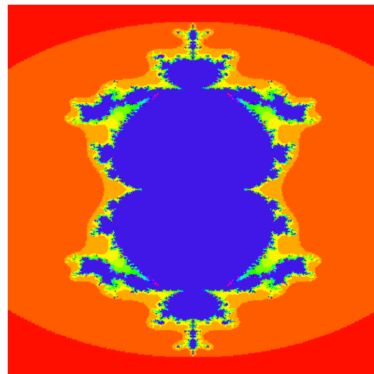


Figure 31. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.5$.

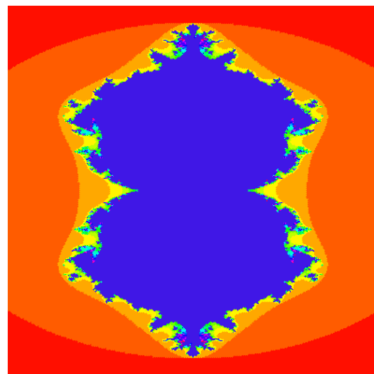


Figure 32. Created in S-orbit for $\alpha_1 = 0.5$.

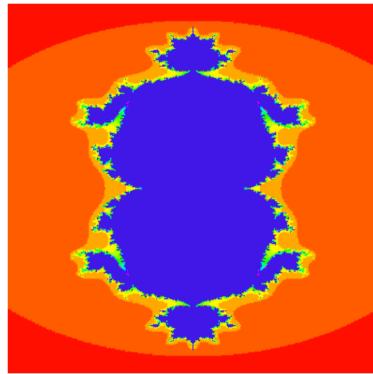


Figure 33. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.6$.

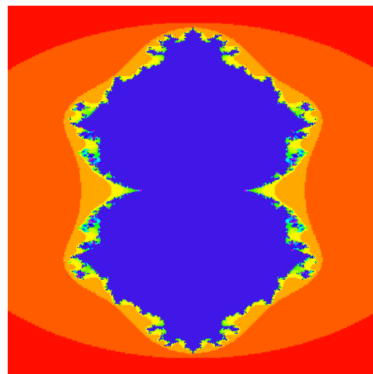


Figure 34. Created in S-orbit for $\alpha_1 = 0.6$.

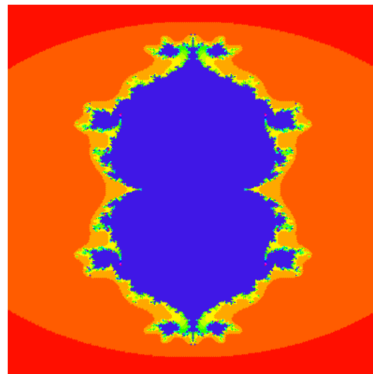


Figure 35. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.7$.

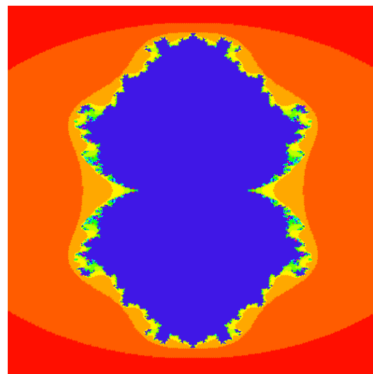


Figure 36. Created in S-orbit for $\alpha_1 = 0.7$.

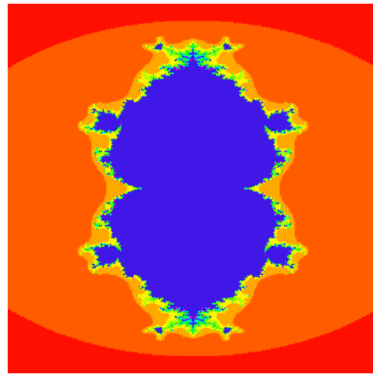


Figure 37. Created in the hybrid Picard S-orbit for $\alpha_1 = 0.8$.

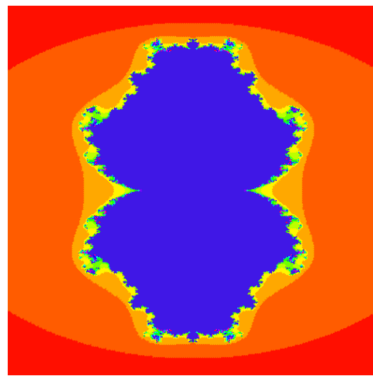


Figure 38. Created in S-orbit for $\alpha_1 = 0.8$.

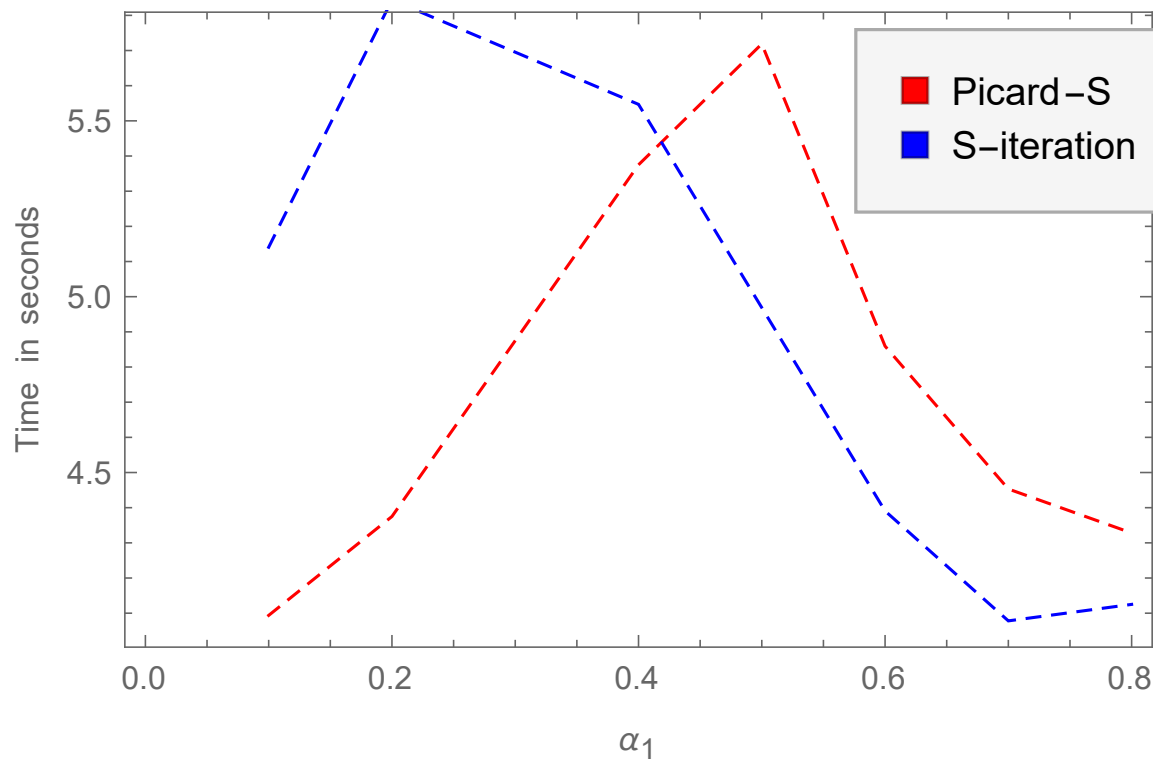


Figure 39. Dependence on time (in seconds) in Picard S-orbit and S-orbit on α_2 .

Table 3. Time comparison of iterative schemes in seconds using cubic functions.

α_1	Time for Picard S-Iteration	Time for S-Iteration
0.1	4.09375	5.14063
0.2	4.37500	5.84375
0.4	5.37500	5.54688
0.5	5.71875	4.96875
0.6	4.85938	4.39063
0.7	4.45313	4.07813
0.8	4.32813	4.12500

5. Conclusions and Discussion

This paper presents the escape criteria for fractals (Mandelbrot sets) concerning hybrid Picard S-iteration and illustrates the novel fractal patterns. Fractals generated by the hybrid Picard S-iterative procedure exhibit drastic changes for varying values of α_1 and α_2 . Using Picard S-orbit, we were able to obtain more spontaneous or natural Mandelbrot sets than with fractals made using S-iteration. We have experienced the following observations:

- It is observed that in Figures 1–10, an obvious variation in the shapes created using the hybrid Picard S-orbit and S orbit for fixed value $\alpha_2 = 0.4$ and similar value of α_1 .
- Similarly in Figures 12–23, we fixed parameter $\alpha_1 = 0.7$ and varied parameter α_2 to obtain quadratic Mandelbrot sets. Mandelbrot sets created using the hybrid Picard S-iteration procedure are quite spontaneous from those created via S-iteration.
- It is seen that in time comparison Tables 1 and 2 and Figures 11 and 24, the hybrid Picard S-iteration procedure takes a short time to create a Mandelbrot set.
- The cubic Mandelbrot sets created in the hybrid Picard S-orbit and S-orbit with fixed parameter $\alpha_2 = 0.5$ also have an obvious variation in figures for varying parameter α_1 .
- In Table 3 and Figure 39, it is seen that for values $\alpha_1 \geq 0.5$, S-iteration takes a rather small amount of time as compared to Picard S-iteration to create cubic Mandelbrot sets. However, the preservation of beauty created by the latter is, foremost, better than the S-iteration.
- Quadratic Mandelbrot sets are symmetrical along the x -axis whereas cubic Mandelbrot sets are symmetrical along both the x -axis and y -axis.

It is evident that changes in the iteration scheme's orbit cause variations in the graphics. People interested in designing aesthetically pleasing patterns will find great satisfaction in the findings of this paper. For further future applications, this work can be combined with [52,53]. In future, comparisons can be made with other iterations and the process can be improved for higher-order polynomials and transcendental functions. Furthermore, a new Picard–Ishikawa hybrid iterative scheme [54] is analyzed for its fast convergence over the others. It proved useful in solving particular differential equations. It is remarkable that for $q_c(z_N) = z_N; \alpha_1 = 1 - \alpha_1$, in (5), Picard–Ishikawa hybrid iterative scheme [54] can be obtained. Hence, our results can be used further to generalize the results of [54] and explore more general differential equations.

Author Contributions: Conceptualization, R.S.; methodology, A.T.; software, R.M.K.; validation, R.S., A.T. and R.M.K.; formal analysis, R.S., A.T. and R.M.K.; investigation, R.S., A.T. and R.M.K.; resources, R.S., A.T. and R.M.K.; data curation, R.S., A.T. and R.M.K.; writing—original draft preparation, R.S., A.T. and R.M.K.; writing—review and editing, R.S., A.T. and R.M.K.; visualization, R.S., A.T. and R.M.K.; supervision, R.S., A.T. and R.M.K.; project administration, R.S., A.T. and R.M.K.; All authors have read and agreed to the published version of the manuscript.

Funding: The author extends the appreciation to the Deanship of Postgraduate Studies and Scientific Research at Majmaah University for funding this research work through the project number (ICR-2024-949).

Data Availability Statement: The research is theoretical in nature. As a result, no data were used.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

MS	Mandelbrot set
MSs	Mandelbrot sets
SO	S Orbit
PSO	Picard S-Orbit
PSIP	Picard S-Iteration Process

References

- Diao, K.; Butler, D.; Ulanicki, B. Fractality in water distribution networks: Application to criticality analysis and optimal rehabilitation. *Urban Water J.* **2021**, *18*, 885–895. [CrossRef]
- Beata, K.E.H.; Dariusz, K. Fractal-heuristic method of water quality sensor locations in water supply network. *Water* **2020**, *12*, 832. [CrossRef]
- Wang, S.; Ji, H.; Zhan, Y.; Li, H.; Li, P. Research on the Model Improvement of a DLA Fractal River Network. *IEEE Access* **2020**, *8*, 100702–100711. [CrossRef]
- Liu, S.-A.; Bai, W.-L.; Liu, G.-C.; Li, W.-H.; Srivastava, H.M. Parallel fractal-compression method for big video data. *Complexity* **2018**, *2018*, 2016976. [CrossRef]
- Martinez, F.; Manriquez, H.; Ojeda, A.; Olea, G. Organization Patterns of Complex River Networks in Chile: A Fractal Morphology. *Mathematics* **2022**, *10*, 1806. [CrossRef]
- Tehrani, D.H.T.; Solaimani, M. Persistent currents and electronic properties of Mandelbrot quantum rings. *Sci. Rep.* **2023**, *13*, 5710. [CrossRef]
- Taylor, R.; Spehar, B.; Hagerhall, C.; Van Donkelaar, P. Perceptual and Physiological Responses to Jackson Pollock's Fractals. *Front. Hum. Neurosci.* **2011**, *5*, 10034. [CrossRef]
- Mandelbrot, B.B. *The Fractal Geometry of Nature*; WH freeman New York: New York, NY, USA, 1982; Volume 2.
- Burger, E.B.; Starbird, M. *The Heart of Mathematics: An Invitation to Effective Thinking*; Springer Science Business Media: Berlin/Heidelberg, Germany, 2004.
- Holtzman, S.R. *Digital Mantras: The Languages of Abstract and Virtual Worlds*; Mit Press: Cambridge, MA, USA, 1995.
- Mitchell, K. The Fractal Art Manifesto. Online. Disponível em. 1999. Available online: <https://www.fractalus.com/info/manifesto.htm> (accessed on 8 December 2023).
- Li, D.; Shahid, A.A.; Tassaddiq, A.; Khan, A.; Guo, X.; Ahmad, M. CR iteration in the generation of antifractals with s-convexity. *IEEE Access* **2020**, *8*, 77214. [CrossRef]
- Dhurandhar, S.V.; Bhavsar, V.C.; Gujar, U.G. Analysis of z -plane fractal images from $z \rightarrow z\alpha + c$ for $\alpha < 0$. *Comput. Graph.* **1993**, *17*, 89–94.
- Lakhtakia, A.; Varadan, V.; Messier, R.; Varadan, V. On the symmetries of the Julia sets for the process $z \rightarrow z\alpha + c$. *J. Phys. A Math. Gen.* **1987**, *20*, 3533–3535. [CrossRef]
- Crowe, W.D.; Hasson, R.; Rippon, P.J.; Strain-Clark, P.E.D. On the structure of the mandelbar set. *Nonlinearity* **1989**, *2*, 541–553. [CrossRef]
- Domínguez P.; Fagella, N. Residual Julia sets of rational and transcendental functions. In *Transcendental Dynamics and Complex Analysis*; Cambridge University Press: Cambridge, UK, 2010; pp. 138–164.
- Peherstorfer, F.; Stroh, C. Connectedness of Julia sets of rational functions. *Comput. Methods Funct. Theory* **2001**, *1*, 61–79. [CrossRef]
- Koss, L. Elliptic functions with disconnected Julia sets. *Int. Bifurc. Chaos* **2016**, *26*, 1650095. [CrossRef]
- Liu, S.-A.; Xu, X.-Y.; Srivastava, G.; Srivastava, H.M. Fractal properties of the generalized Mandelbrot set with complex exponent. *Fractals* **2023**. [CrossRef]
- Katunin, A. Analysis of 4D hypercomplex generalization of Julia sets. In *Computer Vision and Graphics: International Conference, ICCVG 2016, Warsaw, Poland, 19–21 September 2016*; Proceedings 8; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 627–635.
- Dang, Y.; Kauffman, L.; Sandin, D. *Hypercomplex Iterations: Distance Estimation and Higher Dimensional Fractals*; World Scientific: Singapore, 2002.
- Griffin, C.; Joshi, G. Octonionic Julia sets. *Chaos Solitons Fractals* **1992**, *2*, 11–24. [CrossRef]
- Gdawiec, K. Inversion fractals and iteration processes in the generation of aesthetic patterns. In *Computer Graphics Forum*; Wiley Online Library: Hoboken, NJ, USA, 2017; Volume 36, pp. 35–45.
- Singh, S.L.; Mishra, S.; Sinkala, W. A new iterative approach to fractal models. *Commun. Nonlinear Sci. Numer. Simulat.* **2012**, *17*, 521–529. [CrossRef]
- Prasad, B.; Katiyar, K. Fractals via Ishikawa iteration. In Proceedings of the International Conference on Logic, Information, Control and Computation, Gandhigram, India, 25–27 February 2011; pp. 197–203.
- Gdawiec, K.; Kotarski, W.; Lisowska, A. Biomorphs via modified iterations. *J. Nonlinear Sci. Appl.* **2016**, *9*, 2305–2315. [CrossRef]

27. Kittiratanawasin, L.; Yambangwai, D.; Chairatsiripong, C.; Thianwan, T. An Efficient Iterative Algorithm for Solving the Split Feasibility Problem in Hilbert Spaces Applicable in Image Deblurring. *Signal Recover. Polynomiography J. Math.* **2023**, *2023*, 15. [[CrossRef](#)]
28. Rani, M.; Kumar, V. Superior Julia set. *J. Korea Soc. Math. Educ. Ser. D Res. Math. Educ.* **2004**, *8*, 261–277.
29. Rani, M.; Kumar, V. Superior mandelbrot set. *Res. Math. Educ.* **2004**, *8*, 279–291.
30. Rana, R.; Chauhan, Y.S.; Negi, A. Non linear dynamics of ishikawa iteration. *Int. J. Comput. Appl.* **2010**, *7*, 43–49. [[CrossRef](#)]
31. Chauhan, Y.S.; Rana, R.; Negi, A. New julia sets of ishikawa iterates. *Int. J. Comput. Appl.* **2010**, *7*, 34–42. [[CrossRef](#)]
32. Kang, S.M.; Rafiq, A.; Latif, A.; Shahid, A.A.; Kwun, Y.C. Tricorns and Multi-corns of S-iteration scheme. *J. Funct. Spaces* **2015**, *2015*, 1–7.
33. Ashish, M.R.; Chugh, R. Julia sets and mandelbrot sets in Noor orbit. *Appl. Math. Comput.* **2014**, *228*, 615–631. [[CrossRef](#)]
34. Li, D.; Tanveer, M.; Nazeer, W.; Guo, X. Boundaries of filled Julia sets in generalized Jungck-Mann orbit. *IEEE Access* **2019**, *7*, 76859–76867. [[CrossRef](#)]
35. 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* **2019**, *7*, 12167–12176. [[CrossRef](#)]
36. Kwun, Y.C.; Tanveer, M.; Nazeer, W.; Abbas, M.; Kang, S.M. Fractal generation in modified Jungck-S orbit. *IEEE Access* **2019**, *7*, 35060–35071. [[CrossRef](#)]
37. Gdawiec, K.; Shahid, A.A. Fixed point results for the complex fractal generation in the s-iteration orbit with s-convexity. *Open J. Math. Sci.* **2018**, *2*, 56–72. [[CrossRef](#)]
38. Nazeer, W.; Kang, S.M.; Tanveer, M.; Shahid, A.A. Fixed point results in the generation of julia and mandelbrot sets. *J. Equal. Appl.* **2015**, *2015*, 298. [[CrossRef](#)]
39. Cho, S.Y.; Shahid, A.A.; Nazeer, W.; Kang, S.M. *Fixed Point Results for Fractal Generation in Noor Orbit and S-Convexity*; SpringerPlus: Berlin/Heidelberg, Germany, 2016; Volume 5, p. 1843.
40. Zou, C.; Shahid, A.; Tassaddiq, A.; Khan, A.; Ahmad, M. Mandelbrot sets and Julia sets in Picard-Mann orbit. *IEEE Access* **2020**, *8*, 64411–64421. [[CrossRef](#)]
41. Prajapati, D.J.; Rawat, S.; Tomar, A.; Sajid, M.; Dimri, R.C. A Brief Study on Julia Sets in the Dynamics of Entire Transcendental Function Using Mann Iterative Scheme. *Fractal Fract.* **2022**, *6*, 397. [[CrossRef](#)]
42. Tomar, A.; Kumar, V.; Rana, U.S.; Sajid, M. Fractals as Julia and Mandelbrot Sets of Complex Cosine Functions via Fixed Point Iterations. *Symmetry* **2023**, *15*, 478. [[CrossRef](#)]
43. Tanveer, M.; Nazeer, W.; Gdawiec, K. On the Mandelbrot set of $z^p + \log c^t$ via the Mann and Picard-Mann iterations. *Math. Comput. Simul.* **2023**, *209*, 184–204. [[CrossRef](#)]
44. Devaney, R. *A First Course in Chaotic Dynamical Systems: Theory and Experiment*; Addison-Wesley: New York, NY, USA, 1982; Volume 2.
45. Mann, W.R. Mean value methods in iteration. *Proc. Am. Math. Soc.* **1953**, *4*, 506–510. [[CrossRef](#)]
46. Ishikawa, S. Fixed points by a new iteration method. *Proc. Am. Math. Soc.* **1974**, *44*, 147–150. [[CrossRef](#)]
47. Agarwal, R.; Regan, D.O.; Sahu, D. Iterative construction of fixed points of nearly asymptotically nonexpansive mappings. *J. Nonlinear Convex Anal.* **2007**, *8*, 61.
48. Gursoy, F. A picard-s iterative method for approximating fixed point of weak-contraction mappings. *Filomat* **2016**, *30*, 2829–2845. [[CrossRef](#)]
49. Tassaddiq, A.; Tanveer, M.; Azhar, M.; Arshad, M.; Lakhani, F. Escape Criteria for Generating Fractals of Complex Functions Using DK-Iterative Scheme. *Fractal Fract.* **2023**, *7*, 76. [[CrossRef](#)]
50. Tassaddiq, A.; Tanveer, M.; Azhar, M.; Nazeer, W.; Qureshi, S. A Four Step Feedback Iteration and Its Applications in Fractals. *Fractal Fract.* **2022**, *6*, 662. [[CrossRef](#)]
51. Tassaddiq, A. General escape criteria for the generation of fractals in extended Jungck-Noor orbit. *Math. Comput. Simul.* **2022**, *196*, 1–14. [[CrossRef](#)]
52. Tassaddiq, A.; Tanveer, M.; Israr, K.; Arshad, M.; Shehzad, K.; Srivastava, R. Multicorn Sets of $z^k + c^m$ via S-Iteration with h-Convexity. *Fractal Fract.* **2023**, *7*, 486. [[CrossRef](#)]
53. Srivastava, H.M.; Saad, K.M.; Hamanah, W.M. Certain New Models of the Multi-Space Fractal-Fractional Kuramoto-Sivashinsky and Korteweg-de Vries Equations. *Mathematics* **2022**, *10*, 1089. [[CrossRef](#)]
54. Okeke, G.A. Convergence analysis of the Picard-Ishikawa hybrid iterative process with applications. *Afr. Mat.* **2019**, *30*, 817–835. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.