



UNIVERSITY OF AMSTERDAM

Approximation of the area of the Mandelbrot set

STOCHASTIC SIMULATION - ASSIGNMENT 1

Authors:

Nadav Levi, 11806990,

nadav.levi@student.uva.nl

Yunxiang Li, 13189387,

yunxiang.li@student.uva.nl

Abstract

In this paper we have presented the Mandelbrot set; a set which generalizes the convergence of the rational iterative map $z \rightarrow z^2 + c$, and has a fractal shape with self-similarity at all scales. A method of approximating the area of the set was presented, using the Monte Carlo Integration Hit and Miss approach. A number of sampling methods were used, such as Random, Latin Hypercube and Orthogonal sampling methods. Followed by a number of simulations using the various sampling methods on a variety of sample sizes and number of iterations. A comparison of the convergence of each method was then presented. We then obtained an approximate result for the area of the Mandelbrot set and computed its confidence interval.

November 22, 2021

1 Introduction

We begin with the rational function given by $f_c = z^2 + c$, and the iterative map $z \rightarrow z^2 + c$, $z_n = f(f(\dots f(z_0) \dots))$. Now, an interesting idea to consider is for which values of z will this iterative function converge to some point - i.e., does there exist a constant $b \in \mathbb{R}$ such that $\{z_n\} \leq b$ for any $n \in \mathbb{N}$?

Example 1.1. For example, if we restrict ourselves to the one-dimensional real line such that $z \in \mathbb{R}$ and fix $z_0 = 0$, the iterative map f_c centered at c gives the following values:

$z_1 = z_0^2 + c$	$z_2 = (c^2) + c$	$z_3 = ((c^2) + c)^2 + c$	\dots	$z_{n \rightarrow \infty}$
1	$1 + 1$	$2^2 + 1$	\dots	∞
7	$49 + 7$	$3136 + 7$	\dots	∞
2	4	16	\dots	∞
$\frac{1}{2}$	$\frac{1}{4} + \frac{1}{2}$	$\frac{9}{16} + \frac{1}{2}$	\dots	∞
$\frac{1}{4}$	$\frac{16}{16} + \frac{1}{4}$	$\frac{25}{256} + \frac{1}{4}$	\dots	$\frac{1}{4}$
$\frac{1}{10}$	$\frac{1}{100} + \frac{1}{10}$	$\frac{121}{10000} + \frac{1}{10}$	\dots	$\frac{1}{10}$

As can be seen in the example above, for some values the map is stable and converges to some value on the line. However, for other values, such as 7 and 2, we see that with each iteration the value of z just grows, and is never bounded. In fact, the iterative map only converges for certain values. However, a more interesting approach would be to extend z and c to the complex plane.

In the context of Complex Dynamics, such iterative maps were first studied by Julia[6] and Fatou[4] with the Julia and Fatou sets, respectively in the early 20th century. However, during that time calculations that required large amounts of computational power were not possible, so a closer study of the structure of the sets was impossible and the problem was left in the dark for the decades to come[5]. It was not until the 1980's, when a mathematician by the name of Benoit Mandelbrot made contributions to the Fatou-Julia iteration of the quadratic map $z^2 + c$ while working at IBM[8], in what resulted as the *Mandelbrot Set*:

Definition 1.2. Let M denote the Mandelbrot set, which consists of the set of all complex numbers $c \in \mathbb{C}$ such that the sequence $\{z_n\}$ defined by $z_n := z_{n-1}^2 + c$ with initial value given by $z_0 = 0$ is bounded for all $n \geq 0$.

In essence, the Mandelbrot set is a generalized version of Example 1.1, in that it generalizes all the different values for which the iterative map $z^2 + c$ does not diverge to infinity. Hence, it is the set of values of c in the complex plane for which the orbit of the critical point z_0 under iteration of the quadratic map converges

Mandelbrot took a rather unique approach while studying the problem of the iterative map, and chose to visualise the data. What he found was that the set was a fractal - A structure with self similarity at all scales[5] which can be seen below in Figure 1:

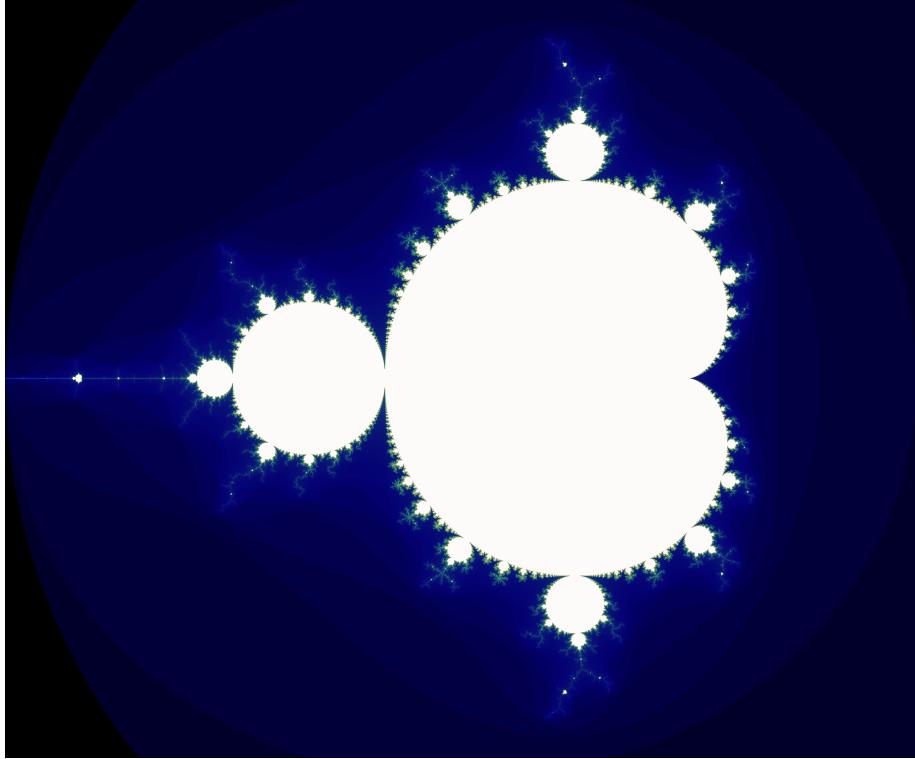


Figure 1: Visualization of the Mandelbrot set

In contrast to the sheer simplicity of formulating the Mandelbrot set, the juxtaposition of the complex structure and beauty of the visualization of the set shown in figure 1 is fascinating.

In the following text, our focus will be on the Mandelbrot set. However, we will explore a slightly different aspect of the set; namely, its area. Given its fractal structure, much about the area of the set M remains to be understood. Previously, it was found that the area of the set was approximately 1.506484[10]. The most common way of approximating the area is by using the Monte Carlo pixel counting approach, for example in [10]. It has however a number of limitation, such as the computation power required to increase the accuracy of the results. In contrast, Bittner et al [2] suggested a new theoretical method of calculating the area of the Mandelbrot set by considering its complement inside the Riemann sphere, which yields an area of approximately 1.68288. This surprising discrepancy can be attributed to the fact that the boundary of the Mandelbrot set has a Hausdorff dimension of 2 - Meaning it could have a positive, non zero area(the boundary) which the pixel counting method is not able to detect, so their potential contribution to the area would never be included in the computation. At the moment of writing, this is still a speculation and remains an active area of research[2].

For a more comprehensive review on the possible approaches, refer to [1].

In our attempt to come up with an approximation of the area, we will be taking the pixel counting approach, and use a Monte Carlo Integration method, known as *Hit and Miss*. A number of sampling methods will be used in order to generate random variables, which will be described in the following chapters.

2 Methods

After introducing the Mandelbrot set in the previous chapter, an interesting question to investigate is the area of the set. We denote by A_M the area of the Mandelbrot set. One approach of doing so involves the Monte Carlo simulation methods - A set of numerical methods yielding an estimate $A_{i,s}$ of the area A_M of the Mandelbrot set, where i refers to the number of iterations and s to the number of samples drawn, respectively. In particular, we will be using the *Hit and Miss* method - On a predefined search area that covers the area A_M , we draw a sample of size s of random points. Then, in order to approximate the area, we will compute the ratio of points in the Mandelbrot set against the total number of points. Then, multiplying the total search area with the ratio would yield an approximate figure for the area of the Mandelbrot set.

Specifically, we set the x on the real axis expands from the left $x = -2$ to right $x = 0.5$ and y on the imaginary axis expands from the bottom $y = -1.2$ to top $y = 1.2$. Whether a sample is in the Mandelbrot set is decided by if $|Z_n| < 2$ after i times iterative computation of $Z_{n+1} = Z_n^2 + c$ ($Z_0 = 0$).

The *Monte Carlo* method is a statistical approach to the study of differential equations and was first formulated in 1949[9]. Subsequently, the range of problems that the method could solve effectively was extended with the advent of modern computers and increase in computational power[12].

In the following section we will describe how the Monte Carlo method can be utilised for numerical integration using random numbers.

2.1 Monte Carlo Integration

We begin by describing the basic idea of numerical integration using the Monte Carlo Hit and Miss approach.

Let $f(x)$ be some rational function on the subset $\Omega = [a, b] \times [c, d] \subset \mathbb{C}$. Note that Ω , is a simple rectangle and computing its area is straightforward. We would then be interested in approximating the area underneath the curve $f(x)$ by $A = \int_a^b f(x)dx$. So in essence, Monte Carlo integration allows us to approximate such area by generating random variables(points) on our grid given by Ω . Then, with a large enough sample, we can simply count the number of points underneath $f(x)$, and divide it with the total number of points. Multiplying this ratio with the total area of Ω would yield an approximation of A .

Note that at this point we purposefully choose not to address the method of generating the random variables, as this will be the topic of the following subsection.

For illustration purposes of the Monte Carlo Integration method, consider the following simple example:

Example 2.1. If we take $f(x) = x$ and $\Omega = [0, 1] \times [0, 1] \subset \mathbb{R}^2$, we would need to find the area given by $A = \int_0^1 xdx$ with the Monte Carlo approach. By simple integration, we know that the area of A is equal to $\frac{1}{2}$. Then, we would simply generate a large amount of random variables and find the ratio of the points that fall underneath the line of $f(x) = x$ and the total number of points. Multiplying this ratio by the total area of Ω , which in this case is 1, would yield an estimate of the area A .

This idea is formalized with the *Monte Carlo Estimator*

Definition 2.2. Let S denote the number of samples drawn, and let X_i denote a random variable. Then, the area is given by

$$A_S = (b - a)(d - c) \frac{1}{S} \sum_i^S f(X_i) \quad (1)$$

In the case of the Mandelbrot set, (1) becomes:

$$A_{i,s} = (b - a)(d - c) \frac{1}{s} \sum_j^s f(z_j^i) \quad (2)$$

Where z^i denotes the i th iteration of the map $f_c(z) = z^2 + c$. Note that the set M must be fully enclosed by Ω .

Corollary 2.2.1. *The Law of Large Numbers gives us that as S tends to infinity, the Monte Carlo Estimator converges in probability to A . Hence,*

$$\mathbb{P}\left(\lim_{s \rightarrow \infty} A_s = A\right) = 1 \quad (3)$$

Additionally, as the variance of the Monte Carlo Estimator is given by $\text{Var}(A_s) = \frac{\sigma^2}{S}$ where σ^2 denotes the variance of the function f . Therefore, by increasing the sample size by 4, the error variance of the estimator is reduced by half.

For a more comprehensive review on the topic, refer to [3].

2.2 Sampling methods

In our discussion of the Monte Carlo Integration method, we have often mentioned the use of random variables. Before we begin with the description of the various sampling methods which were used, a brief definition of a random variable is given:

Definition 2.3. A random variable X is a measurable function $X : \Omega \rightarrow E$ from the set of possible outcomes Ω to a measurable set E . The probability that X takes on a value in $S \subset E$ is given by:

$$\mathbb{P}(X \in S) = \mathbb{P}(\{\omega \in \Omega | X(\omega) \in S\}) \quad (4)$$

In the context of the Monte Carlo Integration of the Mandelbrot set, we have the following:

- The function is $f : \mathbb{C} \rightarrow \mathbb{C}$ and given by $f(z) = z^2 + c$
- The probability that f takes values in $M \subset \Omega \subset \mathbb{C}$ is:

$$\mathbb{P}(z \in M) = \mathbb{P}(\{z \in M | f(z) \in \Omega\})$$

To begin with, we will simulate the area using three different random variable sampling techniques. These are: Random sampling, Latin Hypercube sampling, and Orthogonal Sampling.

The first method of Random sampling is the most basic of the three. In essence, it simply means that we generate a point in our subset of the complex plane, $z \in \Omega$. Then, each consecutive point in our sample is independent from the previous point, and all possible points on the grid are equally likely to be drawn.

Then, the concept of a Latin hypercube is a bit more intricate.

Definition 2.4. A Latin Hypercube is a square grid of size $n \times n$ containing sample positions where each sample is placed on a unique row and column.

Hence, when drawing one sample in a Latin Hypercube grid, we need to remember the row and column of the sample. Then, every subsequent sample must be on a new, previously unused row and column.

In contrast, with *Orthogonal Sampling*, we first divide the sampling space into equal subsets. For example, on a 9×9 grid, we have 9 unique 3×3 grids. Then, we draw our predetermined number of samples simultaneously, ensuring that the resulting sample space is a Latin Hypercube.

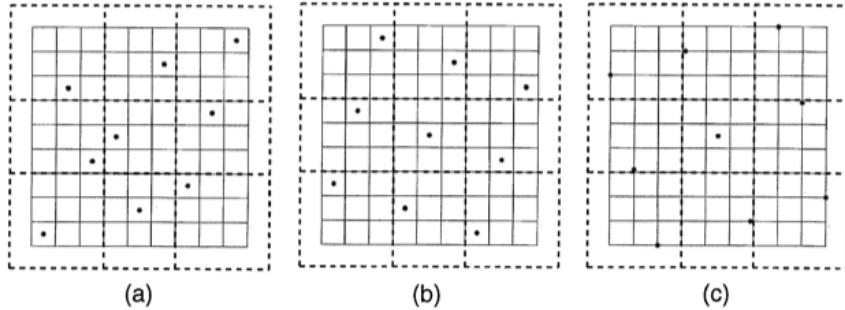


Figure 2: (a) Random sampling, (b) Latin Hypercube, (c) Orthogonal sampling. From Optimal orthogonal-array-based Latin hypercubes[7]

Putting this all together - By using the Latin Hypercube, we ensure that the drawn random numbers in our sample are representative of the real variability. With Orthogonal sampling, the resulting sample space is a good approximation of the real variability, whereas pure random sampling is simply a set of random numbers. We have presented a basic idea of the concepts of each sampling method. However, for a more comprehensive review on the subject, using mathematical notation in terms of permutations, a good review can be found in [7, 11].

Antithetic variates method are used to decrease the variance of the results. The two antithetic variables conform to the same distributions and have negative covariance. Regarding the same number of samples, the computational effort is reduced by half compared with the common approach.

3 Results

3.1 Investigating the proper number of iterations

We first fixed the number of samples $s = 10000$ to investigate the convergence behaviour of area as the number of iterations i was increased. Different i was tested from 100 to 10000 in the steps of 100. The basic pure random sampling approach was taken and each combination of i and s was simulated by 100 runs.

Figure 3 demonstrates the mean values over simulations and the error bands with confidence intervals set to 0.95. The area $A_{i,s}$ computed with the highest $i = 10000$ and $s = 10000$ was considered as the baseline to compare with. In order to obtain a more reliable computation for the area of highest i and s , another 900 simulations (1000 in total) were taken. The result of area is 1.5608 and plotted as a red dotted line in the figure. The estimation of area fast decreases for smaller i and begins to fluctuate around the $A_{i,s}$ in later period after approximately $i = 2000$.

The next step is to investigate s with a fixed i . Thus a reasonable optimal setting of i with feasible computing power is required. The time consumed per each simulation and errors compared with $A_{max(i),s}$ are plotted in figure 4 and figure 5. Since the computing time increases linearly, and no obvious difference shows for $i > 2000$. $i = 2000$ is chosen

for further study. The error of $A_{i=2000, s=10000}$ is 0.0008, which lies in the 95% confidence interval $\bar{X} \pm \frac{1.96S}{\sqrt{n}}$, (1.5013, 1.5116).

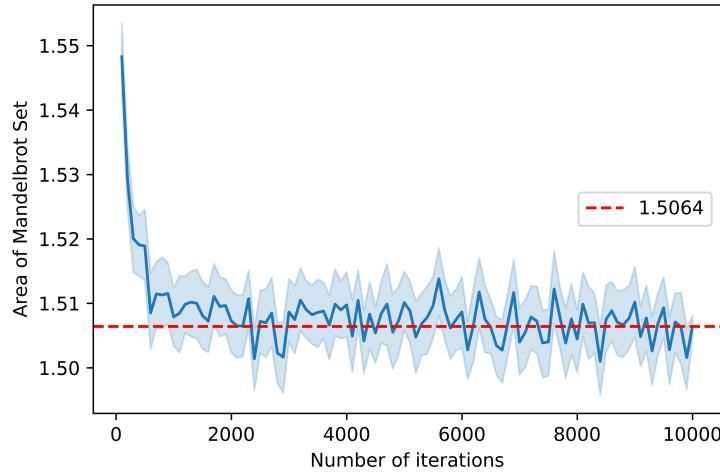


Figure 3: Investigating $A_{j,s}$ for $\forall j < i = 10000$ with $s = 10000$. 100 simulations for each i.

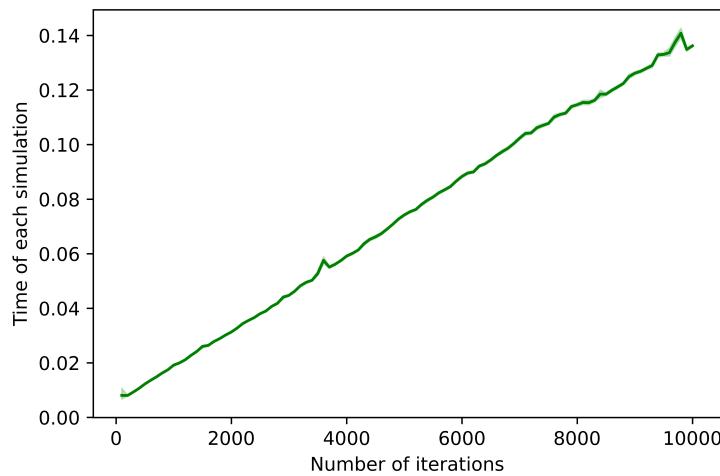


Figure 4: Investigating time consumed for $\forall j < i = 10000$ with $s = 10000$. 100 simulations for each i.

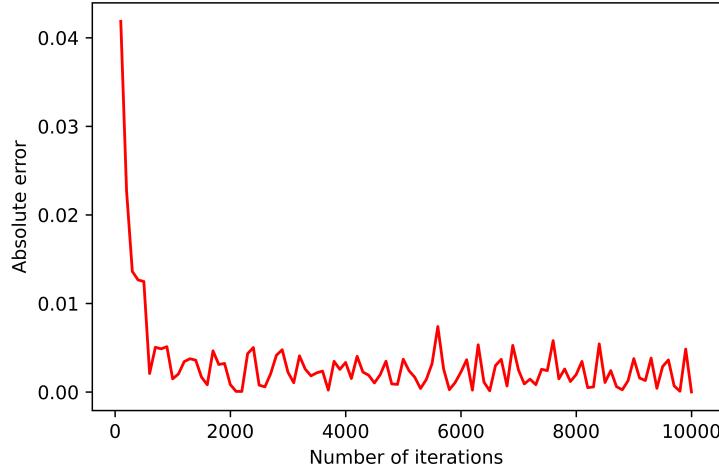


Figure 5: Investigating errors compared to $A_{\max(i),s}$ for $A_{j,s} \forall j < i = 10000$ with $s = 10000$. 100 simulations for each i .

3.2 Investigating the area for increasing samples

In the second part, We looked at the outcome of the simulations as the number of samples increased. The area of Mandelbrot set was calculated for each varied s from 1000 to 100 000 in the steps of 1000 with fixed $i = 2000$. Again, another 900 simulations (1000 in total) were taken to acquire a more accurate estimation for the area with the highest i and s thus $A_{i,\max(s)}$ was updated. Figure 6 shows the result and convergence behaviour with varied s . The result shrinks quickly before $i = 10000$, and fluctuates around the $A_{i,\max(s)}$ afterwards.

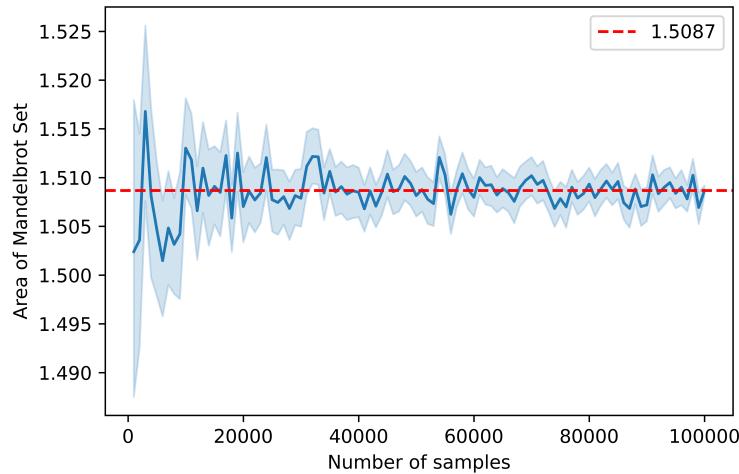


Figure 6: Investigating $A_{i,n}$ for $\forall n < s = 100000$ with $i = 2000$. 100 simulations for each number of samples n .

3.3 Comparing different sampling techniques

Another two different methods, Latin hypercube sampling and Orthogonal sampling were implemented to compare the quality of results. Investigating all three methods with fixed $i = 2000$, s from 100 to 10000 in the steps of 100. 100 simulations were taken for each different s . Figure 7 shows the convergence behaviour for three sampling methods. The three area $A_{max(i),s}$ for each method represented as red dotted lines were the averages over 1000 simulations.

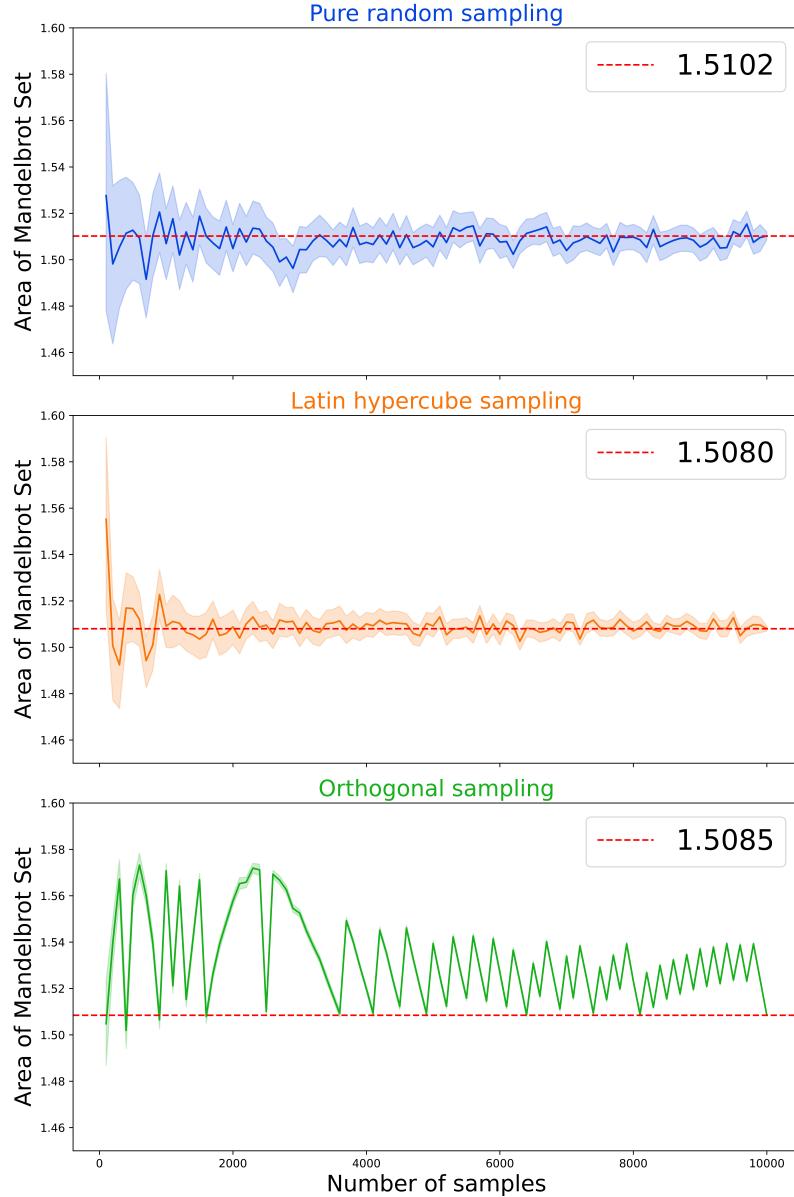


Figure 7: Investigating $A_{i,n}$ for $\forall n < s = 10000$ with $i = 2000$ through three sampling methods. 100 simulations for each number of samples n .

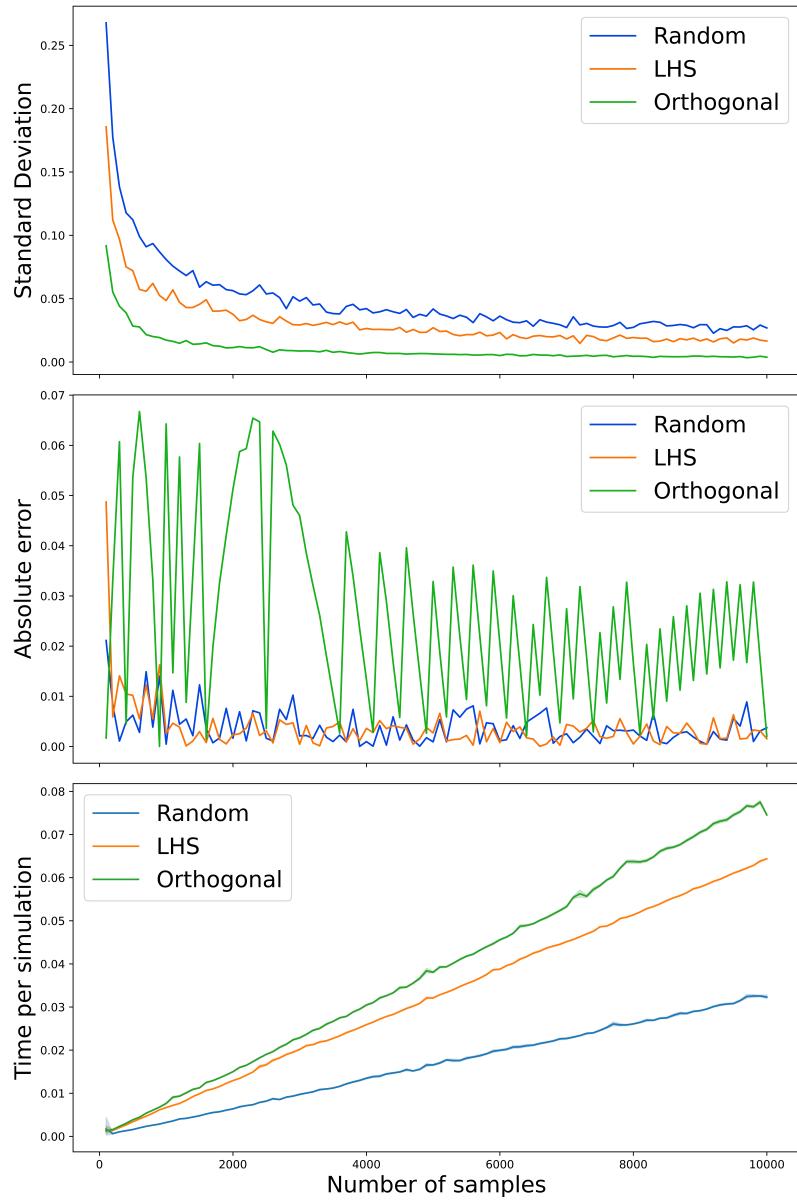


Figure 8: Investigating standard deviations, errors, and times costs of $A_{i,n}$ for $\forall n < s = 10000$ with $i = 2000$ through three sampling methods. 100 simulations for each number of samples n .

Table 1: Summary of estimations.

Sampling methods	i	s	A	SD	CI 95%	\bar{t}
Pure Random	10000	10000	1.5064	0.0262	(1.5013, 1.5116)	0.1362
Pure Random	2000	100000	1.5087	0.0080	(1.5071, 1.5103)	0.3273
Pure Random	2000	10000	1.5102	0.0268	(1.5050, 1.5155)	0.0323
Latin hypercube	2000	10000	1.5080	0.0165	(1.5048, 1.5113)	0.0644
Orthogonal	2000	10000	1.5089	0.0038	(1.5077, 1.5092)	0.0745
Antithetic Random	2000	10000	1.5127	0.0212	(1.5085, 1.5169)	0.0653

As the results compared in 8 and table 1, Latin hypercube sampling and Orthogonal sampling indeed produce results with much lower standard deviations as expected. Their samples are distributed uniformly in the space and even in subspaces for Orthogonal methods. The estimations given by these two methods are closer to the $A = 1.506484[10]$ from the previous study. The computational cost are denoted by average time cost per each simulation, $\bar{t}_{orthogonal} > \bar{t}_{lhs} > \bar{t}_{random}$ as expected. The periodically oscillation with large amplitude is unexpected for Orthogonal sampling.

3.4 Improving results

Antithetic variables were implemented to test if this methods could further improve the convergence rate of the Monte Carlo aproach. The area of Mandelbrot set was calculated for each varied s from 100 to 10000 in the steps of 100 with fixed $i = 2000$.Another 900 simulations (1000 in total) were taken to acquire a more accurate estimation for the area with the highest i and s . The standard deviations, errors compared to 1.506484[10] were demonstrated in figure 9. As the plots show, random sampling with antithetic variables does not obtain a better estimation of the area. But it tends to have a smaller variance and improves the convergence rate in limited extent since the area of Mandelbrot set is not a monotone function.

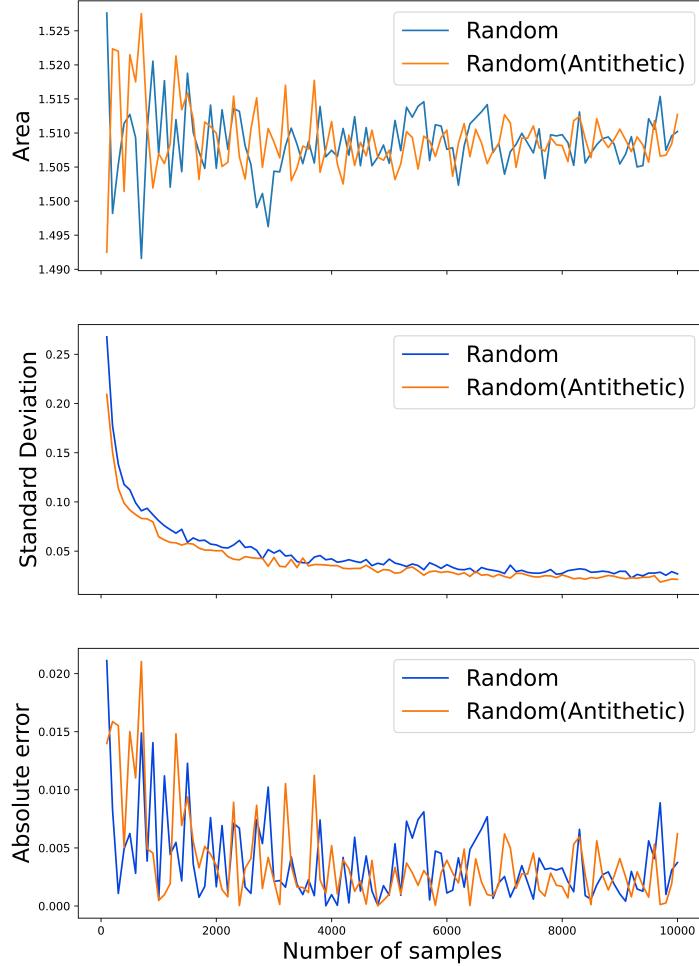


Figure 9: Comparing Antithetic variables sampling with pure random sampling through standard deviations, errors of $A_{i,n}$ for $\forall n < s = 10000$ with $i = 2000$. 100 simulations for each number of samples n .

4 Discussion

As the previous sections show, the convergence behaviours were tested with fixed number of samples as the number of iteration increased and vice versa. A reasonably good i was found to study the influence of larger number of samples. The results indicate that the pure random sampling method has limited improvement regarding the convergence rate and the accuracy of estimation around $i > 2000, s > 20000$. Then two other sampling methods were tested and compared with the random one. They have much less

standard deviations because samples are more evenly distributed from the lower number of samples to the larger settings. The Orthogonal sampling generate equal number of samples in each subspace therefore their standard deviations are quite close to zero. However, a clear oscillation of orthogonal is demonstrated in the plots. It could due to potential correlations between samples during the generating process thus enlarge the amplitude of convergence. In the end, Antithetic variables brings limited improvement on convergence behaviour. The standard deviations do not shrink as \sqrt{s} because of the non-monotonicity of the function. A further study on the oscillated behaviour of Orthogonal sampling and Monte Carlo sampling with Low-discrepancy sequence could be carried on.

References

- [1] Valentín Albillo. Boldly going - mandelbrot set area, 2020.
- [2] Daniel Bittner, Long Cheong, Dante Gates, and Hieu Nguyen. New approximations for the area of the mandelbrot set. *Involve, a Journal of Mathematics*, 10(4):555–572, 2017.
- [3] Russel E. Caflisch. Monte carlo and quasi-monte carlo methods. *Acta Numerica*, 7:1–49, 1998.
- [4] Pierre Fatou. Sur les substitutions rationnelles. *Comptes Rendus de l'Académie des Sciences de Paris*, (164):806–808, 1917.
- [5] Bastian Fredriksson. An introduction to the mandelbrot set, June 2015.
- [6] Julia Gaston. Mémoire sur l'itération des fonctions rationnelles. *Journal de Mathématiques Pures et Appliquées*, (8):47–245, 1918.
- [7] Stephen Leary, Atul Bhaskar, and Andy Keane. Optimal orthogonal-array-based latin hypercubes. *Journal of Applied Statistics*, 30(5):585–598, 2003.
- [8] Benoit B. Mandelbrot, Evertsz C J G., and M. C. Gutzwiller. *Fractals and chaos: The Mandelbrot set and beyond*. Springer, 2011.
- [9] Nicholas Metropolis and S. Ulam. The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341, 1949.
- [10] Kerry Mitchel. A statistical investigation of the area of the mandelbrot set, 2001.
- [11] Art B. Owen. Orthogonal arrays for computer experiments, integration and visualization. *Statistica Sinica*, 2(2):439–452, 1992.
- [12] Oleg Yavoruk. How does the monte carlo method work?, 09 2019.