University of Amsterdam

Assignment 1

Estimating the area of the Mandelbrot set

November 30, 2019

Students:
Marcus van Bergen
10871993

Steven Raaijmakers
10804242

Tutor:
Dongwei Ye

Group:
20

Lecturer:
A. Hoekstra

Course:
Stochastic Simulation

1 Introduction

Sampling methods are widely used to estimate the value of certain phenomena such as *pi*. Sampling also proves to be valuable in estimating integrals or areas of fractals. Previous research has shown that pure sampling methods, such as Monte Carlo sampling, and stratified sampling methods, such as Latin Hyper-cube, can positively effect the efficiency and accuracy while estimating in a specific context [3].

In this research we will experiment with several probabilistic sampling methods to approximate the area of the Mandelbrot set. Our objective is to preform an analysis of different sampling methods in order to determine their accuracy and efficiency.

This paper will start with providing background information (section 2) on the Mandelbrot set and different sampling methods. Afterward, we describe the appliance of the sampling methods to estimate the are of the Mandelbrot set (section 3) after which we preform an analysis on the accuracy and efficiency of the sampling methods. Section 4 contains the results of which we draw conclusions (section 5). The last section of this paper will contain a discussion (section 5.1) where we elaborate on the limitations of our testing and the importance sampling methods can have on the computational time and accuracy of a model.

2 Background Material

2.1 The Mandelbrot set

The Mandelbrot set is a set of points in the complex plane. Each of the points in the complex plane, c, belongs to the Mandelbrot set if and only if:

$$\lim_{n\to\infty} \left\| z_{n+1} = z_n^2 + c \right\| \not\to \infty \text{ where } z_0 = 0. \tag{1}$$

Equation 1 describes an Iterated Function System (IFS) which can be used to determine whether a point from the complex plane lies within the Mandelbrot set. Given a complex point c and $z_0 = 0$ the IFS can be repeated from $n \to \infty$. Each iteration the Euclidean norm of z_n is calculated. If $\|z_n\| \geq 2$ then z_n will eventually trend towards ∞ meaning c will not be in the Mandelbrot set [2]. However given a c where $\|z_n\|$ stays bounded, i.e. $\|z_n\| < 2$ (also known as the bailout radius), then c is part of the Mandelbrot set.

The complex plane is a two dimensional plane containing the points in the Mandelbrot. The x-axis of the complex plane is the real part of the complex numbers, and the y-axis the imaginary part. Because $z_0 = 0$, we know that a c outside of $(-2, 2)$ real and $(-2, 2)$ imaginary, will trend towards infinity following Equation 1. Per definition these numbers will not be part of the Mandelbrot set, hence our sampling of complex takes place within boundaries: $(-2, 2)$ real and $(-2, 2)$ imaginary [2].

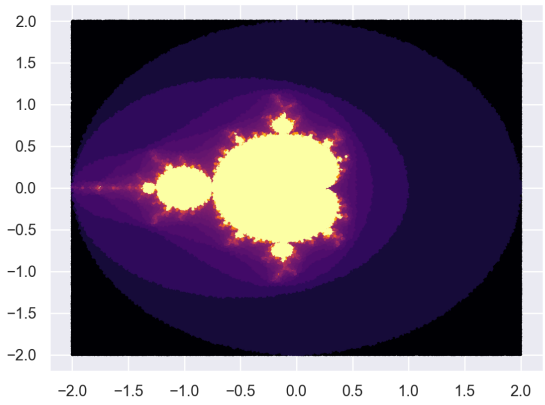


Figure 1: Estimation of the area of the Mandelbrot set ($N_{samples} = 1,000,000$ and $N_{iterations} = 1000$)

Figure 1 visualizes the Mandelbrot set using Monte Carlo sampling. Monte Carlo generates 1 million random points c, e.g. $N_{samples} = 1,000,000$, in the domain after which the IFS (described in equation 1, for $N_{iterations} = 1000$) is evaluated over each point. The shading of each point c in the complex plane is given by the number of iterations that it stays bounded for the condition $\|z_n\| < 2$. The black points almost immediately break the bailout radius condition with $n \ll N_{iterations}$, while the yellow shaded points stay bounded for $N_{iterations} = 1000$.

2.2 Sampling Methods

2.2.1 Monte Carlo sampling

Monte Carlo sampling is a pure random sampling method often used when a conventional numerical method is too complex to compute or too computationally intensive. Quite often it is possible to use Monte Carlo sampling to achieve a fairly accurate approximation of the desired method without being too computationally expensive. The Monte Carlo method generates points that are pseudo-random and are independent and identically distributed (i.i.d.).

In Figure 2a 50 two-dimensional points generated with Monte Carlo sampling are visualized.

2.2.2 Quasi-Monte Carlo Sampling

A Sobol sequence is a method which can be used to generate quasi-random low-discrepancy sequences. In general, low discrepancy sequences are not random but instead deterministic. In our case, it's possible to reintroduce a randomness factor to the sequence in order to be able to estimate variances, and error (shown later in this paper). Adding randomness to these low-discrepancy sequences allows us to define them as Randomized Quasi-Monte Carlo.

Given a point-set generated by the Sobol sequence $X = \{x_1 \dots x_j\}$, we can sample (with Monte Carlo) random points as follows $U = u_1 \dots u_j$. The Randomized Quasi-Monte Carlo (QMC) points can now be calculated according to: $y_j = x_j + U_j (\text{mod} 1)$. In other words all points generated by the Sobol sequence are randomly shifted according to the Monte Carlo points [4]. The (mod) 1 operator makes sure the points stay within range 0-1.

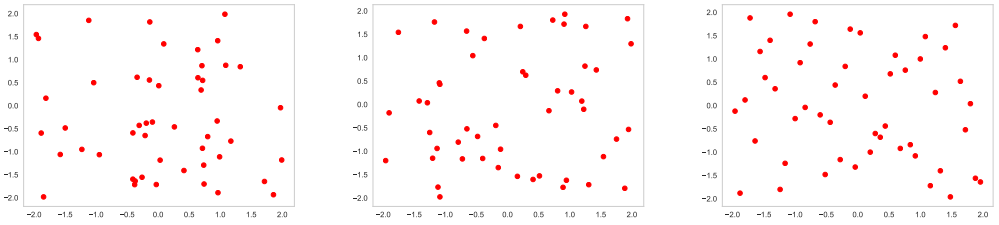
In Figure 2b we can see the spread of 50 QMC points generated by the procedure described above. The points are then linearly translated to a two dimensional plane with coordinates $(-2, 2)$ and $(-2, 2)$.

2.2.3 Latin Hyper-cube sampling

Latin Hyper-cube sampling (LHS) is a form of stratified sampling, which generates evenly spaced samples. It therefore differs from pure random sampling where points are drawn i.i.d.

LHS draws samples within a certain domain. In the case of sampling from a two-dimensional plane, we split the two dimensional plane into n columns and n rows where $n = n_{samples}$. We then sample $n_{samples}$ from this two dimensional where each n_i is in its own row and its own column.

In Figure 2c the spread of 50 points drawn using LHS is shown.



(a) Monte Carlo (pseudo random) (b) Randomized Quasi-Monte Carlo (Sobol sequence + Monte Carlo shifting). (c) Latin Hyper-cube sampling

Figure 2: Spread of 50 points drawn via three different methods.

Figure 2 shows the different behavior between pseudo random samples generated in 2a, the QMC samples in 2b and the LHS samples in 2c. The *quasi* of Quasi-Monte Carlo references to the low-discrepancy of the number sequences, meaning that they should be more uniformly spread over the sampled space. Pseudo random numbers often cluster together, which we also see in Figure 2a; happening between (-0,5 and 1.0) x and (-2.0, 1.5) y. In general, forcing the sampling method to be more uniformly spread over the space, makes the samples lose randomness and independence of one another. Both the QMC and LHS sampling methods have this effect of lowered-discrepancy. However, this same uniformly spread sampling can aid in lowering the variance of the experiment outcomes or allowing the same accuracy of the sample mean while needing less samples.

2.3 Central limit theorem

The samples in the Monte Carlo method are independent and identically distributed (i.i.d.) random variables which allows us to apply the central limit theorem (CLT). It states that for $X_1, X_2, ...$ being i.i.d. with standard deviation σ and expectation μ [1]:

$$\lim_{n\to\infty} P\left(\frac{\sum_{k=1}^{n} X_k - n\mu}{\sigma\sqrt{n}} \leq z\right) = \Phi(z), \tag{2}$$

with Φ being the CDF of the normal distribution.

3 Methods

We estimate the area of the Mandelbrot set using earlier described sampling methods:

1. Monte Carlo sampling,
2. Quasi Monte Carlo sampling,
3. Latin Hyper-cube sampling.

Using the different methods, we generate a set of $N_{samples}$ random points in a fixed domain. For all points we count the amount that lie within the area spanned by the Mandelbrot set N_{hits} according to equation 1. The ratio between N_{hits} and $N_{samples}$ will give us information about the Mandelbrot area since $N_{hits}/N_{samples} \sim area_{mb}$. Multiplying the ratio by the area of the domain all points were drawn from will yield an approximation of the area of the Mandelbrot set: $(N_{hits}/N_{samples}) * area_{domain} \approx area_{mb}$.

For our experiments the domain will be fixed to $-2 < x < 2$ and $-2 < y < 2$ (as described in section 2), spanning an area of $area_{domain} = 16$.

All code is written in Python3 using a combination of NumPy [1] and Numba [2]. Numba in combination with NumPy allows to parallelize parts of the methods limiting the computational time. Since we are working with stochastic simulations we repeat every area estimation experiment 20 times. Each repetition will be saved into a database in order for easy access while also assuring that for different plots we use the same data.

3.1 Accuracy as a function of number of iterations and number of samples

The three different sampling methods will be compared in order to define their accuracy. Since we do not have the exact value of the area of the Mandelbrot set we cannot determine the real error for our estimations. Instead, we use a base area $A^x_{i,s}$, with A being the estimated area of the Mandelbrot set with i iterations ($= N_{iterations}$) of which s samples ($= N_{samples}$) are drawn.

To study the accuracy as a function of the number of iterations we compute $A_{j,s}$ for $\forall j < i$ and examine the error between $A^x_{i,s}$ and $A^x_{j,s}$ for different values of j. Since we are also interested in the error for a function of $N_{samples}$ for different values of $N_{samples}$ we look into the error between $A_{i,r}$ and $A_{i,s}$ for $\forall r < s$.

3.2 Central limit theorem

To prove the CLT described in section 2.3. The CLT states that when $N \to \infty$, with N being the sample size, the sample means will approach a normal distribution. For our experiment this will mean that when we take infinite amount of under-, normal-, and over-estimations, the mean of the sample means will give a good approximation of the real value.

We choose to run an experiment where we draw $1,000$ samples, of sample size $10,000$ of a Monte Carlo simulation with $N_{samples} = 10$. Because we throw only 10 points in the domain, we expect the estimations to be inaccurate. However, since we repeat this $1,000 * 10,000$ times the sample means should approximate the normal distribution according to the CLT.

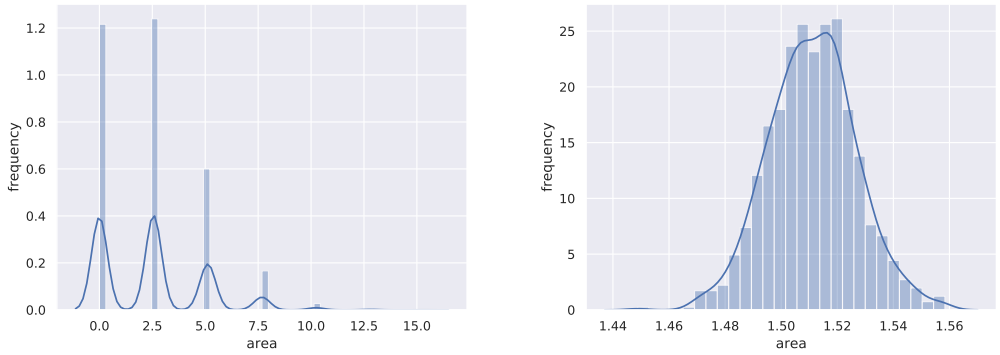
[1] https://numpy.org/
[2] http://numba.pydata.org/

4 Results

Due to the stochastic nature of our simulations we take 20 repetitions of each experiment. Where appropriate, the lines denote the mean of these repetitions while opaque lines denote the corresponding standard deviation.

4.1 Central Limit Theorem

The result of the experiment for applying the CLT to the MC simulation is shown in Figure 3. Figure 3a visualizes the distribution of one sample and since the area of the Mandelbrot ≈ 1.51 it shows that the estimations of one sample are inaccurate. However, in Figure 3b we see if we take $1,000$ of such inaccurate estimations, with sample size 10.000, the sample means will form a normal distribution, confirming the CLT.



(a) One sample.

(b) Means of 1,000 samples with sample size 10,000.

Figure 3: Distributions for area estimation by MC ($N_{samples} = 10$ and $N_{iterations} = 1000$) of the Mandelbrot set.

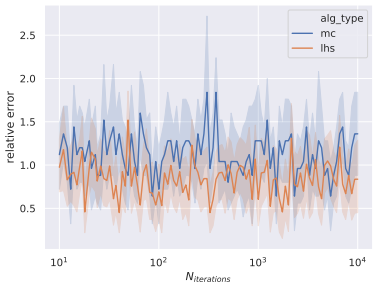
4.2 Accuracy

In Figure 4 we visualize multiple plots showing the comparison of MC versus LHS and MC versus QMC as a function of $N_{iterations}$. In this experiment we also choose to vary $N_{samples}$ in order to examine the performance of the methods with respect to each other for $N_{samples} = \{10, 100, 1000\}$.

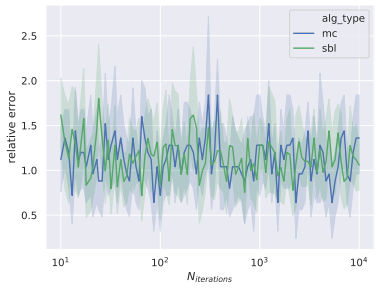
First, we see for $N_{samples} = 1000$ the mean relative error and its standard deviation are smaller in comparison to $N_{samples} = \{10, 100\}$ especially for $N_{iterations} > 10^2$.

Thereafter we note that the different values for $N_{samples}$ have little to no influence on the performance of the methods with respect to each other. MC and QMC preform equal for different $N_{samples}$ while LHS keeps a slight advantage.

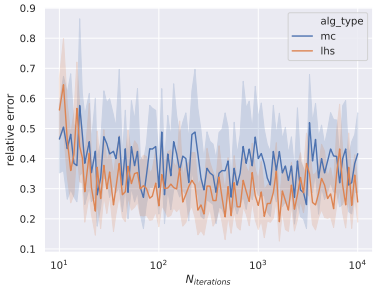
Furthermore, increasing $N_{samples}$ does have its effect on both the standard deviation and the mean of the relative error leading to significant lower values for higher $N_{samples}$.



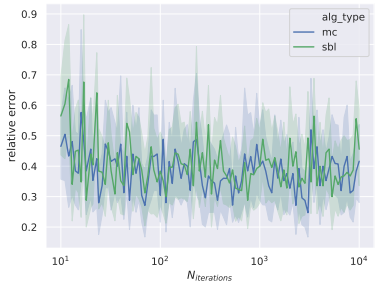
(a) MC vs LHS ($N_{samples} = 10$)



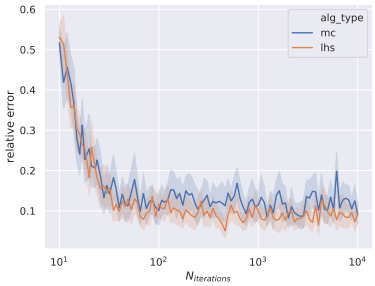
(b) MC vs QMC ($N_{samples} = 10$)



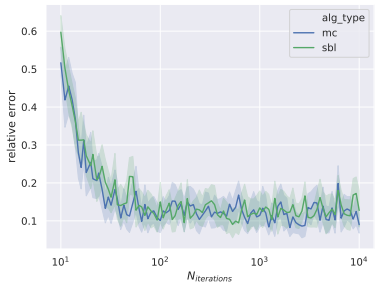
(c) MC vs LHS ($N_{samples} = 100$)



(d) MC vs QMC ($N_{samples} = 100$)



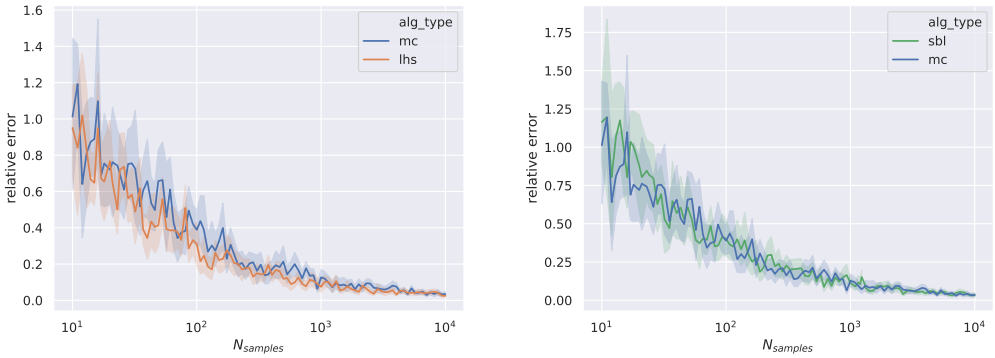
(e) MC vs LHS ($N_{samples} = 1,000$)



(f) MC vs QMC ($N_{samples} = 1,000$)

Figure 4: The relative error as a function of $N_{iterations}$ for different sampling methods with different $N_{samples}$ estimating the area of the Mandelbrot set.

In Figure 5 the relative error for different sampling methods as a function of $N_{samples}$ is shown, for fixed $N_{iterations} = 1,000$. We see that stratified sampling in the form of LHS preforms better than MC, while surprisingly MC and QMC have equal performance. For large $N_{samples}$ it shows that the relative error is shrinking which is also demonstrated in 4. Furthermore, this plot shows that as more $N_{samples}$ are being taken, the variance in the estimated area is also decreasing (shown by the opaque lines).



(a) MC vs LHS (b) MC vs QMC

Figure 5: The relative error as a function of $N_{samples}$ for different sampling methods estimating the area of the Mandelbrot set with $N_{iterations} = 1,000$.

5 Conclusion

In the context of relative error as a function of $N_{iterations}$, shown in Figure 4, we see LHS performs consistently better than both QMC and MC. As $N_{iterations}$ gets larger, till about 10^2, we see the relative error for all sampling algorithms getting smaller. Moreover, given the same test with a larger value for $N_{samples}$ the performance of all sampling algorithms towards each other stays consistent; with LHS generally having the lowest error of all; and both MC and QMC having approximately equal performance.

One explanation for these results could be that LHS creates more uniformly spread values in our complex plane than both MC and QMC. We expected QMC, just as LHS, to have a lower relative error than MC in our experiments. QMC should exhibit low discrepancy sequences and thus have a more uniform spread of samples like LHS. One possible explanation for this would be that the Monte Carlo shifting applied on the Sobol points makes the points less uniformly distributed and re-introduces the clustering effect seen in Figure 2a by the MC sampling.

Furthermore, the lower relative error of LHS can be explained as: with less $N_{iterations}$ on our uniformly selected points we are able to filter out more sampled points which are not in the Mandelbrot set compared to the sampled points by the MC and QMC sampling. Another observation of the experiments is that with an increase in $N_{samples}$ we also see a decrease of the relative error. We have shown that as $N_{samples}$ increases the relative error of LHS compared to MC is almost always lower; suggesting that for a desired error threshold, or a certain accuracy in Mandelbrot area calculation, we need less $N_{samples}$ using LHS compared to MC.

Finally, the results of Figure 3 show us that indeed the CLT can be applied to the Monte Carlo estimation of the Mandelbrot area. We can conclude that when take a large amount of relatively under-, middle-, and over-approximated distributions of samples we are still able to achieve a normally distributed sampling distribution of the mean. With the results from Figure 3 we are still able to estimate the (average) area of the Mandelbrot set, due to this normal distribution.

5.1 Discussion

All in all, our experiments have demonstrated the CLT hold for MC simulations. Also that LHS performs allows us to need less $N_{samples}$ to achieve a certain relative error threshold compared to MC; in other words we need less $N_{samples}$ using LHS to estimate the area of the Mandelbrot to a certain accuracy compared to MC.

Finally, we have given an explanation as to why we believe that QMC is about equal in performance compared to MC. Further research into this is recommended as the behavior shown can be dependent on other factors.

References

[Bar+86] Andrew R Barron et al. "Entropy and the central limit theorem." In: *Ann. Prob.* 14.1 (1986), pp. 336–342.

[Fre15] B. Fredriksson. *An introduction to the Mandelbrot set.* 2015.

[Jan05] Wolfgang Jank. "Quasi-Monte Carlo sampling to improve the efficiency of Monte Carlo EM." In: *Computational statistics & data analysis* 48.4 (2005), pp. 685–701.

[Tuf04] Bruno Tuffin. "Randomization of quasi-Monte Carlo methods for error estimation: survey and normal approximation." In: *Monte Carlo Methods and Applications mcma* 10.3-4 (2004), pp. 617–628.