Investigation of the area of the Mandelbrot set

Coen Lenting *(11332484, coen.lenting@student.uva.nl)*
Sjoerd Terpstra *(11251980, sjoerd.terpstra@student.uva.nl)*

17 November 2020

Abstract

The Mandelbrot set is a famous complex fractal. In this research the area of the set was explored using Monte Carlo integration. The random numbers for this method were obtained by using the PCG64 (PCG XSL RR 128/64) pseudo-random number generator. Three different sampling techniques were employed for the sampling of the domain of the set: pure random sampling (PRS) and two stratified sampling techniques called latin hypercube sampling (LHS) and orthogonal sampling (OS). Antithetic variables were applied to PRS as a possible variance reduction method. The total area was estimated to be 1.50744(8) (PRS), 1.50742(6) (LHS), 1.507432(5) (OS) and 1.50750(10) (PRS with antithetic variables), with a 95% confidence interval.

1 Introduction

The Mandelbrot set is a well-known mathematical iterative equation both inside and outside academia. A complex number belongs to this set if the absolute value of the iterative equation does not go to infinity. Due to its intricate nature and infinite fractals it is an interesting object to study. For example one could look at the area that the Mandelbrot set spans in the complex plane.

While many integrals are either analytically solvable or satisfactory solvable by estimation, there are many integrals, including the integral of the Mandelbrot set which are impossible to integrate (to a satisfactory degree) by these methods. To find solutions for these integrals we have to turn to numerical methods. One of the most famous methods is Monte Carlo (MC) integration, which is a stochastic-based method for solving integrals.

In this research we investigate the area of the Mandelbrot set using a numerical MC integration method. We apply the following types of sampling techniques to this problem: pure random sampling (PRS), latin hypercube sampling (LHS) and orthogonal sampling (OS). Furthermore, we formulate and test a method to improve the convergence rate of the Monte Carlo method by introducing variance reduction. This is done by applying antithetic variables to the PRS.

2 Theory & Methods

2.1 Mandelbrot set

The Mandelbrot set, named after Benoit Mandelbrot (Douady et al., 1984), is a fractal constructed by the complex numbers c for which the equation

$$f_c(z_{n+1}) = z_n^2 + c, \quad z_0 = 0, \tag{1}$$

does not diverge and thereby $|z_n| \leq 2$ ($\forall n \in \mathbb{Z}^{\geq 0}$) is always satisfied. By projecting c on a complex plane, with the colour determined by the number of iterations the point c satisfies $|z| \leq 2$, the well-known illustrations of the Mandelbrot set are created. The observed figure is an example of a fractal, which is a self-similar figure, meaning that at different levels of magnification the same or a similar figure is observable. To create an image of the fractal, we determine the points for generating the image by defining a two-dimensional grid, where each grid point represents a complex number c on the plane. Each complex number is evaluated with Equation 1 for a specified number of iterations i. When c satisfies the condition $|z_n| \leq 2$ throughout the entire simulation, the point is declared part of the Mandelbrot set and otherwise the point is given a color representing the number of iterations the grid point did satisfy the conditions.

2.2 Pseudo-random number generator

Since MC integration is based on the usage of random numbers, we employ a pseudo-random number generator (PRNG). We make use of the Random module of the Numpy library in Python, which incorporates the PCG64 (PCG XSL RR 128/64) pseudo-random number generator as defined by (O'Neill, 2014). PCG stands for Permuted Congruential Generator, which is based on a Linear Congruential Generator (LCG). Permutation functions are applied on the output from the LCG to improve the randomness and statistics of the numbers.

LCG's are built on the recursive formula

$$X_{n+1} = (aX_n + c) \mod m, \quad n \geq 0, \tag{2}$$

with $X_n \in [0, m)$ the current random number. $c \in [0, m)$ and $a \in [0, m)$ are constants to be picked, só that the generated numbers are as uniform and random as possible with preferably a large period. The seed is defined as $X_0 \in [0, m)$. To scale the output on the standard domain of $[0, 1)$, the resulting random number X for each iteration equals X_n/m.

2.3 Monte Carlo integration

As previously stated, when analytical integration is not possible or feasible, a solution is to apply numerical methods such as MC integration. This method is based on evaluating the function with random numbers. Suppose it is desired that the integral

$$z = \int_0^1 g(x)dx, \tag{3}$$

is solved, where $g(x)$ is an arbitrary integrand and dx an arbitrary variable to integrate over. By assuming that U is uniformly distributed over the domain [0,1), it is possible to express z as

$$z = E[g(U)]. \tag{4}$$

When dealing with a finite amount of random numbers $U_1, ..., U_k$ this expression becomes

$$z = \sum_{i=1}^{k} \frac{g(U_i)}{k}, \tag{5}$$

which in the limit of $k \to \infty$, according to the law of large numbers, returns Equation 4.

The Mandelbrot set exists outside of the domain [0,1), thus in order to apply this theorem the random numbers were sampled from a larger domain of $x = \{-2, 0.5\}, y = \{-1.1, 1.1\}$, which were empirically selected to encapsulate the entire Mandelbrot set with a certain padding around it. In order to estimate the area of the Mandelbrot set, random numbers for x and y were generated based on different sampling techniques on the specified domain, after which the coordinate was evaluated according to Equation 1. When a generated point satisfied the conditions for the chosen number of iterations, the point was classified as a 'hit', corresponding to a point inside the set, and otherwise as a 'miss'. After generating a certain amount of points, the area can be estimated according to the equation

$$\text{area} = \frac{\text{hits}}{\text{total samples drawn}} \cdot \text{ sampled area}, \tag{6}$$

where the sampled area represents the area of the domain.

2.4 Sampling techniques

To generate a large amount of random numbers on a domain, one can use different sampling techniques. There are two important consideration to take into account when sampling our space. First, we want the numbers be randomly distributed, since this is the basis of the MC approach. Second, we want the points to be uniformly distributed over the domain. The first one can be satisfied by choosing a PRNG with good statistics as described in the earlier section about pseudo-random number generator. The second one can also be obtained by choosing the right PRNG, but also by choosing a sampling technique which improves the distribution of the numbers. This is especially important since it is not possible to throw an infinite amount of random numbers and thus ensures complete coverage of every subset of the domain.

In this research, we applied three different sampling methods. First we applied PRS, which does not enforce any additional constraints on the distribution. Secondly, we applied two stratified sampling techniques: LHS and OS. These two sampling methods have constraints based on the famous n-rooks problem.

2.4.1 Pure random sampling

PRS is a simple approach to randomly sample a space. In each dimension independent random numbers are thrown and there are no constraints on them, except that they have to fall into their given domain. The numbers from each dimension are then combined with the other dimensions in a coordinate on the given space. The expectation is that the number will be uniformly distributed if the amount of numbers thrown go to infinity (given that a good PRNG is used). In simulations it is not possible to use an infinite pool of numbers. In that case, with PRS, one might not expect a completely uniform distribution of the points.

2.4.2 Latin hypercube sampling

The underlying structure of the stratified sampling technique LHS is based on the n-rooks problem. This is a famous problem in chess whereby n number of rooks are placed on the chessboard, not allowed to check any other rook. The same principle applies for LHS, whereby each random number should not intersect any other random number for every dimension.

A Latin hypercube can be created by first dividing each dimension into equal-length intervals and throwing a random number in each interval, creating a list of all the values. We then permute the list for each dimension and the resulting coordinate is the i-th position of each list combined.

This method forces the created points to be more uniformly distributed than with PRS sampling, because it prevents the numbers from being to closely distributed to each other.

2.4.3 Orthogonal sampling

OS is a stratified sampling technique which adds a restriction on the sampling rules that belong to LHS. Whereas LHS produces samples according to the n-rooks problem, the OS subdivides the grid, where all the individual grids also have to satisfy the n-rooks problem (Leary, Bhaskar, and Keane, 2003; Owen, 1992). This extra constraint solves the problem of correlation arising between the samples that can be observed with LHS. By creating the extra subgrids, it will also create a more uniform distribution for a limited amount of points.

A demonstration of the sampling techniques is shown in Figure 1 for 400 points, where the generated points and the projected histograms on the x and y-axis are shown. It can be observed that LHS and OS are uniform over the domain, whereas PRS shows some deviation in the distribution for this number of samples.

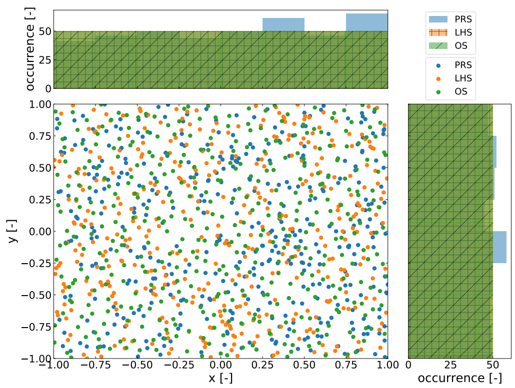

Figure 1: The different sampling techniques demonstrated on the domain $[-1, 1]^2$ with their corresponding histogram projections on the x and y-axis.

2.5 Variance reduction

When using MC integration, it is common to use variance reduction in order to increase the precision of the estimate. Several methods exist, such as the use antithetic variables, control varietes or conditioning. In this research antithetic variables are investigated.

Assume two random numbers X_1 and X_2 are sampled according to PRS. The corresponding variance is

$$\mathrm{Var}\left(\frac{X_1 + X_2}{2}\right) = \frac{1}{4}[\mathrm{Var}(X_1) + \mathrm{Var}(X_2) + 2\mathrm{Cov}(X_1, X_2)], \quad (7)$$

where the variance of the sum is clearly controlled by the individual variances and the covariance between the samples. Whereas the individual variances cannot be directly influenced, it is possible to introduce a sampling technique that generates samples that are correlated. This is the basis of antithetic variables and in the case of a random uniform sample X_1 between [0,1) the complement X_2 can be computed by $1 - X_1$. This results in negative covariance, which results in a decreased total variance. As a positive side effect, half the amount of random numbers need to be generated with the PRNG.

2.6 Experiments

To select the appropriate amount of iterations (i) and number of samples drawn (s), we first examined the convergence rate of the calculation of the area of the Mandelbrot set (A_M) under varying i and s. As the available computational power set a limit on the estimation of A_M the notation $A_{i,s}$ was introduced, which denotes the estimated area of the Mandelbrot set for a specific set of i and s.

2.6.1 Convergence of $A_{i,s}$

The convergence for the estimates area of the Mandelbrot set was examined by fixing s and computing $(A_{j,s} - A_{i,s})$ for $A_{j,s}$ $\forall j < i$, for a maximum number of iterations i. This estimated area was averaged over multiple runs and executed for different values of s to capture the dependency of the convergence on the value of s.

2.6.2 Area of Mandelbrot set

The area of the Mandelbrot set was computed by generating s samples and analyzing their behaviour under equation Equation 1 for a specified number of iterations. The estimated area of the Mandelbrot set and the corresponding confidence interval was computed and compared for each of the samples techniques, PRS, LHS and OS. Under the assumption of the central limit theorem, it was assumed that the distribution of the estimated areas follows a standard normal distribution. This leads to a 95% confidence interval of

$$\bar{A}_{i,s} \pm \frac{1.96S}{\sqrt{n}}, \quad (8)$$

where S is the sample standard deviation, $\bar{A}_{i,s}$ is the mean estimated area and n is the number of simulations.

2.7 Convergence of the confidence interval

The convergence of the confidence interval was investigated by computing the confidence interval for a range of values s for each of the sampling techniques with a fixed i. This test can provide information on how s influences the accuracy of the calculations.

3 Results

In Figure 2 the Mandelbrot set is visible using 256 iterations and a grid size of 1000 by 1000. The colour of each point is determined by the amount of iterations it took for the point to escape the Mandelbrot set i.e. the first n for which $|z_n| > 2$ (Equation 1). The yellow coloured points are part of the set.

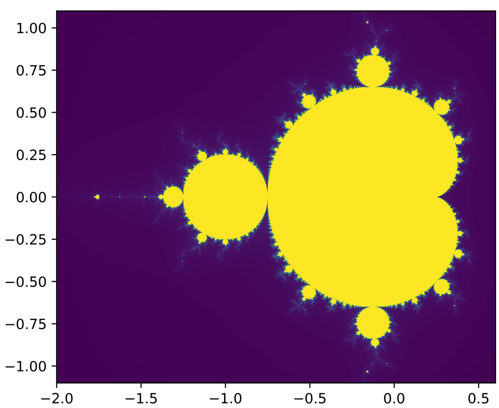

Figure 2: The Mandelbrot set for 256 numbers of iterations with a grid size of 1000 by 1000. The yellow points are part of the Mandelbrot set. The rest of the points are coloured, the colour depending on how many iterations it took for them to escape the set.

The convergence of the area estimation is shown in Figure 3, for $s \in [128^2, 256^2, 512^2]$ and $i_{max} = 256$, indicating the maximum number of iterations, averaged over 10 simulations.

The convergence of the confidence interval as a function of the number of samples generated is shown in Figure 4 for the different sampling techniques with $i = 256$ and s ranges from 64^2 to 512^2. The interval was calculated from the average value of the area over 30 simulations of the integral.

The results from the simulations estimating the Mandelbrot area for the different sampling techniques is shown in Table 1. The simulations were repeated 30 times ($n = 30$) for the parameters $i = 4096$ and $s = 2048^2$. The corresponding 95% confidence intervals

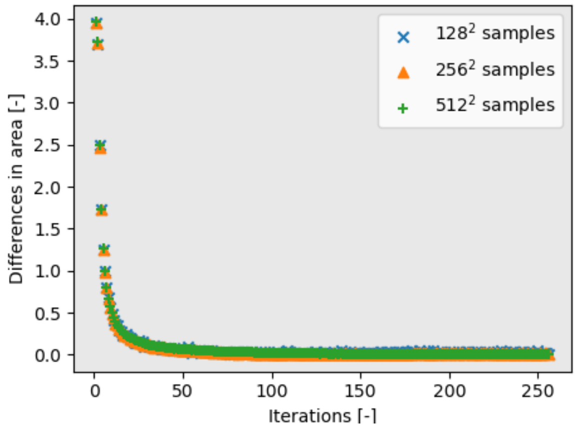

Figure 3: Absolute difference in area of the Mandelbrot set over the number of iterations until $i_{max} = 256$ of equation 1 for $s = [128^2, 256^2$ and $512^2]$ and averaged over 10 simulations.

were calculated with Equation 8. We can see that the confidence interval of OS is roughly a factor ten smaller than the intervals of PRS and LHS. We also observe that the areas of each sampling method lay in each others confidence interval.

Table 1: The sampling techniques and the estimated Mandelbrot area with the corresponding confidence interval from 30 simulations for $i = 4096$ and $s = 2048^2$.

Sampling technique	Estimated A_M ($\pm$ confidence interval)
PRS	1.50744(8)
LHS	1.50742(6)
OS	1.507432(5)

To examine the differences in the resulting areas for the different sampling techniques, we performed a t-test with a 95% confidence interval on the areas obtained by the 30 runs. The results from the t-tests are shown in Table 2. The values represent the p-value from the t-test between the Mandelbrot areas estimated by the two sampling techniques. We can see that there is no significant difference between the three different sampling techniques.

Table 2: The p-values from the performed t-tests on the distributions of the estimated Mandelbrot areas by the different sampling techniques.

	PRS	LHS	OS
PRS		0.941	0.959
LHS	0.941		0.956
OS	0.959	0.956	

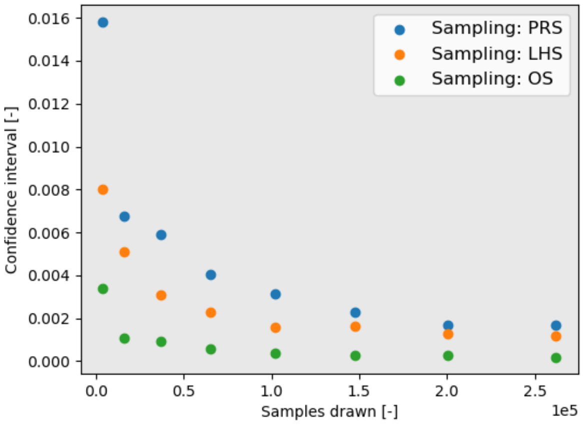

Figure 4: Confidence interval over the number of samples s, ranging from 64^2 to 512^2. The amount of iterations i is set to 256. Each confidence interval is calculated over 30 simulations of the integral.

The simulations estimating the Mandelbrot area with the antithetic variables resulted in an area of 1.50750(10), noticeably producing a slightly less accurate estimate than the estimation done with PRS. The simulation parameters are identical to the parameters for the estimation of the Mandelbrot area for the different sampling techniques ($n = 30, i = 4096$ and $s = 2048^2$).

4 Conclusion/Discussion

In Figure 2 it can be observed that the Mandelbrot set was successfully computed and graphically represented by considering a grid-like structure and computing how many iterations the corresponding coordinate satisfied the constraints.

The convergence of the estimated area of the Mandelbrot set was explored by computing the convergence of the area for multiple fixed values for s and an increasing value for i, shown in Figure 3, where the sampling technique was PRS. Here it can be observed that there is a relatively fast convergence, after which the estimated value of the area steadily decreases for an increasing value of i. This is explained by the limited amount of iterations; some points are treated as they would be part of the Mandelbrot set. However, as the maximum number of iterations is increased, these points may still escape and this will in turn decrease the newly estimated area. This also has implications for the confidence interval, since the expected value of the integral may be lower than its limits. However, because the decreasing of the value of the integral slows down when the amount of iterations and samples is increased, the confidence interval is still a reliable measure of containing the expected value with the given confidence. Another interesting finding in this figure is that the shape of the convergence appears to be independent of the value for s. This means that increasing s will not negatively affect the increase in accuracy caused by increasing the value of i.

The convergence of the confidence interval was successfully investigated by computing the confidence interval for a range of values s for each of the sampling techniques. The first, not surprising, observation is that the confidence interval becomes smaller when the Mandelbrot area is estimated with more samples. The convergence is very steep at first for low amount of samples for each of the sampling techniques, after which the convergence slows down massively. Secondly, it can be observed that the confidence interval for OS is consistently smaller than either PRS or LHS and significantly smaller as well. It is also interesting to note that the difference in the confidence interval between PRS and LHS seems to decrease.

The area of the Mandelbrot set has been successfully estimated for the three sampling techniques. The total area was estimated to be 1.50744 with a 95% confidence interval from 1.50736 to 1.50752 (PRS), 1.50742 with a 95% confidence interval from 1.50736 to 1.50748 (LHS) and 1.507432 with a 95% confidence interval from 1.507427 to 1.607437 (OS). From Table 1 it can be observed that the precision of the estimation increases for the sampling techniques in the order of PRS, LHS and OS. This was expected as the stratified sampling techniques focus on producing better uniformly spread samples than PRS. The difference between PRS and OS is explained by the fact that OS is an advanced technique that makes use of LHS, resulting in even better uniform samples. Analysis of the p-values produced by the t-tests between the estimated areas in Table 2 shows that there is no significant difference between the estimated areas produced by the different sampling techniques.

The attempt to improve the convergence of the Monte Carlo approach was done by making use of antithetic variables. This resulted in an are of 1.50750 with a 95% confidence interval from 1.50740 to 1.60760 (10) (PRS with antithetic variables). The alternative approach unexpectedly performed worse than the normal PRS, whereas a decrease in the confidence interval was expected, was an increase observed. This may be explained by the fact that the number of samples $s = 2048^2$ was sufficiently high, ruling out the benefit of the variance reduction. Hereby not resulting in achieving a higher accuracy.

For future research it might be interesting to look at other variance reduction techniques, for example control varietes. Another suggestion for further research on the

Mandelbrot set is to construct regions in the domain that are proven to belong to the set. This will reduce computational cost as points inside these regions satisfy the conditions linked to the Mandelbrot set and therefore take up a large portion of the computational power, since they have to be evaluated until the maximum selected iteration i.

References

Douady, A., Hubbard, J., Lavaurs, P., Tan, L., & Sentenac, P. (1984). Étude dynamique des polynômes complexes.

Leary, S., Bhaskar, A., & Keane, A. (2003). Optimal orthogonal-array-based latin hypercubes. *Journal of Applied Statistics*, *30*(5), 585. https://search-ebscohost-com.proxy.uba.uva.nl:2443/login.aspx?direct=true&db=buh&AN=9428739&site=ehost-live&scope=site

O'Neill, M. E. (2014). *Pcg: A family of simple fast space-efficient statistically good algorithms for random number generation* (technical report HMC-CS-2014-0905). Harvey Mudd College. Claremont, CA.

Owen, A. B. (1992). Orthogonal arrays for computer experiments, integration and visualization. *Statistica Sinica*, *2*(2), 439–452. http://www.jstor.org/stable/24304869