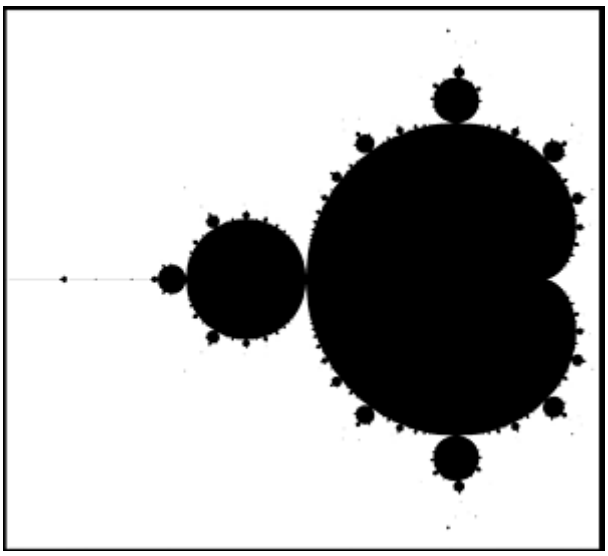
A Statistical Investigation of the Area of the Mandelbrot Set

Background



The Mandelbrot set (M) has been called the most complex object in mathematics, and continues to be the subject of active research. One open question is, what is the area of M? It is well known that the set is bounded by a circle of radius 2, centered at the origin of the complex plane. Thus, the area is certainly less than 4π, or approximately 12.6. Indeed, the area is much less than that. The left-most extent of the set ends with the spike at x = -2, and the right side extends out to approximately x = 0.47. The top and bottom are at approximately y = ± 1.12, respectively. This bounding rectangle has an area of about 5.5, and even this is a gross overestimate, as shown. Here, M is rendered in a binary fashion: points inside the set are colored black and points outside the set are white.

Attempts to determine the area of M have generally involved "pixel counting," wherein M is approximated by a set of points or pixels, and the number of points determined to be inside the set is directly related to the area estimate. Other approaches have been tried, such as finding the area as the result of a series calculation, or finding the area of the periodic components that comprise the interior of the set. Robert Munafo provides a history of attempts at his Mu-ency site.

General Scheme

The Mandelbrot set is defined to be that set of points c such that the iteration $z = z^2 + c$ does not escape to infinity, with z initialized to 0. Consequently, an accurate determination of the area of M would require iterating an infinity of points an infinite number of times each. In practice, a finite set of points is used, and each point is iterated a (finite) maximum number of times each. This leads to an estimate that is clearly flawed due to the relatively small number of points and iterations. However, if a trend in the estimates can be found, then that trend might be extrapolated to infinity, giving a (supposedly) more accurate estimate of the area of M. This was the approach taken in the present work.

Specifically, the number of points and the number of iterations were both related to a "class" parameter. As this parameter increased, so did both the number of points and the maximum number of iterations. In concept, as the class parameter grew to infinity, the estimates of the area should converge to the true area.

For this work, each estimate came from an Ultra Fractal image of M. The square image was centered at approximately (-0.75, 0) with a magnification of 4/3. This gave a real axis range of about -2.25 to 0.75, and an imaginary axis range of about -1.5 to 1.5. M was embedded in a square of area 9. The standard Mandelbrot formula was used, in one-pass linear mode, with periodicity checking turned off. The starting point was (0, 0), power was (2, 0), and the bailout set to 4. Since every pixel was calculated and a simple "inside/outside" determination was all that was needed, no coloring algorithms were used on the inside or outside pixels (transfer function = "none"). No anti-aliasing was used.

The number of pixels on a side of the square was 2^{class}, and this was also the maximum number of iterations. For example, if the class was 8, then each image was 256 pixels on a side, and the maximum iterations was also 256.

For each class, many images were generated. What differentiated the images in the same class was the exact center. It was offset in the real and imaginary directions from the approximate center of (-0.75, 0). The amount of the offset was "randomly" chosen (using a standard random number generator) and was less than the distance between adjacent pixels. Once the image was rendered, it was saved as a Targa file. Then, a separate program interrogated the file and determined the number of pixels inside the set, on the basis of their color. The area was estimated by finding the ratio of inside pixels to total pixels, then multiplying that by the area of the square, 9.

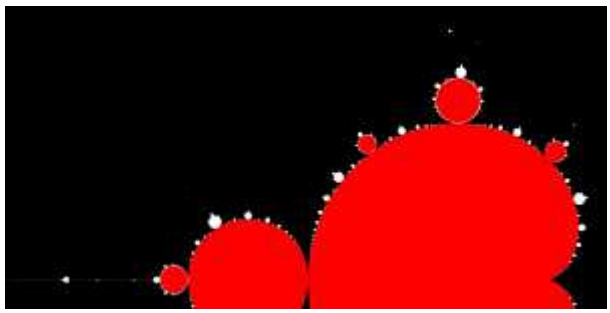
Early Attempts and Subsequent Improvements

Initial attempts used the "Guessing" drawing method. While it was realized that this is less accurate than either of the "linear" methods, it was hypothesized that the inaccuracies would decrease as the class number increased. A sample case was run for a class 9 case, using the same location settings. In this example, 15 pixels were calculated differently. This led to a difference of 0.005 in the area estimate, which was quite significant.

Once it was determined to use the "One-pass Linear" drawing method, several runs were made using single layer images. Points inside the set were colored black and outside points were white, both using the solid color settings. Due to the choice of programming environment, the counting of pixels in the image proceeded much more slowly than the rendering of the images. Consequently, it was decided to employ multi-layered images. Each layer had its own random center point and was essentially an individual datum.

In order to maintain the individuality of each layer for statistical analyses, the pixel coloring was adjusted. First, the scheme was reversed such that outside pixels were colored black, and inside pixels were non-black. Then, the 24-bit nature of Targa files was exploited. Each pixel is represented by 3 bytes, one each for the red, green, and blue components of the pixel's color. A byte can represent 2^8 levels of intensity, or 8 distinct bits of information. One bit was used for each layer and was set to 1 if the pixel was inside the set or 0 otherwise. Each layer's inside solid color was set to a power of 2 for one component only, and 0 for the other two components. The layers were combined using the "Addition" merge mode, insuring that 24 layers could be stored in one image, and each layer's data retrieved intact afterwards.

The final improvement in efficiency came from a version of periodicity checking. (The image on the left shows it being applied to the period 3 disk on top of the main cardioid.) The boundary of the main cardioid is well known, as is the boundary of the large disk centered at -1. Any pixel located in either of these regions is known to be inside M. It can be counted as inside and not calculated, saving quite a bit of time. The other periodic components (disks and small cardioids) have boundaries that are much more difficult to determine exactly. Instead, small circles can be placed inside them such that, if the pixel is inside the circle, and the circle is entirely inside the disk or cardioid, then the pixel is in M and need not be calculated. The circles represent transforms that set the #solid color flag if the pixel was inside. While the solid color is shown red here, it was set to the same color as the inside solid color on each layer.



There was overhead involved with checking each pixel for membership in the periodicity circles, which added to the overall calculation time. For each class, there was a maximum number of circles that could be used before the overhead outweighed the savings in not iterating every pixel. The class 8 runs had only the main cardioid and period 2 disk at -1 checked. For class 13 and 14 runs, the main cardioid and 8 disks were checked, as shown. While

the red and white points are in the Mandelbrot set, only the white points were actually calculated.

Results

The area for each class was treated as a quantity to be estimated by statistical sampling. Each layer's estimate was taken as a single observation of a random variable. By acquiring many samples, the population mean (the true area for that class) can be estimated. For example, for class 8, there is a well-defined figure that results from 256 iterations. It approximates, but is slightly larger than M. The population mean (μ) of all the class 8 measurements is that true area, which is being estimated. The standard deviation of the measurements was used to determine the 95% error tolerance (E). This is the half-width of the confidence interval: the probability is 95% that μ lies in the interval from the mean minus E to the mean + E.

Listed in the below table are:

- class: overall parameter
- grid side: how many pixels were on one side of the grid, 2^{class}
- pixels: total number of pixels, $(\text{grid side})^2$
- iterations: maximum number of iterations per pixel, 2^{class}
- count: how many measurements were made for this class
- mean: arithmetic mean of the observations
- standard deviation: sample standard deviation of the observations
- E: 95% error tolerance as discussed above

class	8	9	10	11	12	13	14
grid side	256	512	1024	2048	4096	8192	16384
pixels	65536	262144	1048576	4194304	16777216	67108864	268435456
iterations	256	512	1024	2048	4096	8192	16384
count	92513	14900	1626	264	32	8	4
mean	1.521814	1.514038	1.510228	1.508360	1.507446	1.507011	1.506793
standard deviation	1.55 E-03	6.23 E-04	2.05 E-04	8.22 E-05	2.78 E-05	1.07 E-05	8.88 E-06
E	1.00 E-05	1.00 E-05	1.00 E-05	9.92 E-06	9.63 E-06	7.38 E-06	8.70 E-06

Analysis

The means decreased with increasing class size, as was expected. Any estimate of the area of M based on finite number of pixels and iterations should be an over-estimate; points thought to be in the set can escape with additional calculation, but escaped points will never return. It was also clear that the rate of decrease decreased with class, so the estimates appeared to be reaching an asymptote. To investigate this further, it was hypothesized that the mean decayed exponentially with class, with a base of 2:

$$\text{mean} = A + B\ 2^{-\text{class}}$$

Where A and B are constants to be determined by fitting the data. In this form, the relationship is linear if the independent variable is taken to be 2^{-class}, instead of the class itself. Then, simple linear regression tools can be used to determine the best fit A and B. A is the value of the mean when 2^{-class} is zero, or the class equals infinity. Thus, A is an estimate to the true area of M. B is the rate at which the area estimates

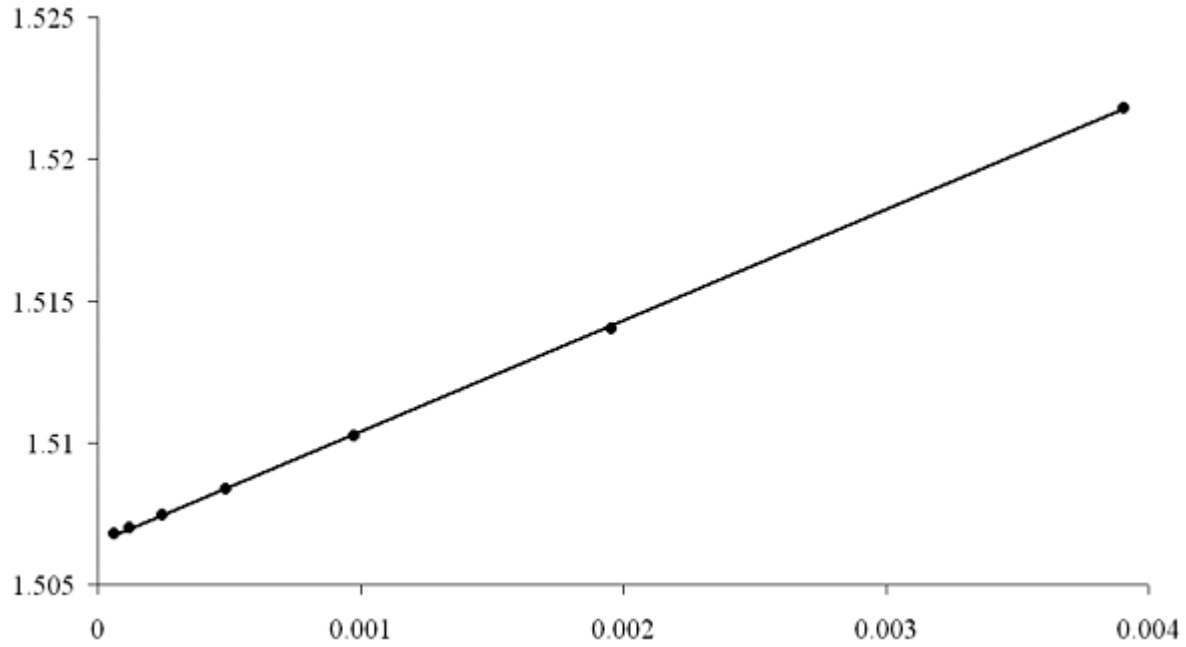
increase with decreasing class size. This is related to how the numbers of pixels and iterations vary with the class.

As the plot below shows, the data are quite well modeled by this formula. The mean is plotted vertically against 2^{-class} on the horizontal axis. The least-squares analysis returned:

$$A = 1.506484193$$
$$B = 3.909157877$$
$$R^2 = 0.9998653$$

Since the measurements used to determine A were statistical in nature, it's reasonable to ask how good this estimate of A is. To do this, the standard error was used. This was the standard deviation divided by the square root of the count. In general, it reflects how the variation in the sample mean decreases with increasing sample size.

For each class, the mean and standard error provided the population parameters for a set of random numbers. One thousand random numbers were generated for each class. Then, a set of 7 points was created by taking one random number from each class. A line was fitted to these data and the intercept became a new random variable. From this analysis, the mean of the intercepts agreed with the A value given above to 0.000003%, and the standard deviation was approximately $2.2\ 10^{-6}$. As formulated, this really represents a standard error, so the 95% error tolerance is $4.35\ 10^{-6}$. Thus, it is concluded that the area of the Mandelbrot set is approximately 1.506484, with a 95% confidence interval from 1.506480 to 1.506488.



Back to the articles page

Up to my home page