Mathematical Simulations through Monte Carlo Methods

Fractal Trees and the Approximation of Pi

Akshansh Bhanjana - Aryan Gupta

Algorithms involving Monte Carlo Methods are comparatively scalable. They see a lot of usage in physical systems. This method can reduce complex models to simpler events, which can be simulated with ease. This paper involves simulations of two such mathematical phenomena: Estimating the Value of Pi, and Fractal Trees.

Introduction

In probability, we find various situations where an analytical solution cannot be calculated directly. This calculation may be difficult due to various reasons such as many random variables, disturbances (noise) in the observations, etc.

In such situations, it's possible to find probability distributions involving random variables using simulations.

Monte Carlo Methods / Experiments are algorithmic simulations which depend on repeating random sampling to get analytical results.[1] They allow significance in the risks of quantitative analysis. This concept uses randomness to find probability distributions.

This technique is quite effective; case in point: it was used to make the atom bomb.

The three basic use cases of Monte Carlo are:

- **estimating** Probability Distribution of a function

- **approximating** mean, variance, or other such quantities

- **optimizing** functions, i.e. maximizing / minimizing functions
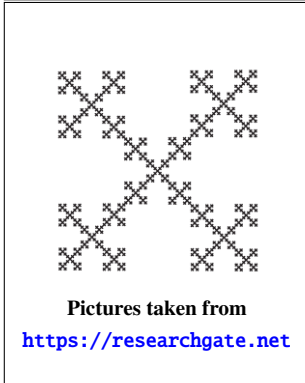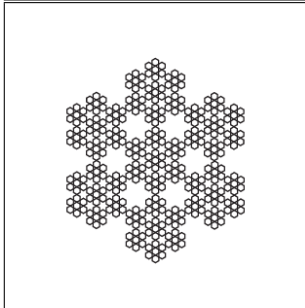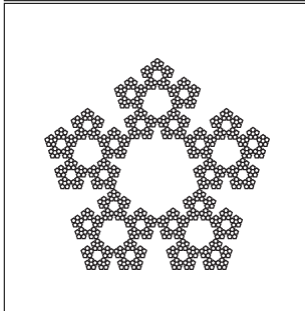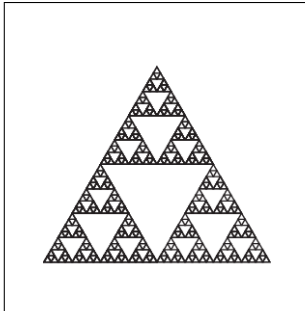
Akshansh Bhanjana is a sophomore at Cluster Innovation Centre, University of Delhi, currently majoring in Information Technology and Mathematical Innovations.

Aryan Gupta is a student at Cluster Innovation Centre, University of Delhi, pursuing majors in Information Technology & Mathematical Innovations.

Fractals

A fractal is 'a rough or fragmented geometric shape that can be subdivided in parts, each of which is (at least approximately) a reduced/size copy of the whole'. The term was coined by *Benoît Mandelbrot* in 1975 and was derived from the Latin fractus meaning broken or fractured.

Fractals are found in many areas of nature such as trees, ferns, flowers, and even river deltas. Growth Spirals follow the Fibonnaci sequence while forming fractals. A good example is that of Romanesco Brocolli! The Human Respiratory System with its level of bronchi, bronchioles, and alveoli are proof of the omnipresence of fractals.

Two main characteristics of Fractals are:

- **Self Similarity**: Every subdivision of a fractal has the same shape as the whole, i.e. self similar, which means that no matter how much one magnifies a fractal, the result would be the same.

- **Non-Integer Dimensions**: Many natural phenomena don't revolve around whole numbers and classical geometries like squares, circles, 3D cubes and spheres, e.g. *Golden Ratio*, value of Pi, etc. Similarly, Fractals don't have a dimension of a whole number, but a number between one and two dimensions. Many natural phenomena are better described using a dimension between two whole numbers.

  So, while a straight line has a dimension of one, a fractal curve will have a dimension between one and two, depending on how much space it takes up as it twists and curves. The more the flat fractal fills a plane, the closer it approaches two dimensions.

Fractal geometry is expanding our ability to invent better devices which resonate with the surrounding nature.

The role of fractals has been crucial in development of the following areas:
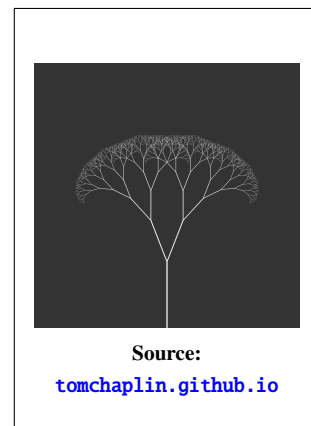
- Generation of new music, art forms

Pictures taken from
https://researchgate.net

- Signal and Image Compression

- Seismology

- Computer Graphic designing (e.g. hills can be made through recursive algorithms of a triangle)

Types of fractals

Here are some examples of Fractal patterns in nature:

1. **Trees**

2. **River Deltas**

3. **Growth Spirals**

4. **Flowers**

Source:
tomchaplin.github.io

Mathematical Foundation

The Law of Large Numbers states that with the increase in the number of random trials, the estimated quantity becomes more accurate.[2]

The Monte Carlo Method approximates a property of a huge distribution, by averaging that property for N of these chosen at random i.e. through drawing samples. Drawing a sample may involve the calculation of the probability of a random event or a computational simulation i.e. Monte Carlo Simulation.[4]

Monte Carlo Methods can be expressed in *mathematical terms* as follows:

Consider a multi-dimensional random variable, **X**, with its Probability Density Function (PDF) as **f$_X$(x)**. Then, the expected value of the function **g(X)**[3] is:

$$E(g(X)) = \sum_{x \varepsilon X} g(x) f_X(x); \; if \; X \; is \; discrete$$

$$E(g(X)) = \int_{x \varepsilon X} g(x) f_X(x); \; if \; X \; is \; continuous$$

Taking an N-sample of X's, and computing the mean of **g(x)** over the sample, the Monte Carlo Approximation equals:

$$\overline{g}_n(x) = \frac{1}{n} \sum_{i=1}^{n} g(x_i)$$

Algorithmic/Computer Implementation

Let's consider an example of **trees**.

1. Fractal Trees

Conventional Statistics involve characterizing surfaces. This multiple scalar nature can be depicted through fractal geometry.[5]

To have an even clearer view of the above, we have prepared a simulation. Attached are some screenshots of the same.

In the given simulation, notice the sliders at the bottom. These represent the angle between branches, the depth of the tree, and the zoom level respectively. Even a slight change in the angle or the depth can produce an entirely different result.

Basically, the simulation of a Fractal Tree is a recursive function. It has 2 parameters:
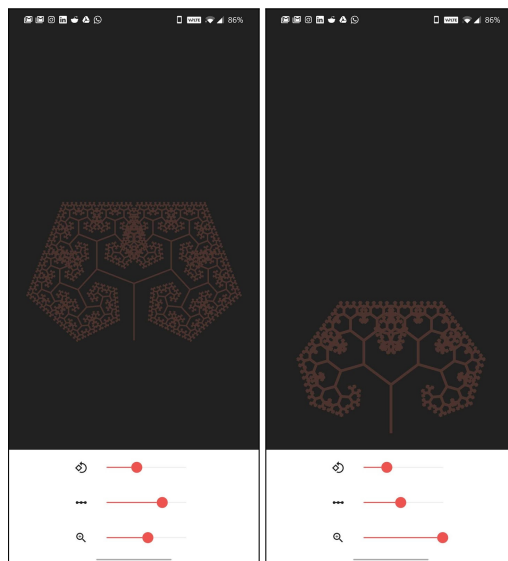
- **angle**

- **length**

The angle determines the angle of inclination between the current branch and its parent branch. The depth determines the depth of the tree (number of levels).
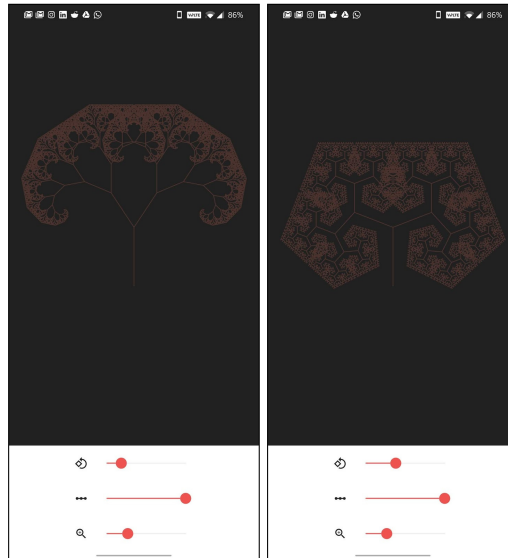
The length of the initial branch is directly proportional to the depth of the tree. After each level, the length of the branch reduces by 30%. Thus, the bigger the depth, the denser the tree.

Here are some more screenshots:

Now, let's try making the tree as dense as possible.

The tree is a lot denser, and a lot taller now. Changing the angle makes the density difference look even more profound. For reference, the video is hosted here.

2. Estimation of Pi

The plan is to simulate random **(x, y)** points on a 2-D plane with the domain as a square of side **2R** units. Consider a circle inside the same domain with the same diameter and inscribed into the square. Basically,

$$\frac{area\ of\ the\ circle}{area\ of\ the\ square} = \frac{number\ of\ points\ inside\ the\ circle}{total\ number\ of\ points\ generated} = \frac{\Pi r^2}{4r2} = \frac{\Pi}{4} = k$$

that is,

$$\Pi = 4 * k$$

The points are generated in the **Dart** language using the *dart:math* library, i.e.

$$x \;=\; rand(-R,\; R)$$

$$y \;=\; rand(-R,\; R)$$

*where* **rand(a, b)** *generates random points between a and b*

This is assuming that the centres of both, the square and the circle, lie at the **origin**.

Now that random points are generated, we start plotting them. If the generated point lies within the circle, we mark it with *green*, else with *red*. The value given by **4 * k** then evaluates to:

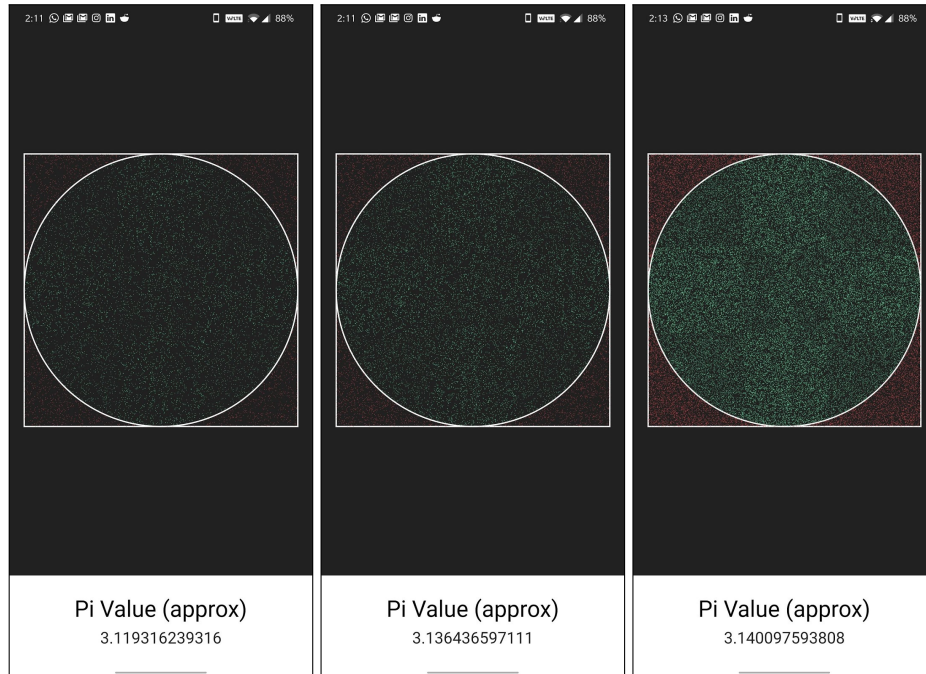$$\Pi \;=\; 4 * k \;=\; 4 * \frac{number\ of\ green\ dots}{total\ number\ of\ dots\ (green\ +\ red)}$$

The best thing is that we don't need to consider graphics for simulation. We only need to output random **(x, y)** pairs and then do the following:

- if the point lies inside the circle - increment number of points inside the circle

- increment the total number of points irrespective of where the point lies

- calculate the above result, i.e. **4 * k**
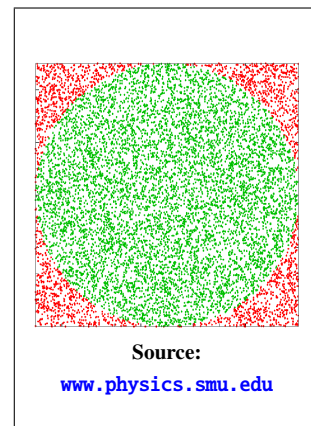
To have a clearer view of the above, we have prepared a simulation for the same. Here are some screenshots of the same:

Source:
www.physics.smu.edu

Pi Value (approx)	Pi Value (approx)	Pi Value (approx)
3.119316239316	3.136436597111	3.140097593808

In the above, it is clearly evident that as the number of points increases, the approximate value of Pi approaches the real value. This is in complete agreement with the hypotheses above, and hence, proves that Monte Carlo simulations are good approximations when one can't calculate the exact value.

https://bit.ly/pi_
approximation

The complete simulation video is available here.

3. Conclusion

In this paper, Monte Carlo simulations have been used to show two test cases:

- simulating Fractal Trees, and

- estimating the value of Pi

This method has been shown to be valid and effective, through testing at various parameters.

Suggested Reading

[1] "A Gentle Introduction to Monte Carlo Sampling for Probability"
https://machinelearningmastery.com/monte-carlo-sampling-for-probability/
[Accessed: 7-April-2020]

[2] "Monte Carlo Simulations"
https://www.palisade.com/risk/monte_carlo_simulation.asp
[Accessed: 10-April-2020]

[3] "Mathematical Foundations of Monte Carlo Methods"
https://www.scratchapixel.com/lessons/mathematics-physics-for-computer-graphics/
monte-carlo-methods-mathematical-foundations/pdf-and-cdf
[Accessed: 12-April-2020]

[4] "Monte Carlo Methods and Importance Sampling"
http://ib.berkeley.edu/labs/slatkin/eriq/classes/guest_lect/mc_
lecture_notes.pdf
[Accessed: 2-May-2020]

[5] Zou, Mingqing, et al. "A Monte Carlo method for simulating fractal surfaces."
Physica A: Statistical Mechanics and its Applications **386.1 (2007): 176-186.**

[6] Lin, Yi-Cheng, and Kamal Sarabandi. "A Monte Carlo coherent scattering
model for forest canopies using fractal-generated trees." *IEEE Transactions
on Geoscience and Remote Sensing* **37.1 (1999): 440-451.**

[7] "Classification of Fractals"
https://www.cs.mcgill.ca/~rwest/wikispeedia/wpcd/wp/f/Fractal.
htm
[Accessed: 28-May-2020]

[8] "General Introduction to Fractal Geometry"
http://www.fractal.org/Bewustzijns-Besturings-Model/
Fractals-Useful-Beauty.htm
[Accessed: 31-May-2020]

[9] "Source of Fractals"
http://www.fractal.org/Bewustzijns-Besturings-Model/
Source-of-Fractals.htm
[Accessed: 6-June-2020]

[10] Pippa, Natassa, et al. "On the Ubiquitous Presence of Fractals and Fractal
Concepts in Pharmaceutical Sciences: A Review." *International Journal of
Pharmaceutics*, Elsevier, 8 Sept. 2013,
www.sciencedirect.com/science/article/pii/S0378517313008211