

Random and Brownian Motion and Fractal Analyses

PMATH 370 Final Project

Michael Sutton

Abstract

Brownian motion describes the apparently random motion of particles suspended in a fluid, named for Scottish botanist Robert Brown, who first observed through a microscope pollen particles moving erratically on the surface of otherwise still water. On an atomic level, such motion is due to collisions with the molecules comprising the suspending fluid, but due to the frequency and unpredictable nature of these collisions, on a macroscopic level we observe essentially random motion. There are many useful applications in the study of such motion, from stock market models to foraging animal paths, and entire fields of study in particle science; the mathematics behind Brownian motion apply in analysis of nearly all things apparently random.

In this paper I will scratch the surface on a few of the important and interesting results that have arisen from the study of random motion, with a special eye toward fractal conclusions. The field of mathematics spawned from the analysis of random motion is staggering, and even the most basic facts need dozens of calculus and statistical theorems to sort out much of the randomness that is occurring in the background. As such most results will simply be stated and cited for brevity, while their significance and relation to fractal geometry is discussed.

Random Walks

Conceptually simpler than Brownian motion, Random Walks are a good place to start when examining more complex random motion. Also called a “Drunkard’s Walk,” Random Walks are a series of discrete steps taken on a d-Dimensional lattice. There are interesting probabilistic conclusions that can be drawn dependent on number of paths and the dimension the motion takes place in.

Below are a few images of Random Walks, the first with a wider lattice and lower step count allows us to easily visualize the process, whereas the latter two show the shapes the motion takes in higher dimensions or with larger step counts.

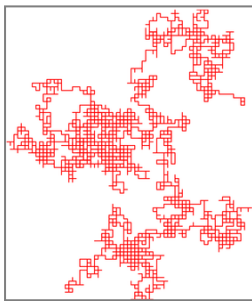


Figure 1A-Random Walk on 2D lattice

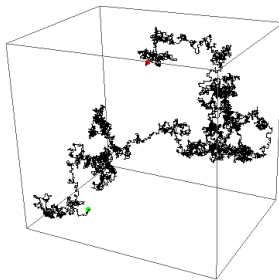


Figure 1B-Random Walk on 3D lattice

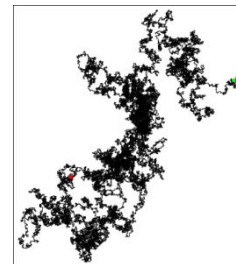


Figure 1C-Random Walk with large step count

As the step size approaches zero and the number of steps increases to infinity, random walks begin to approximate Brownian motion. We see in Figures 1B and 1C that we begin to lose the lattice

and the path blends to something more chaotic motion as step size decreases, reminiscent of the Brownian motion we will later observe.

On a one dimensional lattice, the distance from the origin takes a Gaussian distribution when considered as a random variable, so this gives us the idea that in lower dimensions Random Walks will tend toward their origin, which explains why as the step count increases we will almost always observe fully saturated portions of the lattice – the dark areas in Figures 2 and 3 above. This idea is important in later sections – when we observe at the Hausdorff Dimension of Brownian motion, consider the approximation as a random walk on an infinitely dense lattice. On the tendencies of random walks to their origin, the following section will discuss the probabilities of random walks returning to their origination point and related mathematical constants.

Pólya's Random Walk Constants

Hungarian Mathematician George Pólya derived the probabilistic constants $p(d)$ defined as the probability an infinite random walk on a d-Dimensional lattice will return to the origin.

In 1921, Pólya demonstrated that $p(1) = p(2) = 1$, and furthermore that $p(d) < 1$ for $d > 2$. It's an intriguing result that an infinite random walk in 1 or 2 dimensions will always return to the origin.

It was not until 1939 that G.N. Watson showed

$$p(3) = 1 - \frac{1}{u(3)} = 0.340537 \dots$$

Where $u(3)$ is the third Watson Triple Integral, modulo a multiplicative constant:

$$u(3) = \frac{3}{(2\pi)^3} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \frac{dx dy dz}{3 - \cos x - \cos y - \cos z} = 1.516386 \dots$$

A table of Pólya constants for dimensions 3 through 8 can be seen below:

d	$p(d)$
3	0.340537
4	0.193206
5	0.135178
6	0.104715
7	0.0858449
8	0.0729126

The 2-Dimensional random walk is often interpreted as a drunken man wandering a city, the namesake of the Drunkard's Walk. At each intersection, the man will pick a random direction to travel (including the possibility of doubling back). Pólya's constants above tell us that should the man wander

for long enough, he will eventually end up back where he started. Piloting a helicopter, though, said drunken man may become lost – not to mention the consequences of the z-axis value dropping below 0.

Brownian Motion

It was not until 1905 that Albert Einstein created the physics theories behind Brownian motion, describing the buffeting of a suspended particle in stationary liquid by the surrounding molecules. Since these particles undergo somewhere around 10^{21} collisions per second, the calculations involved the density of Brownian particles, diffusion constant D , and moment of inertia at time t in the Brownian motion. His work describing such motion laid the groundwork for much of molecular-kinetic theory. Allegedly Einstein wasn't even aware of Brown's work until nearing the 1905 publication of *On the movement of small particles suspended in a stationary fluid as demanded by the laws of kinetic theory*.

Einstein's formula for mean square distance as a function of time t was:

$$\lambda_x = \sqrt{\frac{tRT}{N} \frac{1}{3\pi kP}}$$

Where t is time, R is gas constant, T is temperature, N is Avogadro's Number, k is viscosity, and P is particle radius.

Though the physics behind Brownian motion raises some interesting problems, we will be sticking to a mathematical definition of the random motion, which in some ways is even more chaotic than that described by Einstein. Einstein's calculations involved inertia and momentum, meaning that aside from at initialization, the current state of the particle affects its future trajectory. For our purposes, though, we will be using a stochastic process known as the Wiener Process to define Brownian motion as a function of t :

For an interval $I = [0, \alpha]$ or $[0, \infty)$, we define $\{B_t\}_{t \in I}$ to be Standard Brownian motion if it is a Gaussian process satisfying the following conditions:

- i. $B_0 = 0$
- ii. $\forall t, s \in I$ where $t < s$, $B_t - B_s$ and B_s are independent
- iii. $\forall t \in I$, $B_t \sim N(0, t)$
- iv. $t \mapsto B_t$ is continuous on I .

The process was developed by American Mathematician Norbert Wiener in 1923, after he had taken a keen interest in Brownian motion as described in Einstein's works. The Wiener Process quickly became synonymous with Brownian motion, and is used as the primary model for the motion.

Lévy proved in 1948 that Standard Brownian motion could be defined on $I = [0, \infty)$ - previously a conjecture by Wiener in 1923, giving us the chance to consider infinite Brownian paths satisfying the above properties.

Brownian Motion Basic Properties

Brownian motion generated by the Wiener process is nowhere differentiable and has no intervals of monotonicity.

Consider $\{B_t\}_{t \geq 0}$, Brownian motion from $t = 0$. Each of the following is also Brownian motion:

$$\forall a > 0 \left\{ \frac{1}{a} B_{a^2 t} \right\}_{t \geq 0} \quad \text{and} \quad \left\{ \frac{1}{a^2} B_{at} \right\}_{t \geq 0} \quad (\text{scaling})$$

$$\forall s > 0 \left\{ B_{s+t} - B_s \right\}_{t \geq 0} \quad (\text{shifting})$$

$$\text{For } X_0 = 0, X_t = t B_{\frac{1}{t}}, \left\{ X_t \right\}_{t \geq 0} \quad (\text{inversion})$$

The scaling property, specifically, describes a fractal nature of the Wiener process, no matter how much we stretch or compress the process, the result is still Brownian motion. This, combined with the fact that there are no intervals of monotonicity, gives us the infinitely detailed and self-copying properties that define many fractal shapes.

Additionally, relating to how fast Brownian motion will travel from the origin, we have the following limits:

$$\lim_{t \rightarrow \infty} \frac{B_t}{t} = 0 \quad \limsup_{t \rightarrow \infty} \frac{B_t}{\sqrt{t}} = \infty \quad \liminf_{t \rightarrow \infty} \frac{B_t}{\sqrt{t}} = -\infty$$

And the expected value $E(B_t) = 0$

This tells us a bit about the rate at which the boundary of Brownian motion grows, and we'll revisit said boundaries later.

Brownian Paths and Collisions

When considering a Brownian path – the trace left from a single walker performing Brownian motion – we can use a fractal Cantor-like analysis to show how multiple Brownian paths will collide or avoid each other in different dimensions.

In \mathbb{R}^2 , any number of Brownian paths intersect with positive probability. In \mathbb{R}^3 , two paths intersect with positive probability, but no more than two. In higher dimensions, no two paths intersect with positive probability. These mathematical theorems come from Dvoretzky, Erdős, Kakutani and Taylor in the 1950's, and the construction that leads to these conclusions is a random Cantor-like division of $[0,1]^k$.

For example, take $k = 3$, and the construction is as follows:

Divide $[0,1]^3$ into $8 \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2}$ cubes, and retain each cube with probability $\frac{1}{2}$. Repeat this process with each cube kept in the first iteration. Since the expected number of cubes to keep in a single iteration from a single cube is 4, we don't expect any iteration to be empty.

Define $Q\left(3, \frac{1}{2}\right)$ to be the intersection of these iterations. If A is a closed subset of $[0,1]^3$ and $\{B_t\}_{t \in [0, \infty)}$ Brownian motion, where $B_0 \in [0,1]^3$, chosen uniformly, then it can be shown that for some positive constants C_1 and C_2 , that:

$$C_1 P\left(Q\left(3, \frac{1}{2}\right) \cap A \neq \emptyset\right) \leq P(\exists t \in [0, \infty) \text{ s.t. } B_t \in A) \leq C_2 P\left(Q\left(3, \frac{1}{2}\right) \cap A \neq \emptyset\right)$$

Where P is the usual probability function. We can then reduce the problem of finding an intersection between two Brownian paths to finding the intersection between two $Q\left(3, \frac{1}{2}\right)$ sets, which will be of the form $Q\left(3, \frac{1}{4}\right)$ - we keep any of the 8 child cubes with probability $\frac{1}{4}$. Since the expected number of child cubes in an iteration is 2, we observe that $P\left(Q\left(3, \frac{1}{4}\right) \neq \emptyset\right) > 0$, but for 3 paths, when we're looking at $Q\left(3, \frac{1}{8}\right)$, the expected number of children from any single cube is 1, and we experience a "die-off" of the branches, wherein we observe $Q\left(3, \frac{1}{8}\right) = \emptyset$ with probability 1, and the result for 3 or more Brownian paths intersecting in \mathbb{R}^3 follows.

It is a curious result to consider that Brownian paths in \mathbb{R}^4 and higher dimensions will never intersect. Conceptually, one might envision two random paths crossing in any dimension, but the fact remains that the probability of such an event in higher dimensions is zero. This is one of the quirks of probabilistic results with non-discrete distributions. Though we can point a particular occurrence, the probability of that event may well be zero. You can consider picking a number at random on the interval $[0,1]$, but for any given $x \in [0,1]$ the probability that this particular number was picked is 0, since there are infinite equally probable results.

Brownian Paths Hausdorff Dimension

Consider Brownian motion $\{B_t\}_{t \in I}$ on the plane in \mathbb{R}^2 . The outer boundary of this motion is called the "frontier" of the Brownian motion.

A Brownian Path itself has a fractal dimension of 2, meaning there will always be space-filling for some open set in \mathbb{R}^2 , which of itself is a curious result related to the nowhere differentiable, yet continuous properties of Brownian motion; a Brownian path crosses itself infinitely often in \mathbb{R}^2 and \mathbb{R}^3 , and in no dimension does the motion have any interval of monotonicity. This means macroscopically, though we see a continuous path we can follow visually, as we consider smaller and smaller intervals in the motion we will still find infinite self-intersection, creating a 2-dimensional shape rather than 1-

dimensional path. Reflecting back on the infinitely dense lattice Random Walk approximation, we see that a fully-saturated infinitely dense lattice is space-filling.

It was first conjectured by Mandelbrot in 1982 that the fractal dimension of this frontier for $\{B_t\}_{t \in [0,1]}$, hereafter referred to as $B[0,1]$, is $\frac{4}{3}$. This was later proved to be true by Lawler, Schramm, and Werner in 2000.

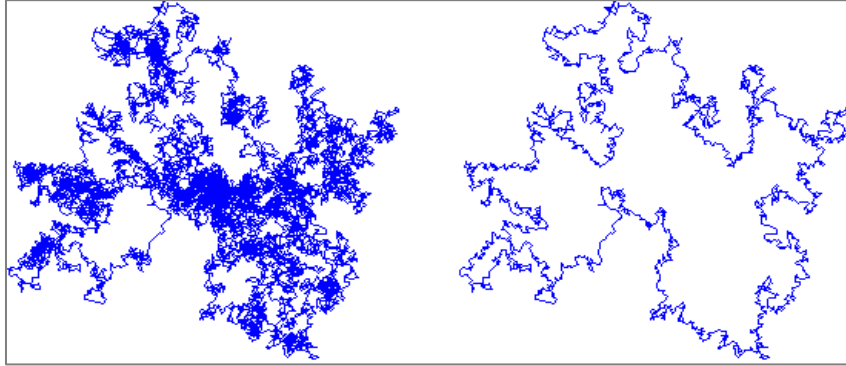


Figure 2 - A Brownian path and its boundary

Other relevant fractal dimensions when considering a single Brownian path are those of cut points and pioneer points. Cut points are points where B_s where $B[0,s] \cap B[s,t] = \emptyset$, and they have Hausdorff dimension $\frac{3}{4}$ for $B[0,1]$. A pioneer point is a point B_s that lies on the frontier at time s , and the Hausdorff dimension of the set of pioneer points for $B[0,1]$ is $\frac{7}{4}$.

Of special note is that these dimensions are deterministic. Out of motion whose basis is almost entirely random, we get totally deterministic dimensions of these three sets of points.

Brownian Trees

Brownian trees, or Diffusion Limited Aggregates (DLA), are formed by particles following Brownian motion called “walkers” and a single or group of seed particles. When a walker contacts the seed structure, it incorporates itself and the seed structure grows. This is a natural phenomenon called “diffusion-limited aggregation” wherein accumulation in systems where diffusion is the form of transportation (fluid suspension) aggregation of particles form these Brownian Tree structures – observable in many crystallization and precipitation processes.

Unlike Brownian motion, not much about Brownian trees is deterministic, so far as has been concluded. The average Hausdorff dimension of a Brownian tree in \mathbb{R}^2 is 1.7, a measured average.

Pending other limiting or guiding structures in the environment, the process leads to some astounding fractal-like shapes.

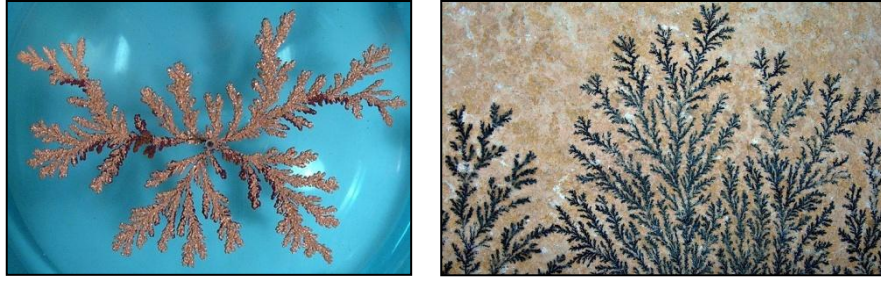


Figure 3 - Copper Aggregate from Solution (left), Manganese Dendrites (right)

Among the many other fractals observed in the natural world, Brownian Trees stand out as being the result of a process with a high degree of randomization. From crystals to clouds and oil spills, the study of DFA gives us insight to the shapes these objects will take, and why they take on these peculiar shapes.

Take, for example, a cloud. At first the formulation of a cloud might seem far-removed from processes like crystallization, but when we consider the mechanism behind them, we draw many parallels: Water particles (walkers) precipitating from the air (suspending fluid) and accumulating into clouds (seed structures). The primary difference between this and solid precipitation processes being that the seed structure is not rigid.

Brownian trees are surprisingly simple to simulate programmatically, keeping in mind that the base process only requires a simulation of Brownian motion that terminates upon contacting the seed structure. Brownian motion is often approximated programmatically by the approximation mentioned near the beginning of this paper – a random walk with a miniscule step size and a large number of steps, utilizing pseudorandom number generation to dictate the direction taken at each step. Below are two Brownian trees generated by a brief bit of Java code that can be found in the appendix.

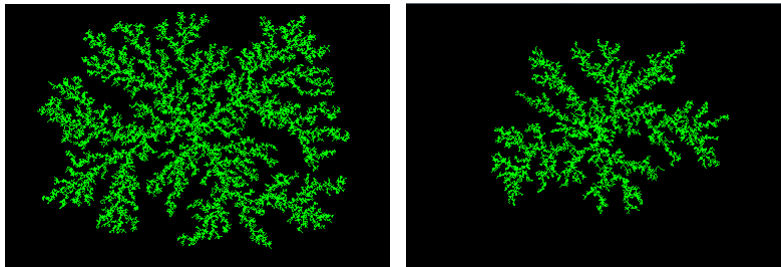


Figure 4 - Brownian Tree Simulations – 30000 walkers (left) and 20000 walkers (right)

The code can be tweaked in ways that let us observe different properties of Brownian trees. For example, if we colour code based on when a walker enters the tree.

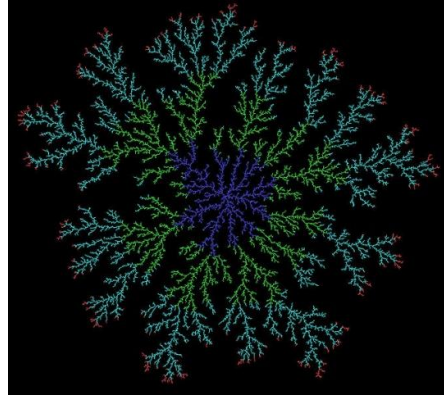


Figure 5 - Brownian tree colour-coded simulation

We may also add boundaries limiting the possible motion of the walkers:

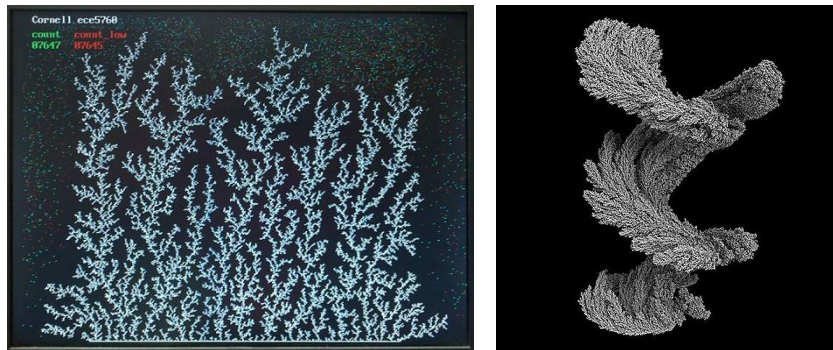


Figure 6 - Brownian trees with lower boundary (left) and helical boundary (right)

Again, these simulations give us some insight into the natural processes they model, as different constraints can arise from different environmental factors, such as crystal growth constrained to a fissure in the surrounding rock, and we can immediately see a striking resemblance between even the simple simulations and the natural processes they model.

Conclusion

From sticky stochastic processes simulating random motion to beautiful crystalline shapes growing in solution, Brownian motion shapes our world on many levels. The very building blocks of our universe are chaotic and unpredictable by nature, as a certain Scottish botanist began to grasp watching pollen dance on the surface of placid water. Sitting on that surface, every immeasurable fraction of a sliver of a second billions of collisions take place – more data than any computational engine can hope to process. Since the famous observations of Robert Brown, many giants of Mathematics and Physics have brought some order to the chaos, giving us constants and theorems and constructions that help us analyze random motion, and their conclusions have made ripples in many important fields that deal with

unpredictability. It still remains though, no matter how many numbers we've found to describe the path it will follow, we will never know the direction that speck of pollen will dart next.

References

Kozdon, Michael (1998). *An Introduction to Random Walks from Pólya to Self-Avoidance*.
http://stat.math.uregina.ca/~kozdrn/Research/Talks/duke_polya.pdf

Renn, Jürgen (2005). *Einstein's Invention of Brownian Motion*.
<http://www.physik.uni-augsburg.de/theo1/hanggi/History/Renn.pdf>

Peres, Yuval (2001). *An Invitation to Sample Paths of Brownian Motion*.
<http://stat-www.berkeley.edu/~peres/bmall.pdf>

Lawler, Schramm & Werner (2000). *The Dimension of the Planar Brownian Frontier is 4/3*.
<http://arxiv.org/pdf/math/0010165v2.pdf>

Benhamou, Eric (date unknown). *Wiener Process*.
<http://www.ericbenhamou.net/documents/Encyclo/Wiener%20process.pdf>

Images

Figure 1 - <http://mathgis.blogspot.ca/2009/08/visualize-irrational-number-as-random.html>

Figure 2 - http://video.ias.edu/webfm_send/705

Figure 3 - <http://www.mindat.org/min-26645.html>
http://en.wikipedia.org/wiki/Diffusion-limited_aggregation

Figure 4 - Generated by code in appendix

Figure 5 - http://en.wikipedia.org/wiki/Diffusion-limited_aggregation

Figure 6 - <http://fffffound.com/image/c8f584511e5e51d905ca9a0537ec9d936bc5e7cc>

Appendix

Code used to generate DLA trees in Figure 5.

Source: http://rosettacode.org/wiki/Brownian_tree

```
import java.awt.Graphics;
import java.awt.image.BufferedImage;
import java.util.*;
import javax.swing.JFrame;

public class BrownianTree extends JFrame implements Runnable {

    BufferedImage I;
    private List<Particle> particles;
    static Random rand = new Random();

    public BrownianTree() {
        super("Brownian Tree");
        setBounds(100, 100, 400, 300);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        I = new BufferedImage(getWidth(), getHeight(),
BufferedImage.TYPE_INT_RGB);
        I.setRGB(I.getWidth() / 2, I.getHeight() / 2, 0xff00);
        particles = new LinkedList<Particle>();
    }

    @Override
    public void paint(Graphics g) {
        g.drawImage(I, 0, 0, this);
    }

    public void run() {
        for (int i = 0; i < 20000; i++) {
            particles.add(new Particle());
        }
        while (!particles.isEmpty()) {
            for (Iterator<Particle> it = particles.iterator(); it.hasNext();) {
                if (it.next().move()) {
                    it.remove();
                }
            }
            repaint();
        }
    }

    public static void main(String[] args) {
        BrownianTree b = new BrownianTree();
        b.setVisible(true);
        new Thread(b).start();
    }

    private class Particle {

        private int x, y;

        private Particle() {
            x = rand.nextInt(I.getWidth());
            y = rand.nextInt(I.getHeight());
        }
    }
}
```

```
/* returns true if either out of bounds or collided with tree */
private boolean move() {
    int dx = rand.nextInt(3) - 1;
    int dy = rand.nextInt(3) - 1;
    if ((x + dx < 0) || (y + dy < 0)
        || (y + dy >= I.getHeight()) || (x + dx >= I.getWidth()))
    {
        return true;
    }
    x += dx;
    y += dy;
    if ((I.getRGB(x, y) & 0xff00) == 0xff00) {
        I.setRGB(x - dx, y - dy, 0xff00);
        return true;
    }
    return false;
}
}
```