Brownian Motion and Harmonic Functions

Harini Chandramouli [1] , Kiya Holmes [2], Brandon Reeves [3], Nora Stack [4]

Abstract

In this research we are looking at Kakutani's classical result on the connection between Brownian motion, a form of random movement, and harmonic functions, which are solutions to the Laplace equation. Kakutani's theorem is basically a generalization of the mean value property of harmonic functions. We will use this result to solve the Laplace equation in various regions with certain boundary conditions.

Walk on Spheres (WoS) is used to simulate the Brownian motion of a particle suspended in liquid. The average time needed for the particle to hit the boundary of certain regions will be discussed. The distribution of the point of first encounter with the boundary of the region is of interest to us. We will also discuss our use of conformal maps to find probability density functions on certain regions. Additionally, the rate of convergence of the Brownian motion to the boundary as well as the overall computational effort needed to estimate values of the harmonic function using the Monte Carlo algorithm will also be discussed. Lastly, we looked into less expensive real world applications of our research.

Acknowledgments

First, we wish to thank our advisor Dr. Igor Nazarov, for all of his guidance throughout the REU program. Additionally, we would also like to acknowledge Dr. Nicholas Boros for his help with our research during the REU program. We would like to extend our thanks Michigan State University and Lymann Briggs College for being our host institution. Finally, we would like to thank the National Security Agency for funding our research through grant number H98230-11-10222.

[1]University of Pittsburgh, Pittsburgh, PA 15213
[2]Medgar Evers College, Brooklyn, NY 11225
[3]Gonzaga University, Spokane, WA 99207
[4]St. Mary's College of Maryland, St. Mary City, MD 20686

Contents

1 Introduction

Given a region $\mathcal{R}$ with boundary condition u_0, a problem central to applied mathematics is solving for heat dissipation, population migration, chemical diffusion, etc. for points interior to $\mathcal{R}$. Using Fourier transforms, one can find an expression for the temperatures (population densities, chemical concentrations, etc.) at any point inside of the region $\mathcal{R}$ at a given time t. By letting t tend towards infinity, we obtain a solution that is no longer dependent on time, i.e. a steady-state solution. It is well-known that the functional solution to the steady-state equilibrium, say u, is a harmonic function, that is, $\Delta u = 0$.

In 1944, however, Kakutani (see [**?**]) showed that one can express the steady-state solution in terms of Brownian motion. He showed that to find the value of the steady-state solution at a particular point,one simply needs to consider a particle undergoing Brownian motion beginning from that point. Under Brownian motion, the particle will travel randomly until it first encounters the boundary. When this happens, one records the value at the boundary at the point of first encounter. After repeating this process a sufficiently large number of times, the mean of the recorded values will converge probabilistically to the value of the harmonic function inside of $\mathcal{R}$.

One efficient way to simulate Brownian motion is the Walk on Spheres method (introduced in [**?**]). We will use the Walk on Spheres method with discrete time steps in order to simulate Brownian motion. In this process, a walker beginning at an initial point takes a random step to a new point. From this point, the walker again takes a random step to a new point. This process continues until the walker is sufficiently close to the boundary.

In this paper, we discuss the probability density function for the point of first encounter. In essence, we wish to find a function that describes the relative probability that our random walk process will terminate on a segment of our boundary given an initial starting point. Furthermore, we shall explore how altering the lengths of the random walks will influence the rates of convergence to the boundary. Lastly, our paper will consider efficient ways to approximate a solution to the steady-state equilibrium while calculating the fewest number of points along the boundary.

2 A Solution to Laplace's Equation: The Half-Plane

In this section we will solve Laplace's equation on the half-plane, which will be pivotal in our exploration of Laplace's equation in more generalized regions. Find $u(x, y)$ where

$$\begin{aligned} u_{xx} + u_{yy} &= 0 \text{ with} && (1) \\ u(x, 0) &= u_0(x) \text{ for all x} && (2) \end{aligned}$$

Recall, that the Fourier transform of u(x,y) is $\tilde{u}(\omega, y)$ where

$$\tilde{u}(\omega, y) = \int_{-\infty}^{\infty} e^{i\omega x} u(x, y) dx$$

Fourier transforming both (1) and (2) with respect to x we obtain the new system:

$$\begin{aligned} -(\omega)^2 \tilde{u} + \tilde{u_{yy}} &= 0 && (3) \\ \tilde{u}(\omega, 0) &= \tilde{u_0}(\omega) && (4) \end{aligned}$$

Notice, that (3) is simply an ordinary differential equation with characteristic equation that has the solution:

$$\tilde{u}(\omega, y) = e^{-|\omega| y} \cdot C(\omega) \quad (5)$$

Utilizing (4) we find that

$$\begin{aligned} \tilde{u}(\omega, 0) &= e^{-|\omega| * 0} \cdot C(\omega) \\ &= C(\omega) \\ &= \tilde{u_0}(\omega) \end{aligned}$$

Hence, we know

$$\tilde{u}(\omega, y) = e^{-|\omega| y} \tilde{u_0}(\omega) \quad (6)$$

By applying the inverse Fourier transform to (6) and recognizing that the inverse Fourier transform of $e^{-|\omega| y}$ is simply the Poisson kernel, we find that

$$u(x, y) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{y}{(x - t)^2 + y^2} u_0(t) dt \quad (7)$$

From (7) we can find the probability density function for the point of first encounter for a particle undergoing Brownian motion from the point (x,y). Recall, that the expected value of a function is defined to be

$$E(u_0) = \int_{-\infty}^{\infty} u_0(x) f(x) dx$$

where f(x) is the probability distribution of the random variable.

Analogously, from (7) we know that the probability density function for the point of first encounter for a particle undergoing Brownian motion on the half-plane beginning from the point (x,y) is:

$$f(t) = \frac{1}{\pi} \frac{y}{(x-t)^2 + y^2} \tag{8}$$

3 Walk on Spheres Method

It is well known that the Random Walk on Circles effectively simulates Brownian motion on two-dimensional regions. The Random Walk on Circles process consists of taking stochastic jumps from one point, p_i to the next successive point, p_{i+1}, which is a specified distance, say a distance of r, away from p_i. The new point, p_{i+1}, is chosen by selecting a point from a uniform distribution along a circle of radius r centered at the previous point, p_i. It is important that this circle, from which p_{i+1} is chosen, lies entirely in the region. If the distance from the point p_i and the boundary, say d, is less than r, then the circle from which p_{i+1} is chosen has radius of d rather than radius r. Once the this process comes sufficiently close to the boundary, i.e. within a specified tolerance limit, the process terminates. It is important to note that this process will converge to the boundary with probability one. The exact rate of convergence depends upon the upper bound for the "lengths" of these random walks. We will discuss more about the relative rates of convergence in section (5)

4 Our Programs

4.1 General Process - Two Dimensional Regions

As stated in the previous section, the rate of convergence of this random walks method depends upon the lengths of these walks, i.e. the radius of the circle at each point. We can make this process converge quickly (i.e. in the fewest number of iterations) by increasing the size of the random walks. In

fact, the fastest way we can converge to the boundary is to pick the largest possible radius at each successive step of the random walks process. Thus, in finding the next successive point, say p_{i+1}, we first must find the distance between the previous point, p_i, and the boundary. We then set the length of the random walk, i.e. the radius of the circle from which p_{i+1} is chosen, to be this distance. Using this algorithm, we constructed several programs to simulate Brownian motion on several different regions. For some of these regions (such as the half-plane, the circle, etc) have well known solutions. Others (such as the parabolic regions, squares, triangles, etc.) have solutions which are difficult to find an analytic solution. Since the general algorithm for each of the programs is the same, we will only discuss how one finds the minimum distance from any point interior to the region and the boundary of that region.

4.2 Various Regions

4.2.1 Upper Half-Plane

For the half-plane, the maximum possible length for a random walk at any given point (x_0, y_0) is simply the value of the y-coordinate of the point, in this case y_0. Below, we have included a picture of the process. At each given stage, we plotted the point as well as the circle from which we select the next point. As you can see, the convergence rates were relatively quick despite the fact that the tolerance level was quite low.

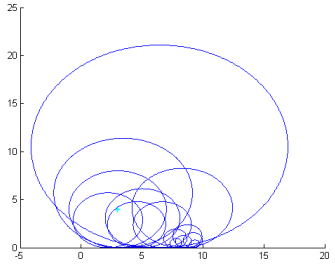


Figure 1: **Initial Point:** (3,4), **Count:** 16, **Tolerance:**0.001, **Ending Point:** $(7.1225, 8.2200 \times 10^{-7})$

4.2.2 Circle

Within a circular region, the maximum possible length of a random walk from any given point, say (x_0, y_0), is less-trivial than in the half-plane. We begin by noting that the minimum distance between (x_0, y_0) and a point on our boundary, say (x_b, y_b) is the length of the line segment between these two points. Furthermore, if this line segment is to be the shortest line from (x_0, y_o) to the boundary, we know this line segment must also be normal to the tangent line at the point (x_b, y_b). Since on a circle any normal line on the boundary of the circle must pass through the center of the circle, say (x_c, y_c), we know that the minimum distance from (x_0, y_0) to the boundary must be the radius of our circular region less the distance from the center of the circular region to the point (x_0, y_0). Below are some images from the program. The largest circle is the boundary region.

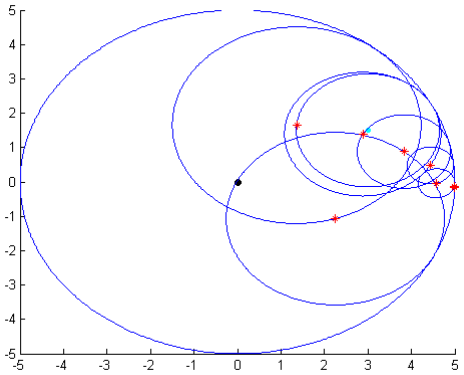


Figure 2: **Boundary Equation:** $x^2 + y^2 = 25$, **Initial Point:** (3,1.5), **Count:** 9, **Tolerance:** 0.001, **Ending Point:** (4.9972, -0.1506)

4.2.3 Parabola

For a parabolic region, we obtained the maximum possible length of any given random walk from a point (x_0, y_0) by finding the minimum of the derivative of the distance function between (x_0, y_0) and any point on the boundary of the region. We begin by noting that we can minimize the distance function by finding the minimum of the square of the distance function.

$$D^2 = (x_1 - x_0)^2 + (y_1 - y_0)^2$$
$$D^2 = (x_1 - x_0)^2 + (ax_1^2 - y_0)^2$$

$$\frac{dD}{dx} = 2(x_1 - x_0) + 2 * 2ax_1(ax_1^2 - y_0)$$
$$\frac{dD}{dx} = 4a^2x_1^3 - 4ax_1y_0 + 2x_1 - 2x_0$$

We then set the derivative equal to zero and we get these equations.

$$4a^2x_1^3 - 4ax_1y_0 + 2x_1 - 2x_0 = 0$$
$$2a^2x_1^3 - 2ax_1y_0 + x_1 - x_0 = 0$$

MATLAB then solves for the roots of this cubic equation. After disregarding any imaginary numbers, we select the x-value that provides the absolute minimum of the distance function.
Below, we have included an image of the random walks program on a parabolic region. For this picture or parabola was the parabola $y = x^2$ and our program starts at the point (1,2).

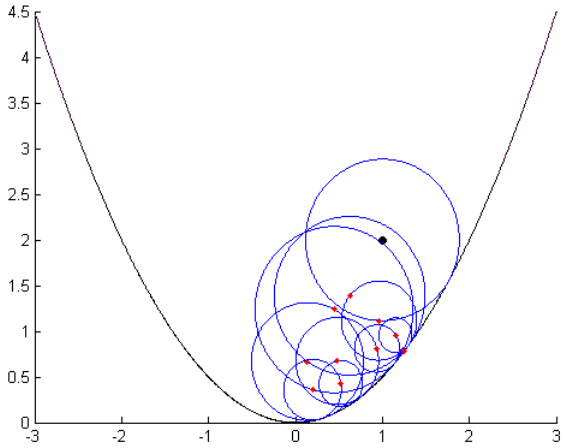


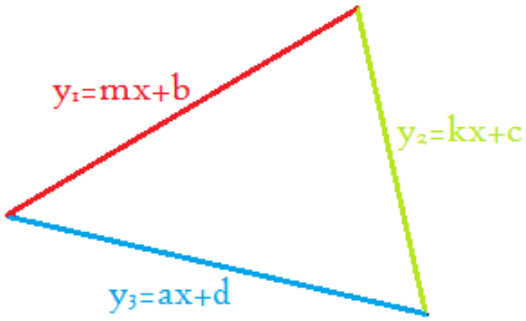
Figure 3: Random Walk on Circles in a Parabolic Region

4.2.4 Square

I think we should change this to be a rectangle, no sense in needlessly losing generality.... Finding the maximum possible length of a random walk from any point interior to a square region is trivial. If we place the lower-left corner of the rectangle at the origin, with the x-axis corresponding to the

width and the y-axis corresponding to the height we can easily obtain the maximum length of a random walk. After defining the width (w) and height (h) of this rectangle, we must simply chose the minimum value of x, (w - x), y, and (h - y).

4.2.5 Triangle



Using the equations

$$y_1 = mx + b$$
$$y_2 = kx + c$$
$$y_3 = ax + d$$

we are able to construct a triangle. We then find the minimum radius by taking the derivative of each line. For example, in y_1, the derivative is m. We then have the slope of the tangent line. Using the slope of each, we calculate the slope of the perpendicular line. Using y_1, we have

$$y = \frac{-1}{m}x + b$$

We can then use our point (x_0, y_0) to find b. We then get an equation in the form

$$y = \frac{-1}{m}x + (y_0 + \frac{1}{m}x_0)$$

which we then set equal to y_1

$$mx + b = \frac{-1}{m}x + (y_0 + \frac{1}{m}x_0)$$

and we then find out where these lines intersect. Thus we get the point (x_1, y_1). Once we repeat this process for y_2 and y_3 we use the distance formula to figure out the distance between (x_0, y_0) and each of our new points. Our program then picks the minimum of the three distances as our radius for a circle centered at (x_0, y_0). If the triangle is very obtuse, the distance calculated has the chance to extend outside of the actual triangle. However, this distance will never be shorter than a distance to a point inside the triangle. Thus, it does not create a problem.

4.2.6 Upper Quarter-Plane

The maximum length of a random walk on the quarter-plane (i.e. the first quadrant) is rather trivial: it is simply the minimum of our x-coordinate or our y-coordinate at any given point.
Below we have included a graphic to show Brownian simulation on the quarter plane.

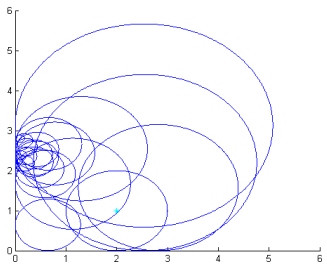


Figure 4: **Initial Point:** (2,1), **Count:** 21, **Tolerance:** 0.01, **Ending Point:** (0.6753,0.0051)

4.3 General Process - Three Dimensional Regions

For three dimensional regions, we utilize the process known as Random Walks on Spheres. Random Walks on Spheres is the three dimensional analog of the Random Walks on Circles method. The general process is the same as Random Walks on Circles, except for the fact that the next sequential point is selected from a uniform distribution on the surface of a sphere centered at the previous point.

In order to pick a point from a uniform distribution from the circle in the walk on circles method, we were able to pick θ from a uniform distribution. Intuition suggests that we can generalize this to a sphere by picking ψ and

θ from uniform distributions on [0,π] and [0,2π], respectively. Doing so, however, will result to a higher probability of picking a point near the poles of the sphere than at the equator of the sphere. Rather, it is well-known that one must chose θ from a uniform distribution on [0,pi] and ψ = arccos(2u - 1) where u comes from a uniform distribution on [0,1].

4.3.1 Upper Half Space

In the upper half space, the maximum possible radius is simply the z-coordinate the point we are considering. Fortunately enough, the program for the upper half space is almost identical to our program for the upper half plane.
One natural problem arising in three dimensions is the lack of clarity in images. For instance, a visual representation of our process is rather unintelligible as it is virtually impossible to ascertain what the program is doing purely from a visual representation in three dimensions.
I'm trying to think of a better way to describe this nondimensionalization

4.3.2 Sphere

The maximum length of a random walk on a spherical region is analogous to the maximum length of a random walk on a circular region. In this case, our maximum length of our random walk is simply the radius of our spherical region less distance from our current point (x_0, y_0, z_0) to the center of the sphere (x_c, y_c, z_c)

5 Rates of Convergence

A natural question to ask is how the "lengths" of the random walks affect the number of iterations the process takes to converge to the boundary. Intuition suggests that random walks of larger lengths will tend to converge much quicker to the boundary, as more distance is covered in fewer steps.

In order to determine the effect of altering the radius size, we constructed a program that would track the average number of iterations required for the random walks method to converge to the boundary as the lengths of the random walk changed. This program began with a small radius (size?) and performed the random walks method several times (1000) to find the average number of iterations required to converge. Then, the program augmented the radius by a small amount (size?) and repeated the process. This program terminated when the radius became sufficiently large (i.e. when it reached a user-specified value).

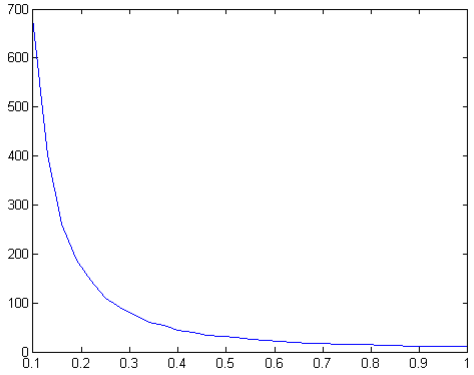


Figure 5: Number of Steps vs. Radius Size

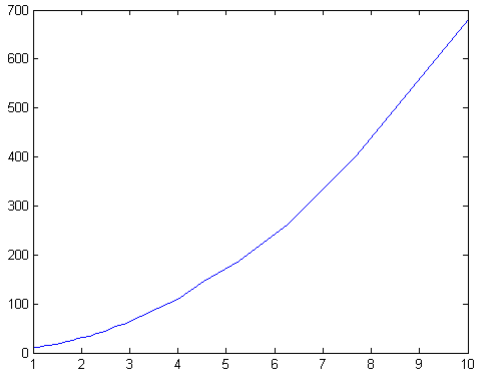


Figure 6: Number of Steps vs. $\frac{1}{RadiusSize}$

I think we need to make sure we know we are doing this part right about mentioning Einstein's theory

6 Probability Density Functions for Known Regions

6.1 Half-Plane

One of the main goals of our research this summer is to establish empirically a probability density function for the point of first encounter for a particle in a

region $\mathcal{D}$ under Brownian motion. Using the random walk on circles method as a way to simulate Brownian motion, we have constructed an algorithm that will allow us to simulate Brownian motion on the half-plane. In order to find an estimate for the probability density function for the boundary of the half-plane (the x-axis), we can simulate Brownian motion several thousand times (in this case 10,000), record the point of first encounter for each iteration, and then construct a histogram displaying the frequencies of the points of first encounter. Since the number of iterations are large, the shape of the histogram should match the theoretical PDF for the boundary.

Fortunately enough, the theoretical PDF for the half-plane is known to be the Cauchy-distribution of the form : $\frac{1}{\pi}\frac{a}{x^2+a^2}$ where is the y-coordinate of our initial starting point (i.e. the height of our initial point). When we executed the algorithm described above, we obtained a histogram describing the frequencies of the point of first encounter along the x-axis. We then plotted the known distribution on top of our histogram (with the appropriate scaling measures) and we obtained the following result:

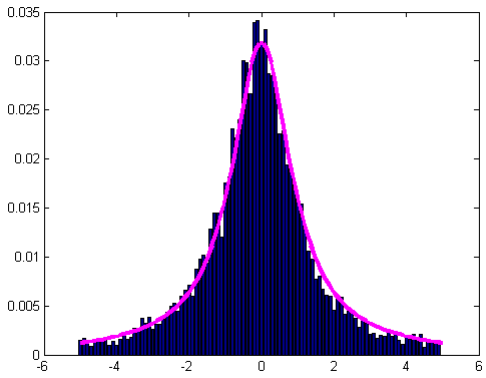


Figure 7: Empirical (blue) vs Theoretic (Cauchy Distribution-pink) and PDFs for half-plane

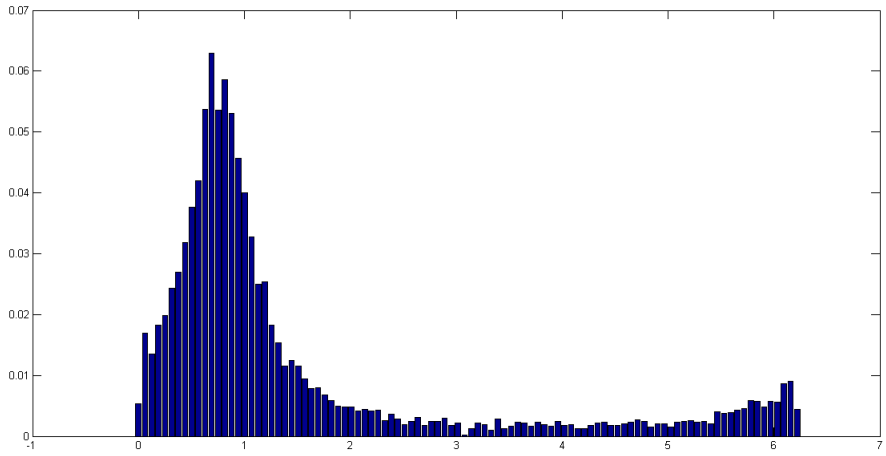
6.1.1 Goodness of Fit

We can perform Pearson's Chi-Squared Test to test whether or not our observed sample values came from the Cauchy distribution. The test statistic is $\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$. This test statistic follows a Chi Square distribution with k - 1 degrees of freedom. Using this test statistic, we were unable to reject

the null hypothesis that the sample observations did come from the Cauchy distribution at the $\alpha = .05$ significance level.

6.2 Circle

Similarly, we elected to create a histogram for the point of first encounter on a circular region. In order to do this, we converted our point of first encounter into polar form. In this case, since the radius of the point of first encounter must be, by definition, the radius of our circular region, the only difference between distinct points of first encounter were the θ values. In this case, we recorded the θ values and plotted them in a histogram.



As of yet, we have no theoretical distribution with which we can compare our results

7 Conformal Mappings and Probability Density Functions

To find the theoretical distribution for unknown regions, we can use conformal mapping ([**?**]) techniques. A conformal mapping is a bijective mapping between two regions that preserves angles between smooth curves passing through a point. The objective is to find a mapping from a region where the

probability density function for the point of first encounter is unknown to a region where the probability density function is known. We then use the conformal mapping - and the known probability density function - to find the unknown probability density function.

One powerful theorem that make use of is the Riemann Mapping Theorem, which states that for any open, simply-connected subset of the complex plane (that is not empty nor the entire complex plane itself) there exists a biholomorphic mapping to the upper half plane ([5]). With this theorem, we should be able to find a generalized algorithm for finding the probability density function for the point of first encounter on any such subset of the complex plane.

7.1 Empirical Probability on the Quarter-Plane

We made a program that performs walk on circles n number of times and stores the points where the program terminates. We then made a program that linearizes this data so we can make a histogram. If the program terminated at a point on the y-axis, then the program linearizes the point to be on the negative x-axis. If the program terminated at a point on the x-axis, then the program linearizes the point to be on the positive x-axis. We then used our histogram plotter to make a histogram of this data.

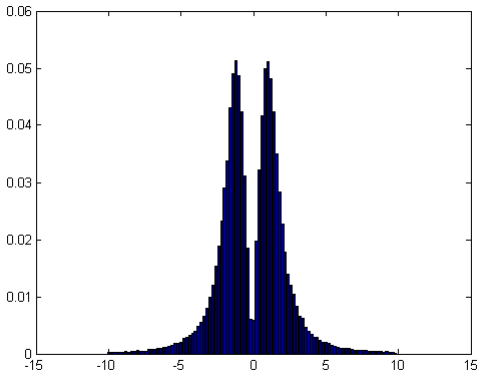


Figure 8: Approximate probability density of the quarter-plane with an **initial point** $(1, 1)$, **tolerance**=0.01, 100,000 **iterations** of Walk on Circles, and 100 **bins**.

The resulting histogram was in line with our expectations. The two biggest

peaks were at 1 on the x-axis and 1 on the y-axis (which corresponds to -1 on the x-axis in histogram above). There is such a drastic dip at 0 because there is no way that the point could actually reach 0, it would reach the tolerance before it ever hit 0.

7.2 Conformal Mappings on the Quarter-Plane

There is a conformal map that takes the quarter-plane into the half-plane, and this is the map $f(z) = z^2$.Since z is a complex number, we can say $z = x + \imath y$, for some arbitrary x and y such that $x, y \in \mathbb{R}$. Assume $u(x, y)$ satisfies Laplace's Equation in the half-plane,

$$\begin{aligned} f(z) &= z^2 \\ &= (x^2 - y^2) + (2xy)\imath \end{aligned}$$

We want to apply this conformal mapping to find h(x,y) in the quarter plane.

$$\begin{aligned} h(x, y) &= h(z) = u(f(z)) = u(x^2 - y^2, 2xy) \\ h_0(x) &= h(x, 0) = u(x^2, 0) \\ h_1(y) &= h(0, y) = u(-y^2, 0) \\ u_0(\xi) &= \begin{cases} h_0(\sqrt{\xi}) \text{ if } \xi > 0 \\ h_1(\sqrt{-\xi}) \text{ if } \xi < 0 \end{cases} \end{aligned}$$

Given the boundary condition $u_0(t)$ for $-\infty < t < \infty$, we know that our solution is

$$\begin{aligned} u(x_0, y_0) &= \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{y_0}{(x_0 - t)^2 + {y_0}^2} u_0(t) dt \\ &= \frac{1}{\pi} \int_{-\infty}^{0} \frac{y_0}{(x_0 - t)^2 + {y_0}^2} h_1(\sqrt{-t}) dt \\ &+ \frac{1}{\pi} \int_{0}^{\infty} \frac{y_0}{(x_0 - t)^2 + {y_0}^2} h_0(\sqrt{t}) dt \end{aligned}$$

Let t = -τ^2 for the first and t = τ^2 for the second integral.

$$\begin{aligned} u(x_0, y_0) = &\frac{1}{\pi} \int_{0}^{\infty} \frac{2\tau y_0}{(x_0 + \tau^2)^2 + {y_0}^2} h_1(\tau) d\tau \\ &+ \frac{1}{\pi} \int_{0}^{\infty} \frac{2 y_0 \tau}{(x_0 - \tau^2)^2 + {y_0}^2} h_1(\tau) d\tau \end{aligned}$$

Recall that $u({x_0}^2 - {y_0}^2, 2x_0y_0) = h(x_0, y_0)$, so we need to substitute

$$\begin{aligned} h(x_0, y_0) &= u({x_0}^2 - {y_0}^2, 2x_0y_0) \\ &= \frac{1}{\pi}\int_0^\infty \frac{4x_0y_0\tau}{\left({x_0}^2 - {y_0}^2 + \tau^2\right)^2 + \left(2x_0y_0\right)^2} h_1(\tau)d\tau \\ &+ \frac{1}{\pi}\int_0^\infty \frac{4x_0y_0\tau}{\left({x_0}^2 - {y_0}^2 - \tau^2\right)^2 + \left(2x_0y_0\right)^2} h_0(\tau)(\tau)d\tau \end{aligned}$$

Thus, we know that the PDF for the quarter-plane is:

$$\frac{1}{\pi}\frac{4x_0y_0\tau}{\left({x_0}^2 - {y_0}^2 + \tau^2\right)^2 + \left(2x_0y_0\right)^2}$$

on the y-axis and

$$\frac{1}{\pi}\frac{4x_0y_0\tau}{\left({x_0}^2 - {y_0}^2 - \tau^2\right)^2 + \left(2x_0y_0\right)^2}$$

on the x-axis.

7.3 Finding the Theoretical Probability Density Function for the Quarter-Plane

We will first look at the equation we derived from the previous section.

$$\frac{1}{\pi}\int_0^\infty \frac{4x_0y_0\tau}{\left({x_0}^2 - {y_0}^2 - \tau^2\right)^2 + \left(2x_0y_0\right)^2} d\tau$$

To solve this integral, we will first do a u-substitution. As this is a function of τ, we will take the derivative with respect to τ

$$\begin{aligned} u &= {x_0}^2 - {y_0}^2 - \tau^2 \\ du &= -2\tau d\tau \end{aligned}$$

Plugging this into the integral, we see

$$\frac{1}{\pi}\int_{-\infty}^{{x_0}^2 - {y_0}^2} \frac{2x_0y_0}{u^2 + \left(2x_0y_0\right)^2} du$$

$$\left[\frac{1}{\pi}\arctan\left(\frac{u}{2x_0y_0}\right)\right]_{-\infty}^{{x_0}^2 - {y_0}^2}$$

$$\frac{1}{\pi}\arctan\left(\frac{{x_0}^2 - {y_0}^2}{2x_0y_0}\right) + \frac{1}{2}$$

This is the formula we found for the probability density function on the x-axis. To find the probability density function on the y-axis, just take 1 minus the function on the x-axis, which is

$$\frac{1}{2} - \frac{1}{\pi} \arctan\left(\frac{{x_0}^2 - {y_0}^2}{2x_0y_0}\right)$$

A way to test to see if this is accurate is to plug in a point on the line $y = x$. Intuition suggests that any point on this line would have a $\frac{1}{2}$ chance of landing on either the x or y-axis. Plugging $y = x$ into this formula gives us $\frac{1}{2}$ for both the probability of landing on the x and y-axis.

7.4 Conformal Mapping of a Parabolic Region

Given a parabola x = -1 + $(\frac{y}{p})^2$,

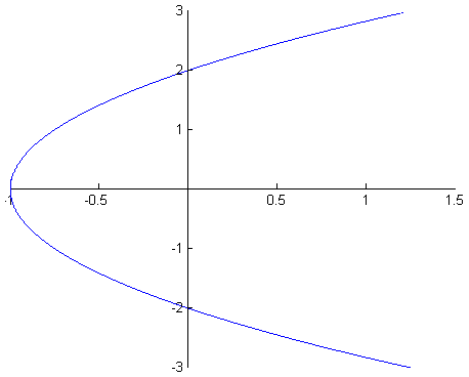


Figure 9: Parabola with p=2

f(x,y) = $i\cosh(\frac{\pi}{2}\sqrt{x+i\frac{2}{p}y}) = u + iv$ is the map that takes this parabola to the half-plane. This map can be rewritten as $f(x,y) = \frac{1}{2}(e^{i\frac{\pi}{2}+\sqrt{x+i\frac{2}{p}y}} + e^{i\frac{\pi}{2}-\sqrt{x+i\frac{2}{p}y}})$. After much simplification, we came up with the following formulas:

$$u(x,y) = \frac{1}{2}e^{\frac{\pi}{2}\left(\sqrt{\frac{x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}+1\right)}\cos\left(\sqrt{\frac{-x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}+1\right)$$

$$+\frac{1}{2}e^{\frac{-\pi}{2}\left(1-\sqrt{\frac{x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}\right)}\cos\left(\sqrt{\frac{-x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}+1\right)$$

$$v(x,y) = \frac{1}{2}e^{\frac{\pi}{2}\left(\sqrt{\frac{x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}+1\right)}\sin\left(\sqrt{\frac{-x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}+1\right)$$

$$+\frac{1}{2}e^{\frac{-\pi}{2}\left(1-\sqrt{\frac{x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}\right)}\sin\left(\sqrt{\frac{-x+\sqrt{x^2+(\frac{2}{p}y)^2}}{2}}+1\right)$$

From our parabola, we have the boundaries (x,$2\sqrt{x+1}$) and (x,-$2\sqrt{x+1}$). If we plug y = $\pm 2\sqrt{x+1}$, we find out where our boundary is 'sent' to:

$$h_0(x) : f(x, p\sqrt{x+1}) = (-\sinh(\frac{\pi}{2}\sqrt{x+1}), 0)$$
$$h_1(x) : f(x, -p\sqrt{x+1}) = (\sinh(\frac{\pi}{2}\sqrt{x+1}), 0)$$

We can describe our boundary in the half-plane in terms of the boundary of the parabola under the conformal mapping:

$$u_0(\tau) \quad = \quad \begin{cases} h_0\left((\frac{2}{\pi}\sinh^{-1}(-\tau))^2 - 1\right) \text{if} \tau < 0 \\ h_1\left((\frac{2}{\pi}\sinh -1(\tau))^2 - 1\right) \text{if} \tau > 0 \end{cases}$$

The Cauchy distribution states

$$u(x_0, y_0) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{y_0}{(x_0 - \tau)^2 + {y_0}^2} u_0(t) d\tau$$

Thus our solution in the parabolic region is

$$\begin{aligned} h(s_0, t_0) \quad &= \quad \int_{-1}^{\infty} \frac{\cosh(\frac{\pi}{2}\sqrt{\tau+1})t_0}{4\sqrt{\tau+1}(\left(s_0 + \sinh(\frac{\pi}{2}\sqrt{\tau+1})\right)^2 + {t_0}^2)} h_0(\tau) d\tau \\ &+ \quad \int_{-1}^{\infty} \frac{\cosh(\frac{\pi}{2}\sqrt{\tau+1})t_0}{4\sqrt{\tau+1}(\left(s_0 - \sinh(\frac{\pi}{2}\sqrt{\tau+1})\right)^2 + {t_0}^2)} h_1(\tau) d\tau \end{aligned}$$

Where $s_0 = u(x_0, y_0)$ and $t_0 = (x_0, y_0)$

8 Real World Application

Using the random walks method as we described before to solve Laplace's equation in a region $\mathcal{R}$, we must calculate the boundary value for each iteration of the random walk method. As a result, we must calculate values on the boundary a significant number of times. Although this presents no problem if the boundary condition is known *a priori*, the shear number of calculations this process requires is impractical if an expression for the boundary condition is not known. Rather, it is reasonable to assume that we do not have an expression for the boundary values, but we ascertain boundary values through observations methods. For instance, we could survey population densities or temperatures at a particular point on the boundary of the region. Determining this boundary value might be expensive. In this case, it

is natural to try and restrict the number of times one must determine the boundary value.

One such method to limit the number of times one must determine the boundary value is to choose n distinct points along the boundary. At each of these points, the researcher observes and records the boundary values. Then, using computer simulations, the researcher utilizes the random walks method to obtain a point on the boundary, say p_0. Typically, the researcher would calculate the boundary value at p_0. In this method, however, the researcher finds the closest of the n pre-selected points and instead uses the boundary value at this point as an approximation to the boundary value at p_0. Therefore, the researcher is able to limit the number of times he must compute the boundary. A natural question to ask is how one should pick such points along the boundary in order to ensure the highest possible degree of accuracy?

8.1 Obtaining Efficient Estimates - Gaussian Quadrature

Given that we want to limit the number of times we find the boundary values to n times, we want to find a way to approximate our solution. In essence, we want to find D_i and x_i for i=1,2,...,n such that

$$\int_{-\infty}^{\infty} D(x)u_0(x)dx \approx \sum_{i=1}^{n} D_i u(x_i) \tag{9}$$

where D(x) is the probability density function for the point of first encounter and $u_0(x)$ is the boundary-value condition.

Given any collection of n distinct points along the boundary, we can select our weights D_i such that the approximation is exact in (16) so long as u_0(x) is a polynomial with degree < n. In fact, it can be shown that we must pick our weights to satisfy the system of n equations given by:

$$\int_{-\infty}^{\infty} x^k D(x)dx = \sum_{i=1}^{n} {x_i}^k D_i \tag{10}$$

for k = 0, 1, 2, . . . , n -1

To show that (17) is exact for deg($u_0(x)$)<n let $u_0(x)$ be a polynomial of degree n-1. Hence,

$$u_0(x) = \sum_{j=0}^{n-1} \alpha_j x^j$$

Therefore,

$$\begin{aligned}
\int_{-\infty}^{\infty} D(x)u_0(x)dx &= \int_{-\infty}^{\infty} D(x)\sum_{j=0}^{n-1}\alpha_j x^j dx \\
&= \sum_{j=0}^{n-1}\alpha_j \int_{-\infty}^{\infty} x^j D(x)dx \\
&= \sum_{j=0}^{n-1}\alpha_j \sum_{i=1}^{n} {x_i}^j D_i \\
&= \sum_{i=1}^{n}\sum_{j=0}^{n-1}\alpha_j {x_i}^j D_i \\
&= \sum_{i=1}^{n} u_0(x_i)D_i
\end{aligned}$$

We can, however, obtain more accuracy through wise choices for our x_i terms.

8.2 Selecting x_is

One way to pick our x_is in an efficient manner is to pick our n x_is as the n real roots of the nth degree polynomial, say $p_n(x)$, orthogonal to all polynomials of lesser degree with respect to the distribution D(x). We define the inner product on pairs of polynomials p and q as

$$(p,q) = \int_{-\infty}^{\infty} p(x)q(x)D(x)dx$$

Since two polynomials are orthogonal if and only if their inner product is zero, we want to find p_n(x) such that (p_n(x),f(x)) = 0 for all f(x) with deg(f(x)) < n. One way to find such p_n(x) is to solve the following system of equations:

$$\begin{aligned}
(p_n,1) &= \int_{-\infty}^{\infty} p_n(x)D(x)dx = 0 \\
(p_n,x) &= \int_{-\infty}^{\infty} p_n(x)xD(x)dx = 0 \\
(p_n,x^2) &= \int_{-\infty}^{\infty} p_n(x)x^2D(x)dx = 0 \\
&\vdots \\
(p_n,x^{n-1}) &= \int_{-\infty}^{\infty} p_n(x)x^{n-1}D(x)dx = 0
\end{aligned}$$

where $p_n(x) = \sum_{i=0}^{n} \alpha_i x^i$.
Notice, however, that this system consists of n+1 unknowns but only n equations. As a result, in order to determine p_n(x) uniquely, we will choose p_n(x) to be a monic polynomial.

Now, we shall show that choosing our x_is and D_is in such a manner will result in an exact answer for (16) given u_0(x) is a polynomial of degree 2n-1. Let u_0(x) be a polynomial of degree up to 2n - 1.

By the division algorithm, we know there exists polynomials $\alpha(x)$ and r(x) such that $u_0(x) = \alpha(x)p_n(x)+r(x)$ with deg(α(x)) = deg(u_0(x)) - deg(p_n(x)) $\leq$ 2n -1 - n = n - 1 and deg(r(x))<deg(p_n(x))=n.

Hence,

$$\int_{-\infty}^{\infty} u_0(x)D(x)dx = \int_{-\infty}^{\infty} \alpha(x)p_n(x)D(x)dx + \int_{-\infty}^{\infty} r(x)D(x)dx$$

Since we constructed $p_n(x)$ to be orthogonal to all polynomials of lesser degree with respect to the weight D(x) and since α(x) has degree less than n, this reduces to

$$= \int_{-\infty}^{\infty} r(x)D(x)dx$$

Recall that we selected our weights, D_i in such a way that this integral would match the summation provided our degree was n-1 or less. Furthermore, we know that the degree of r(x) must be less than n. Hence, we have

$$= \sum_{i=1}^{n} r(x_i)D_i$$

Since we selected x_is to be the roots of p_n(x) we have that $\sum_{i=1}^{n} \alpha(x)p_n(x_i)D_i = 0$. Hence,

$$\begin{aligned} &= \sum_{i=1}^{n} \alpha(x)p_n(x_i)D_i + \sum_{i=1}^{n} r(x_i)D_i \\ &= \sum_{i=1}^{n} u_o(x_i)D_i \text{ and hence,} \\ \int_{-\infty}^{\infty} u_0(x)D(x)dx &= \sum_{i=1}^{n} u_o(x_i)D_i \end{aligned}$$

Thus, we know that we can select our x_is and our D_is in such a way to ensure that our new method is exact provided that $u_0(x)$ is a $2n-1^{th}$ or less

degree polynomial. Furthermore, if u_0(x) is sufficiently smooth (i.e. the 2n-1 derivatives exist) then we know our new method will be accurate provided u_0(x) is well approximated by a polynomial.

Since any distribution can be transformed to an uniform distribution on an interval, we can simplify the calculations of the abscissae x_i and weights w_i by implemeting the Legendre integration (a version of the Gaussian quadrature for a uniform weight function). The points and weights are available in Table 25.4 in [**?**] for high order of Legendre polynomials. They can be translated to the border of the upper half-plane by applying the tangent function (the inverse of the Cauchy cdf).

9 Conclusion

This summer our group looked at working with an alternative way to solve Laplaces equation in various regions. We did this using Brownian motion, a form of chaotic and random movement discovered by biologist Robert Brown. To simulate Brownian motion in MATLAB, we used the Walk on Spheres method with discrete time steps. This method was done in various regions and in various dimensions. The probability density function for the point of first encounter was also explored. These PDFs were used to find a function to represent the relative probability of where our random walk will terminate given an initial starting point. Using known PDFs, we could check the accuracy of our simulations in MATLAB. To find the PDFs for some regions where they werent known, we used conformal mappings. The last thing we explored this summer were inexpensive real world applications of our research. We started work on finding a way to pick a very small number of points along the boundary while maintaining a good amount of accuracy when approximating the solution to the steady-state equilibrium.

References

[1] Shizuo Kakutani. Two-dimensional Brownian motion and harmonic functions. *Proc. Imp. Acad. Tokyo*, 20:706–714, 1944.

[2] M. E. Muller. Some continuous Monte Carlo methods for the Dirichlet problem. *Ann. Math. Statist.*, 27:569589, 1956.

[3] Albert Einstein, Investigation on the Theory of the Brownian Movement, New York: Dover 1956.

[4] Milton Abramowitz, Irene Stegun, eds. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, New York: Dover Publications, 1972.

[5] Murray Spiegel, Seymour Lipschutz, John Schiller, Dennis Spellman. *Schaum's Outline of Complex Variables, 2ed (Schaum's Outline Series).* New York: McGraw-Hill, 2009

[6] David Poole. *Linear Algebra, A Modern Introduction, 2ed.* Thompson Brooks/Cole 2006

[7] Pierre Picco, Jaime San Martin. From Classical to Modern Probability: Cimpa Summer School 2001. Birkhäuser Bäsel 2004.