Brownian Motion and Harmonic Functions

Harini Chandramouli
Kiya Holmes
Brandon Reeves
Nora Stack

Michigan State University

2012.08.02
Points interior to the region describe population densities, temperatures, chemical concentrations, etc.
Points interior to the region describe population densities, temperatures, chemical concentrations, etc.

- The boundary values are given by $u_0(x,y)$
Points interior to the region describe population densities, temperatures, chemical concentrations, etc.

- The boundary values are given by \( u_0(x,y) \)

- After a sufficient amount of time, the temperature at points interior are time-independent
Laplace’s Equation

- Points interior to the region describe population densities, temperatures, chemical concentrations, etc.
  - The boundary values are given by $u_0(x,y)$
- After a sufficient amount of time, the temperature at points interior are time-independent
  - $u_{xx} + u_{yy} = 0$ (Laplace’s Equation)
Laplace’s Equation

Points interior to the region describe population densities, temperatures, chemical concentrations, etc.
- The boundary values are given by $u_0(x,y)$

After a sufficient amount of time, the temperature at points interior are time-independent
- $u_{xx} + u_{yy} = 0$ (Laplace’s Equation)
- In nice regions, the solution is well-known

Brownian Motion and Harmonic Functions
Kakutani (1944) showed that the solution, \( u(x_0, y_0) \), to Laplace’s equation can be approximated by considering Brownian motion from the point \( (x_0, y_0) \).
Kakutani (1944) showed that the solution, \( u(x_0, y_0) \), to Laplace’s equation can be approximated by considering Brownian motion from the point \((x_0, y_0)\).
Kakutani (1944) showed that the solution, \( u(x_0, y_0) \), to Laplace’s equation can be approximated by considering Brownian motion from the point \( (x_0, y_0) \).
Kakutani (1944) showed that the solution, \( u(x_0, y_0) \), to Laplace’s equation can be approximated by considering Brownian motion from the point \((x_0, y_0)\).
Kakutani (1944) showed that the solution, $u(x_0, y_0)$, to Laplace’s equation can be approximated by considering Brownian motion from the point $(x_0, y_0)$.
Kakutani (1944) showed that the solution, \( u(x_0, y_0) \), to Laplace's equation can be approximated by considering Brownian motion from the point \((x_0, y_0)\).
Kakutani (1944) showed that the solution, $u(x_0, y_0)$, to Laplace’s equation can be approximated by considering Brownian motion from the point $(x_0, y_0)$.

We can use random walks to simulate Brownian motion:

- Specifically, the random walks on circles (RWoC) and spheres (RWoS).
- We simulated Brownian motion in various regions and studied the probability density functions (PDFs) of the point of first encounter in these regions.
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
- Create circle with \((x_0, y_0)\) as the center
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
- Create circle with \((x_0, y_0)\) as the center
- Randomly pick a point \((x_1, y_1)\) on the circle

Brownian Motion and Harmonic Functions
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
- Create circle with \((x_0, y_0)\) as the center
- Randomly pick a point \((x_1, y_1)\) on the circle
- \((x_1, y_1)\) is either on the boundary or is somewhere else in the region
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
- Create circle with \((x_0, y_0)\) as the center
- Randomly pick a point \((x_1, y_1)\) on the circle
- \((x_1, y_1)\) is either on the boundary or is somewhere else in the region
- If \((x_1, y_1)\) if on the boundary then walk on circles ends
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
- Create circle with \((x_0, y_0)\) as the center
- Randomly pick a point \((x_1, y_1)\) on the circle
- \((x_1, y_1)\) is either on the boundary or is somewhere else in the region
- If \((x_1, y_1)\) if on the boundary then walk on circles ends
- If not, create circle with \((x_1, y_1)\) as center
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
- Create circle with \((x_0, y_0)\) as the center
- Randomly pick a point \((x_1, y_1)\) on the circle
- \((x_1, y_1)\) is either on the boundary or is somewhere else in the region
- If \((x_1, y_1)\) if on the boundary then walk on circles ends
- If not, create circle with \((x_1, y_1)\) as center
- Randomly pick a point \((x_2, y_2)\) on the circle
Walk on Circles

- Pick point \((x_0, y_0)\) in the region
- Create circle with \((x_0, y_0)\) as the center
- Randomly pick a point \((x_1, y_1)\) on the circle
- \((x_1, y_1)\) is either on the boundary or is somewhere else in the region
- If \((x_1, y_1)\) if on the boundary then walk on circles ends
- If not, create circle with \((x_1, y_1)\) as center
- Randomly pick a point \((x_2, y_2)\) on the circle
- Continue until process until you hit boundary of area
Walk on Circles in Regions

- Made programs to simulate walk on circles for:
Walk on Circles in Regions

- Made programs to simulate walk on circles for:
  - Line
Walk on Circles in Regions

- Made programs to simulate walk on circles for:
  - Line
  - Circle (Analytic solution known)
  - Upper Half-Plane (Analytic solution is known)
Made programs to simulate walk on circles for:

- Line
- Circle (Analytic solution known)
- Upper Half-Plane (Analytic solution is known)
- Parabola
- Quarter-Plane
Made programs to simulate walk on circles for:
- Line
- Circle (Analytic solution known)
- Upper Half-Plane (Analytic solution is known)
- Parabola
- Quarter-Plane
- Square
- Triangle
Made programs to simulate walk on circles for:
- Line
- Circle (Analytic solution known)
- Upper Half-Plane (Analytic solution is known)
- Parabola
- Quarter-Plane
- Square
- Triangle
- Upper Half-Space
- Sphere
Beginning at \((x_0, y_0)\), with \(y_0 > 0\) we simulate Brownian motion on the upper half plane.
Beginning at \((x_0, y_0)\), with \(y_0 > 0\) we simulate Brownian motion on the upper half plane.
Beginning at \((x_0, y_0)\), with \(y_0 > 0\) we simulate Brownian motion on the upper half plane.
How did our simulation perform?
Solution on the half-plane is known:

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

Where \( f_{y_0}(x_0 - \tau) = \frac{1}{\pi y_0} \frac{1}{(x_0 - \tau)^2 + y_0^2} \)

Hence, our PDF is:

\[ f(x) = \frac{1}{\pi y_0} \frac{1}{x^2 + y_0^2} \]
More General Regions

Solution on the half-plane is known:

\[ u(x_0, y_0) = \int_{-\infty}^{\infty} f_{y_0}(x_0 - \tau)u_0(\tau)d\tau \]

\[ = \int_{-\infty}^{\infty} \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} u_0(\tau)d\tau \]

Where

\[ f_{y_0}(x_0 - \tau) = \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} \]

Hence, our PDF is:

\[ f(x) = \frac{1}{\pi} \frac{y_0}{(x_0 - x)^2 + y_0^2} \]

What about more general regions in the plane?

Conformal Mappings

Map one region bijectively into another region

Riemann Mapping Theorem

Brownian Motion and Harmonic Functions
Solution on the half-plane is known:

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

Where \( f_{y_0}(x_0 - \tau) = \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} \)
Solution on the half-plane is known:

\[ u(x_0, y_0) = \int_{-\infty}^{\infty} f_{y_0}(x_0 - \tau)u_0(\tau) \, d\tau \]

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

Where \( f_{y_0}(x_0 - \tau) = \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} \)

Hence, our PDF is:

\( f(x) = \frac{1}{\pi} \frac{y_0}{(x_0 - x)^2 + y_0^2} \)

What about more general regions in the plane?
More General Regions

- Solution on the half-plane is known:

  \[ u(x_0, y_0) = \int_{-\infty}^{\infty} f_{y_0}(x_0 - \tau)u_0(\tau)d\tau \]

  \[ = \int_{-\infty}^{\infty} \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2}u_0(\tau)d\tau \]

- Where \( f_{y_0}(x_0 - \tau) = \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} \)

- Hence, our PDF is:

  \[ f(x) = \frac{1}{\pi} \frac{y_0}{(x_0 - x)^2 + y_0^2} \]

- What about more general regions in the plane?
  - Conformal Mappings
Solution on the half-plane is known:

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

Where \( f_{y_0}(x_0 - \tau) = \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} \)

Hence, our PDF is:

\[
    f(x) = \frac{1}{\pi} \frac{y_0}{(x_0-x)^2 + y_0^2}
\]

What about more general regions in the plane?

- Conformal Mappings
- Map one region bijectively into another region
More General Regions

- Solution on the half-plane is known:

\[ u(x_0, y_0) = \int_{-\infty}^{\infty} f_{y_0}(x_0 - \tau)u_0(\tau)d\tau \]

\[ = \int_{-\infty}^{\infty} \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} u_0(\tau)d\tau \]

- Where \( f_{y_0}(x_0 - \tau) = \frac{1}{\pi} \frac{y_0}{(x_0 - \tau)^2 + y_0^2} \)

- Hence, our PDF is:

\[ f(x) = \frac{1}{\pi} \frac{y_0}{(x_0 - x)^2 + y_0^2} \]

- What about more general regions in the plane?
  - Conformal Mappings
  - Map one region bijectively into another region
  - Riemann Mapping Theorem
Using Conformal Mappings

PDF

- x-axis

\[
\frac{1}{\pi} \frac{4x_0 y_0 \tau}{(x_0^2 - y_0^2 - \tau^2)^2 + (2x_0 y_0)^2}
\]

- y-axis

\[
\frac{1}{\pi} \frac{4x_0 y_0 \tau}{(x_0^2 - y_0^2 + \tau^2)^2 + (2x_0 y_0)^2}
\]

PDF

- x-axis

\[
f(x) = \frac{1}{\pi} \frac{y_0}{(x_0 - x)^2 + y_0^2}
\]
What if we have no expression for our boundary values, but we can compute these values instead at particular points?
Real World Applications

- What if we have no expression for our boundary values, but we can compute these values instead at particular points?
  - These points might be expensive
What if we have no expression for our boundary values, but we can compute these values instead at particular points?

- These points might be expensive
- Our process requires computing many boundary values
Real World Applications

- What if we have no expression for our boundary values, but we can compute these values instead at particular points?
  - These points might be expensive
  - Our process requires computing many boundary values
- Can we limit the number of times we compute the boundary values while still maintaining accurate approximations?
What if we have no expression for our boundary values, but we can compute these values instead at particular points?

- These points might be expensive
- Our process requires computing many boundary values

Can we limit the number of times we compute the boundary values while still maintaining accurate approximations?
What if we have no expression for our boundary values, but we can compute these values instead at particular points?

- These points might be expensive
- Our process requires computing many boundary values

Can we limit the number of times we compute the boundary values while still maintaining accurate approximations?
Real World Applications

- What if we have no expression for our boundary values, but we can compute these values instead at particular points?
  - These points might be expensive
  - Our process requires computing many boundary values
- Can we limit the number of times we compute the boundary values while still maintaining accurate approximations?

![Diagram showing points X1 and X2 on a boundary with crosses at various points along the boundary.](image)
Real World Applications

- What if we have no expression for our boundary values, but we can compute these values instead at particular points?
  - These points might be expensive
  - Our process requires computing many boundary values
- Can we limit the number of times we compute the boundary values while still maintaining accurate approximations?
Recall:

\[ u(x, y) = \int_{-\infty}^{\infty} D(\tau)u_0(\tau)\,d\tau \]

where \( D \) is the probability density function

Assumption: \( u_0 \) is a polynomial

Then we can find some numbers \( D_i \) such that

\[ \int_{-\infty}^{\infty} D(\tau)u_0(\tau)\,d\tau = \sum_{i=1}^{10} D_i u_0(x_i) \]

where \( u_0 \) is up to a 9th degree polynomial

But we can do better, \( u_0 \) can be up to a \( 2^{10} - 1 \) degree polynomial

Brownian Motion and Harmonic Functions
Recall:

\[ u(x, y) = \int_{-\infty}^{\infty} D(\tau) u_0(\tau) d\tau \]

where \( D \) is the probability density function

**Assumption**: \( u_0 \) is a polynomial
Recall:

\[ u(x, y) = \int_{-\infty}^{\infty} D(\tau)u_0(\tau)d\tau \]

where \( D \) is the probability density function

**Assumption:** \( u_0 \) is a polynomial

Then we can find some numbers \( D_i \) such that

\[ \int_{-\infty}^{\infty} D(\tau)u_0(\tau)d\tau = \sum_{i=1}^{10} D_i u_0(x_i) \]

where \( u_0 \) is up to a 9th degree polynomial
Recall:

\[ u(x, y) = \int_{-\infty}^{\infty} D(\tau)u_0(\tau) d\tau \]

where \( D \) is the probability density function

Assumption: \( u_0 \) is a polynomial

Then we can find some numbers \( D_i \) such that

\[ \int_{-\infty}^{\infty} D(\tau)u_0(\tau) d\tau = \sum_{i=1}^{10} D_i u_0(x_i) \]

where \( u_0 \) is up to a 9th degree polynomial

But we can do better, \( u_0 \) can be up to a \( 2(10) - 1 \) degree polynomial
Given $D(x)$ (probability density function) we can construct an inner product on polynomials as

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

Brownian Motion and Harmonic Functions
Given $D(x)$ (probability density function) we can construct an inner product on polynomials as

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

Given this inner product, we can find (by Gram-Schmidt Process) the $n$th degree polynomial, $p_n(x)$, orthogonal to all polynomials of lesser degree.
Given $D(x)$ (probability density function) we can construct an inner product on polynomials as

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

Given this inner product, we can find (by Gram-Schmidt Process) the $n$th degree polynomial, $p_n(x)$, orthogonal to all polynomials of lesser degree

- Pick $x_i$ as the roots of $p_n(x)$
How do we obtain exact answers up to degree $2(10)-1$?
Checking the Efficiency

How do we obtain exact answers up to degree 2(10)-1?

- Let $u_0(\tau)$ be our boundary condition, and $\deg(u_0(\tau)) = 19$
- $u_0(\tau) = \alpha(\tau)p_{10}(\tau) + r(\tau)$ where $\deg(r) < \deg(p_{10}) = 10$
- Let $D(\tau)$ represent the distribution function

$$\int_{-\infty}^{\infty} u_0(\tau)D(\tau)d\tau$$
Checking the Efficiency

- How do we obtain exact answers up to degree 2(10)-1?
  - Let $u_0(\tau)$ be our boundary condition, and $\deg(u_0(\tau)) = 19$
  - $u_0(\tau) = \alpha(\tau)p_{10}(\tau) + r(\tau)$ where $\deg(r) < \deg(p_{10}) = 10$
  - Let $D(\tau)$ represent the distribution function

$$\int_{-\infty}^{\infty} u_0(\tau)D(\tau)d\tau = \int_{-\infty}^{\infty} \alpha(\tau)p_{10}(\tau)D(\tau)d\tau + \int_{-\infty}^{\infty} r(\tau)D(\tau)d\tau$$
How do we obtain exact answers up to degree $2(10)-1$?

Let $u_0(\tau)$ be our boundary condition, and $\deg(u_0(\tau)) = 19$

$u_0(\tau) = \alpha(\tau)p_{10}(\tau) + r(\tau)$ where $\deg(r) < \deg(p_{10}) = 10$

Let $D(\tau)$ represent the distribution function

$$
\int_{-\infty}^{\infty} u_0(\tau)D(\tau)d\tau = \int_{-\infty}^{\infty} \alpha(\tau)p_{10}(\tau)D(\tau)d\tau + \int_{-\infty}^{\infty} r(\tau)D(\tau)d\tau
$$

$$= \int_{-\infty}^{\infty} r(\tau)D(\tau)d\tau$$
How do we obtain exact answers up to degree 2(10)-1?

Let \( u_0(\tau) \) be our boundary condition, and \( \deg(u_0(\tau)) = 19 \)

\[ u_0(\tau) = \alpha(\tau)p_{10}(\tau) + r(\tau) \text{ where } \deg(r) < \deg(p_{10}) = 10 \]

Let \( D(\tau) \) represent the distribution function

\[
\int_{-\infty}^{\infty} u_0(\tau)D(\tau)d\tau = \int_{-\infty}^{\infty} \alpha(\tau)p_{10}(\tau)D(\tau)d\tau + \int_{-\infty}^{\infty} r(\tau)D(\tau)d\tau
\]

\[
= \int_{-\infty}^{\infty} r(\tau)D(\tau)d\tau
\]

\[
= \sum_{i=1}^{10} r(x_i)D_i
\]
Checking the Efficiency

- How do we obtain exact answers up to degree 2(10)-1?
  - Let $u_0(\tau)$ be our boundary condition, and $\text{deg}(u_0(\tau)) = 19$
  - $u_0(\tau) = \alpha(\tau)p_{10}(\tau) + r(\tau)$ where $\text{deg}(r) < \text{deg}(p_{10}) = 10$
  - Let $D(\tau)$ represent the distribution function

\[
\int_{-\infty}^{\infty} u_0(\tau)D(\tau) d\tau = \int_{-\infty}^{\infty} \alpha(\tau)p_{10}(\tau)D(\tau) d\tau + \int_{-\infty}^{\infty} r(\tau)D(\tau) d\tau
\]

\[
= \int_{-\infty}^{\infty} r(\tau)D(\tau) d\tau
\]

\[
= \sum_{i=1}^{10} r(x_i)D_i
\]

\[
= \sum_{i=1}^{10} \alpha(x_i)p_{10}(x_i)D_i + \sum_{i=1}^{10} r(x_i)D_i
\]
Checking the Efficiency

- How do we obtain exact answers up to degree 2(10)-1?
  - Let $u_0(\tau)$ be our boundary condition, and $\text{deg}(u_0(\tau)) = 19$
  - $u_0(\tau) = \alpha(\tau)p_{10}(\tau) + r(\tau)$ where $\text{deg}(r) < \text{deg}(p_{10}) = 10$
  - Let $D(\tau)$ represent the distribution function

\[
\int_{-\infty}^{\infty} u_0(\tau)D(\tau)d\tau = \int_{-\infty}^{\infty} \alpha(\tau)p_{10}(\tau)D(\tau)d\tau + \int_{-\infty}^{\infty} r(\tau)D(\tau)d\tau
\]

= \int_{-\infty}^{\infty} r(\tau)D(\tau)d\tau

= \sum_{i=1}^{10} r(x_i)D_i

= \sum_{i=1}^{10} \alpha(x_i)p_{10}(x_i)D_i + \sum_{i=1}^{10} r(x_i)D_i

= \sum_{i=1}^{10} u_0(x_i)D_i
So if $u_0$ is a “nice” (smooth) function, then

$$u(x, y) = \int_{-\infty}^{\infty} D(\tau) u_0(\tau) d\tau$$

$$\approx \sum_{i=1}^{10} u_0(x_i) D_i$$

This will be a good approximation
Summary

- Brownian Motion and Laplace’s Equation
- Walk on Circles and Spheres
- Simulating Walk on Circles and Spheres in different regions
- Probability Density Functions and Conformal Mapping Techniques
- Less “Expensive” Real World Applications
Acknowledgements

- NSA
  - Grant: H98230-11-10222.
- Dr. Igor Nazarov
- Dr. Nick Boros
Questions?