MONTE CARLO SOLUTION TO LAPLACE'S EQ ON A RECTANGULAR DOMAIN

MATT CASSINI, MARISSAH MCNEIL, AND MOISES RAMOS

ABSTRACT. We use the Tour Du Wino Method, an algorithm based on Monte-Carlo simulation, to approximate a solution to Laplace's Eq on a rectangular domain with Dirichlet boundary conditions

1. Introduction

We seek to solve Laplace's Equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \tag{1}$$

on a rectangular domain $0 \leq x \leq 7, \quad 0 \leq y \leq 9$ with boundary conditions $u(0, y) = u(7, y) = u(x, 0) = 0$ and $u(x, 9) = 12$

2. Background

Write about application (electric) and how this equation comes about

3. Implementation

We programmed a Monte-Carlo simulation using the Tour Du Wino method that utilizes a 2D random walker on a uniform Cartesian grid that records the boundary point on which it lands. It then repeats this process m times starting at this same point. It then averages the value of the boundary condition at each point recorded and assigns this to the value of the solution at the given point.[1]

The proof of this is very straightforward. The expected value for some point a where R(a) is the value it will take is

$E[R(a)] = \frac{1}{4}(E[R(b)] + E[R(c)] + E[R(d)] + E[R(e)])$

where b,c,d, and e are the neighboring points.

This can be rewritten as

$$u_{ij} = \frac{1}{4}(u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1}) \tag{2}$$

which can be rearranged as

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{ij} = 0$$

The centered finite difference approximations for the second partial derivatives are

---

Date: December 19, 2022.
Math 450H Final Project.

$$u_{xx} = \frac{u_{i+1,j} + u_{i-1,j} - 2u_{i,j}}{h^2} \qquad u_{yy} = \frac{u_{i,j+1} + u_{i,j-1} - u_{ij}}{h^2}$$

It is now clear the above formulation can be expressed as $h^2 u_{xx} + h^2 u_{yy} = 0$ and dividing away h^2 leaves us $u_{xx} + u_{yy} = 0$ which is then Laplace's Eq.

We parallelized the algorithm and have the following

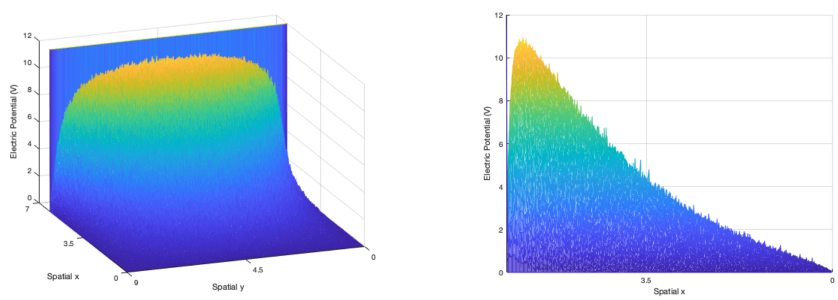
Algorithm 1 Tour Du Wino

1: Generate $1xn^2$ array $\sim U(0,1)$

---

4. Computational Results

Figure 1. Numerical solution with a grid size of 350x350 and 350 realizations



4.1. **Plots.**

4.2. **Convergence Towards Analytical Solution.**
Put in plots when error analysis code finishes

4.3. **Computation Time.**
We tested various different grid sizes and number of trials as seen above. We have found the computation time to be $O(n^4 m^2)$ where n is the size of the grid and m is the number of trials.

For $n = 8$ and $m = 10,000$, the computation time was 0.47 seconds. For the final solution plot above, we used $n = m = 350$ and found the computation time to be 553 minutes or approximately 9.2 hours.

References

[1] S.J. Farlow. *Partial Differential Equations for Scientists and Engineers*. Dover Books on Mathematics. Dover Publications, 2012.

Department of Mathematical Sciences, New Jersey Institute of Technology, University Heights, Newark, NJ 07102
Email address: mc225@njit.edu

Department of Mathematical Sciences, New Jersey Institute of Technology, University Heights, Newark, NJ 07102
Email address: mm2458@njit.edu

Department of Mathematical Sciences, New Jersey Institute of Technology, University Heights, Newark, NJ 07102
Email address: mlr4@njit.edu