A Finite Difference Approach and Its Error Estimate to the Two-Dimensional Poisson Equation for Dirichlet Boundary Conditions

*Matthew Whalen**

Department of Computer Science, University of Kentucky, Lexington, KY

https://doi.org/10.33697/ajur.2025.140

Students: mrwh233@uky.edu, mrwhalen22@gmail.com*
*Mentor: gi6036@wayne.edu**

ABSTRACT
This study introduces a regular five-point finite difference method for approximating the two-dimensional Poisson equation with Dirichlet boundary condition for convex polygonal domains. The Poisson equation frequently emerges in many fields of science and engineering, such as field potentials and heat transfer. As exact solutions are rarely possible, numerical approaches are important to develop efficient and practical modeling. This introductory paper addresses the uniqueness of the problem, finite difference discretization, consistency of the problem, and maximum norm error analysis. We also provide numerical results that not only validate theoretical results but also demonstrate the method's efficiency. Principally, this paper intends to serve as a compilation of the research which underpins the finite difference methods in a way that is unified, consistent, and accessible to undergraduates. Additionally, we have made the MATLAB code for these results publicly available at the end of this paper for reference and practical implementation.

KEYWORDS
Dirichlet; Poisson; Convex; Finite Differences; Norm; Computer Simulation; Numerical Methods

INTRODUCTION

Many physical problems in engineering and physics can be modeled mathematically, giving researchers the chance to precisely model the solutions, often via computers. These modeled problems are often governed by partial differential equations. The two-dimensional Poisson equation is a particular type of partial differential equation that describes field potentials, heat transfer, image restoration and denoising, and many other applications. To be valuable, the two-dimensional Poisson problem needs to be quickly solvable with a high degree of accuracy to ensure valid modeling of these problems. Since the equation is a partial differential equation (PDE), solving problems of its type is challenging to do using standard algebraic methods. Because only a small number of these PDEs have simple algebraic solutions, we will attempt to solve the two-dimensional Poisson equation using sufficiently accurate numerical approximations. We refer the readers to[1–6] for a comprehensive discussion on partial differential equations, numerical analysis and its applications.

Finite-difference,[7, 8] finite element,[9, 10] and finite volume method[11, 12] are three useful methods to numerically solve partial differential equations. This study uses the finite difference method to numerically approximate the solution to the Poisson equation. It works by replacing the continuous derivative operators with approximate finite differences. The finite difference method is simple, effective, and one of the oldest methods to solve the Poisson equation. Although it is one of the oldest methods ever devised, comprehensive information is difficult to find compiled in a single reference. Therefore, this paper provides a complete study of the two-dimensional Poisson equation through a literature review, proof of uniqueness and consistency results using the 5-point difference scheme, theoretical error analysis and numerical results.

One of the challenges of the two-dimensional version of this problem is the boundary constraints become complicated and take up a large portion of the solution's runtime.[13] These irregular domains tend to show singular behavior near sharp corners when using numerical solutions.[14–17] In the presence of discontinuous coefficients and singular source terms, these problems become more challenging.[18, 19] Due to this, we will use rectangular domains, which simplify the method and allow for easier understanding. For resources on problems involving non-convex domain, please refer to[5, 20–22] for extra study.

The rest of the article is organized as follows: In *Methods and Procedures*, we introduce our model problem, which is the Poisson equation, and prove its uniqueness. Then, we present a five-point finite difference scheme to approximate the solution to the Poisson equation. In *Error Estimates*, we present the theoretical maximum norm error estimates. Code results and discussion are presented in *Results* to validate the theoretical results and the efficiency of the method. Lastly, we conclude with a discussion of potential future works and a reiteration of the impact of this work. Please note that the methods, theorems, and proofs presented in the following sections are drawn from other works, and as such are available in the references listed in each section.

METHODS AND PROCEDURES

Finite difference methods are techniques to find approximate solutions to ordinary differential equations (ODEs) and partial differential equations (PDEs) numerically. They are based on the idea of replacing the ordinary or partial derivatives with a finite difference quotient. In some sense, a finite difference formulation offers a more direct and intuitive approach to the numerical solution of partial differential equations than other formulations.

Poisson equation

In this section we discuss the Poisson equation in two dimensions

$$\begin{aligned} -\Delta u(x,y) &= f(x,y) \text{ in } \Omega \\ u(x,y) &= g(x,y) \text{ on } \partial\Omega \end{aligned} \qquad \textbf{Equation 1.}$$

where the operator $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ stands for the Laplacian operator and f is a given source function. The solution u of **Equation 1** is an unknown scalar potential function. In the Cartesian coordinate system, the two-dimensional Poisson equation can be written as

$$-\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = f(x,y) \qquad \textbf{Equation 2.}$$

where *x, y* represent independent spatial dimensions. Since the equation does not have a time-dependent component, there are no initial conditions, making it a boundary value problem. The domain denoted as $\Omega = (a,b) \times (c,d)$ representing a rectangle grid of this size and is subject to a Dirichlet boundary condition. The boundary of the domain denoted as $\partial\Omega$. Next, we present the uniqueness theorem for the Poisson's equation. We refer interested readers to[1, 23–26] and references therein.

Uniqueness of Solutions to the Poisson Equation

The general version of the uniqueness theorem can be found in[25]. However, we will state and derive the two-dimensional version of the uniqueness theorem below to verify the uniqueness of the solution to equation 2.

Theorem 1. *Let Ω be an open subset in $\mathbb{R}^2$. Then there exists at most one solution u in Ω to* ***Equation 1*** *with Dirichlet boundary conditions.*

Proof. Suppose that p and q are two solutions to **Equation 1**. Let s be the difference between these two solutions such that,

$$s = p - q. \qquad \text{Equation 3.}$$

From **Equation 1** and **Equation 3** we get,

$$\begin{aligned} -\Delta s &= -\Delta(p-q) \ \ in \ \ \Omega \\ s &= p - q = g - g = 0 \ \ on \ \ \partial\Omega. \end{aligned} \qquad \text{Equation 4.}$$

Due to the linearity property of the Poisson's equation, and since p and q are solutions to **Equation 1**, we have $-\Delta(p - q) = -\Delta(p) + \Delta(q) = f - f = 0 \ \ in \ \ \Omega$, with $s = 0 \ \ on \ \ \partial\Omega$. This gives that $-\Delta s = 0 \ \ in \ \ \Omega$. Thus, we can see that s is twice differentiable and satisfies the Laplace equation and hence it is a harmonic function. The maximum principle for harmonic functions says that every non-constant s must attain its maximum and minimum values on the boundary $\partial\Omega$. Since $s = 0$ on the boundary $\partial\Omega$, by **Equation 3**, and s attains its maximum and minimum values on the boundary s must be equal to 0 in all of the domain Ω. Since $s = p - q$ we have $p = q$ for all x, y in Ω. Therefore, we can conclude that only one solution exists to Poisson's equation (**Equation 1**). □

Finite Difference Scheme

This section presents the five-point finite difference stencil[27] for the two-dimensional Poisson equation, which involves discretizing **Equation 1** into a system of linear equations solvable through iterative methods. The following three steps outline the process of generating the discretized system of equations.[7]

Step 1: Generate a mesh.

The mesh can be generated in a uniform Cartesian coordinate system as follows:

$$x_i = a + ih_x, \qquad i = 0, 1, 2, 3, ..., n \qquad h_x = \frac{b-a}{n} \qquad \text{Equation 5.}$$

$$y_j = c + jh_y, \qquad j = 0, 1, 2, 3, ..., m \qquad h_y = \frac{d-c}{m} \qquad \text{Equation 6.}$$

Let $u(x_i, y_j)$ be the exact solution at a typical point (x_i, y_j) in domain Ω and $u_{i,j}$ be the approximate solution at that same point. The solution on the boundary points is given in **Equation 1** and using it our goal is to approximate the solution at the interior points.

Step 2: Discretize derivatives.

From the Taylor series expansion for two variables, we represent the second order partial derivatives at the grid point using the following derivation:

$$u(x_{i+1}, y_j) = u(x_i, y_j) + h_x \frac{\partial u}{\partial x} + \frac{(h_x)^2}{2!}\frac{\partial^2 u}{\partial x^2} + \frac{(h_x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \frac{(h_x)^4}{4!}\frac{\partial^4 u}{\partial x^4} + ...$$

$$u(x_{i-1}, y_j) = u(x_i, y_j) - h_x \frac{\partial u}{\partial x} + \frac{(h_x)^2}{2!}\frac{\partial^2 u}{\partial x^2} - \frac{(h_x)^3}{3!}\frac{\partial^3 u}{\partial x^3} + \frac{(h_x)^4}{4!}\frac{\partial^4 u}{\partial x^4} + ...$$

Summing the two equations, we get

$$u(x_{i+1}, y_j) + u(x_{i-1}, y_j) = 2u(x_i, y_j) + (h_x)^2\frac{\partial^2 u}{\partial x^2} + \frac{(h_x)^4}{12}\frac{\partial^4 u}{\partial x^4} + ...$$

Solving for $\frac{\partial^2 u}{\partial x^2}$, we get

$$\frac{\partial^2 u}{\partial x^2} = u_{xx} \approx \frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{(h_x)^2} - \frac{(h_x)^2}{12}\frac{\partial^4 u}{\partial x^4} + ... \qquad \text{Equation 7.}$$

Performing the same expansion $u(x_i, y_{j+1})$ and $u(x_i, y_{j-1})$, we can get the terms for $\frac{\partial^2 u}{\partial y^2}$:

$$\frac{\partial^2 u}{\partial y^2} = u_{yy} \approx \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{(h_y)^2} - \frac{(h_y)^2}{12}\frac{\partial^4 u}{\partial y^4} + ... \qquad \textbf{Equation 8.}$$

By adding **Equation 7** and **Equation 8**, the two-dimensional Poisson equation can be approximated as follows:

$$\begin{aligned} -(f(x_i, y_j) + e_{i,j}) = &\frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{(h_x)^2} \\ &+ \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{(h_y)^2} \\ &\text{where} \quad i = 1, 2, 3, ..., n-1 \quad \text{and} \quad j = 1, 2, 3, ..., m-1 \end{aligned} \qquad \textbf{Equation 9.}$$

Where $e_{i,j}$ includes the higher order terms of the Taylor series expansion. This is called the local truncation error, which is accrued during the approximation process when choosing to stop the approximation at a specific term. For our methods, we choose to derive to the fourth order approximation, with the fourth partial term representing the largest term in the rest of the infinite series, and the majority of our error. The error $e_{i,j}$ that we use is defined in **Equation 10**.

$$e_{i,j} \approx \frac{(h_x)^2 \partial^4 u}{12 \partial x^4}(x_i, y_i) + \frac{(h_y)^2 \partial^4 u}{12 \partial y^4}(x_i, y_i) + O(h^4), \text{ where } h = \max(h_x, h_y) \qquad \textbf{Equation 10.}$$

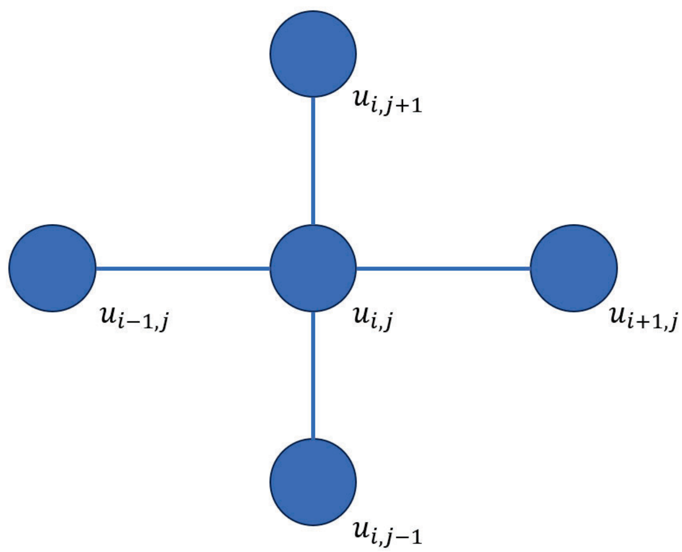

Figure 1. Five-point Finite Difference Stencil.

Since the local truncation error is small for small h values, it is reasonable to ignore $e_{i,j}$ for $h << 0.5$. For our five-point stencil, as can be seen in **Figure 1**, we can ignore the error term $e_{i,j}$ and replace the exact solution $u(x_i, y_j)$ at the grid points with the approximate solution $u_{i,j}$. Then we get the following discretized formula:

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{(h_x)^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{(h_y)^2} = -f(x_i, y_j) \qquad \textbf{Equation 11.}$$

Equation 11 can be modified as follows:

$$\frac{u_{i+1,j} + u_{i-1,j}}{(h_x)^2} + \frac{u_{i,j+1} + u_{i,j-1}}{(h_y)^2} - \left(\frac{2}{(h_x)^2} + \frac{2}{(h_y)^2}\right) u_{i,j} = -f(x_i, y_j) \qquad \textbf{Equation 12.}$$

Note that the finite difference equation at the grid point (x_i, y_j) involves 5 neighboring grid points (x_{i+1}, y_j), (x_{i-1}, y_j), (x_i, y_{j+1}), (x_i, y_{j-1}), and (x_i, y_j) in a five-point stencil. The five-point stencil, shown in **Figure 1**, is simply a finite difference method in which five adjacent points in a grid are used. Without loss of generality, we can assume that $h_x = h_y = h$, so **Equation 12** can be updated to be

$$u_{i+1,j} + u_{i-1,j} - 4u_{i,j} + u_{i,j+1} + u_{i,j-1} = -h^2 f(x_i, y_j). \qquad \textbf{Equation 13.}$$

Step 3: Build a system of linear equations.

From **Equation 13** we can build a system of linear equations that can be described by the following matrix system. In this case, we can express the discretized version of **Equation 1** as a matrix equation shown below:

$$-AU = F \qquad \textbf{Equation 14.}$$

A represents the stiffness matrix corresponding to the Δ operator and F represents the given source term corresponding to $f(x, y)$, and U represents the discrete solution of **Equation 1**. The new matrix-vector form of **Equation 14** yields the following:

$$A = \frac{1}{h^2}\begin{bmatrix} M & -1 & 0 \\ -1 & \ddots & -1 \\ 0 & -1 & M \end{bmatrix}_{m\times m} \quad \text{where } M = \begin{bmatrix} 4 & -1 & 0 \\ -1 & \ddots & -1 \\ 0 & -1 & 4 \end{bmatrix},$$
$$F = \begin{bmatrix} -f(x_1, y_1) \\ \vdots \\ -f(x_m, y_m \end{bmatrix}, \text{ and } U = \begin{bmatrix} u_{1,1} \\ \vdots \\ u_{i,j} \end{bmatrix}. \qquad \textbf{Equation 15.}$$

Error Estimate

This section discusses how well the numerical approximation determined by the five-point finite difference scheme approximates the exact solution of the two-dimensional Poisson equation for Dirichlet boundary conditions on a convex polygonal domain. In this paper, we use rectangular domains only in order to simplify the computation and make it easier to explain. Other polygonal domains can still be computed in this way, by simply performing a domain transformation form the non-rectangular domain into a domain similar to the one in this paper. More information can be found in.[13, 17] To this end we start with the following three definitions and **Theorem 2**.

Definition 1 (Consistency[2]). Let $\Delta_h(u) = -(u_{i-1,j} + u_{i,j-1} - 4u_{i,j} + u_{i,j+1} + u_{i+1,j})$ denote the finite difference approximation associated with the domain Ω_h having mesh size h to the second order partial differential operator $\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$. For a given function $u \in C^\infty(\Omega)$, the truncation error of $\Delta_h(u)$ is given by $e_h(x, y) = (\Delta - \Delta_h)u(x, y)$. The approximation $\Delta_h(u)$ is consistent with Δ if $\lim_{h\to 0} e_h(x, y) = 0$ for all $(x, y) \in \Omega$ and for all $u \in C^\infty(\Omega)$. Moreover, the approximation is consistent to order p if $e_h(x, y) = (h^p)$.

Definition 2 (Convergence[2]). Suppose $-\Delta_h \phi(x_j) = f(x_j)$ to be a finite difference approximation to the partial differential equation $-\Delta u(x) = f(x)$ defined on a grid with mesh size h for a simple connect domain $\Omega \subseteq \mathbb{R}^2$. Assume that $\phi(x, y) = u(x, y)$ at all points (x, y) on the boundary $\partial\Omega$. Then the finite difference scheme converges if

$$\max_j |u(x_j) - \phi(x_j)| \to 0 \text{ as } h \to 0.$$

Definition 3 (Maximum norm[7]). For any grid function $u : \Omega_h \cup \partial\Omega_h \to \mathbb{R}$, we define the maximum norms as follows:

$$\|u\|_{\infty,\Omega} = \max_{(x_i,y_j)\in\Omega_h} |u_{ij}|,$$
$$\|u\|_{\infty,\partial\Omega} = \max_{(x_i,y_j)\in\partial\Omega_h} |u_{ij}|.$$

In the context of the two-dimensional Poisson equation, the Discrete Maximum Principle is crucial for proving the uniqueness of the solution to the equation. It ensures that, for the discrete version of the problem, the solution behaves similarly to the continuous case where the maximum or minimum of the solution is achieved on the boundary. If the discrete solution u_h satisfies the Discrete Maximum Principle, it is guaranteed that the discrete solution converges to the true continuous solution as the grid resolution increases (in the case of a consistent and stable numerical method).

Theorem 2 (Discrete Maximum Principle[7]).

(i) *If* $\Delta_h u_{i,j} \geq 0$ *for all points* $(x_i, y_j) \in \Omega_h$ *then,* $\max_{(x_i,y_j)\in\Omega_h} u_{i,j} \leq \max_{(x_i,y_j)\in\partial\Omega_h} u_{i,j}$.

(ii) *If* $\Delta_h u_{i,j} \leq 0$ *for all points* $(x_i, y_j) \in \Omega_h$ *then,* $\min_{(x_i,y_j)\in\Omega_h} u_{i,j} \geq \min_{(x_i,y_j)\in\partial\Omega_h} u_{i,j}$.

Proof. We can prove *(i)* by contradiction. Assume that there exists some $u_{m,n} \in \Omega_h$, where $u_{m,n} = K > 0$, and $K > \max_{\partial\Omega} u_{i,j}$. Given a boundary value $g(x_i, y_j) \geq 0$, we have

$$f_{m-0.5,n}u_{m-1,n} + f_{m+0.5,n}u_{m+1,n} \geq [f_{m-0.5,n} + f_{m+0.5,n} + h^2 g_{m,n}]u_{m,n}.$$

Since h^2 is small, we remove it and the inequality holds. We also replace $u_{m,n}$ with K and get

$$f_{m-0.5,n}u_{m-1,n} + f_{m+0.5,n}u_{m+1,n} \geq [f_{m-0.5,n} + f_{m+0.5,n}]K.$$

Since $u_{m-1,n}$ and $u_{m+1,n}$ are smaller than K by assumption, we have

$$f_{m-0.5,n}u_{m-1,n} + f_{m+0.5,n}u_{m+1,n} \leq [f_{m-0.5,n} + f_{m+0.5,n}]K.$$

With the inequality holding unless $u_{m+1,n} = u_{m-1,n} = K$. If we repeat this argument for all interior points in both grid directions, we can conclude that all boundary points $(x_i, y_j) \in \partial\Omega_h$ are equal to K, which contradicts our assumption of a strict inequality, thus proving *(i)*. The exact same logic using the minimum proves *(ii)*.

□

Theorem 3. *The five-point finite difference analog* $\Delta_h(u) = -(u_{i-1,j} + u_{i,j-1} - 4u_{i,j} + u_{i,j+1} + u_{i+1,j})$ *is consistent to order 2 for the Laplace operator* $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$.[2, 7]

Proof. Let $u \in C^\infty(\Omega)$ and $(x, y) \in \Omega$ be a point such that $(x \pm h, y), (x, y \pm h) \in \Omega \cup \partial\Omega$. Then, by the Taylor Theorem we have

$$u(x+h,y) = u(x,y) + h\frac{\partial u}{\partial x}(x,y) + \frac{h^2}{2!}\frac{\partial^2 u}{\partial x^2}(x,y) + \frac{h^3}{3!}\frac{\partial^3 u}{\partial x^3}(x,y) + \frac{h^4}{4!}\frac{\partial^4 u}{\partial x^4}(\eta^\pm, y) \qquad \textbf{Equation 16.}$$

$$u(x-h,y) = u(x,y) - h\frac{\partial u}{\partial x}(x,y) + \frac{h^2}{2!}\frac{\partial^2 u}{\partial x^2}(x,y) - \frac{h^3}{3!}\frac{\partial^3 u}{\partial x^3}(x,y) + \frac{h^4}{4!}\frac{\partial^4 u}{\partial x^4}(\eta^\pm, y) \qquad \textbf{Equation 17.}$$

where $\eta^\pm \in (x-h, x+h)$. By adding **Equation 16** and **Equation 17**, we get

$$\frac{1}{h^2}\left[u(x+h,y) - 2u(x,y) + u(x-h,y)\right] - \frac{\partial^2 u}{\partial x^2}(x,y) = \frac{h^2}{4!}\left[\frac{\partial^4 u}{\partial x^4}(\eta^+, y) + \frac{\partial^4 u}{\partial x^4}(\eta^-, y)\right] \qquad \textbf{Equation 18.}$$

Recall that by the intermediate value theorem, we have

$$\left[\frac{\partial^4 u}{\partial x^4}(\eta^+, y) + \frac{\partial^4 u}{\partial x^4}(\eta^-, y)\right] = 2\frac{\partial^4 u}{\partial x^4}(\eta, y)$$

for some value $\eta \in (x-h, x+h)$. Suppose

$$\beta_x^2(x,y) = \frac{1}{h^2}\left[u(x+h,y) - 2u(x,y) + u(x-h,y)\right]$$

and thus

$$\beta_x^2(x,y) = \frac{\partial^2 u}{\partial x^2}(x,y) + \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\eta, y). \qquad \textbf{Equation 19.}$$

Similarly, we can obtain

$$\beta_y^2(x,y) = \frac{\partial^2 u}{\partial y^2}(x,y) + \frac{h^2}{12}\frac{\partial^4 u}{\partial y^4}(x, \mu). \qquad \textbf{Equation 20.}$$

for some $\mu \in (y-h, y+h)$. Finally, by adding **Equation 19** and **Equation 20**, we can conclude that $e_h(x,y) = (\Delta - \Delta_h)u(x,y) = O(h^2)$. Thus, the five-point finite difference scheme is consistent to order 2. □

Proposition 1. *If $\Delta_h u_{ij} = 0$ for $(x_i, y_j) \in \Omega_h$, and $u_{ij} = 0$ for $(x_i, y_j) \in \partial\Omega_h$, then the zero function $u_{ij} = 0$ for all $(x_i, y_j) \in \Omega_h \cup \partial\Omega_h$ is the only solution to the finite difference problem.*

Proposition 2. *If $\Delta_h u_{ij} = f_{ij}$ for $(x_i, y_j) \in \Omega_h$, and $u_{ij} = g_{ij}$ for $(x_i, y_j) \in \partial\Omega_h$, then there exists a unique solution to the finite difference problem for grid functions f_{ij} and g_{ij}.*

Lemma 1. *Suppose that the grid function $u : \Omega_h \cup \partial\Omega_h \to \mathbb{R}$ satisfies the boundary condition $u_{ij} = 0$ for $(x_i, y_j) \in \partial\Omega_h$, then we have the following estimate[7]:*

$$\|u\|_{\infty,\Omega} \le \frac{1}{8}\|\Delta_h u\|_{\infty,\Omega} \qquad \textbf{Equation 21.}$$

By leveraging **Proposition 2** and **Lemma 1**, we can establish the validity of our main theorem. This theorem asserts that the solution derived from the five-point finite difference scheme converges to the exact solution of **Equation 1**. Our proof for this convergence is outlined in the next section.

Theorem 4 ([2,7]). *Let u be a solution to the Poisson equation (**Equation 1**) and let $\hat{u}$ be the grid function that satisfies the discrete system as follows:*

$$\begin{aligned} -\Delta_h \hat{u}_{ij} &= f_{ij} \textit{ for } (x_i, y_j) \in \Omega_h, \\ \hat{u}_{ij} &= g_{ij} \textit{ for } (x_i, y_j) \in \partial\Omega_h. \end{aligned} \qquad \textbf{Equation 22.}$$

Then, there exists a positive constant C and a constant $K = \left\| \frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4} \right\|_{\infty,\Omega}$ such that

$$\|u - \hat{u}\|_{\infty,\Omega} \le CKh^2.$$

Proof. For smooth functions f and g the theorem assumes that $u \in C^4(\bar{\Omega})$ since the constant K involves 4$^{\text{th}}$ derivatives. By **Equation 19** and **Equation 20** we have the following estimate:

$$(\Delta_h - \Delta)u_{ij} = \frac{h^2}{12}\left[\frac{\partial^4 u}{\partial x^4}(\eta_i, y_j) + \frac{\partial^4 u}{\partial y^4}(x_i, \mu_j)\right] \qquad \textbf{Equation 23.}$$

for some $\eta_i \in (x_{i-1}, x_{i+1})$ and $\mu_j \in (y_{j-1}, y_{j+1}$. Since $-\Delta u_i j = f_{ij}$, we have

$$-\Delta_h u_{ij} = f_{ij} - \frac{h^2}{12}\left[\frac{\partial^4 u}{\partial x^4}(\eta_i, y_j) + \frac{\partial^4 u}{\partial y^4}(x_i, \mu_j)\right]. \qquad \textbf{Equation 24.}$$

By subtracting **Equation 22** from **Equation 24**, we get

$$\Delta_h(u_{ij} - \hat{u}_{ij}) = \frac{h^2}{12}\left[\frac{\partial^4 u}{\partial x^4}(\eta_i, y_j) + \frac{\partial^4 u}{\partial y^4}(x_i, \mu_j)\right]. \qquad \textbf{Equation 25.}$$

Note that $u - \hat{u}$ vanishes on $\partial\Omega_h$ due to the imposed boundary conditions of **Equation 1** and **Equation 22**. If we replace u by $u - \hat{u}$ in **Lemma 1**, we obtain the following:

$$\begin{aligned}\|u-\hat{u}\|_{\infty,\Omega} &\leq \frac{1}{8}\|\Delta_h(u-\hat{u})\|_{\infty,\Omega}\\ &\leq \frac{1}{8}\cdot\frac{h^2}{12}\left\|\frac{\partial^4 u}{\partial x^4}(\eta_i, y_j)+\frac{\partial^4 u}{\partial y^4}(x_i,\mu_j)\right\|\\ &\leq KCh^2\end{aligned}$$

The second equality is derived from **Equation 25** and the third is derived from the definition of *K*. The last equality concludes **Theorem 4**. □

RESULTS

In this section we present a few numerical experiments to illustrate the computational method discussed in this paper. The numerical experiments are performed on a laptop computer with MATLAB R2022a using a MacBook Air with M1 chip. We use the following formula for the order of convergence. We use the following formula for the order of convergence.

$$\text{Order of convergence } R = \frac{\log\left((u_{i+1}-\hat{u}_{i+1})/(u_i-\hat{u}_i)\right)}{\log\left(\frac{1}{2}\right)},$$

where u_{i+1} is the numerical solution in the state $(i+1)^{th}$ and $\hat{u}_{i+1}$ is the analytical solution in the state $(i+1)^{th}$.

Example 1. Consider the Poisson equation:

$$\begin{aligned}-\Delta u(x,y) &= -2\pi^2\sin(\pi x)\sin(\pi y) \text{ in } \Omega\\ u(x,y) &= 0 \text{ on } \partial\Omega,\end{aligned} \quad \textbf{Equation 26.}$$

where $\Omega = (0,2)\times(0,2)$ and the analytical solution $u(x,y) = \sin(\pi x)\sin(\pi y)$.

Test Case 1. In this example, we compare the numerical solution and the analytical solution with a sequence of different meshes obtained through mesh refinements. We record the maximum norm error between the analytical and numerical solutions in each mesh stage. **Table 1** shows that numerical solutions approximate the true solution better when we decrease the step size. This guarantees that our numerical solution converges to the true solution. **Figure 2** shows the analytical solution to **Equation 26**, and **Figure 3** shows a sequence of numerical solutions to **Equation 26** obtained through six consecutive mesh refinements.

Degrees of Freedom	Max Norm Error $\|u - \hat{u}\|_\infty$
4	0.999984501426793
16	0.177517468002680
64	0.033384181128561
256	0.007287288315757
1024	0.001695751741142
4096	3.989211285880812e-04
16384	8.648598269755947e-05

Table 1. Maximum norm errors for consecutive mesh refinements.

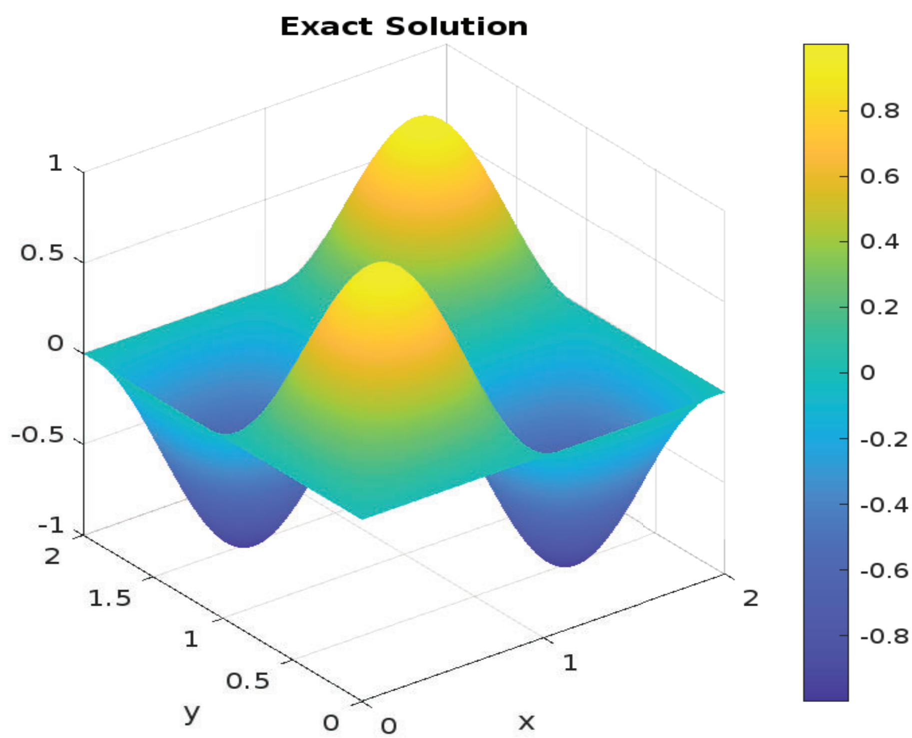

Figure 2. Exact Solution.

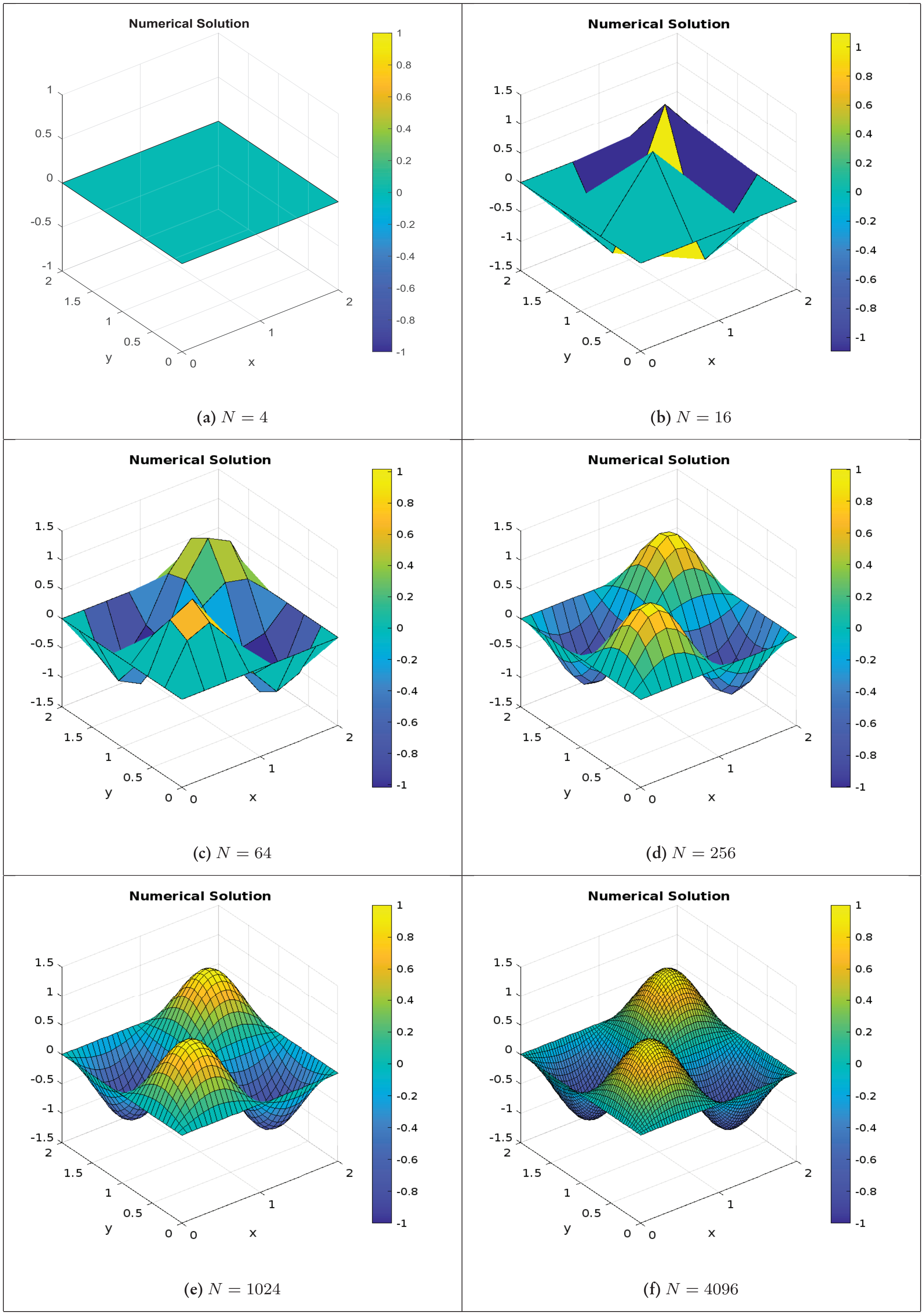

Figure 3. Numerical Solution for Six Consecutive Meshes.

Test Case 2. Consider the Poisson equation (**Equation 26**) on a domain, $\Omega = (0,1) \times (0,1)$. For this test case, we record the error between numerical solution and analytical solution for each node on the same mesh. The solution values and errors are displayed in **Table 2**. The error is small even with the initial stage of the mesh refinement. **Figure 4** shows the analytical solution and the numerical solution on a mesh with degrees of freedom $N = 25$.

Node (x, y)	Analytical Solution	Numerical Solution	Error
$(0.25, 0.25)$	0.500000000000000	0.526514643772757	0.026514643772757
$(0.5, 0.25)$	0.707106781186547	0.744604150011472	0.037497368824925
$(0.75, 0.25)$	0.500000000000000	0.526514643772757	0.026514643772757
$(0.25, 0.5)$	0.707106781186547	0.744604150011472	0.037497368824925
$(0.5, 0.5)$	1.000000000000000	1.053029287545515	0.053029287545515
$(0.75, 0.5)$	0.707106781186547	0.744604150011472	0.037497368824925
$(0.25, 0.75)$	0.500000000000000	0.526514643772757	0.026514643772757
$(0.5, 0.75)$	0.707106781186547	0.744604150011472	0.037497368824925
$(0.75, 0.75)$	0.500000000000000	0.526514643772757	0.026514643772757

Table 2. Comparison of Analytical and Numerical Solutions on the same Mesh Refinement.

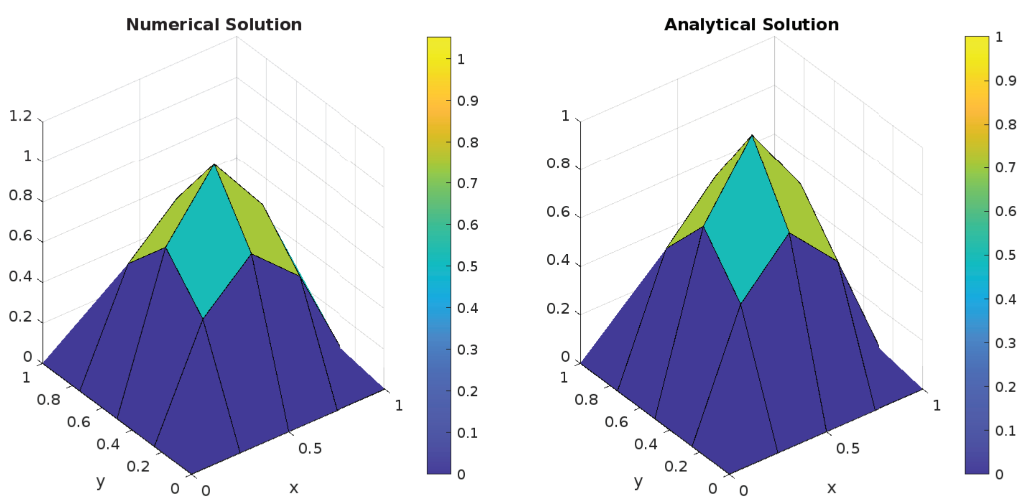

Figure 4. Comparison of Numerical and Analytical Solutions at N = 25.

Example 2. In **Example 2**, we report convergence orders of the numerical solution of **Equation 26** obtained through a sequence of mesh refinements. **Table 3** shows that the numerical convergence order $R \approx 2$, which is in strong agreement with the convergent order shown in **Theorem 3**.

DoF	4	16	64	256	1024	4096	16384
R	-	-	2.4107	2.1957	2.1035	2.0877	2.2056

Table 3. Convergence Orders R for a Sequence of Meshes.

Example 3. In this example, we report the CPU time required by the five-point stencil finite difference method to solve **Equation 26** with $f = 1$ on a square domain $\Omega = (0, 1) \times (0, 1)$. The results are presented in seconds for varying degrees of freedom (DoF) for each mesh in **Table 4**. From **Table 4**, we can observe that the finite difference methods converge to the true solution quickly.

DoF	4	16	64	256	1024	4096	16384
$\mathbf{t_{CPU}}$	0.00280	0.00591	0.00943	0.01425	0.03049	0.86497	44.80878

Table 4. CPU Time in Seconds.

CONCLUSIONS

This study presents a complete analysis of the two-dimensional Poisson equation. We present the derivation of a five-point finite difference scheme and prove its uniqueness and consistency. We also present an infinity norm error estimate and validate those results through numerical simulations. The numerical results presented in the tables and graphs show that the present method approximates the exact solution efficiently with a quadratic convergence, validating our theoretical assumption.

This problem can be extended in several directions. Solving Poisson's equation on a three-dimensional nonconvex polygonal domains with a singularly perturbed parameter added to the Poisson's equation will be an interesting problem which requires more advanced numerical methods such as finite element methods to discretize the Poisson's equation. Another challenging task is to use the point delta function or the line delta function as the source term for the three-dimensional Poisson's equation. Lastly, this can be extended to higher-order PDEs, which have many real-world applications.

The MATLAB code can be accessible via the following link: `https://github.com/mrwhalen22/2D\_poisson\_finite\_difference.git`

ACKNOWLEDGEMENTS

The author thanks their research mentor, Dr. Charuka D. Wickramasinghe, from the University of Kentucky for their guidance on this article during Winter 2023.

REFERENCES

1. Evans, L. C. (2022) Partial differential equations. Vol. 19. *American Mathematical Society*.
2. Lapidus, L. and Pinder G. F. (1999) Numerical solution of partial differential equations in science and engineering. *John Wiley & Sons*.
3. Özişik, M., Necati, Helcio RB Orlande, Marcelo J. Colaço, and Renato M. Cotta. (2017) Finite difference methods in heat transfer. *CRC press*.
4. Phillips, G. M. and Taylor, P. J. eds. (1996) Theory and applications of numerical analysis. *Elsevier*.
5. Wickramasinghe, C. D., & Ahire, P. (2024). A Graded Mesh Refinement For 2d Poisson's Equation on Non-Convex Polygonal Domains. *Asia Pac. J. Math. 2024 11:63.* `https://doi.org/10.28924/APJM/11-63`.

6. Wickramasinghe, Charuka. 2025. “A Numerical Scheme to Solve Fourth Order Convection Reaction Diffusion Problems With Boundary Layers”. Communications in Mathematics and Applications 15 (3). https://www.rgnpublications.com/journals/index.php/cma/article/view/2778.
7. Li, Zhilin, Zhonghua Qiao, and Tao Tang. (2017) Numerical solution of differential equations: introduction to finite difference and finite element methods. *Cambridge University Press*.
8. Yoon, G. J. and Min, C. (2015) Analyses on the finite difference method by Gibou et al. for Poisson equation, J. Comput. Phys. **280**, 184–194. MR3273131.
9. Kim, Nam-Ho, Bhavani V. Sankar, and Ashok V. Kumar. (2018) Introduction to finite element analysis and design. *John Wiley & Sons*.
10. Dilhara Wickramasinghe, C. (2021) "An Efficient Numerical Approximation to Poisson Problem in Two Dimensions."
11. Mazumder, S. (2016) *Numerical methods for partial differential equations*, *Elsevier/Academic Press*, Amsterdam, MR3443468.
12. Moukalled, Fadl, Mangani, L., Darwish, M., Moukalled, F., Mangani, L., and Darwish, M. (2016) The finite volume method. *Springer International Publishing*.
13. Matte, J. -P. and Lafrance, G. (1977) Solution of the discrete Poisson equation with complicated boundaries, *J. Comput. Phys*. **23**, no. 1, 86–91. MR0502093
14. Brandstetter, G. and Govindjee, S. (2015) A high-order immersed boundary discontinuous-Galerkin method for Poisson’s equation with discontinuous coefficients and singular sources, *Internat. J. Numer. Methods Engrg*. **101**, no. 11, 847–869. MR3311660.
15. Chen, L. and Li, H. (2010) Superconvergence of gradient recovery schemes on graded meshes for corner singularities, *J. Comput. Math*. **28** (2010), no. 1, 11–31. MR2603578
16. Feng, Q., Han, B., and Minev, P. D. (2021) Sixth order compact finite difference schemes for Poisson interface problems with singular sources, *Comput. Math. Appl*. **99** (2021), 2–25. MR4302074
17. Li, H., Wickramasinghe, C. D., and Peimeng Yin. (2022) "A C^0 finite element method for the biharmonic problem with Dirichlet boundary conditions in a polygonal domain." *arXiv* preprint arXiv:2207.03838.
18. Musolino, P. (2013) A singularly perturbed Dirichlet problem for the Poisson equation in a periodically perforated domain. A functional analytic approach, in *Advances in harmonic analysis and operator theory*, 269–289, Oper. Theory Adv. Appl., 229, Birkhäuser/Springer Basel AG, Basel. MR3060419
19. Wickramasinghe, Charuka Dilhara, "A C^0 Finite Element Method For The Biharmonic Problem In A Polygonal Domain" (2022). Wayne State University Dissertations. 3704. https://digitalcommons.wayne.edu/oa_dissertations/3704.
20. Butler, J. (2021) Numerical Methods for Differential Equations with Python, *https://github.com/john-s-butler-dit/Numerical-Analysis-Python*.
21. Gregory, D. T. and Wickramasinghe, C.D. (2023) An Upwind Finite Difference Method to Singularly Perturbed Convection Diffusion Problems on a Shishkin Mesh. *arXiv* preprint arXiv:2306.03181.
22. Kadalbajoo, M. K. and Patidar, K. C. (2003) "Singularly perturbed problems in partial differential equations: a survey." *Applied Mathematics and Computation* 134, no. 2-3: 371-429.
23. Barton, G. (1989) Elements of Green’s functions and propagation: potentials, diffusion, and waves. *Oxford University Press*.
24. Brown, J. and Churchill, R. (2011) Fourier Series and Boundary Value Problems, 8th Edition (*McGraw-Hill Companies*, Inc., New York, NY, 2011).
25. Hössjer, E.(2012) "Uniqueness of Solutions to the Poisson Equation." (2012).
26. Alan, J. (2003) Applied partial differential equations: an introduction. *Academic Press*, 2003.
27. Causon, D. M. and Mingham, C. G. (2010) Introductory finite difference methods for PDEs. *Bookboon*.

ABOUT THE STUDENT AUTHOR

Matthew Whalen completed this work during the summer of 2023, prior to graduating with a bachelor's of Computer Science at the University of Kentucky in May 2024. He is now planning to graduate from the University of Kentucky again in 2025 with a Master's of Business Administration and Management.

PRESS SUMMARY

This paper describes a numerical procedure for simulating solutions to Dirichlet Poisson equations using MATLAB. In addition, it can serve as a uniform summary of related literature needed for such a procedure such that anyone can read and use finite difference schemes. This method of approximating differential equations is valuable in numerous fields of engineering and physics. The main goal of this procedure is to make these equations automatable and solvable using computer technology for applications like fluid and heat simulations. Hopefully, this work can continue to be expanded and adapted to non-Dirichlet boundary condition problems.