Proceedings of Science and Mathematics
Faculty of Science, Universiti Teknologi Malaysia
Vol. 9, 2022, page 64-72

Application of Finite Element Method in 2D Poisson Equation

Farah Hanis Binti Baharum*, Yeak Su Hoe
Department of Mathematical Sciences, Faculty of Science
Universiti Teknologi Malaysia, 81310 Johor Bahru, Malaysia
Corresponding author: *thefarahh@gmail.com, s.h.yeak@utm.my

Abstract
The purpose of this study is to solve numerically the 2D Poisson equations in heat problems using the finite element method (FEM). 2D Poisson equation elements are defined by three or more nodes in a 2D plane. 2D Poisson equations are simple but the boundary condition may be complicated. The 2D problems can be solved using numerical methods namely FEM. FEM is a numerical method that is commonly used to solve engineering and mathematical problems. FEM is a high-accuracy approximation that can be used to solve engineering problems. The accuracy error of the solution for 2D Poisson equations is calculated for both methods FEM and FDM. The algorithm of FEM and FDM is developed in MATLAB and Python programs where the result is validated by comparing the surface integration error. The surface integration error for application of FEM and FDM is very small. The loss of precision because of truncation error caused by the computer rounding off decimal quantities. For future work, researcher should solve on Neumann boundary conditions since this project report only focus on Dirichlet boundary conditions.
Keywords: 2D Poisson Equations; Finite Element Method (FEM); Finite Difference Method (FDM); Heat transfer; MATLAB; Python

1. Introduction

Heat is a form of energy that cannot be replicated or destroyed. Heat conduction, heat convection, and heat radiation are the three types of heat transfer processes. Heat transfer is used in many applications, including thermal and nuclear power plant design. It can also be transferred from one system to another, for instance from a higher to lower temperature.

The purpose of reviewing the outcomes is to calculate and investigate the error of the approximate value when compared to the exact values. Numerical analysis is the division of mathematics and computer science that develops, analyses, and implements algorithms for numerically solving continuous mathematics problems.

The finite element method (FEM) is a commonly used numerical method to solve engineering and mathematical physics problems. Discretization methods use numerical model equations to approximate the differential equations. FEM establishes credible stability and provides more flexibility, such as handling inhomogeneity and complex geometries.

This study focuses on solving 2D problems using the FEM. FEM two-dimensional (2D) elements are defined by three or more nodes in a 2D plane. Due to the large scale of the problem, all computations for this numerical method will be performed using Python software. To solve the Poisson equation in complex geometry, a numerical solution is needed. Analytical solutions only exist for simple geometry and do not exist for complex geometry. Python and MATLAB code will be developed for computations to assure the accuracy of the solutions for the 2D problem.

The extended surface will help to keep the system from overheating when using natural or forced convection. Thermal conductivity and heat transfer coefficient, for example, are critical engineering parameters. It is also significant to keep an eye on the coefficient that's best for engineering problems.

When using FDM to solve mathematical problems, there may be some fault in the accuracy of the solution. The primary source of error in this method is the finite difference method's truncation error. This method is suitable for simple geometries but hard to write in the program.

FEM is a high-accuracy approximation to its solution that produces fixed solutions. Though it is problem-dependent, the FEM approximation is higher than the corresponding FDM approach. The method used in obtaining the solution requires an understanding of 2D geometries.

1. Literature Review

1.1. Finite Element Method

FEM is a widely used numerical method for solving engineering and mathematical physics problems. The Nernst-Planck Equation (N-P) and the Laplace equation are the two most used modeling approaches. Pitting, crevice corrosion, galvanic corrosion, atmospheric localized corrosion, and corrosion test design are all FEM examples.

Formula for approximating solution of differential equations (FEM) has been developed by Olaiju et al [1]. FEM involves discretization form formulation, effective solution of finite element equations and physical phenomenon's partial differential equations. The integral version is known as the weak form.

A bidirectional data exchange is realized after the data transfer interface between the aerostatic bearing's FEM model and the thrust plate. According to Gao Q. et al. (2021), the pressure distribution of the air film is analysed using two different models [2].

1.2. Finite Difference Method

The Taylor series expansion around a point can be applied for any sufficiently differentiable function in each domain. According to Liszka. T and Orkisz J. (1980) the expansion can be used for uneven grids, as shown in the figure below [3].

$$f = f_0 + h\frac{\partial f_0}{\partial x} + k\frac{\partial f_0}{dy} + \frac{h^2}{2}\frac{\partial^2 f_0}{\partial x^2} + \frac{k^2}{2}\frac{\partial^2 f_0}{\partial y^2} + kh\frac{\partial^2 f_0}{\partial x \partial y} + O\left(\Delta^3\right) \tag{1}$$

Where $f\left(x,y\right),\ f_0 = f\left(x_0,y_0\right),\ h = x - x_0,\ k = y - y_0,\ \Delta = \sqrt{h^2+k^2}$ writing equation (1) for each of the nodes in the mesh, derive the set of linear equations $(m \geq 5)$ as equation (2) below

$$\left[A\right]\left\{Df\right\} - \left\{f\right\} = \left\{0\right\} \tag{2}$$

with matrix $\left[A\right]$ and unknown function as equation (3) and equation (4) below

$$\left[A\right] = \begin{bmatrix} h_1 & k_1 & h_1^2/2 & k_1^2/2 & h_1.k_2 \\ h_2 & \cdots & \cdots & \cdots & \cdots \\ \vdots & & & & \\ \vdots & & & & \\ h_m & & & & \end{bmatrix} \tag{3}$$

$$\left\{f\right\}^T = \left\{f_1 - f_{0,} f_2 - f_0, \ldots, f_m - f_0\right\} \tag{4}$$

where the five unknown derivatives at the point (x_0, y_0) are $\{Df\}^T = \left\{ \frac{\partial f_0}{\partial y}, \frac{\partial f_0}{\partial y}, \frac{\partial^2 f_0}{\partial x^2}, \frac{\partial^2 f_0}{\partial y^2}, \frac{\partial^2 f_0}{\partial x \partial y} \right\}$

Finite difference method (FDM) requires significantly higher mesh densities to achieve a given accuracy level. Zaghloul N. A. (1981) stated that the FDM which has irregularly shaped boundaries difficult to handle [4]. Then, the size of the elements can be varied which the property allows the element grid to be expanded or redefined.

Difficulty may occur when combining energy formulation and finite differences of arbitrary meshes. FDM method is simple and flexible in providing different numerical solutions of the differential equations. Also, FDM does not need a large computer memory and simple problems can be solved using hand calculators.

2. Methodology

2.1. Mathematical Model

The two-dimensional heat equation is needed to approximate the temperature distribution in heat conduction. Heat transmission through conduction, as opposed to convection and radiation, is widely recognized. Galerkin's approach for second order differential equation as equation (5)

$$a(x)y'' + b(x)y' + c(x)y = f(x), y(x_1) = \bar{y}'_N \tag{5}$$

$$\int_{x_1}^{x_N} \phi\big(a(x)y'' + b(x)y' + c(x)\big)dx = \int_x^x \phi f(x)dx \tag{6}$$

Using integration by part, we get $\int_{x=a}^{x=b} u dv = uv\Big|_{x=a}^{x=b} - \int_{x=a}^{x=b} v du$

$$\int_{x_1}^{x_N} \phi a(x) y'' dx = \phi a y' \Big|_{x_1}^{x_N} - \int_{x_1}^{x_N} [\phi' a + \phi a'] y' dx \tag{7}$$

$$\phi a y' \Big|_{x_1}^{x_N} + \int_{x_1}^{x_N} -[a\phi' y' + a'\phi y'] + \phi b y' + \phi c y dx = \int_x^x \phi f(x) dx \tag{8}$$

$$\psi^T \mathbf{N}^T a y' \Big|_{x_1}^{x_N} + \int_{x_1}^{x_N} -[a\mathbf{N}'\psi\mathbf{N}'\mathbf{Y} + a'\mathbf{N}\psi\mathbf{N}'\mathbf{Y}] + b\mathbf{N}\psi\mathbf{N}'\mathbf{Y} + c\mathbf{N}\psi\mathbf{N}\mathbf{Y} dx = \int_x^x \mathbf{N}\psi f(x) dx \tag{9}$$

$$\psi^T \mathbf{N}^T a y' \Big|_{x_1}^{x_N} + \int_{x_1}^{x_N} -a(\mathbf{N}'\psi)^T \mathbf{N}'\mathbf{Y} - a'(\mathbf{N}\psi)^T \mathbf{N}'Y + b(\mathbf{N}\psi)^T \mathbf{N}'\mathbf{Y} + c(\mathbf{N}\psi)^T \mathbf{N}\mathbf{Y}$$

$$= \int_x^x (\mathbf{N}\psi)^T f(x) dx \tag{10}$$

$$\psi^T \mathbf{N}^T a y' \Big|_{x_1}^{x_N} + \psi^T \int_{x_1}^{x_N} -a\mathbf{N}'^T \mathbf{N}'\mathbf{Y} - a'\mathbf{N}^T\mathbf{N}'\mathbf{Y} + b\mathbf{N}^T\mathbf{N}'\mathbf{Y} + c\mathbf{N}^T\mathbf{N}\mathbf{Y} dx = \psi^T \int_x^x \mathbf{N}^T f(x) dx \tag{11}$$

$$\psi^T \left[\int_{x_1}^{x_N} a\mathbf{N}'^T \mathbf{N}' + a'\mathbf{N}^T\mathbf{N}' - b\mathbf{N}^T\mathbf{N}' - c\mathbf{N}^T\mathbf{N} dx \right] Y = \psi^T \mathbf{N}^T a y' \Big|_{x_1}^{x_N} - \psi^T \int_x^x \mathbf{N}^T f(x) dx \tag{12}$$

2.1.1. Boundary Conditions

Consider that 2D heat conduction equation (13) where T is temperature and Q represent a heat source. The boundary conditions as below:

$$\frac{\partial}{\partial x}\left(\frac{k\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\frac{k\partial T}{\partial y}\right) + Q = 0 \tag{13}$$

where q_x is the components of heat flux index x, q_y is the components of heat flux index y, T is the temperature, Q is the heat source and k is a constant.

$$T = T_0 \text{ on } S_T \tag{14}$$

$$q_n = q_0 \text{ on } S_q \tag{15}$$

$$q_n = h(T - T_\infty) \text{ on } S_c \tag{16}$$

2.1.2. Triangular element

Temperature field within an element is

$$T = N_1T_1 + N_2T_2 + N_3T_3 \tag{17}$$

or

$$T = N\boldsymbol{T}^e \tag{18}$$

where $N = [\xi, \eta, 1-\xi-\eta]$ are the element shape function

$$\boldsymbol{T}^e = [T_1, T_2, T_3]^T \tag{19}$$

$$T = \xi T_1 + \eta T_2 + (1-\xi-\eta)T_3 \tag{20}$$

$$x = N_1x_1 + N_2x_2 + N_3x_3 = \xi x_1 + \eta x_2 + (1-\xi-\eta)x_3 \tag{21}$$

$$y = N_1y_1 + N_2y_2 + N_3y_3 = \xi y_1 + \eta y_2 + (1-\xi-\eta)y_3 \tag{22}$$

By using chain rule,

$$\frac{\partial T}{\partial \xi} = \frac{\partial T}{\partial x}\cdot\frac{\partial x}{\partial \xi} + \frac{\partial T}{\partial y}\cdot\frac{\partial y}{\partial \xi} \tag{23}$$

$$\frac{\partial T}{\partial \eta} = \frac{\partial T}{\partial x}\cdot\frac{\partial x}{\partial \eta} + \frac{\partial T}{\partial y}\cdot\frac{\partial y}{\partial \eta} \tag{24}$$

or

$$\begin{bmatrix} T_\xi \\ T_\eta \end{bmatrix} = \begin{bmatrix} x_\xi & y_\xi \\ x_\eta & y_\eta \end{bmatrix} \cdot \begin{bmatrix} T_x \\ T_y \end{bmatrix} = \begin{bmatrix} x_{13} & y_{13} \\ x_{23} & y_{23} \end{bmatrix} \cdot \begin{bmatrix} T_x \\ T_y \end{bmatrix} = \mathbf{J}\begin{bmatrix} T_x \\ T_y \end{bmatrix} \tag{25}$$

where

$$x_{ij} = x_i - x_j, y_{ij} = x_{ij} = y_i - y_j \tag{26}$$

By using the inverse matrix 2×2 becomes

$$\begin{bmatrix} T_x \\ T_y \end{bmatrix} = J^{-1}\begin{bmatrix} T_\xi \\ T_\eta \end{bmatrix} = \frac{1}{\det \mathbf{J}}\begin{bmatrix} y_{23} & -y_{13} \\ -x_{23} & x_{13} \end{bmatrix} \bullet \begin{bmatrix} T_1 - T_3 \\ T_2 - T_3 \end{bmatrix}$$

$$= \frac{1}{\det \mathbf{J}}\begin{bmatrix} y_{23} & -y_{13} \\ -x_{23} & x_{13} \end{bmatrix} \bullet \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \end{bmatrix}\mathbf{T}^e \tag{27}$$

which can be written as

$$\nabla T = \nabla(\mathbf{N}T^e) = (\nabla\mathbf{N})\mathbf{T}^e = \mathbf{B}_T T^e \tag{28}$$

$$\mathbf{B}_T = \nabla \mathbf{N} = \nabla \begin{bmatrix} N_1 & N_2 & N_3 \end{bmatrix} = \begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial x} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial y} & \frac{\partial N_3}{\partial y} \end{bmatrix} \quad (29)$$

$$\begin{bmatrix} \frac{\partial T}{\partial x} \\ \frac{\partial T}{\partial y} \end{bmatrix} = \mathbf{B}_T \mathbf{T}^e \quad (30)$$

where

$$\mathbf{B}_T = \frac{1}{\det \mathbf{J}} \begin{bmatrix} y_{23} & -y_{13} & (y_{23} - y_{13}) \\ -x_{23} & x_{13} & (x_{23} - x_{13}) \end{bmatrix}$$
$$= \frac{1}{\det \mathbf{J}} \begin{bmatrix} y_{23} & y_{31} & y_{12} \\ x_{23} & x_{13} & x_{21} \end{bmatrix} \quad (31)$$

2.2. Finite Difference Method (FDM)

Finite difference method (FDM) aims to approximate the values of the continuous function f (x, y) on a set of discrete points in (x, y) plane. The central difference method is used to discretize the governing equation.

Consider the Poisson equation,

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -f(x, y) \quad (32)$$

where u is and f (x, y) is considered as Q which is represent an equation for heat source as stated in equation

$$\frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{\Delta x^2} + \frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{\Delta y^2} = -f_i^j \quad (33)$$

Then, multiply both side of equation by gives

$$\left(u_{i+1}^j - 2u_i^j + u_{i-1}^j\right) + \frac{\Delta x^2}{\Delta y^2}\left(u_i^{j+1} - 2u_i^j + u_i^{j-1}\right) = -\Delta x^2 f_i^j \quad (34)$$

Let $r = \Delta x^2 / \Delta y^2$ then the equation becomes

$$ru_i^{j-1} + u_{i-1}^j - \left(2 + 2r\right)u_i^j + u_{i+1}^j + ru_i^{j+1} = -\Delta x^2 f_i^j \quad (35)$$

3. Results and discussion

3.1. Numerical Result for 2D Poisson Equation Problem

Given steady state of heat equation

$$U_{xx} + U_{yy} = -f(x, y) \quad (36)$$

The finite element method (FEM) was used to solve Problem 1, and Table 1 tabulates the integration of error surface, norm error, and precise answer. Figure 1 illustrated the difference between the surface of integration and the norm in various colours. There are 2 elements, 4 elements, 8 elements, 12 elements, and 16 elements.

Consider steady state heat equation (36) with $f(x, y) = 2x(1-x) + 2y(1-y)$ for Problem 1:

The analytical solution for Problem 1 as equation (37):

$$U(x,y) = x(1-x)y(1-y) \quad (37)$$

Table 1: Convergence study error for Problem 1

Mesh	Integration of error surface	Norm error
2×2	0.003906250000000007	0.015625000000000028
4×4	0.001332310267857165	0.007200174605486422
8×8	0.0003576416741075891	0.0034819932048637072
12×12	0.0001610284426987297	0.002305379863406472
16×16	9.099002983606229e-05	0.0017247706861778687

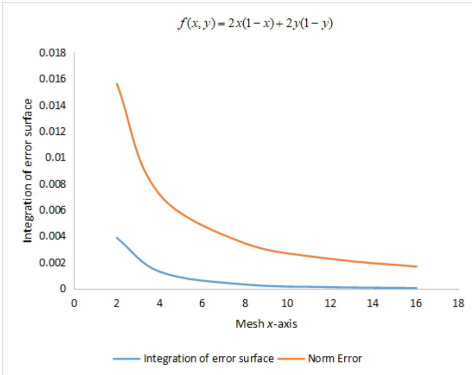

Figure 1 Integration of error surface and norm error for Problem 1 using FEM

The findings in Table 1 suggest that the value integration of error surface and norm error are very small. These data demonstrate the precision of the solution based on error. As the number of elements increases, the error decreases until it reaches zero. The error is converged to the integration of error surface in Figure 1. Because the value of error converges to zero, FEM was appropriate for addressing Problem 1.

Consider steady state of heat equation (4.13) with $f(x,y) = -e^{-x} - 8e^{-2y}$ for Problem 2:

The analytical solution for Problem 2 as equation (38):

$$U(x,y) = e^{-x} + 2e^{-2y} \quad (38)$$

The finite element method (FEM) in Python was used to solve Problem 2, and Figure 2 illustrated the differences between the integration of error surface and norm error. According to Figure 2, the integration of error surface and norm error result varies from four to eight elements. This is due to a truncation error. Precision loss because of truncation error induced by the computer rounding off decimal figures.

Table 2: Convergence study error of Problem 2

Mesh	Integration of error surface	Norm error
2×2	1.7781795719207238e-07	7.112718287682895e-07
4×4	3.979098890660726e-09	2.1859495710634103e-08
8×8	6.770337762640466e-11	6.680711563994133e-10
12×12	6.035159255062755e-12	8.747892204279621e-11
16×16	1.0776397135758842e-12	2.0674363002134307e-11

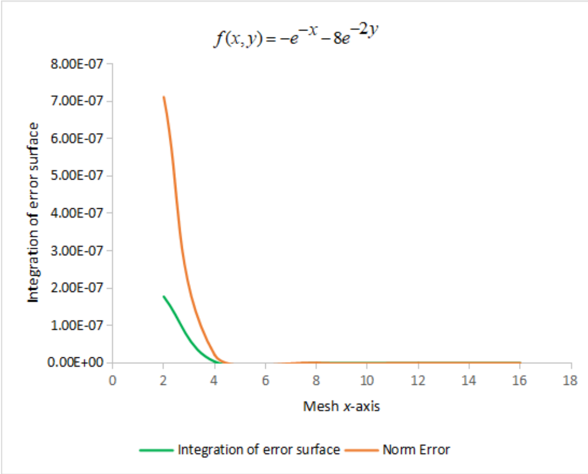

Figure 2 Integration of error surface and norm error for Problem 2 using FEM

The larger the mesh size, the lesser the error in Problem 2. According to Table 2, the integration of error surface and norm error value decrease as the number of nodes increases. This demonstrates that Problem 2 is converged. Figure 2 shows that the error has converged to the surface of the integration error.

3.2. Solution using Finite Difference Method

The finite difference method (FDM) was used for 2D Poisson equations using MATLAB and the value integration of error surface Problem 1 in Table 3. According to Table 3, the value integration of error surface varies with the number of elements. While 2 and 4 elements converged to error, the value of error increased for 8 elements. There might be a fault in the precision of the FDM solution. The value of error is thus the same for 12 and 16 elements may be the solution has reached the equilibrium point or has truncation error.

Table 3: Integration of error surface for Problem 1

Mesh	Integration of error surface
2×2	0.0156
4×4	0.0117
8×8	0.0122
12×12	0.0124
16×16	0.0124

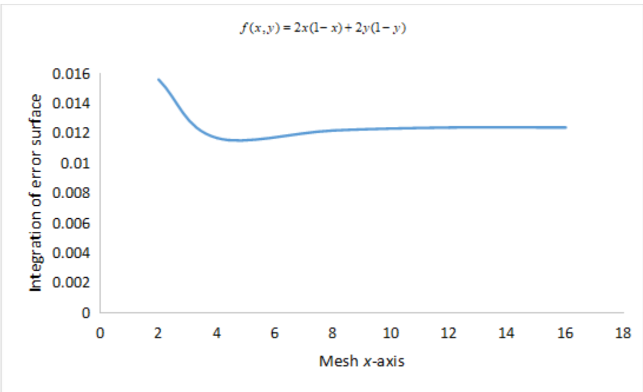

Figure 3 Integration of error surface for Problem 1 using FDM

According to Figure 3, at 12 and 16 elements maybe has the equilibrium points is 0.0124. At 8 elements, the inaccuracy is increasing, and at 12 elements, it has reached an equilibrium point. Problem 2 does not converge to zero maybe it has reached the equilibrium points or truncation error. The FDM approach is a simple numerical methodology, however, it is less stable and inaccurate, resulting in inconsistent solutions.

The integration of error surface for Problem 2 is converged, as shown in Figure 4. The inaccuracy decreases as the number of elements increases. As the number of elements increases, the value integration of error surface decreases.

Table 4: Integration of error surface for Problem 2

Mesh	Integration of error surface
2×2	0.0042
4×4	0.0020
8×8	5.9207e-04
12×12	2.7132e-04
16×16	1.5427e-04

According to Figure 4, as the number of elements increased, the error gradually converged to zero. The value of error at 16 elements is, which is quite tiny. It demonstrates that FDM can resolve Problem 2.

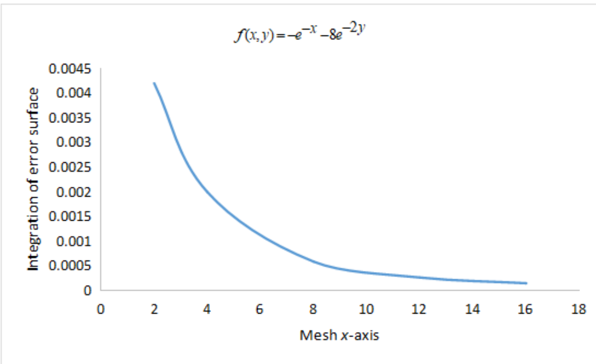

Figure 4 Integration of error surface for Problem 2 using FDM

3.3. Comparison integration of error surface between FEM and FDM

The integration of error surface is compared between FEM and FDM in this section. The absolute error for both methods is determined, as shown in Tables 5 and 6. To observe the pattern of error for both methods, the computation is performed for different nodes: 2 elements, 4 elements, 8 elements, 12 elements, and 16 elements.

Table 5: Comparison between FEM and FDM for integration of error surface in Problem 1

	Integration of error surface	
Mesh	**Finite Element Method (FEM)**	**Finite Difference Method (FDM)**
2×2	0.003906250000000007	0.0156
4×4	0.001332310267857165	0.0117
8×8	0.0003576416741075891	0.0122
12×12	0.0001610284426987297	0.0124
16×16	9.099002983606229e-05	0.0124

Based on Table 5, the integration of error surface for FEM is smaller than FDM. FEM is a high-accuracy approximation to its solution that provides fixed points. Thus, the error of FEM is smaller since the solution is nearest to the exact solution.

Table 6: Comparison between FEM and FDM for integration of error surface in Problem 2.

Integration of error surface		
Mesh	**Finite Element Method (FEM)**	**Finite Difference Method (FDM)**
2×2	1.7781795719207238e-07	0.0042
4×4	3.979098890660726e-09	0.0020
8×8	6.770337762640466e-11	5.9207e-04
12×12	6.035159255062755e-12	2.7132e-04
16×16	1.0776397135758842e-12	1.5427e-04

From Table 6, the integration of error surface for FEM is very small compared to FDM. As for 2 elements, error for FEM is 1.7781795719207238e-07 whereas error of FDM is 0.0042. The difference between FEM and FDM error is bigger. Since 2D Poisson equations have a complex boundary, then FEM is a better method than FDM. Since FDM is less accurate and less stable than FEM.

Conclusion

In this research, 2D Poisson equations were employed as a mathematical model to determine the integration of error surface using finite element methods (FEM) and finite difference techniques (FDM). The numerical solution of 2D Poisson equation problems was explored using FEM and FDM with Dirichlet boundary conditions. The issues are then calculated in Python for FEM and MATLAB for FDM to validate the accuracy of the solution.

In general, FDM is easier to be implemented and coded than FEM, although some issues involving complex geometry are either difficult or impossible to solve with FDM. Even though FEM is difficult to be constructed and computed in Python, it is a strong tool for solving complicated geometry issues and is free software. The solution for 2D Poisson equations in Problems 1, 2, and 3 using integration of error surface shown that the error of FEM was less than FDM, which is more accurate.

This study shows that solving 2D Poisson equations with either FEM or FDM is dependent on boundary conditions. The smaller the value integration of error surface as the number of elements increases, the lower the value norm error. Dirichlet boundary conditions were employed for these problems, but answers may be different if Neumann or Robin boundary conditions are applied.

Acknowledgement

The researcher would like to thank all people who have supported to finish this paper.

References

[1] Olaiju, O. A., Hoe, Y. S., & Ogunbode, E. B. (2018). Finite element and finite difference numerical simulation comparison for air pollution emission control to attain cleaner environment. Chemical Engineering Transactions, 63, 679–684. https://doi.org/10.3303/CET1863114

[2] Gao, Q., Qi, L., Gao, S., Lu, L., Song, L., & Zhang, F. (2021). A FEM based modeling method for analyzing the static performance of aerostatic thrust bearings considering the fluid-structure interaction. Tribology International, 156. https://doi.org/10.1016/j.triboint.2020.106849

[3] Liszka, T., and J. Orkisz. 1980. "The Finite Difference Method at Arbitrary Irregular Grids and Its Application in Applied Mechanics." Computers & Structures 11(1–2):83–95. doi: 10.1016/0045-7949(80)90149-2.

[4] Zaghloul, N. A. (1982). The application of the FEM and FDM to flow separation pattern: A comparison. Applied Mathematical Modelling, 6(1), 28–34. https://doi.org/10.1016/S0307-904X(82)80019-X