Convergence Analysis in 2D Heat Conduction Using Finite Element Meshes

Wisang Sugiarta
Department of Computer Science, University of Colorado, Boulder

This project applies the Finite Element Method (FEM) to solve a 2D heat conduction problems, focusing on the convergence behavior of FEM solutions as the mesh is refined. Two types of meshes are examined: bilinear quadratic and triangular elements. The aim is to evaluate how the choice of mesh impacts the accuracy of the computed temperature field and associated thermal quantities, the to explore the computation time of each mesh configuration. The analysis is performed for a square plate with specified boundary conditions, where both types of meshes yield solutions that closely approximate the analytical solution.

I. Introduction

Finite Element Analysis (FEA) is a popular numerical model used to approximate solutions of physical problems governed by differential equations. It is widely applied across engineering problems such as structural mechanics and fluid dynamics where analytical solutions are difficult. FEA breaks a complicated problem into smaller, simpler geometries, which are easier to solve, allowing for the analysis of the broader problem and its boundary conditions.

Convergence analysis is essential to numerical modeling in that it allows researchers to ensure numerical solutions approach true solutions as models are refined or changed. In the context of 2D heat conduction, this analysis will ensure that the computed temperature field and associated thermal quantities approach the analytical solution with different meshes.

The 2D heat conduction problem, in particular, is governed by partial differential equations and often involves complex geometries that lack analytical solutions. FEA provides a framework for approximating these solutions by discretizing the domain into smaller elements which will be disucesed further. However, the accuracy of FEA depends significantly on factors such as element size, shape, and mesh structure. A poorly designed mesh may lead to inaccurate results or excessive computational costs, while an overly fine mesh might unnecessarily increase computational time without proportional gains in accuracy.

The primary objective of this study is to systematically investigate the convergence behavior of FEM solutions to 2D heat conduction problems under mesh refinement. Finally, this convergence study provides valuable insights into the performance of finite element models for 2D heat conduction and serves as a foundation for solving more complex thermal problems. The main contribution of this work is the code from scratch enabling the mesh convergence.

II. Background and Methodology

A. 2D Heat Conduction

In this section, we want to build the framework for the 2D heat conduction. Consider a square plate with sides of length L as depicted in Figure 3. The plate will have boundary conditions of prescribed temperatures of g_1 on its side, $g2$ on top and 0 on the bottom. We will assume not exterior heat flux. We introduce the equation that governs the temperature in strong form:

$$(S)\begin{cases} \text{Find } \theta(x,y) \text{ such that:} \\ q_{i,i} = f \quad \in \omega, \\ \theta = g_\theta \text{ on } \Gamma_\theta, \\ -q_i n_i = q \text{ on } \Gamma_q. \end{cases} \tag{1}$$

Where, $\boldsymbol{q} = -\boldsymbol{k} \cdot \frac{\partial \theta}{\partial \boldsymbol{x}}$

We apply the Method of Weighted Residuals to the balance of energy. We use the weighting function, $w = \partial\theta$. Then following [1], we can write the weak form:

$$\int_\Omega w(q_{i,i} - f)da = 0 \tag{2}$$

$$(W)\begin{cases}\text{Find } \theta(x,y) \in \theta : \Omega \rightarrow R, \theta \in H^1, \theta = g_\theta \in \Gamma_\theta \text{ such that,}\\ \int_\Omega w_{,i}\kappa_{ij}\theta_j da = \int_\Omega wf da + \int_\Gamma wq ds\\ \text{holds } \forall w(x,y) \in w : \Omega \rightarrow, w \in H^1, w = 0 \in \Gamma_\theta\end{cases} \tag{3}$$

Finally, we can present the Galerkin form of the heat equation from [1]

$$(W)\begin{cases}\text{Find } \theta^h(x,y) \in \theta^h : \Omega^h \rightarrow R, \theta^h \in H^1, \theta^h = g_\theta \in \Gamma_\theta^h \text{ such that,}\\ \int_\Omega w_{,i}\kappa_{ij}\theta_j da = \int_\Omega wf da + \int_\Gamma wq ds\\ \text{holds } \forall w(x,y) \in w : \Omega \rightarrow, w \in H^1, w = 0 \in \Gamma_\theta\end{cases} \tag{4}$$

The Galerkin form is a type of weighted residual method where the residual of the governing equation 1 is minimized over the solution domain. The Galerkin method leads to a system of linear or nonlinear equations:

$$\boldsymbol{K} \cdot \boldsymbol{d} = \boldsymbol{F} \tag{5}$$

Where d is the vector of nodal temperatures to be solved for and the boundary condition are applied on $\boldsymbol{K}$ and $\boldsymbol{F}$.

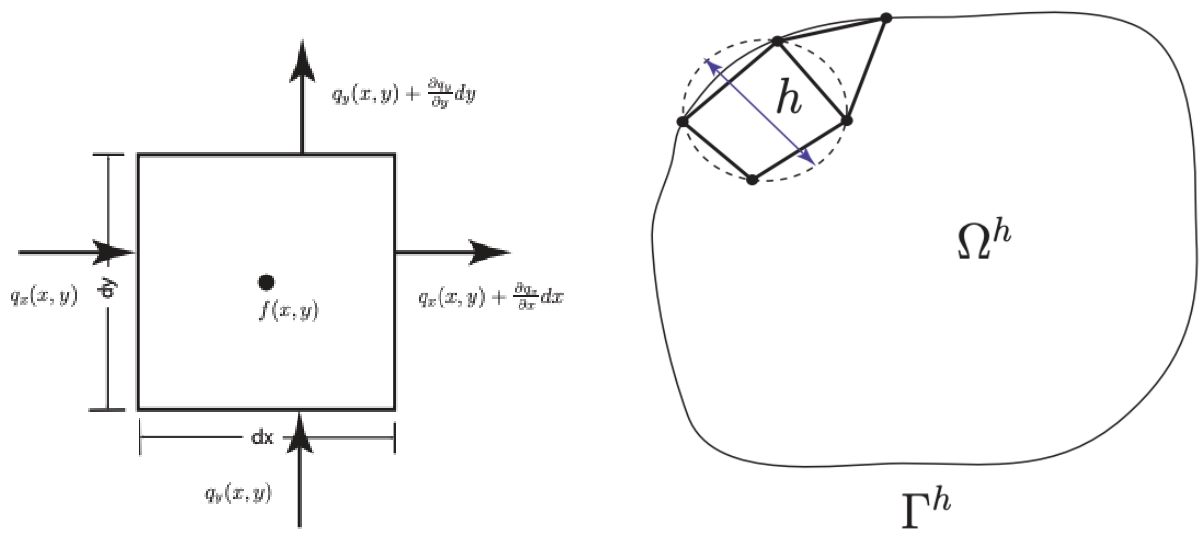

Fig. 1 Visualization of 2D Head Conduction from [2].

We can also take advantage of the symmetry in this problem. We will only consider the right half of the plate (X domain: $[0.05, 0.1]$) and assume the solution will be symmetrical, allowing us to solve for a smaller problem.

In the 2D head conduction problem, there is an analytical solution proposed in [1] and shown in Equation 6

B. Analytic Solution

Rarely, there are analytical solutions to PDEs. However, in the 2D head conduction problem, there is an analytical solution proposed in [1] and shown in Equation 6. The solution can be illustrated on half the plate as in Figure 2.

$$\theta(x,y) = ((g2 - g1) + g_1) \cdot \frac{2}{\pi}\sum_{n=1}^{\infty}\left[\frac{(-1)^{n+1}+1}{n}\left(\sin\frac{n\pi x}{L}\right)\frac{\cosh\frac{n\pi y}{L}}{\cosh n\pi}\right] \tag{6}$$

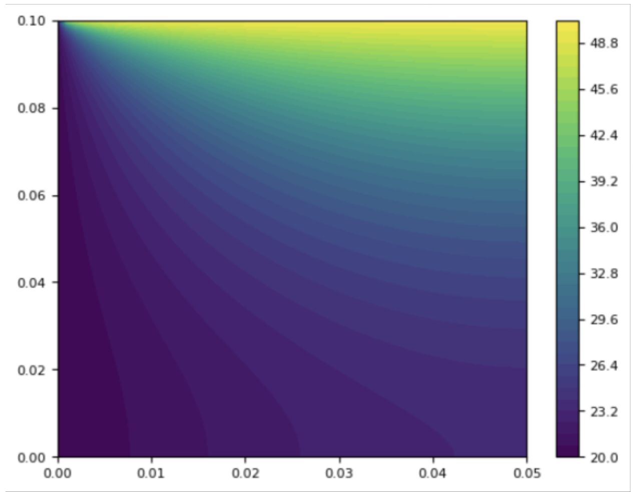

Fig. 2 Temperature plot for analytic solution.

C. Bilinear Quadratic Mesh

For higher-order accuracy, bilinear elements are used. These elements are based on quadrilateral geometries. Since we already dove deeply into these meshes in class, I will just refer to the course text [2].

D. Triangular Mesh

We aim to discretize this domain of the 2D problem into smaller elements, which in this case are triangles. The number of triangles is specified by the number of divisions in the x-axis, N_x, and the y-axis N_y. These parameters will control the refinement of the mesh. The x-coordinate of each node is determined by dividing the horizontal length of the domain into N_x intervals, same is true for the y-coordinate. These nodes represent the vertices of the triangles and form the foundation for creating the triangular elements. To construct the mesh elements, each rectangular region formed by four adjacent nodes is split into two triangles by drawing a diagonal between opposite corners. This results in two triangles per rectangle, each defined by three nodes.

Boundary conditions are then applied by identifying the nodes along the edges of the 2D shape, exactly the same as with the other element shapes. The nodes along the boundary edges are assigned prescribed temperature values. For each triangular element, the shape functions are computed. In the case of linear triangular elements, these are the simplest form of shape functions, which linearly interpolate the temperature field within each element based on the values at the three nodes. The exact shape functions are the following, found in [1].

$$N_1(\xi, \eta) = 1 - \xi - \eta$$

$$N_2(\xi, \eta) = \xi$$

$$N_3(\xi, \eta) = \eta$$

Once the shape functions are defined, the element stiffness matrix and force vector for each triangular element can be computed, similarly to quads.

After the individual element matrices are computed, they are assembled into the global stiffness matrix K and global force vector F using boundary conditions. The global system of equations is then solved.

III. Results

In this section, we explore the results of the different meshes and computation time. Since we have already explored the results of the bilinear quadratic mesh in class extensively, we will focus more on the triangular meshes.

From Figure 3, we can see there's an obvious bias in the meshes comparing to the analytical solution. While it has the same shape, the only difference is the initial temperature which can only be attributed to inaccurate boundary

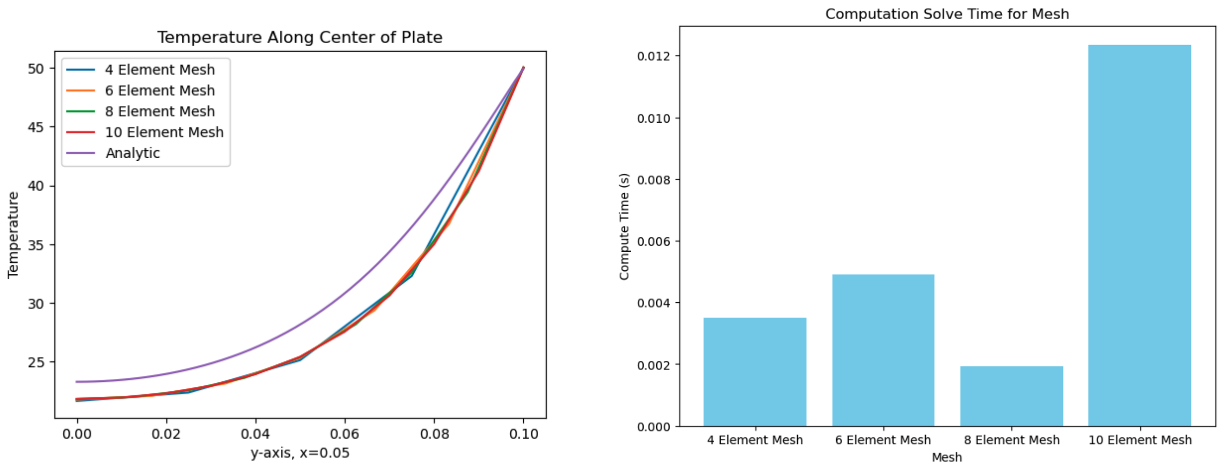

Fig. 3 Bilinear quadratic mesh temperature and computation time.

condition on some node. As I wanted to keep the same code template as in PS6, I didn't modify it such that it matched the analytical solution. We can also observe something uncharacteristic in the computation time solve. We would expect to find that for larger mesh sizes, we get larger computation times. While this is true in this example, the 8 element node has a smaller run time. I check this numerous times with other sizes and 8 mesh size remains constantly lower than sizes of 4, 5, and 6.

Mesh Configuration	**Computation Time**	**Mean Percent Error**
4 Bilinear Quadratic Elements	3.86×10^{-3}s	0.0748%
6 Bilinear Quadratic Elements	5.46×10^{-3}s	0.0749%
8 Bilinear Quadratic Elements	2.08×10^{-3}s	0.0755%
10 Bilinear Quadratic Elements	12.1×10^{-3}s	0.076 %
8 Triangular Elements	5.39×10^{-3}s	0.0197 %
18 Triangular Elements	8.26×10^{-3}s	0.00843 %
32 Triangular Elements	11.5×10^{-3}s	0.00439 %
50 Triangular Elements	15.8×10^{-3}s	0.00267 %

Table 1 Computational resource required for mesh configuration.

Similarly, the triangular mesh performance can be observed in Figure 4. It's easy to see that the smaller meshes are more coarse and there are not many nodes along the $x = 0.05$ line. As we increase size, we can see the amount of nodes (and thus temperature points) increase and more closely align with the analytical solution and thus converges.

The results show that both the bilinear quadratic and triangular meshes provide reasonable approximations to the analytical solution, with the temperature distribution closely matching the expected results as the mesh is refined. However, the triangle quadratic mesh consistently exhibited higher accuracy, with lower compute time.

In terms of computational efficiency, the triangular mesh proved to be more efficient than the bilinear quadratic mesh for coarser grids, requiring less computational time for similar resolutions. The flexibility of triangular elements makes them well-suited for handling geometrically complex domains, where a regular quadrilateral mesh might be less practical.

The convergence analysis also confirmed that both meshes exhibit the expected behavior as the mesh is refined: the solution converges toward the analytical solution, and the mean error decreases with finer meshes. While the triangular mesh shows faster convergence and lower error for a given mesh size.

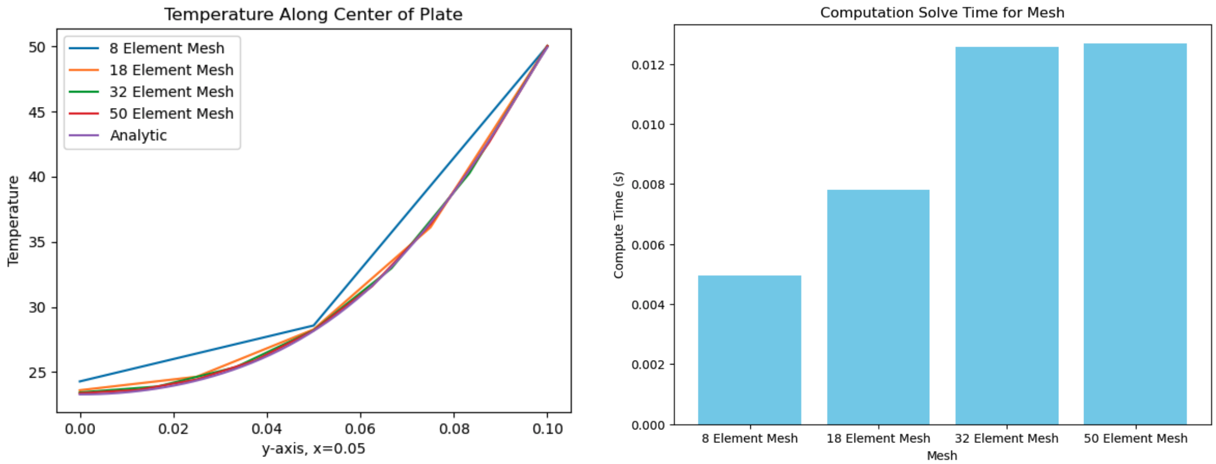

Fig. 4 Triangular mesh temperature and computation time.

IV. Conclusion

In this study, we explored the application of the Finite Element Method (FEM) for solving the 2D heat conduction problem. The primary goal was to investigate the convergence behavior of FEM solutions under mesh refinement, comparing the performance of two types of mesh: bilinear quadratic and triangular elements. Through this comparison, we gained valuable insights into the accuracy, computational efficiency, and practical implications of these mesh types in solving heat conduction problems.

In conclusion, the choice between bilinear quadratic and triangular meshes depends on the specific requirements of the problem. If high accuracy is critical and computational resources are sufficient, the bilinear quadratic mesh is the preferable option. However, for problems with more complex geometries or when computational efficiency is a priority, the triangular mesh provides a more flexible and resource-efficient solution.

Future work could focus on hybrid mesh approaches that combine the strengths of both mesh types, potentially enhancing both accuracy and efficiency.

Appendix

I found out pretty early that the only convergence to show for this problem was compute time and error. I thought that the best way to present results is in the figures.

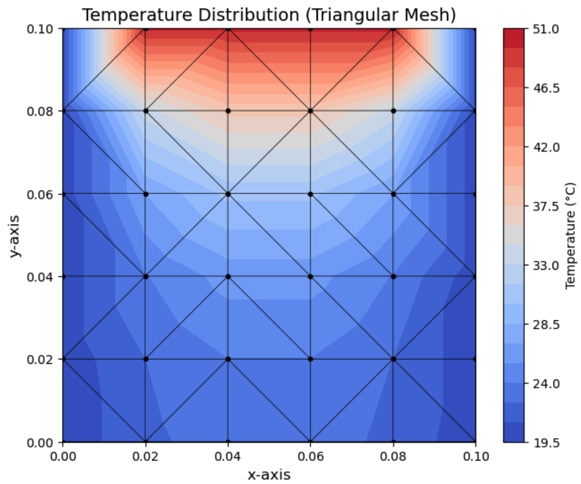

Fig. 5 Temperature plot of 2D plate with triangular meshes.

References

[1] Hughes, T. J. R., *The finite element method: Linear static and dynamic finite element analysis*, Dover Publications, 2000.

[2] Regueiro, R. A., *CVEN 4511/5511: Introduction to Finite Element Analysis*, CU BOULDER, 2024.