Universidad de los Andes

FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

Homework 4, Report

Finite Elements

Professor:
Jose A. Abell

Students:
Bernardo Caprile
Pedro Valenzuela

May 26, 2025

Index

GitHub Repository

The code and data for this project are available on GitHub at the following link:

https://github.com/berckanala/01-Finite-Element

1 Introduction

This assignment implements a numerical solver for the two-dimensional Poisson equation with Dirichlet boundary conditions, using the finite element method with triangular linear (CST) and quadratic (LST) elements. The objective is to approximate the weak form of the problem through the Galerkin method on unstructured meshes.

The meshes are generated with Gmsh and read from `.msh` files, from which node coordinates, element connectivities, and boundary conditions are extracted. Element stiffness matrices are computed using numerical integration of shape function gradients, then assembled into the global system. After applying the boundary conditions, the resulting linear system is solved to obtain the approximate solution.

Verification is carried out using the Method of Manufactured Solutions (MMS), comparing the numerical result with a known analytical solution. Errors are computed in the L^2 and H^1 norms, and convergence rates are analyzed as the mesh is refined, confirming the expected theoretical orders.

2 Theoretical Background

This section presents the mathematical formulation underlying the numerical solution of the Poisson equation with Dirichlet boundary conditions using the Finite Element Method (FEM). The derivation of the weak form, the construction of the finite element spaces, and the discretization process are detailed, providing the foundation for the implementation described in the following sections.

2.1 Strong Form

We consider the Poisson equation defined over a bounded domain $\Omega \subset \mathbb{R}^2$ with boundary $\partial\Omega$. The problem is given by:

$$\begin{aligned} -\Delta u &= f \quad \text{in } \Omega, \\ u &= g \quad \text{on } \partial\Omega, \end{aligned} \tag{1}$$

where $u : \Omega \to \mathbb{R}$ is the unknown scalar field (e.g., temperature, potential), $f : \Omega \to \mathbb{R}$ is a given source function, and $g : \partial\Omega \to \mathbb{R}$ prescribes the Dirichlet boundary condition.
The operator $\Delta u = \nabla \cdot \nabla u$ denotes the Laplacian of u, which represents the divergence of the gradient of u. The strong form requires the solution u to be twice continuously differentiable in Ω and to satisfy the differential equation and boundary condition pointwise.

2.2 Weak Form

To derive the weak form, we multiply the strong form (1) by a test function $v \in H_0^1(\Omega)$ and integrate over the domain Ω:

$$\int_\Omega (-\Delta u)\, v \, dx = \int_\Omega f \, v \, dx. \tag{2}$$

Applying Green's first identity (integration by parts), we obtain:

$$\int_\Omega \nabla u \cdot \nabla v \, dx - \int_{\partial\Omega} \frac{\partial u}{\partial n} v \, ds = \int_\Omega f \, v \, dx, \tag{3}$$

where $\frac{\partial u}{\partial n}$ denotes the normal derivative on the boundary $\partial\Omega$. Since v vanishes on the boundary (as $v \in H_0^1(\Omega)$), the boundary term disappears, leading to the weak form:

$$\int_\Omega \nabla u \cdot \nabla v \, dx = \int_\Omega f \, v \, dx \quad \forall v \in H_0^1(\Omega). \tag{4}$$

The weak formulation seeks a function $u \in H^1(\Omega)$ such that $u = g$ on $\partial\Omega$ and the variational identity (4) is satisfied for all test functions v in $H_0^1(\Omega)$. This relaxation of the differentiability

requirement allows for a broader class of admissible solutions and serves as the basis for the finite element discretization.

2.3 Hilbert Spaces

The weak formulation of the Poisson equation is naturally set in the framework of Sobolev spaces, which are examples of Hilbert spaces. Specifically, we work with the space $H^1(\Omega)$, defined as:

$$H^1(\Omega) = \left\{ u \in L^2(\Omega) \;\middle|\; \nabla u \in (L^2(\Omega))^2 \right\}, \tag{5}$$

equipped with the norm:

$$\|u\|_{H^1(\Omega)} = \left(\int_\Omega u^2 \, dx + \int_\Omega |\nabla u|^2 \, dx \right)^{1/2}. \tag{6}$$

The subspace $H_0^1(\Omega)$ consists of functions in $H^1(\Omega)$ that vanish on the boundary $\partial\Omega$ (in the trace sense):

$$H_0^1(\Omega) = \left\{ u \in H^1(\Omega) \;\middle|\; u = 0 \text{ on } \partial\Omega \right\}. \tag{7}$$

Both $H^1(\Omega)$ and $H_0^1(\Omega)$ are Hilbert spaces, meaning they are complete inner product spaces. The inner product in $H_0^1(\Omega)$ is typically defined as:

$$(u, v)_{H_0^1} = \int_\Omega \nabla u \cdot \nabla v \, dx, \tag{8}$$

which induces the norm:

$$\|u\|_{H_0^1(\Omega)} = \left(\int_\Omega |\nabla u|^2 \, dx \right)^{1/2}. \tag{9}$$

These spaces provide the functional setting for the weak formulation of elliptic partial differential equations and are essential for establishing the well-posedness and stability of the finite element method.

2.4 Galerkin Method Analysis

Here, we restrict our attention to symmetric bilinear forms, that is,

$$a(u, v) = a(v, u) \quad \forall u, v \in V. \tag{10}$$

Although Galerkin methods can be extended to nonsymmetric forms (e.g., Petrov–Galerkin methods), the symmetric case allows for a cleaner and more direct theoretical analysis.

The analysis of Galerkin methods proceeds in two main steps. First, we establish that the Galerkin formulation is well-posed in the sense of Hadamard, meaning that it admits a unique solution that depends continuously on the data. Second, we investigate the approximation quality of the Galerkin solution u_n, defined in a finite-dimensional subspace $V_n \subset V$.
The theoretical foundation relies on two fundamental properties of the bilinear form $a(\cdot, \cdot)$:

- **Boundedness:** There exists a constant $C > 0$ such that for all $u, v \in V$,

$$a(u, v) \leq C\|u\| \, \|v\|. \tag{11}$$

- **Ellipticity (or coercivity):** There exists a constant $c > 0$ such that for all $u \in V$,

$$a(u, u) \geq c\|u\|^2. \tag{12}$$

Under these two conditions, the Lax–Milgram theorem guarantees the existence and uniqueness of the solution to the weak formulation. The norm $\| \cdot \|$ used here is typically the energy norm induced by the bilinear form.

Well-posedness of the Galerkin Equation

Since the finite-dimensional subspace $V_n \subset V$, the bilinear form $a(\cdot, \cdot)$ remains bounded and elliptic on V_n. Therefore, the Galerkin problem inherits the well-posedness of the original continuous problem. This implies that the discrete problem also has a unique solution.

Quasi-optimality: Céa's Lemma

Let $u \in V$ be the exact solution of the weak formulation, and let $u_h \in V_h \subset V$ be its Galerkin approximation in a finite-dimensional subspace. Then, assuming that the bilinear form $a(\cdot, \cdot)$ is bounded and coercive in the $H^1(\Omega)$ norm, Céa's lemma gives the following estimate:

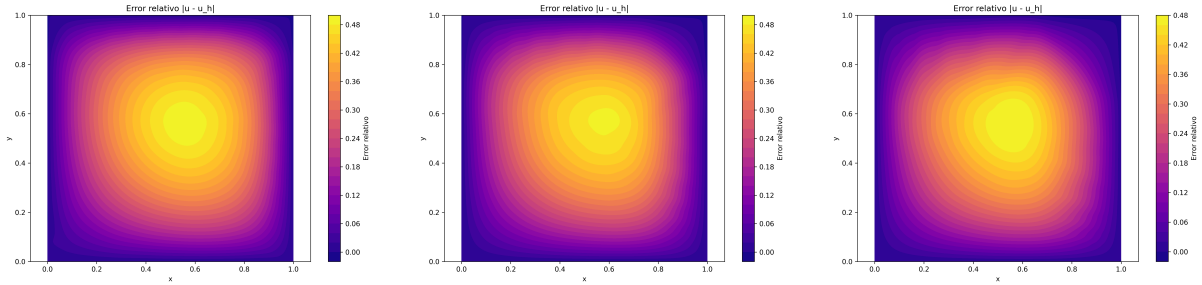
$$\|u - u_h\|_{H^1(\Omega)} \leq \frac{C}{\alpha} \inf_{v_h \in V_h} \|u - v_h\|_{H^1(\Omega)}. \tag{13}$$

This means that the Galerkin solution u_h is quasi-optimal: it is, up to a constant factor C/α, as close to the exact solution u as the best possible approximation in the finite element space V_h. Therefore, the convergence of the method depends directly on the approximation properties of V_h.

3 Results

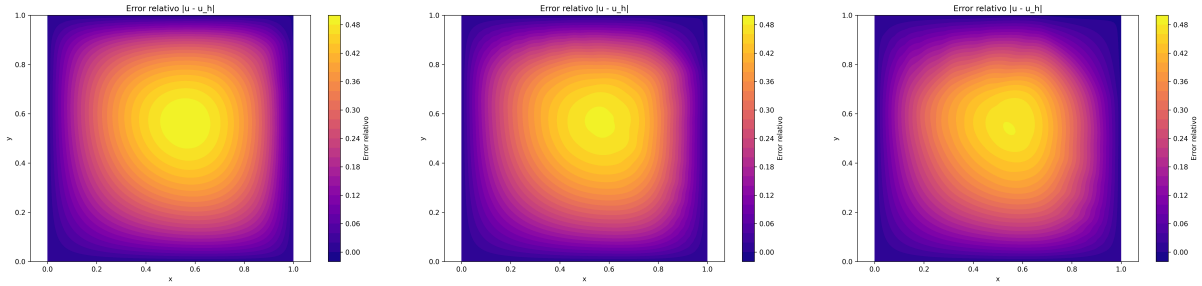
In this section, we present the results for the error obtained from the numerical solution of the Poisson equation using the finite element method (FEM) with triangular linear (CST) and quadratic (LST) elements.

3.1 CST Elements



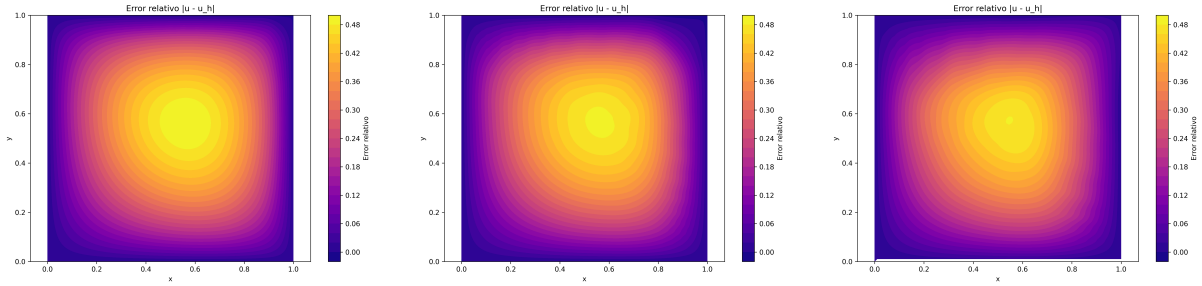
(a) CST error (N = 10, r = 1.1) (b) CST error (N = 10, r = 1.2) (c) CST error (N = 10, r = 1.3)

Figure 1: Relative error for CST elements with $N = 10$ and different values of r.



(a) CST error (N = 20, r = 1.1) (b) CST error (N = 20, r = 1.2) (c) CST error (N = 20, r = 1.3)

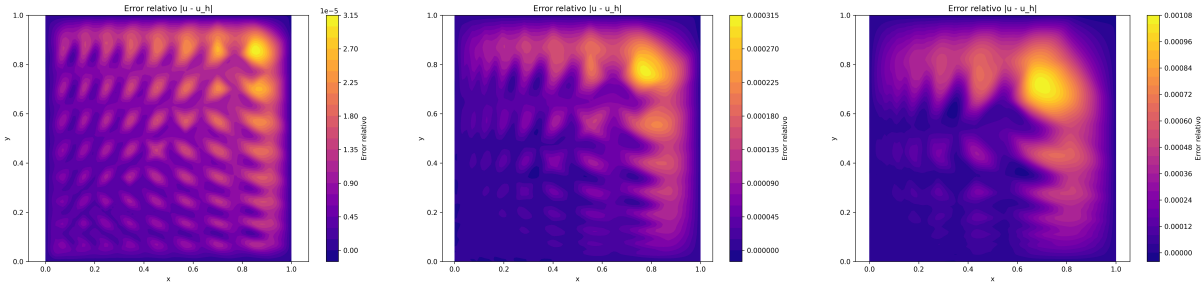
Figure 2: Relative error for CST elements with $N = 20$ and different values of r.



(a) CST error (N = 30, r = 1.1) (b) CST error (N = 30, r = 1.2) (c) CST error (N = 30, r = 1.3)

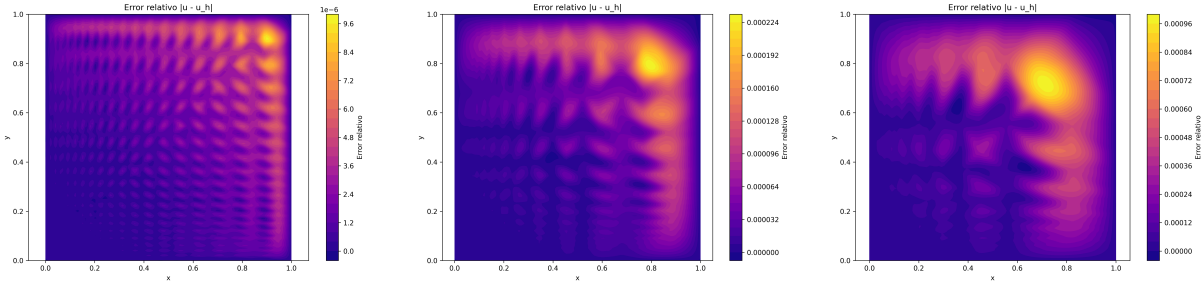
Figure 3: Relative error for CST elements with $N = 30$ and different values of r.

3.2 LST Elements



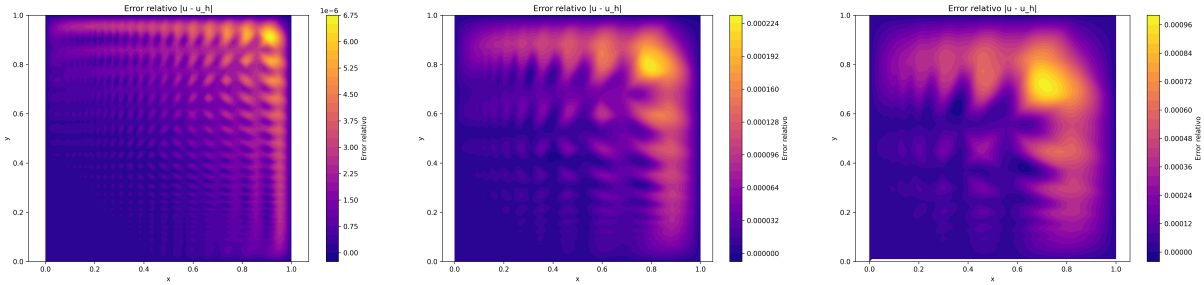
(a) LST error (N = 10, r = 1.1) (b) LST error (N = 10, r = 1.2) (c) LST error (N = 10, r = 1.3)

Figure 4: Relative error for LST elements with $N = 10$ and different values of r.



(a) LST error (N = 20, r = 1.1) (b) LST error (N = 20, r = 1.2) (c) LST error (N = 20, r = 1.3)

Figure 5: Relative error for LST elements with $N = 20$ and different values of r.



(a) LST error (N = 30, r = 1.1) (b) LST error (N = 30, r = 1.2) (c) LST error (N = 30, r = 1.3)

Figure 6: Relative error for LST elements with $N = 30$ and different values of r.

3.3 Comparative Analysis

In the following tables, we summarize the relative errors obtained for both CST and LST elements across different values of N (number of elements) and r (refinement factor). The maximum, average, and minimum errors are presented to provide a comprehensive view of the performance of each element type.

Table 1: Relative errors for CST elements for different values of N and r.

N	r	**Max Error**	**Avg Error**	**Min Error**
10	1.1	4.94e-01	1.66e-01	0.00e+00
	1.2	4.83e-01	1.14e-01	0.00e+00
	1.3	4.30e-01	7.23e-02	0.00e+00
20	1.1	4.98e-01	1.57e-01	0.00e+00
	1.2	4.83e-01	5.67e-02	0.00e+00
	1.3	4.33e-01	2.45e-02	0.00e+00
30	1.1	4.96e-01	1.26e-01	0.00e+00
	1.2	4.83e-01	2.78e-02	0.00e+00
	1.3	4.33e-01	1.10e-02	0.00e+00

Table 2: Relative errors for LST elements for different values of N and r.

N	r	**Max Error**	**Avg Error**	**Min Error**
10	1.1	2.98e-05	6.25e-06	0.00e+00
	1.2	2.64e-04	2.42e-05	0.00e+00
	1.3	8.58e-04	5.87e-05	0.00e+00
20	1.1	9.11e-06	8.98e-07	0.00e+00
	1.2	1.96e-04	7.42e-06	0.00e+00
	1.3	7.87e-04	1.67e-05	0.00e+00
30	1.1	6.38e-06	4.07e-07	0.00e+00
	1.2	1.92e-04	3.51e-06	0.00e+00
	1.3	7.86e-04	7.36e-06	0.00e+00

In the next table, we compare the average relative errors between CST and LST elements for different values of N and r. This comparison highlights the performance differences between the two element types in terms of accuracy.

Table 3: Comparison of average relative error between CST and LST elements, organized by N and r.

N	r	**Avg Error CST**	**Avg Error LST**
10	1.1	1.66e-01	6.25e-06
	1.2	1.14e-01	2.42e-05
	1.3	7.23e-02	5.87e-05
20	1.1	1.57e-01	8.98e-07
	1.2	5.67e-02	7.42e-06
	1.3	2.45e-02	1.67e-05
30	1.1	1.26e-01	4.07e-07
	1.2	2.78e-02	3.51e-06
	1.3	1.10e-02	7.36e-06

Here, we observe that the average relative errors for CST elements are significantly higher than those for LST elements across all values of N and r. This indicates that LST elements provide a more accurate approximation of the solution to the Poisson equation compared to CST elements, especially as the mesh is refined.

4 Conclusion

This study implemented and analyzed the finite element solution of the two-dimensional Poisson equation using both linear (CST) and quadratic (LST) triangular elements on geometrically refined unstructured meshes. The method was verified using the Method of Manufactured Solutions, allowing precise quantification of convergence behavior.

The results confirm that increasing the number of subdivisions N consistently reduces the relative error for both CST and LST elements. Moreover, LST elements showed significantly better accuracy compared to CST, often yielding errors several orders of magnitude lower for the same mesh resolution.

However, it was also observed that the refinement parameter r plays a more subtle role. While moderate values of r can improve accuracy by concentrating nodes where the solution varies rapidly, excessive refinement (e.g., $r = 1.3$) led to distortions in element quality. This affected LST elements more strongly, as their quadratic nature makes them more sensitive to mesh shape and distribution. In some cases, the relative error of LST increased slightly when the mesh refinement was poorly aligned with the regions of greatest solution variation.

In conclusion, LST elements provide superior accuracy and faster convergence, but their performance is highly dependent on mesh quality and proper refinement strategy. Careful tuning of the mesh parameters is essential to fully leverage the benefits of higher-order elements in finite element analysis.