NTNU

Kunnskap for en bedre verden

TMA4212 - Numerical solution of differential equations by difference methods

Finite element methods for Poisson equation and optimal control problem

Authors:
Sindre Skau Gulliksrud Hans Ljungquist

28.02.2025

Table of Contents

1 Introduction

In this paper, the aim is to solve the 1D Posson equation numerically by implementing a Galerkin method by using quadratic basis functions. Our goal is to approximate the exact solution by searching for a solution in a finite subspace V_h of the complete infinite solution space V. We then analyze the methods accuracy and numerical properties, before extending the approach to a related optimal control problem, and as such demonstrating the viability of using a finite element framework in real world scenarios.

We start by considering the 1D Poisson problem:

$$-\Delta u = f \text{ on } \Omega = (0,1), \quad u(0) = u(1) = 0$$

which has the variational form

$$\text{find } u \in V \text{ such that } a(u,v) = F(v), \quad \forall v \in V,$$

where

$$V = H_0^1(\Omega), \quad a(u,v) = \int_0^1 u_x v_x dx, \quad F(v) = \int_0^1 fv dx \tag{1}$$

Since the exact solution is in the space $V = H_0^1(\Omega)$, where $\Omega = (0,1)$, we approximate it by looking for a solution in the discretized finite space $V_h = X_h^2 \cap H_0^1(\Omega)$, where X_h^2 is the Lagrange finite dimensional element space.

$$X_h^2 = \{v \in C^0(\Omega) : v|_K \in \mathbb{P}_2, \forall K \in \mathcal{T}_h\}$$

This leads to the Galerkin method for solving this then being:

$$\text{Find } u_h \in V_h \text{ such that } a(u,v) = F(v), \quad \forall v_h \in V_h \tag{2}$$

We also remember that a function $v \in V_h$ can be written as a linear combination of constant weights and basis functions as follows:

$$v(x) = \sum_{i=0}^{M} v_i \phi_i(x)$$

2 Numerical implementation

2.1 Explanation of method

We want to write and test code for solving (2) in the space $V_h = X_h^2 \cap H_0^1(\Omega)$.

To do this we first choose a partition $\mathcal{T}_h$, which creates the elements $K_k = [x_{2k}, x_{2k+2}]$, each with a midpoint x_{2k+1}. For now the nodes in the partition is equidistributed, but the method works just as well for varying step sizes. Then we define a reference element $\hat{K} = [0,1]$, and shape functions $\Psi_i \in \mathbb{P}^2$ on the reference element $\hat{K}$ as follows:

$$\Psi_0(\xi) = 2(\xi - 0.5)(\xi - 1)$$
$$\Psi_1(\xi) = -4\xi(\xi - 1)$$
$$\Psi_2(\xi) = 2\xi(\xi - 0.5)$$

Each of the shape functions satisfy the Kroenecker-Delta property $\Psi(\xi_\beta) = \delta_{\alpha,\beta}$ at the nodes $\xi_\beta \in \{0, \frac{1}{2}, 1\}$ on the reference element.

We also define a mapping from the reference element to the physical elements $\Phi_K : \hat{K} \to K_k$:

$$\Phi_K(\xi) = x_{2k} + (x_{2k+2} - x_{2k})\xi,$$

And a local to global mapping θ, remembering that neighbouring elements share a common global node, $i = \theta(k, \alpha)$, where k is the element index, and α is the node index of the reference element $\hat{K}$:

$$i = 2k + \alpha$$

The mapping Φ_K is used in the implementation to transform the integral over each of the physical element, to integrals over the reference element. For integrals over the derivatives of the basis functions, such as the stiffness matrix, this substitution and the chain rule leads to:

$$\int_{K_k} \phi_{\theta(k,\alpha)}(x)\phi_{\theta(k,\beta)}(x)dx = \frac{1}{(x_{2k+2} - x_{2k})} \int_0^1 \Psi_\alpha(\xi)\Psi_\beta(\xi)d\xi$$

And likewise for the load vectors, where we integrate over the basis functions themselves, this leads to:

$$\int_{K_k} f(x)\phi_{\theta(k,\alpha)}(x)dx = (x_{2k+2} - x_{2k}) \int_0^1 f(\Phi_K(\xi))\Psi_\alpha(\xi)d\xi$$

These transformed integrals can then be computed using Simpson's rule. By dividing the domain of the reference element, which in particular is the domain we integrate over, into 7 subintervals, Simpson's rule becomes exact up to the order necessary for our use case. The computed elemental stiffness matrix A^{K_k} and the elemental load vector b^{K_k} are then assembled into the extended stiffness matrix and load vector.

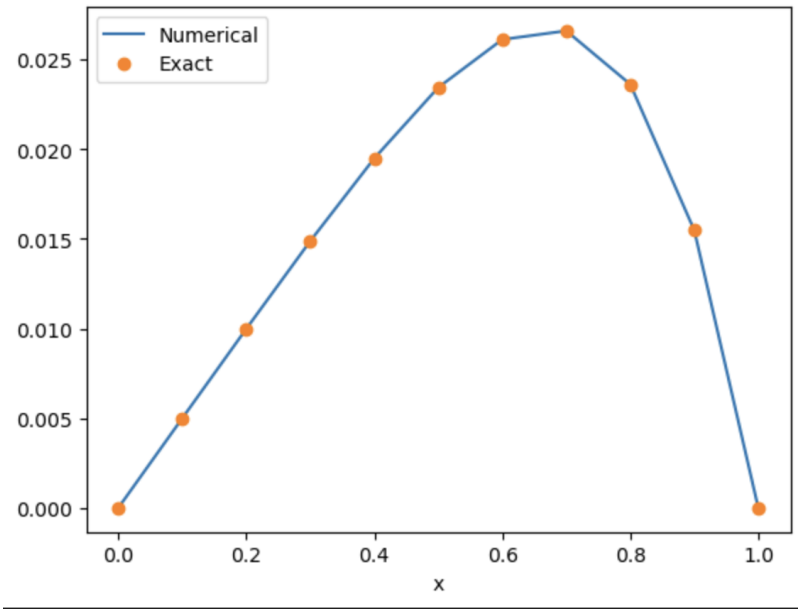


Figure 1

In Figure 1 we have plotted the exact solution of $f(x) = x^3$ with Dirichlet boundary conditions $u_0 = u_1 = 0$ against the numerical approximation. From the plot, we see that the numerical solution obtained by our finite element method follows the same shape as the exact solution. Specifically, both solutions rise from zero at $x = 0$, reach a peak around $x = 0.7$, and then decrease back to zero at $x = 1$. This illustrates that the quadratic elements capture the features of the exact solution with good accuracy. This behavior is consistent with the second order convergence of a piecewise quadratic FEM.

3 Error analysis

We wish to find an upper bound for the error in the method above. To do so, we must first ensure that the weak formulation is well posed by verifying that the Lax–Milgram theorem holds. This is crucial because the theorem guarantees the existence, uniqueness, and stability of the weak solution, and these properties are essential for our error estimates, such as those derived from Céa's lemma, which relies on the bilinear form being coercive and continuous.

3.1 Lax-Milgram theorem

3.1.1 Continuity of the bilinear form

We must show there exists a constant $M > 0$ such that

$$|a(u, v)| \leq M\|u\|_{H^1(\Omega)}\|v\|_{H^1(\Omega)} \quad \forall\, u, v \in H_0^1(\Omega).$$

Using the Cauchy–Schwarz inequality,

$$|a(u, v)| = \left|\int_\Omega \nabla u \cdot \nabla v \, dx\right| \leq \|\nabla u\|_{L^2(\Omega)}\|\nabla v\|_{L^2(\Omega)}.$$

Since in $H_0^1(\Omega)$ the norm $\|w\|_{H^1(\Omega)}$ is equivalent to $\|\nabla w\|_{L^2(\Omega)}$ (by the Poincaré inequality), we may take $M = 1$.

3.1.2 Coercivity of the bilinear form

We also need to prove there exists $\alpha > 0$ such that

$$a(v, v) \geq \alpha\|v\|^2_{H^1(\Omega)} \quad \forall\, v \in H_0^1(\Omega).$$

Because

$$a(v, v) = \int_\Omega |\nabla v|^2 \, dx = \|\nabla v\|^2_{L^2(\Omega)},$$

and the Poincaré inequality provides a constant $C_P > 0$ with

$$\|v\|_{L^2(\Omega)} \leq C_P\|\nabla v\|_{L^2(\Omega)},$$

we have

$$\|v\|^2_{H^1(\Omega)} = \|v\|^2_{L^2(\Omega)} + \|\nabla v\|^2_{L^2(\Omega)} \leq (C_P^2 + 1)\|\nabla v\|^2_{L^2(\Omega)}.$$

And therefore

$$\|\nabla v\|^2_{L^2(\Omega)} \geq \frac{1}{C_P^2 + 1}\|v\|^2_{H^1(\Omega)}.$$

We can now choose:

$$\alpha = \frac{1}{C_P^2 + 1},$$

which proves coercivity.

Since the bilinear form $a(u, v)$ is continuous and coercive, the Lax–Milgram theorem guarantees that there exists a unique solution $u \in H_0^1(\Omega)$ to the weak formulation.

3.2 Error bound in $H^1(\Omega)$

We wish to derive an upper bound for the error measured in the H^1-norm,

$$\|u - u_h\|_{H^1(\Omega)},$$

where u is the exact solution of the Poisson problem and $u_h \in V_h$ is its finite element approximation in a space V_h consisting of piecewise polynomials of degree r.

The Galerkin approximation $u_h \in V_h \subset H_0^1(\Omega)$ satisfies:

$$a(u_h, v_h) = F(v_h) \quad \forall\, v_h \in V_h.$$

Subtracting this from the weak formulation yields the Galerkin orthogonality:

$$a(u - u_h, v_h) = 0 \quad \forall, v_h \in V_h.$$

We now choose v_h to be the best possible interpolant of $u \in V_h$, ie. the one that is closest to achieving $\inf_{v_h \in V_h} ||u - v_h||_V$, and by then invoking Céa's lemma, which follows from the previously proven coercivity and continuity of $a(\cdot, \cdot)$, we have that the finite element error is bounded by a constant times the best interpolant.

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

where M is the continuity constant and α is the coercivity constant of $a(\cdot, \cdot)$

By the polynomial interpolation error estimate described in Lemma 4.3 in Charles Curry's notes, if $u \in H^{r+1}(\Omega)$, then there exists an interpolant $v_h \in V_h$ such that:

$$\|u - v_h\|_{H^1(\Omega)} \leq C\, h^r\, \|u\|_{H^{r+1}(\Omega)}.$$

By now substituting the approximation property into Céa's lemma, we obtain:

$$\|u - u_h\|_{H^1(\Omega)} \leq \frac{M}{\alpha} C\, h^r\, \|u\|_{H^{r+1}(\Omega)}$$

We are not particularly interested in the constants, and therefore combine them all in to one common C, and get that the final error bound in the $H^1(\Omega)$-norm is:

$$\|u - u_h\|_{H^1(\Omega)} \leq C\, h^r\, \|u\|_{H^{r+1}(\Omega)}.$$

And since we have quadratic polynomials $r = 2$:

$$\underline{\underline{\|u - u_h\|_{H^1(\Omega)} \leq C\, h^2\, \|u\|_{H^3(\Omega)}}}$$

This means that the error decreases proportionally with the square of the largest step size, and from this it follows that for constant step sizes the order of convergence is therefore $r = 2$.

3.3 Error bound in $L^2(\Omega)$, and numerical verification

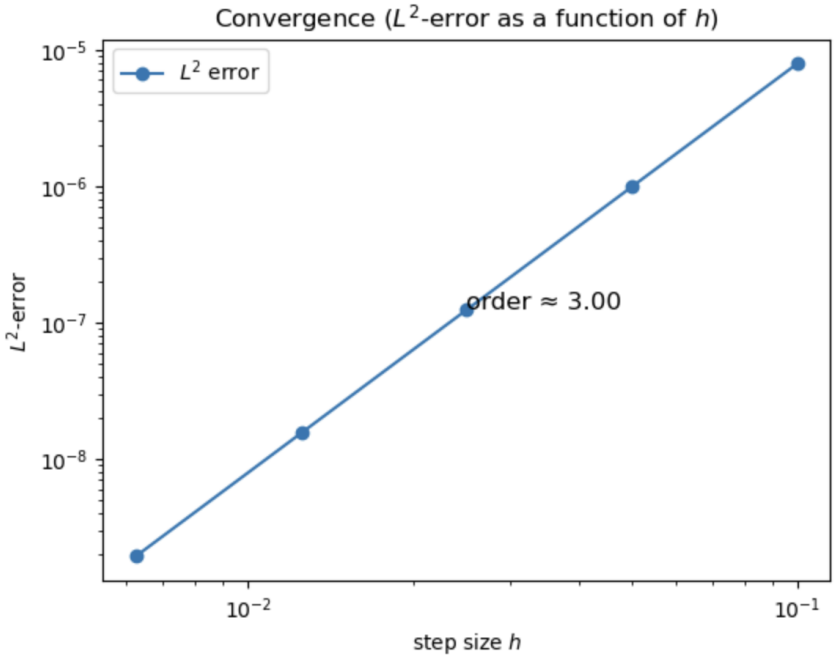


Figure 2

The plot in Figure 2 follows from a numerical analysis of the order of the error $\|u - u_h\|_{L^2(\Omega)}$ with the number of nodes being $[11, 21, 41, 81, 161]$ respectively, ie. we double the number of elements each time $[5, 10, 20, 40, 80]$, and calculate the L^2-norm of the error for each.

It shows that the L^2-norm decreases at a rate of approximately h^3, ie. order $r = 3$, which is one order higher than $r = 2$ that we have for H^1. This is exactly what we expect based Lemma 4.4 in Charles Curry's note and the calculated error in H^1. This improvement in convergence rate of the L^2-error compared to the H^1-error is also a widely known phenomenon for FEM methods.

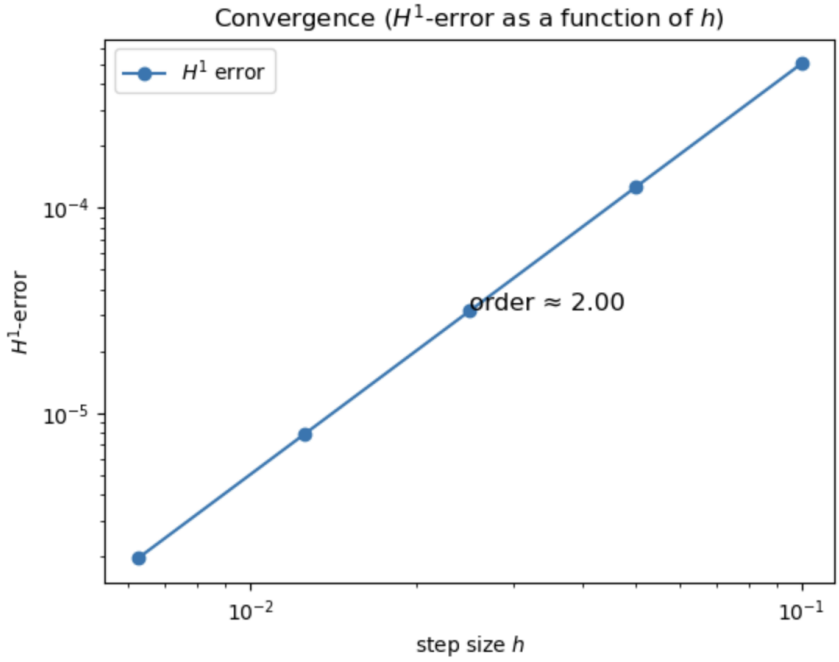


Figure 3

Figure 3 shows the equivalent plot for the H^1-error and confirms that the order is $r = 2$, just as we calculated in section 3.2, and also expected from the numerical calculations of the error in L^2.

4 Optimal control problem

4.1 Overview

Building on the foundation from the Galerkin FEM, we now extend our study to a related optimal control problem. In most real applications, it is not just important to solve a PDE, but also control its behavior to achieve a desired result. One such example is that of the temperature profile of a physical object, given a heat source and an associated cost of heating and cooling. In this case, we are considering a 1 dimensional rod, meaning we are still dealing with the 1D Poisson equation. Our goal is then to control the temperature profile of the rod y, such that it matches a desired temperature profile y_d as much as possible by applying a heat source u, while still keeping the cost low. Mathematically this optimal control problem is expressed as:

$$\min_{y,u} J(y,u) = \frac{1}{2}\int_0^1 |y - y_d|^2 dx + \frac{\alpha}{2}\int_0^1 u^2 dx, \tag{3}$$

$$\text{subject to } -\Delta y = u \text{ and } y(0) = y(1) = 0, \tag{4}$$

in the weak sense. With this formulation, we also have the ability to penalize excessive energy spending while also keeping the temperature close to where we want.

4.2 Finite element problem

We will now approximate 3 and 4 with the following constrained finite element minimization problem:

$$\min_{y_h, u_h \in V_h} \frac{1}{2} \| y_h - \overline{y}_d \|^2_{L^2(\Omega)} + \frac{\alpha}{2} \| u_h \|^2_{L^2(\Omega)} \tag{5}$$
$$\text{such that } a(y_h, v) = \langle u_h, v \rangle^2_L(\Omega) \text{ for all } v \in V_h$$

Where $\overline{y}_d$ is the interpolation of y_d onto X_h^2, and a is the bilinear form given in 1. To solve (5), we must interpret the problem as a real-valued minimization problem on the unknown coefficients $\mathbf{u} = (u_1, ..., u_{2N-1})$, $\mathbf{y} = (y_1, ..., y_{2N-1})$, where N is the number of elements. That is we want to express the real-valued minimization problem as

$$\min_{\mathbf{y}, \mathbf{u} \in \mathbb{R}^{2N-1}} G(\mathbf{y}, \mathbf{u}) \text{ subject to } B\mathbf{y} = F\mathbf{u}. \tag{6}$$

Several techniques are combined to obtain (6). We start by recalling that $\|x\|^2_{L^2(\Omega)} = \langle x, x \rangle_{L^2(\Omega)}$ and $\langle f, g \rangle_{L^2(\Omega)} = \int_\Omega fg dx$. Taking advantage of that fact that $y_h, u_h, \overline{y}_d \in X_h^2$ which lets us write these functions as linear combinations of the basis functions ϕ_i of X_h^2. These linear combinations are simply summations which can be expressed in matrix form, while vectors $\mathbf{x}$ in $\mathbb{R}^M$ becomes column vectors.

The constraint of (5) written out is:

$$\int_0^1 \sum_{i=0}^{2N} v_i \phi_i' \sum_{j=0}^{2N} y_j \phi_j' dx = \int_0^1 \sum_{i=0}^{2N} v_i \phi_i \sum_{j=0}^{2N} u_j \phi_j dx,$$

since in particular $y_h, u_h, v \in V_h$ the coefficients of ϕ_0 and ϕ_{2N} for all functions become zero, letting us remove those indices from the summation limits. Together with the distributive property of summation as well as linearity of integration we get

$$\sum_{i=1}^{2N-1} \sum_{j=1}^{2N-1} v_i \int_0^1 \phi_i' \phi_j' dx y_j = \sum_{i=1}^{2N-1} \sum_{j=1}^{2N-1} v_i \int_0^1 \phi_i \phi_j dx u_j$$

This can be expressed in matrix form as $\mathbf{v^T} B\mathbf{y} = \mathbf{v}^T F\mathbf{u}$ and since this must hold for all $\mathbf{v}$ it's equivalent to $B\mathbf{y} = F\mathbf{u}$, where

$$B_{ij} = \int_0^1 \phi_i' \phi_j' dx, \quad F_{ij} = \int_0^1 \phi_i \phi_j dx$$

Turning our attention to the function to be minimized, in particular the term containing $u_h \in V_h$, we have

$$\|u_h\|^2_{L^2(\Omega)} = \langle u_h, u_h \rangle_{L^2(\Omega)} = \int_0^1 \sum_{i=0}^{2N} u_i \phi_i \sum_{j=0}^{2N} u_j \phi_j dx = \sum_{i=1}^{2N-1} \sum_{j=1}^{2N-1} u_i \int_0^1 \phi_i \phi_j dx u_j = \mathbf{u}^T F \mathbf{u}$$

with F the same as before. The first term, i.e. the term containing $y_h - \overline{y}_d$ requires some additional work since $\overline{y}_d$ might only be in X_h^2 and not in V_h. We start by splitting the term using the linearity of the inner product

$$\|y_h - \overline{y}_d\|^2_{L^2(\Omega)} = \langle y_h - \overline{y}_d, y_h - \overline{y}_d \rangle_{L^2(\Omega)} = \langle y_h, y_h \rangle_{L^2(\Omega)} + \langle \overline{y}_d, \overline{y}_d \rangle_{L^2(\Omega)} - 2\langle \overline{y}_d, y_h \rangle_{L^2(\Omega)}$$

Analogously as above we arrive at $\langle y_h, y_h \rangle_{L^2(\Omega)} = \mathbf{y}^T F \mathbf{y}$ and $\langle \overline{y}_d, \overline{y}_d \rangle_{L^2(\Omega)} = \overline{\mathbf{y}}_\mathbf{d}^T F^+ \overline{\mathbf{y}}_\mathbf{d}$, where F^+ symbolizes that we haven't thrown away the first and last index of the summation, corresponding to the value of $\overline{y}_d$ at the boundaries, which might be nonzero. We now treat the last term with $\overline{y}_d \in X_h^2$ and $y_h \in V_h$ in mind,

$$\langle \overline{y}_d, y_h \rangle_{L^2(\Omega)} = \int_0^1 \sum_{i=0}^{2N} (\overline{y}_d)_i \phi_i \sum_{j=1}^{2N-1} y_j \phi_j dx = \int_0^1 \left[((\overline{y}_d)_0 \phi_0 + (\overline{y}_d)_{2N} \phi_{2N}) \sum_{j=1}^{2N-1} y_j \phi_j + \sum_{i=1}^{2N-1} (\overline{y}_d)_i \phi_i \sum_{j=1}^{2N-1} y_j \phi_j \right] dx$$

$$= \sum_{j=1}^{2N-1} y_j \int_0^1 \phi_j \phi_0 (\overline{y}_d)_0 dx + \sum_{j=1}^{2N-1} y_j \int_0^1 \phi_j \phi_{2N} (\overline{y}_d)_{2N} dx + \sum_{i=1}^{2N-1} \sum_{j=1}^{2N-1} (\overline{y}_d)_i \int_0^1 \phi_i \phi_j dx y_j$$

$$= \mathbf{y}^T \mathbf{b}_0 + \mathbf{y}^T \mathbf{b}_{2N} + \mathbf{y}_d^T F \mathbf{y},$$

where

$$(b_0)_j = (\overline{y}_d)_0 \int_0^1 \phi_j \phi_0 dx, \quad (b_{2N})_j = (\overline{y}_d)_{2N} \int_0^1 \phi_j \phi_{2N} dx$$

All of our efforts gives us the following expression for G

$$G(\mathbf{y}, \mathbf{u}) = \frac{1}{2}\left(\mathbf{y}^T F \mathbf{y} + \overline{\mathbf{y}}_\mathbf{d}{}^T F^+ \overline{\mathbf{y}}_\mathbf{d} - 2(\mathbf{y}^T \mathbf{b}_0 + \mathbf{y}^T \mathbf{b}_{2N} + \mathbf{y}_d^T F \mathbf{y})\right) + \frac{\alpha}{2}\left(\mathbf{u}^T F \mathbf{u}\right)$$

4.3 System

The method of Lagrange multipliers tells us that the solution of (6) is a critical point of the Lagrange function

$$\mathcal{L}(\mathbf{y}, \mathbf{u}, \lambda) = G(\mathbf{y}, \mathbf{u}) - \lambda^T (B\mathbf{y} - F\mathbf{u})$$

where $\lambda \in \mathbb{R}^{2N-1}$ is the vector of Lagrange multipliers. In order to find the critical points of $\mathcal{L}$ we calculate the gradient with respect to the different variables, taking advantage of the fact that F and B is symmetric, as follows,

$$\nabla_\mathbf{y} \mathcal{L} = F\mathbf{y} - B\lambda - (\mathbf{b}_0 + \mathbf{b}_{2N} + F\mathbf{y}_d)$$

$$\nabla_\mathbf{u} \mathcal{L} = \alpha F \mathbf{u} + F\lambda$$

$$\nabla_\lambda \mathcal{L} = B\mathbf{y} - F\mathbf{u}$$

We find the critical point by setting all the gradients equal to zero and solving for $\mathbf{y}$, $\mathbf{u}$ and λ. In order to solve this numerically we up a $3(2N-1)$ by $3(2N-1)$ matrix, consisting of blocks made up of F, B and αF such that they correspond with the vector of unknowns $\left[\mathbf{y}^T, \mathbf{u}^\mathbf{T}, \lambda^T\right]^T$ and set this equal to a vector made up of $\mathbf{b}_0 + \mathbf{b}_{2N} + F\mathbf{y}_d$ in its first $2N-1$ entries followed by $2(2N-1)$ zeros.

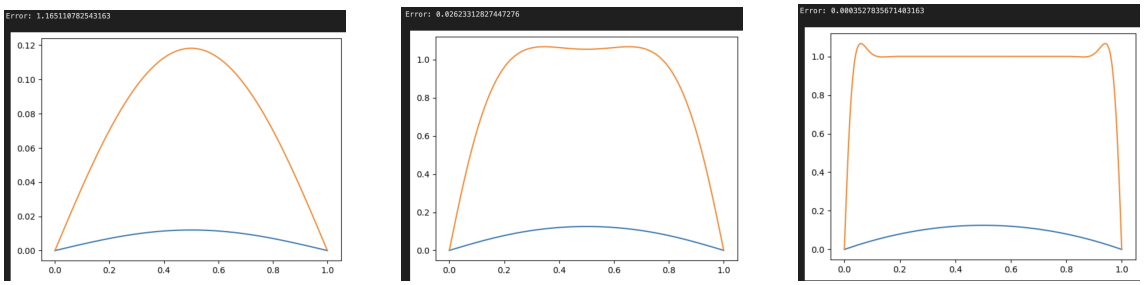
4.4 Properties of solutions

In the figures 4, 5 and 6 we present y_h in blue and u_h in orange for different values of the cost parameter α, on a partition of 100 elements.

We observe that the optimal control u_h, and the optimal state y_h for $\alpha = 1$ is unsolvable, as it is then "infinitely" expensive to control the temperature, and we therefore do not apply any control and just let the original temperature be. For other large values of α, such as $0,1$ and $0,01$, the control is still very minimized, and the optimal profile therefore misses the desired one by quite some distance. This makes a lot of sense in the context of (5), as the control has much less influence on the subsequent y_h.

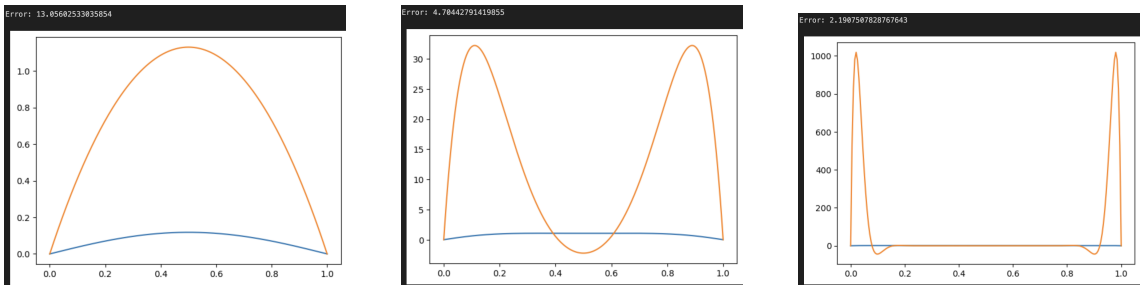
We can then see that as we lower the cost in 5 and 6, our ability to control the temperature profile increases, that is the optimal control u_h, and the difference between the desired profile and the actual optimal one decreases significantly, exactly as one would expect.

5 is the only one that is not part of $H_0^1(\Omega)$, and we observe in this case that y_h is still forced to be in $H_0^1(\Omega)$ due to the problem constraints which leads to noticeable errors and indicates that even when the control is less penalized, the match between the desired state and the solution space is an important factor for achieving a good approximation.



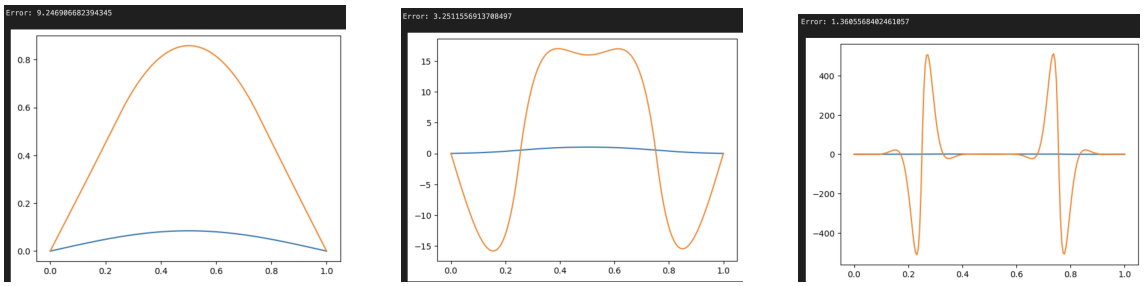
(a) $\alpha = 10^{-1}$ (b) $\alpha = 10^{-4}$ (c) $\alpha = 10^{-7}$

Figure 4: $y_d = \frac{1}{2}x(1-x)$



(a) $\alpha = 10^{-1}$ (b) $\alpha = 10^{-4}$ (c) $\alpha = 10^{-7}$

Figure 5: $y_d = 1$



(a) $\alpha = 10^{-1}$ (b) $\alpha = 10^{-4}$ (c) $\alpha = 10^{-7}$

Figure 6: $y_d = \begin{cases} 1 & \text{if} \quad x = [\frac{1}{4}, \frac{3}{4}], \\ 0 & \text{if otherwise} \end{cases}$

5 Conclusion

In this paper, we have developed a finite element method for solving the 1 dimensional Poisson equation, and analyzed its accuracy by deriving error bounds and verifying them by numerically testing the expected convergence rates, which led us to conclude that the method was $\mathcal{O}(h^2)$ in the H^1-norm and $\mathcal{O}(h^3)$ in the L^2-norm. We then proceeded to extending the framework to an optimal control problem in which the Poisson equation shows up as a constraint of a minimization problem. Through numerical experiments, we then demonstrated how the cost parameter α decides how much the temperature can be controlled towards the desired profile y_d. As expected, we observed that when the cost of heating and cooling α is small, more control is applied and the actual temperature gets closer to the desired state. This proved that a FE approach doesn't just provide a way to solve the Poisson equation, but also can be used as a tool for handling PDE-constrained optimization problems.