Solving a 1D Poisson Equation with the Galerkin Finite Element Method

Steinnes L.

Institute of Physics, University of Oslo

lasse.steinnes@fys.uio.no

November 9, 2020

Abstract

The Galerkin finite element method is applied to a 1D poisson equation. P2-elements are used and the linear system is assembled from reference element matrices and element vectors. The assembly of finite elements correctly reproduces the exact solution. The convergence rate is 3.5 up to a certain threshold for element step size h $(1/N_e)$ and number of elements N_e. Not surprisingly, P2-elements performs best for a curved exact solution.

I Introduction

There are two main ways to solve a partial differential equation (PDE) numerically. One is finite difference methods, using numerical approximations to the derivative. Another solution strategy is the finite element method, in which interpolation with finite basis functions are used to approximate the unknown target function u. In this instance, the problem is reduced to solving a linear system.

This paper will explain how to apply the finite element Galerkin (projection) method on a 1D Poisson equation on the form

$$-\frac{\partial^2 u}{\partial x^2} = 2x - 1, \qquad x \in \Omega = [0, 1], \tag{1}$$

accompanied by the boundary conditions (BCs)

$$u'(0) = C, \tag{2}$$

and

$$u(1) = D, \tag{3}$$

where C and D are scalar constants.

The paper is laid out by first presenting the underlying methods and theory behind P2-elements and how to assemble the resulting linear system from reference elements. Results are then presented for specific C and D values, with relevant benchmarks. Thereafter, a brief discussion and concluding remarks are given.

All relevant programmes are available https://github.com/lasse-steinnes/IN5270/tree/master/FEM1D_project.

II Theory and Methods

The theory section is mainly based upon Xing Cai's lecture notes on FEM in the course Numerical Methods for Partial Differential Equation (IN5270) at UiO [1] and the online resources provided by Hans Petter Langtangen [2]. For more supplementary literature on FEM-methods, see Langtangen and Mardal [3].

Assume u can be approximated with piecewise quadratic basis functions φ_j (P2-elements), so that

$$u(x) \approx \sum_{j \in I_s} c_j \varphi_j(x), \tag{4}$$

where $I_s = \{0, 1, ..., N = N_n - 1\}$ and N_n is the number of mesh points x_i, also called nodes. If $|\varphi_j| = 1$, the the $c_j = u(x_j)$ at node j.

The residual in the approximation becomes

$$\begin{aligned} \mathscr{L}(u) = u''(x) + 2x - 1 &= \frac{\partial^2}{\partial x^2}\left(\sum_{j \in I_s} c_j \varphi_j(x)\right) + f(x) \qquad (5) \\ &= \sum_{j \in I_s} c_j \varphi_j''(x) + f(x) \overset{\text{exact}}{=} 0. \qquad (6) \end{aligned}$$

The Galerkin method requires the residual to be orthogonal to all basis functions, so that

$$(u'' + f, \varphi_i) = 0 \quad \forall \;\; \varphi_i \in \text{Span}(\varphi_0, ..., \varphi_{N_{n-1}}), \tag{7}$$

which can be expressed by

$$(u'', \varphi_i) = -(f, \phi_i). \tag{8}$$

Thus, the (strong) variational formulation becomes

$$\sum_{j\in I_s} (\varphi_j'' \varphi_i) c_j = -(f, \phi_i) \quad \forall \;\; i \in I_s. \tag{9}$$

By applying integration by parts (IBP) on the domain Ω on the left hand side, yields

$$\left[u'\varphi_i\right]_0^1 - (u', \varphi_i) = \tag{10}$$

$$\left[u'(1)\varphi_i(1) - u'(0)\varphi_i(0)\right] - u'(0)\varphi(0). \tag{11}$$

Here $u'(0) = C$ from the Neumann BC (eq. 2). Note that $u'(1)\varphi_i(1) \neq 0$. However, in the modification of the linear system, to be presented later, setting $b_N = D$ erases this contribution. The value D comes from the Dirichlet BC (eq. I). Hence, applying IBP to eq. 9 results in the weak variational formulation

$$(u', \varphi_i) = \sum_{j\in I_s} \left(\int_0^1 \varphi_j' \varphi_i' dx \right) c_j \tag{12}$$

$$= \int_0^1 f(x)\varphi_i(x) dx - C\varphi_i(0). \tag{13}$$

Only $\varphi_0 \neq 0$ at $x = 0$. Thus C contributes to the equations only for $i = 0$. The unknown coefficients c_i associated to each basis function can then be found by solving the linear system $Ac = b$, in which

$$A_{i,j} = \int_0^1 \varphi_j' \varphi_i' dx \tag{14}$$

$$b_i = \int_0^1 f(x)\varphi_i(x) dx - C\varphi_i(0). \tag{15}$$

Now, instead of operating on the entire physical domain $\Omega = x \in [0, 1]$, one can operate on each element domain $\Omega_e = x \in [x_l, x_r]$, and make the mapping $[x_l, x_r] \to [-1, 1]$. Then

$$x = x_m + \frac{1}{2} hX, \qquad X \in [-1, 1], \tag{16}$$

where

$$h = x_r - x_l \quad \text{and} \quad x_m = \frac{x_l + x_r}{2}. \tag{17}$$

In this way, the element basis becomes equal for all elements, and one only computes the inner products of basis functions which is non-zero. In addition, the integration domain is the same for all elements. Thus, elementwise, or cellwise, calculation on a reference domain makes computations efficient and the calculations easier.

Assume a uniform mesh is used for the nodes. This mesh is then split into N_e elements, where each element consists of a given number of local nodes $r \in I_d$. For P2-elements $I_d = \{0,1,2\}$, the Lagrange interpolation polynomials for local nodes becomes

$$\tilde{\varphi}_0(X) = \frac{1}{2}(X-1)X, \tag{18}$$

$$\tilde{\varphi}_1(X) = 1 - X^2, \tag{19}$$

$$\tilde{\varphi}_2(X) = \frac{1}{2}(X+1)X. \tag{20}$$

A reference element can then be described by

$$\tilde{A}_{r,s}^{(e)} = \int_{\Omega^{(e)}} \varphi'_{q(e,r)}(x)\varphi'_{q(e,s)}(x)dx \tag{21}$$

$$= \int_{-1}^{1} \varphi'_{q(e,r)}(X)\varphi'_{q(e,s)}(X)\det(J)dX, \tag{22}$$

where $\det(J) = h/2$ in 1D from eq. 16, and the global nodes are described by $i = q(e,r)$. Applying the same transformation to b, one can assemble the element matrices and vectors to the physical domain as follows

$$A_{q(e,r),q(e,s)} = A_{q(e,r),q(e,s)} + \tilde{A}_{r,s}^{(e)}, \tag{23}$$

$$b_{q(e,r)} = b_{q(e,r)} + \tilde{b}_r^{(e)}. \tag{24}$$

In the case of P2-elements, to compute the element matrix on the reference domain, one need

$$\tilde{\varphi}_0{}'(X) = X - \frac{1}{2}, \tag{25}$$

$$\tilde{\varphi}_1{}'(X) = -2X, \tag{26}$$

$$\tilde{\varphi}_2{}'(X) = X + \frac{1}{2}. \tag{27}$$

From the chain rule, then

$$\frac{d\tilde{\varphi}_r}{dx} = \frac{d\varphi_i}{dX}\frac{dX}{dx} = \frac{2}{h}\frac{d\varphi_i}{dX}. \tag{28}$$

Using eq. 28, one can make the transition in 21, and compute the element matrix. For $\tilde{A}^{(e)}_{0,0}$

$$\tilde{A}^{(e)}_{0,0} = \frac{2}{h}\int_{-1}^{1}(X - \frac{1}{2})(X - \frac{1}{2})dX = \frac{7}{3h}. \tag{29}$$

The same procedure is followed to compute the other $\tilde{A}^{(e)}_{i,j}$ elements, so that

$$\tilde{A}^{(e)} = \frac{1}{3h}\begin{bmatrix} 7 & -8 & 1 \\ -8 & 16 & -8 \\ 1 & -8 & 7 \end{bmatrix}, \quad e = (1, ..., N_e - 2). \tag{30}$$

For $\tilde{b}^{(e)}$ $(e = (1, ..., N_e - 2))$, the vector elements becomes

$$\tilde{b}^{(e)}_0 = \int_{-1}^{1} f(x(X))\tilde{\varphi}_0(X)dX = -\frac{h^2}{6} + \frac{h}{3}x_m - \frac{h}{6}, \tag{31}$$

$$\tilde{b}^{(e)}_1 = \int_{-1}^{1} f(x(X))\tilde{\varphi}_1(X)dX = \frac{4h}{3}x_m - \frac{2h}{3}, \tag{32}$$

$$\tilde{b}^{(e)}_2 = \int_{-1}^{1} f(x(X))\tilde{\varphi}_2(X)dX = \frac{h^2}{6} + \frac{h}{3}x_m - \frac{h}{6}. \tag{33}$$

$$\tag{34}$$

The boundary conditions have to be considered in the first and last element matrix. The Neumann condition only affects $\tilde{b}^{e=0}_0 = \tilde{b}^{(e)}_0(x_m) - C$, where x_m is the midpoint of the first element. The Dirichlet condition however, must be taken into account, by both altering $\tilde{A}^{(e=N_e)}$ and $\tilde{b}^{e=N_e}$. The extended matrix becomes

$$\tilde{A}\tilde{b}^{(e=N_e)} = \frac{1}{3h}\begin{bmatrix} 7 & -8 & 1 & \tilde{b}^{(e)}_0 \\ -8 & 16 & -8 & \tilde{b}^{(e)}_1 \\ 0 & 0 & 3h & D \end{bmatrix}. \tag{35}$$

This ensures $\tilde{c_N} = D$.

The element matrices and vectors are assembled in the class `FEM-P2-solver` available at https://github.com/lasse-steinnes/IN5270/tree/master/FEM1D_ project. The PDE problem (eq. 1) is then solved by finding the unknown c, and representing u by the chosen basis functions.

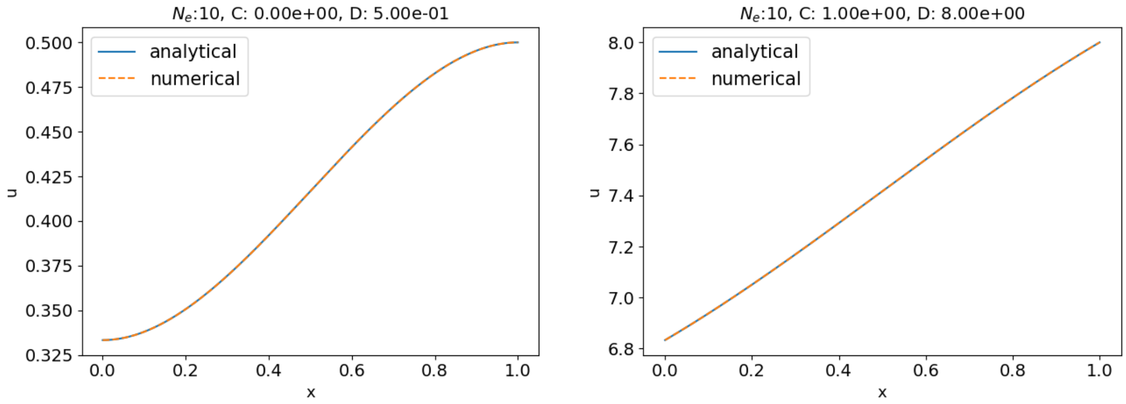
II.I Benchmarks

The numerical solution is compared with it's analytical solution

$$u_e(x) = -\frac{1}{3}x^3 + \frac{1}{2}x^2 + Cx + \left(D - C - \frac{1}{6}\right), \tag{36}$$

which is derived in sec. VI). In addition, a convergence test is made for the two cases $C_1 = 0.0, D_1 = 0.5$ and $C_2 = 1.0, D - 2 = 8.0$. For thus purpose, the L2-norm of $e(x) = u_e(x) - u(x)$ is used.

III Results



(a) Case 1

(b) Case 2

Figure 1: **Finite Element solution of a 1D poisson Equation**: Two different cases are presented, where both numerical solutions are in agreement with the analytical ones. **Case 1:** $C = 0.0$, $D = 0.5$. **Case 2)** $C = 1$, $D = 8$. C is the Neumann Boundary condtion and D is the Dirichlet Boundary Condition.

The results from the convergence tests are presented in tab. 1-2.

Table 1: **Benchmarks FEM Case 1**: Convergence rate results for case 1 ($C : 0.0, D : 0.5$). Parameters: h - Step size of an element e, L2- l2-norm, r - convergence rate.

h	L2	r
2.00×10^{-1}	4.11×10^{-3}	–
5.00×10^{-2}	3.21×10^{-5}	3.5×10^{0}
1.25×10^{-2}	2.51×10^{-7}	3.5×10^{0}
3.13×10^{-3}	1.96×10^{-9}	3.5×10^{0}
7.81×10^{-4}	1.53×10^{-11}	3.5×10^{0}
1.95×10^{-4}	1.30×10^{-12}	1.78×10^{0}
4.88×10^{-5}	2.60×10^{-12}	-4.98×10^{-1}

Table 2: **Benchmarks FEM Case 2**: Convergence rate results for case 1 ($C : 1.0, D : 8.0$). Parameters: h - Step size of an element e, L2- l2-norm, r - convergence rate.

h	L2	r
2.00×10^{-1}	4.11×10^{-3}	–
5.00×10^{-2}	3.21×10^{-5}	3.5×10^{0}
1.25×10^{-2}	2.51×10^{-7}	3.5×10^{0}
3.13×10^{-3}	1.96×10^{-9}	3.5×10^{0}
7.81×10^{-4}	2.06×10^{-11}	3.29×10^{0}
1.95×10^{-4}	2.7×10^{-12}	-2.07×10^{-1}
4.88×10^{-5}	5.46×10^{-12}	-4.97×10^{-1}

IV Discussion

From the result section (sec. III), it is clear that the numerical solution using FEM is in agreement with the analytical solution (fig. 1). However, there are some trade-off between number of elements (N_e) chosen and the absolute error (tab. 1-2). Both cases have a convergence rate of 3.5, suggesting that with P2-elements, one get O($n^{3.5}$) accuracy. The trade-off occurs around $h = 1.95 \times 10^{-4}$ for case 1, whereas $h = 7.81 \times 10^{-4}$ for case 2. For smaller h values than this ($h = 1/N_e$), the error increases again. Since the trade-off occurs faster in case 2 compared to case 1, P2-elements are more suitable for a curved exact solution, rather than linear ones.

V Conclusion

The finite element method is an alternative to using finite difference schemes when solving PDEs numerically. Here, the Galerkin finite element method is applied to a 1D poisson equation (eq. 1).

As seen in fig.1, the assembly of finite elements correctly reproduces the exact solution. The convergence rate is 3.5 up to a certain threshold for element step size h $(1/N_e)$ and number of elements N_e. The threshold for the lowest value of h which gives an optimal numerical approximation depends on the given boundary conditions. Not surprisingly, P2-elements performs best for a curved exact solution.

References

[1] Cai X.; Langtangen H.P. IN5270 Resources, 2020. Accessed: October 2020.

[2] Langtangen H.P. Introduction to finite element methods, 2020. Accessed: October 2020.

[3] Kent-Andre Mardal Hans Petter Langtangen. *Introduction to Numerical Methods for Variational Problems*. Springer, 2019.

VI Appendix

The analytical solution to eq. 1 is derived below.

For a non-zero right hand side then $u_e = u_h + u_p$ where u_h is the homogoneous solution and u_p is the particular solution.

Integrating up $u'' = 0$ wrt. x yields

$$u_h = Ax + b. \tag{37}$$

For the particular solution, make a guess on the form

$$u_p = \alpha x^3 + \beta x^2 + \gamma x + \phi. \tag{38}$$

Derivation wrt. x yields

$$u_p'' = 6\alpha x + 2\beta \overset{\text{def}}{=} -2x + 1. \tag{39}$$

Thus $\alpha = -1/3$ and $\beta = 1/2$.

Thus

$$u_e = u_h + u_p = -\frac{1}{3}x^3 + \frac{1}{2}x^2 + \hat{A}x + \hat{B}. \tag{40}$$

Applying the boundary conditions yields

$$u_e = -\frac{1}{3}x^3 + \frac{1}{2}x^2 + Cx + \left(D - C - \frac{1}{6}\right). \tag{41}$$