

# ME 280A Finite Element Analysis

## HOMEWORK 3: Potential and efficient solution techniques

---

Xin Wang

October 19, 2014

## 1 INTRODUCTION

The objective of this project is to solve a 1-D differential equation using linear elements.

$$\begin{aligned}
 \frac{d}{dx}(A_1(x)\frac{du}{dx}) &= f(x) \\
 f(x) &= 256\sin(\frac{3\pi kx}{4})\cos(16\pi x) \\
 A_1(x) &= \text{piecewise} \\
 L &= 1 \\
 u(0) &= 0 \\
 A_1(L)\frac{du}{dx}(L) &= 1
 \end{aligned} \tag{1}$$

$A_1$  is a piece wise function, defined as bellow:

$$\begin{aligned}
 \text{For } 0.0 \leq x < 0.1: \quad A_1 &= 2.0 \\
 \text{For } 0.1 \leq x < 0.2: \quad A_1 &= 2.5 \\
 \text{For } 0.2 \leq x < 0.3: \quad A_1 &= 1.25 \\
 \text{For } 0.3 \leq x < 0.4: \quad A_1 &= 0.25 \\
 \text{For } 0.4 \leq x < 0.5: \quad A_1 &= 4.0 \\
 \text{For } 0.5 \leq x < 0.6: \quad A_1 &= 1.75 \\
 \text{For } 0.6 \leq x < 0.7: \quad A_1 &= 0.5 \\
 \text{For } 0.7 \leq x < 0.8: \quad A_1 &= 0.75 \\
 \text{For } 0.8 \leq x < 0.9: \quad A_1 &= 3.25 \\
 \text{For } 0.9 \leq x \leq 1.0: \quad A_1 &= 1.0
 \end{aligned} \tag{2}$$

## 2 ANALYTICAL SOLUTION

The general solution for this equation on every segment can be found after two integrations:

$$\begin{aligned}
 A_1(x)\frac{du}{dx} &= \int 256\sin(\frac{3\pi kx}{4})\cos(16\pi x)dx \\
 &= \int 128[\sin(\frac{67\pi x}{4}) - \sin(\frac{61\pi x}{4})] \\
 &= \frac{512(67\cos(\frac{61\pi x}{4}) - 61\cos(\frac{67\pi x}{4}))}{4087\pi} + C1
 \end{aligned} \tag{3}$$

$$\frac{du}{dx} = \frac{1}{A_1(x)} \left( \frac{512(67\cos(\frac{61\pi x}{4}) - 61\cos(\frac{67\pi x}{4}))}{4087\pi} + C1 \right) \quad (4)$$

Integrating  $\frac{du}{dx}$  and applying the boundary condition at  $x=0$ , we obtain the analytical solution for the differential equation:

$$u(x) = \frac{\frac{137216\sin(\frac{61\pi x}{4})}{61\pi} - \frac{124928\sin(\frac{67\pi x}{4})}{67\pi}}{4087\pi} + \frac{C1x}{A_1(x)} + C2 \quad (5)$$

At the interface of each segments  $x_N$  ( $N=1, \dots, 10$ ), the function  $u$  and its derivation is continuous, i.e.,

$$\begin{aligned} u(x_N^+) &= u(x_N^-) \\ A1(x_N^+) \frac{du}{dx}(x_N^+) &= A1(x_N^1) \frac{du}{dx}(x_N^-) \end{aligned} \quad (6)$$

In particular, at the left endpoint,  $u(0) = u_0 = 0$  and at the right end point,  $A1(L) \frac{du}{dx}(L) = A1(L) (\frac{du}{dx})_N = 1$ .

Applying these boundary conditions and continuation conditions at the interfaces, we can solve the equation system and obtain the analytical solution:

$$\begin{aligned} \text{For } 0.0 \leq x < 0.1: \quad C_1 &= 1.1692 \\ \text{For } 0.1 \leq x < 0.2: \quad C_1 &= 1.1692 \\ \text{For } 0.2 \leq x < 0.3: \quad C_1 &= 1.1692 \\ \text{For } 0.3 \leq x < 0.4: \quad C_1 &= 1.1692 \\ \text{For } 0.4 \leq x < 0.5: \quad C_1 &= 1.1692 \\ \text{For } 0.5 \leq x < 0.6: \quad C_1 &= 1.1692 \\ \text{For } 0.6 \leq x < 0.7: \quad C_1 &= 1.1692 \\ \text{For } 0.7 \leq x < 0.8: \quad C_1 &= 1.1692 \\ \text{For } 0.8 \leq x < 0.9: \quad C_1 &= 1.1692 \\ \text{For } 0.9 \leq x \leq 1.0: \quad C_1 &= 1.1692 \end{aligned} \quad (7)$$

$$\begin{aligned}
\text{For } 0.0 \leq x < 0.1 : C_2 &= 0 \\
\text{For } 0.1 \leq x < 0.2 : C_2 &= 0.0101 \\
\text{For } 0.2 \leq x < 0.3 : C_2 &= -0.0964 \\
\text{For } 0.3 \leq x < 0.4 : C_2 &= 1.4040 \\
\text{For } 0.4 \leq x < 0.5 : C_2 &= 0.2742 \\
\text{For } 0.5 \leq x < 0.6 : C_2 &= 0.1166 \\
\text{For } 0.6 \leq x < 0.7 : C_2 &= -0.8391 \\
\text{For } 0.7 \leq x < 0.8 : C_2 &= -0.2384 \\
\text{For } 0.8 \leq x < 0.9 : C_2 &= 0.7997 \\
\text{For } 0.9 \leq x \leq 1.0 : C_2 &= 0.0931
\end{aligned} \tag{8}$$

### 3 FINITE ELEMENT METHOD

The first step of FEM is to derive the weak form of the differential equation:

Find  $u$ ,  $u|_{\Gamma} u = d$ , such that  $\forall v, v|_{\Gamma} u = 0$

$$\int_{\Omega} \frac{dv}{dx} A_1 \frac{du}{dx} dx = \int_{\Omega} f v dx + t v|_{\Gamma} \tag{9}$$

We approximate the real solution  $u$  by

$$u(x) = \sum_{j=1}^N a_j \phi_j(x) \tag{10}$$

and we choose the test function  $v$  with the same approximation functions

$$v(x) = \sum_{i=1}^N b_i \phi_i(x) \tag{11}$$

where  $N$  is the number of degree of freedom (number of nodes).

Then the equation becomes:

$$\int_{\Omega} \frac{d}{dx} \left( \sum_{j=1}^{N+1} a_j \phi_j(x) \right) A_1 \frac{d}{dx} \left( \sum_{i=1}^{N+1} b_i \phi_i(x) \right) dx = \int_{\Omega} f \left( \sum_{i=1}^{N+1} b_i \phi_i(x) \right) dx + \left( \sum_{i=1}^{N+1} b_i \phi_i(x) t \right) |_{\Gamma}, \forall b_i \tag{12}$$

We can regroup the terms into:

$$\sum_{i=1}^{N+1} b_i \int_{\Omega} \left( \sum_{j=1}^{N+1} a_j \frac{d}{dx} \phi_j(x) A_1 \frac{d}{dx} \phi_i(x) \right) dx = \sum_{i=1}^{N+1} b_i \int_{\Omega} f \phi_i(x) dx + \sum_{i=1}^{N+1} b_i (\phi_i(x) t) |_{\Gamma_t} \quad (13)$$

As the equation should be valid for any  $b_i$ , we obtain the matrix system to solve:

$$\begin{aligned} K_{ij} &= \int_{\Omega} \frac{d}{dx} \phi_j(x) A_1 \frac{d}{dx} \phi_i(x) dx \\ R_i &= \int_{\Omega} f \phi_i(x) dx + \phi_i(x) t |_{\Gamma_t} \\ K a &= R \end{aligned} \quad (14)$$

In this homework, piece-wise linear, quadratic or cubic basis functions are used. The numerical computation is carried over the corresponding master elements and mapped to the global elements.

Shape functions are defined for elements with different polynomial orders (figure ??).

Linear shape functions are:

$$\begin{aligned} \hat{\phi}_1 &= \frac{1-\xi}{2} \\ \hat{\phi}_2 &= \frac{1+\xi}{2} \end{aligned} \quad (15)$$

and

$$\begin{aligned} \frac{d\hat{\phi}_1}{d\xi} &= -\frac{1}{2} \\ \frac{d\hat{\phi}_2}{d\xi} &= \frac{1}{2} \end{aligned} \quad (16)$$

Corresponding global coordinate  $x$  can be calculate  $x$  from  $\xi$ :

$$x = \sum \chi_i \hat{\phi}_i \quad (17)$$

where  $\chi_i$  are the coordinates of the nodes in the global element.

The method to compute the matrices  $K$  and  $R$  element by element and assembling procedures are the same as in homework 1. The difference for higher order elements is the size of  $K_e$  and  $R_e$ .

The data structure used for store the matrix K is a 3xN matrix where only three numbers in the 2x2 element K matrix are stores.

$$ke\_table = \{k_{11}^1, k_{12}^1, k_{22}^1\} \quad (18)$$

$$k_{11}^2, k_{12}^2, k_{22}^2\} \quad (19)$$

## 4 PRECONDITIONED CONJUGATE-GRADIENT SOLVER

There are two approaches to solve the equation for  $[K]a = R$ , direct and iterative. The number of operation in direct solvers are  $O(N^3)$ . They are employed when the number of unknowns are not very large, and there are multiple load vectors. Iterative solver are preferred when the systems are very large. Conjugate gradient type iterative techniques need  $O(N^2)$  operations.

We define a potential(eq21) whose derivative is the residual of the equation to solve. So the minimizer of the potential  $\Pi$  is the solution to the discrete system.

$$\nabla \Pi = [K]\{a\} - \{R\} \quad (20)$$

$$\Pi = \frac{1}{2} \{a\}^T [K] \{a\} - \{a\}^T \{R\} \quad (21)$$

$$\begin{aligned} \{r\}^i &= -\nabla \Pi = \{R\} - [K]\{a\}^i \\ \{a\}^{i+1} &= \{a\}^i + \lambda^i \{r\}^i \end{aligned} \quad (22)$$

The idea of steepest descent is to find this minimizer of potential by going to the direction of the gradient of the potential, i.e.  $\{a\}^{i+1} = \{a\}^i + \lambda^i \{z\}^i$ , where  $\{z\}^i = \{r\}^i + \theta^i \{z\}^{i-1}$ .

The coefficient  $\theta$  is chosen so that  $z$  is  $[K]$  conjugate to  $\{z\}^{i-1}$ , i.e.,

$$\begin{aligned} \{z\}^{T,i} [K] \{z\}^{i-1} &= 0 \\ \theta^i &= -\frac{\{r\}^{T,i} [K] \{z\}^{i-1}}{\{z\}^{T,i-1} [K] \{z\}^{i-1}} \end{aligned} \quad (23)$$

Looking for  $\lambda$  to minimize the potential, we force  $\frac{\partial \Pi}{\partial \lambda^i} = 0$  and solve for  $\lambda$ :

$$\lambda^i = \frac{\{z\}^{T,i} \{r\}^i}{\{z\}^{T,i} [K] \{z\}^i} \quad (24)$$

#### 4.1 PRECONDITIONING OF [K]

The rate of convergence of the CG method is related to the condition number

$$\|\{a\} - \{a\}^i\|_K \leq \left(\frac{\sqrt{C([K])} - 1}{\sqrt{C([K])} + 1}\right)^i \|\{a\} - \{a\}^1\|_K \quad (25)$$

To reduce the condition number, preconditioning of [K] is done by forming the transformation of variables:

$$\begin{aligned} \{a\} &= [T]\{A\} \\ [\bar{K}] &= [T]^T [K] [T] \end{aligned} \quad (26)$$

Diagonal preconditioning is an inexpensive widely used, where [T] is defined from the diagonal of [K] as:

$$T_{ij} = \frac{1}{\sqrt{K_{ij}}}, i, j = 1, \dots, ndof \quad (27)$$

#### 4.2 ITERATION

Starting from an initial guess for  $\{a\}^0$ , we can calculate the residual defined as in equation 28.

$$\begin{aligned} \{r\}^0 &= -\nabla \Pi = \{R\} - [K]\{a\}^0 \\ \{z\}^0 &= \{r\}^0 \\ \lambda^0 &= \frac{\{z\}^{T,0} \{r\}^0}{\{z\}^{T,0} [K] \{z\}^0} \\ \{a\}^1 &= \{a\}^0 + \lambda^0 \{z\}^0 \end{aligned} \quad (28)$$

For iteration  $i > 0$ ,

$$\begin{aligned} \{r\}^i &= \{R\} - [K]\{a\}^i \\ \theta^i &= -\frac{\{r\}^{T,i} [K] \{z\}^{i-1}}{\{z\}^{T,i-1} [K] \{z\}^{i-1}} \\ \{z\}^i &= \{r\}^i + \theta^i \{z\}^{i-1} \\ \lambda^i &= \frac{\{z\}^{T,i} \{r\}^i}{\{z\}^{T,i} [K] \{z\}^i} \\ \{a\}^{i+1} &= \{a\}^i + \lambda^i \{z\}^i \end{aligned} \quad (29)$$

and compute the error for each iteration as:

$$e^i = \frac{\|\{a\}^i - \{a\}^i\|_K}{\|\{a\}^i\|_K} = \frac{|\lambda^i| \|\{z\}\|_K}{\|\{a\}_K^i\|} \quad (30)$$

If  $e^i < \tau$  stop, otherwise reiterate.

## 5 ERROR CALCULATIONS

The error is defined as

$$\begin{aligned} e^N &= \frac{\|u - u^N\|_{A_1(\Omega)}}{\|u\|_{A_1(\Omega)}} \\ \|u\|_{A_1(\Omega)} &= \sqrt{\int_{\Omega} \frac{du}{dx} A_1 \frac{du}{dx} dx} = \sqrt{\int_{\Omega} A_1 \left(\frac{du}{dx}\right)^2 dx} \\ \|u - u^N\|_{A_1(\Omega)} &= \sqrt{\int_{\Omega} A_1 \left(\frac{d(u - u^N)}{dx}\right)^2 dx} \end{aligned} \quad (31)$$

To compute the error numerically, we calculate the two following quantities element by element, and then assemble them to obtain the overall error:

$$\begin{aligned} \|u\|_{A_1(\Omega)}^2 &= \int_{\Omega} A_1 \left(\frac{du}{dx}\right)^2 dx \\ \|u - u^N\|_{A_1(\Omega)}^2 &= \int_{\Omega} A_1 \left(\frac{d(u - u^N)}{dx}\right)^2 dx \\ &= A_1 \int_{\Omega} \left(\frac{du}{dx} - \frac{du_N}{dx}\right)^2 dx \\ &= A_1 \sum_{e=1}^{N_e} \int_{x_e}^{x_{e+1}} \left(\frac{du}{dx} - \frac{du_N}{dx}\right)^2 dx \\ &= A_1 \sum_{e=1}^{N_e} \int_{-1}^1 \left(\frac{du}{d\xi} - \frac{du_N}{d\xi}\right)^2 J d\xi \end{aligned} \quad (32)$$

## 6 POSTPROCESSING

The postprocessing is more complicated for higher order element. In order to plot the result, we calculate the value of the numerical solution for 10 points in each



P	$Ne_{opt}$	$N_{opt} = Ne_{opt} * P + 1$
1	1348	1349
2	94	189
3	34	103

Table 1: Minimum number of nodes needed for the error criteria

elements. For any given  $\xi$ , the corresponding value for the solution is calculated, as well as the global coordinate  $x$  for the given  $\xi$ :

$$\begin{aligned} u(x) &= u(x(\xi)) = \sum_{i=1}^{P+1} a_i^e \hat{\phi}_i \\ x(\xi) &= \sum x^e \hat{\phi}_i(\xi) \end{aligned} \quad (33)$$

For each element, we can also calculate the derivative of  $u(x)$ :

$$\frac{du}{dx} = \frac{d}{dx} \sum_{i=1}^{P+1} a_i \phi_i = \left( \frac{d}{d\xi} \sum_{i=1}^{P+1} a_i \hat{\phi}_i \right) \frac{d\xi}{dx} \quad (34)$$

## 7 RESULTS

Minimum number of elements and number of nodes to achieve to error criteria for different order of elements are tabulated bellow.  $N_{opt}$  drops dramatically with the order of the elements.

The numerical solution and the analytical solution are compared in figure ??.

### 7.1 RELATIONSHIP BETWEEN THE ERROR AND THE ELEMENT SIZE

Error estimate for FEM is:

$$e^N \leq Ch^{\gamma=\min(r-1,P)} \quad (35)$$

$r$  is smoothness, and  $P$  is the polynomial order.

From the error plotting, we can observe the linear relation between  $\log(e^N)$  and  $\log(h)$ ,

$$\gamma = 0.98, P = 1$$

$$\gamma = 1.98, P = 2$$

$$\gamma = 2.93, P = 3$$

(36)

Overall,  $\gamma$  is similar to  $P$ .

## 8 CONCLUSION

This homework extends the previous one to higher order of element. Having higher order element shape functions can reduce the number of degree of freedom needed for solving the problem by better capturing the fluctuations in the solution.