One D problem : Single variable : February 18, 2021

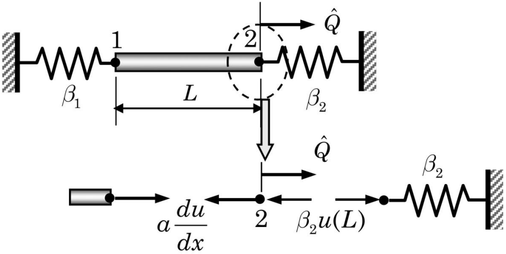
- A(u(x)) = f(x) in interval 0 <x<L   B(u) = g
- Consider the differential equation

$$-\frac{d}{dx}\left(k(x,u)\frac{du}{dx}\right) + b(x,u)\frac{du}{dx} + c(x,u)u = f(x) \quad 0 < x < L$$

$$\text{Boundary conditions} \qquad (1)$$

$$n_x k\frac{du}{dx} + \beta(x,u)(u - u_\infty) = \hat{Q} \quad or \quad u = \hat{u}$$

- Note that $n_x = -1, \beta = \beta_1$ at $x = x_a$ and $n_x = 1, \beta = \beta_2$ at $x = x_b$
- For a bar with a spring, $u_\infty = 0$ and we get the equation that the bar should be equal to the spring force βu. Or $-k\frac{du}{dx} - \beta_2 u = Q_2$ where Q2 is the extrenal force.



- Therefore we can keep it generally and nicely as:

$$
\begin{aligned}
A(u) = f \quad in \quad 0 < x < L \qquad B(u) = g \quad at \quad x = 0 or L \\
A = -\frac{d}{dx}\left(a\frac{d}{dx}\right) + \frac{d}{dx} + c. \quad B = n_x a\frac{d}{dx} + \beta, \quad g = \beta u_\infty + \hat{Q}
\end{aligned} \tag{2}
$$

- If a,b,c are functions of u then A and B become nonlienear.
- In heat a = kA, b = 0 and c = perimeter $.\beta$

4.2 Weak Formulation

Suppose that the domain $\Omega = (0, L)$ is divided into N line elements. A typical element from the collection of N elements is denoted as $\Omega^e = (x_a, x_b)$, where x_a and x_b denote the global coordinates of the end nodes of the line element. The weak form of Eq. (4.1.1) over the element can be developed as follows (see Section 3.2 for details):

$$
\begin{aligned}
0 &= \int_{x_a}^{x_b} \left(a \frac{dw_i^e}{dx} \frac{du_h^e}{dx} + b w_i^e \frac{du_h^e}{dx} + c w_i^e u_h^e - w_i^e f \right) dx - \left[w_i^e \left(a \frac{du_h^e}{dx} \right) \right]_{x_a}^{x_b} \\
&= \int_{x_a}^{x_b} \left[a(x,u) \frac{dw_i^e}{dx} \frac{du_h^e}{dx} + b(x,u) w_i^e \frac{du_h^e}{dx} + c(x,u) w_i^e u_h^e - w_i^e f(x) \right] dx \\
&\quad - \left\{ Q_a^e - \beta_a \left[u_h^e(x_a) - u_\infty^a \right] \right\} w_i^e(x_a) - \left\{ Q_b^e - \beta_b \left[u_h^e(x_b) - u_\infty^b \right] \right\} w_i^e(x_b)
\end{aligned}
\tag{4.2.1}
$$

where $w_i^e(x)$ is the ith weight function. The number of weight functions is equal to the number of unknowns in the approximation of u_h. The first line of Eq. (4.2.1) suggests that u is the primary variable and $Q = a(du/dx)$ is the secondary variable of the formulation. Using the mixed boundary condition in Eq. (4.1.2), we can express $a(du/dx)$ in terms of (Q_a^e, Q_b^e) and (u_∞^a, u_∞^b) as

$$
\begin{aligned}
-\left[a \frac{du_h^e}{dx} \right]_{x=x_a} &= Q_a^e - \beta_a \left[u_h^e(x_a) - u_\infty^a \right] \\
\left[a \frac{du_h^e}{dx} \right]_{x=x_b} &= Q_b^e - \beta_b \left[u_h^e(x_b) - u_\infty^b \right]
\end{aligned}
\tag{4.2.2}
$$

where (Q_a^e, Q_b^e) are the nodal values, (u_∞^a, u_∞^b) denote the values of the variable u_∞, and (β_a, β_b) denote certain physical parameters (e.g. film conductances) at the left and right ends of the element, respectively. When a node is in the

- If we use the discretisation then we will get

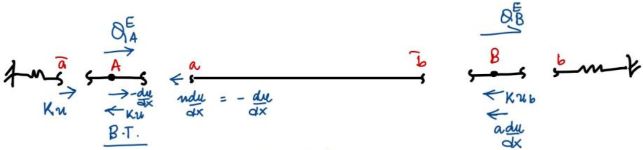
$$K(U)U = F \tag{3}$$

where

$$K_{ij}^e = \int_{x_a}^{x_b} \left[a(x, u_h^e) \frac{d\psi_i^e}{dx} \frac{d\psi_j^e}{dx} + b(x, u_h^e) \psi_i^e \frac{d\psi_j^e}{dx} + c(x, u_h^e) \psi_i^e \psi_j^e \right] dx$$
$$+ \beta_a \psi_i^e(x_a) \psi_j^e(x_a) + \beta_b \psi_i^e(x_b) \psi_j^e(x_b) \tag{4.3.3}$$
$$F_i^e = \int_{x_a}^{x_b} f(x) \psi_i^e \, dx + \beta_a u_\infty^a \psi_i^e(x_a) + \beta_b u_\infty^b \psi_i^e(x_b) + Q_a \psi_i^e(x_a) + Q_b \psi_i^e(x_b)$$

Note that the coefficient matrix $\mathbf{K}^e$ is a function of the unknown nodal values u_j^e, and it is an unsymmetric matrix when $b \neq 0$; when $b = 0$, $\mathbf{K}^e$ is a symmetric matrix. The term involving c is symmetric, independent of whether it depends on u and/or du/dx. Therefore, it is advisable to include nonlinear terms of the type $u\,(du/dx)$ in a differential equation as the c-term in the equation by writing it as $u\,(du/dx) = cu$, with $c = du/dx$; otherwise, it will be unsymmetric and convergence of the solution may become a problem. The coefficients involving β in $\mathbf{K}^e$ and $\mathbf{F}^e$ should be included only in elements that have end nodes with the convection type boundary condition. Example 4.3.1 provides more insight into the make-up of the coefficient matrix $\mathbf{K}^e$.

- The Boundary terms still confuses me. Especially the sign part. So here it is! Qe is the external force. F is the nodal force with equivalent parts. We actually get the external force from the internal Boundary terms!



- Suppose $\frac{du}{dx} = +ve$ (Tension) so

At a Internal force $= n\,a\frac{du}{dx} = -a\frac{du}{dx}$

At $\bar{a}$ Internal force $= Ku$

At A : $Q_A^E - Ku = -a\frac{du}{dx}$

At B : $Q_B^E - Ku = a\frac{du}{dx}$.

$$\left[w_i^e\left(a\frac{du_h^e}{dx}\right)\right]_{x_a}^{x_b}$$

$$= w_1^e(x_b)\left[a\frac{du_h^e}{dx}(x_b)\right] + w_1^e(x_a)\left[-\frac{a\,du_h^e(x_a)}{dx}\right]$$

$$= \left(Q_A^e - Ku_a\right) + \left(Q_A^B - Ku_b\right)$$

- Check problem from Reddy for nonlinear constraints etc.

- Direct iteration procedure
- Newton rhapson method

- We solve this system of equations using direct iteration, Picard iteration or method of successive substitutions

$$\mathbf{K}(\mathbf{U}^{(\mathbf{r-1})})\mathbf{U}^{\mathbf{r}} = \mathbf{F}(\mathbf{U}^{(\mathbf{r-1})}) \tag{4}$$

Box 4.4.1: Steps involved in the direct iteration scheme.

1. **Initial solution vector.** Assume an initial solution vector $\mathbf{U}^{(0)}$ such that it (a) satisfies the specified boundary conditions on $\mathbf{U}$ and (b) does not make $\mathbf{K}^e$ singular.
2. **Computation of K and F.** Use the latest known vector $\mathbf{U}^{(r-1)}$ ($\mathbf{U}^{(0)}$ during the first iteration) to evaluate $\mathbf{K}^e$ and $\mathbf{F}^e$, assemble them to obtain global $\mathbf{K}$ and $\mathbf{F}$, and apply the specified boundary conditions on the assembled system.
3. **Computation of $\mathbf{U}^{(r)}$.** Compute the solution at the rth iteration

$$\mathbf{U}^{(r)} = [\mathbf{K}(\mathbf{U}^{(r-1)})]^{-1}\mathbf{F}^{(r-1)}$$

4. **Convergence check.** Compute the residual

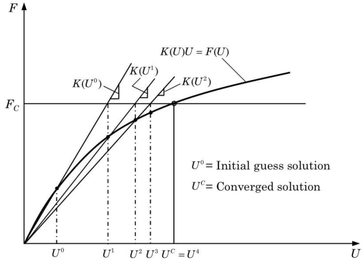
$$\mathbf{R}^{(r)} = \mathbf{K}(\mathbf{U}^{(r)})\mathbf{U}^{(r)} - \mathbf{F}^{(r)}$$

with the latest known solution and check if

$$\|\mathbf{R}^{(r)}\| \leq \epsilon \|\mathbf{F}^{(r)}\| \quad \text{or} \quad \|\mathbf{U}^{(r)} - \mathbf{U}^{(r-1)}\| \leq \epsilon \|\mathbf{U}^{(r)}\|$$

where $\|\cdot\|$ denotes the euclidean norm and ϵ is the convergence tolerance (read as an input). If the solution has converged, print the solution and move to the next "load" level or quit if it is the only or final load; otherwise, continue.

5. **Maximum iteration check.** Check if $r < itmax$, where $itmax$ is the maximum number of iterations allowed (read as an input). If *yes*, set $r \to r + 1$ and go to Step 2; if *no*, print a message that the iteration scheme did not converge and quit.



- The method can be accelerated by using a weighted average of solutions from the last two iterations

$$\mathbf{U}^{\mathbf{r}} = \mathbf{K}(\bar{\mathbf{U}})^{-\mathbf{1}}\mathbf{F}(\bar{\mathbf{U}})$$
$$\bar{\mathbf{U}} = \beta\mathbf{U}^{(\mathbf{r-2})} + (\mathbf{1}-\beta)\mathbf{U}^{(\mathbf{r-1})} \tag{5}$$

- Check Reddy 184 for some pages

- In NM we expand he residual vector $\mathbf{R}^{(\mathbf{r})}$ in Taylor series about a kown solution $\mathbf{U}^{(\mathbf{r-1})}$ to get

$$\mathbf{R}^{\mathbf{r}} = \mathbf{R}^{(\mathbf{r-1})} + \left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right)^{(\mathbf{r-1})} {}'\mathbf{U} + \mathbf{O}(\mathbf{h^2}) \tag{6}$$

where $\Delta U = U^r - U^{(r-1)}$

- And saying that R in the next iteration should be zero we get

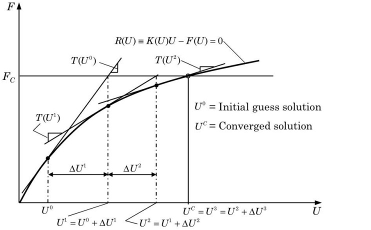
$$\begin{gathered}\left(\frac{\partial R}{\partial U}\right)^{(r-1)} \Delta U = -R^{(r-1)} \\ T^{(r-1)} U^r = -R^{(r-1)} + T^{(r-1)} U^{r-1}\end{gathered} \tag{7}$$

where T is the tangent matrix can be found at element level given as

$$\left(T_{IJ} = \frac{\partial R_I}{\partial U_J} = K_{IJ} + \sum_{m=1}^{N} \left(\frac{\partial K_{Im}}{\partial U_j} U_m\right) - \frac{\partial F_I}{\partial U_J}\right)^e \tag{8}$$

The force derivative is zero if it is not a function of the load

1. **Initial solution vector.** Assume an initial solution vector $\mathbf{U}^{(0)}$ such that: (a) it satisfies the specified boundary conditions on $\mathbf{U}$ and (b) it does not make $\mathbf{T}^e$ singular.
2. **Computation of T and R.** Use the latest known vector $\mathbf{U}^{(r-1)}$ ($\mathbf{U}^{(0)}$ during the first iteration) to: (a) evaluate $\mathbf{K}^e$, $\mathbf{F}^e$, $\mathbf{T}^e$, and $-\mathbf{R}^e = \mathbf{F}^e - \mathbf{K}^e\mathbf{U}^e$, (b) assemble $\mathbf{T}^e$ and $\mathbf{R}^e$ to obtain global $\mathbf{T}$ and $\mathbf{R}$, and (c) apply the specified *homogeneous* boundary conditions (since $\mathbf{U}^{(0)}$ already satisfies the actual boundary conditions) on the assembled system, $\mathbf{TU} = -\mathbf{R}$.
3. **Computation of $\mathbf{U}^{(r)}$.** Compute the solution increment at the rth iteration
$$\Delta\mathbf{U} = -[\mathbf{T}(\mathbf{U}^{(r-1)})]^{-1}\mathbf{R}^{(r-1)}$$
and update the total solution
$$\mathbf{U}^{(r)} = \mathbf{U}^{(r-1)} + \Delta\mathbf{U}$$
4. **Convergence check.** Compute the residual
$$\mathbf{R}^{(r)} = \mathbf{K}(\mathbf{U}^{(r)})\mathbf{U}^{(r)} - \mathbf{F}^{(r)}$$
and check if
$$\|\mathbf{R}^{(r)}\| \leq \epsilon\|\mathbf{F}^{(r)}\| \quad \text{or} \quad \|\Delta\mathbf{U}\| \leq \epsilon\|\mathbf{U}^{(r)}\|$$
where ϵ is the convergence tolerance (read as an input). If the solution has converged, print the solution and move to the next "load" level or quit. Otherwise, continue.
5. **Maximum iteration check.** Check if $r < itmax$, where $itmax$ is the maximum number of iterations allowed (read as an input). If *yes*, set $r \to r+1$ and go to Step 2; if *no*, print a message that the iteration scheme did not converge and quit.



- In the direct iteration method, the actual bc are applied at each iteration. In NR we find the increment to the known solution. If previous displ satisfies, then the increment should be zero and satisfy the boundary condition.
- The symmetrry of K and T depends on the weak form. even if K is symmetric, T may not be symmetric.
- T can be approximate, and convergance is only when the residual is small. If it is only updated once then it is called the modified Netwons method.
- See the problem in Reddy -188 . Finding T