5 Non-Linear Differential Equations

Application of the Finite Element Method to the solution of linear differential equations leads to a system of linear algebraic equations of the form $\mathbf{Ax} = \mathbf{b}$; with non-linear differential equations one arrives at a system of non-linear equations, which cannot be solved by elementary elimination methods. Thus, much of the focus here is on methods of solving the resulting systems of FE non-linear equations.

5.1 Methods for the Solution of Non-Linear Equations

There are a number of basic techniques for solving non-linear equations. For example, there are the

1. Substitution method
2. Newton-Raphson method
3. Incremental (step by step) method
   - Initial Stress Method
   - Modified Newton-Raphson method

The Substitution Method is not commonly used, but is given in the Appendix to this Chapter. The Newton-Raphson method is the primary solution scheme for the non-linear equations which arise in the FEM and will be discussed in detail.

5.1.1 The Newton-Raphson Method

Consider first the one-dimensional case: the non-linear equation

$$R(u) = 0, \tag{5.1}$$

whose exact solution is $u^{(e)}$. Suppose one has an initial estimate of the solution, $u^{(0)}$. Using a Taylor expansion and dropping higher order terms,

$$R(u^{(e)}) = R(u^{(0)}) + \Delta u \left.\frac{\partial R}{\partial u}\right|_{u^{(0)}}, \tag{5.2}$$

where $\Delta u = u^{(e)} - u^{(0)}$. Using $R(u^{(e)}) = 0$ then leads to the approximation

$$u^{(e)} \approx u^{(0)} - \frac{R(u^{(0)})}{\partial R / \partial u\big|_{u^{(0)}}} \tag{5.3}$$

which is an expression for the next approximation.

Consider the following one-dimensional example: solve the non-linear equation

$$-2u^2 + \tfrac{8}{3}u = -\tfrac{2}{3} \tag{5.4}$$

One has

$$R(u) = -2u^2 + \tfrac{8}{3}u + \tfrac{2}{3}, \qquad \frac{\partial R}{\partial u} = -4u + \tfrac{8}{3} \tag{5.5}$$

and the algorithm

$$\Delta u^i = -\frac{R\big|_{u^{i-1}}}{\left(\partial R / \partial u\right)_{u^{i-1}}}, \qquad u^i = u^{i-1} + \Delta u^i \tag{5.6}$$

The function $R(u)$ is called a **residual** function, in the sense that the objective is to drive this function to zero.

Using an initial estimate of $u = 1$ gives the sequence

		error
$u^{(0)}$	1.00000000	0.54858377
$u^{(1)}$	2.00000000	0.45141623
$u^{(2)}$	1.62500000	0.07641623
$u^{(3)}$	1.55163043	0.00304666
$u^{(4)}$	1.54858901	0.00000524
$u^{(5)}$	1.54858377	0.00000000
exact	**1.54858377**	

Table 5.1: Newton-Raphson solution

If one had started with an initial estimate of -1, one would have converged to the other root, -0.21525044.

A geometrical interpretation of the Newton-Raphson method is shown below for this example. In deriving the Newton-Raphson method, the process of *linearization* has been used. The linearised model is the tangent to the nonlinear residual function (and hence the name "tangent matrix").

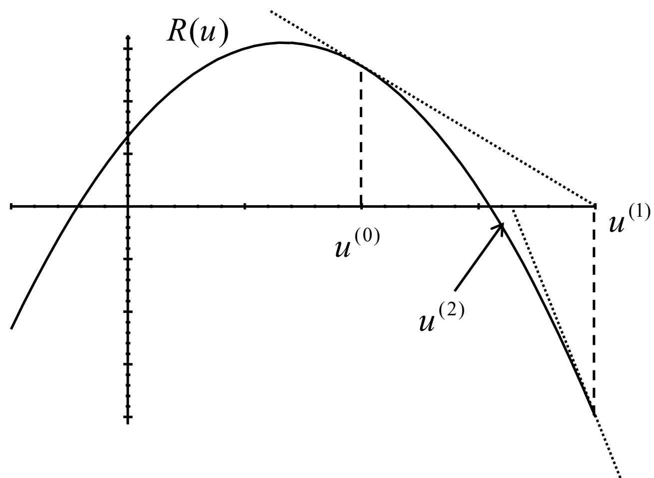

Figure 5.1: Geometric interpretation of the Newton-Raphson Solution Scheme

Notes on Convergence and Accuracy

(1) if the derivative $\partial R / \partial u$ is continuous in a neighbourhood of the solution, and if $u^{(i-1)}$ lies in that neighbourhood, then the next iterated solution $u^{(i)}$ *will be closer to the solution* than $u^{(i-1)}$ and the scheme will converge towards the solution.

(2) if $\partial R / \partial u\big|_{u^{(i-1)}} = 0$, the method will fail – geometrically, this occurs when the tangent to R is horizontal

(3) The convergence is quadratic, that is, if the error after iteration $i-1$ is ε, the error after iteration i is ε^2. This can be seen as follows: write

$$u^{(i)} = u^{(i-1)} - \frac{R(u^{(i-1)})}{R'(u^{(i-1)})} \equiv g(u^{(i-1)})$$

Let the true solution be $u^{(e)}$, so that $u^{(i-1)} = u^{(e)} - \varepsilon_{i-1}$, where ε_{i-1} is the error of $u^{(i-1)}$. By a Taylor series,

$$u^{(i)} = g(u^{(i-1)}) = g(u^{(e)} - \varepsilon_{i-1}) = g(u^{(e)}) - \varepsilon_{i-1} g'(u^{(e)}) + \tfrac{1}{2}\varepsilon_{i-1}^2 g''(u^{(e)}) - \cdots$$

Now

$$g(u^{(e)}) = u^{(e)}, \qquad g'(u^{(e)}) = \frac{R(u^{(e)})R''(u^{(e)})}{\left[R'(u^{(e)})\right]^2} = 0, \qquad g''(u^{(e)}) = \frac{R''(u^{(e)})}{R'(u^{(e)})} \neq 0,$$

since $R(u^{(e)}) = 0$. Thus the error in the next estimate is

$$\varepsilon_i = u^{(e)} - u^{(i)} \approx -\tfrac{1}{2}\varepsilon_{i-1}^2 g''(u^{(e)})$$

and so the method is of second-order (quadratic) – provided $R'(u^{(e)}) \neq 0$

Multi-dimensional Equations

Consider the following system of non-linear equations,

$$\mathbf{R}(\mathbf{u}) = \begin{bmatrix} R_1(\mathbf{u}) \\ R_2(\mathbf{u}) \\ \vdots \\ R_n(\mathbf{u}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \qquad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}, \tag{5.7}$$

whose exact solution is $\mathbf{u}^{(e)}$. Suppose one has an initial estimate of the solution, $\mathbf{u}^{(0)}$. Using a Taylor expansion and dropping higher order terms,

$$\mathbf{R}(\mathbf{u}^{(e)}) = \mathbf{R}(\mathbf{u}^{(0)}) + \Delta\mathbf{u} \left.\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right|_{\mathbf{u}^{(0)}}, \tag{5.8}$$

where $\Delta\mathbf{u} = \mathbf{u}^{(e)} - \mathbf{u}^{(0)}$. Using $\mathbf{R}(\mathbf{u}^{(e)}) = 0$ then leads to

$$\left.\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right|_{\mathbf{u}^{(0)}} \Delta\mathbf{u} = -\mathbf{R}(\mathbf{u}^{(0)}) \tag{5.9}$$

Solving this system of linear equations leads to a new approximation $\mathbf{u}^{(1)} = \mathbf{u}^{(0)} + \Delta\mathbf{u}$. An algorithm is then

Newton-Raphson Algorithm:

$$\left[\mathbf{K}_T(\mathbf{u}^{i-1})\right]\Delta\mathbf{u}^i = -\mathbf{R}(\mathbf{u}^{i-1}), \qquad \mathbf{K}_T(\mathbf{u}^{i-1}) = \left.\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right|_{\mathbf{u}^{(i-1)}} \tag{5.10}$$
$$\mathbf{u}^i = \mathbf{u}^{i-1} + \Delta\mathbf{u}^i$$

The matrix $\mathbf{K}_T$ is called the *tangent matrix*.

Tangent Matrix:

$$\mathbf{K}_T = \frac{\partial \mathbf{R}}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial R_1}{\partial u_1} & \frac{\partial R_1}{\partial u_2} & \cdots & \frac{\partial R_1}{\partial u_n} \\ \frac{\partial R_2}{\partial u_1} & \frac{\partial R_2}{\partial u_2} & \cdots & \cdots \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial R_2}{\partial u_n} & \cdots & \cdots & \frac{\partial R_n}{\partial u_n} \end{bmatrix} \tag{5.11}$$

A Two-Dimensional Problem

Consider the following system of non-linear equations

$$\begin{aligned} \tfrac{2}{3} + \tfrac{8}{3}u_1 + \tfrac{7}{3}u_2 - 2u_1^2 - \tfrac{16}{5}u_1u_2 - \tfrac{7}{5}u_2^2 &= 0 \\ \tfrac{7}{12} + \tfrac{7}{3}u_1 + \tfrac{34}{15}u_2 - \tfrac{8}{5}u_1^2 - \tfrac{14}{5}u_1u_2 - \tfrac{47}{35}u_2^2 &= 0 \end{aligned} \tag{5.12}$$

One has

$$\mathbf{R}(\mathbf{u}) = \begin{bmatrix} \mathbf{R}_1(\mathbf{u}) \\ \mathbf{R}_2(\mathbf{u}) \end{bmatrix} = \begin{bmatrix} \tfrac{2}{3} + \tfrac{8}{3}u_1 + \tfrac{7}{3}u_2 - 2u_1^2 - \tfrac{16}{5}u_1u_2 - \tfrac{7}{5}u_2^2 \\ \tfrac{7}{12} + \tfrac{7}{3}u_1 + \tfrac{34}{15}u_2 - \tfrac{8}{5}u_1^2 - \tfrac{14}{5}u_1u_2 - \tfrac{47}{35}u_2^2 \end{bmatrix} \tag{5.13}$$

and

$$\mathbf{K}_T = \frac{\partial \mathbf{R}}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial R_1}{\partial u_1} & \frac{\partial R_1}{\partial u_2} \\ \frac{\partial R_2}{\partial u_1} & \frac{\partial R_2}{\partial u_2} \end{bmatrix} = \begin{bmatrix} \tfrac{8}{3} - 4u_1 - \tfrac{16}{5}u_2 & \tfrac{7}{3} - \tfrac{16}{5}u_1 - \tfrac{14}{5}u_2 \\ \tfrac{7}{3} - \tfrac{16}{5}u_1 - \tfrac{14}{5}u_2 & \tfrac{34}{15} - \tfrac{14}{5}u_1 - \tfrac{94}{35}u_2 \end{bmatrix} \tag{5.14}$$

The Newton-Raphson scheme then converges to the exact solutions $(u_1, u_2) = (-0.19401, -0.02477)$, $(0.00541, 1.90911)$, $(3.09001, -2.659787)$ or $(-3.03474, 4.508781)$.

5.1.2 The Incremental Method (with the Newton-Raphson Method)

Consider the general residual function

$$R(u) = K(u) - F = 0 \tag{5.15}$$

Here, u is the unknown and F will in general be some known. In a mechanics problem, for example, u would be the unknown displacement and F would be due to the known applied loads. In a linear problem, one can specify F and one can solve for u., Fig. 5.2a. In a non-linear problem, however, this is not so straight-forward. As an extreme example, the true relationship between F and u might look like that illustrated in Fig. 5.2b; there will in general be more than one solution for u corresponding to any given F. Referring to Fig. 5.2b, it would be unlikely that one could find $\bar{u}$ given $\bar{F}$, unless one began the algorithm close to $\bar{u}$. For this reason, in a practical FE problem, one does not try to solve a non-linear problem "in one hit". There is a danger that, if the initial prediction $u^{(0)}$ is inaccurate, the solution will not be found.

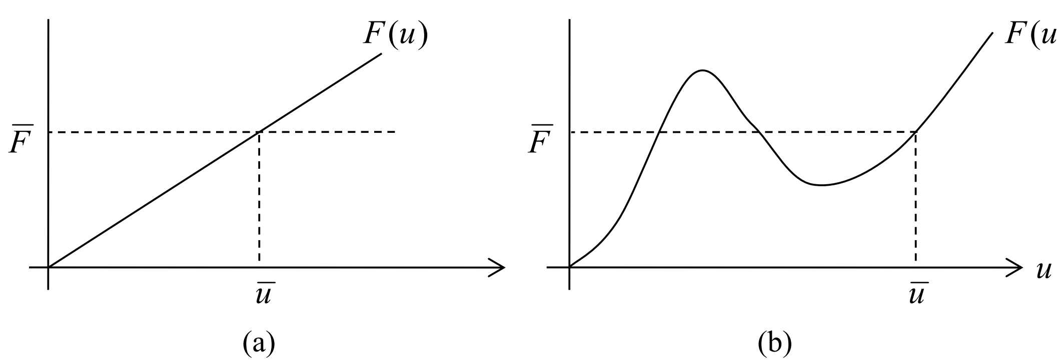

Figure 5.2: Example relationship between *F* and *u* in the residual function: (a) linear problem, (b) non-linear problem

For this reason, the problem is split up into a number of **increments**: the known "loading" function *F* is split into the increments $F_0, F_1, F_2, \ldots$. One first solves the equation

$$K(u) = F_0 \tag{5.16}$$

using the Newton-Raphson method until the solution converges. Once the solution is found, another increment in *F* is made, and the equation to be solved is

$$K(u) = F_1, \tag{5.17}$$

using the previous found value of *u* as the new prediction. *F* is incremented in this fashion until the final value is reached and the solution is obtained. The procedure is illustrated in Fig. 5.3.

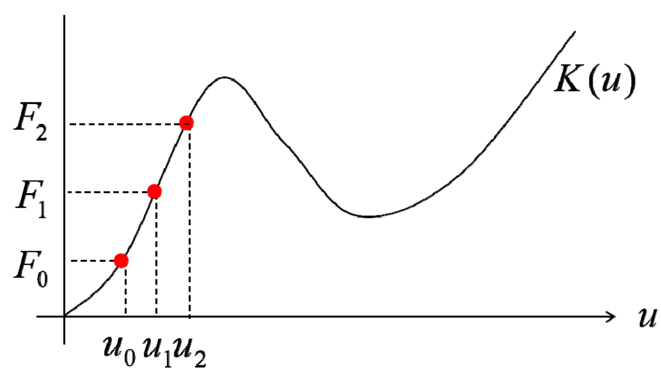

Figure 5.3: Solving non-linear equations using a number of stepped increments in the known function *F*

To be clear, the term **increment** is used to mean a change in *F*, whereas an **iteration** is used to mean a step in the Newton-Raphson algorithm (denoted by the "*i*" of Eqns. 5.10). Thus, at each increment, there are a number of iterations to convergence.

Assuming that one has equal increments in *F*, the algorithm can be written as (letting $F = (j-1)\hat{F}$, $j = 1, 2, \ldots$, with increment *j* and iteration *i*)

$$\begin{gathered} \left[K_T(a_j^{i-1})\right]\Delta a_j^i = -R(a_j^{i-1}) \qquad R(a_j^{i-1}) = K(a_j^{i-1}) - (j-1)\hat{F} \\ a_j^i = a_j^{i-1} + \Delta a_j^i \end{gathered} \tag{5.18}$$

Results are shown below for the one-dimensional example considered earlier, Eqn. 5.4, with four equal increments $\hat{F} = -1/6$, and three iterations per increment.

increment	Fj	iteration	a
0	0	0	1.00000000
		1	1.50000000
		2	1.35000000
		3	1.33353659
1	-1/6	0	1.33353659
		1	1.39581432
		2	1.39315469
		3	1.39314982
2	-1/3	0	1.39314982
		1	1.45050376
		2	1.44840544
		3	1.44840263
3	-1/2	0	1.44840263
		1	1.50170281
		2	1.50000174
		3	1.50000000
4	-2/3	0	1.50000000
		1	1.55000000
		2	1.54858491
		3	1.54858377

Initial stress method: use this K_T throughout the complete

Modified Newton-Raphson method: use this K_T for all iterations during this increment. Update K_T at the start of the

There are a number of variations of the incremental method:

1. The **initial stress method**: here the tangent matrix used at the start of the solution process is used throughout the analysis. The effort required with this method is greatly reduced, since only one evaluation of K_T is required. This reduced effort is offset by the fact that the scheme will inevitable converge more slowly (or even diverge).

2. The **modified Newton-Raphson method**: here the tangent matrix used at the start of an increment, $K_T(u_j^0)$, is used for all iterations during that increment – it is an approach somewhat in between the initial stress and full Newton-Raphson methods.

Quasi-Newton or **matrix update** methods, for example the BFGS and DFP schemes, are a compromise between the full Newton-Raphson method and the other schemes which use a tangent matrix method from a previous configuration. These schemes often involve secants to the curve.

All the above solution methods have their advantages, whether they be "constant K" or "variable K" methods. The precise choice of the optimal methodology is problem dependent and although many comparative solution cost studies have been published, the differences are often marginal. There is little doubt, however, that the full Newton-Raphson process has to be used when convergence is difficult to achieve. An automatic procedure that self-adaptively chooses an effective technique is most attractive.

Note: small increments reduce the total number of iterations required per increment and in many finite element software programs automatic guidance on the size of an increment to preserve a (nearly) constant number of iterations is provided.

Comparison with a simple explicit solution

The equations $\mathbf{K}(\mathbf{u}) = \mathbf{F}$ can also be solved incrementally using a simple explicit procedure. In this case one can write

$$\Delta\mathbf{K}(\mathbf{u}) = \Delta\mathbf{F} \tag{5.19}$$

or

$$\frac{\partial \mathbf{K}}{\partial \mathbf{u}} \Delta\mathbf{u} = \Delta\mathbf{F} \tag{5.20}$$

For example, consider again the one-dimensional equation $-2u^2 + \frac{8}{3}u = F$. Then $K_T = \partial K / \partial u = -4u + \frac{8}{3}$, and the algorithm is $K_T(u_k)\Delta u_k = \Delta F_k$. This can now be solved by specifying the "step size" ΔF_k. The solution is shown in Fig. 5.4 for $\Delta F_k = 0.1$ and $\Delta F_k = 0.2$. One can observe the drift away from the exact solution with successive steps, typical of explicit methods.

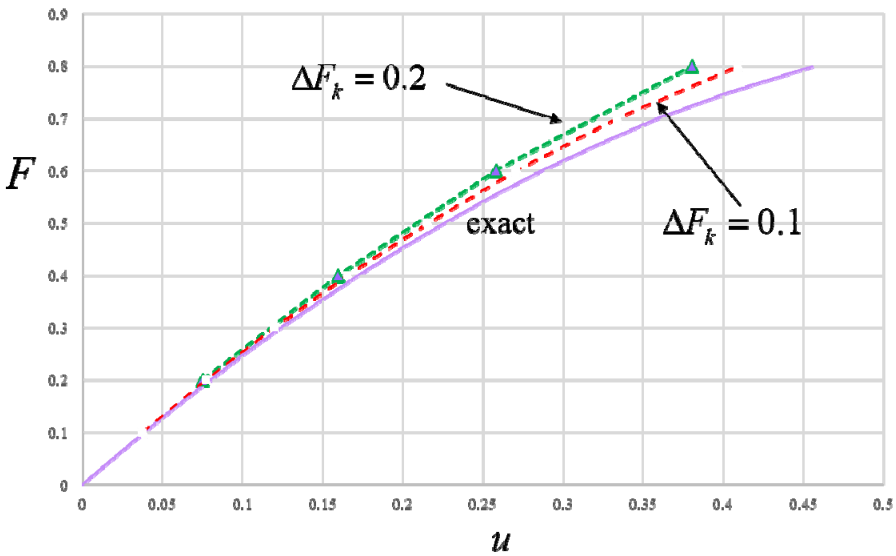

Figure 5.4: Explicit solution to one-dimensional equation

In contrast to the above explicit method, the Newton-Raphson scheme "pulls" the solution back towards the exact solution at each increment. This is illustrated in Fig. 5.5.

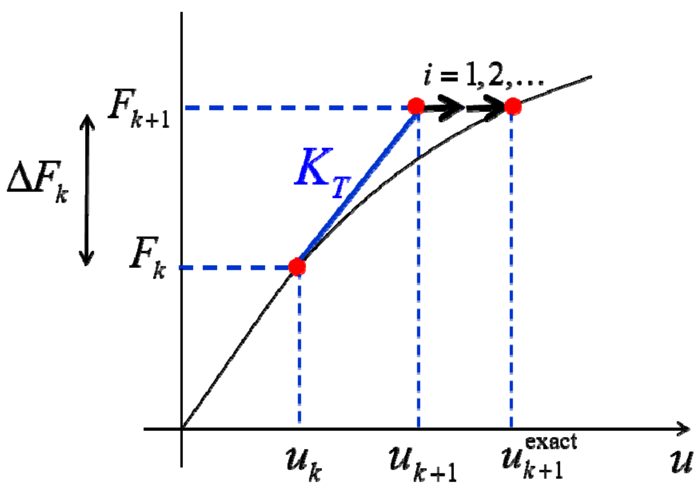

Figure 5.5: Newton-Raphson method pulling solution back towards exact solution at each increment

5.2 The FEM for the Solution of Non-Linear ODEs

Here, the FEM solution for non-linear ODEs is outlined by considering again the non-linear differential equation considered in Chapter 1, Eqn. 1.70,

$$2\frac{du}{dx}\frac{d^2u}{dx^2}+1=0, \qquad u(0)=0, \quad \left.\tfrac{\partial u}{\partial x}\right|_{x=1}=1 \tag{5.21}$$

[the exact solution is $u(x)=\frac{2}{3}\left(2^{3/2}-(2-x)^{3/2}\right)$]

Formation of the weighted residual, integration by parts, and substituting in the shape functions for linear elements gives

$$-u_1^2\int_0^1\left(\frac{dN_1}{dx}\right)^2\frac{dN_j}{dx}dx-2u_1u_2\int_0^1\frac{dN_1}{dx}\frac{dN_2}{dx}\frac{dN_j}{dx}dx-u_2^2\int_0^1\left(\frac{dN_2}{dx}\right)^2\frac{dN_j}{dx}dx+\int_0^1 N_j dx=-\left[\left(\frac{du}{dx}\right)^2\omega\right]_0^1 \tag{5.22}$$

Converting to local coordinates and evaluating the integrals:

$$\begin{aligned}
\int_{x_i}^{x_{i+1}}\left(\frac{dN_i}{dx}\right)^2\frac{dN_j}{dx}dx &= \left(\frac{d\xi}{dx}\right)^2\int_{-1}^{+1}\left(\frac{dN_i}{d\xi}\right)^2\frac{dN_j}{d\xi}d\xi=\frac{4}{L^2}\begin{bmatrix}-\frac{1}{4} & +\frac{1}{4}\\ -\frac{1}{4} & +\frac{1}{4}\end{bmatrix}\\
\int_{x_i}^{x_{i+1}}\frac{dN_1}{dx}\frac{dN_2}{dx}\frac{dN_j}{dx}dx &= \left(\frac{d\xi}{dx}\right)^2\int_{-1}^{+1}\frac{dN_1}{d\xi}\frac{dN_2}{d\xi}\frac{dN_j}{d\xi}d\xi=\frac{4}{L^2}\begin{bmatrix}+\frac{1}{4}\\ -\frac{1}{4}\end{bmatrix}
\end{aligned} \tag{5.23}$$

leads to the element equations

$$\begin{aligned}
+u_1^2\frac{1}{L^2}-2u_1u_2\frac{1}{L^2}+u_2^2\frac{1}{L^2}+\frac{L}{2} &= +\left.\left(\frac{du}{dx}\right)^2\right|_{x_i}\\
-u_1^2\frac{1}{L^2}+2u_1u_2\frac{1}{L^2}-u_2^2\frac{1}{L^2}+\frac{L}{2} &= -\left.\left(\frac{du}{dx}\right)^2\right|_{x_{i+1}}
\end{aligned} \tag{5.24}$$

Rather than form the global matrix, and thence derive the tangent matrix through differentiation of the global matrix, it is more efficient to form the elemental tangent matrix and then from there to form the global tangent matrix. The element tangent matrix is, differentiating the element matrix,

$$\mathbf{K}_T^{(el)} = \frac{2}{L^2}\begin{bmatrix} +u_1 - u_2 & -u_1 + u_2 \\ -u_1 + u_2 & +u_1 - u_2 \end{bmatrix} \tag{5.25}$$

Using two elements leads to the system of three non-linear equations (with $L = 1/2$)

$$\begin{aligned} +4u_1^2 - 8u_1u_2 + 4u_2^2 + \frac{1}{4} &= +\left(\frac{du}{dx}\right)^2\bigg|_0 \\ -4u_1^2 + 8u_1u_2 - 8u_2u_3 + 4u_3^2 + \frac{1}{2} &= 0 \\ -4u_2^2 + 8u_2u_3 - 4u_3^2 + \frac{1}{4} &= -\left(\frac{du}{dx}\right)^2\bigg|_1 \end{aligned} \tag{5.26}$$

Applying the boundary conditions $u_1 = u(0) = 0, \quad u'(1) = 1$ leads to

$$\begin{aligned} -8u_2u_3 + 4u_3^2 + \tfrac{1}{2} &= 0 \\ -4u_2^2 + 8u_2u_3 - 4u_3^2 + \tfrac{5}{4} &= 0 \end{aligned} \tag{5.27}$$

The equations are now solved using the Newton-Raphson method, for which

$$\mathbf{R}(\mathbf{u}) = 8\begin{bmatrix} -u_2u_3 + \frac{1}{2}u_3^2 + \frac{1}{16} \\ -\frac{1}{2}u_2^2 + u_2u_3 - \frac{1}{2}u_3^2 + \frac{5}{32} \end{bmatrix} \tag{5.28}$$

$$\mathbf{K}_T(\mathbf{u}) = 8\begin{bmatrix} -u_3 & -u_2 + u_3 \\ -u_2 + u_3 & u_2 - u_3 \end{bmatrix} \tag{5.29}$$

Solving the equations $\mathbf{K}_T(\mathbf{u}^{(i-1)})\Delta\mathbf{u}^{(i)} = -\mathbf{R}(\mathbf{u}^{(i-1)})$ iteratively, with the initial conditions $\mathbf{u}^{(0)} = \begin{bmatrix} 0 & 0.1 & 0.2 \end{bmatrix}^{\mathrm{T}}$, results in the solution sequence

$u_2^{(0)}$	0.10000000	$u_3^{(0)}$	0.20000000
$u_2^{(1)}$	2.23750000	$u_3^{(1)}$	3.85000000
$u_2^{(2)}$	1.21651536	$u_3^{(2)}$	2.11966459
$u_2^{(3)}$	0.78807456	$u_3^{(3)}$	1.41265492
$u_2^{(4)}$	0.67161254	$u_3^{(4)}$	1.23407069
$u_2^{(5)}$	0.66151490	$u_3^{(5)}$	1.22054242
$u_2^{(6)}$	0.66143783	$u_3^{(6)}$	1.22045483
$u_2^{(7)}$	0.66143783	$u_3^{(7)}$	1.22045482
exact	**0.66087321**	exact	**1.21895142**

Table 5.2: Solution of non-linear equations

Using three elements and applying the boundary conditions leads to

$$\begin{aligned} -18u_2u_3 + 9u_3^2 + \tfrac{1}{3} &= 0 \\ -9u_2^2 + 18u_2u_3 - 18u_3u_4 + 9u_4^2 + \tfrac{1}{3} &= 0 \\ -9u_3^2 + 18u_3u_4 - 9u_4^2 + \tfrac{7}{6} &= 0 \end{aligned} \tag{5.30}$$

so that

$$\mathbf{R}(\mathbf{u}) = 18\begin{bmatrix} -u_2u_3 + \frac{1}{2}u_3^2 + \frac{1}{54} \\ -\frac{1}{2}u_2^2 + u_2u_3 - u_3u_4 + \frac{1}{2}u_4^2 + \frac{1}{54} \\ -\frac{1}{2}u_3^2 + u_3u_4 - \frac{1}{2}u_4^2 + \frac{7}{108} \end{bmatrix} \tag{5.31}$$

$$\mathbf{K}_T(\mathbf{u}) = 18\begin{bmatrix} -u_3 & -u_2 + u_3 & 0 \\ -u_2 + u_3 & u_2 - u_4 & -u_3 + u_4 \\ 0 & -u_3 + u_4 & u_3 - u_4 \end{bmatrix} \tag{5.32}$$

which can be used to obtain better accuracy.

5.3 The FEM for the Solution of Non-Linear PDEs

Here, the FEM solution for non-linear PDEs is outlined by considering again Eqn. 5.21, but now with a first order term in t,

$$\frac{\partial u}{\partial t}+2\frac{\partial u}{\partial x}\frac{\partial^2 u}{\partial x^2}+1=0, \qquad u(0,t)=0, \quad \tfrac{\partial u}{\partial x}(1,t)=1, \quad u(x,0)=\sin(-2.074x) \tag{5.33}$$

The formulation of the FE problem is the same as before, only now one has a capacitance matrix **C**:

$$\int_{x_i}^{x_{i+1}} \frac{\partial u}{\partial t}\omega_j \, dx = \dot{u}_1 \int_{x_i}^{x_{i+1}} N_1 N_j \, dx + \dot{u}_2 \int_{x_i}^{x_{i+1}} N_2 N_j \, dx \quad \rightarrow \quad \mathbf{C}^{(el)} = \frac{L}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \tag{5.34}$$

The time-independent part of the element equations can be written as $\mathbf{K}^{(el)}(\mathbf{u})=\mathbf{F}^{(el)}$,

$$\mathbf{K}^{(el)}(\mathbf{u})=\frac{1}{L^2}\begin{bmatrix} +u_1^2-2u_1u_2+u_2^2 \\ -u_1^2+2u_1u_2-u_2^2 \end{bmatrix}, \qquad \mathbf{F}^{(el)}=\begin{bmatrix} -\frac{L}{2}+\left(u'\right)^2\Big|_{x_i} \\ -\frac{L}{2}-\left(u'\right)^2\Big|_{x_{i+1}} \end{bmatrix} \tag{5.35}$$

with, as before,

$$\mathbf{K}_T^{(el)}=\frac{2}{L^2}\begin{bmatrix} +u_1-u_2 & -u_1+u_2 \\ -u_1+u_2 & +u_1-u_2 \end{bmatrix} \tag{5.36}$$

Using two elements and applying the boundary conditions leads to

$$\mathbf{C}\dot{\mathbf{u}}(t)+\mathbf{K}\big(\mathbf{u}(t)\big)=\mathbf{F}(t) \tag{5.37}$$

where

$$\mathbf{C}=\frac{1}{12}\begin{bmatrix} 4 & 1 \\ 1 & 2 \end{bmatrix}, \quad \mathbf{K}(\mathbf{u}(t))=4\begin{bmatrix} -2u_2u_3+u_3^2 \\ -u_2^2+2u_2u_3-u_3^2 \end{bmatrix}, \quad \mathbf{F}=\begin{bmatrix} -\frac{1}{2} \\ -\frac{5}{4} \end{bmatrix}, \quad \mathbf{u}=\begin{bmatrix} u_2 \\ u_3 \end{bmatrix} \tag{5.38}$$

An explicit and an implicit means of solving these equations are discussed next.

5.3.1 Explicit formula

In the explicit scheme, the governing equations are written at time t, $\mathbf{C}\dot{\mathbf{u}}(t)+\mathbf{K}(\mathbf{u}(t))=\mathbf{F}(t)$, and then the time derivative is replaced by the forward difference approximation, leading to

$$\mathbf{Cu}(t+\Delta t)=\mathbf{Cu}(t)+\Delta t\left[\mathbf{F}(t)-\mathbf{K}(\mathbf{u}(t))\right] \tag{5.39}$$

or, in incremental form,

Explicit Algorithm:

$$\begin{aligned} &\mathbf{C}\Delta\mathbf{u}=\mathbf{R}(t) \\ &\quad \text{where } \mathbf{R}(t)=\Delta t\left[\mathbf{F}(t)-\mathbf{K}(\mathbf{u}(t))\right] \\ &\mathbf{u}(t+\Delta t)=\mathbf{u}(t)+\Delta\mathbf{u} \end{aligned} \tag{5.40}$$

There is little difficulty in implementing this strategy – the only difference between this scheme and the corresponding linear one is that the linear equations $\mathbf{Ku}$ are replaced by the non-linear equations $\mathbf{K}(\mathbf{u})$. Note that there are *no* non-linear equations to be solved here, so *the Newton-Raphson scheme is not necessary*.

5.3.2 Implicit formula

The governing equations are written at time $t+\Delta t$ and then rewritten using the backward difference formula as

$$\mathbf{C}\left[\mathbf{u}(t+\Delta t)-\mathbf{u}(t)\right]=\Delta t\left[\mathbf{F}(t+\Delta t)-\mathbf{K}(\mathbf{u}(t+\Delta t))\right] \tag{5.41}$$

Introduce a residual function

$$\mathbf{R}(\mathbf{u}(t+\Delta t))=\mathbf{C}\left[\mathbf{u}(t+\Delta t)-\mathbf{u}(t)\right]-\Delta t\mathbf{F}(t+\Delta t)+\Delta t\mathbf{K}(\mathbf{u}(t+\Delta t)) \tag{5.42}$$

Suppose that one already has an estimate of $\mathbf{u}(t+\Delta t)$, say $\mathbf{u}^{(0)}(t+\Delta t)$ - one can use the value $\mathbf{u}(t)$ as the initial prediction $\mathbf{u}^{(0)}(t+\Delta t)$ - or one could perhaps obtain an initial prediction by extrapolation through $\cdots,\mathbf{u}(t-\Delta t),\mathbf{u}(t)$.

One has $\mathbf{R}(\mathbf{u}^{(0)}(t+\Delta t)) \neq 0$ and it is required that the expression $\mathbf{R}(\mathbf{u}^{(1)}(t+\Delta t)) = 0$ holds. Following the Newton-Raphson procedure, expand in a Taylor series (note that the following is *for fixed time* $t+\Delta t$)

$$\mathbf{R}\left(\mathbf{u}^{(1)}(t+\Delta t)\right) = \mathbf{R}\left(\mathbf{u}^{(0)}(t+\Delta t)\right) + \Delta\mathbf{u}\left.\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right|_{\mathbf{u}^{(0)}(t+\Delta t)} + \cdots = \mathbf{0} \tag{5.43}$$

where $\Delta\mathbf{u} = \mathbf{u}^{(1)}(t+\Delta t) - \mathbf{u}^{(0)}(t+\Delta t)$. Dropping the higher order terms gives the implicit algorithm

Implicit Algorithm:

$$\begin{gathered}
\left[\mathbf{K}_T\left(\mathbf{u}^{i-1}(t+\Delta t)\right)\right]\Delta\mathbf{u}^{i}\left(t+\Delta t\right) = -\mathbf{R}\left(\mathbf{u}^{i-1}(t+\Delta t)\right), \qquad \mathbf{K}_T = \left.\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right|_{\mathbf{u}^{i-1}(t+\Delta t)} \\
\mathbf{R}\left(\mathbf{u}^{i-1}(t+\Delta t)\right) = \mathbf{C}\left[\mathbf{u}^{i-1}(t+\Delta t) - \mathbf{u}(t)\right] - \Delta t\mathbf{F}(t+\Delta t) + \Delta t\mathbf{K}\left(\mathbf{u}^{i-1}(t+\Delta t)\right) \\
\mathbf{K}_T = \mathbf{C} + \Delta t\left.\frac{\partial \mathbf{K}}{\partial \mathbf{u}}\right|_{\mathbf{u}^{i-1}(t+\Delta t)}
\end{gathered} \tag{5.44}$$

Results using two elements are shown in Fig. 5.6.

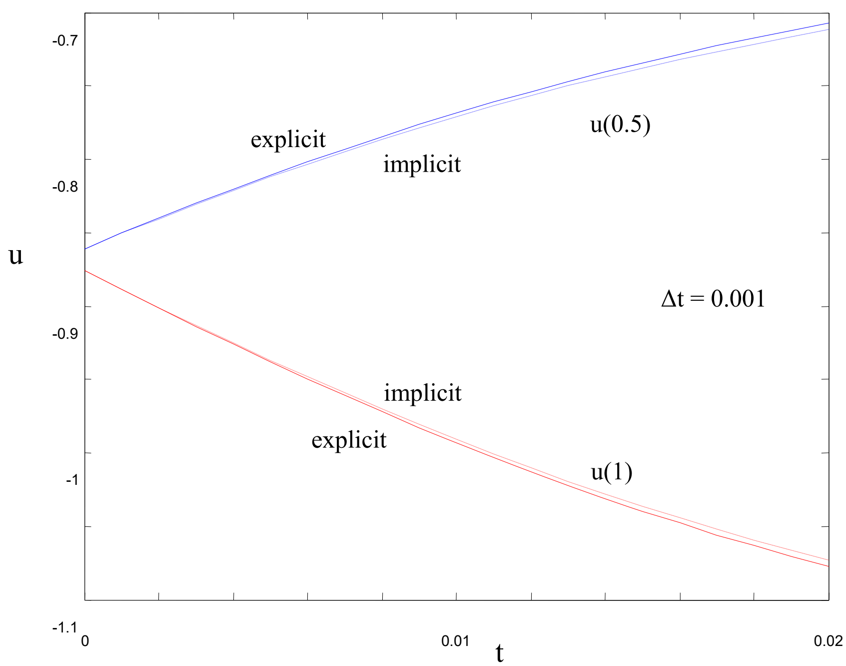

Figure 5.6: FE Solution to the PDE (5.30)

5.4 Problems

1. Solve the following system of two non-linear equations using the Newton-Raphson scheme

$$5 - 3a_1 + 2a_1 a_2 = 0$$
$$2 + a_2 - a_1^2 = 0$$

[the exact solution is $(1,-1)$ and $(1.1583,-0.6583)$]

5.5 Appendix to Chapter 5

5.5.1 The Substitution Method

Write the non-linear equation as

$$R = \left[K(a)\right]a - F, \qquad K(a) = -2a + \tfrac{8}{3}, \quad F(a) = -\tfrac{2}{3} \tag{5.A1}$$

and so the residual for the approximation a^{i-1} is

$$R^i = \left[K(a^{i-1})\right]a^{i-1} - F \neq 0 \tag{5.A2}$$

The next approximation a^i is determined by solving the linear system $\left[K(a^{i-1})\right]a^i = F$, where a^{i-1} is the previous approximate value. In incremental form, one has, with $a^i = a^{i-1} + \Delta a^i$,

$$\begin{gathered} \left[K(a^{i-1})\right]\Delta a^i = -R^i, \qquad R^i = \left[K(a^{i-1})\right]a^{i-1} - F \\ a^i = a^{i-1} + \Delta a^i \end{gathered} \tag{5.A3}$$

One can now use the following algorithm:

Predict a^{i-1}

Iterate to correct

for $i = 1, 2, \ldots$

solve $K(a^{i-1})\Delta a^i = R^i$

$a^i = a^{i-1} + \Delta a^i$

test convergence

The convergence test is usually of the form: is $\left|\Delta a^i\right| < \varepsilon$?, is $\left|R^i\right| < \varepsilon$?, is $\left|\dfrac{\Delta a^i}{a^i}\right| < \varepsilon$?

Using an initial estimate of $a = 1$ gives the sequence

$a^{(0)}$	1.00000000
$a^{(1)}$	-1.00000005

$a^{(2)}$	-0.14285715
$a^{(3)}$	-0.22580646
$a^{(4)}$	-0.21379311
$a^{(5)}$	-0.21545320
exact	**-0.21525044**

A geometrical interpretation of the iteration from $a^{(0)} \rightarrow a^{(1)}$ is shown here. The method applied to this equation cannot locate the root at $a = 1.549$. Further, the solution may diverge for certain functions (which undergo two or more changes in curvature within the search area).