T U T O R I A L

"Numerical Methods for Solving Partial Differential Equations"

to the Lectures on NuPDE

T IV Monday, 19 November 2007 (Time: 08:30 – 10:00, Room: T 212)

1.4 FEM for BVPs for second-order ODEs (continued)

We consider the same setting in Tutorial III, i. e., the boundary value problem (1.21).

In the following, we denote input parameters of a function by $\downarrow$, output parameters by $\uparrow$, and input/output parameters by $\updownarrow$.

19 Tidy-up your current program!

- Eliminate dynamic arrays where static can be used
- Memory: Make sure that you don't lose pointers, and that you delete everything you allocated with new.
- Make sure that you match the interfaces given in Tutorial III. In particular use reference calls where output parameters are indicated, not function return values/pointers. For instance

```
typedef double (*RealFunction)(double x);
typedef double Vec2[2];
void ElementLoadVector (RealFunction f, double xa, double xb, Vec2& element_vector);
```

20 Write a function `ImplementRobinBC(`$\downarrow$`i, `$\downarrow$`g, `$\downarrow$`alpha, `$\updownarrow$`matrix, `$\updownarrow$`vector)` to implement the Robin boundary condition

$$u'(x_i) = \alpha(x_i)\left(g_R(x_i) - u(x_i)\right)$$

for given values `g`$=g_R(x_i)$, `alpha`$=\alpha(x_i)$ at the boundary node x_i identified by the index `i`$=i$. The function `ImplementRobinBC` must update the stiffness matrix `matrix` and the load vector `vector`, previously computed by `AssembleStiffnessMatrix` and `AssembleLoadVector`, respectively, in the case of homogeneous Neumann conditions.

21 Write a function `ImplementDirichletBC(`$\downarrow$`i, `$\downarrow$`g, `$\updownarrow$`matrix, `$\updownarrow$`vector)` to implement the Dirichlet boundary condition

$$u(x_i) = g_D(x_i)$$

for a given value `g`$=g_D(x_i)$ at the boundary node x_i identified by the index `i`$=i$. The function `ImplementDirichletBC` must update the stiffness matrix `matrix` and the load vector `vector`, previously computed by `AssembleStiffnessMatrix` and

`AssembleLoadVector`, respectively, in the case of homogeneous Neumann conditions, and by `ImplementRobinBC`.

Hint: Assume that applying `AssembleStiffnessMatrix`, `AssembleLoadVector` and `ImplementRobinBC` yields the following linear system

$$\begin{pmatrix} K_{00} & K_{01} & K_{02} \\ K_{10} & K_{11} & K_{12} \\ K_{20} & K_{21} & K_{22} \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} f_0 \\ f_1 \\ f_2 \end{pmatrix}$$

and that we want to impose the Dirichlet boundary condition $u_0 = u(x_0) = g_D(x_0) = g_0$. In this case, we can replace the first equation by $K_{00}u_0 = K_{00}g_0$ and substitute u_0 by g_0 in the remaining equations. The modified system reads

$$\begin{pmatrix} K_{00} & 0 & 0 \\ 0 & K_{11} & K_{12} \\ 0 & K_{21} & K_{22} \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} K_{00}g_0 \\ f_1 - K_{10}g_0 \\ f_2 - K_{20}g_0 \end{pmatrix}.$$

$\boxed{22}$ Let

$$\widehat{K} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}, \qquad \widehat{M} = \begin{pmatrix} 1/3 & 1/6 \\ 1/6 & 1/3 \end{pmatrix}, \qquad \widehat{D} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Show that

$$\frac{1}{6}\widehat{D} \le \widehat{M} \qquad \text{and} \qquad \widehat{K} \le 2\widehat{D}.$$

$\boxed{23}$ Consider the one-dimensional boundary value problem

$$\begin{aligned} -u''(x) &= f(x) \qquad x \in (0,\,1) \\ u(0) &= g_0 \\ u(1) &= g_1\,. \end{aligned}$$

Let K_h denote the stiffness matrix obtained by the finite element method using the Courant elements on a subdivision $0 = x_0 < x_1 < \cdots < x_{N_h} = 1$.

Show that

$$\frac{\min_k h_k^2}{6c_F^2} D_h \le K_h \le D_h\,,$$

where $D_h = \operatorname{diag}(K_h)$, c_F is the constant arising in Friedrichs' inequality, and $h_k = x_k - x_{k-1}$.

Hint: Use

$$(D_h\,\underline{v}_h,\,\underline{v}_h) = D_h^{(1)} v_1^2 + \sum_{k=2}^{N_h} \left(D_h^{(k)} \begin{pmatrix} v_{k-1} \\ v_k \end{pmatrix}, \begin{pmatrix} v_{k-1} \\ v_k \end{pmatrix} \right)_{\ell_2}$$

with

$$D_h^{(1)} = K_h^{(1)} = \frac{1}{h_1} \qquad \text{and} \qquad D_h^{(k)} = \operatorname{diag}(K_h^{(k)}) = \frac{1}{h_k}\operatorname{diag}(\widehat{K}) = \frac{1}{h_k}\widehat{D}\,.$$

$\boxed{24}$ Let M_h denote the mass matrix (see your lecture notes) obtained by the finite element method using the Courant elements on a uniform mesh $0 = x_0 < x_1 < \ldots x_{N_h} = 1$, with $h_k = x_k - x_{k-1} = h$ for all $k = 1, \ldots, N_h$.

Show that the condition number $\kappa(M_h) = \operatorname{cond}_2(M_h) = \mathcal{O}(1)$ for $h \to 0$.