FEBioHeat

Theory Manual

1 Introduction

The heat transfer equation is defined as follows.

$$\rho C \frac{\partial u}{\partial t} + \nabla \cdot \mathbf{q} = Q \tag{1.1}$$

Here, u is the *temperature*, ρ the *density*, C the *specific heat*, $\mathbf{q}$ the *heat flux*, and Q the *heat source*. Furthermore, the following boundary conditions are assumed.

$$\begin{aligned} u &= u_0 && \text{on } \Gamma_D \\ \mathbf{q} \cdot \mathbf{n} &= q_0 && \text{on } \Gamma_N \end{aligned} \tag{1.2}$$

Furthermore, it is assumed that the heat flux is defined via Fourier's law,

$$\mathbf{q} = -k \nabla u \tag{1.3}$$

where k is the thermal conductivity.

2 Weak Formulation

To construct the weak form, first multiply the left- and right-hand side with an arbitrary function w (that satisfies the homogenous boundary conditions, i.e. $w = 0$ on Γ_D).

$$w\left(\rho C \frac{\partial u}{\partial t} + \nabla \cdot \mathbf{q} \right) = wQ \tag{1.4}$$

Then, multiply over the problem domain.

$$\int_\Omega w\left(\rho C \frac{\partial u}{\partial t} + \nabla \cdot \mathbf{q} \right) d\Omega = \int_\Omega wQ d\Omega \tag{1.5}$$

Using integration by parts and the boundary conditions, we can rewrite the second term on the left-hand side.

$$\int_\Omega w \rho C \frac{\partial u}{\partial t} d\Omega + \int_{\Gamma_N} w q_0 d\Gamma - \int_\Omega \nabla w \cdot \mathbf{q} d\Omega = \int_\Omega wQ d\Omega \tag{1.6}$$

Using equation (1.3) results in,

$$\boxed{\int_\Omega w \rho C \frac{\partial u}{\partial t} d\Omega + \int_\Omega k \nabla w \cdot \nabla u d\Omega = \int_\Omega wQ d\Omega - \int_{\Gamma_N} w q_0 d\Gamma} \tag{1.7}$$

For steady-state analysis, we simply omit the first term.

$$\boxed{\int_{\Omega} k\nabla w \cdot \nabla u d\Omega = \int_{\Omega} wQ d\Omega - \int_{\Gamma_N} w q_0 d\Gamma} \text{ (steady-state analysis)} \tag{1.8}$$

3 Discretization

The domain is discretized using iso-parametric elements.

$$\begin{aligned} \mathbf{r} &= \sum_a N_a \mathbf{r}_a \\ u &= \sum_a N_a u_a \\ w &= \sum_a N_a w_a \end{aligned} \tag{1.9}$$

Considering first the simpler case of steady state, the discretized equation becomes the following.

$$\sum_{a,b} w_a \left(\int_{\Omega} k\nabla N_a \cdot \nabla N_b d\Omega \right) u_b = \sum_a w_a \left(\int_{\Omega} N_a Q d\Omega \right) - \sum_a w_a \left(\int_{\Gamma_N} N_a^{'} q_0 d\Gamma \right) \tag{1.10}$$

Remark 2.1. Note that in the second term on the right-hand side, the shape functions are restricted to the boundary surface.

Remark 2.2. Equation (1.10) needs to be satisfied for every value of w_a.

This can be written more concisely.

$$\mathbf{Ku} = \mathbf{F} \tag{1.11}$$

Here, **K** is the *stiffness matrix* and is calculated from,

$$K_{ab} = \int_{\Omega} k\nabla N_a \cdot \nabla N_b d\Omega \tag{1.12}$$

Note that the stiffness matrix is *symmetric*.

The vector **u** collects all the unknown temperature values, and **F** collects all the known force loads,

$$F_a = \int_{\Omega} N_a Q d\Omega - \int_{\Gamma_N} N_a^{'} q_0 d\Gamma \tag{1.13}$$

Note again the slight abuse of notation, since the second term is evaluated over a surface, using different shape functions than the first term.

To include prescribed boundary conditions, we partition the system of equations as follows,

$$\begin{bmatrix} \mathbf{K}_{ff} & \mathbf{K}_{fp} \\ \mathbf{K}_{pf} & \mathbf{K}_{pp} \end{bmatrix} \begin{bmatrix} \mathbf{u}_f \\ \mathbf{u}_p \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \mathbf{0} \end{bmatrix} \tag{1.14}$$

Here, $\mathbf{u}_f$ are the *free* degrees of freedom (i.e. the unknown), and $\mathbf{u}_p$ are the known prescribed degrees of freedom ($\mathbf{u}_p = \mathbf{u}_0$). We can reduce the system of equations to,

$$\mathbf{K}_{ff}\mathbf{u}_f = \mathbf{F} - \mathbf{K}_{fp}\mathbf{u}_0 \tag{1.15}$$

Remark 2.3. In FEBio, the degrees of freedom of *prescribed* degrees of freedom are not eliminated from the global system of equations. Instead, the linear system is transformed to an equivalent system of equations.

$$\begin{bmatrix} \mathbf{K}_{ff} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{u}_f \\ \mathbf{u}_p \end{bmatrix} = \begin{bmatrix} \mathbf{F} - \mathbf{K}_{fp}\mathbf{u}_0 \\ \mathbf{u}_0 \end{bmatrix} \tag{1.16}$$

Note that *zero temperature* degrees of freedom (i.e. those dofs for which $u_0 = 0$) are eliminated from the global system of equations, and thus the corresponding stiffness terms are ignored during assembly.

Returning now to the transient case, we still need to discretize the term,

$$\int_{\Omega} w\rho C\dot{u}\, d\Omega \tag{1.17}$$

Here, we use $\dot{u}$ to denote the time derivative of the temperature. Introducing shape functions, similar as before,

$$\dot{u} = \sum_a N_a \dot{u}_a \tag{1.18}$$

The discretized integral then becomes,

$$\sum_{a,b} w_a \left(\int_{\Omega} \rho C N_a N_b \, d\Omega \right) \dot{u}_b = \sum_{a,b} w_a M_{ab} \dot{u}_b \tag{1.19}$$

where,

$$M_{ab} = \int_{\Omega} \rho C N_a N_b \, d\Omega \tag{1.20}$$

is the *capacitance matrix*.

Bringing all the terms together results in the following equation,

$$\mathbf{M}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{F} \tag{1.21}$$

This is the *semi-discrete* equation. Solving it will be discussed below.

4 Stiffness Matrix

The stiffness matrix is evaluated by first evaluating the element stiffness matrices, i.e. the restriction of the stiffness matrix to a single element and then assembling this element contribution into the global stiffness matrix.

$$K_{ab}^{e} = \int_{\Omega^e} k\nabla N_a \cdot \nabla N_b d\Omega \tag{1.22}$$

To evaluate this integral, we need to calculate the spatial gradient of the shape functions. However, shape functions are more often defined in the parametric space of the element, and so it is more convenient to perform the integral in this parametric space.

$$K_{ab}^{e} = \int_{\square^e} k\nabla N_a \cdot \nabla N_b J d\xi \tag{1.23}$$

Here, J is the determinant of the Jacobian matrix $\mathbf{J}$,

$$\left[\mathbf{J}\right]_{ij} = \frac{\partial x_i}{\partial \xi_j} \tag{1.24}$$

To evaluate the Jacobian, we make use of the element shape functions,

$$J_{ij} = \frac{\partial x_i}{\partial \xi_j} = \sum_a \frac{\partial N_a(\xi)}{\partial \xi_j} x_{ai} \tag{1.25}$$

To evaluate the spatial gradient of the shape functions, we proceed similarly.

$$\frac{\partial N_a}{\partial x_i} = \frac{\partial N_a}{\partial \xi_j}\frac{\partial \xi_j}{\partial x_i} \tag{1.26}$$

We can write this more concisely,

$$\nabla N_a = \mathbf{J}^{-T}\nabla_{\xi} N_a \tag{1.27}$$

5 Time Discretization

We now turn our attention to solving the semi-discrete equation, which is repeated here.

$$\mathbf{M}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{F} \tag{1.28}$$

We'll solve this equation using the *implicit-Euler* method (also known as *backward difference*). This method is unconditionally stable and first-order accurate.

First, we replace $\dot{\mathbf{u}}$ with a finite difference approximation.

$$\dot{\mathbf{u}} = \frac{1}{\Delta t}\left(\mathbf{u}^{n+1} - \mathbf{u}^{n}\right) \tag{1.29}$$

Then, we solve equation (1.28) at time point *n+1*.

$$\left(\frac{1}{\Delta t}\mathbf{M} + \mathbf{K}\right)\mathbf{u}^{n+1} = \mathbf{F} + \frac{1}{\Delta t}\mathbf{M}\mathbf{u}^{n} \tag{1.30}$$

Once again, we recover a linear system of equations that can be solved.