MASTER OF SCIENCE IN AERONAUTICS AND SPACE

Aeronautical, Mechanics and Energetics

Author:

SOWMIYA AYYAPPAN, Tejas Shripal

ZHANG, Di

Professor:

BENSELAMA, Adel Mustapha

INDEX

List of Figures

Nomenclature

C_p	Specific Heat Capacity
α	Thermal Diffusivity
ρ	Density
Δt	Time Step
Δr	Space Step
K	Thermal conductivity
r	Radius
T	Temperatures
t	Time

1. Objective

FEM method is a mathematic method for solving physics and engineering problem. It divides a whole system into smaller parts called “elements”, and solving each of them to finally achieve a general solution.

In this project, we consider a brake disk that is heated by friction. The main goal of this project is using FEM method to solve the heat equation of the brake disk, and get the heat distribution of the disk at any given time.

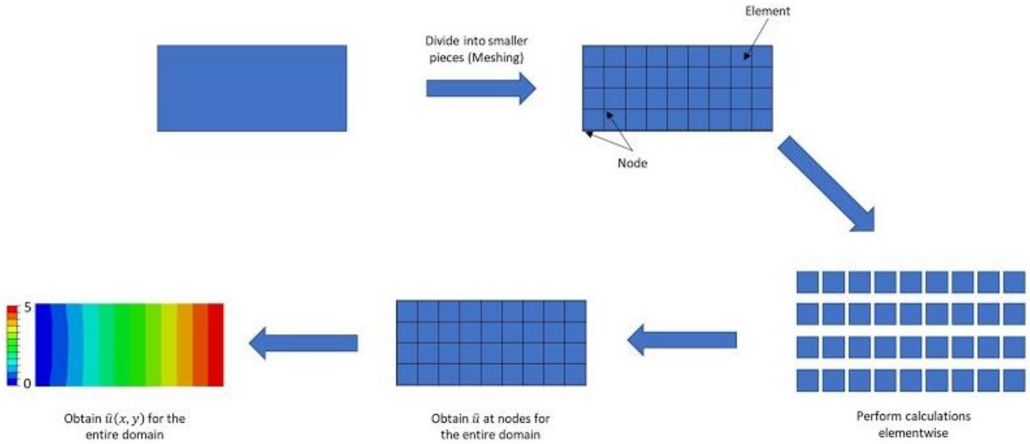

Figure 1 FEM process

2. Problem Description

In this case, we are considering a homogeneous brake disk with a radius of a, heaten by friction. At beginning (t=0), the whole disk is at the initial temperature of T_0.

At the edge of this disk, a Newton condition is applied, which can be represented as:

$$\varphi = h(T - T_\infty)$$

Where h is the heat exchange coefficient with the environment.

As the brake disk is experiencing a heat, a source term has to be considered.

The problem is defined in cylindrical coodinates. However, as there is no axial transfer occurs, we can simplify this problem to a 1D heat transfer system. That means to all points of the same r, the temperature is equal, and the angle θ doesn't affect the temperature. Also, in this case, we consider only heat conduction in the disk brake.

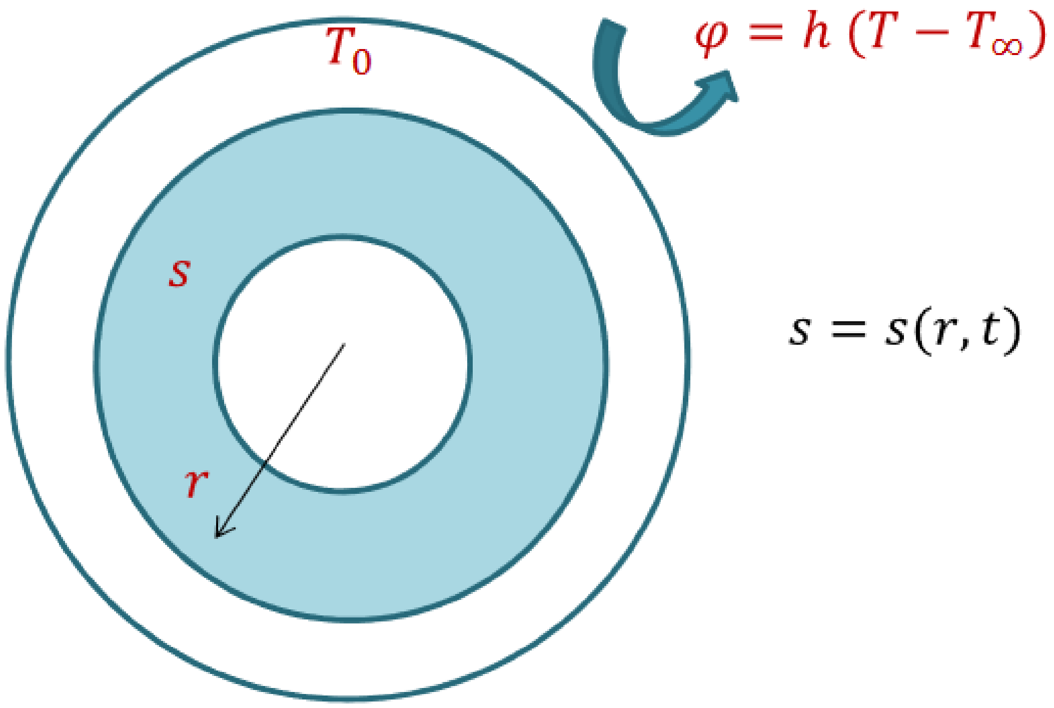

Figure 2 Problem

3. Numerical Formulation (FEM)

For this question, we can introduce the heat equation as following:

$$\rho C_p \frac{dT}{dt} = \frac{k}{r}\frac{d}{dr} r \frac{dT}{dr} + s$$

Where $\boldsymbol{\rho}$ is the density of the disk brake, $\boldsymbol{C_p}$ is the specific heat capacity, **S** is the source term, **K** is the thermal conductivity of the disk brake.
With the initial condition: the brake disk at t=0 has a temperature T_0, which means

$$T(r, \theta, z, t = 0) = T_0$$

3.1 Boundary conditions:

- Newton condition at the edge: $\varphi = h(T - T_\infty)$ at $r = a$
- The temperature of the air is T_∞, which means

$$T(r = a, \theta, z, t) = -\frac{2\pi h}{k}(T_n - T_\infty)$$

$r = \frac{a}{N}$, $\Delta t = 0.01s$, $t = 1s$

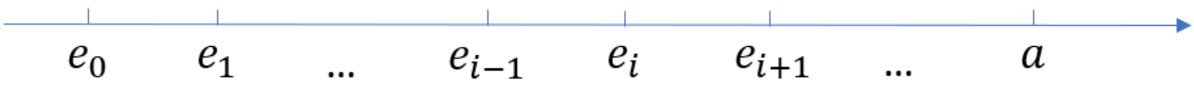

Figure 3 Mesh

With the following weight functions:

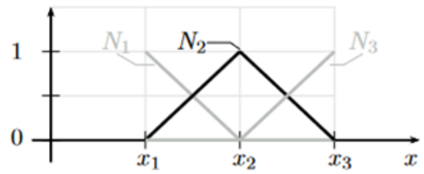

Figure 4 Weight function

$$N_i(r) = \begin{cases} \frac{r - r_{i-1}}{\Delta r}; r_{i-1} \le r \le r_i \\ -\frac{r - r_{i+1}}{\Delta r}; r_i \le r \le r_{i+1} \\ 0 \quad ; \ elsewhere \end{cases}$$

$$N_{i-1}(r) = \begin{cases} \frac{r - r_{i-2}}{\Delta r}; r_{i-2} \le r \le r_{i-1} \\ -\frac{r - r_i}{\Delta r}; r_{i-1} \le r \le r_i \\ 0 \quad ; \ elsewhere \end{cases}$$

$$N_{i+1}(r) = \begin{cases} \dfrac{r - r_i}{\Delta r}; r_i \leq r \leq r_{i+1} \\ -\dfrac{r - r_{i+2}}{\Delta r}; r_{i+1} \leq r \leq r_{i+2} \\ 0 \quad ;\ elsewhere \end{cases}$$

We apply WRM method as following:

$$R = \rho C_p \frac{dT}{dt} - \frac{k}{r}\frac{d}{dr} r \frac{dT}{dr} - S$$

$$\int_D N_i \, R dr = 0$$

$$\int_D N_i \, R r dr d\theta dz = 0$$

$$\int_{r=0}^{a} \int_{\theta=0}^{2\pi} \int_{r=0}^{L} NiRrdrd\theta dz = 0$$

$$2\pi \int_0^a N_i \, R r dr = 0$$

$$2\pi \int_0^a N_i \, (\rho C_p \frac{dT}{dt} - \frac{k}{r}\frac{d}{dr} r \frac{dT}{dr} - S) r dr = 0$$

3.2 Mass term

For the first term,

$$\int_0^a N_i \rho C_p \frac{dT}{dt} r dr$$

Using the approximation function,

$$T(r,t) = \sum_{j=0}^{N} N_j T_j$$

We have:

$$\int_0^a N_i \rho C_p \frac{d\left(\sum_{j=0}^{N} N_j T_j\right)}{dt} r dr$$

$$= \sum_{j=0}^{N} \int_0^a N_i N_j \rho C_p \frac{dT_j}{dt} r dr$$

$$= \sum_{j=0}^{N} \left(\int_0^a N_i N_j \rho C_p\, r dr\right) \frac{dT_j}{dt}$$

$$= \rho C_p \sum_{j=i-1}^{i+1} \frac{dT_j}{dt} \int_{r_{i-1}}^{r_{i+1}} N_i N_j r dr$$

We obtain the **Mass term** as follows:
First element –

$$= \frac{dT_i}{dt}\left[\frac{-(r_i - r_{i+1})^3(3r_i + r_{i+1})}{12\Delta r^2}\right] + \frac{dT_{i+1}}{dt}\left[\frac{-(r_i - r_{i+1})^3(r_i + r_{i+1})}{12\Delta r^2}\right]$$

$$= \frac{dT_i}{dt}\left[\frac{(3r_i + r_{i+1})\Delta r}{12}\right] + \frac{dT_{i+1}}{dt}\left[\frac{\Delta r(r_i + r_{i+1})}{12}\right]$$

For $n = 2, n-1$,

$$\rho C_p\left[\frac{dT_{i-1}}{dt}\left(\frac{1}{3} r_i \Delta r\right) + \frac{dT_i}{dt}\left(\frac{2}{3} r_i \Delta r\right) + \frac{dT_{i+1}}{dt}\left(\frac{(r_i + r_{i+1})}{12}\Delta r\right)\right]$$

Last element –

$$\rho C_p\left[\frac{dT_{i-1}}{dt}\left(\int_{r_{i-1}}^{r_i} N_i N_{i-1}\, r dr\right) + \frac{dT_i}{dt}\left(\int_{r_{i-1}}^{r_i} N_i^2\, r dr\right)\right]$$

$$\frac{dT_{i-1}}{dt}\left(\frac{\Delta r(r_{i-1} + r_i)}{12}\right) + \frac{dT_i}{dt}\left(\frac{\Delta r(r_{i-1} + 3r_i)}{12}\right)$$

3.3 Diffusive term(stiffness):

First element –

$$-K\left[T_i\left(-\frac{2r_i + r_{i+1}}{6}\right) + T_{i+1}\left(-\frac{r_i + 2r_{i+1}}{6}\right)\right]$$

For $n = 2, n - 1,$

$$-K\left\{T_{i-1}\left(\frac{2r_{i-1} + r_i}{6}\right) + T_i\left[\left(\frac{r_{i-1} + 2r_i}{6}\right) - \left(\frac{2r_i + r_{i+1}}{6}\right)\right] + T_{i+1}\left[\frac{-(2r_{i+1} + r_i)}{6}\right]\right\}$$

Last element –

$$-K\left[T_{i-1}\left(-\frac{2r_{i-1} + r_i}{6}\right) + T_i\left(\frac{r_{i-1} + 2r_i}{6}\right)\right] - h(T_i - T_\infty)$$

$$= -KT_{i-1}\left(-\frac{2r_{i-1} + r_i}{6}\right) - T_i\left[\left(\frac{r_{i-1} + 2r_i}{6}\right)K - h\right] + hT_\infty$$

Source term:

The matrix involved with the source vector is essentially the same as the mass matrix. We use the following approximation equation for the source term,

$$S(r,t) = \sum_{j=0}^{N} N_j S_j$$

3.4 Building of matrices:

Assembling all the FEM formulations for n elements i.e. n equations into matrices, we obtain:

$$\begin{bmatrix} \frac{(3r_i + r_{i+1})\Delta r}{12} & \frac{(r_i + r_{i+1})\Delta r}{12} & & \cdots & & 0 \\ \frac{1}{3} r_i \Delta r & \frac{2}{3} r_i \Delta r & \frac{(r_i + r_{i+1})\Delta r}{12} & & & \\ & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & \ddots \\ 0 & & \cdots & & \frac{(r_{i-1} + r_i)\Delta r}{12} & \frac{(r_{i-1} + 3r_i)\Delta r}{12} \end{bmatrix} * \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix}^{K+1} =$$

$$\left\{ \begin{bmatrix} \frac{(3r_i + r_{i+1})\Delta r}{12} & \frac{(r_i + r_{i+1})\Delta r}{12} & & \cdots & & 0 \\ \frac{1}{3} r_i \Delta r & \frac{2}{3} r_i \Delta r & \frac{(r_i + r_{i+1})\Delta r}{12} & & & \\ & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & \ddots \\ 0 & & \cdots & & \frac{(r_{i-1} + r_i)\Delta r}{12} & \frac{(r_{i-1} + 3r_i)\Delta r}{12} \end{bmatrix} + \begin{bmatrix} \beta \frac{(2r_i + r_{i+1})}{6} & \beta \frac{(r_i + 2r_{i+1})}{6} & & \cdots & & 0 \\ -\beta \frac{(2r_{i-1} + r_i)}{6} & \beta \frac{\Delta r}{3} & -\beta \frac{(2r_{i+1} - r_i)}{6} & & & \\ & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & \ddots \\ 0 & & \cdots & & -\beta \frac{(2r_{i-1} + r_i)}{6} & -\beta \left(\frac{(r_{i-1} + 2r_i)}{6} - h \right) \end{bmatrix} \right\} * \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix}^{K} +$$

$$\begin{bmatrix} \frac{(3r_i + r_{i+1})\Delta r}{12} & \frac{(r_i + r_{i+1})\Delta r}{12} & & \cdots & & 0 \\ \frac{1}{3} r_i \Delta r & \frac{2}{3} r_i \Delta r & \frac{(r_i + r_{i+1})\Delta r}{12} & & & \\ & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & \ddots \\ 0 & & \cdots & & \frac{(r_{i-1} + r_i)\Delta r}{12} & \frac{(r_{i-1} + 3r_i)\Delta r}{12} \end{bmatrix} * \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_n \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ \\ hT_\infty \Delta t \end{bmatrix},$$

where $\beta = \frac{K\Delta t}{\rho C_p}$ or $\beta = \alpha \Delta t$

4. Calculation Process

After obtaining the matrices, we must now focus on employing the computer to solve these matrices in the form AX = B iteratively.

The Python program associated with this report consists of the following:

- **Using of function tridiagonal_matrix(lower, middle, upper)**: We create tridiagonal matrices with the lower, middle and upper diagonal elements.

- **Variables definition:** We define the physical variables like density (ρ, rhow), conductivity constant (k), specific heat capacity (C_p, cp), the condition variables like initial temperatures (t_0, t0) and surrounding temperatures (T_∞, tinf), and the modelling variables like number of elements(n), time step(del_t), space step(dr), and maximum time in seconds(t_max).

- **Construction of matrix**: We build matrices for numerical application using function of tridiagonal_matrix.

- **Definition of boundary conditions:** We use the array named BC to represent boundary conditions.

- **Iteration loop**: We start the calculation using a loop function. This is the main function controlling the emulation process. It runs from t = 0 until the defined t_max.

- **File output**: We finally have an array of temperature distribution as a text file in the "results" directory. This is obtained for every iteration.

The python script related to this report can be found in the zip file uploaded.

5. Conclusions

In this project, we applied the Finite Element Method (FEM) to disc braking problem. The theoretical foundations were carefully developed, and the corresponding matrices were meticulously programmed. However, the results obtained from our computational implementation did not align with the expected outcomes. This deviation highlights the intricate challenges inherent in numerical modeling and simulation, especially in transient heat transfer problems.

Several factors might have contributed to this discrepancy. Firstly, the complexity of the FEM, particularly in the formulation of the FEM equations, might have introduced errors. With our algorithms were rigorously tested, it was clear that the problem aroused from the theoretical formulation of the FEM equations on paper, which could have significantly impacted the results.

This experience, albeit challenging, has been immensely educational. It underscores the importance of thorough verification and validation in computational modeling. For future work, we recommend a detailed review and debugging of the code, perhaps by employing alternative programming tools or collaborating with peers for a fresh perspective. Additionally, a closer examination of the model's assumptions and a comparison with simpler models or analytical solutions could provide insights into the sources of error.

In conclusion, while our results were not as anticipated, the learnings from this endeavor are invaluable. They not only offer an opportunity for academic growth but also contribute to a deeper understanding of the complexities and nuances of applying FEM in practical scenarios. This project lays the groundwork for further investigation and refinement, which is vital in the field of computational sciences.