Finite Element Method for Heat Transfer

Rezgar Shakeri Leila Ghaffari

University of Colorado Boulder

Rezgar.Shakeri@colorado.edu Leila.Ghaffari@colorado.edu

December 7, 2020

Outline

1. Heat Equation
2. Finite Element Method (FEM)
3. Method of Manufactured Solutions (MMS)
   - Primary Results
4. Future Work

Heat Equation

- Predicts the temperature of the body (domain) subjected to the heat source
- Derived from Fourier's law and the conservation of energy
- Of fundamental importance in diverse scientific fields

Fourier's Law $\rightarrow \boldsymbol{q}$ = -k ∇u

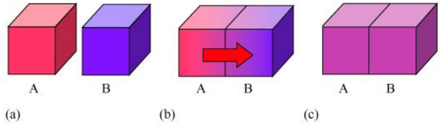


Figure: Heat conduction

Heat Equation

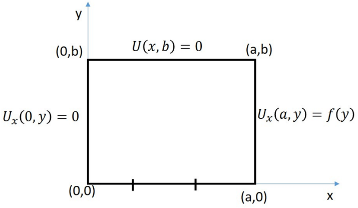
$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u = \alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right), \tag{1}$$

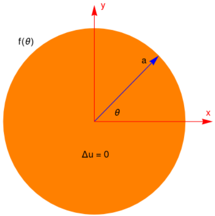
$u(x, y, z, t) =$ Temperature

$\alpha = \frac{k}{c_p \rho} =$ Thermal Diffusivity $\quad c_p =$ Specific Heat Capacity

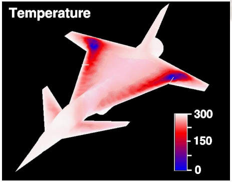
$k =$ Thermal conductivity $\quad \rho =$ Mass density

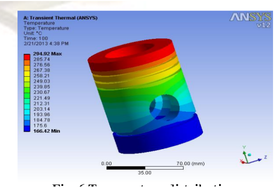
We can solve (1) analytically in **simple geometries** like rectangular (cubic in 3D) or circular by the **Separation of Variables** method.





But how about the complex region? Like aircraft, blade of the turbine?!





Finite Element Method (FEM)

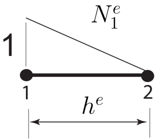
- Divide the domain to small pieces called **Element**
- Implement the weak form of **Heat Equation** for each element
- Expand the unknown variable (here temperature) using shape functions at each element
- Assemble the general form of the equation and find the solution

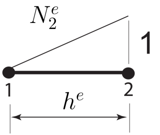
$$
\begin{aligned}
&\frac{\partial^2 u}{\partial x^2} = f(x) \Longrightarrow \int_0^L w(x)\left[\frac{\partial^2 u}{\partial x^2} - f(x)\right] dx = 0 \\
&\left(w\frac{\partial u}{\partial x}\right)_0^L - \int_0^L \frac{\partial w}{\partial x}\frac{\partial u}{\partial x} dx - \int_0^L w(x) f(x) dx = 0
\end{aligned} \tag{2}
$$

Above equation is correct for each piece of the domain (element)

Finite Element Method (FEM)

$$
\begin{aligned}
u^{h^e}(x) = \sum_{a=1}^{n_{\mathrm{en}}} N_a^e(x) d_a^e &= N_1^e(x) d_1^e + N_2^e(x) d_2^e \\
&= \begin{bmatrix} N_1^e & N_2^e \end{bmatrix} \begin{bmatrix} d_1^e \\ d_2^e \end{bmatrix} \\
&= \boldsymbol{N}^e(x) \cdot \boldsymbol{d}^e
\end{aligned}
$$





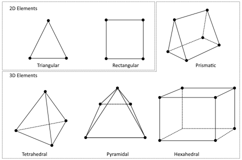


Figure: Different types of element and example of the Piston mesh

Method of Manufactured Solutions (MMS)

Let's say we are interested in finding a numerical solution, v, of the following initial-boundary-value-problem

$$\begin{aligned}
&\frac{\partial u^2}{\partial t^2} = \frac{\partial u^2}{\partial x^2} + f, \quad 0 \le x \le 1,\ t \ge 0,\\
&u(0,t) = g_1(t), \quad u(1,t) = g_2(t) \quad t \ge 0,\\
&u(x,0) = u_\circ(x), \quad u_t(x,0) = v_\circ(x), \quad 0 \le x \le 1,
\end{aligned} \tag{3}$$

by implementing FEM.

How can we convince ourselves (or our advisor/instructor) that the code we implemented is correct?

Method of Manufactured Solutions (MMS)

Simple answer: We compute the error and check if you get the expected convergence rate, $O(h^r)$.

$$\varepsilon(t) = \sqrt{\sum_{n=1}^{N} (u(x_n, t) - v_n)^2}, \tag{4}$$

Method of Manufactured Solutions (MMS)

Example: Consider $f(x) = x^2$, let's find the error of the first derivative using the Forward Euler method at $x_n = 1$ using different mesh size h.

h	$f' = 2x$	$\frac{f(x_{n+1}) - f(x_n)}{h}$	$\varepsilon \approxeq O(h)$
1	2.0	$\frac{f(2) - f(1)}{h} = 3.0$	1.00
0.5	2.0	$\frac{f(1.5) - f(1)}{h} = 2.5$	0.50
0.25	2.0	$\frac{f(1.25) - f(1)}{h} = 2.25$	0.25

Method of Manufactured Solutions (MMS)

Computing the error, ε, requires the knowledge of exact solution, which we never know!
This is where the Method of Manufactured Solutions comes in.
Lets start with a simple example.

$$Ax = b \Longrightarrow x = A^{-1}b$$

how can we check the algorithm that we wrote for A^{-1} is correct?

$$A * 1 = b_\circ$$

then

$$Ax = b_\circ \Longrightarrow x = A^{-1}b_\circ$$

now you can easily compute the error $\varepsilon = \|x - 1\|$

Method of Manufactured Solutions (MMS)

To do MMS for equation (3) we start with this solution,

$$u(x,t) = \sin(\omega t - kx), \tag{5}$$

How should we adjust (3) so that (5) holds? We just need to choose the forcing, initial and boundary values we get by plugging (5) in (3). For example here,

$$\begin{aligned} f &= \frac{\partial u^2}{\partial t^2} - \frac{\partial u^2}{\partial x^2} = (k^2 - \omega^2)\sin(\omega t - kx) \\ u(0,t) &= \sin(\omega t), \quad u(1,t) = \sin(\omega t - k) \\ u(x,0) &= \sin(-kx), \quad u_t(x,0) = \omega\cos(-kx), \end{aligned} \tag{6}$$

Method of Manufactured Solutions (MMS)

Now, if we rewrite (3) with the above forcing and initial and boundary conditions, we will have

$$\begin{aligned} &\frac{\partial u^2}{\partial t^2} = \frac{\partial u^2}{\partial x^2} + (k^2 - \omega^2)\sin(\omega t - kx), \quad 0 \le x \le 1,\ t \ge 0, \\ &u(0,t) = \sin(\omega t), \quad u(1,t) = \sin(\omega t - k) \quad t \ge 0, \\ &u(x,0) = \sin(-kx), \quad u_t(x,0) = \omega\cos(-kx), \quad 0 \le x \le 1, \end{aligned} \tag{7}$$

We definitely know the exact solution is $u(x,t) = \sin(\omega t - kx)$ and simply computed the error, ε, and see whether the code is correctly implemented or not.

Primary Results

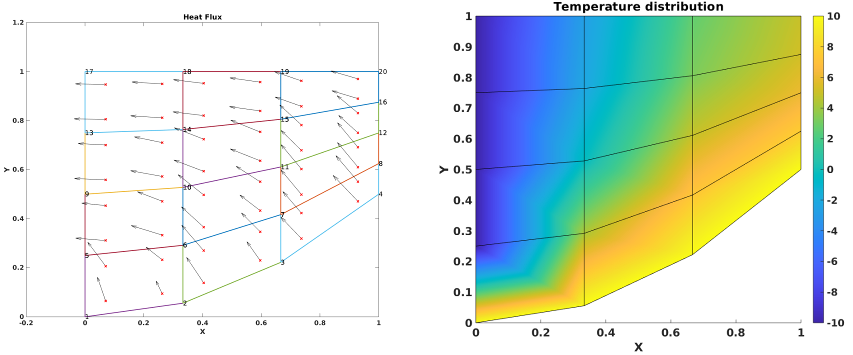


Figure: Heat flux and temperature distribution, bottom and left edges are subjected to the 10 and -10 boundary conditions. Right and top edges are insulated.

first we found temperature by solving $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$ then we computed flux $\boldsymbol{q}$ = -k ∇u

Future Work

- Transfer existing MATLAB code to Python
- Implement MMS
- Add unit and functional tests

Question