doThe Wave Equation

$$\frac{\partial^2 u}{\partial t^2} = c^2 \frac{\partial^2 u}{dx^2}, \quad x \in [0, L], t \geq 0$$

Boundary conditions: Dirichlet, Neumann, Mixed or Robin.

For example:

$$u(0,t) = A, \quad u(L,t) = B$$

Initial conditions:

$$u(x,0) = f(x), \quad \frac{\partial u}{dt}(x,0) = g(x)$$

Applying central difference spatially and temporally,

$$\frac{\partial^2 u}{dt^2} = c^2 \frac{\partial^2 u}{dx^2}$$

Becomes:

$$\frac{u_n^{k+1} - 2u_n^k + u_n^{k-1}}{\Delta t^2} = c^2 \frac{u_{n+1}^k - 2u_n^k + u_{n-1}^k}{\Delta x^2} + O(\Delta t^2, \Delta x^2)$$

Neglecting big O, and solving for u_n^{k+1}:

$$u_n^{k+1} = 2u_n^k - u_n^{k-1} + c^2 \frac{\Delta t^2}{\Delta x^2} \left(u_{n+1}^k - 2u_n^k + u_{n-1}^k\right)$$

Note that 2 time-steps, k and $k-1$, must be determined before the above iteration scheme can be applied.

Stability conditions:

$$c \frac{\Delta t}{\Delta x} \leq 1$$

Boundary conditions: dealt with the same way as the Heat Equation.

Initial conditions: $u(x,0) = f(x)$ and $\frac{\partial u}{\partial t}(x,0) = g(x)$ are discretized in the temporal domain:

$$u_n^0 = f(x_n)$$

$$\frac{u(x, 0 + \Delta t) - u(x - \Delta t)}{2\Delta t} = g(x)$$

The latter leads to:

$$u_n^{-1} = u_n^1 - 2\Delta t g(x_n)$$

The iteration scheme:

$$u_n^{k+1} = 2u_n^k - u_n^{k-1} + c^2 \frac{\Delta t^2}{\Delta x^2}\left(u_{n+1}^k - 2u_n^k + u_{n-1}^k\right)$$

When $k = 0$ becomes:

$$u_n^1 = u_n^0 + \Delta t g(x_n) + c^2 \frac{\Delta t^2}{2\Delta x^2}(u_{n+1}^0 - 2u_n^0 + u_{n-1}^0)$$

The iteration steps: (for Dirichlet boundary conditions) assign initial condition u_n^0:

$u_0^1 \leftarrow A$
$u_N^1 \leftarrow B$
for $n = 1, \dots, N-1$
$\quad u_n^1 \leftarrow u_n^0 + \Delta t g(x_n) + c^2 \frac{\Delta t^2}{\Delta x^2}(u_{n+1}^0 - 2u_n^0 + u_{n-1}^0)$
end

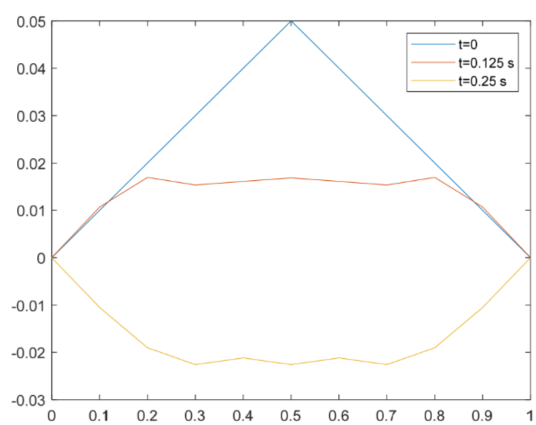
for $k = 1, \dots, K-1$
$\quad u_0^1 \leftarrow A$
$\quad u_N^1 \leftarrow B$
$\quad$for $n = 1, \dots, N-1$
$\quad\quad u_n^{k+1} \leftarrow 2u_n^k - u_n^{k-1} + c^2 \frac{\Delta t^2}{\Delta x^2}\left(u_{n+1}^k - 2u_n^k + u_{n-1}^k\right)$
$\quad$end
end

Example
$c = 3$
$L = 1$
$N = 10$
$\Delta t = 0.025\ sec$
$t \in [0\ ,10]$

$u(0,t) = 0, \qquad u(L,t) = 0$

$u(x,0) = f(x) = the\ blue\ line$

$\frac{\partial u}{\partial t}(x,0) = g(x) = f(x)$



An excel sheet will accompany this file. The sheet has results computed with $\Delta x = 0.1, \Delta t = 0.025$ for 10 time-steps.

Check against stability condition:

$$c\frac{\Delta t}{\Delta x} = 3\frac{0.025}{0.1} = 0.75 \le 1$$

The Poisson's Equation

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -F(x,y), \quad x \in [0,a], \; y \in [0,b]$$

If $F(x,y) = 0$, the Poisson's equation becomes the Laplace's equation. They are to describe the diffusion (or spread) of $F(x,y)$ (which may be, for example, a heat source, an electric charge, etc.) For the Lapalce's equation, one investigates the diffusion of boundary conditions.

Boundary conditions: Dirichlet, Neumann, Mixed or Robin.

Focusing on the Dirichlet boundary conditions:

$$u(x,0) = f_1(x), \qquad u(x,b) = f_2(x)$$

$$u(0,y) = g_1(y), \qquad u(a,y) = g_2(y)$$

Discretizing the rectangular spatial domain so that the node points (mesh points) are:

$$x_n = n\Delta x, \qquad n = 0, 1, \dots, N$$

$$y_m = m\Delta y, \qquad m = 0, 1, \dots, M$$

Assuming central for the second derivatives, the Poisson's equation:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -F(x,y)$$

becomes, denoting $u(x_n, y_m)$ by $u_{n,m}$

$$\frac{u_{n+1,m} - 2u_{n,m} + u_{n-1,m}}{\Delta x^2} + \frac{u_{n,m+1} - 2u_{n,m} + u_{n,m-1}}{\Delta y^2} + O(\Delta x^2, \Delta y^2) = -F(x,y)$$

Defining $\beta = \Delta x/\Delta y$, neglecting the big O, and solving for $u_{n,m}$:

$$u_{n,m} = \frac{1}{2(1+\beta^2)}\left[u_{n+1,m} + u_{n-1,m} + \beta^2 u_{n,m+1} + \beta^2 u_{n,m-1} + \Delta x^2 F(x,y)\right]$$

The problem with the above approach is, $u_{n+1,m}$ *and* $u_{n,m+1}$ are unknown. The scheme is therefore implicit.

There are a number of approaches.

- Direct Solution
- Jacobi Iteration
- Successive Over Relaxion (SOR)
- …

Direct Solution:

Put the $(M-1)*(N-1)$ unknowns in a vector **U;**

Each equation of:

$$u_{n,m} = \frac{1}{2(1+\beta^2)}\left[u_{n+1,m} + u_{n-1,m} + \beta^2 u_{n,m+1} + \beta^2 u_{n,m-1} + \Delta x^2 F(x_n, y_m)\right]$$

is a row in a matrix $\boldsymbol{A}$ and an element in vector $\boldsymbol{R}$**;**

$\boldsymbol{A} \cdot \boldsymbol{U} = \boldsymbol{R}$ is formed;

$\boldsymbol{U}$ is then solved.

Jacobi Iteration:

$$u_{n,m}^{(k+1)} = \frac{1}{2(1+\beta^2)}\left[u_{n+1,m}^{(k)} + u_{n-1,m}^{(k)} + \beta^2 u_{n,m+1}^{(k)} + \beta^2 u_{n,m-1}^{(k)} + \Delta x^2 F(x_n, y_m)\right]$$

Step 1:

Boundary nodes are assigned boundary conditions;

$$k = 0;$$

Interior nodes are assigned zero value, $u_{n,m}(0) \leftarrow 0$;

$$\boldsymbol{u}_{old} \leftarrow \boldsymbol{u}^{(0)};$$

Step 2:

Compute all interior nodes' values by evaluating $u_{n,m}^{(k+1)}$;

Compute $\Delta = \left|\left|\boldsymbol{u}^{(k+1)} - \boldsymbol{u}_{old}\right|\right|$;

Step 3:

If $\Delta \leq$ tolerance, $\mathbf{u}_{\text{old}} \leftarrow \mathbf{u}^{(k+1)}$, $k \leftarrow k+1$, go back to Step 2.

Successive Over Relaxation (SOR):

Point SOR:

From:

$$u_{n,m}^{(k+1)} = \frac{1}{2(1+\beta^2)}\left[u_{n+1,m}^{(k)} + u_{n-1,m}^{(k)} + \beta^2 u_{n,m+1}^{(k)} + \beta^2 u_{n,m-1}^{(k)} + \Delta x^2 F(x_n, y_m)\right]$$

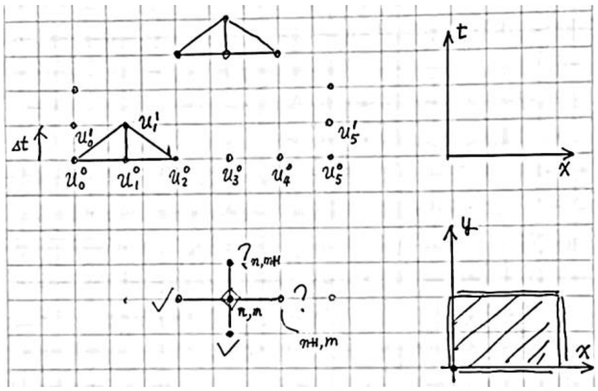
The SOR scheme is:

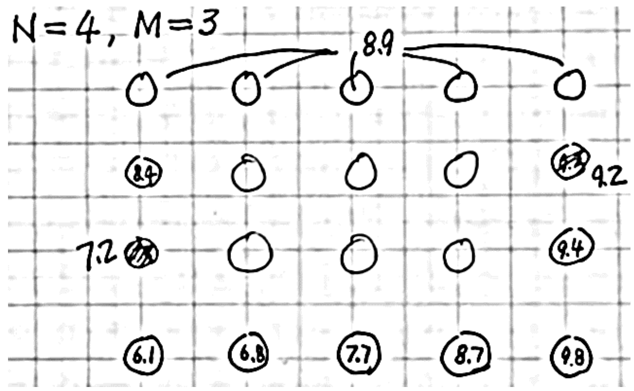
$$u_{n,m}^{(k+1)} = (1-w)u_{n,m}^{(k)} + \frac{w}{2(1+\beta^2)}\left[u_{n+1,m}^{(k)} + u_{n-1,m}^{(k)} + \beta^2 u_{n,m+1}^{(k)} + \beta^2 u_{n,m-1}^{(k)} + \Delta x^2 F(x_n, y_m)\right]$$

Where $1 < w < 2$ for over relaxation, and $0 < w < 1$ for under relaxation.

What is the best value to use for w? It depends.

There is also Line SOR.





Example:

$$\Delta x = \Delta y \quad \therefore \beta = 1; \quad 2(1+\beta^2) = 4;$$

$$F(x,y) = 0$$

$$\therefore -4U_{n,m} + U_{n-1,m} + U_{n+1,m} + U_{n,m-1} + U_{n,m+1} = -\Delta x^2 \cdot F(x_n, y_m) = 0$$

$$-4U_1 + 7.2 + U_2 + 6.8 + U_4 = 0$$

$$-4 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \qquad -14$$

$$\vdots$$

$$-4U_5 + U_4 + U_6 + U_2 + 8.9 = 0$$

$$0 \quad 1 \quad 0 \quad 1 \quad -4 \quad 1 \qquad -8.9$$

$$-4U_6 + U_5 + 9.2 + U_3 + 8.9 = 0$$

$$0 \quad 0 \quad 1 \quad 0 \quad 1 \quad -4 \qquad -18.1$$

$$\therefore \vec{U} = \begin{Bmatrix} 7.6391 \\ 8.1764 \\ 8.7858 \\ 8.3800 \\ 8.5807 \\ 8.8666 \end{Bmatrix}$$

Introduction

1. $\begin{cases} \text{Finite Element Method (FEM)} \\ \quad (for\ academia, \\ \quad\ software\ developers \ldots) \\ \text{Finite Element Analysis (FEA)} \\ \quad (for\ users) \end{cases}$

2. What is FEM/FEA?

Physically (physical systems' perspective)

The continuous physical model is divided into finite pieces (a.k.a. the elements), and the laws of nature/physics/chemistry are applied. The results are subsequently recombined to represent the continuum.

Mathematically,
The differential equation representing the system is converted into a variational form, which is approximated by the combination of a finite set of trial functions (a.k.a. shape functions).

It has been proven that ,as long as the elements meet certain conditions, then as the elements get smaller and smaller, the finite element result will converge to the "exact" solution.

4. Steps in FEA:

Discretization (Pre-processing):

- Divide the physical domain into pieces (or elements whose attributes are appropriate for the problem at hand)
- Constrain the mesh by appropriate boundary conditions
- Apply loads (forces, moments, temperature, pressure, …)

Solution:

- Solve: the system of equations

$$e.g.\quad [K]\{U\} = \{F\}$$
$$\{U\} = [K]^{-1}\{F\}$$

Post-processing:

- Calculate: displacements, strains, stresses, and plot results