

Project Report

one-dimensional heat conduction

Algorithm framework

In this project, the governing equation of one-dimensional heat conduction problem, the boundary conditions and the initial value can be expressed as

$$\rho c \frac{\partial u}{\partial t} - \kappa \frac{\partial^2 u}{\partial x^2} = f \quad \text{on } \Omega \times (0, T) \quad (1)$$

$$u = 0 \quad \text{on } \Gamma \times (0, T) \quad (2)$$

$$u|_{t=0} = u_0 = e^x \quad \text{in } \Omega. \quad (3)$$

Here, $\Omega := (0, 1)$ is the 1D domain. The boundary domain is $\Gamma = \{0, 1\}$. ρ , c , κ are the density, heat capacity and heat conductivity respectively. $f = \sin(\pi x)$ is the heat supply per unit volume.

Both explicit and implicit method is applied to solve this problem. The derivation of the two schemes are provided as follows.

Explicit Euler method

$$\begin{aligned} \rho c \frac{\partial u}{\partial t} - \kappa \frac{\partial^2 u}{\partial x^2} &= f \\ \frac{\partial u}{\partial t} &= \frac{\kappa}{\rho c} \frac{\partial^2 u}{\partial x^2} + \frac{f}{\rho c} \\ \frac{\partial u}{\partial t} &\sim \frac{U_i^{n+1} - U_i^n}{\Delta t} \\ \frac{\partial^2 u}{\partial x^2} &\sim \frac{U_{i-1}^n - 2U_i^n + U_{i+1}^n}{\Delta x^2} \\ \frac{U_i^{n+1} - U_i^n}{\Delta t} &= \frac{\kappa}{\rho c} \frac{U_{i-1}^n - 2U_i^n + U_{i+1}^n}{\Delta x^2} + \frac{f}{\rho c} \\ U_i^{n+1} - U_i^n &= \frac{\kappa \Delta t}{\rho c \Delta x^2} (U_{i-1}^n - 2U_i^n + U_{i+1}^n) + \frac{f \Delta t}{\rho c} \\ U_i^{n+1} &= \frac{\kappa \Delta t}{\rho c \Delta x^2} U_{i-1}^n + (1 - 2 \frac{\kappa \Delta t}{\rho c \Delta x^2}) U_i^n + \frac{\kappa \Delta t}{\rho c \Delta x^2} U_{i+1}^n + \frac{f \Delta t}{\rho c} \\ \mathbf{A} &= \mathbf{I} + \frac{\kappa \Delta t}{\rho c \Delta x^2} \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix} \\ \mathbf{U}^{n+1} &= \mathbf{A} \mathbf{U}^n \end{aligned}$$

Implicit Euler method

$$\begin{aligned}
\rho c \frac{\partial u}{\partial t} - \kappa \frac{\partial^2 u}{\partial x^2} &= f \\
\frac{\partial u}{\partial t} &= \frac{\kappa}{\rho c} \frac{\partial^2 u}{\partial x^2} + \frac{f}{\rho c} \\
\frac{\partial u}{\partial t} &\sim \frac{U_i^{n+1} - U_i^n}{\Delta t} \\
\frac{\partial^2 u}{\partial x^2} &\sim \frac{U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1}}{\Delta x^2} \\
\frac{U_i^{n+1} - U_i^n}{\Delta t} &= \frac{\kappa}{\rho c} \frac{U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1}}{\Delta x^2} + \frac{f}{\rho c} \\
U_i^{n+1} - U_i^n &= \frac{\kappa \Delta t}{\rho c \Delta x^2} (U_{i-1}^{n+1} - 2U_i^{n+1} + U_{i+1}^{n+1}) + \frac{f \Delta t}{\rho c} \\
&- \frac{\kappa \Delta t}{\rho c \Delta x^2} U_{i+1}^{n+1} + \left(1 + 2 \frac{\kappa \Delta t}{\rho c \Delta x^2}\right) U_i^{n+1} - \frac{\kappa \Delta t}{\rho c \Delta x^2} U_{i-1}^{n+1} = U_i^n + \frac{f \Delta t}{\rho c} \\
\mathbf{A} &= \mathbf{I} + \frac{\kappa \Delta t}{\rho c \Delta x^2} \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix} \\
\mathbf{A} \mathbf{U}^{n+1} &= \mathbf{U}^n
\end{aligned}$$

The convergence condition for explicit scheme is

$$\frac{\kappa \Delta t}{\rho c \Delta x^2} < \frac{1}{2},$$

and the implicit scheme always converges.

The steady state is characterized by $\partial u / \partial t = 0$. Thus, $u = \frac{\sin(\pi x)}{\pi^2}$ at steady state.

Error Analysis

Here, I focused on the value at the center of the domain, i.e. at $x=0.5$. In explicit scheme, the error of the data is related to the discretization of space and time:

$$e \approx C_1 \Delta x^\alpha + C_2 \Delta t^\beta. \quad (4)$$

To get the value of α , Δt is fixed at $5e-6$, and the domain is divided into 8, 16, 32, 64, 128, 256 parts respectively. The value of u changes little after 1.5 seconds, i.e., the steady state is roughly after 1.5 seconds. In this part, the iteration number is set 400000 to stop after 2 seconds. The corresponding error is defined as the difference between the numerical data and the theoretical data at $x=0.5$. Note that the theoretical data at $x=0.5$ at steady state is $1/(\pi^2)$. The results are plot in log form. As shown in Fig. 1, the slop of the line is around 2.001. Thus, the value of α is determined as 2.

To get the value of β , Δx is fixed at 0.005, Δt varies at $1e-6$, $5e-6$, $2.5e-6$, $1.25e-6$, $6.25e-7$, $3.125e-7$. Each sample stop at 0.1 second. The error is defined as the difference between each sample at the sample of $\Delta t = 3.125e - 7$. The results are plot in log form. As shown in Fig. 2, the slop of the line passing through the red points is 0.94. Thus, the value of β is determined as 1.

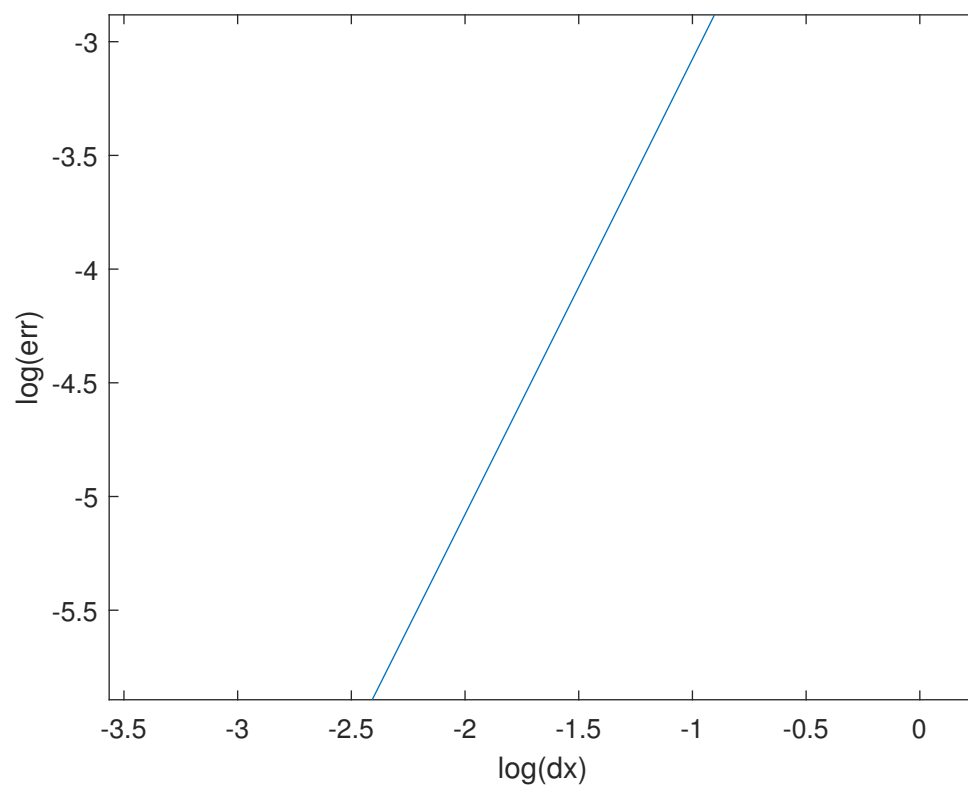


Figure 1: err versus dx

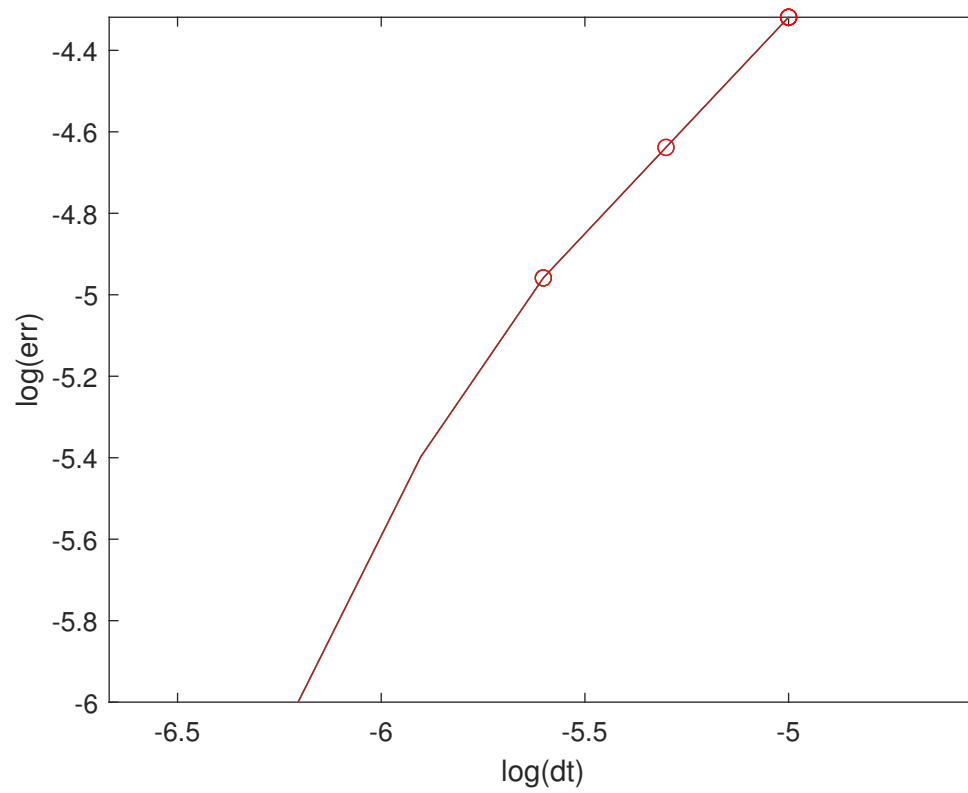


Figure 2: err versus dt

Parallelism

In this part, examples of parallel behaviours are presented from explicit scheme. To observe Amdahl's Law, the domain is divided into 40000 parts, and the processor number is set 1, 2, 4, 8 respectively. Fig. 3 shows the speed-up ratio with respect to processor number. The yellow line has a slope of 1 for reference.

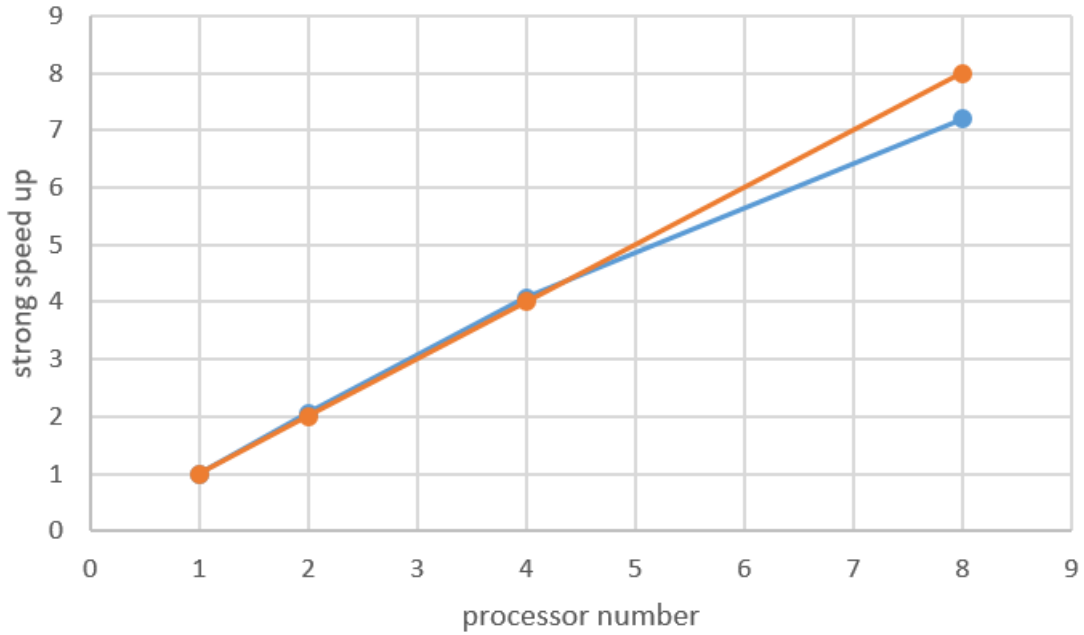


Figure 3: speed-up ratio versus processor number

To observe Gustafson's Law, division number is set 500, 1000, 1500, 2000, 2500, 3000 with processor number at 1, 4, 9, 16, 27, 36 respectively. Fig. 4 illustrates the time spent with respect to processor number. Time spent increase gradually as processor number increases. While the time cost using 4 processors is significant higher abnormally, which needs further discussion.

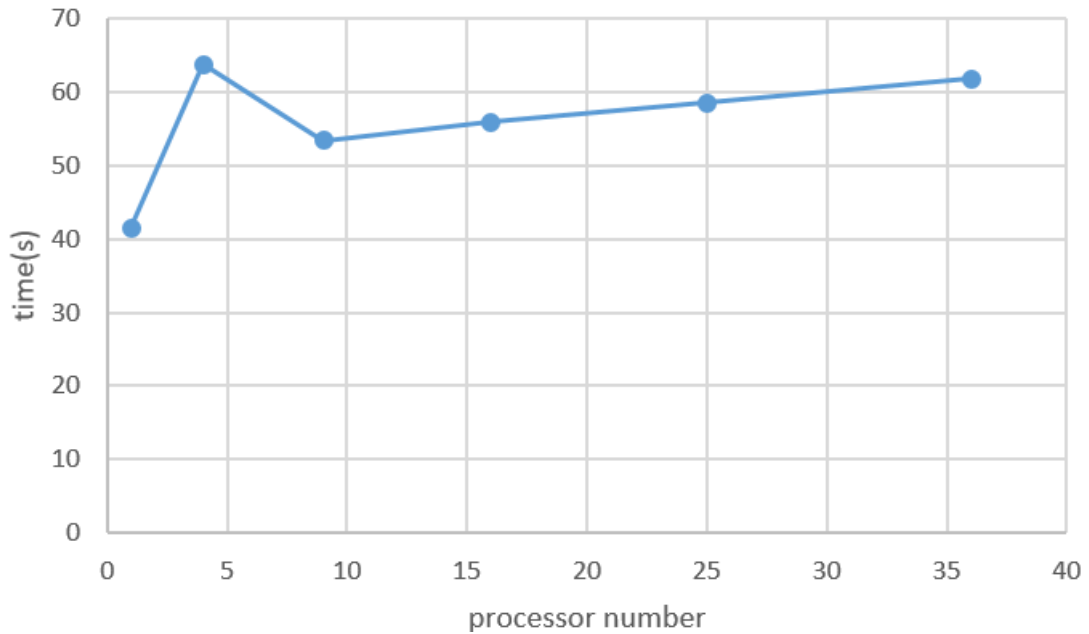


Figure 4: time spent versus processor number

Visualization

To visualize the results, output data is written in tecplot manner. Fig. 5 is from implicit scheme with matrix size of 1000 by 1000.

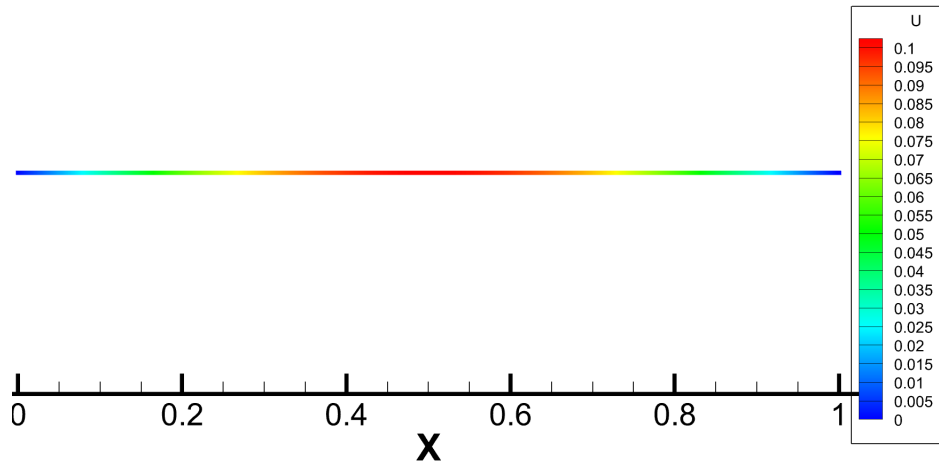


Figure 5: result shown by tecplot. X is the coordinate