Numerical treatment of the heat equation

Explicit method

Numerical treatment of heat equation

Initial and boundary value problem of 1-dim. heat equation : Find the function $u : [a,b] \times [0,T) \to \mathbb{R}$ s.t.

$$\begin{cases} \dfrac{\partial u}{\partial t}(x,t) = c\dfrac{\partial^2 u}{\partial x^2}(x,t), & \text{where } (x,t) \in (a,b) \times (0,T) \\ u(x,0) = f(x) & (f : \text{given}) \\ u(a,t) = u(b,t) = 0 \end{cases}$$

where c : constant, $T > 0$: given

- Line 1 : Heat equation. Physically, $u = u(x,t)$: temperature distribution.
- Line 2 : Initial condition. Physically $u(x,0)$: temperature distribution at at time 0 and given by $f(x)$.
- Line 3 : Boundary conditions.

Finite difference method for Heat eq.
1. Explicit method (陽解法)

- Discretize

$$\frac{\partial u}{\partial t}(x,t) = c\frac{\partial^2 u}{\partial x^2}(x,t), c: \text{constant}$$

$$\frac{\partial u}{\partial t}(x,t) \approx \frac{u(x,t+\Delta t) - u(x,t)}{\Delta t},$$

$$\frac{\partial^2 u}{\partial x^2}(x,t) \approx \frac{u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)}{(\Delta x)^2}.$$

[(God)We want, the value of next time step $u(x,t+\Delta t)$...]

That is,

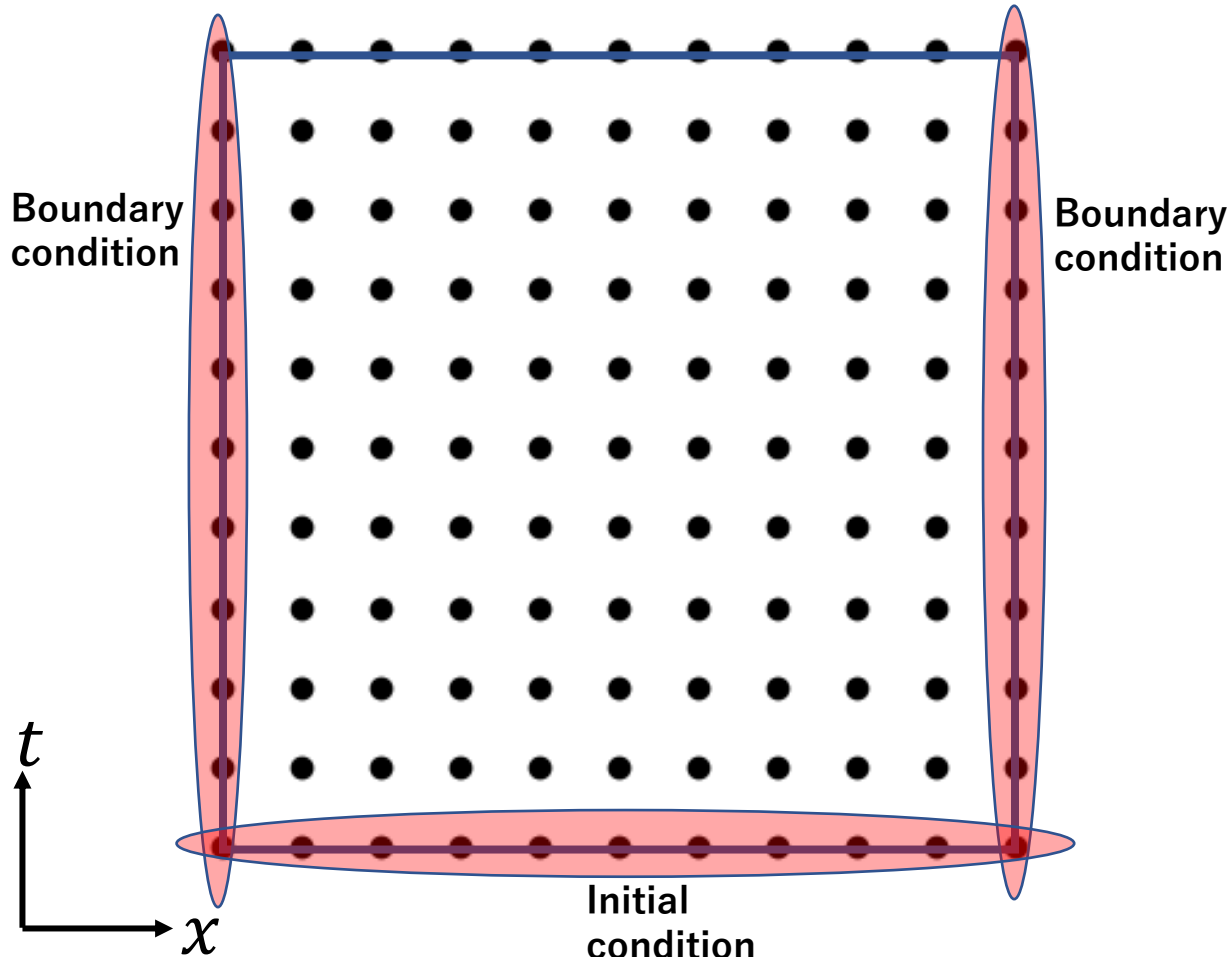
$$u(x,t+\Delta t) = \left(1 - \frac{2c\Delta t}{(\Delta x)^2}\right)u(x,t) + \frac{c\Delta t}{(\Delta x)^2}[u(x+\Delta x,t) + u(x-\Delta x,t)].$$

- To do this, discretize space and time.

space: $a, a+\Delta x, a+2\Delta x, \cdots, a+M\Delta x = b, M \coloneqq \frac{b-a}{\Delta x}$

time: $0, \Delta t, 2\Delta t, \cdots, N\Delta t = T, N \coloneqq \frac{T}{\Delta t}$

Each grid points $= (i\Delta x, j\Delta t)$



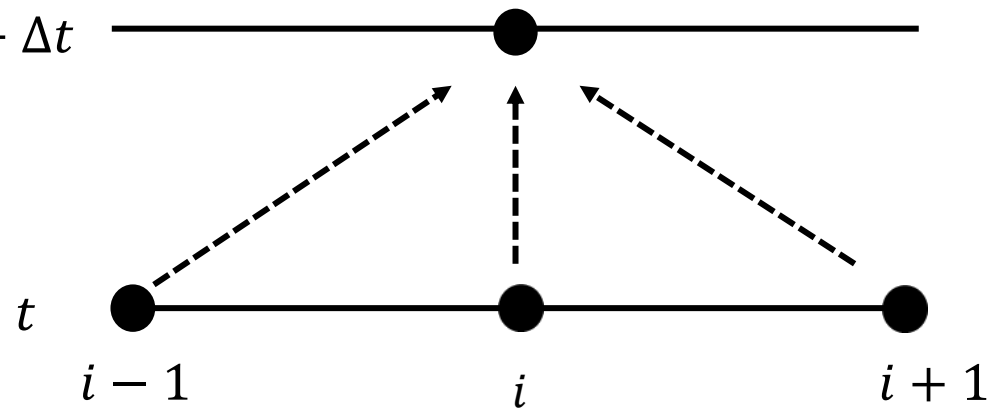
• Put $u_{i,j+1} \coloneqq u(a + i\Delta x, (j+1)\Delta t),\ u_{i,j} \coloneqq u(a + i\Delta x, j\Delta t)$

$$u_{i,j+1} = \left(1 - \frac{2c\Delta t}{(\Delta x)^2}\right) u_{i,j} + \left(\frac{c\Delta t}{(\Delta x)^2}\right)(u_{i+1,j} + u_{i-1,j})$$

where $i = 1, \ldots, M-1$.

Determine the value of next time step by three values of current time step

Repeating this procedure, the values on all grid points can be determined!



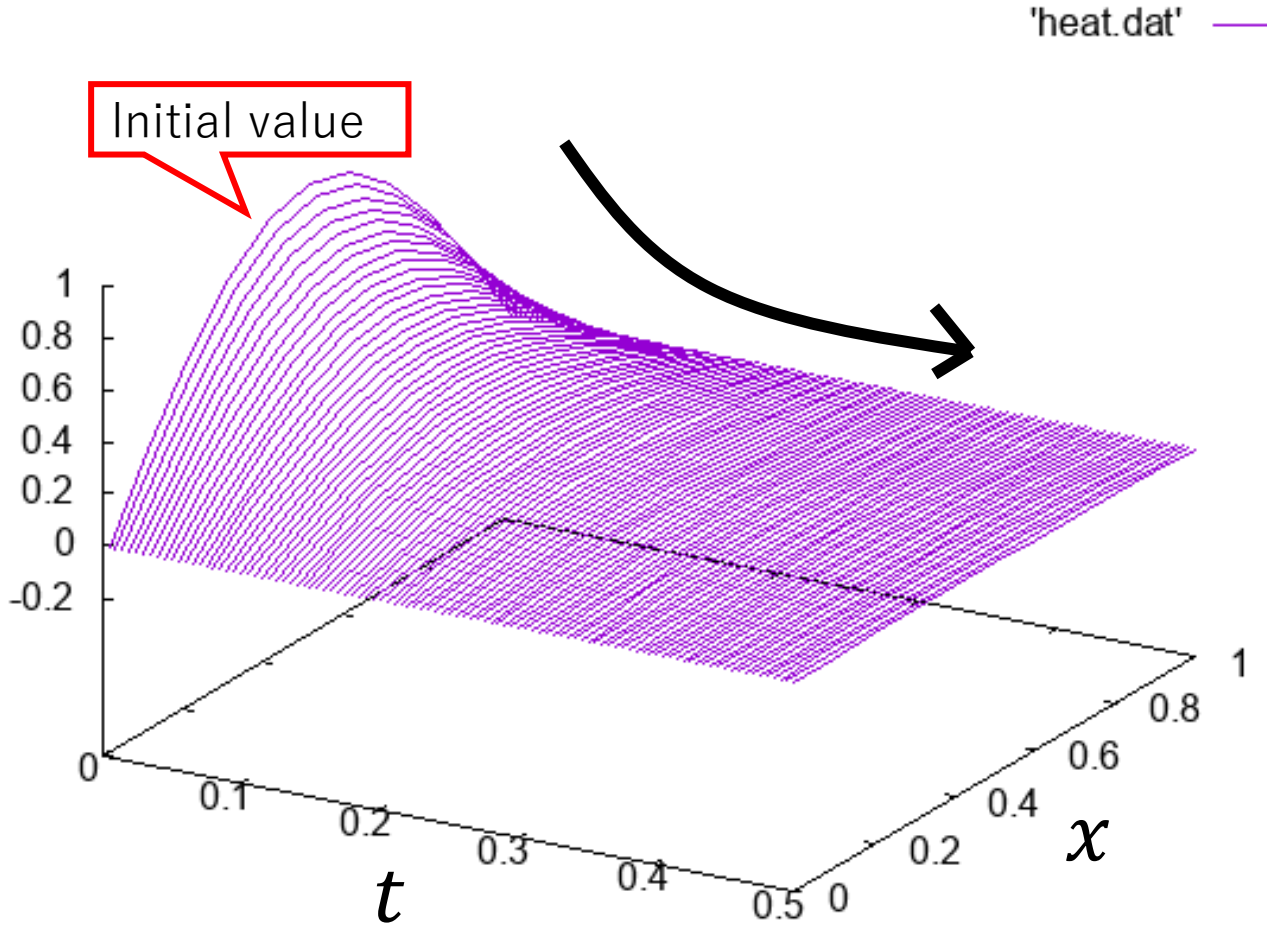
When $i = 1$ or $i = N$, these grid points correspond the boundary values which is given.

• Example of numerical result for heat eq.

Eq. $u_t = u_{xx}$ in $(0,1) \times (0, 0.5)$

I.C. $u(x, 0) = \sin \pi x$ in $(0,1)$

B.C. $u(0, t) = u(1, t) = 0$



Next :

• Review of array(treatment vector in Fortran)

• Feature of explicit method, example of codes, results

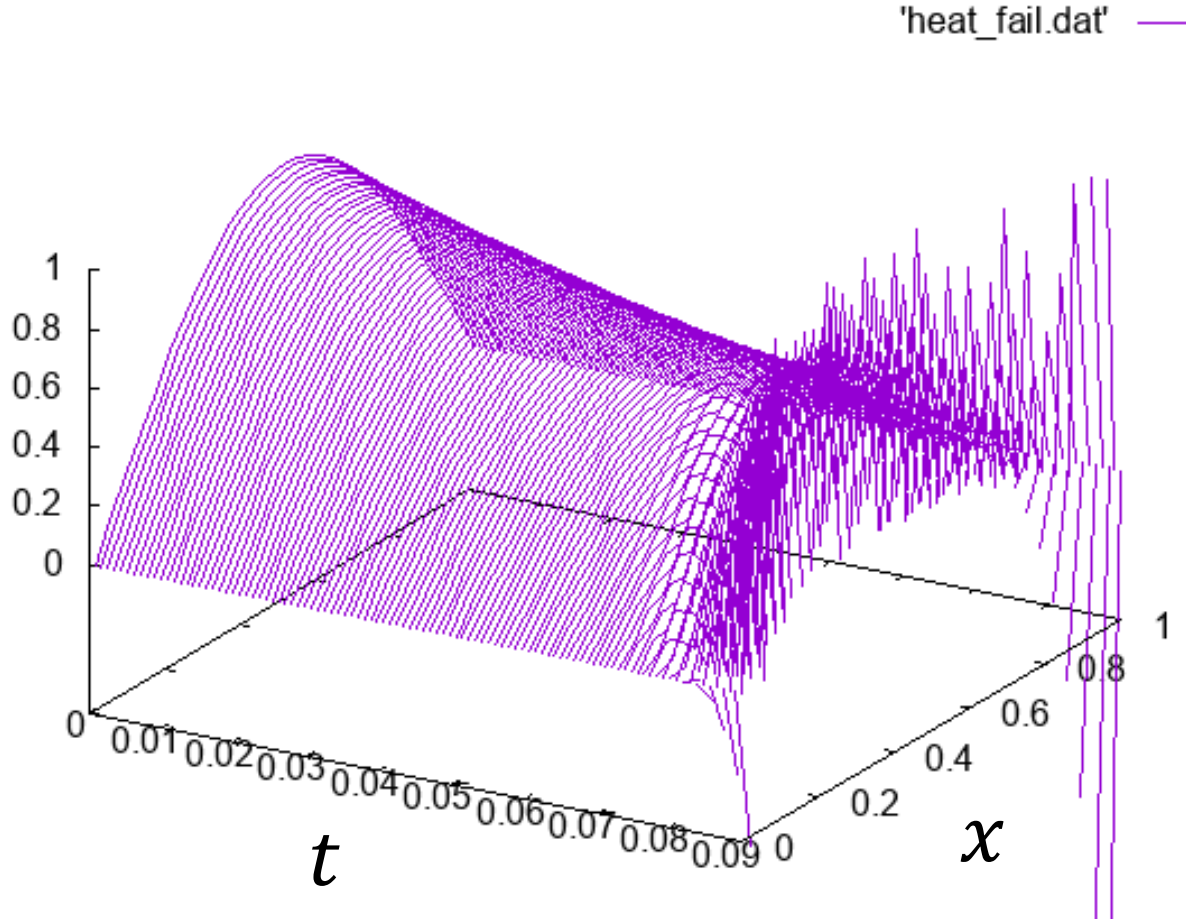
Appendix : Numerical instability (数値不安定性)for explicit scheme

- Numerical instability may occur when $1 - \frac{2\Delta t}{(\Delta x)^2} < 0$ $(c = 1)$.

We should take $\Delta t, \Delta x$ (dt, dx in code) such that $1 - \frac{2\Delta t}{(\Delta x)^2} \geq 0$.

(Ex.) $\Delta x = 0.04$ $(M = 26), \Delta t = 0.0009,$

$\rightarrow 1 - \frac{2\Delta t}{(\Delta x)^2} = -0.125 < 0.$



An example of python program

```
#matplotlib nbagg
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.animation as animation

dx = 1.0 / 50.0
dt = 0.0001
tmin = 0.0
tmax = 0.2 #simulate time

#line domain
xmin = 0.0
xmax = 1.0 #np.pi

rsq = dt / dx**2 #constant

nx = int((xmax-xmin)/dx) + 1
nt = int((tmax-tmin)/dt) + 2
print(nx,nt)
```

```
X = np.linspace(xmin, xmax, nx)
#solution
u = np.zeros((nt,nx))

#initial data
u_0 = np.sin(np.pi*X) #2*X*(1-X) #np.sin(X) #defalt
np.exp(-((X-1)**2)*10)
u[0] = u_0

#simulation
for t in range(0,nt-1):
    for x in range(1,nx-1):
        u[t+1,x] = (1 - 2 * rsq)*u[t,x] + rsq*(u[t,x-
1]+u[t,x+1])
    #Neumann condition, if we don't setting
automatically zero dierichlet
    #u[t+1,0] = u[t+1,1]
    #u[t+1,nx-1] = u[t+1,nx-2]
```

```
fig = plt.figure()
fig.set_dpi(100)
ax1 = fig.add_subplot(1,1,1)
def animate(i):
    ax1.clear()
    plt.ylim([-1.2,1.2])
    p, = plt.plot(u[i,:])

anim =
animation.FuncAnimation(fig,animate,frames=nt-
1,interval=50,repeat=False)
#anim.save("wave1dim21.gif", writer="imagemagick")
plt.show()
```