Rovnice Tepla

středa 10. března 2021 22:38

1. ODVOZENÍ ROVNICE TEPLA

JEDNOTKY

		ZÁKL.
čas	$t\,[s]$	
teplota	$T\,[K]$	
hmotnost	$m\,[kg]$	
délka	$L\,[m]$	

ODVOZENÉ

teplo	$Q[J] = [kg \cdot m^2 \cdot s^{-2}]$
měrná tep. kapac.	$c[J\,kg^{-1}\,T^{-1}] = [m^2\,s^{-2}\,K^{-1}]$
součinitel tep. vodivosti	$\lambda[W\,m^{-1}\,K^{-1}]$
hustota	$\rho[kg\,m^{-3}]$

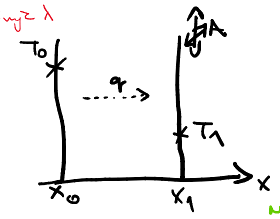
DEFINICE TEPLA

definuje c

$$Q = m \cdot T \cdot c$$

FOURIERŮV ZÁKON

definuje λ



teplotní spád $\frac{T_1 - T_0}{x_1 - x_0} = \frac{\Delta T}{\Delta x}$

$$Q = -\lambda \cdot A \cdot \frac{\Delta T}{\Delta x} =$$

JAK ROZDĚLIT ZNAČENÍ?

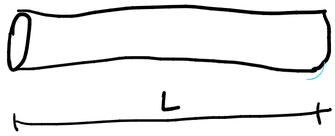
NEJASNÉ tepelný tok $q = \frac{Q}{t \cdot A}$ -----> $Q = q \cdot t \cdot A$

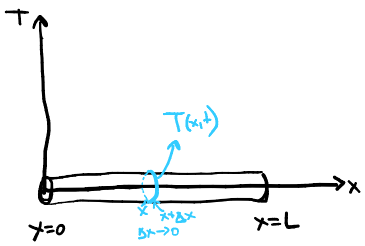
hustota tep. toku $q = \frac{Q}{A}$

1D $q = -\lambda \frac{\Delta T}{\Delta x} = -\lambda \frac{\partial T}{\partial x}$

3D $q = -\lambda \nabla T = -\lambda\left(\frac{\partial T}{\partial x} + \frac{\partial T}{\partial y} + \frac{\partial T}{\partial z}\right)$

ZÁKON ZACHOVÁNÍ ENERGIE





T... neznámá

ρ, c, λ, A ...... konstanty

teplo proudí pouze ve směru x
(stěny válce jsou izolovány)

teplo proudí pouze ve směru x
(strany válce jsou izolovány)

TEPLO SEGMENTU $(x - x+\Delta x)$:

$$Q = c \cdot m \cdot T(x,t)$$
$$= c \cdot \overbrace{\rho \cdot A \cdot \Delta x}^{} \cdot T(x,t)$$

$Q = q \cdot A \cdot t$

CELKOVÁ ZMĚNA Q V ČASE = PŘÍCHOZÍ Q (ZLEVA) − ODCHOZÍ Q (ZPRAVA)

$$Q(x,t+\Delta t) - Q(x,t) = q_{zleva,x} \cdot A \cdot \Delta t - q_{zprava,x+\Delta x} \cdot A \cdot \Delta t$$

$$c \cdot \rho \cdot \cancel{A} \cdot \Delta x \cdot T(x,t+\Delta t) - c\rho \cancel{A}\, \Delta x \cdot T(x,t) = \cancel{A}\Delta t\left(-\lambda \frac{\partial T}{\partial x}\right)_x - \cancel{A}\Delta t\left(-\lambda \frac{\partial T}{\partial x}\right)_{x+\Delta x}$$

$/ : c$
$: \rho$
$: \Delta t$
$: \Delta x$

$$\frac{T(x,t+\Delta t) - T(x,t)}{\Delta t} = \frac{-\lambda\left(\frac{\partial T}{\partial x}\right)_x + \lambda\left(\frac{\partial T}{\partial x}\right)_{x+\Delta x}}{c\rho\,\Delta x}$$

ROZDÍL TEPLOT ZA ČAS

ROZDÍL ROZDÍLU TEPLOT PODLE X

$$\frac{T(x,t+\Delta t) - T(x,t)}{\Delta t} = \frac{\lambda}{c \cdot \rho} \frac{\left(\frac{\partial T}{\partial x}\right)_{x+\Delta x} - \left(\frac{\partial T}{\partial x}\right)_{x+\Delta x}}{\Delta x}$$

$$\Delta t, \Delta x \longrightarrow 0$$

$$\boxed{\frac{\partial T}{\partial t} = \frac{\lambda}{c \cdot \rho} \frac{\partial^2 T}{\partial x^2}}$$

ROVNICE TEPLA:
teplota se v čase mění úměrně druhé derivaci funkce teploty po délce
- změna teploty je přímo úměrná křivosti funkce teploty

2. ŘEŠENÍ METODOU KONEČNÝCH DIFERENCÍ

JEDNODUCÝ PŘÍPAD - OKRAJOVÉ PODMÍNKY = DANÁ TEPLOTA

$$\frac{\partial T}{\partial t} = \frac{\lambda}{\rho c} \frac{\partial^2 T}{\partial x^2}$$

EXPLICITNÍ METODA

T_i^j — j → čas, i → prostor

CENTERED FINITE-DIVIDED DIFFERENCE

$$\frac{\partial^2 T}{\partial x^2} = \frac{T_{i+1}^j - 2T_i^j + T_{i-1}^j}{\Delta x^2} \quad (\text{chyba } O[(\Delta x)^2])$$

FORWARD FINITE-DIVIDED DIFFERENCE

$$\frac{\partial T}{\partial t} = \frac{T_i^{j+1} - T_i^j}{\Delta t} \quad (\text{chyba } O(\Delta t))$$

dosadím

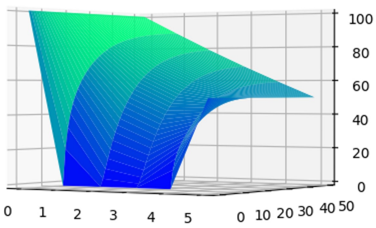
$$\frac{T_i^{j+1} - T_i^j}{\Delta t} = \frac{\lambda}{\rho c} \frac{T_{i+1}^j - 2T_i^j + T_{i-1}^j}{\Delta x^2}$$

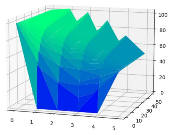
$$T_i^{j+1} = T_i^j + \underbrace{\frac{\lambda}{\rho c}\frac{\Delta t}{\Delta x^2}}_{\text{(někde značeno } \lambda)}\left(T_{i+1}^j - 2T_i^j + T_{i-1}^j\right)$$

Kód 1

```python
# pro vyzkouseni si finite difference method, pujdu podle ucebnice a napisu jednoduchej program pomoci explictini metody
# 1
# jednoduche okrajove podminky - jen zadane teploty
import numpy as np
import matplotlib.pyplot as plt
# ty hodnoty jsou v americkych jednotkach odpovidajici hliniku
lamb = 0.49 # W/(m.K) soucinitel tepelne vodivosti
c = 0.2174 # J/(m.K) merna tepelna kapacita
ro = 2.7 # kg/m3 hustota
L = 10 # m celkova delka
dx = 2 #m delka kroku v x
dt = 1 #s delka kroku v case
#pocatecni podminky
T0 = 0 # C
# okrajove podminky
T_levo = 100 # C
T_pravo = 50 # C
n_no = int(L/dx+1)
T = np.zeros([50, n_no])
#vypocitam integracni keficient lamb/(ro*c) * (dt/dx**2)
# pokud je tento koeficient > 0.5 tak je tato metoda nestabilni
int_ko = lamb/(ro * c) * dt/(dx**2)
print(int_ko) # melo by vyjit 0.0208695
#rucne zadam prvni teploty a pak pojedu dal
T[0] = T0
T[0, 0] = T_levo
T[0, -1] = T_pravo
# ted zadam zbytek
# pocitam vzdy jen prostredni hodnoty, krajni zadavam z okrajovych podminek
for i in range(1, len(T)):
   for j in range(n_no):
      if j == 0:
         T[i, j] = T_levo
      elif j == n_no-1:
         T[i, j] = T_pravo
      else:
         T[i,j] = T[i-1,j] + int_ko * (T[i-1,j+1] - 2*T[i-1,j] + T[i-1,j-1])
# tady jen nastavim aby se mi tiskly jen 2 desetinne pozice
np.set_printoptions(formatter={'float': lambda x: "{0:0.2f}".format(x)})
print(T)
# 3d graf
x = np.arange(n_no)
y = np.arange(len(T))
X, Y = np.meshgrid(x, y)
Z = T
fig = plt.figure()
ax = plt.axes(projection='3d')
ax.plot_surface(X, Y, Z, rstride=1, cstride=1,
            cmap='winter', edgecolor='none')
# ax.set_xlabel('x')
# ax.set_ylabel('y')
# ax.set_zlabel('z');
plt.show()
```

```
0.2086953558894681
[[100.00 0.00 0.00 0.00 0.00 50.00]
 [100.00 20.87 0.00 0.00 10.43 50.00]
 [100.00 33.03 4.36 2.18 16.51 50.00]
 [100.00 41.02 9.88 5.62 20.51 50.00]
 [100.00 46.83 15.49 9.62 23.56 50.00]
 [100.00 51.39 20.81 13.75 26.17 50.00]
 [100.00 55.15 25.72 17.82 28.55 50.00]
 [100.00 58.37 30.21 21.71 30.79 50.00]
 [100.00 61.18 34.31 25.38 32.90 50.00]
 [100.00 63.67 38.05 28.81 34.90 50.00]
 [100.00 65.91 41.47 32.01 36.78 50.00]
 [100.00 67.92 44.60 34.98 38.54 50.00]
 [100.00 69.75 47.46 37.73 40.19 50.00]
 [100.00 71.41 50.08 40.27 41.72 50.00]
 [100.00 72.93 52.49 42.62 43.15 50.00]
 [100.00 74.31 54.69 44.79 44.47 50.00]
 [100.00 75.58 56.72 46.79 45.69 50.00]
 [100.00 76.74 58.58 48.63 46.82 50.00]
 [100.00 77.80 60.30 50.33 47.86 50.00]
 [100.00 78.78 61.87 51.90 48.82 50.00]
 [100.00 79.68 63.32 53.34 49.71 50.00]
 [100.00 80.51 64.65 54.66 50.53 50.00]
 [100.00 81.27 65.88 55.88 51.28 50.00]
 [100.00 81.96 67.00 57.01 51.97 50.00]
 [100.00 82.61 68.04 58.04 52.61 50.00]
 [100.00 83.20 68.99 59.00 53.20 50.00]
 [100.00 83.74 69.87 59.87 53.74 50.00]
 [100.00 84.24 70.68 60.68 54.24 50.00]
 [100.00 84.70 71.42 61.42 54.70 50.00]
 [100.00 85.12 72.11 62.11 55.12 50.00]
 [100.00 85.51 72.73 62.74 55.51 50.00]
 [100.00 85.87 73.31 63.31 55.87 50.00]
 [100.00 86.20 73.85 63.85 56.20 50.00]
 [100.00 86.50 74.34 64.34 56.50 50.00]
 [100.00 86.78 74.79 64.79 56.78 50.00]
 [100.00 87.04 75.20 65.20 57.04 50.00]
 [100.00 87.27 75.59 65.59 57.27 50.00]
 [100.00 87.49 75.94 65.94 57.49 50.00]
 [100.00 87.69 76.26 66.26 57.69 50.00]
 [100.00 87.87 76.56 66.56 57.87 50.00]
 [100.00 88.04 76.83 66.83 58.04 50.00]
 [100.00 88.20 77.09 67.09 58.20 50.00]
 [100.00 88.34 77.32 67.32 58.34 50.00]
 [100.00 88.48 77.53 67.53 58.48 50.00]
 [100.00 88.60 77.73 67.73 58.60 50.00]
 [100.00 88.71 77.91 67.91 58.71 50.00]
 [100.00 88.81 78.08 68.08 58.81 50.00]
 [100.00 88.91 78.23 68.23 58.91 50.00]
 [100.00 88.99 78.37 68.37 58.99 50.00]
 [100.00 89.07 78.50 68.50 59.07 50.00]]
```





OKRAJOVÉ PODMÍNKY = ZADANÝ TEPELNÝ TOK

$$T_i^{j+1} = T_i^j + \frac{\lambda}{\rho c}\frac{\Delta t}{\Delta x^2}\left(T_{i+1}^j - 2T_i^j + T_{i-1}^j\right)$$

ZAVEDEME VIRTUÁLNÍ BOD MIMO STĚNU, ABYCHOM VYJÁDŘILI TEPELNÝ TOK NA OKRAJI

$$T_0^{j+1} = T_0^j + \frac{\lambda}{\rho c}\frac{\Delta t}{\Delta x^2}\left(T_1^j - 2T_0^j + T_{-1}^j\right)$$

$$q = -\lambda\frac{\partial T}{\partial x} = -\lambda\frac{T_{i+1} - T_i}{\Delta x}$$ ZDE JSEM POUŽIL FORWARD F.DIF. APROXIMACI

$$q_0^j = -\lambda\frac{T_0^j - T_{-1}^j}{\Delta x}$$

TADY BYCH TO UDĚLAL LÉPE PŘESTUP TEPLA – $q_0 = h\cdot A(T_0 - T_1)$

→ TAKHLE ZJIŠŤOVAT, [illegible] JDE VEN

$$T_{-1}^j = T_0^j - \frac{q_0^j \cdot \Delta x}{\lambda}$$

$$\boxed{T_0^{j+1} = T_0^j + \frac{\lambda}{\rho c}\frac{\Delta t}{\Delta x^2}\left(T_1^j - 2T_0^j + \left(T_0^j - \frac{q_0^j\cdot\Delta x}{\lambda}\right)\right)}$$

VZOREC PRO KRAJNÍ BOD

Kód 2

```
# napotreti to uvedu do praxe, na jedne strane budu menit teplotu z prubehu dne, na druhe strane budu
drzet 20C
# k tomu budu merit kolik musim dodatvat tepla aby to tak bylo
import numpy as np
import matplotlib.pyplot as plt
# ----------------------------------------
# zadani
# ----------------------------------------
# zadam hodnoty pro stenu cihelnou stenu CP
lamb =  0.8 # W/(m.K) soucinitel tepelne vodivosti
c = 900 # J/(m.K) merna tepelna kapacita
ro = 1700# kg/m3 hustota
L = 0.4 # m celkova delka
dx = 0.05 #m delka kroku v x
dt = 600 #s delka kroku v case
# ----------------------------------------
# pocatecni a okrajove podminky
# ----------------------------------------
# okrajove podminky
# venkovni teplota
# funkce nacte teplotu pro zadany den ze souboru z roku 2000 z jedne nemecke stanice
# zadat den ve format YYYYMMDD napr 0311 pro 3.11.
# vrati vektor T_out = prubeh teplot daneho dne po 10 minutach
def get_temp(day):
   data = np.loadtxt("2000_po_10_min.txt", comments="#", delimiter=";", unpack=False)
   T_out = np.array([])
for i in range(len(data)):
      if str(data[i, 1])[:8] == "2000" + day:
         T_out = np.append(T_out, [data[i, -1]])
return T_out
T_out = get_temp('0311')
# vnitrni teplota
T_in = 20 #stupnu C
n_no = int(L/dx + 1) # number of nodes
# pocatecni teplota
# linearne rozdelim teplotu mezi Tin a Tout
T0 = np.linspace(T_in, T_out[0],n_no)
# pocatecni podminky aplikovane na prazdnou matici
n_no = int(L/dx + 1) # number of nodes
T = np.zeros([len(T_out), n_no]) # prubeh teploty v prostoru a case
T[0] = T0
T[:, 0] = T_in
T[:, -1] = T_out
print(T0)
# ----------------------------------------
# vypocet
# ----------------------------------------
int_ko = lamb/(ro*c) * dt/(dx**2)
print(int_ko)
# pocitam vzdy jen prostredni hodnoty, krajni zadavam z okrajovych podminek
for i in range(1, len(T)):
   for j in range(n_no):
      if j == 0:
         continue
      elif j == n_no-1:
         continue
      else:
         T[i,j] = T[i-1,j] + int_ko * (T[i-1,j+1] - 2*T[i-1,j] + T[i-1,j-1])
np.set_printoptions(formatter={'float': lambda x: "{0:0.1f}".format(x)})
print(T)
print(T[0])
print(T[1])
# vypocet tepelneho toku
# merim tohle spravne?
q = -lamb * (T[:,0] - T[:,1])/dx
print('tepelny tok na vnitrni strane')
print(q)
# ----------------------------------------
# 3d graf
# ----------------------------------------
x = np.arange(n_no)
y = np.arange(len(T))
X, Y = np.meshgrid(x, y)
Z = T
fig = plt.figure()
ax = plt.axes(projection='3d')
ax.plot_surface(X, Y, Z, rstride=1, cstride=1, cmap='magma', edgecolor='none')
ax.set_xlabel('šířka stěny')
ax.set_ylabel('č(po 10 s')
ax.set_zlabel('teplota');
plt.show()
```

```
[20.     17.1875 14.375  11.5625  8.75    5.9375  3.125   0.3125 -2.5   ]
0.12549019607843134
[[20.0 17.2 14.4 ... 3.1 0.3 -2.5]
 [20.0 17.2 14.4 ... 3.1 0.3 -2.5]
 [20.0 17.2 14.4 ... 3.1 0.3 -2.4]
 ...
 [20.0 17.9 15.8 ... 6.2 3.0 -0.5]
 [20.0 17.9 15.8 ... 6.2 2.9 -0.5]
 [20.0 17.9 15.8 ... 6.1 2.9 -0.7]]
[20.0 17.2 14.4 11.6 8.8 5.9 3.1 0.3 -2.5]
[20.0 17.2 14.4 11.6 8.8 5.9 3.1 0.3 -2.5]
tepelny tok na vnitrni strane
[-45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0
 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0 -45.0
 -45.0 -45.0 -44.9 -44.9 -44.9 -44.9 -44.9 -44.8 -44.8 -44.8 -44.7 -44.7
 -44.6 -44.6 -44.5 -44.5 -44.4 -44.4 -44.3 -44.2 -44.2 -44.1 -44.0 -44.0
 -43.9 -43.8 -43.8 -43.7 -43.6 -43.5 -43.4 -43.3 -43.2 -43.2 -43.1 -43.0
 -42.9 -42.8 -42.6 -42.5 -42.4 -42.3 -42.2 -42.0 -41.9 -41.7 -41.6 -41.4
 -41.3 -41.1 -41.0 -40.8 -40.6 -40.4 -40.3 -40.1 -39.9 -39.7 -39.5 -39.4
 -39.2 -39.0 -38.8 -38.6 -38.5 -38.3 -38.1 -37.9 -37.8 -37.6 -37.4 -37.3
 -37.1 -36.9 -36.8 -36.6 -36.5 -36.4 -36.2 -36.1 -35.9 -35.8 -35.7 -35.5
 -35.4 -35.3 -35.2 -35.1 -35.0 -34.9 -34.8 -34.7 -34.6 -34.5 -34.4 -34.3
 -34.2 -34.1 -34.0 -33.9 -33.9 -33.8 -33.7 -33.6 -33.6 -33.5 -33.5 -33.4
 -33.4 -33.3 -33.3 -33.3 -33.3 -33.2 -33.2 -33.2 -33.2 -33.2 -33.3 -33.3]
```

VELKÁ SETRVAČNOST,
MOHL BYCH POROVNAT STĚNU
SE STEJNÝM U, ALE JINÝM C