Przepływ ciepła w sztabce aluminium

Krzysztof Konieczny, Wojciech Kura

grudzień 2020,
Politechnika Krakowska im. Tadeusza Kościuszki w Krakowie

Spis treści

1 Wstęp

W przyrodzie naturalne jest to, że ciepło przepływa od gorącego do zimnego, czyli z regionów o wysokiej temperaturze do obszarów o niskiej temperaturze. Tą zasadę oddaje wyrażenie matematyczne stwierdzając, że szybkość przepływu ciepła **H** przez materiał jest proporcjonalna do gradientu temperatury **T** w całym materiale:

$$\mathbf{H} = -K\nabla T(\mathbf{x}, t)$$

gdzie **K** jest przewodnictwem cieplnym danego materiału. Całkowita ilość ciepła **Q(t)** w materiale i dowolnym momencie jest proporcjonalna do całki temperatury po objętości materiału:

$$Q(t) = \int d\mathbf{x}\, C\rho(\mathbf{x})\, T(\mathbf{x}, t)$$

gdzie **C** to ciepło właściwe materiału, natomiast ρ to jego gęstość. Jako, że energia jest zachowana, tempo spadku Q w czasie musi być równe ilości ciepła wypływającego z materiału. Po uzyskaniu takiego bilansu energetycznego i zastosowaniu twierdzenia o rozbieżności, otrzymujemy równanie ciepła:

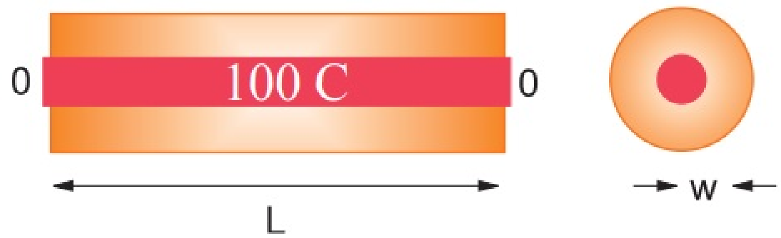
$$\frac{\partial T(\mathbf{x}, t)}{\partial t} = \frac{K}{C\rho}\nabla^2 T(\mathbf{x}, t)$$

Równanie ciepła jest **parabolicznym równaniem różniczkowym cząstkowym z przestrzenią i czasem, które są zmiennymi niezależnymi.** W naszym problemie że nie ma wahań temperatury w kierunkach prostopadłych do pręta (y i z), więc mamy tylko jedną współrzędną przestrzenną:

$$\frac{\partial T(x, t)}{\partial t} = \frac{K}{C\rho}\frac{\partial^2 T(x, t)}{\partial x^2}$$

2 Cel pracy

Mamy aluminiowy pręt o długości L = 1m i szerokości wyrównanej do osi X. Jest on izolowany na całej długości, poza końcami. Temperatura początkowa pręta wynosi **100** C^o. Następnie oba końce stykamy z lodowatą wodą o temperaturze **0** C^o**.** Ciepło wypływa tylko z nieizolowanych końców. Naszym zadaniem było napisanie programu w Pythonie, który pozwoli na określenie, jak będzie się zmieniać temperatura na całej długości pręta.



3 Rozwiązanie równania różniczkowego dla ciepła

Przejdźmy teraz do rozwiązania analitycznego naszego równania. Zaczynamy od założenia, że rozwiązanie rozdziela się na iloczyn funkcji czasu i przestrzeni:

$$T(x,t) = X(x)\mathcal{T}(t)$$

gdy podstawimy to do równania ciepła i podzielimy przez X (x) T (t), otrzymamy:

$$\frac{d^2X(x)}{dx^2} + k^2X(x) = 0, \quad \frac{d\mathcal{T}(t)}{dt} + k^2\frac{C}{C\rho}\mathcal{T}(t) = 0$$

gdzie **k** jest stałą. Warunek brzegowy, że temperatura przy x = 0 jest równa funkcji sinus dla X:

$$X(x) = A\sin kx$$

warunek brzegowy, że temperatura jest równa zeru przy x = L, to wówczas mamy:

$$\sin kL = 0 \;\; \Rightarrow \;\; k = k_n = n\pi/L, \quad n = 1, 2, \ldots .$$

funkcja czasu jest zanikająca w k i mamy, że:

$$\mathcal{T}(t) = e^{-k_n^2 t/C\rho}, \;\; \Rightarrow \;\; T(x,t) = A_n \sin k_n x e^{-k_n^2 t/C\rho}$$

gdzie **n** może być dowolną liczbą całkowitą, a **An** jest dowolną stałą. Jako, że nasze równanie ciepła jest liniowe, najbardziej ogólnym rozwiązaniem będzie liniowa superpozycja wszystkich wartości n:

$$T(x,t) = \sum_{n=1}^{\infty} A_n \sin k_n x \, e^{-k_n^2 t/C\rho}$$

współczynniki **An** są określone przez warunek początkowy, że w czasie **t** = **0** cały pręt ma temperaturę **T** = **100** C^o**:**

$$T(x, t=0) = 100 \;\; \Rightarrow \;\; \sum_{n=1}^{\infty} A_n \sin k_n x = 100$$

dla funkcji sinusoidalnych **An** = **4T0/n**π **dla n nieparzyste**, co ostatecznie daje nasze rozwiązanie, czyli zależność temperatury pręta od położenia i czasu:

$$T(x,t) = \sum_{n=1,3,\ldots}^{\infty} \frac{4T_0}{n\pi} \sin k_n x e^{-k_n^2 K t/(C\rho)}$$

4 Kod programu - Python

```python
# aluminium

from numpy import *
import matplotlib.pylab as p
from mpl_toolkits.mplot3d import Axes3D

Nx = 101;  # dlugosc preta
Nt = 40500;  # ilosc obliczen
Dx = 0.03;
Dt = 0.9;  # time step
KAPPA = 237.;  # przewodnictwo ciepla
SPH = 900.;  # cieplo wlasciwe
RHO = 2700.;  # gestosc
T = zeros((Nx, 2), float);
Tpl = zeros((Nx, 181), float);
print("pracuje, poczekaj...")

for ix in range(1, Nx - 1):  T[ix, 0] = 100.0;
T[0, 0] = 0.0;
T[0, 1] = 0.  # pierwszy i ostatni punkt w 0
T[Nx - 1, 0] = 0.;
T[Nx - 1, 1] = 0.0
cons = KAPPA / (SPH * RHO) * Dt / (Dx * Dx);  # stala
m = 1  # counter for rows, one every 150 time steps

for t in range(1, Nt):  # iteracja czasu
    for ix in range(1, Nx - 1):  # skonczone roznice
        T[ix, 1] = T[ix, 0] + \
            cons * (T[ix + 1, 0] + \
                    T[ix - 1, 0] - 2. * T[ix, 0])
    if t % 225 == 0 or t == 1:
        for ix in range(1, Nx - 1, 2):
            Tpl[ix, m] = T[ix, 1]
        print(m)
        m = m + 1  # zwiekszanie m co 150 timestepow
```

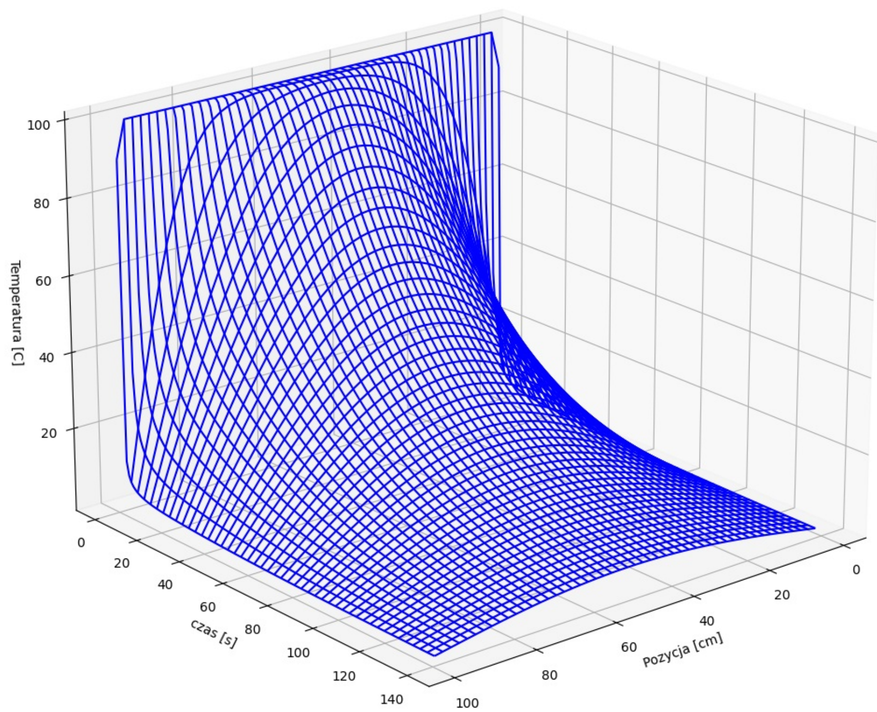
```
    for ix in range(1, Nx - 1): T[ix, 0] = T[ix, 1]
x = list(range(1, Nx - 1, 2))
y = list(range(1, 140))  # co 10 punktow w y (zasieg)
X, Y = p.meshgrid(x, y)  # siatka pozycji i czasu


def functz(Tpl):  # funkcja zwraca temperature
    z = Tpl[X, Y]
    return z


Z = functz(Tpl)
fig = p.figure()  # tworzenie figury
ax = Axes3D(fig)  # rysowanie osi
ax.plot_wireframe(X, Y, Z, color='b')  # kolor
ax.set_xlabel('Pozycja [cm]')  # opisy osi
ax.set_ylabel('czas [s]')
ax.set_zlabel('Temperatura [C]')
p.show()  # shows figure, close Python shell
print("skonczone :)")
```

5 Wyniki i podsumowanie

Wynik pracy programu przedstawia poniższy wykres:



Pokazuje on nam jak temperatura na całej długości pręta ulega zmianie w czasie, spełniając tym samym i rozwiązując nasze zadanie.

6 Bibliografia

- *Rubin H. Landau, José Páez, and Cristian C. Bordeianu, —* „A Survey of Computational Physics: Introductory Computational Science" Princeton University Press 2011