

# ULEPSZONY PRZEPŁYW CIEPŁA: METODA CRANKA-NICOLSONA

Emil Śmiech, Amelia Królczyk

Grudzień 2020

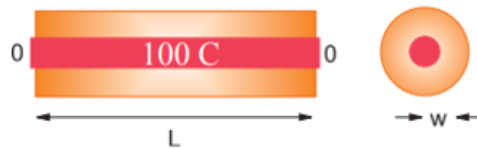
Część I

**Wstęp**

Celem naszego ćwiczenia było rozwiązanie poniższego problemu przy użyciu metody Cranka - Nicolsona:

Mamy pręt aluminiowy o długości  $L = 1$  [m] i szerokości „w”. Jest on izolowany na całej długości, ale nie na końcach. Początkowo pręt ma jednakową temperaturę  $100^{\circ}\text{C}$ , a następnie oba końce styka się z lodowatą wodą o temperaturze  $0^{\circ}\text{C}$ . Ciepło wypływa tylko z nieizolowanych końców. Naszym zadaniem jest określić jak z biegiem czasu będzie się zmieniać temperatura na całej długości tego pręta.

Rysunek 1: Metalowy pręt izolowany na całej długości, którego końce stykają się z lodem. Pręt jest pomalowany na czerwono, a izolacja w jaśniejszym kolorze



Temperatura początkowa pręta i warunki brzegowe:

$$T(x, t = 0) = 100\text{ C}, \quad T(x = 0, t) = T(x = L, t) = 0\text{ C}.$$

Podstawowym faktem natury jest to, że ciepło przepływa od gorącego do zimnego, czyli z regionów o wysokiej temperaturze do obszarów o niskiej temperaturze.

Szybkość przepływu ciepła przez materiał jest proporcjonalna do gradientu temperatury w całym materiale. Całkowita ilość ciepła w materiale w dowolnym momencie jest proporcjonalna do całki temperatury po objętości materiału. Ponieważ energia jest zachowana, tempo spadku ciepła w czasie musi być równe ilości ciepła wypływającego z materiału. Po uzyskaniu tego bilansu energii i zastosowaniu twierdzenia o rozbieżności, otrzymujemy równanie ciepła:

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

gdzie:

$K$  jest przewodnością cieplną materiału,

$Q(t)$  to całkowita ilość ciepła,

$C$  to ciepło właściwe materiału,

$\rho$  to jego gęstość

## Część II

# Zastosowanie metody Cranka - Nicolsona Część teoretyczna

Metoda Cranka-Nicolsona zapewnia wyższy stopień dokładności równania ciepła. Ta metoda oblicza pochodną czasową z przybliżeniem centralnej różnicy, w przeciwieństwie do wcześniej używanego przybliżenia różnicy w przód. Aby uniknąć wprowadzenia błędu dla początkowego kroku czasowego, gdy znana jest tylko pojedyncza wartość czasu, metoda wykorzystuje przedział czas, tak aby czas był przesuwany od czasu  $t$  do  $t + \frac{\Delta t}{2}$ .

$$\frac{\partial T}{\partial t} \left( x, t + \frac{\Delta t}{2} \right) \simeq \frac{T(x, t + \Delta t) - T(x, t)}{\Delta t} + O(\Delta t^2).$$

Przybliżenie centralnej różnicy drugiej pochodnej przestrzeni dla czasu  $t = t + \frac{\Delta t}{2}$ , wynosi:

$$\begin{aligned} 2(\Delta x)^2 \frac{\partial^2 T}{\partial x^2} \left( x, t + \frac{\Delta t}{2} \right) &\simeq [T(x - \Delta x, t + \Delta t) - 2T(x, t + \Delta t) + T(x + \Delta x, t + \Delta t)] \\ &+ [T(x - \Delta x, t) - 2T(x, t) + T(x + \Delta x, t)] + O(\Delta x^2). \end{aligned}$$

Z tego dostajemy że równanie różnicy ciepła to:

$$\begin{aligned} T_{i,j+1} - T_{i,j} &= \frac{\eta}{2} [T_{i-1,j+1} - 2T_{i,j+1} + T_{i+1,j+1} + T_{i-1,j} - 2T_{i,j} + T_{i+1,j}], \\ x &= i\Delta x, \quad t = j\Delta t, \quad \eta = \frac{K\Delta t}{C\rho\Delta x^2}. \end{aligned}$$

Po zgrupowaniu razem wyrażeń dotyczących tej samej temperatury, otrzymujemy równanie z czasami przyszłymi po lewej stronie równania i czasami obecnymi po prawej stronie równania:

$$-T_{i-1,j+1} + \left( \frac{2}{\eta} + 2 \right) T_{i,j+1} - T_{i+1,j+1} = T_{i-1,j} + \left( \frac{2}{\eta} - 2 \right) T_{i,j} + T_{i+1,j}.$$

To równanie reprezentuje niejawny schemat dla temperatury  $T(i, j)$ , gdzie słowo „niejawne” oznacza, że musimy rozwiązać równoczesne równania, aby otrzymać pełne rozwiązanie dla całej przestrzeni. Natomiast jawny schemat wymaga iteracji, aby dojść do rozwiązania.

Warunki brzegowe na końcach słupka dla wszystkich czasów i przybliżone wartości z pierwszej pochodnej:

$$\begin{aligned} T_{i,0}, \text{ known}, & \quad T_{0,j}, \text{ known}, & \quad T_{N,j}, \text{ known}, \\ T_{0,j+1} = T_{0,j} = 0, & \quad T_{N,j+1} = 0, & \quad T_{N,j} = 0. \end{aligned}$$

Po przekształceniu równania tak, abyśmy mogli użyć tych znanych wartości  $T$  do przesuwania rozwiązania  $j = 0$  do przodu w czasie, wyrażając to równanie

jako zbiór równoczesnych równań liniowych (w postaci macierzowej) otrzymujemy:

$$\begin{pmatrix} (\frac{2}{\eta} + 2) & -1 & & & & & & & \\ -1 & (\frac{2}{\eta} + 2) & -1 & & & & & & \\ & -1 & (\frac{2}{\eta} + 2) & -1 & & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & -1 & (\frac{2}{\eta} + 2) & -1 & & \\ & & & & & -1 & (\frac{2}{\eta} + 2) & & \\ & & & & & & -1 & (\frac{2}{\eta} + 2) & \end{pmatrix} \begin{pmatrix} T_{1,j+1} \\ T_{2,j+1} \\ T_{3,j+1} \\ \vdots \\ T_{n-2,j+1} \\ T_{n-1,j+1} \end{pmatrix} \\ = \begin{pmatrix} T_{0,j+1} + T_{0,j} + (\frac{2}{\eta} - 2)T_{1,j} + T_{2,j} \\ T_{1,j} + (\frac{2}{\eta} - 2)T_{2,j} + T_{3,j} \\ T_{2,j} + (\frac{2}{\eta} - 2)T_{3,j} + T_{4,j} \\ \vdots \\ T_{n-3,j} + (\frac{2}{\eta} - 2)T_{n-2,j} + T_{n-1,j} \\ T_{n-2,j} + (\frac{2}{\eta} - 2)T_{n-1,j} + T_{n,j} + T_{n,j+1} \end{pmatrix}.$$

Wszystkie  $T$  po prawej stronie równania dla różnych pozycji są w chwili obecnej:  $j$ , a dla dwóch końców, których  $T$  są znane przez cały czas poprzez warunki brzegowe, w przyszłości:  $j + 1$ . Równania Cranka – Nicolsona mają standardową postać  $[A]x = b$  dla równań liniowych, macierz współczynników  $[A]$  jest trójdzielna (zerowe elementy, oprócz głównej przekątnej i dwóch przekątnych po obu jej stronach).

$$\begin{pmatrix} d_1 & c_1 & 0 & 0 & \dots & \dots & \dots & 0 \\ a_2 & d_2 & c_2 & 0 & \dots & \dots & \dots & 0 \\ 0 & a_3 & d_3 & c_3 & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & a_{N-1} & d_{N-1} & c_{N-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & a_N & d_N \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{N-1} \\ x_N \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_{N-1} \\ b_N \end{pmatrix}$$

Rozwiązujemy równanie macierzowe, manipulując poszczególnymi równaniami, aż macierz współczynników będzie górnotrójkątna z wszystkimi elementami głównej przekątnej równymi 1. Zaczynamy od podzielenia pierwszego równania przez  $d_1$ , a następnie odejmujemy  $a_2$  razy pierwsze równanie, a następnie podzielenie drugiego równania przez drugi przekątny element:

$$\begin{pmatrix} 1 & \frac{c_1}{d_1} & 0 & 0 & \dots & \dots & \dots & 0 \\ 0 & 1 & \frac{c_2}{d_2 - a_2 \frac{c_1}{d_1}} & 0 & \dots & \dots & \dots & 0 \\ 0 & a_3 & d_3 & c_3 & \dots & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & a_{N-1} & d_{N-1} & c_{N-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & a_N & d_N \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} \frac{b_1}{d_1} \\ \frac{b_2 - a_2 \frac{b_1}{d_1}}{d_2 - a_2 \frac{c_1}{d_1}} \\ b_3 \\ \vdots \\ \vdots \\ b_N \end{pmatrix}$$

Zakładając, że możemy powtórzyć te kroki bez dzielenia przez zero, układ równań zostanie zredukowany do górnej postaci trójkątnej, gdzie  $h_1 = \frac{c_1}{d_1}$  i  $p_1 = \frac{b_1}{d_1}$ . Następnie powtarzamy dla innych elementów:

$$h_i = \frac{c_i}{d_i - a_i h_{i-1}}, \quad p_i = \frac{b_i - a_i p_{i-1}}{d_i - a_i h_{i-1}}.$$

Wreszcie podstawienie wsteczne prowadzi do jednoznacznego rozwiązania niewiadomych:

$$x_i = p_i - h_i x_{i-1}; \quad i = n-1, n-2, \dots, 1, \quad x_N = p_N.$$

## Część III

# Zastosowanie metody Cranka - Nicolsona Część praktyczna



Rysunek 2: Zaimportowanie potrzebnych bibliotek, takich jak matplotlib. Podanie ilości kroków czasowych - co najmniej 100. Zdefiniowanie macierzy.

```
from cmath import sin, pi
from numpy import zeros

import matplotlib.pyplot as p
from mpl_toolkits.mplot3d import Axes3D
from vpython import *

Max = 201; n = 100; m = 100
Ta = zeros(Max, float); Tb = zeros(Max, float); Tc = zeros(Max, float)
Td = zeros(Max, float); a = zeros(Max, float); b = zeros(Max, float)
c = zeros(Max, float); d = zeros(Max, float); x = zeros(Max, float)
t = zeros((Max, Max), float)
```

Rysunek 3: Zdefiniowanie sposobu obliczeń dla macierzy trójdiagonalnej.

```
def Tridiag(a, d, c, b, Ta, Td, Tc, Tb, x, n):
    Max = 201
    h = zeros(Max, float)
    p = zeros(Max, float)
    for i in range(1, n + 1):
        a[i] = Ta[i]
        b[i] = Tb[i]
        c[i] = Tc[i]
        d[i] = Td[i]
    h[1] = c[1] / d[1]
    p[1] = b[1] / d[1]
    for i in range(2, n + 1):
        h[i] = c[i] / (d[i] - a[i] * h[i - 1])
        p[i] = (b[i] - a[i] * p[i - 1]) / (d[i] - a[i] * h[i - 1])
    x[n] = p[n]
    for i in range(n - 1, 1, -1): x[i] = p[i] - h[i] * x[i + 1]
```

Rysunek 4: Określenie parametrów wykresu.

```
width = 1.0; height = 0.1; ct = 1.0
for i in range(0, n):    t[i, 0] = 0.0
for i in range(1, m):    t[0][i] = 0.0
h = width / (n - 1)
k = height / (m - 1)
r = ct * ct * k / (h * h)

for j in range(1, m + 1):
    t[1, j] = 0.0
    t[n, j] = 0.0
for i in range(2, n):    t[i][1] = sin(pi * h * i)
for i in range(1, n + 1): Td[i] = 2. + 2./r
Td[1] = 1.; Td[n] = 1.
for i in range(1, n + 1): Ta[i] = -1.0;   Tc[i] = -1.0;
Ta[n - 1] = 0.0; Tc[1] = 0.0; Tb[1] = 0.0; Tb[n] = 0.0
print("I'm working hard, wait for fig while I count to 100")
```

Rysunek 5: Podanie rozwiązania układu równań. Wizualizacja wykresu.

```
for j in range(2, m + 1):
    print(j)
    for i in range(2, n): Tb[i] = t[i - 1][j - 1] + t[i + 1][j - 1] + (2 / r - 2) * t[i][j - 1]
    Tridiag(a, d, c, b, Ta, Td, Tc, Tb, x, n)
    for i in range(1, n + 1): t[i][j] = x[i]
x = list(range(1, 101))
y = list(range(1, m + 1))
X, Y = p.meshgrid(x, y)

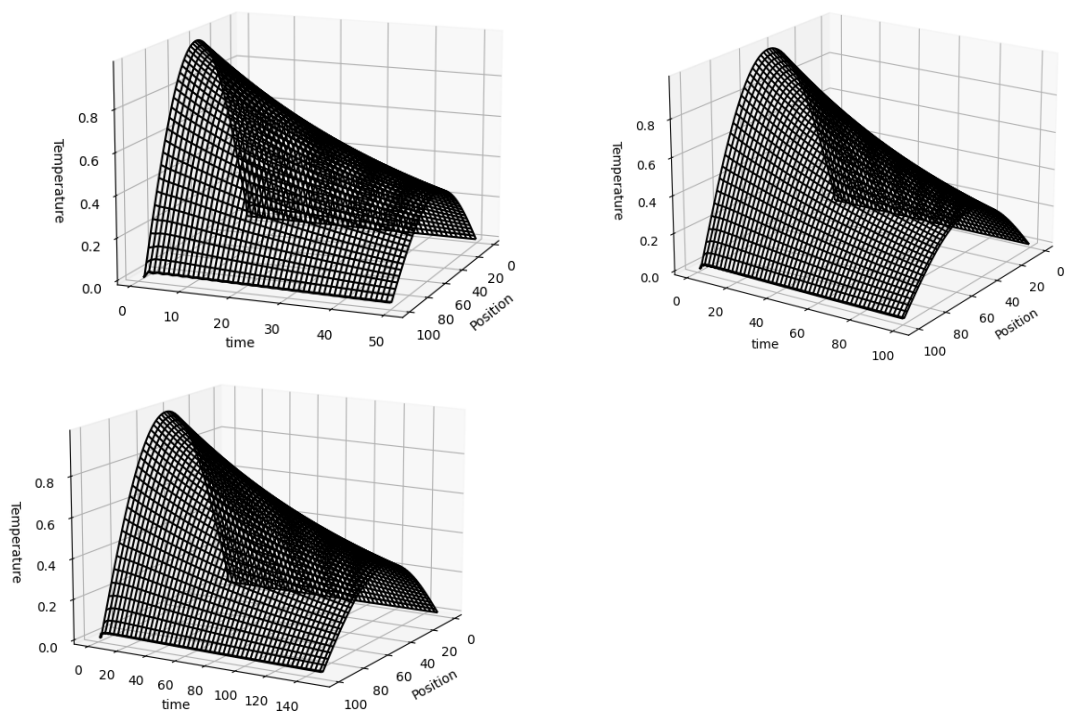
def functz(t):
    z = t[X, Y]
    return z

Z = functz(t)
fig = p.figure()
ax = Axes3D(fig)
ax.plot_wireframe(X, Y, Z, color = 'k')
ax.set_xlabel('Position')
ax.set_ylabel('time')
ax.set_zlabel('Temperature')
p.show()
print("Finnished")
```

Część IV

**Zastosowanie metody  
Cranka - Nicolsona  
Wnioski**

Wygenerowaliśmy wykresy dla 3 różnych ilości skoków czasowych, kolejno dla: 50, 100, i 150:



Rysunek 6: Z powyższych wykresów możemy wywnioskować że przy zastosowaniu metody Cranka - Nicolsona, zmiana ilości kroków czasowych nie wpływa na otrzymywany wynik. Temperatura jest wyrażona w  $[100^{\circ}\text{C}]$ , czas w  $[\text{s}]$ , a pozycja w  $[\text{cm}]$ .

Podsumowując, zwykle schemat Cranka – Nicolsona jest najdokładniejszym schematem dla małych kroków czasowych. W przypadku większych kroków czasowych schemat niejawnny działa lepiej, ponieważ jest mniej wymagający obliczeniowo. Jasny schemat jest najmniej dokładny i może być niestabilny, ale jest też najłatwiejszy do wdrożenia i najmniej intensywny liczbowo.