

Решение одномерного параболического уравнения методом сеток.
(Простейшая явная и неявная схемы)

Постановка задачи

Решить одномерное уравнение теплопроводности методом сеток с помощью явной и неявной схемы. Уравнение имеет вид:

$$\frac{\partial u}{\partial t} = a_0(x, t) \frac{\partial^2 u}{\partial x^2} + a_1(x, t) \frac{\partial u}{\partial x} + a_2(x, t)u + f(x, t) \quad (1)$$

граничные условия:

$$\frac{\partial u}{\partial x}(a, t) = \psi_0; \quad \frac{\partial u}{\partial x}(b, t) = \psi_1 \quad (2)$$

начальные данные:

$$u(x, 0) = \phi(x) \quad (3)$$

Условие данной задачи:

$$\begin{aligned} a_0 = 1 \quad a_1 = 0 \quad a_2 = -1 \quad f = 0 \\ \phi(x) = \frac{1}{(1+x^2)^2} \quad \psi_0 = \psi_1 = 0 \end{aligned} \quad (4)$$

Вывод формул

Введем сетку: $x_i = ih$, $t_k = k\tau$, $i = 0 \dots n$; $k = 0 \dots M$, где $h = \frac{1}{n}$; $\tau = \frac{1}{2M}$.
Область: $x \in [0, 1]$; $t \in [0, 1]$. Численное решение ищут в виде: $u_i^k \approx u(x_i, t_k)$
Сеточные уравнения:

$$\mathcal{L}_h u_i^k = a_0 \frac{u_{i+1}^k - 2u_i^k + u_{i-1}^k}{h^2} + a_1 \frac{u_{i+1}^k - u_{i-1}^k}{h} + a_2 u_i^k \quad (5)$$

Аппроксимация граничных условий:

$$\frac{-u_2^k + 4u_1^k - 3u_0^k}{2h} = \psi_0(t_k) \quad (6)$$

$$\frac{u_{n-2}^k - 4u_{n-1}^k + 3u_n^k}{2h} = \psi_1(t_k) \quad (7)$$

1. Простейшая явная схема:

$$\frac{u_i^{k+1} - u_i^k}{\tau} = \mathcal{L}_h u_i^k + f(x_i, t_k) \quad (8)$$

$$u_i^{k+1} = A_i^k u_{i-1}^k + B_i^k u_i^k + C_i^k u_{i+1}^k + D_i^k \quad (9)$$

здесь:

$$\begin{aligned} A_i^k &= \sigma a_0 - \frac{h}{2} \sigma a_1 & B_i^k &= 1 - 2\sigma a_0 + \tau a_2 & \sigma &= \frac{\tau}{h^2} \\ C_i^k &= \sigma a_0 + \frac{h}{2} \sigma a_1 & D_i^k &= \tau f(x_i, t_k) \end{aligned} \quad (10)$$

Схема устойчива при условии: $\sigma = \frac{\tau}{h^2} < 0.5$

2. Простейшая неявная схема:

$$\frac{u_i^k - u_i^{k-1}}{\tau} = \mathcal{L}_h u_i^k + f(x_i, t_k) \quad (11)$$

Прямой счет невозможен. Необходимо решать систему с тремя неизвестными:

$$A_i^k u_{i-1}^k - B_i^k u_i^k + C_i^k u_{i+1}^k = D_i^k \quad (12)$$

здесь:

$$\begin{aligned} A_i^k &= \sigma a_0 - \frac{h}{2} \sigma a_1 & B_i^k &= 1 + 2\sigma a_0 - \tau a_2 & \sigma &= \frac{\tau}{h^2} \\ C_i^k &= \sigma a_0 + \frac{h}{2} \sigma a_1 & D_i^k &= -\tau f(x_i, t_k) - u_i^{k-1} \end{aligned} \quad (13)$$

На каждом слое дополняем систему уравнений граничными условиями. Затем используем метод матричной прогонки, приводя систему к виду:

$$u_i^k = \alpha_i u_{i+1}^k + \beta_i \quad (14)$$

где:

$$\alpha_i = \frac{C_i^k}{B_i^k - A_i^k \alpha_{i-1}}; \beta_i = \frac{A_i^k \beta_{i-1} - D_i^k}{B_i^k - A_i^k \alpha_{i-1}}$$

Текст программы

Явная схема

```
1 import math
import numpy as np
from numpy import *
n = raw_input('n = '); n = int(n)
M = raw_input('M = '); M = int(M)
6 a = 0.0; b = 1.0; T = 1.0
h = (b - a) / n; tau = T / M; sigma = tau / h**2
if (sigma > 0.5) is True:
    print 'sigma =', sigma, ' — disturbed stability condition warning'
def fi(x):
11     return 1.0 / (1.0 + x**2)**2
uk = zeros(n+1); ukk = zeros(n+1)
for i in range(n+1):
    uk[i] = fi(a+i*h)
for k in range(M):
16     for i in range(1,n):
        ukk[i] = sigma * uk[i-1] + (1 - 2*sigma-tau) * uk[i] + sigma * uk[i+1]
        ukk[0] = (4.0*ukk[1] - ukk[2]) / 3.0
        ukk[n] = (4.0*ukk[n-1] - ukk[n-2]) / 3.0
        for i in range(n+1):
21            uk[i] = ukk[i]
        if ((k+1)%(M/10)==0):
            print k+1, ' ', uk
```

Неявная схема

```

n = raw_input('n = '); n = int(n)
M = raw_input('M = '); M = int(M)
a = 0.0; b = 1.0; T = 1.0
h = (b - a) / n; tau = T / M; sigma = tau / h**2
def fi(x):
    return 1.0 / (1.0 + x**2)**2
7 u_minus = zeros(n+1); u = zeros(n+1)
for i in range(n+1):
    u_minus[i] = fi(a+i*h)
for k in range(1,M+1):
    alpha = [0]; beta = [0]
12     alpha.append( 2.0*sigma / (2.0*sigma + 3.0*tau + 3.0) )
    beta.append( 3.0*u_minus[1] / (2.0*sigma + 3.0*tau + 3.0) )
    for i in range(2,n):
        alpha.append( sigma / ( 2.0*sigma + tau + 1.0 - sigma * alpha[i-1] ) )
        beta.append( (sigma*beta[i-1] + u_minus[i]) / ( 2.0*sigma + tau + 1.0
- sigma * alpha[i-1] ) )
17     u[n] = ((4.0 - alpha[n-2])*beta[n-1] - beta[n-2]) / (3.0 + alpha[n-1]*(alpha[n
-2] - 4.0))
    for i in range(n-1,0,-1):
        u[i] = alpha[i]*u[i+1] + beta[i]
    u[0] = (4.0*u[1] - u[2]) / 3.0
    for i in range(n+1):
22         u_minus[i] = u[i]
    if ((k+1)%M/10==0):
        print k+1, ' ', u

```

Результаты

Явная схема

50	[0.70458405	0.69890241	0.68185749	0.65513983	0.62139643	0.58396558
	0.54654555	0.51283034	0.48614708	0.46912919	0.46345656]	
100	[0.56955818	0.56762369	0.56182022	0.55272134	0.54122624	0.52847091
	0.5157158	0.50422127	0.49512309	0.48932017	0.48738587]	
150	[0.49223038	0.49157115	0.48959346	0.48649274	0.48257539	0.4782285
	0.47388162	0.46996428	0.46686357	0.46488589	0.46422667]	
200	[0.437534	0.43730934	0.43663536	0.43557866	0.43424366	0.43276227
	0.43128088	0.42994588	0.42888918	0.4282152	0.42799054]	
250	[0.39324477	0.39316821	0.39293852	0.39257841	0.39212345	0.3916186
	0.39111376	0.3906588	0.39029868	0.39006899	0.38999243]	
300	[0.35494074	0.35491465	0.35483637	0.35471365	0.3545586	0.35438655
	0.35421451	0.35405946	0.35393674	0.35385846	0.35383237]	
350	[0.3208831	0.32087421	0.32084754	0.32080571	0.32075287	0.32069424
	0.32063561	0.32058277	0.32054095	0.32051427	0.32050538]	
400	[0.29026949	0.29026646	0.29025737	0.29024312	0.29022511	0.29020513
	0.29018515	0.29016714	0.29015289	0.2901438	0.29014077]	
450	[0.26263662	0.26263559	0.26263249	0.26262763	0.26262149	0.26261468
	0.26260788	0.26260174	0.26259688	0.26259378	0.26259275]	
500	[0.2376548	0.23765445	0.23765339	0.23765174	0.23764965	0.23764733
	0.237645	0.23764291	0.23764126	0.2376402	0.23763985]	

Неявная схема

5	50	[0.69773927	0.6922835	0.67591621	0.65025395	0.61783133	0.58185086
			0.54586704	0.51343569	0.48776277	0.47138709	0.46592853]	
10	100	[0.56499475	0.56317861	0.55773019	0.54918786	0.53839572	0.52642016
			0.51444455	0.50365228	0.4951098	0.48966126	0.48784508]	
15	150	[0.48911514	0.48851069	0.48669736	0.48385432	0.48026251	0.47627685
			0.4722912	0.46869939	0.46585635	0.46404301	0.46343856]	
20	200	[0.43518274	0.43498157	0.43437807	0.43343186	0.43223645	0.43090997
			0.42958348	0.42838807	0.42744187	0.42683836	0.42663719]	
25	250	[0.39128646	0.39121951	0.39101866	0.39070375	0.3903059	0.38986442
			0.38942295	0.3890251	0.38871018	0.38850933	0.38844238]	
30	300	[0.35320187	0.35317959	0.35311274	0.35300793	0.35287552	0.35272859
			0.35258167	0.35244925	0.35234445	0.3522776	0.35225532]	
35	350	[0.31928759	0.31928017	0.31925792	0.31922304	0.31917897	0.31913007
			0.31908117	0.3190371	0.31900222	0.31897998	0.31897256]	
40	400	[0.28878434	0.28878187	0.28877447	0.28876286	0.28874819	0.28873192
			0.28871564	0.28870098	0.28868937	0.28868196	0.2886795]	
45	450	[0.26124673	0.26124591	0.26124345	0.26123958	0.2612347	0.26122928
			0.26122387	0.26121899	0.26121512	0.26121266	0.26121184]	
50	500	[0.23635217	0.2363519	0.23635108	0.23634979	0.23634817	0.23634636
			0.23634456	0.23634294	0.23634165	0.23634083	0.23634056]	