

Квазилинейное уравнение переноса

1. Постановка задачи

Используя схему бегущего счёта и итерационные методы, решить задачу:

$$\begin{cases} \frac{\partial u}{\partial t} - \frac{1}{1+u} \frac{\partial u}{\partial x} = 0 \\ u(x, 0) = \cos \frac{\pi x}{2}, \quad -1 < x \leq 0 \\ u(0, t) = e^{-t} \end{cases}$$

2. Метод решения

Преобразуем данное уравнение, написав его в дивергентной форме:

$$\frac{\partial u}{\partial t} - \frac{\partial(\ln(u+1))}{\partial x} = 0$$

Введем разностную сетку в области $\Omega = \{(x, t) : -1 \leq x < 0, 0 < t < T\}$:

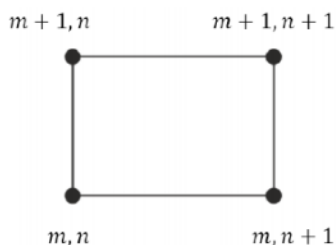
$$\omega_{h_x, h_t} = \begin{cases} x_n = n \cdot h_x, \quad h_x = \frac{1}{N}, \quad n = \overline{0, N} \\ t_m = m \cdot h_t, \quad h_t = \frac{1}{M}, \quad m = \overline{0, M} \end{cases}$$

где N – число узлов вдоль оси x , M – число узлов вдоль оси t .

На ω_{h_x, h_t} будем рассматривать сеточную функцию $y_n^m = u(x_n, t_m)$

Шаблон

Для рассматриваемой задачи будем использовать четырехточечный шаблон. Он безусловно устойчив и аппроксимирует задачу как $O(h_x^2 + h_t^2)$



Таким образом, разностная схема задачи имеет вид:

$$\frac{y_n^{m+1} - y_n^m + y_{n+1}^{m+1} - y_{n+1}^m}{2h_t} - \frac{\ln(y_n^{m+1} + 1) - \ln(y_n^m + 1) + \ln(y_{n+1}^{m+1} + 1) - \ln(y_{n+1}^m + 1)}{2h_x} = 0$$

Также начальное и граничное условия:

$$\begin{cases} y_n^0 = \cos \frac{\pi n h_x}{2} \\ y_0^m = e^{-mh_t} \end{cases}$$

Схема бегущего счета

Данную задачу будем решать при помощи схемы бегущего счета. Значение сеточной функции y_{n+1}^{m+1} неизвестно, но нам известны все значения, соответствующие начальному (y_n^0) и граничному (y_0^m) условиям. Таким образом, зная значения в трех соседних точках: y_0^0, y_1^0, y_0^1 мы можем численно найти значение в четвертой точке y_1^1 . Зная это значение, мы можем найти по трем известным точкам либо y_1^2 , либо y_2^1 . И так далее, заполняя найденными значениями сетку.

Также начальное и граничное условия:

$$\begin{cases} y_n^0 = \cos \frac{\pi n h_x}{2} \\ y_0^m = e^{-mh_t} \end{cases}$$

Это неявное уравнение относительно y_{n+1}^{m+1} . Будем решать его итерационным методом Ньютона.

$$y_{n+1}^{m+1(s+1)} = y_{n+1}^{m+1(s)} - \frac{f(y_{n+1}^{m+1(s)})}{f'(y_{n+1}^{m+1(s)})}$$

$$\text{До достижения заданной точности } \varepsilon: \quad |y_{n+1}^{m+1(s+1)} - y_{n+1}^{m+1(s)}| \leq \varepsilon$$

Исследование характеристик

В решении задачи используется промежуток времени, на котором не происходит пересечений характеристик уравнения, в точках пересечения проекций его характеристик, решение будет разрывным. Посмотрим как будут вести себя проекции характеристик в заданных областях.

$$\frac{dt}{1} = \frac{-(1+u)dx}{1} = \frac{du}{0}$$

$$\int_{u_0}^u du = 0$$

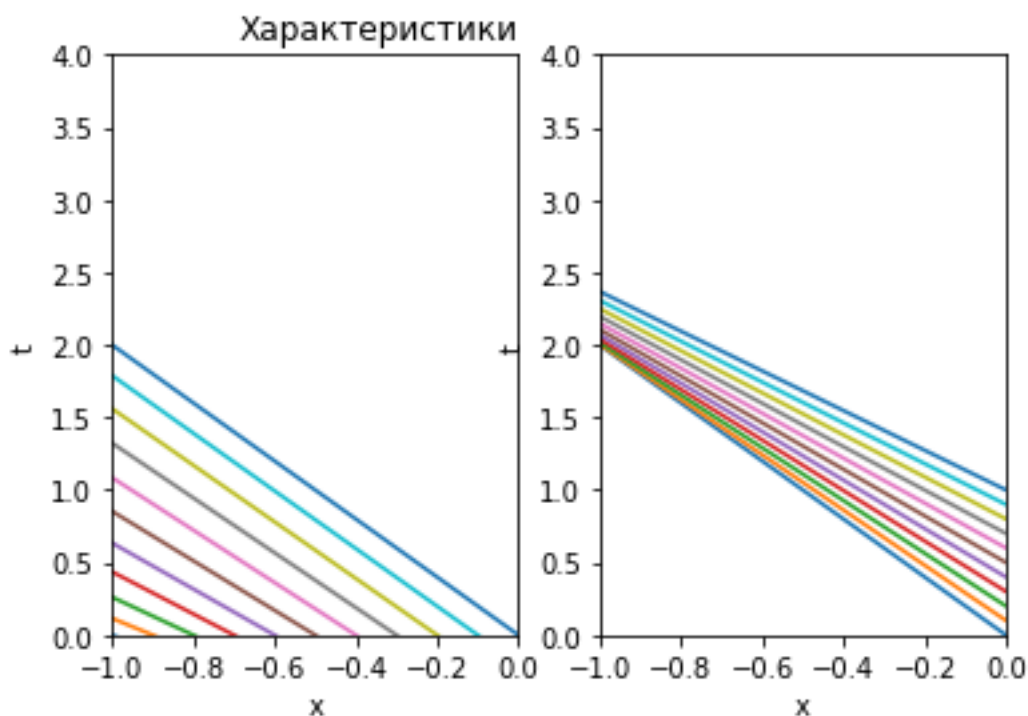
$$\int_{x_0}^x dx = - \int_{t_0}^t \frac{1}{1+u} dt$$

$$t = (1 + u_0)(x_0 - x) + t_0$$

Воспользуемся начальным и граничным условиями для получения двух семейств кривых:

$$1) \quad t_0 = 0, u_0 = \cos \frac{\pi x_0}{2} : \quad t = (1 + \cos \frac{\pi x_0}{2})(x_0 - x)$$

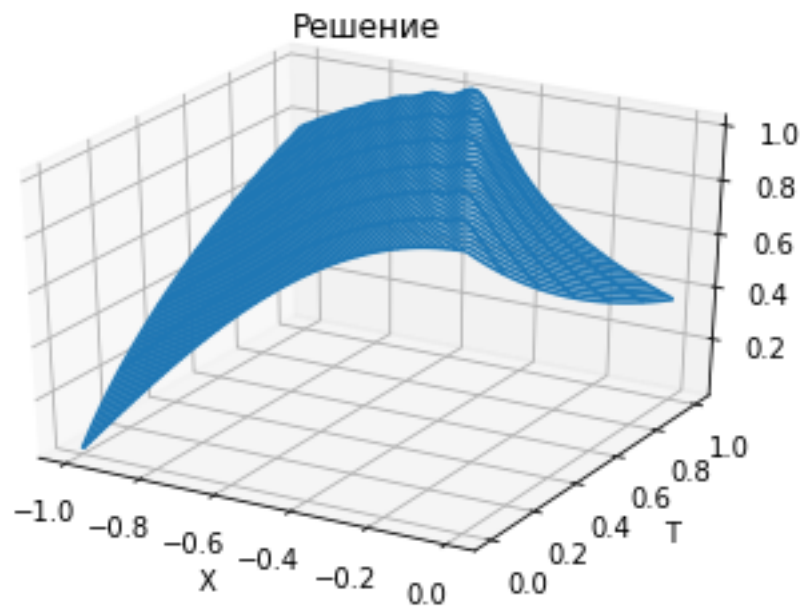
$$2) \quad x_0 = 0, u_0 = e^{-t_0} : \quad t = (1 + e^{-t_0})(-x) + t_0$$



Как видим, в заданных областях пересечений нет. Следовательно, нет так называемого опрокидывания волны, и во всей области решение будет представимо через разностную схему.

Точное решение

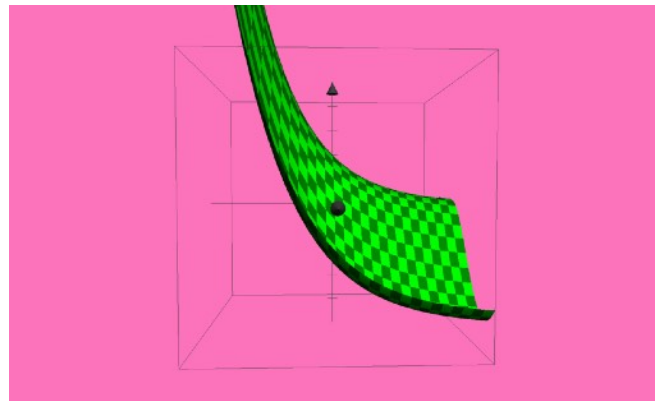
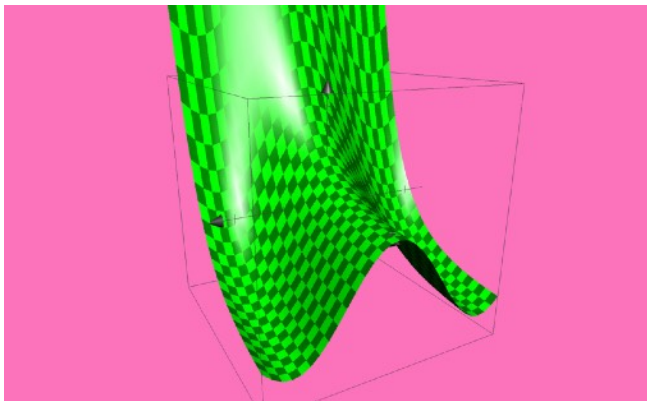
Введем функцию $\Phi(C1, C2)$. Заметим, что данная функция неявно задаёт искомую функцию $u(x, t)$. Подставим граничные условия:



$$\begin{cases} \Phi(x(1 + \cos \frac{\pi x}{2}), \cos(\frac{\pi x}{2})) = 0, t = 0 \\ \Phi(t, e^{-t}) = 0, x = 0 \end{cases}$$

Заметим, что $\Phi((1+u)x + t, u) = 0$ станет тождеством при

$$u(x, t) = \cos(\frac{\pi x}{2}) + e^{-t} - 1 - \text{аналитическое решение}$$



Устойчивость и аппроксимация

Критерий Неймана(необходимый)

Зафиксируем коэффициент перед $\frac{\partial u}{\partial x}$. Выберем произвольную точку (x_0, t_0) исследуемой области Ω и обозначим $C = \frac{1}{1 + u(x_0, t_0)}$. Теперь исследуемая схема приобретет вид:

$$\frac{U_{n+1}^{m+1} - U_{n+1}^m + U_n^{m+1} - U_n^m}{h_t} + C \frac{U_{n+1}^{m+1} - U_n^{m+1} + U_{n+1}^m - U_n^m}{h_x} = 0$$

Будем искать решение данного уравнения в виде $U_n^m = \rho^m e^{i\alpha x_n}$. Подставив замену в уравнение, получим:

$$qe^{i\alpha} - e^{i\alpha} + q - 1 + \frac{Ch_t}{h_x}(qe^{i\alpha} - q + e^{i\alpha} - 1) = 0$$

Тогда для q получим:

$$q = \frac{e^{i\alpha} + 1 + \frac{Ch_t}{h_x}(1 - e^{i\alpha})}{e^{i\alpha} + 1 + \frac{Ch_t}{h_x}(e^{i\alpha} - 1)}$$

$$|q| = 1$$

Из данного соотношения получаем, что условие $|q(\alpha)| \leq 1$ справедливо для любых соотношений шагов по координате и времени, и, следовательно, спектральный критерий Неймана выполнен.

Критерий Куранта(достаточный)

Перепишем исследуемую разностную схему, поставив для нее задачу в виде:

$$\begin{cases} \frac{U_{n+1}^{m+1} - U_{n+1}^m + U_n^{m+1} - U_n^m}{h_t} + C \frac{U_{n+1}^{m+1} - U_n^{m+1} + U_{n+1}^m - U_n^m}{h_x} = \varepsilon_n^m \\ U_n^0 = \phi_n \\ U_0^m = \mu^m \end{cases}$$

Преобразуем уравнение к виду:

$$U_{n+1}^{m+1}(1 + \frac{Ch_t}{h_x}) + U_n^{m+1}(1 - \frac{Ch_t}{h_x}) = U_{n+1}^m(1 - \frac{Ch_t}{h_x}) + U_n^m(1 + \frac{Ch_t}{h_x}) + h_t \varepsilon_n^m,$$

где ε_n^m - некоторое возмущение исходной схемы.

Оценим данное соотношение по равномерной норме:

$$\|U^{m+1}\|(1 + \frac{Ch_t}{h_x}) + \|U^{m+1}\|(1 - \frac{Ch_t}{h_x}) \leq \|U^m\|(1 - \frac{Ch_t}{h_x}) + \|U^m\|(1 + \frac{Ch_t}{h_x}) + h_t \|\varepsilon^m\|$$

$$2\|U^{m+1}\| \leq 2\|U^m\| + h_t \|\varepsilon\|$$

Тогда по индукции получаем:

$$\|U^m\| \leq \|\phi\| + \frac{mh_t}{2} \|\varepsilon\|$$

$$\|U^m\| \leq \|\phi\| + \frac{T}{2} \|\varepsilon\|$$

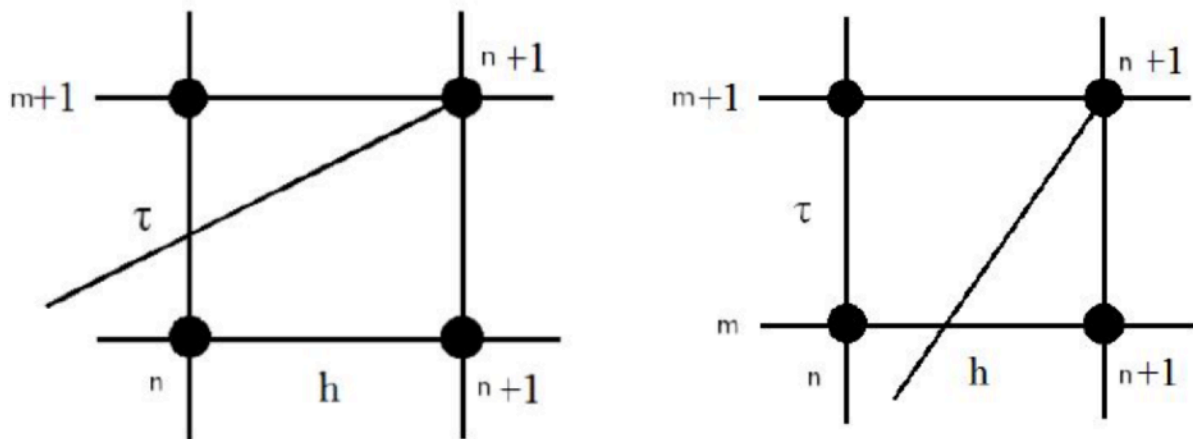
Переобозначая:

$$\|U^m\| \leq M \|\phi\| + N \|\varepsilon\|,$$

где T - величина интервала времени, на котором мы ищем решение, а M и N - константы, не зависящие от шагов сетки. Таким образом получаем, что и критерий Куранта выполнен для рассматриваемой схемы для любых соотношений шагов по времени и координате, то есть фактически выполнено определение устойчивости.

Геометрический критерий¶

Рассмотрим исследуемый шаблон. Пусть через точку (x_{n+1}, t^{m+1}) , в которой ищется значение исследуемой функции, проходит одна из характеристик. Исходя из того, что в остальных точках шаблона значение исследуемой функции известно, при любых соотношениях шагов по времени и координате характеристика пересечет отрезок, соединяющий точки



шаблона, в которых значение исследуемой функции известно.

Таким образом, исследуемая схема является безусловно устойчивой, согласно условиям Неймана и Куранта.

Порядок аппроксимации

Вычислим порядок аппроксимации. Для этого разложим значения функции U в узлах сетки в ряд до члена третьего порядка включительно в точке $(x_n + \frac{h_x}{2}; t_m + \frac{h_t}{2})$:

$$U^{m+1} = U^{m+0.5} + \frac{h_t}{2} U'_{m+0.5} + \frac{1}{2} \frac{h_t^2}{4} U''_{m+0.5} + \frac{1}{6} \frac{h_t^3}{8} U'''_{m+0.5} + O(h_t^4)$$

$$U^m = U^{m+0.5} - \frac{h_t}{2} U'_{m+0.5} + \frac{1}{2} \frac{h_t^2}{4} U''_{m+0.5} - \frac{1}{6} \frac{h_t^3}{8} U'''_{m+0.5} + O(h_t^4)$$

$$U_{n+1} = U_{n+0.5} + \frac{h_x}{2} U'_{n+0.5} + \frac{1}{2} \frac{h_x^2}{4} U''_{n+0.5} + \frac{1}{6} \frac{h_x^3}{8} U'''_{n+0.5} + O(h_x^4)$$

$$U_n = U_{n+0.5} - \frac{h_x}{2} U'_{n+0.5} + \frac{1}{2} \frac{h_x^2}{4} U''_{n+0.5} - \frac{1}{6} \frac{h_x^3}{8} U'''_{n+0.5} + O(h_x^4)$$

Из данных соотношений получим:

$$\frac{U_{n+1}^{m+1} - U_{n+1}^m + U_n^{m+1} - U_n^m}{h_t} + C \frac{U_{n+1}^{m+1} - U_n^{m+1} + U_{n+1}^m - U_n^m}{h_x} - \frac{\partial U}{\partial t} - C \frac{\partial U}{\partial x} = O(h_t^2 + h_x^2)$$

Погрешности

M,N		delta
	10	0,6794
	20	0,2846
	40	0,1304
	60	0,0625
	80	0,0845
	100	0,0496

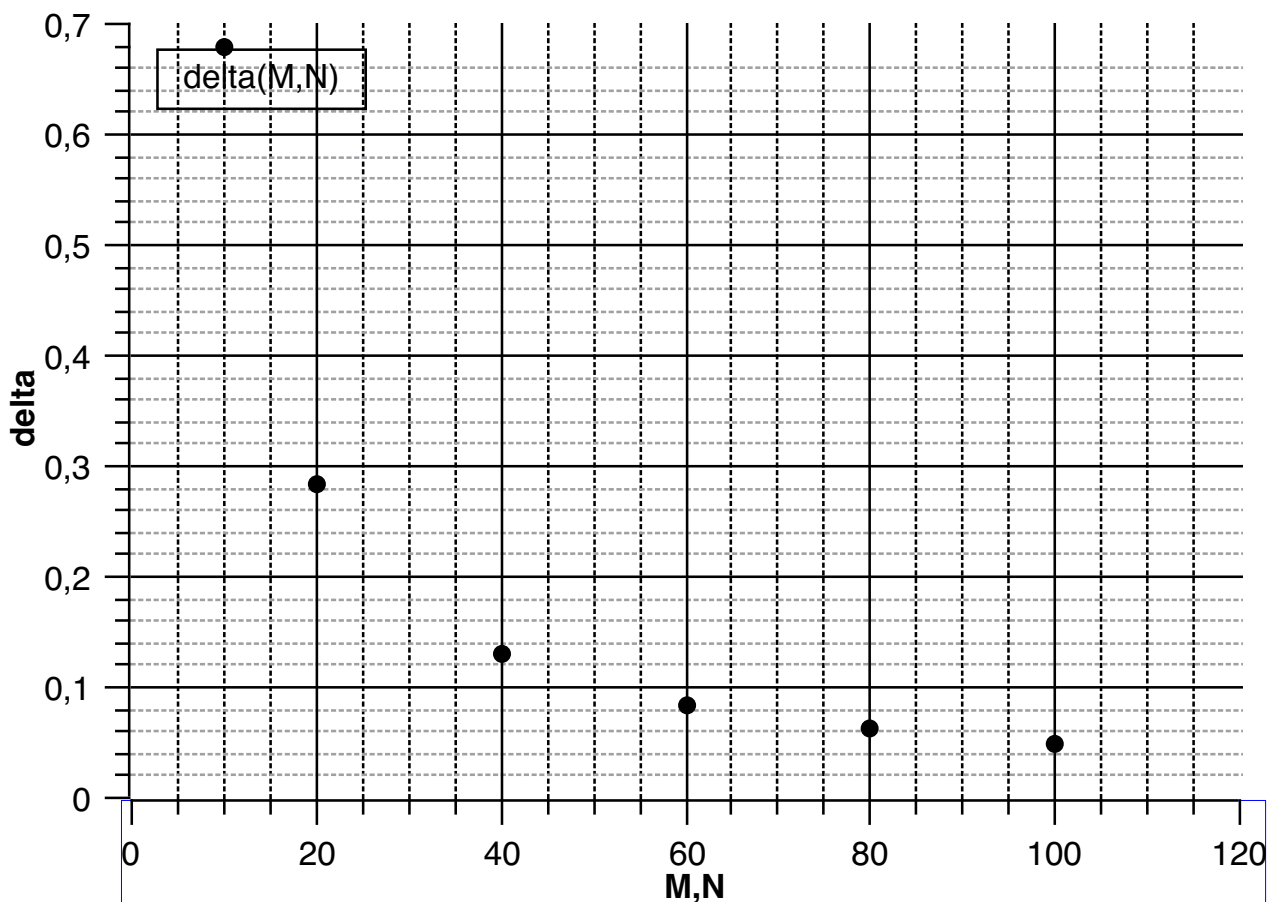
Вычитая решение полученное методом бегущего счета из точного решения получаем погрешность. По графику видно, что погрешность убывает в 2^p раз

Код программы

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
```

Created on Thu May 16 11:00:54 2019

Зависимость погрешности от шага



@author: Mikhailov Mikhail

|||||

```
import matplotlib.pyplot as plt
import numpy as np
import math as mt
from mpl_toolkits.mplot3d import axes3d
```

```
# Зададим:  $\epsilon$  - точность в методе Ньютона, N - количество шагов по x,
# M - количество шагов по y, а также границы нашей сетки.
```

```
epsilon = 0.00001
N = 60; M = 60
T_begin = 0; T_end = 1
X_begin = 0; X_end = -1
```

```
# Соответственно, элементарные шаги.
```

```
h_x = (X_end - X_begin) / (N - 1)
h_t = (T_end - T_begin) / (M - 1)
```

```
# Создадим двумерный массив размерами с нашу сетку (N×M),
```


в ячейках которого будут храниться соответствующие искомые значения.

```
y=np.zeros((M,N))
```

Начнем заполнять его начальным и граничным значениями.

```
for n in range(N):  
    y[0][n] = (mt.cos(mt.pi * h_x * n * 0.5))
```

```
for m in range(M):  
    y[m][0] = mt.exp(-h_t * m)
```

Определим вспомогательные функции.

```
def F(m,n):  
    return mt.log(y[m][n] + 1)
```

```
def df(mp1,np1):  
    return (1/(2*h_t) - 0.5/(h_x*(y[mp1][np1]+1)))
```

Разностная схема будет иметь вид.

```
def f(mp1, np1):  
    n = np1-1  
    m = mp1-1  
    de = (y[mp1][n]-y[m][n] + y[mp1][np1]-y[m][np1]) / (2.*h_t) - (F(mp1, np1)-F(mp1,n) +  
F(m, np1)-F(m,n)) / (2.*h_x)  
    return (de)
```

Перейдем к методу Ньютона, пробегаая по всей сетке.

```
eps = epsilon + 1;  
while eps > epsilon:  
    eps = 0  
    for m in range(M)[0:M-1]:  
        for n in range(N)[0:N-1]:  
            ep = f(m+1, n+1) / df( m+1, n+1)  
            y[m+1][n+1] = y[m+1][n+1] - ep  
            if abs(ep) > eps:  
                eps = abs(ep)
```

Построим график решения.

```
tm = np.linspace(T_begin,T_end, num=M)  
xn = np.linspace(X_begin,X_end, num=N)  
X, T = np.meshgrid(xn, tm)  
fig = plt.figure()  
ax = fig.add_subplot(111, projection='3d')  
surf_1 = ax.plot_wireframe(X , T , y, rstride=10,cstride=1)  
plt.title('Решение')  
plt.xlabel('X')  
plt.ylabel('T')
```

```
plt.show()
```

```
# Погрешности
```

```
print('y=',y[5][5])
```

```
print('yтеор=',mt.cos(mt.pi*xn[5]/2)+mt.exp(-tm[5])-1)
```

```
print(y[5][5]-(mt.cos(mt.pi*xn[5]/2)+mt.exp(-tm[5])-1))
```