ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В. ЛОМОНОСОВА»

ФИЗИЧЕСКИЙ ФАКУЛЬТЕТ
КАФЕДРА КВАНТОВОЙ ЭЛЕКТРОНИКИ

ОТЧЁТ
по практическому заданию №4
по курсу
«ЧИСЛЕННЫЕ МЕТОДЫ В ФИЗИКЕ»

Студентки 427 группы
Кузьменок Дианы Андреевны
(T0 = 1.2)

Москва
2023

Содержание

1 Постановка задачи **3**

2 Методы **3**

3 Устойчивость **4**

3.1 Явная схема 4

3.2 Неявная схема 5

4 Диффузия **5**

5 Листинг программы **6**

6 Результаты **11**

1 Постановка задачи

Численно решить уравнение теплопроводности

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{\mathrm{d}^2T}{\mathrm{d}x^2}$$

на отрезке $x = [0, 100]$ для $T(0) = T(100) = 0$ и интервале времени $t = [0, 100]$ с начальными условиями:

$$T(x, t = 0) = T_0(\frac{x-x_0}{a_0})^2 e^{-(\frac{x-x_0}{a_0})^2}$$

где $x_0 = 50,\ a_0 = 10,\ T_0 = 1.2$

2 Методы

Для решения использованы следующие схемы:

- Явная схема численного решения
  $T_j^{n+1} = T_j^n + \alpha(T_{j+1}^n - 2T_j^n + T_{j-1}^n);$
  $\alpha = \dfrac{\Delta t}{(\Delta x)^2}$
- Схема Кранка-Николсона и метод прогонки
  $T_j^{n+1} = T_j^n + \alpha(T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}) \quad \Rightarrow$
  $T_{j+1}^{n+1} - (2+s)T_j^{n+1} + T_{j-1}^{n+1} = -sT_j^n;$
  $s = \dfrac{(\Delta x)^2}{\Delta t};$
  $x_{j-1} = \dfrac{-c_j}{b_j + a_j x_j},\ y_{j-1} = \dfrac{f_j - a_j y_j}{b_j + a_j x_j},\ x_{N-1} = y_{N-1} = 0;$ (прямая прогонка)
  $u_{j+1} = x_j u_j + y_j,\ u_0 = 0;$ (обратная прогонка)
  $u_j = T_j^{n+1},\ a_j = c_j = 1,\ b_j = -(2+s),\ f_j = -sT_j^n,$

3 Устойчивость

3.1 Явная схема

$$T_j^{n+1} = T_j^n + \alpha(T_{j+1}^n - 2T_j^n + T_{j-1}^n)$$

Пространств. Фурье-мода: $\hat{T}_\varkappa^n e^{i\varkappa x_j}$, $\underbrace{\hat{T}_\varkappa^{n+1} = \lambda \hat{T}_\varkappa^n}$

$$\hat{T}_\varkappa^{n+1} e^{i\varkappa x_j} = \hat{T}_\varkappa^n e^{ihx_j} + \alpha\left(\hat{T}_\varkappa^n e^{i\varkappa(x_j+\Delta x)} - 2\hat{T}_\varkappa^n e^{i\varkappa x_j} + \hat{T}_\varkappa^n e^{i\varkappa(x_j-\Delta x)}\right)$$

$$\hat{T}_\varkappa^{n+1} e^{i\varkappa x_j} = \hat{T}_n^k e^{i\varkappa x_j}\left[1 + \alpha(e^{i\varkappa\Delta x} - 2 + e^{-i\varkappa\Delta x})\right]$$

$$\lambda = 1 - 2\alpha(1 - \cos\varkappa\Delta x) = 1 - 4\alpha\sin^2\frac{\varkappa\Delta x}{2}$$

$$|\lambda| = \left|1 - 4\alpha\sin^2\frac{\varkappa\Delta x}{2}\right| \le 1$$

$$1 - 4\alpha\sin^2\frac{\varkappa\Delta x}{2} \ge -1 \Rightarrow 4\alpha\sin^2\frac{\varkappa\Delta x}{2} \le 2$$

$$\alpha = \frac{\Delta t}{\Delta x^2} \le \frac{1}{2} \Rightarrow \boxed{\Delta t \le \frac{\Delta x^2}{2}}$$

а) $\Delta t = 0{,}01$; $\Delta x = 0{,}1$

$0{,}01 > \frac{0{,}01}{2} \Rightarrow$

неустойчива!

б) $\Delta t = 0{,}005$; $\Delta x = 0{,}1$

$0{,}005 \le 0{,}005 \Rightarrow$

устойчива

3.2 Неявная схема

$$T_j^{n+1} = T_j^n + \alpha\left(T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}\right)$$

Пространств. Фурье-мода: $\hat{T}_\varkappa^n e^{i\varkappa x_j}$. $\hat{T}_\varkappa^{n+1} = \lambda \hat{T}_\varkappa^n$

$$\hat{T}_\varkappa^{n+1} e^{i\varkappa x_j} = \hat{T}_\varkappa^n e^{i\varkappa x_j} + \alpha\left(\hat{T}_\varkappa^{n+1} e^{i\varkappa(x_j+\Delta x)} - 2\hat{T}_\varkappa^{n+1} e^{i\varkappa x_j} + \hat{T}_\varkappa^{n+1} e^{i\varkappa(x_j-\Delta x)}\right)$$

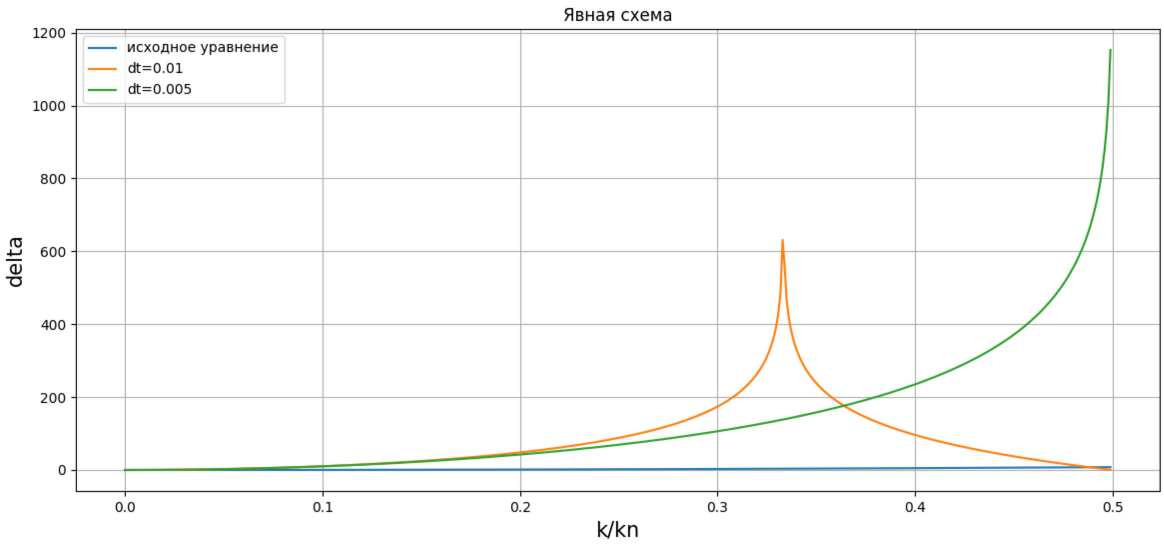
$$\hat{T}_\varkappa^{n+1} e^{i\varkappa x_j}\left[1 - \alpha\left(e^{i\varkappa\Delta x} - 2 + e^{-i\varkappa\Delta x}\right)\right] = \hat{T}_\varkappa^n e^{i\varkappa x_j}$$

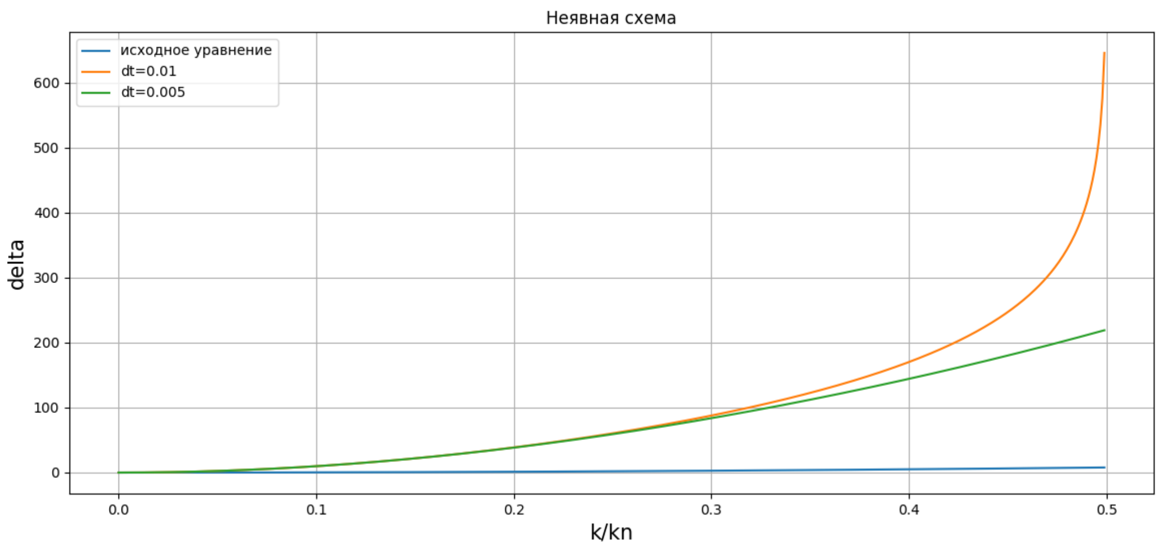
$$\lambda = \frac{1}{1-\alpha\left(e^{i\varkappa\Delta x} - 2 + e^{-i\varkappa\Delta x}\right)} = \frac{1}{1+4\alpha\sin^2\frac{\varkappa\Delta x}{2}}$$

$|\lambda| \leq 1 \Rightarrow$ безусл. устойчива

4 Диффузия

- Для исходного (непрерывного) уравнения: $\omega = i\kappa^2$ $\omega = i\delta \Rightarrow \delta = \kappa^2$
- Для явной схемы: $\delta\Delta t = -\ln\lambda \Rightarrow \delta = -\frac{1}{\Delta t}\ln\left(1 - 4\frac{\Delta t}{(\Delta x)^2}\sin^2(\frac{\pi\kappa}{2\kappa_n})\right)$, $\kappa_n = \frac{\pi}{\Delta x}$
- Для неявной схемы: $\delta\Delta t = -\ln\lambda \Rightarrow \delta = \frac{1}{\Delta t}\ln\left(\frac{1+2\frac{\Delta t}{(\Delta x)^2}\sin^2(\frac{\pi\kappa}{2\kappa_n})}{1-2\frac{\Delta t}{(\Delta x)^2}\sin^2(\frac{\pi\kappa}{2\kappa_n})}\right)$, $\kappa_n = \frac{\pi}{\Delta x}$





5 Листинг программы

```
import numpy as np
from tqdm import tqdm
import matplotlib.pyplot as plt

T0 = (1 + 5 + 12) / 15
x0 = 50
a0 = 10
dx = 0.1
Tf = 100
```

```
Xf = 100
dT1 = 0.01
dT2 = 0.005

def start():
  t0 = np.zeros(int(Xf / dx) + 1)
  f = lambda x: T0 * ((x - x0) / a0) ** 2 * np.exp(-((x - x0) / a0) ** 2)
  for i in range(int(Xf / dx) + 1):
    t0[i] = f(i / 10)
  return t0


def alf(dt):
  return(dt / (dx ** 2))


def s(dt):
  return 1 / alf(dt)


def expl(t0, dt):
  t1 = np.zeros(int(Xf / dx) + 1)
  for i in range(1, int(Xf / dx)):
    t1[i] = t0[i] + alf(dt) * (t0[i + 1] - 2 * t0[i] + t0[i - 1])
  return t1


def impl(t0, dt):
  xy = np.zeros((2, int(Xf / dx) - 1))
  t1 = np.zeros(int(Xf / dx) + 1)

  for i in range(int(Xf / dx) - 2, 0, -1):
    xy[0][i - 1] = -1 / (-2 - s(dt) + xy[0][i])
    xy[1][i - 1] = (-s(dt) * t0[i] - xy[1][i]) / (-2 - s(dt) + xy[0][i])

```

```python
    for i in range(1, int(Xf / dx) - 1):
        t1[i] = xy[0][i - 1] * t1[i - 1] + xy[1][i - 1]
    return t1

#step_t=0.01
Pp = [0, 10, 20, 30, 50, 100]
P = [Pi / 0.01 for Pi in Pp]
T1_e = []; T1_i = []
ex = start()
im = start()
T1_e.append(start()); T1_i.append(start())

for i in tqdm(range(1, int(Tf / dT1) + 1), mininterval = 1, position=0,
    leave=True):
    im_ex = im
    im = impl(im_ex, dT1)
    if i in P:
        T1_i.append(im)

Pp_bad = [0, 0.22, 0.25, 0.5, 3, 5]
P_bad = [Pi / 0.01 for Pi in Pp_bad]
for i in tqdm(range(1, 504), mininterval = 1, position=0, leave=True):
    ex_ex = ex
    ex = expl(ex_ex, dT1)
    if i in P_bad:
        T1_e.append(ex)

fig = plt.subplots(3, 2, figsize=(10, 15))
for i in range(len(P)):
    plt.subplot(3, 2, i+1)
    plt.plot(np.arange(0, 100 + dx, dx), T1_i[i], label = 'dt=0.01, t=%s' %
        Pp[i])
    plt.grid(True)
    plt.xlabel('x', size = 15)
    plt.ylabel('T', size = 15)
```

```
    plt.legend()
plt.savefig("1_i")

fig = plt.subplots(3, 2, figsize=(10, 15))
for i in range(len(Pp_bad)):
    plt.subplot(3, 2, i+1)
    plt.plot(np.arange(0, 100 + dx, dx), T1_e[i], label = 'dt=0.01, t=%s %
        Pp_bad[i])
    plt.grid(True)
    plt.xlabel('x', size = 15)
    plt.ylabel('T', size = 15)
    plt.legend(loc = 'upper right')
plt.savefig("1_e")

#step_t=0.005
Rr = [0, 10, 20, 30, 50, 100]
R = [Ri / 0.005 for Ri in Rr]
T2_e = []; T2_i = []
ex = start()
im = start()
T2_e.append(start()); T2_i.append(start())

for i in tqdm(range(1, int(Tf / dT2) + 1), mininterval = 1, position=0,
    leave=True):
    im_ex = im
    ex_ex = ex
    ex = expl(ex_ex, dT2)
    im = impl(im_ex, dT2)
    if i in R:
        T2_i.append(im)
        T2_e.append(ex)

fig = plt.subplots(3, 2, figsize=(10, 15))
for i in range(len(R)):
    plt.subplot(3, 2, i+1)
```

```
    plt.plot(np.arange(0, 100 + dx, dx), T2_i[i], label = 'dt=0.05, t=%s' % Rr[i])
    plt.grid(True)
    plt.xlabel('x', size = 15)
    plt.ylabel('T', size = 15)
    plt.legend()
plt.savefig("2_i")

fig = plt.subplots(3, 2, figsize=(10, 15))
for i in range(len(R)):
    plt.subplot(3, 2, i+1)
    plt.plot(np.arange(0, 100 + dx, dx), T2_e[i], label = 'dt=0.005, t=%s' % Rr[i])
    plt.grid(True)
    plt.xlabel('x', size = 15)
    plt.ylabel('T', size = 15)
    plt.legend(loc = 'upper right')
plt.savefig("2_e")

#
x = np.arange(0, 0.5, 0.001)
D = [xi ** 2 * np.pi / dx for xi in x]

de = lambda x, t: np.real(-np.log(complex(1 - 4 * t * (np.sin(np.pi * x / 2)) ** 2 / (dx) ** 2)) / t)
di = lambda x, t: np.real(np.log(complex((1 + 2 * t * (np.sin(np.pi * x / 2)) ** 2 / (dx) ** 2) / abs(1 - 2 * t * (np.sin(np.pi * x / 2)) ** 2 / (dx) ** 2))) / t)

De1 = [de(xi, dT1) for xi in x]
De2 = [de(xi, dT2) for xi in x]
Di1 = [di(xi, dT1) for xi in x]
Di2 = [di(xi, dT2) for xi in x]
labels = ['                                        ', 'dt=0.01', 'dt=0.005']

```

```
plt.figure(figsize=(14,6))
for y, label in zip([D, De1, De2], labels):
    plt.plot(x, y, label = label)
plt.grid(True)
plt.legend()
plt.xlabel('k/kn', size = 15)
plt.ylabel('delta', size = 15)
plt.savefig("dt1")

plt.figure(figsize=(14,6))
for y, label in zip([D, Di1, Di2], labels):
    plt.plot(x, y, label = label)
plt.grid(True)
plt.xlabel('k/kn', size = 15)
plt.ylabel('delta', size = 15)
plt.legend()
plt.savefig("dt2")
```

6 Результаты

- dt = 0.01: Для этого шага явная схема является неустойчивой. В силу ограниченности вычислительных ресурсов и быстрого роста функции со временем, графики $T(x)$ удалось построить лишь в моменты времени до 5 (представлено ниже). Неявная схема является безусловно устойчивой, с ней всё хорошо.

- dt = 0.005: Этот шаг расположен на границе устойчивости явной схемы, для него удалось построить все указанные в задании времена. С неявной схемой по прежнему всё хорошо.

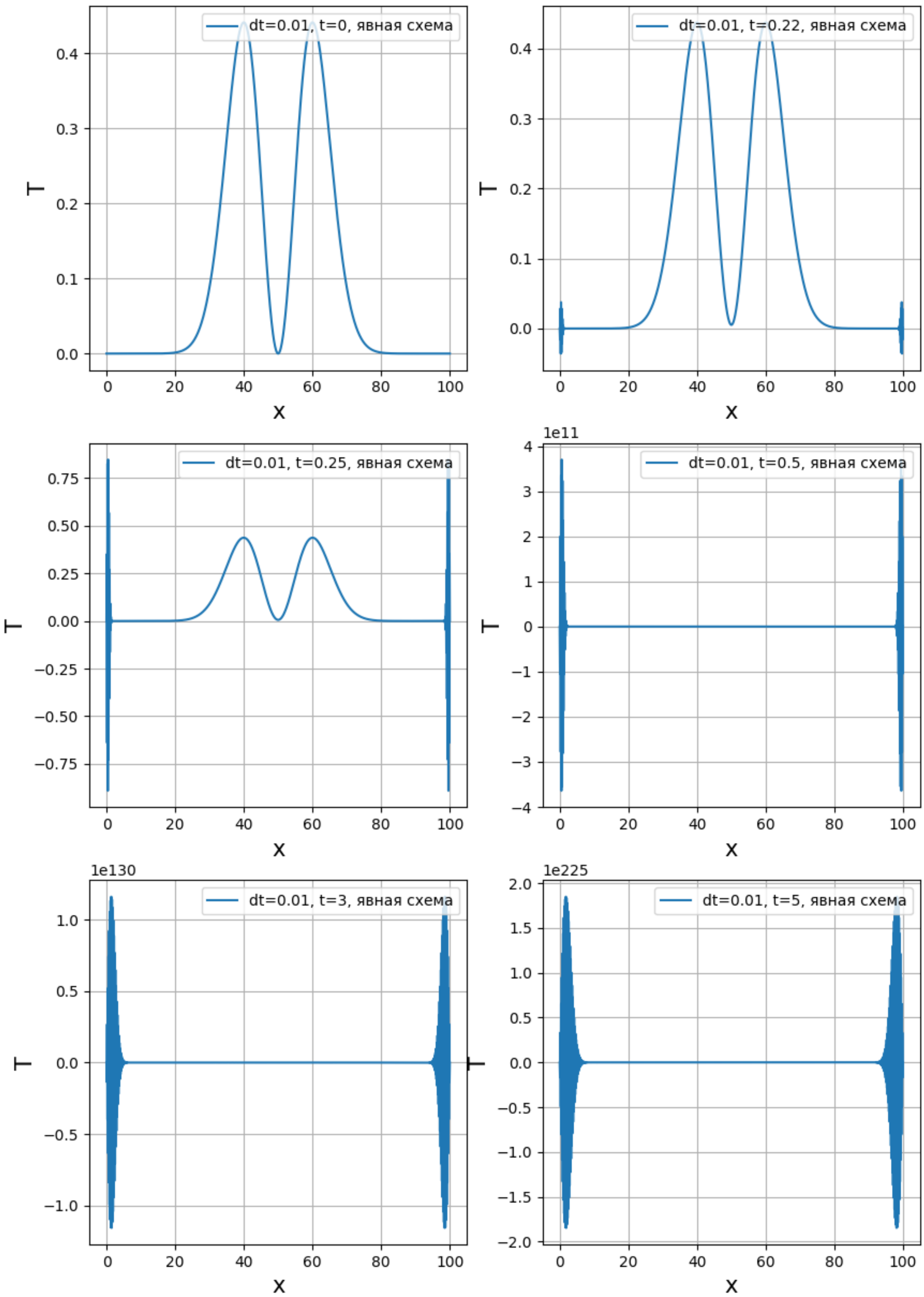


Рис. 1: Явная схема, шаг dt=0.01

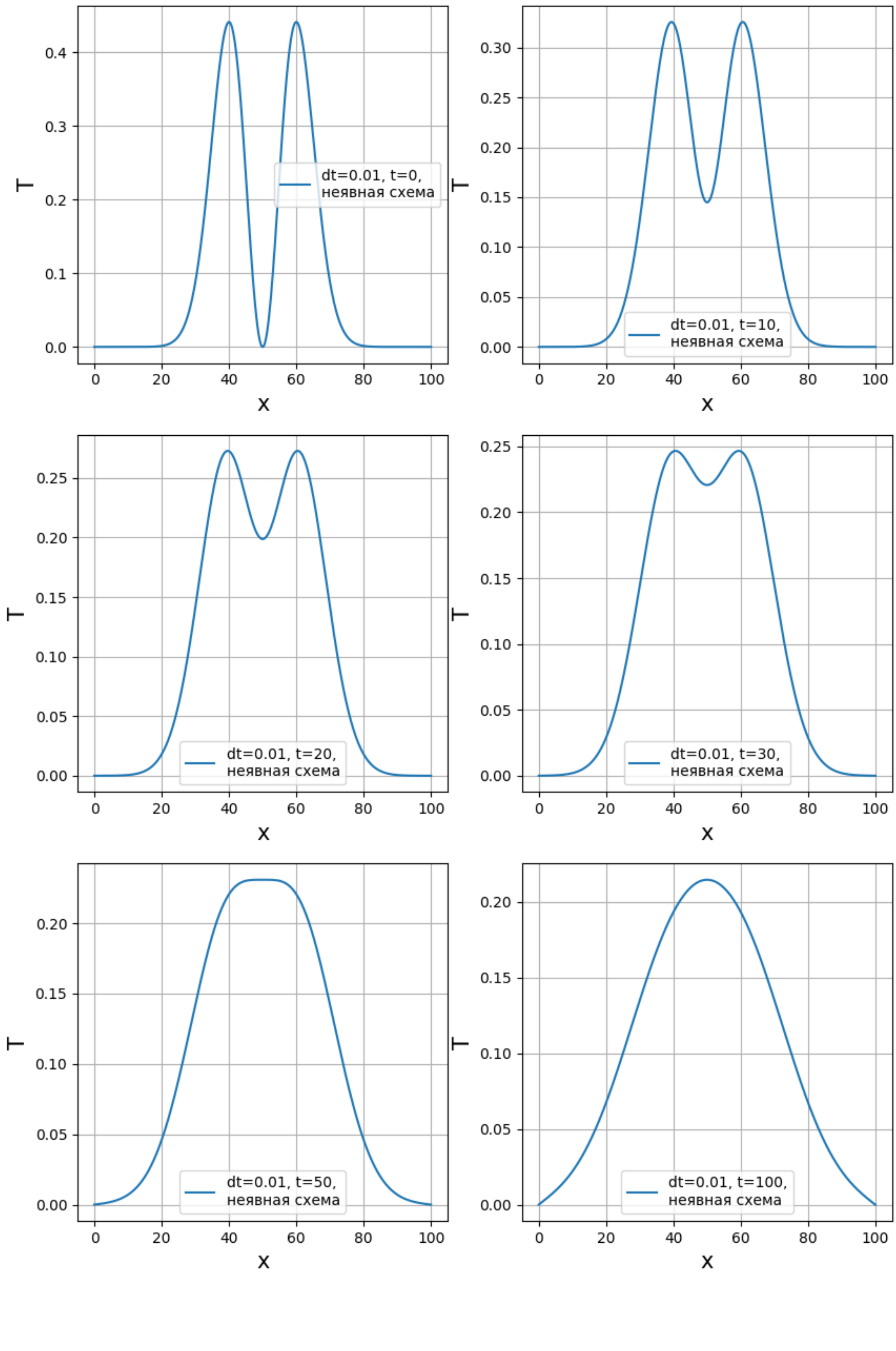


Рис. 2: Неявная схема, шаг dt=0.01

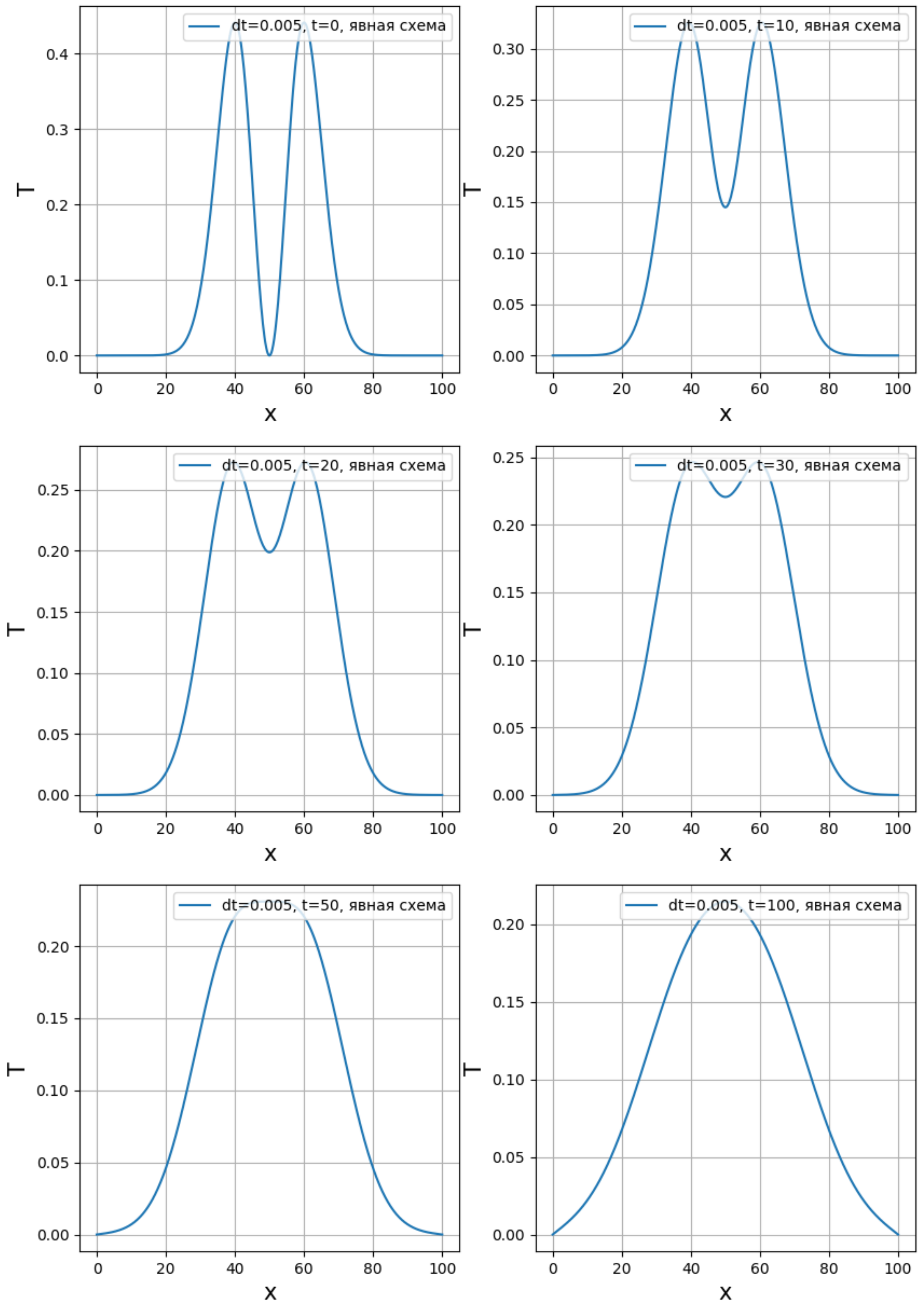


Рис. 3: Явная схема, шаг dt=0.005

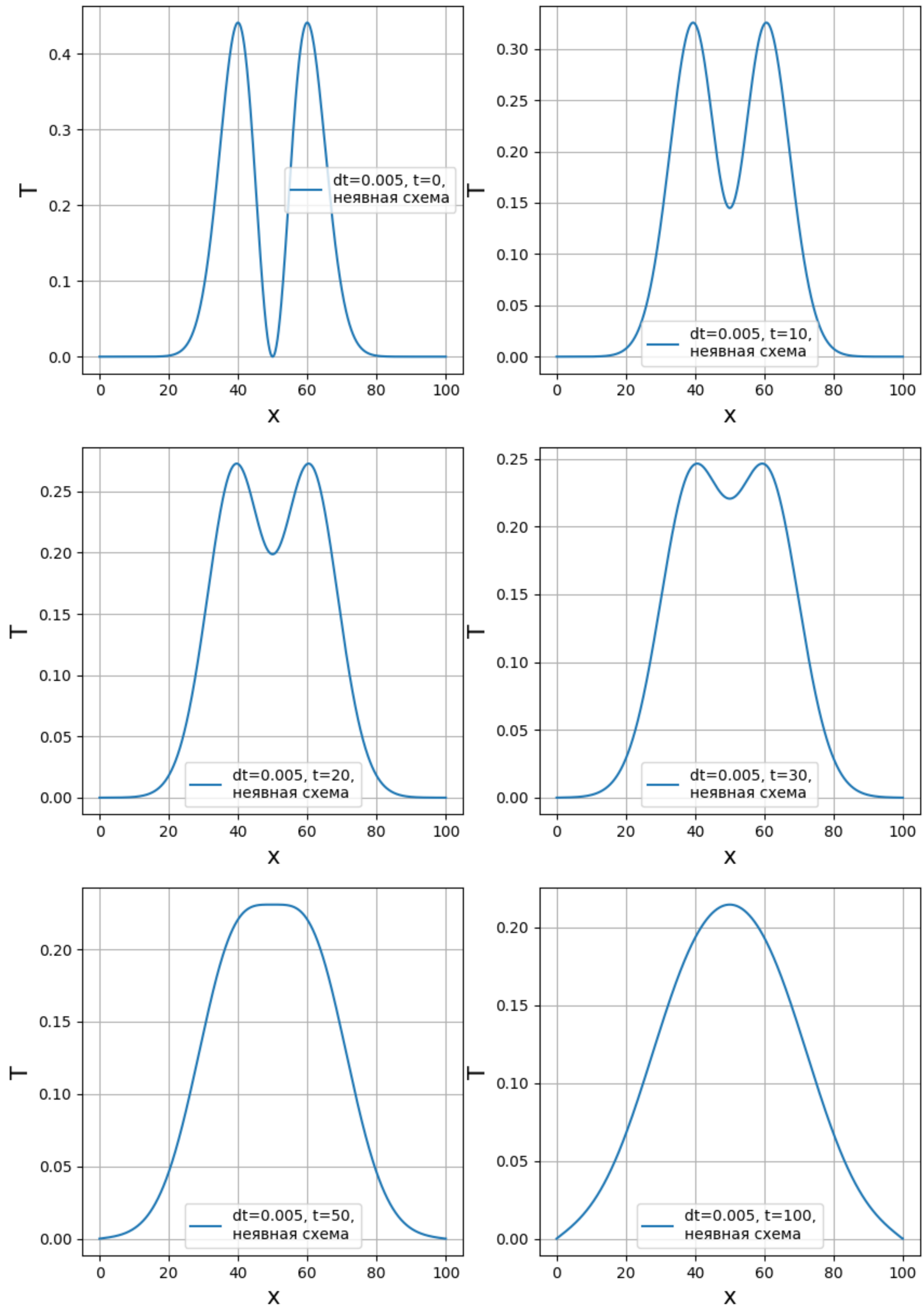


Рис. 4: Неявная схема, шаг dt=0.005