Национальный исследовательский университет
"Московский энергетический институт"

Отчет по курсовой работе
по дисциплине: «Численные методы»

Тема курсовой работы

«Начально-краевая задача для одномерного уравнения колебаний»

Группа: А -05 -18
Студентка: Поиленкова Анна
Преподаватель: Амосова О.А.

Москва 2020

Теоретическая часть

Постановка задачи

Вторая краевая задача для уравнения колебаний:

$$\begin{cases} u_{tt} = a^2 u_{xx} + f(x,t) & 0 < x < l, 0 < t < T \\ u(x,0) = \Phi(x), u_t(x,0) = \varphi(x) & 0 \le x \le l \\ u_x(0,t) = \mu_1(t), u_x(l,t) = \mu_2(t) & 0 \le t \le T \end{cases}$$

1.10. Задача **(1.2)** **.** Явная схема. Моделирование колебаний в зависимости от внешнего сосредоточенного импульса.

$f(x,t) = g(x,\xi,\varepsilon)$, $g(x,\xi,\varepsilon) = \begin{cases} 2\varepsilon, & x \in [\xi - \varepsilon, \xi + \varepsilon] \\ 0, & x \not\subset [\xi - \varepsilon, \xi + \varepsilon] \end{cases}$, $\varepsilon = 0.1l,\ 0.01l$, $\xi = l/4,\ l/2,\ l/3$.

$\phi(x) = 0,\ \ \varphi(x) = 0$

Тестовые примеры:

$$\begin{cases} u_{tt} = u_{xx} + 10e^{-3t}\cos x, \\ 0 < x < \pi,\ \ 0 < t < T, \\ u(x,0) = 0, u_t(x,0) = 0,\ \ 0 \le x \le \pi, \\ u_x(0,t) = 0,\ \ u_x(\pi,t)\ \ 0,\ \ 0 \le t \le T. \end{cases}$$

$$\begin{cases} u_{tt} = u_{xx} + 8\cos 2t \cos 4x, \\ 0 < x < \pi,\ \ 0 < t < T \\ u(x,0) = 0,\ \ u_t(x,0) = 0,\ \ 0 \le x \le \pi, \\ u_x(0,t) = 0,\ \ u_x(\pi,t) = 0,\ \ 0 \le t \le T. \end{cases}$$

Литература [2], Часть 3 , Глава 1, §5 .

Аналитическое решение

Нахождение аналитического решения задачи для тестового примера методом собственных функций

1. Рассмотрим вторую краевую задача для уравнения колебаний:

$$\begin{cases} u_{tt} = a^2 u_{xx} + f(x,t) & 0 < x < l, 0 < t < T \\ u(x,0) = \Phi(x), u_t(x,0) = \varphi(x) & 0 \le x \le l \\ u_x(0,t) = \mu_1(t), u_x(l,t) = \mu_2(t) & 0 \le t \le T \end{cases} \qquad (1)$$

1. Поставим основную вспомогательную задачу:

возьмем однородные краевые условия $\begin{cases} u_x(0,t) = 0 \\ u_x(l,t) = 0 \end{cases}$ (2)

и правую часть $f(x,t) = 0$ => $u_{tt} = a^2 u_{xx}$

2. Будем искать решение методом Фурье, то есть представим решение в виде произведения двух функций: $u(x,t) = X(x)T(t)$, одна из которых зависит только от x, а другая от t.

3. Тогда подставив в уравнение $u_{tt} = a^2 u_{xx}$ наше решение $u(x,t) = X(x)T(t)$ получим:

$$X(x)T''(t) = a^2 T(t)X(x)''$$

4. Поделим обе части уравнения на $a^2 T(t)X(x)$

$$\frac{T(t)''}{a^2 T(t)} = \frac{X(x)''}{X(x)}$$

5. Получили равенство, в котором левая часть зависит только от t, а правая от x. Функции разных переменных могут быть равны между собой только тогда, когда они равны константе. Обозначается эта константа $-\lambda$.

$$\frac{T(t)''}{a^2 T(t)} = \frac{X(x)''}{X(x)} = -\lambda$$

6. После преобразований получим два уравнения:

$$X(x)'' + \lambda X(x) = 0$$

$$T(t)'' + a^2 \lambda T(t) = 0$$

Задача Штурма-Лиувилля.

7. Рассмотрим уравнение $X(x)'' + \lambda X(x) = 0$.

Граничные условия (2) дают

$$u_x(0,t) = X'(0)T(t) = 0$$

$$u_x(l,t) = X'(l)T(t) = 0$$

Следовательно $X(x)$ удовлетворяет условиям $\quad X'(0) = X'(l) = 0$

Переходим к задаче о собственный значениях:

Найти те значения λ, при которых существует нетривиальные условия задачи

$$\begin{cases} X(x)'' + \lambda X(x) = 0 \\ X'(0) = 0, X'(l) = 0 \end{cases}$$

а также найти эти решения .

Такая задача называется задача Штурма-Лиувилля.

- $\lambda < 0$ Нет тривиальных решений
- $\lambda = 0$ Нет тривиальных решений

- <u>λ > 0</u>

Общее решение уравнения

$$X(x) = D_1 cos\sqrt{\lambda}x + D_2 sin\sqrt{\lambda}x$$

$$X'(x) = -\sqrt{\lambda}D_1 sin\sqrt{\lambda}x + \sqrt{\lambda}D_2 cos\sqrt{\lambda}x$$

Граничные условия дают

$X'(0) = -\sqrt{\lambda}D_1 sin 0 + \sqrt{\lambda}D_2 cos 0 = \sqrt{\lambda}D_2 = 0$ => $D_2 = 0$

$X'(x) = -\sqrt{\lambda}D_1 sin\sqrt{\lambda}l + \sqrt{\lambda}D_2 cos\sqrt{\lambda}l = -\sqrt{\lambda}D_1 sin\sqrt{\lambda}l = 0$

Если $X(x)$ не равно тождественно нулю, то $D_1 \neq 0$, а λ > 0, поэтому $-sin\sqrt{\lambda}l = 0$

$\sqrt{\lambda} = \frac{\pi n}{l}$, где n - ∀ целое число. Следовательно, нетривиальные решения задачи возможны лишь при значениях. $\lambda_n = (\frac{\pi n}{l})^2$

$$X_n(x) = D_n cos\frac{\pi n}{l}x$$, где D - произвольная постоянная

$$X_n(x) = cos\frac{\pi n}{l}x$$

8. Уравнение $T(t)'' + a^2T(t) = 0$ решим в общем виде, подставив найденное λ_n

$T_n(t) = A_n cos\frac{\pi n}{l}at + B_n sin\frac{\pi n}{l}at$

A_n и B_n - произвольные постоянный

9. Вернемся к задаче (1)

$$U_n(x,t) = X_n(x)T_n(t) = (A_n cos\frac{\pi n}{l}at + B_n sin\frac{\pi n}{l}at)cos\frac{\pi n}{l}x$$

Являются частными решениями уравнения (1), удовлетворяющими граничным условиям.

В силу линейности и однородности уравнения (1) сумма частных решений

$$U(x,t) = \sum_{i=1}^{\infty} U_n(x,t) = \sum_{i=1}^{\infty} (A_n cos\frac{\pi n}{l}at + B_n sin\frac{\pi n}{l}at)cos\frac{\pi n}{l}x$$

10. Найдем A_n и B_n , используя условия $u(x,0) = \Phi(x), u_t(x,0) = \varphi(x)$

$$\sum_{i=1}^{\infty}\frac{\partial U_n}{\partial t}(x,t)=\sum_{i=1}^{\infty}(-A_n\frac{\pi n}{l}a*sin\frac{\pi n}{l}at+B_n\frac{\pi n}{l}a*cos\frac{\pi n}{l}at)cos\frac{\pi n}{l}x$$

$$\begin{cases} u(x,0)=\Phi(x)=\sum_{i=1}^{\infty}U_n(x,0)=\sum_{i=1}^{\infty}(A_n cos0+B_n sin0)cos\frac{\pi n}{l}x=\sum_{i=1}^{\infty}(A_n)cos\frac{\pi n}{l}x \\ u_t(x,0)=\varphi(x)=\sum_{i=1}^{\infty}\frac{\partial U_n}{\partial t}(x,0)=\sum_{i=1}^{\infty}(-A_n\frac{\pi n}{l}a*sin0+B_n\frac{\pi n}{l}a*cos0)cos\frac{\pi n}{l}x=\sum_{i=1}^{\infty}(B_n\frac{\pi n}{l}a)cos\frac{\pi n}{l}x \end{cases}$$

11. Используем ряды Фурье

(у нас четная функция)

$$F(x)=\frac{a_0}{2}+\sum_{n=1}^{\infty}\left(a_n\cos\frac{\pi n}{l}x+b_n\sin\frac{\pi n}{l}x\right),$$

$$a_n=\frac{1}{l}\int_{-l}^{+l}F(\xi)\cos\frac{\pi n}{l}\xi\,d\xi,\qquad b_n=\frac{1}{l}\int_{-l}^{+l}F(\xi)\sin\frac{\pi n}{l}\xi\,d\xi.$$

Сравнивая формулы видно, что $A_n=\frac{2}{l}\int_0^l\Phi(x)cos\frac{\pi n}{l}xdx$ и $B_n=\frac{2}{\pi na}\int_0^l\varphi(x)cos\frac{\pi n}{l}xdx$

Получаем,

$$U(x,t)=\sum_{i=1}^{\infty}((\frac{2}{l}\int_0^l\Phi(x)cos\frac{\pi n}{l}xdx)cos\frac{\pi n}{l}at+(\frac{2}{\pi na}\int_0^l\varphi(x)cos\frac{\pi n}{l}xdx)sin\frac{\pi n}{l}at)cos\frac{\pi n}{l}x$$

Тестовый пример 1

$$\begin{cases} u_{tt} = a^2 u_{xx} + 10e^{-3t} cosx & 0 < x < \pi, 0 < t < T \\ u(x,0) = 0, u_t(x,0) = 0 & 0 \le x \le \pi \\ u_x(0,t) = 0, u_x(\pi, t) = 0 & 0 \le t \le T \end{cases} \quad (1.1)$$

=> $l = \pi \qquad a = 1$

Решение $U(x,t)$ будем искать в виде

$U(x,t) = \sum_{i=1}^{\infty} T_k(t) cos(kx)$,где $T_k(t)$ - неизвестные функции (1.2)

Дифференцируем и поставляем в (1.1)

$$\sum_{i=1}^{\infty} [T_k''(t) + (\frac{\pi k}{l})^2 T_k(t)] cos(kx) = f(x,t)$$

При произвольно фиксированном t представляем собой разложение функции f(x,t) в ряд Фурье по косинусам. Поэтому для коэффициентов разложения имеем

$$T_k''(t) + k^2 T_k(t) = f_k(t) \quad k = 1,2,3.. \qquad (1.3)$$

$$f_k(t) = \frac{1}{\pi} \int_0^{\pi} f(x,t) cos(kx) dx = \frac{1}{\pi} \int_0^{\pi} 10e^{-3t} cos(x) * cos(kx) dx$$

$= 10e^{-3t}$ при $k = 1$

$= 0$ при $k \ne 1$

$U_t(x,t) = \sum_{i=1}^{\infty} T_k'(t) cos(kx)$ //из (1.2)

С учетом, что

$u(x,0) = 0$ и $u_t(x,0) = 0$

$\sum_{i=1}^{\infty} T_k(0) cos(kx) = 0$ и $\sum_{i=1}^{\infty} T_k'(0) cos(kx) = 0$

$T_k(0) = 0$ и $T_k'(t) = 0 \quad k = 1,2,3..$ (1.4)

Следовательно, для определения функций $T_k(t)$ имеем задачу (1.3) и (1.4), представляющую задачу Коши для обыкновенного дифференциального уравнения второго порядка с постоянными коэффициентами и начальными условиями. Ее решение может быть получено с помощью метода вариации произвольных постоянных в следующем виде

$$T_k''(t) + k^2T_k(t) = \frac{1}{\pi}\int_0^\pi 10e^{-3t}cos(x) * cos(kx)dx$$

(при $k = 1$)

$T_k''(t) + T_k(t) = 10e^{-3t}$

+ $T_k(0) = 0$ и $T_k'(t) = 0$

$T_k(t) = e^{-3t} + 3sin(t) - cos(t)$

(При $k \neq 1$)

$T_k''(t) + k^2T_k(t) = 0$

+ $T_k(0) = 0$ и $T_k'(t) = 0$ $k = 1,2,3..$

нет решений

Подставляем в (1.1) и получаем ответ

$U(x,t) = \sum_{i=1}^{\infty} [e^{-3t} + 3sin(t) - 6cos(t)]cos(kx)$

Тут будет только одно слагаемое (при к=1), а все остальные будут равны 0

=> $U(x,t) = [e^{-3t} + 3sin(t) - cos(t)]cos(x)$

Тестовый пример 2

$$\begin{cases} u_{tt} = a^2 u_{xx} + 8cos2tcos4x & 0 < x < \pi, 0 < t < T \\ u(x,0) = 0, u_t(x,0) = 0 & 0 \le x \le \pi \\ u_x(0,t) = 0, u_x(\pi, t) = 0 & 0 \le t \le T \end{cases} \qquad (2.1)$$

=> $l = \pi \qquad a = 1$

Проводим аналогичные размышления, что и при решении тестового примера 1 и начинаем с пункта (1.3)

$$T_k''(t) + k^2 T_k(t) = f_k(t) \quad k = 1,2,3.. \qquad (1.3)$$

$$f_k(t) = \frac{1}{\pi}\int_0^{\pi} f(x,t)cos(kx)dx = \frac{1}{\pi}\int_0^{\pi} 8cos(2t)cos(4x) * cos(kx)dx$$

$= 8cos(2t)$ при $k = 4$

$= 0$ при $k \neq 4$

(Опять пропускаем аналогичные рассуждения)

(при $k = 4$)

$T_k''(t) + 16T_k(t) = 8cos(2t)$

\+ $T_k(0) = 0$ и $T_k'(t) = 0$

$$T_k(t) = \frac{4\left(2\cos(2t) + 1\right)}{3}\sin^2(t)$$

(При $k \neq 4$)

$T_k''(t) + k^2 T_k(t) = 0$

\+ $T_k(0) = 0$ и $T_k'(t) = 0$ $k = 1,2,3..$ нет решений

Подставляем в (2.1) и получаем ответ

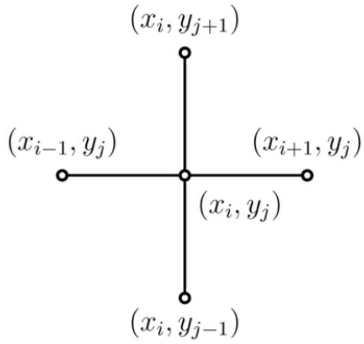
$$U(x,t) = \sum_{i=1}^{\infty}[\frac{4\left(2\cos(2t) + 1\right)}{3}\sin^2(t)]cos(kx)$$

Тут будет только одно слагаемое (при к=4), а все остальные будут равны 0

=> $U(x,t) = [\frac{4\left(2\cos(2t) + 1\right)}{3}\sin^2(t)]cos(4x)$

Численное решение

$$\begin{cases} u_{tt} = a^2 u_{xx} + 10e^{-3t}cosx & 0 < x < \pi, 0 < t < T \\ u(x,0) = 0, u_t(x,0) = 0 & 0 \le x \le \pi \\ u_x(0,t) = 0, u_x(\pi,t) = 0 & 0 \le t \le T \end{cases}$$



Введем сетку по x и по t с шагом h

$\varpi_h = (x_i = ih, i = 0,1,2..N_1)$

$\varpi_h = (t_j = jt, j = 0,1,2..N_2)$

Решение задачи будем искать на сетке $\Omega = \varpi_h \times \varpi_\tau$

Для аппроксимации возьмем пятиточечный шаблон, где сеточная функция аппроксимирует функцию $f(x,y)$ в правой части уравнения. В случае, если эта функция непрерывна, то можно положить $F_{i,j} = f(x_i, y_j)$

Для аппроксимации начальных условий выражение $\dfrac{U_i^1 - U_i^0}{t} = 0$ не подойдет, потому что имеет второй порядок точности, будем использовать:

$$\frac{u(x,\tau) - u(x,0)}{\tau} = \frac{\partial u(x,0)}{\partial t} + \frac{\tau}{2}\frac{\partial^2 u(x,0)}{\partial t^2} + O(\tau^2)$$

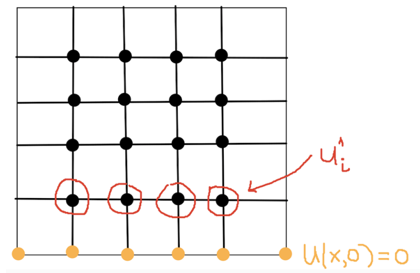
$$\frac{\partial u(x,0)}{\partial t} = \frac{u(x,\tau) - u(x,0)}{\tau} - \frac{\tau}{2}u_0''(x) + O(\tau^2),$$

$\dfrac{\partial u(x,0)}{\partial t}$ $\qquad u_{tt} = a^2 u_{xx} + f(x,t)$

$$\frac{\partial u(x,0)}{\partial t} = \frac{U_i^1 - U_i^0}{t} - \frac{t}{2}\left(a^2\frac{U_{i+1}^0 - 2U_i^0 + U_{i-1}^0}{h^2} + f(x,t)\right) = 0$$ (выражаем U_i^1)

$=> U_i^1 = \dfrac{t^2}{2}f(x,t)$ (применим $U_i^0 = 0$) (для i = 1..N-1)

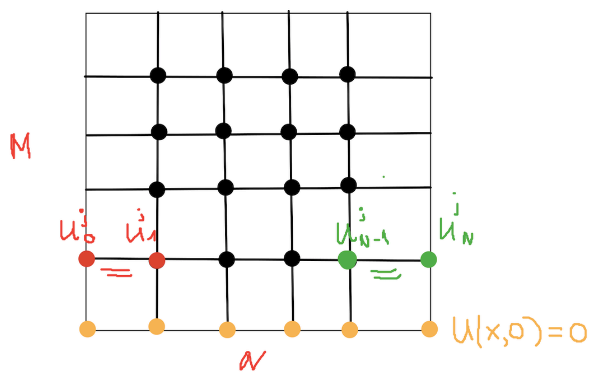
$$\frac{U_i^1 - U_i^0}{t} - \frac{t}{2}\left(a^2\frac{U_{i+1}^0 - 2U_i^0 + U_{i-1}^0}{h^2} + f(x,t)\right) = 0$$



$\dfrac{\partial u(0,t)}{\partial h}$ $\qquad \dfrac{\partial u(pi,t)}{\partial h}$

$$\frac{U_1^j - U_0^j}{h} = 0, \frac{U_{N_1}^j - U_{N_1-1}^j}{h} = 0$$

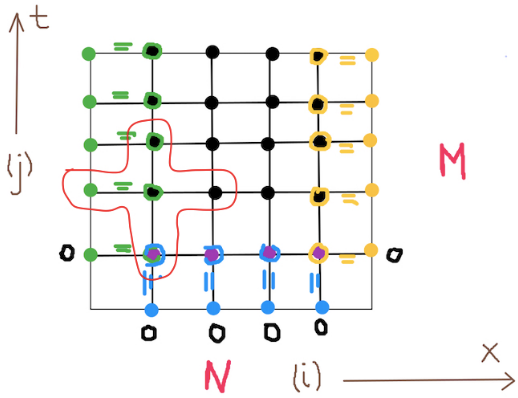
$$=> U_1^j = U_0^j, U_{N_1}^j = U_{N_1-1}^j$$



$$\begin{cases} \dfrac{U_i^{j+1} - 2U_i^j + U_i^{j-1}}{t^2} - a^2\dfrac{U_{i+1}^j - 2U_i^j + U_{i-1}^j}{h^2} = f(x_i, t_j) \\ U_i^0 = 0, U_i^1 = \frac{t^2}{2} f(x,t)(i = 1, N-1) \\ U_1^j = U_0^j,\ U_{N_1}^j = U_{N_1-1}^j \end{cases}$$

$$\frac{U_i^{j+1}}{t^2} - \frac{2U_i^j}{t^2} + a^2\frac{2U_i^j}{h^2} + \frac{U_i^{j-1}}{t^2} - a^2\frac{U_{i+1}^j}{h^2} - a^2\frac{U_{i-1}^j}{h^2} = f(x_i, t_j)$$

$U_{N_1}^j = U_{N_1-1}^j$ и $U_1^j = U_0^j$ =>



Распишем отмеченную точку:

$$\frac{U_1^3}{t^2} - \frac{2U_1^2}{t^2} + a^2\frac{2U_1^2}{h^2} + \frac{U_1^1}{t^2} - a^2\frac{U_2^2}{h^2} - a^2\frac{U_0^2}{h^2} = f(x_i, t_j)$$

$$\frac{U_1^3}{t^2} - \frac{2U_1^2}{t^2} + a^2\frac{2U_1^2}{h^2} - a^2\frac{U_1^2}{h^2} + \frac{U_1^1}{t^2} - a^2\frac{U_2^2}{h^2} = f(x_i, t_j)$$

Но нам не известна точка U_1^3

$$U_1^3 = 2U_1^2 - a^2\frac{2U_1^2t^2}{h^2} + a^2\frac{U_1^2t^2}{h^2} - U_1^1 + a^2\frac{U_2^2t^2}{h^2} + f(x_i, t_j)t^2$$

Общий вид:

$U_i^{j+1} = 2U_i^j - a^2\dfrac{2U_i^jt^2}{h^2} - U_i^{j-1} + a^2\dfrac{U_{i+1}^jt^2}{h^2} + a^2\dfrac{U_{i-1}^jt^2}{h^2} + f(x_i, t_j)t^2$, но

для $U_{N_1-1}^j$ и U_1^j , получил другое (обоснование выше в расписанной точке)

- для U_1^j из-за того, что $U_1^j = U_0^j$

  (при i = 1)

  $$U_1^{j+1} = 2U_1^j - a^2\frac{2U_1^jt^2}{h^2} - U_1^{j-1} + a^2\frac{U_2^jt^2}{h^2} + a^2\frac{U_1^jt^2}{h^2} + f(x_1, t_j)t^2$$

- для $U_{N_1-1}^j$ из-за того, что $U_{N_1}^j = U_{N_1-1}^j$

  (при i = N-1)

  $$U_i^{j+1} = 2U_i^j - a^2\frac{2U_i^jt^2}{h^2} - U_i^{j-1} + a^2\frac{U_i^jt^2}{h^2} + a^2\frac{U_{i-1}^jt^2}{h^2} + f(x_i, t_j)t^2$$

Тогда разностная схема примет следующий вид:

$U_i^1 = \frac{t^2}{2} f(x,t)$ для j = 0, i = 1..N-1

$U_i^{j+1} = 2U_i^j - a^2 \frac{2U_i^j t^2}{h^2} - U_i^{j-1} + a^2 \frac{U_{i+1}^j t^2}{h^2} + a^2 \frac{U_{i-1}^j t^2}{h^2} + f(x_i, t_j) t^2$ для j = 1..M-1, i = 2..N-2

$U_1^{j+1} = 2U_1^j - a^2 \frac{2U_1^j t^2}{h^2} - U_1^{j-1} + a^2 \frac{U_2^j t^2}{h^2} + a^2 \frac{U_1^j t^2}{h^2} + f(x_1, t_j) t^2$ для j = 1..M-1, i = 2

$U_0^j = U_1^j$ для j = 1..M-1

$U_i^{j+1} = 2U_i^j - a^2 \frac{2U_i^j t^2}{h^2} - U_i^{j-1} + a^2 \frac{U_i^j t^2}{h^2} + a^2 \frac{U_{i-1}^j t^2}{h^2} + f(x_i, t_j) t^2$ для j = 1..M-1, i = N-1

$U_N^j = U_{N-1}^j$ для j = 1..M-1

Численное решение примера 1

Результаты вычислений на нескольких временных слоях приведены в следующей таблице:

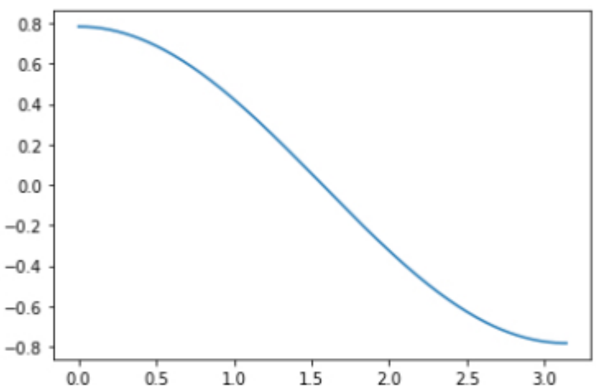
(для удобства в таблицу внесены значения не во всех точках полученного массива):

Левый столбец - ответ, полученный разностной схемой

Правый - точное решение

	t = 1		**t = 5**	
u(0,t)	0.0	0.0001	0.003041	0.003048
u(0.5,t)	0.000106	0.000107	0.002639	0.002644
u(1.04,t)	6.227942e-05	6.29056e-05	0.00153	0.001541
u(1.56,t)	1.2319e-06	1.24438e-06	3.043419	3.04965
u(2.08,t)	-6.014145e-05	-6.074613e-05	3.043419	-0.001488
u(2.6,t)	-0.000105	-0.000106	-0.002608	-0.0026
u(3.12,t)	-0.00012	-0.000124	-0.00304	-0.00304

Графики в один момент времени:



получен с помощью разностной схемы

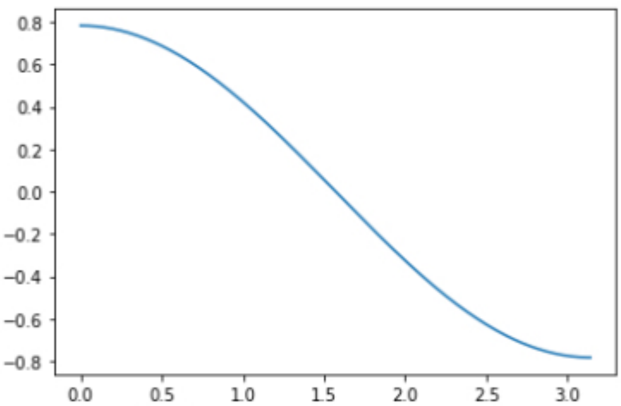
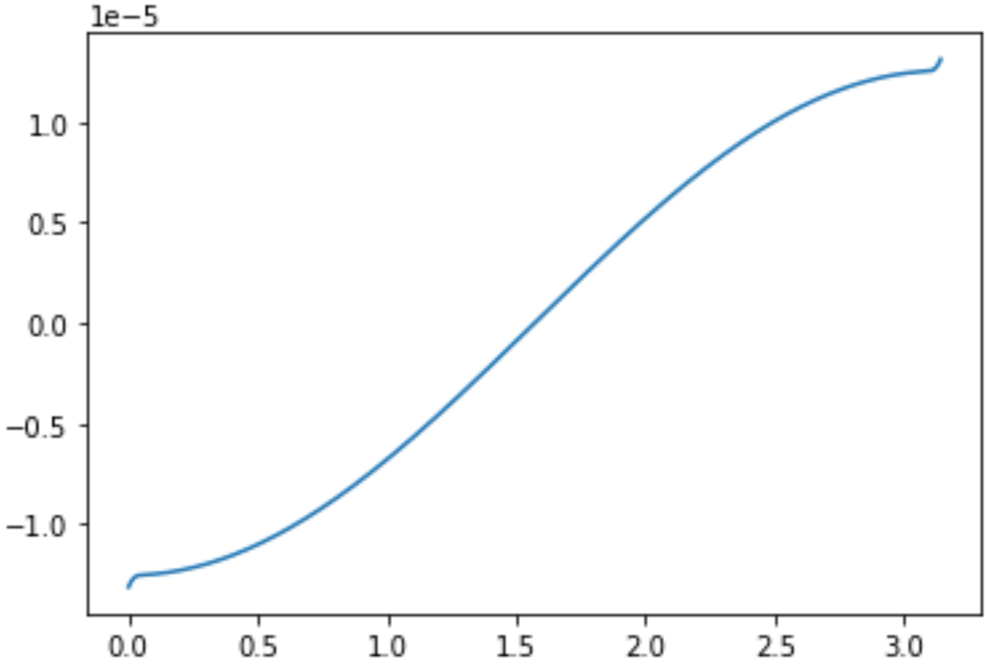


График точного решения

Построим график погрешности, как разность точного решения и решения, найденного численным методом



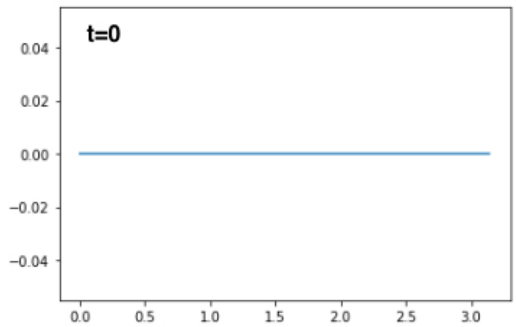
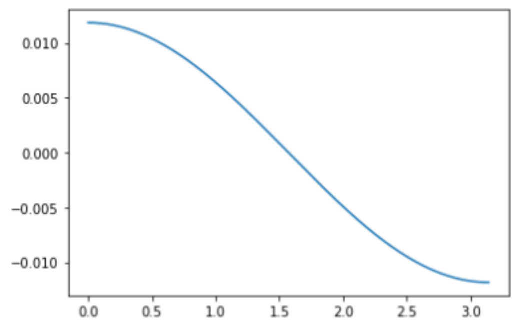


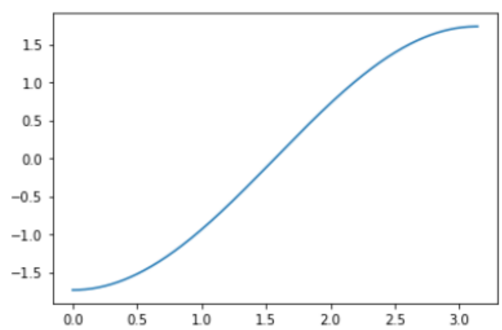
График в разные моменты времени:



Вывод:

Графики точного решения и решения, построенного разностной схемой в один и тот же момент времени совпадают

Погрешность с течением времени будет увеличивать из-за того, что мы находим узлы схемы последовательно, начиная с времени t = 0



Численное решение примера 2

Результаты вычислений на нескольких временных слоях приведены в следующей таблице:

(для удобства в таблицу внесены значения не во всех точках полученного массива):

Левый столбец - ответ, полученный разностной схемой

Правый - точное решение

	t = 1		t = 3	
u(0,t)	0.000039	0.00015	0.0000899	0.0003594
u(0.5,t)	−0.000019	−0.000078	−0.000043	−0.000175
u(1.04,t)	−0.000020	−0.000083	−0.000047	−0.000187
u(1.56,t)	0.000039	0.000159	0.0000898	0.0003591
u(2.08,t)	−0.000018	−0.00007	−0.000040	−0.000162
u(2.6,t)	−0.000022	−0.00008	−0.00005	−0.000200
u(3.12,t)	0.000039	0.000159	0.000089	0.00035830

Построим график погрешности, как разность точного решения и решения, найденного численным методом

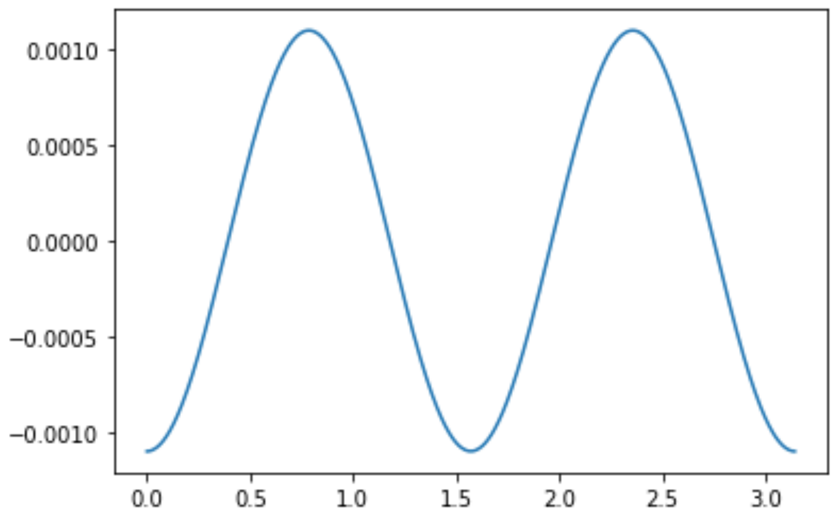
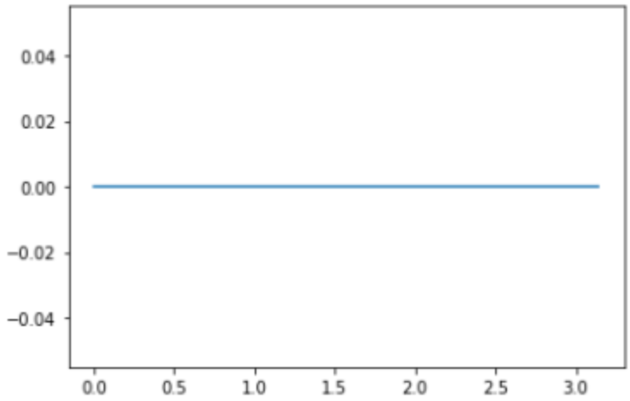
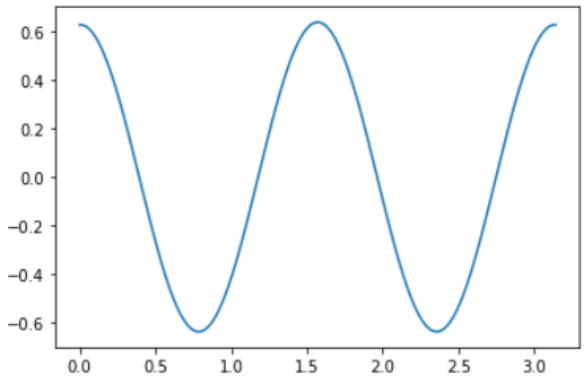
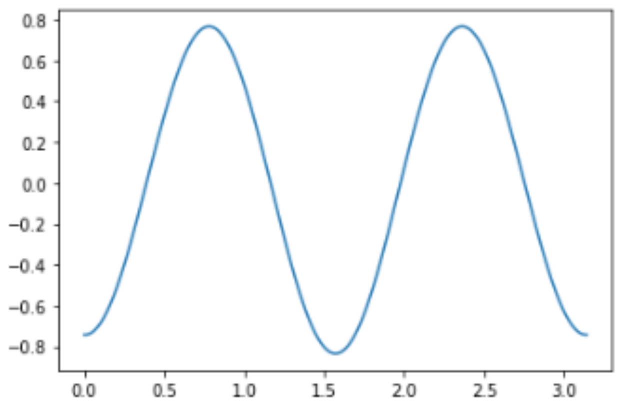


График в разные моменты времени:







Вывод:
Графики точного решения и решения, построенного разностной схемой в один и тот же момент времени совпадают

Моделирование колебаний в зависимости от внешнего сосредоточенного импульса

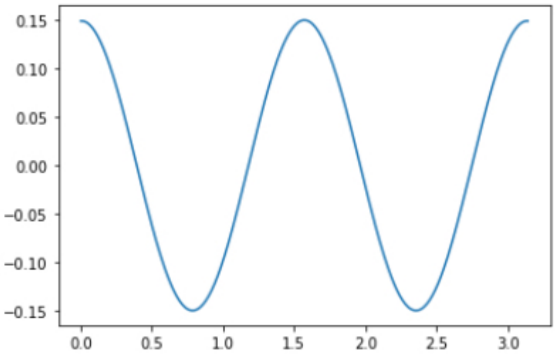
1.10. Задача **(1.2)** **.** Явная схема. Моделирование колебаний в зависимости от внешнего сосредоточенного импульса.

$$f(x,t) = g(x,\xi,\varepsilon), \quad g(x,\xi,\varepsilon) = \begin{cases} 2\varepsilon, & x \in [\xi - \varepsilon, \xi + \varepsilon] \\ 0, & x \not\subset [\xi - \varepsilon, \xi + \varepsilon] \end{cases}, \quad \varepsilon = 0.1l,\ 0.01l,\ \xi = l/4,\ l/2,\ l/3.$$

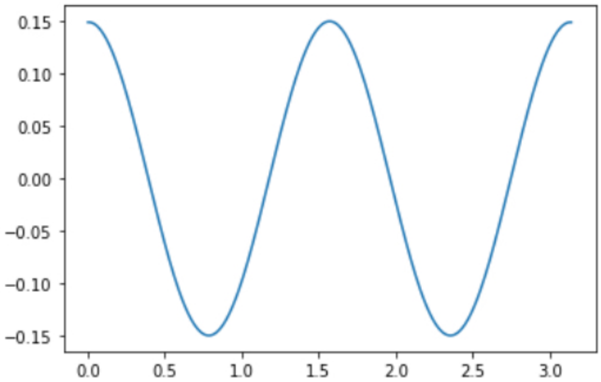
$$\phi(x) = 0, \quad \varphi(x) = 0$$

Имеем модель в которую можем передавать входные параметры. (передаем разные вариации)

Примеры некоторых графиком:



e = 0.1l g = l/3



e = 0.01l g = l/4

e = 0.01l g = l/3 (взяли время в начале колебания)

Заключение:

В курсовой работе рассмотрена разностная схема по построению второй краевой задачи для уравнения колебаний. Были решены и построены два тестовых примера. Отладка происходила при сравнении точного решения и решения, полученного разностной схемой.

В итоге мы добились того, что программа помогает моделировать колебания в зависимости от внешнего сосредоточенного импульса

Приложение

Код программы

```
import numpy as np
import matplotlib.pyplot as plt
import math
```

```
# граничные условия для всех данных
def U0_x(x):
    return 0
```

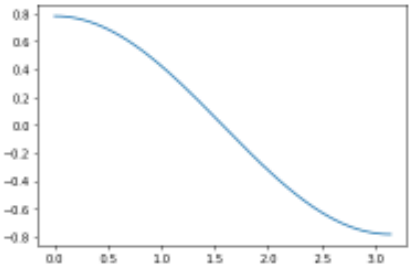
```
# разностная схема
def diff_scheme(l, T, h, ty, f, a, U0_x):
    N = int(l/h) + 1
    M = int(T/ty) + 1

    x = np.linspace(0, l, N)
    t = np.linspace(0, T, M)

    U = np.zeros((N, M))

    for i in range(N):    # U(x,0) and U_t(x,0)
        U[i,0] = U0_x(x[i])
        U[i,1] = U0_x(x[i])

    for i in range(1,N-1):
        U[i,1]=ty**2/2*f(x[i],t[1])

    for j in range(1, M-1):
        U[1,j+1] = U[1,j]*2 - U[1,j]*(2*a**2*ty**2)/h**2 + U[1+1,j]*(a**2*ty**2)/h**2 - U
[1,j-1] + (a**2*ty**2)/(h**2)*U[1,j] + f(x[1],t[j])*ty**2
        U[0,j+1] = U[1,j+1]

        for i in range(2, N-2):
            U[i, j+1] = U[i,j]*2 - U[i,j]*(2*a**2*ty**2)/h**2 + U[i+1,j]*(a**2*ty**2)/h**
2 - U[i,j-1] + (a**2*ty**2)/(h**2)*U[i-1,j] + f(x[i],t[j])*ty**2

        U[N-2,j+1] = U[N-2,j]*2 - U[N-2,j]*(2*a**2*ty**2)/h**2 + U[N-2,j]*(a**2*ty**2)/h*
*2 - U[N-2,j-1] + (a**2*ty**2)/(h**2)*U[N-2-1,j] + f(x[N-2],t[j])*ty**2
        U[N-1,j+1] = U[N-2,j+1]
    return x, t, np.transpose(U)
```

```
#функция вывода графиков точного решения
def show_tochn(f,moment,N,M):
    fig = plt.subplots()
    xval = np.linspace(0, math.pi, N)
    yval = np.linspace(0, 10, M)
    x, t = np.meshgrid(xval, yval)
    tochn = u(x, t)

    plt.plot(x[moment], tochn[moment])
    plt.show()
```

```
#функция вывода графиков решения
def show(x,z,moment):
    fig = plt.subplots()
    plt.plot(x, z[moment])
    plt.show()
```

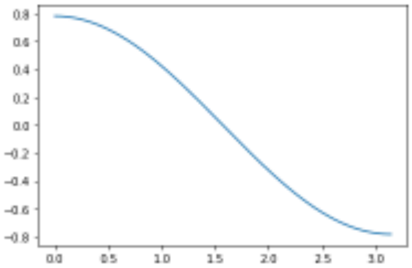
Тестовый пример 1

```
def Func1(x,t):
    return 10*math.e**(-3*t)*np.cos(x)

l = math.pi
a = 1
T = 10
h = 0.01
th = 0.01
x,t,z =diff_scheme(l, T, h, th, Func1, a, U0_x)[0:3]

show(x,z,50) # 3 параметр - момент времени
```



точное решение

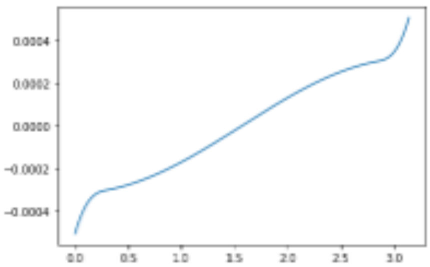
```
u = lambda x, t: (math.e**(-3*t) + 3*np.sin(t) - np.cos(t))*np.cos(x)
N = int(l/h) +1
M = int(T/th) +1

show_tochn(u,50,N,M)
```




```
# вывод решения в виде таблицы
moment = 5
x = diff_scheme(l, T, h, th, Func1, a, U0_x)[0]
xval = np.linspace(0, math.pi, N)
yval = np.linspace(0, 10, M)
x, t = np.meshgrid(xval, yval)
tochn = u(x, t)


for i in range(0,len(x[0]) ,int(len(x[0])/6)):
    print("[",x[moment][i],"] =", z[moment][i], "    ", tochn[moment][i])
```

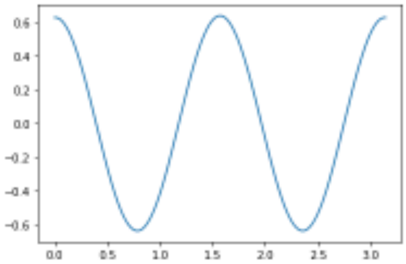
```
[ 0.0 ] = 0.011844246891550997     0.011895223842126601
[ 0.520263751549902 ] = 0.010278252391381928     0.010321344134060993
[ 1.040527503099804 ] = 0.005991073375752229     0.006016191049696079
[ 1.5607912546497063 ] = 0.00011851371646701298     0.00011901058717805053
[ 2.081055006199608 ] = -0.005785407493630604     -0.005809662909971427
[ 2.6013187577495103 ] = -0.010158372148733362     -0.010200961291518089
[ 3.1215825092994125 ] = -0.011842851149465492     -0.011892842462835477
```

```
#график погрешности
moment = 30
fig = plt.subplots()
plt.plot(x[moment], z[moment]-tochn[moment])
plt.show()
```

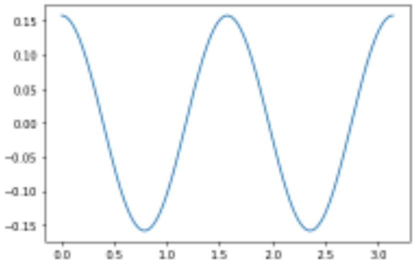


Тестовый пример 2

```
def Func2(x,t):
    return 8*np.cos(2*t)*np.cos(4*x)

l = math.pi
a = 1
T = 5
h = 0.01
t = 0.01
x,t,z =diff_scheme(l, T, h, th, Func2, a, U0_x)[0:3]

show(x,z,50)
```



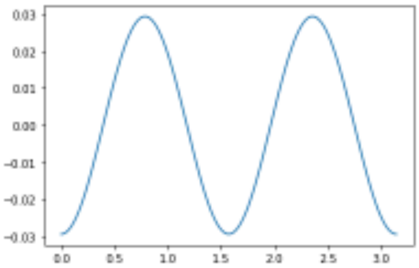
```
u = lambda x, t: 4/3*((1 + 2 *np.cos(2*t))*np.sin(t)**2)*np.cos(4*x)
N = int(l/h) +1
M = int(T/th)+1

show_tochn(u,50,N,M)
```



```
# вывод решения в виде таблицы
moment = 8
x = diff_scheme(l, T, h, th, Func2, a, U0_x)[0]
z = diff_scheme(l, T, h, th, Func2, a, U0_x)[2]
xval = np.linspace(0, math.pi, N)
yval = np.linspace(0, 10, M)
x, t = np.meshgrid(xval, yval)
tochn = u(x, t)


for i in range(0,len(x[0]),int(len(x[0])/6)):
    print("[",x[moment][i],"] =", z[moment][i], "    ", tochn[moment][i])
```

```
[ 0.0 ] = 0.025266407380480544      0.09809310679876551
[ 0.5202637515499O2 ] = -0.012371875930861405      -0.04790896681358884
[ 1.040527503099804 ] = -0.013246363707069175      -0.05129534136853565
[ 1.5607912546497063 ] = 0.025311003254957772      0.0980145631702905
[ 2.081055006199608 ] = -0.011477575711998592      -0.04444587037240904
[ 2.6013187577495103 ] = -0.014099638627376943      -0.05459957106404358
[ 3.1215825092994125 ] = 0.025221159260186823      0.09777905806539734
```

```
#график погрешности
moment = 5
fig = plt.subplots()
plt.plot(x[moment], z[moment]-tochn[moment])
plt.show()
```

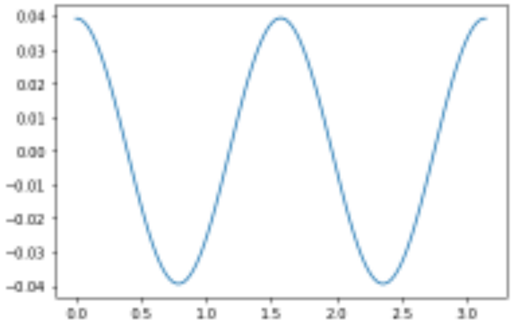


Моделирование колебаний в зависимости от внешнего сосредоточенного импульса

```
def Func(x,g,e):
    if (x < (e + g) & (x > (g - e))):
         return 2*e
    else:
        return 0
```

```
# входные данные
l = math.pi
#eps = 0.1*l
eps = 0.01*l

#g = 1/4
#g = 1/2
g = 1/3
```

```
a = 1
T = 5
h = 0.01
t = 0.01
x,t,z =diff_scheme(l, T, h, th, Func2, a, U0_x)[0:3]

show(x,z,10)
```



Содержание

Теоретическая часть **2**

Постановка задачи 2

Аналитическое решение 2

Тестовый пример 1 6

Тестовый пример 2 8

Практическая часть **12**

Численное решение примера 1 12

Численное решение примера 2 14

Моделирование колебаний в зависимости от внешнего сосредоточенного импульса 16

Код программы **17**

Содержание **21**