

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ

Лабораторная работа №2

**«Разностные схемы для обыкновенного дифференциального уравнения
второго порядка»**

Вариант 2

Выполнил:

Бовт Тимофей Анатольевич
студент 4 курса 7 группы

Преподаватель:

Репников Василий Иванович

Минск, 2024 г.

Постановка задачи

Поставлена дифференциальная задача, описывающая процесс стационарного распределения тепла в стержне единичной длины

$$\begin{cases} (k(x)u'(x))' - q(x)u(x) = -f(x), & 0 < x < 1, \\ u(0) = \mu_1, \\ -u'(1) = \sigma_2 u(1), \end{cases} \quad (1)$$

где

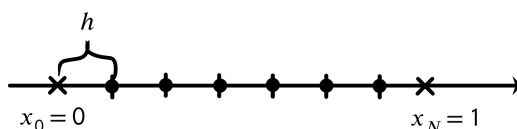
- $k(x) = 4 - x^2$ – это коэффициент теплопроводности материала стержня;
- $q(x) = x^2$ отвечает за мощность стоков или источников тепла;
- $f(x) = 4 \cos x - 2x \sin x$ – это плотность распределения внешних источников или стоков тепла;
- $\mu_1 = 1$;
- $\sigma_2 = \tan 1$;
- $u(x) = \cos x$ – температура стержня (точное решение).

1. Построить разностную схему, заменяя дифференциальные производные разностными;
2. Методом баланса построить консервативную схему, составив уравнение баланса и проведя интегрирование по $[x_{i-1}, x_i]$;
3. Построить вариационно-разностную схему методом Ритца;
4. Используя метод разностной прогонки, составить программу решения исходной задачи с помощью разностных схем п.п. 1-2, выполнить контрольные расчеты на ЭВМ и провести сравнительный анализ результатов.

1 Построение разностной схемы

Зададим равномерную сетку узлов на отрезке $[0, 1]$, на котором поставлена наша задача, следующим образом

$$\bar{\omega}_h = \left\{ x_i = ih, \quad i = \overline{0, N}, \quad h = \frac{1}{N} \right\}. \quad (2)$$



На этой сетке определим сеточную функцию $y = y(x)$, которая будет являться приближенным решением поставленной задачи (1), то есть это будет сеточное приближение функции $u = u(x)$.

Заменяя дифференциальные производные разностными, можем построить разностную схему в безиндексной форме

$$\begin{cases} \left(k \left(x - \frac{h}{2} \right) y(x)_{\bar{x}} \right)_x - q(x)y(x) = -f(x), & x \in \omega_h, \\ y(0) = \mu_1, \\ -y_{\bar{x}}(1) = \sigma_2 y(1). \end{cases} \quad (3)$$

Если мы распишем все разностные производные, то получим разностную схему в индексной форме

$$\begin{cases} \frac{k_{i+\frac{1}{2}}y_{i+1} - k_{i+\frac{1}{2}}y_i - k_{i-\frac{1}{2}}y_i + k_{i-\frac{1}{2}}y_{i-1}}{h^2} - q(x_i)y_i = -f(x_i), & i = \overline{1, N-1}, \\ y_0 = \mu_1, \\ -\frac{y_N - y_{N-1}}{h} = \sigma_2 y_N \end{cases} \quad (4)$$

Исследуем порядок аппроксимации дифференциальной задачи построенной разностной схемой. Сперва рассмотрим погрешность аппроксимации дифференциального уравнения

$$\psi_h(x) = \left(k \left(x - \frac{h}{2} \right) u_{\bar{x}} \right)_x - q(x)u(x) + f(x)$$

Учитывая, что

$$\left(k \left(x - \frac{h}{2} \right) u_{\bar{x}} \right)_x = (k(x)u'(x))' - \frac{h^2}{8}k''(x)u''(x) + O(h^3),$$

подставим в уравнение для погрешности и получим

$$\psi_h(x) = (k(x)u'(x))' - \frac{h^2}{8}k''(x)u''(x) + O(h^3) - q(x)u(x) + f(x) = -\frac{h^2}{8}k''(x)u''(x) + O(h^3) = O(h^2).$$

Таким образом, дифференциальное уравнение аппроксимируется на шаблоне со вторым порядком.

Рассмотрим погрешность аппроксимации левого граничного условия. Так как на левой границе у нас отсутствует аппроксимация производной, то левое граничное условие аппроксимируется точно, то есть

$$\nu_h(0) = u(0) - \mu_1 = 0.$$

Рассмотрим погрешность аппроксимации правого граничного условия

$$\nu_h(1) = -u_{\bar{x}}(1) - \sigma_2 u(1).$$

Так как

$$u_{\bar{x}}(x) = u'(x) - \frac{h}{2}u''(x) + O(h^2),$$

то

$$\nu_h(1) = -u'(1) + \frac{h}{2}u''(1) + O(h^2) - \sigma_2 u(1) = \frac{h}{2}u''(1) + O(h^2) = O(h).$$

Таким образом, правое граничное условие аппроксимируется с первым порядком. Тогда общий порядок аппроксимации дифференциальной задачи разностной схемой равен

$$\Psi_h(x) = \psi_h(x) + \nu_h(0) + \nu_h(1) = O(h),$$

то есть получили аппроксимацию первого порядка.

Повысим порядок аппроксимации разностной схемы, не изменяя минимального шаблона. Поскольку мы получаем аппроксимацию первого порядка именно из-за правого граничного условия, то представим его в другом виде

$$-y_{\bar{x}}(1) = \bar{\sigma}_2 y(1) + \bar{\mu}_2,$$

где $\bar{\sigma}_2, \bar{\mu}_2$ – это сеточные параметры, которые подлежат определению. Рассмотрим невязку над точным решением, чтобы определить вид введенных параметров

$$\nu_h(1) = -u_{\bar{x}}(1) - \bar{\sigma}_2 u(1) - \bar{\mu}_2 = -u'(1) + \frac{h}{2} u''(1) + O(h^2) - \bar{\sigma}_2 u(1) - \bar{\mu}_2.$$

Предположим, что дифференциальное уравнение поставленной задачи (1) выполняется на правой границе, то есть

$$(k(1)u'(1))' - q(1)u(1) = -f(1),$$

тогда получим

$$k'(1)u'(1) + k(1)u''(1) - q(1)u(1) = -f(1).$$

Отсюда можем получить

$$u''(1) = \frac{q(1)u(1) - f(1) - k'(1)u'(1)}{k(1)}.$$

Таким образом, учитывая, что $-u'(1) = \sigma_2 u(1)$, получим

$$\nu_h(1) = \sigma_2 u(1) + \frac{h}{2} \cdot \frac{q(1)u(1) - f(1) + k'(1)\sigma_2 u(1)}{k(1)} + O(h^2) - \bar{\sigma}_2 u(1) - \bar{\mu}_2.$$

Тогда можно выбрать

$$\begin{cases} \bar{\sigma}_2 = \sigma_2 \cdot \left(1 + \frac{h}{2} \cdot \frac{k'(1)}{k(1)}\right) + \frac{h}{2} \cdot \frac{q(1)}{k(1)}, \\ \bar{\mu}_2 = -\frac{h}{2} \cdot \frac{f(1)}{k(1)}. \end{cases}$$

При таком выборе мы получим аппроксимацию правого граничного условия со вторым порядком. Следовательно, вся дифференциальная задача будет аппроксимироваться со вторым порядком разностной схемой вида

$$\begin{cases} \left(k \left(x - \frac{h}{2} \right) y(x_{\bar{x}}) \right)_x - q(x)y(x) = -f(x), \quad x \in \omega_h, \\ y(0) = \mu_1, \\ -y_{\bar{x}}(1) = \left[\sigma_2 \cdot \left(1 + \frac{h}{2} \cdot \frac{k'(1)}{k(1)}\right) + \frac{h}{2} \cdot \frac{q(1)}{k(1)} \right] y(1) - \frac{h}{2} \cdot \frac{f(1)}{k(1)}. \end{cases} \quad (5)$$

Но для реализации разностной схемы нам нужна индексная форма записи

$$\begin{cases} \frac{k_{i+\frac{1}{2}}y_{i+1} - k_{i+\frac{1}{2}}y_i - k_{i-\frac{1}{2}}y_i + k_{i-\frac{1}{2}}y_{i-1}}{h^2} - q(x_i)y_i = -f(x_i), \quad i = \overline{1, N-1}, \\ y_0 = \mu_1, \\ -\frac{y_N - y_{N-1}}{h} = \left[\sigma_2 \cdot \left(1 + \frac{h}{2} \cdot \frac{k'(1)}{k(1)} \right) + \frac{h}{2} \cdot \frac{q(1)}{k(1)} \right] y_N - \frac{h}{2} \cdot \frac{f(1)}{k(1)} \end{cases} \quad (6)$$

Чтобы применить к разностной схеме метод прогонки, выпишем коэффициенты, которые будут образовывать трехдиагональную матрицу. Если мы задаем трехдиагональную матрицу в виде

$$\begin{pmatrix} \gamma_0 & \beta_0 & 0 & \dots & 0 & 0 & \left| \begin{array}{c} g_0 \\ g_1 \\ g_2 \\ \vdots \\ g_{N-1} \\ g_N \end{array} \right. \\ \alpha_1 & \gamma_1 & \beta_1 & \dots & 0 & 0 & \\ 0 & \alpha_1 & \gamma_2 & \dots & 0 & 0 & \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \\ 0 & 0 & 0 & \dots & \gamma_{N-1} & \beta_{N-1} & \\ 0 & 0 & 0 & \dots & \alpha_N & \gamma_N & \end{pmatrix} \quad (7)$$

то в соответствии с нашей разностной схемой (6) имеем

$$\begin{aligned} \gamma_0 &= 1, \quad \beta_0 = 0, \quad g_0 = \mu_1, \\ \alpha_i &= \frac{k(x_i - \frac{h}{2})}{h^2}, \quad \gamma_i = -\frac{k(x_i - \frac{h}{2}) + k(x_i + \frac{h}{2})}{h^2} - q(x_i), \quad \beta_i = \frac{k(x_i + \frac{h}{2})}{h^2}, \quad g_i = -f(x_i), \\ \alpha_N &= \frac{1}{h}, \quad \gamma_N = -\left[\frac{1}{h} + \sigma_2 \cdot \left(1 + \frac{h}{2} \cdot \frac{k'(1)}{k(1)} \right) + \frac{h}{2} \cdot \frac{q(1)}{k(1)} \right], \quad g_N = -\frac{h}{2} \cdot \frac{f(1)}{k(1)}. \end{aligned}$$

Покажем, что в данном случае метод прогонки сходится. Для сходимости метода необходимо выполнение следующих условий:

$$|\beta_0| \leq |\gamma_0|, \quad |\alpha_i| + |\beta_i| \leq |\gamma_i|, \quad |\alpha_N| \leq |\gamma_N|.$$

Действительно,

$$|\beta_0| \leq |\gamma_0| \Rightarrow 0 < 1,$$

то есть первое условие верно, причем равенство строгое;

$$|\alpha_i| + |\beta_i| \leq |\gamma_i| \Rightarrow \left| \frac{k(x_i - \frac{h}{2})}{h^2} \right| + \left| \frac{k(x_i + \frac{h}{2})}{h^2} \right| \leq \left| -\frac{k(x_i - \frac{h}{2}) + k(x_i + \frac{h}{2})}{h^2} - q(x_i) \right|,$$

оба выражения под модулями слева неотрицательны, так как являются сеточными функциями, а справа при раскрытии модуля меняем знак

$$\frac{k(x_i - \frac{h}{2})}{h^2} + \frac{k(x_i + \frac{h}{2})}{h^2} \leq \frac{k(x_i - \frac{h}{2}) + k(x_i + \frac{h}{2})}{h^2} + q(x_i),$$

отсюда

$$q(x_i) \geq 0,$$

что верно для каждого x_i из сетки;

$$|\alpha_N| \leq |\gamma_N| \Rightarrow \left| \frac{1}{h} \right| \leq \left| -\left[\frac{1}{h} + \sigma_2 \cdot \left(1 + \frac{h}{2} \cdot \frac{k'(1)}{k(1)} \right) + \frac{h}{2} \cdot \frac{q(1)}{k(1)} \right] \right|,$$

где при раскрытии модулей очевидно неравенство будет выполняться, так как справа мы прибавляем положительное число к $\frac{1}{h}$. Таким образом, метод прогонки для реализации разностной схемы сходится.

2 Построение консервативной разностной схемы методом баланса

Для построения разностной схемы нам нужно привести поставленную задачу (1) к подходящему виду. В общем случае разностная схема по методу баланса строится для задачи вида

$$\begin{cases} (k(x)u'(x))' - q(x)u(x) = -f(x), & 0 < x < 1, \\ k(0)u'(0) = \sigma_1 u(0) - \mu_1, \\ -k(1)u'(1) = \sigma_2 u(1) - \mu_2. \end{cases}$$

В нашей задаче граничные условия имеют другой вид, поэтому приведем их к нужному виду. Рассмотрим левое граничное условие. Если мы полагаем, что наше решение $u'(0) = 0$ и $\sigma_1 = 1$, то как раз получим левое граничное условие задачи (1):

$$0 = u(0) - \mu_1 \Rightarrow u(0) = \mu_1,$$

поэтому в левом граничном условии мы ничего менять не будем. В задаче у правого граничного условия есть множитель $k(1)$, который в нашем случае равен

$$k(1) = 4 - 1^2 = 3.$$

Таким образом, мы должны домножить на это значение правое граничное условие, тогда

$$-3u'(1) = 3\sigma_2 u(1).$$

Таким образом, метод баланса мы будем применять к дифференциальной задаче

$$\begin{cases} (k(x)u'(x))' - q(x)u(x) = -f(x), & 0 < x < 1, \\ u(0) = \mu_1, \\ -3u'(1) = 3\sigma_2 u(1), \end{cases} \quad (8)$$

Из предыдущего пункта мы возьмем заданную равномерную сетку узлов $\bar{\omega}_h$ и заданную на ней сеточную функцию $y = y(x)$.

По методу баланса можно построить разностную схему в безиндексном виде

$$\begin{cases} (ay_{\bar{x}})_x - dy = -\varphi, & x \in \omega_h, \\ a_1 y_{x,0} = \left(\sigma_0 + \frac{h}{2}d_0\right) y_0 - \left(\mu_0 + \frac{h}{2}\varphi_0\right), \\ -a_N y_{\bar{x},N} = \left(\sigma_1 + \frac{h}{2}d_N\right) y_N - \left(\mu_1 + \frac{h}{2}\varphi_N\right); \end{cases} \quad (9)$$

где

$$a_i = \left[\frac{1}{h} \int_{x_{i-1}}^{x_i} \frac{1}{k(x)} dx \right]^{-1}, \quad d_i = \frac{1}{h} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} q(x) dx, \quad \varphi_i = \frac{1}{h} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(x) dx, \quad (10)$$

$$d_0 = \frac{2}{h} \int_0^{\frac{h}{2}} q(x) dx, \quad \varphi_0 = \frac{2}{h} \int_0^{\frac{h}{2}} f(x) dx, \quad (11)$$

$$d_N = \frac{2}{h} \int_{1-\frac{h}{2}}^1 q(x)dx, \quad \varphi_N = \frac{2}{h} \int_{1-\frac{h}{2}}^1 f(x)dx. \quad (12)$$

Или, что то же самое, в индексном виде

$$\begin{cases} \frac{1}{h} \left(a_{i+1} \frac{y_{i+1} - y_i}{h} - a_i \frac{y_i - y_{i-1}}{h} \right) - d_i y_i = -\varphi_i, \quad i = \overline{1, N-1} \\ a_1 y_{x,0} = \left(\sigma_0 + \frac{h}{2} d_0 \right) y_0 - \left(\mu_0 + \frac{h}{2} \varphi_0 \right), \\ -a_N y_{\bar{x},N} = \left(\sigma_1 + \frac{h}{2} d_N \right) y_N - \left(\mu_1 + \frac{h}{2} \varphi_N \right); \end{cases} \quad (13)$$

Для того, чтобы явно записать консервативную разностную схему для поставленной задачи (1), нам необходимо подогнать схему вида (11) под вид дифференциальной задачи (6) и вычислить коэффициенты по формулам (8), (9), (10). При этом будем строить наилучшую консервативную разностную схему, а значит интегралы будем вычислять точно. Итак, учитывая вид граничных условий задачи (6), получим разностную схему

$$\begin{cases} \frac{1}{h} \left(a_{i+1} \frac{y_{i+1} - y_i}{h} - a_i \frac{y_i - y_{i-1}}{h} \right) - d_i y_i = -\varphi_i, \quad i = \overline{1, N-1} \\ y_0 = \mu_1, \\ -a_N \frac{y_N - y_{N-1}}{h} = \left(3\sigma_2 + \frac{h}{2} d_N \right) y_N - \frac{h}{2} \varphi_N; \end{cases} \quad (14)$$

в которой мы исключили аппроксимацию левого граничного условия, поскольку оно вычисляется точно. Определим коэффициенты, используя входные данные,

$$\begin{aligned} a_i &= \left[\frac{1}{h} \int_{x_{i-1}}^{x_i} \frac{dx}{4-x^2} \right]^{-1} = \left[\frac{1}{h} \cdot \frac{1}{2} \ln \frac{2+x}{2-x} \Big|_{x_{i-1}}^{x_i} \right]^{-1} = \frac{2h}{\ln \frac{(2+x_i)(2-x_{i-1})}{(2-x_i)(2+x_{i-1})}}, \\ d_i &= \frac{1}{h} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} x^2 dx = \frac{1}{h} \cdot \frac{x^3}{3} \Big|_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = \frac{1}{h} \cdot \frac{(x_{i+\frac{1}{2}}^3 - x_{i-\frac{1}{2}}^3)}{3}, \\ \varphi_i &= \frac{1}{h} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} [4 \cos x - 2x \sin x] dx = \frac{2}{h} (\sin x + x \cos x) \Big|_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = \\ &= \frac{2}{h} (\sin x_{i+\frac{1}{2}} - \sin x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}} \cos x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \cos x_{i-\frac{1}{2}}), \\ d_N &= \frac{2}{h} \cdot \frac{x^3}{3} \Big|_{1-\frac{h}{2}}^1 = \frac{2}{3h} - \frac{2(1-\frac{h}{2})^3}{3h}, \\ \varphi_N &= \frac{4}{h} (\sin x + x \cos x) \Big|_{1-\frac{h}{2}}^1 = \frac{4}{h} \left(\sin 1 + \cos 1 - \sin \left(1 - \frac{h}{2} \right) - \left(1 - \frac{h}{2} \right) \cos \left(1 - \frac{h}{2} \right) \right). \end{aligned}$$

Таким образом, мы имеем общую формулу (12) для итераций и явные выражения для всех коэффициентов из этой разностной схемы.

Полученная таким образом разностная схема обладает вторым порядком аппроксимации, а также для нее сходится метод прогонки. Запишем в соответствии с матрицей (7) вид коэффициентов для метода прогонки

$$\begin{aligned}\gamma_0 &= 1, \quad \beta_0 = 0, \quad g_0 = \mu_1, \\ \alpha_i &= \frac{a_i}{h^2}, \quad \gamma_i = -\frac{a_i + a_{i+1}}{h^2} - d_i, \quad \beta_i = \frac{a_{i+1}}{h^2}, \quad g_i = -\varphi_i, \\ \alpha_N &= \frac{a_N}{h}, \quad \gamma_N = -\left[\frac{a_N}{h} + 3\sigma_2 + \frac{h}{2}d_N\right], \quad g_N = -\frac{h}{2}\varphi_N.\end{aligned}$$

3 Построение вариационно-разностной схемы методом Ритца

Из первого пункта мы возьмем заданную равномерную сетку узлов $\bar{\omega}_h$ и заданную на ней сеточную функцию $y = y(x)$.

По методу Ритца мы можем построить трехдиагональную систему вида

$$\begin{cases} \alpha_{ii-1}y_{i-1} + \alpha_{ii}y_i + \alpha_{ii+1}y_{i+1} = \beta_i, \quad i = \overline{1, N-1}, \\ \alpha_{00}y_0 + \alpha_{01}y_1 = \beta_0, \\ \alpha_{NN-1}y_{N-1} + \alpha_{NN}y_N = \beta_N, \end{cases} \quad (15)$$

где

$$\begin{aligned}\alpha_{ii} &= \frac{1}{h^2} \left[\int_{x_{i-1}}^{x_{i+1}} k(x)dx + \int_{x_{i-1}}^{x_i} q(x)(x - x_{i-1})^2 dx + \int_{x_i}^{x_{i+1}} q(x)(x_{i+1} - x)^2 dx \right], \quad i = \overline{1, N-1}, \\ \alpha_{ii+1} &= \frac{1}{h^2} \left[- \int_{x_i}^{x_{i+1}} k(x)dx + \int_{x_i}^{x_{i+1}} q(x)(x - x_i)(x_{i+1} - x)dx \right], \quad i = \overline{0, N-1},\end{aligned}$$

причем $\alpha_{ii+1} = \alpha_{i+1i}$. Тогда можно вычислить

$$\beta_i = \frac{1}{h} \left[\int_{x_{i-1}}^{x_i} f(x)(x - x_{i-1})dx + \int_{x_i}^{x_{i+1}} f(x)(x_{i+1} - x)dx \right], \quad i = \overline{1, N-1},$$

$$\alpha_{00} = \frac{1}{h^2} \left[\int_0^h k(x)dx + \int_0^h q(x)(x - h)^2 dx \right] + \sigma_1,$$

$$\alpha_{NN} = \frac{1}{h^2} \left[\int_{1-h}^1 k(x)dx + \int_{1-h}^1 q(x)(x - 1 + h)^2 dx \right] + \sigma_2,$$

$$\beta_0 = \frac{1}{h} \left[\int_0^h f(x)(h-x)dx + \mu_1 \right], \quad \beta_N = \frac{1}{h} \left[\int_{1-h}^1 f(x)(x-1+h)dx + \mu_2 \right].$$

Нас интересуют граничные условия другого вида, поэтому мы будем вычислять

$$a_i = -h\alpha_{ii-1}, \quad \varphi_i = \frac{1}{h}\beta_i, \quad d_i = \frac{1}{h}(\alpha_{ii-1} + \alpha_{ii} + \alpha_{ii+1}),$$

$$d_0 = \frac{2}{h^2} \int_0^h q(x)(h-x)dx, \quad d_N = \frac{2}{h^2} \int_{1-h}^1 q(x)(x-1+h)dx,$$

$$\varphi_0 = \frac{2}{h^2} \int_0^h f(x)(x-h)dx, \quad \varphi_N = \frac{2}{h^2} \int_{1-h}^1 f(x)(x-1+h)dx,$$

откуда разностная схема будет иметь вид

$$\begin{cases} \frac{a_i}{h^2}y_{i-1} - \left(\frac{a_i + a_{i+1}}{h} + d_i \right) y_i + \frac{a_{i+1}}{h^2}y_{i+1} = -\varphi_i, \quad i = \overline{1, N-1} \\ a_1 y_{x,0} = \left(\sigma_0 + \frac{h}{2}d_0 \right) y_0 - \left(\mu_0 + \frac{h}{2}\varphi_0 \right), \\ -a_N y_{\bar{x},N} = \left(\sigma_1 + \frac{h}{2}d_N \right) y_N - \left(\mu_1 + \frac{h}{2}\varphi_N \right); \end{cases}$$

но в нашем случае (по аналогии с методом баланса)

$$\begin{cases} \frac{a_i}{h^2}y_{i-1} - \left(\frac{a_i + a_{i+1}}{h} + d_i \right) y_i + \frac{a_{i+1}}{h^2}y_{i+1} = -\varphi_i, \quad i = \overline{1, N-1} \\ y_0 = \mu_1, \\ -a_N \frac{y_N - y_{N-1}}{h} = \left(3\sigma_2 + \frac{h}{2}d_N \right) y_N - \frac{h}{2}\varphi_N; \end{cases} \quad (16)$$

Тогда, подставляя известные значения, получим

$$\alpha_{ii} = \frac{1}{h^2} \left[\left(4x - \frac{x}{3} \right) \Big|_{x_{i-1}}^{x_{i+1}} + \left(\frac{x^5}{5} - \frac{x^4 x_{i-1}}{2} + \frac{x^3 x_{i-1}^2}{3} \right) \Big|_{x_{i-1}}^{x_i} + \left(\frac{x^5}{5} - \frac{x^4 x_{i+1}}{2} + \frac{x^3 x_{i+1}^2}{3} \right) \Big|_{x_i}^{x_{i+1}} \right],$$

$$\alpha_{ii+1} = \frac{1}{h^2} \left[- \left(4x - \frac{x}{3} \right) \Big|_{x_i}^{x_{i+1}} + \left(\frac{x^4(x_i + x_{i+1})}{4} - \frac{1}{3}x_i x_{i+1} x^3 - \frac{x^5}{5} \right) \Big|_{x_i}^{x_{i+1}} \right],$$

$$\alpha_{ii-1} = \frac{1}{h^2} \left[- \left(4x - \frac{x}{3} \right) \Big|_{x_{i-1}}^{x_i} + \left(\frac{x^4(x_{i-1} + x_i)}{4} - \frac{1}{3}x_{i-1} x_i x^3 - \frac{x^5}{5} \right) \Big|_{x_{i-1}}^{x_i} \right],$$

$$\beta_i = \frac{1}{h} \left[-2(x_{i-1} \sin x - x(x - x_{i-1}) \cos x) \Big|_{x_{i-1}}^{x_i} + 2(x_{i+1} \sin x + x(x_{i+1} - x) \cos x) \Big|_{x_i}^{x_{i+1}} \right],$$

$$d_N = \frac{2}{12h^2} [x^3 \cdot (4h + 3x - 4)] \Big|_{1-h}^1,$$

$$\varphi_N = \frac{2}{h^2} [2(h-1) \sin x + 2x(h+x-1) \cos x] \Big|_{1-h}^1.$$

Полученная таким образом разностная схема обладает вторым порядком аппроксимации, а также для нее сходится метод прогонки.

Для нее коэффициенты метода прогонки возьмем такие же, как и в предыдущем случае, поскольку мы привели разностную схему к такому же виду, как для метода баланса.

4 Реализация на ЭВМ

Подключим все необходимые библиотеки для вычислений и для визуализации

```
[1]: import numpy as np
import math
import matplotlib.pyplot as plt
```

Определим функцию для решения системы с трехдиагональной матрицей

```
[2]: def tridiagonal_algorithm(a,b,c,f):
    a, b, c, f = tuple(map(lambda k_list: list(map(float, k_list)), (a,
    ↪b, c, f)))

    alpha = [-b[0] / c[0]]
    beta = [f[0] / c[0]]
    n = len(f)
    x = [0]*n

    for i in range(1, n):
        alpha.append(-b[i]/(a[i]*alpha[i-1] + c[i]))
        beta.append((f[i] - a[i]*beta[i-1])/(a[i]*alpha[i-1] + c[i]))

    x[n-1] = beta[n - 1]

    for i in range(n - 1, 0, -1):
        x[i - 1] = alpha[i - 1]*x[i] + beta[i - 1]

    return x
```

В соответствии с входными данными определим функции:

- $k(x) = 4 - x^2$ – это коэффициент теплопроводности материала стержня;
- $q(x) = x^2$ отвечает за мощность стоков или источников тепла;
- $f(x) = 4 \cos x - 2x \sin x$ – это плотность распределения внешних источников или стоков тепла;
- $\mu_1 = 1$;
- $\sigma_2 = \tan 1$;

Также нам понадобится функция

- $k'(x) = -2x$.

```
[4]: def k(x):
      return 4 - x**2

      def dkdx(x):
          return -2*x

      def q(x):
          return x**2

      def f(x):
          return 4*np.cos(x) - 2*x*np.sin(x)

      mu_1 = 1
      sigma_2 = np.tan(1)
```

Определим функцию, соответствующую точному решению дифференциальной задачи

$$u(x) = \cos x$$

```
[30]: def u(x):
       return np.cos(x)
```

Определим сетку узлов ω_h следующим образом: пусть $[a, b] = [0, 1]$, а число разбиений отрезка $N = 5$.

```
[31]: a, b = 0, 1
      N = 5
      x = np.linspace(start=a, stop=b, num=N+1)
      h = (b-a)/N
```

4.1 Разностная аппроксимация

Итак, у нас есть разностная схема

$$\begin{cases} \frac{k_{i+\frac{1}{2}}y_{i+1} - k_{i+\frac{1}{2}}y_i - k_{i-\frac{1}{2}}y_i + k_{i-\frac{1}{2}}y_{i-1}}{h^2} - q(x_i)y_i = -f(x_i), \quad i = \overline{1, N-1}, \\ y_0 = \mu_1, \\ -\frac{y_N - y_{N-1}}{h} = \left[\sigma_2 \cdot \left(1 + \frac{h}{2} \cdot \frac{k'(1)}{k(1)} \right) + \frac{h}{2} \cdot \frac{q(1)}{k(1)} \right] y_N - \frac{h}{2} \cdot \frac{f(1)}{k(1)} \end{cases}$$

Чтобы найти с помощью нее приближенное решение дифференциальной задачи, нам нужно преобразовать эту разностную схему к задаче на решение системы с трехдиагональной матрицей. Таким образом, задача на решение системы уравнений

примет вид

$$\left(\begin{array}{cccccc|c} \gamma_0 & \beta_0 & 0 & \dots & 0 & 0 & g_0 \\ \alpha_1 & \gamma_1 & \beta_1 & \dots & 0 & 0 & g_1 \\ 0 & \alpha_1 & \gamma_2 & \dots & 0 & 0 & g_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \gamma_{N-1} & \beta_{N-1} & g_{N-1} \\ 0 & 0 & 0 & \dots & \alpha_N & \gamma_N & g_N \end{array} \right)$$

где

$$\begin{aligned} \gamma_0 &= 1, \beta_0 = 0, g_0 = \mu_1, \\ \alpha_i &= \frac{k(x_i - \frac{h}{2})}{h^2}, \gamma_i = -\frac{k(x_i - \frac{h}{2}) + k(x_i + \frac{h}{2})}{h^2} - q(x_i), \beta_i = \frac{k(x_i + \frac{h}{2})}{h^2}, g_i = -f(x_i), \\ \alpha_N &= \frac{1}{h}, \gamma_N = -\left[\frac{1}{h} + \sigma_2 \cdot \left(1 + \frac{h}{2} \cdot \frac{k'(1)}{k(1)} \right) + \frac{h}{2} \cdot \frac{q(1)}{k(1)} \right], g_N = -\frac{h}{2} \cdot \frac{f(1)}{k(1)}. \end{aligned}$$

Реализуем вычисление всех коэффициентов:

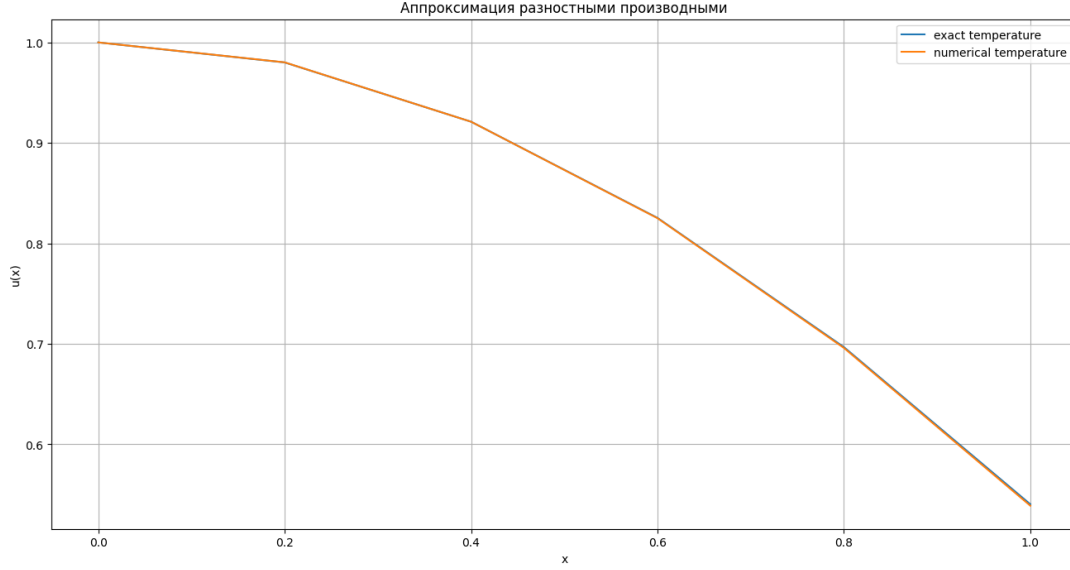
```
[36]: gamma = [1]
      beta = [0]
      g = [mu_1]
      alpha = [0]
      for i in range(1, N):
          alpha.append(k(x[i] - h/2) / h**2)
          gamma.append(-(k(x[i] - h/2) + k(x[i] + h/2)) / h**2 - q(x[i]))
          beta.append(k(x[i] + h/2) / h**2)
          g.append(-f(x[i]))
      alpha.append(1/h)
      gamma.append(-(1/h + sigma_2 * (1 + h/2 * dkdx(1) / k(1)) + h/2 * q(1) /
          ↪ k(1)))
      g.append(-h/2 * f(1) / k(1))
      beta.append(0)
```

Теперь определим приближенное решение разностной задачи как решение системы с трехдиагональной матрицей

```
[37]: y = tridiagonal_algorithm(alpha,beta,gamma,g)
```

Визуализируем полученные результаты

```
[38]: plt.figure(figsize=(16, 8))
      plt.plot(x, u(x), label='exact temperature')
      plt.plot(x, y, label='numerical temperature')
      plt.title('Аппроксимация разностными производными')
      plt.grid(True)
      plt.xlabel('x')
      plt.ylabel('u(x)')
      plt.legend()
      plt.show()
```



Как можно видеть из графика, приближенное решение максимально близко описывает точное решение задачи.

4.2 Метод баланса

Итак, у нас есть разностная схема

$$\begin{cases} \frac{1}{h} \left(a_{i+1} \frac{y_{i+1} - y_i}{h} - a_i \frac{y_i - y_{i-1}}{h} \right) - d_i y_i = -\varphi_i, & i = \overline{1, N-1} \\ y_0 = \mu_1, \\ -a_N \frac{y_N - y_{N-1}}{h} = \left(3\sigma_2 + \frac{h}{2} d_N \right) y_N - \frac{h}{2} \varphi_N; \end{cases}$$

где

$$a_i = \left[\frac{1}{h} \int_{x_{i-1}}^{x_i} \frac{dx}{4 - x^2} \right]^{-1} = \left[\frac{1}{h} \cdot \frac{1}{2} \ln \frac{2+x}{2-x} \Big|_{x_{i-1}}^{x_i} \right]^{-1} = \frac{2h}{\ln \frac{(2+x_i)(2-x_{i-1})}{(2-x_i)(2+x_{i-1})}},$$

$$d_i = \frac{1}{h} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} x^2 dx = \frac{1}{h} \cdot \frac{x^3}{3} \Big|_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = \frac{1}{h} \cdot \frac{(x_{i+\frac{1}{2}}^3 - x_{i-\frac{1}{2}}^3)}{3},$$

$$\begin{aligned} \varphi_i &= \frac{1}{h} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} [4 \cos x - 2x \sin x] dx = \frac{2}{h} (\sin x + x \cos x) \Big|_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} = \\ &= \frac{2}{h} (\sin x_{i+\frac{1}{2}} - \sin x_{i-\frac{1}{2}} + x_{i+\frac{1}{2}} \cos x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \cos x_{i-\frac{1}{2}}), \end{aligned}$$

$$d_N = \frac{2}{h} \cdot \frac{x^3}{3} \Big|_{1-\frac{h}{2}}^1 = \frac{2}{3h} - \frac{2(1-\frac{h}{2})^3}{3h},$$

$$\varphi_N = \frac{4}{h} (\sin x + x \cos x) \Big|_{1-\frac{h}{2}}^1 = \frac{4}{h} \left(\sin 1 + \cos 1 - \sin \left(1 - \frac{h}{2}\right) - \left(1 - \frac{h}{2}\right) \cos \left(1 - \frac{h}{2}\right) \right).$$

Определим функции для вычисления коэффициентов

```
[46]: def a_i(x, h):
        return 2 * h / np.log((2+x)*(2-(x-h)) / ((2-x)*(2+(x-h))))

def d_i(x, h):
    return ((x+h/2)**3 - (x-h/2)**3)/(3*h)

def phi_i(x, h):
    return 2/h * (np.sin(x+h/2) - np.sin(x-h/2) + (x+h/2)*np.cos(x+h/2)
    ↪ - (x-h/2)*np.cos(x-h/2))

def d_N(h):
    return 2/(3*h) * (1 - (1-h/2)**3)

def phi_N(h):
    return 4/h * (np.sin(1) + np.cos(1) - np.sin(1-h/2) - (1-h/2)*np.
    ↪ cos(1-h/2))
```

Аналогично предыдущему пункту сведем решение разностной задачи к решению системы с трехдиагональной матрицей, определив коэффициенты следующим образом

$$\begin{aligned} \gamma_0 &= 1, \beta_0 = 0, g_0 = \mu_1, \\ \alpha_i &= \frac{a_i}{h^2}, \gamma_i = -\frac{a_i + a_{i+1}}{h^2} - d_i, \beta_i = \frac{a_{i+1}}{h^2}, g_i = -\varphi_i, \\ \alpha_N &= \frac{a_N}{h}, \gamma_N = -\left[\frac{a_N}{h} + 3\sigma_2 + \frac{h}{2}d_N\right], g_N = -\frac{h}{2}\varphi_N. \end{aligned}$$

Вычислим все эти коэффициенты по указанным формулам

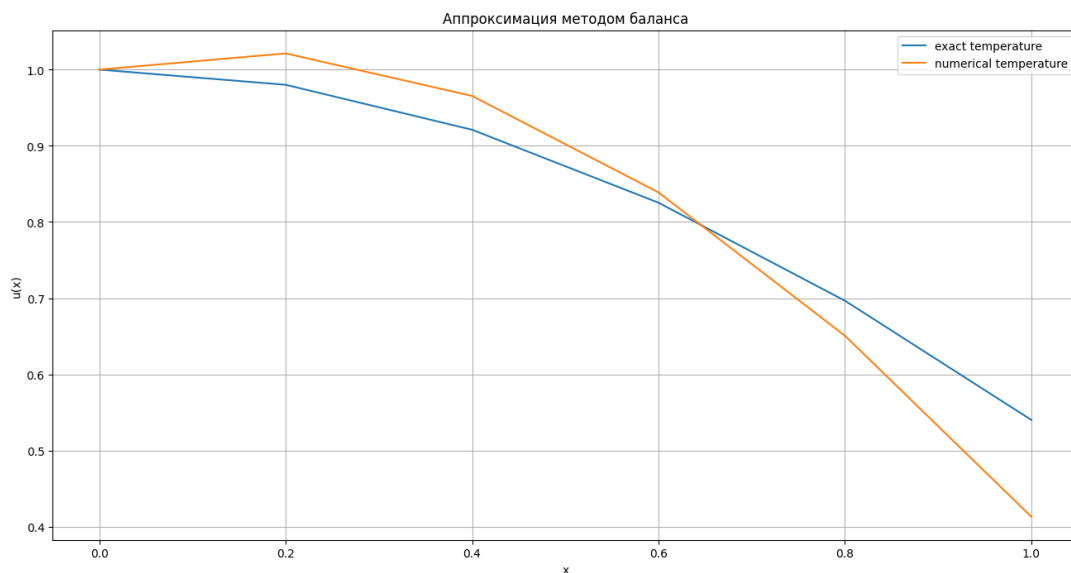
```
[47]: gamma = [1]
beta = [0]
g = [mu_1]
alpha = [0]
for i in range(1, N):
    alpha.append(a_i(x[i], h) / h**2)
    gamma.append(-(a_i(x[i],h) + a_i(x[i+1],h))/h**2 - d_i(x[i], h))
    beta.append(a_i(x[i+1], h) / h**2)
    g.append(-phi_i(x[i], h))
alpha.append(a_i(x[N], h)/h)
gamma.append(-(a_i(x[N], h) / h + 3 * sigma_2 + h/2 * d_N(h)))
g.append(-h/2 * phi_N(h))
beta.append(0)
```

Теперь определим приближенное решение разностной задачи как решение системы с трехдиагональной матрицей

```
[48]: y = tridiagonal_algorithm(alpha,beta,gamma,g)
```

Визуализируем полученные результаты

```
[49]: plt.figure(figsize=(16, 8))
plt.plot(x, u(x), label='exact temperature')
plt.plot(x, y, label='numerical temperature')
plt.title('Аппроксимация методом баланса')
plt.grid(True)
plt.xlabel('x')
plt.ylabel('u(x)')
plt.legend()
plt.show()
```



Как можно видеть из графика, приближенное решение неплохо описывает точное решение задачи.

4.3 Метод Рунге

Итак, у нас есть разностная схема

$$\begin{cases} \frac{a_i}{h^2} y_{i-1} - \left(\frac{a_i + a_{i+1}}{h} + d_i \right) y_i + \frac{a_{i+1}}{h^2} y_{i+1} = -\varphi_i, & i = \overline{1, N-1} \\ y_0 = \mu_1, \\ -a_N \frac{y_N - y_{N-1}}{h} = \left(3\sigma_2 + \frac{h}{2} d_N \right) y_N - \frac{h}{2} \varphi_N; \end{cases}$$

где

$$a_i = -h\alpha_{ii-1}, \quad \varphi_i = \frac{1}{h}\beta_i, \quad d_i = \frac{1}{h}(\alpha_{ii-1} + \alpha_{ii} + \alpha_{ii+1}),$$

$$\alpha_{ii} = \frac{1}{h^2} \left[\left(4x - \frac{x}{3} \right) \Big|_{x_{i-1}}^{x_{i+1}} + \left(\frac{x^5}{5} - \frac{x^4 x_{i-1}}{2} + \frac{x^3 x_{i-1}^2}{3} \right) \Big|_{x_{i-1}}^{x_i} + \left(\frac{x^5}{5} - \frac{x^4 x_{i+1}}{2} + \frac{x^3 x_{i+1}^2}{3} \right) \Big|_{x_i}^{x_{i+1}} \right],$$

$$\alpha_{ii+1} = \frac{1}{h^2} \left[- \left(4x - \frac{x}{3} \right) \Big|_{x_i}^{x_{i+1}} + \left(\frac{x^4 (x_i + x_{i+1})}{4} - \frac{1}{3} x_i x_{i+1} x^3 - \frac{x^5}{5} \right) \Big|_{x_i}^{x_{i+1}} \right],$$

$$\alpha_{ii-1} = \frac{1}{h^2} \left[- \left(4x - \frac{x}{3} \right) \Big|_{x_{i-1}}^{x_i} + \left(\frac{x^4 (x_{i-1} + x_i)}{4} - \frac{1}{3} x_{i-1} x_i x^3 - \frac{x^5}{5} \right) \Big|_{x_{i-1}}^{x_i} \right],$$

$$\beta_i = \frac{1}{h} \left[-2(x_{i-1} \sin x - x(x - x_{i-1}) \cos x) \Big|_{x_{i-1}}^{x_i} + 2(x_{i+1} \sin x + x(x_{i+1} - x) \cos x) \Big|_{x_i}^{x_{i+1}} \right],$$

$$d_N = \frac{2}{12h^2} [x^3 \cdot (4h + 3x - 4)] \Big|_{1-h}^1,$$

$$\varphi_N = \frac{2}{h^2} [2(h-1) \sin x + 2x(h+x-1) \cos x] \Big|_{1-h}^1.$$

```
[50]: def alpha_ii(x_i, x_im1, x_ip1, h):
    term1 = (4 * x_ip1 - x_ip1 / 3) - (4 * x_im1 - x_im1 / 3)
    term2 = ((x_i**5 / 5 - x_i**4 * x_im1 / 2 + x_i**3 * x_im1**2 / 3) -
              (x_im1**5 / 5 - x_im1**4 * x_im1 / 2 + x_im1**3 * x_im1**2
    ↪ / 3))
    term3 = ((x_ip1**5 / 5 - x_ip1**4 * x_ip1 / 2 + x_ip1**3 * x_ip1**2
    ↪ / 3) -
              (x_i**5 / 5 - x_i**4 * x_ip1 / 2 + x_i**3 * x_ip1**2 / 3))

    result = (1 / h**2) * (term1 + term2 + term3)
    return result

def alpha_ii_plus_1(x_i, x_ip1, h):
    term1 = -(4 * x_ip1 - x_ip1 / 3) + (4 * x_i - x_i / 3)
    term2 = ((x_ip1**4 * (x_i + x_ip1) / 4 - 1/3 * x_i * x_ip1 *
    ↪ x_ip1**3 - x_ip1**5 / 5) -
              (x_i**4 * (x_i + x_ip1) / 4 - 1/3 * x_i * x_ip1 * x_i**3
    ↪ x_i**5 / 5))

    result = (1 / h**2) * (term1 + term2)
    return result

def alpha_ii_minus_1(x_im1, x_i, h):
    term1 = -(4 * x_i - x_i / 3) + (4 * x_im1 - x_im1 / 3)
    term2 = ((x_i**4 * (x_im1 + x_i) / 4 - 1/3 * x_im1 * x_i * x_i**3
    ↪ x_i**5 / 5) -
              (x_im1**4 * (x_im1 + x_i) / 4 - 1/3 * x_im1 * x_i *
    ↪ x_im1**3 - x_im1**5 / 5))

    result = (1 / h**2) * (term1 + term2)
    return result

def beta_i(x_i, x_im1, x_ip1, h):
```



```

    term1 = -2 * ((x_i * np.sin(x_i) - x_i * (x_i - x_im1) * np.
↪cos(x_i)) -
                    (x_im1 * np.sin(x_im1) - x_im1 * (x_im1 - x_im1) * np.
↪cos(x_im1)))
    term2 = 2 * ((x_ip1 * np.sin(x_ip1) + x_ip1 * (x_ip1 - x_ip1) * np.
↪cos(x_ip1)) -
                    (x_i * np.sin(x_i) + x_i * (x_ip1 - x_i) * np.
↪cos(x_i)))

    result = (1 / h) * (term1 + term2)
    return result

def d_N(h):
    term1 = (1 - h)**3 * (4 * h + 3 * (1 - h) - 4)
    term2 = 1**3 * (4 * h + 3 * 1 - 4)
    result = (2 / (12 * h**2)) * (term2 - term1)
    return result

def phi_N(h):
    term1 = 2 * ((h - 1) * np.sin(1) + 1 * (h + 1 - 1) * np.cos(1)) - \
            2 * ((h - 1) * np.sin(1 - h) + (1 - h) * (h + (1 - h) - 1)
↪* np.cos(1 - h))
    result = (2 / h**2) * term1
    return result

def a_i(x, h):
    return -h * alpha_ii_minus_1(x-h, x, h)

def d_i(x, h):
    return 1/h * (alpha_ii_minus_1(x-h, x, h) + alpha_ii(x, x-h, x+h,
↪h) + alpha_ii_plus_1(x, x+h, h))

def phi_i(x, h):
    return 1/h * beta_i(x, x-h, x+h, h)

```

Как и в методе баланса сведем решение разностной задачи к решению системы с такой же трехдиагональной матрицей

```

[54]: gamma = [1]
      beta = [0]
      g = [mu_1]
      alpha = [0]
      for i in range(1, N):
          alpha.append(a_i(x[i], h) / h**2)
          gamma.append(-(a_i(x[i], h) + a_i(x[i+1], h))/h**2 - d_i(x[i], h))
          beta.append(a_i(x[i+1], h) / h**2)
          g.append(-phi_i(x[i], h))
      alpha.append(a_i(x[N], h)/h)

```

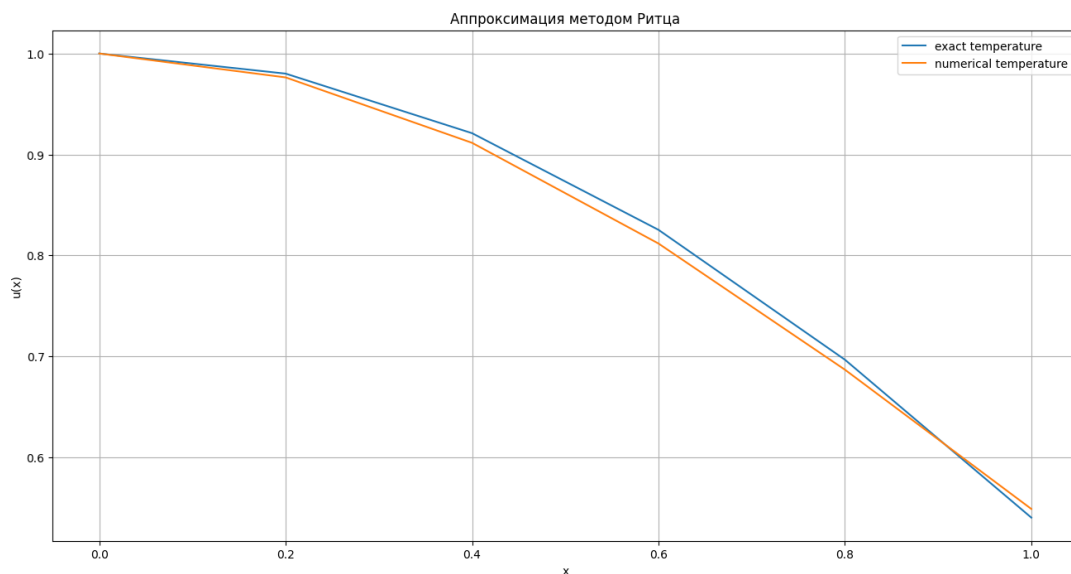
```
gamma.append(-(a_i(x[N], h) / h + 3 * sigma_2 + h/2 * d_N(h)))
g.append(-h/2 * phi_N(h))
beta.append(0)
```

Теперь определим приближенное решение разностной задачи как решение системы с трехдиагональной матрицей

```
[52]: y = tridiagonal_algorithm(alpha,beta,gamma,g)
```

Визуализируем полученные результаты

```
[53]: plt.figure(figsize=(16, 8))
plt.plot(x, u(x), label='exact temperature')
plt.plot(x, y, label='numerical temperature')
plt.title('Аппроксимация методом Рунге')
plt.grid(True)
plt.xlabel('x')
plt.ylabel('u(x)')
plt.legend()
plt.show()
```



Как можно видеть из графика, приближенное решение достаточно близко описывает точное решение задачи.

Таким образом, на основе проведенных исследований мы получили, что разностная схема, полученная путем замены производных разностными аналогами дает нам самую лучшую аппроксимацию. Разностная схема, построенная по методу Рунге дает достаточно близкую аппроксимацию, но хуже, чем разностная аппроксимация. Метод баланса дал нам самый худший результат.