Вычислительный практикум (9-ый семестр, 2021)

Интегральные уравнения первого рода

Руководитель: А. Г. Доронина   *Выполнил: П. Л. Соболев*

Задачи

- Получить решение интегрального уравнения первого рода при заданных интервалах интегрирования, ядре и правой части.

Теория

Интегральное уравнение первого рода:

$$\int_a^b K(x,s)\,z(s)\,ds = u(x) \tag{1}$$

Будем полагать, что ядро $K(x,s)$ есть вещественная, непрерывная в области $\{a \leqslant s \leqslant b;\, c \leqslant x \leqslant d\}$ функция. Возьмем в качестве стабилизирующего функционала $\Omega\,[z]$ функционал вида

$$\Omega\,[z] = \int_a^b \{z^2 + p(z')^2\}\,ds, \tag{2}$$

где p — положительное число.

Обозначим за F_n класс непрерывных на $[a,b]$ функций $z(s)$, имеющих обобщенные производные до n-го порядка, интегрируемые с квадратом на $[a,b]$. Пусть точное решение $z_{\text{т}}(s)$ принадлежит F_1 и удовлетворяет краевым условиям: $z'(a)=0$, $z'(b)=0$. Тогда в качестве регуляризованных решений z_α уравнения (1) можно брать функции, являющиеся решениями следующей краевой задачи для уравнения Эйлера:

$$\int_a^b \overline{K}(s,t)\,z(t)\,dt + \alpha\{z(s) - pz''(s)\} = g(s), \tag{3}$$

$$z'(a) = 0, z'(b) = 0, \tag{4}$$

где

$$\overline{K}(s,t) = \int_c^d K(x,s)\,K(x,t)\,dx, \quad g(s) = \int_c^d K(x,s)\,u(x)\,dx \tag{5}$$

Напишем разностный аналог уравнения (3) на равномерной сетке с шагом h. Разобьем промежуток $[a,b]$ на n равных частей и возьмем в качестве узловых точек сетки середины полученных отрезков, т. е. полагаем

$$s_i = a + 0.5\cdot h + (i-1)h, \quad i = 1,2,\dots,n; \quad h = \frac{b-a}{n} \tag{6}$$

Заменив в левой части уравнения (3) интеграл соответствующей ему интегральной суммой, например, по формуле прямоугольников, а $z''(s)$ — соответствующим разностным отношением, получим

$$\sum_{j=1}^{n} \overline{K}(s_i, t_j)\, h z_j + \alpha z_i + \alpha\, \frac{2z_i - z_{i-1} - z_{i+1}}{h^2} = g_i, \tag{7}$$

где

$$i = 1, 2, \dots, n, \quad g_i = \int_c^d K(x, s_i)\, u(x)\, dx \tag{8}$$

Значения $\overline{K}(s_i, t_j)$ и g_i либо вычисляются аналитически, либо получаются с помощью соответствующих квадратурных формул.

При $i = 1$ и $i = n$ в (7) входят не определенные ещё значения z_0 и z_{n+1}. Чтобы удовлетворить граничным условиям, полагаем $z_0 = z_1$ и $z_{n+1} = z_n$. Пусть B — матрица с элементами $B_{ij} = \overline{K}(s_i, t_j)\, h$. Тогда систему уравнений (7) относительно вектора z с компонентами $(z_1, z_2, \dots, z_n)$ можно записать в виде

$$B_\alpha z \equiv Bz + \alpha C z = g, \tag{9}$$

где g — вектор с компонентами $(g_1, g_2, \dots, g_n)$, а αC — симметричная матрица вида

$$\begin{bmatrix}
\alpha\left(1 + \frac{1}{h^2}\right) & -\frac{\alpha}{h^2} & 0 & \dots & 0 & 0 \\
-\frac{\alpha}{h^2} & \alpha\left(1 + \frac{2}{h^2}\right) & -\frac{\alpha}{h^2} & \dots & 0 & 0 \\
0 & -\frac{\alpha}{h^2} & \alpha\left(1 + \frac{2}{h^2}\right) & \dots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \dots & \alpha\left(1 + \frac{2}{h^2}\right) & -\frac{\alpha}{h^2} \\
0 & 0 & 0 & \dots & -\frac{\alpha}{h^2} & \alpha\left(1 + \frac{1}{h^2}\right)
\end{bmatrix}$$

Таким образом, задача сводится к решению СЛАУ (9).

Реализация

Алгоритм реализован на языке программирования Julia в виде скрипта и расположен в GitLab репозитории Computational Workshop S09-2021 в папке A1. Для воспроизведения результатов следуй инструкциям в файле `README.md`.

Для проверки алгоритма возьмем интегральное уравнение

$$\int_0^1 e^{sx}\, z(s)\, ds = \frac{e^{x+1} - 1}{x + 1}, \tag{10}$$

которое имеет аналитическое решение $z(s) = e^z$.

Листинг 1: Определение ядра и правой части тестового уравнения

```
# Define the kernel (Test)
K(x, s) = exp(s * x)

# Define the right part of the equation (Test)
u(x) = (exp(x + 1) - 1) / (x + 1)
```

Листинг 2: Определение интервала и подготовка узлов

```
# Set the integration intervals
a = 0
b = 1
c = 0
d = 1

# Set the number of nodes
n = 100

# Set the initial value of the regularization parameter
α = 0.001

# Calculate the step
h = (b - a) / n

# Calculate the nodes for the s argument
s = [ a + 0.5 * h + (i - 1) * h for i in 1:n ]

# Calculate the nodes for the t argument
t = copy(s)
```

Листинг 3: Вычисление вектора g, подготовка других переменных

```
# Compute the g vector
g = [
  quadgk(x -> K(x, s[i]) * u(x), c, d; rtol=1e-8)[1]
  for i in 1:n
]

# Prepare a matrix for the computation
Bα = Matrix{Float64}(undef, n, n)

# Prepare a vector for the solution
z = Vector{Float64}(undef, n)

# Prepare a range of nodes for the residual calculation
xr = range(c, d; length=1000)
```

Функция **quadgk** (из пакета `QuadGK.jl`) вычисляет значение интеграла методом Гаусса–Кронрода.

Программа создает и оптимизирует функцию с параметром α, возвращающую значение невязки для вычисленного решения:

Листинг 4: Определение функции, вычисляющей невязку

```
# Compute the residual of the solution with the
# specified regularization parameter
function residual(θ::Vector{Float64})::Float64
    # Unpack the parameters
    α = θ[1]

    # Compute the Bα matrix
    for i in 1:n, j in 1:n
        αc = if (i == j == 1) || (i == j == n)
            α * (1 + 1 / h^2)
        elseif i == j
            α * (1 + 2 / h^2)
        elseif (i == j + 1) || (i + 1 == j)
            -α / h^2
        else
            0
        end
        Bα[i, j] =
        quadgk(
          x -> K(x, s[i]) * K(x, t[j]), c, d; rtol=1e-8
        )[1] * h + αc
    end

    # Compute the solution
    z .= Symmetric(Bα) \ g

    # Calculate the residual
    r = norm(
      [ sum(K.(x, s) .* z .* h) for x in xr ] .- u.(xr)
    )

    return r
end
```

Листинг 5: Оптимизация по значению невязки

```
# Optimize over the regularization parameter
res = Optim.optimize(
    residual,
    [α,],
    LBFGS(),
    Optim.Options(
        show_trace = false,
        extended_trace = true,
        store_trace = true,
    );
    inplace = false,
)
```

Полученные значения параметра регуляризации и невязки для тестового интегрального уравнения: $\alpha = 1.5813849 \cdot 10^{-7}$, $r = 0.0021192$.

График сравнения решений:

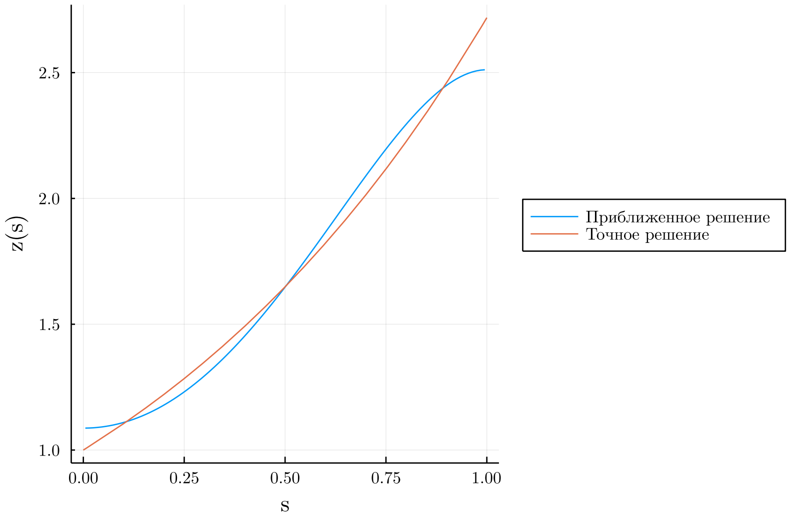

Рис. 1: Сравнение точного и приближенного решений тестового интегрального уравнения

Ядро и правая часть из условий задания:

$$K(x,s) = \frac{1}{1+x+s} \tag{11}$$

$$u(x) = \frac{1}{\sqrt{2-x}} \left(\ln\frac{2+x}{1+x} + 2\ln\frac{\sqrt{3}+\sqrt{2-x}}{2+\sqrt{2-x}} \right) \tag{12}$$

Отрезки интегрирования те же: $[0, 1]$.

Листинг 6: Определение ядра и правой части заданного уравнения

```
# Define the kernel
K(x, s) = 1 / (1 + x + s)

# Define the right part of the equation
u(x) = 1 / sqrt(2 - x) * (log((2 + x) / (1 + x)) +
       2 * log(
         (sqrt(3) + sqrt(2 - x)) / (2 + sqrt(2 - x)))
       )
```

Полученные значения параметра регуляризации и невязки для заданного интегрального уравнения: $\alpha = 2.2101166 \cdot 10^{-6}$, $r = 0.0007930$.

График решения:

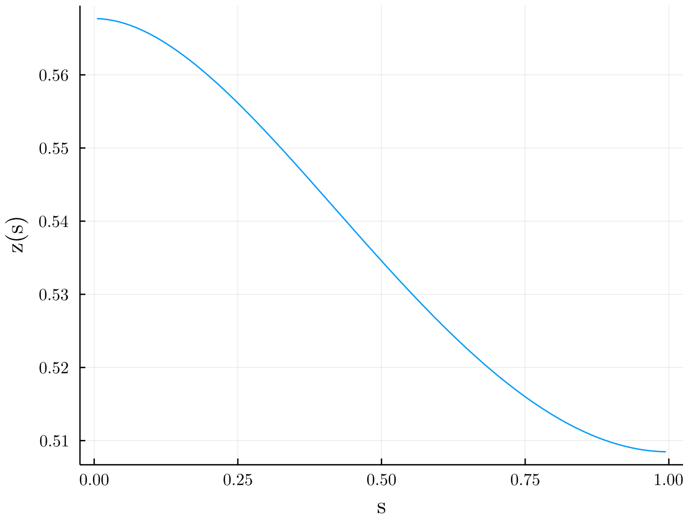

Рис. 2: Приближенное решение заданного интегрального уравнения