

МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГУ»)

Факультет прикладной математики, информатики и механики  
Кафедра математического и прикладного анализа

**ЛАБОРАТОРНАЯ РАБОТА №3**

**ЧИСЛЕННОЕ РЕШЕНИЕ СТАЦИОНАРНОГО  
УРАВНЕНИЯ ШРЁДИНГЕРА: РАСЧЕТ  
ОСНОВНОГО КВАНТОВОГО СОСТОЯНИЯ  
ЧАСТИЦЫ В ОДНОМЕРНОЙ ПОТЕНЦИАЛЬНОЙ  
ЯМЕ С БЕСКОНЕЧНЫМИ СТЕНКАМИ С  
ИСПОЛЬЗОВАНИЕМ РАЗЛОЖЕНИЯ ИСКОМОЙ  
ВОЛНОВОЙ ФУНКЦИИ ПО БАЗИСУ**

Направление: 01.04.02 - Прикладная математика и информатика  
Профиль: Компьютерные технологии в задачах математической физики,  
оптимизации и управления

Выполнил:  
студент 11 группы 2 курса магистратуры  
Маркин Р.О.

Преподаватель:  
доктор физ.-мат. наук, профессор  
Тимошенко Ю.К.

Воронеж 2024

## Содержание

1	Цели и задачи работы . . . . .	3
2	Одномерное стационарное уравнение Шрёдингера. Математический формализм. Общие свойства решений . . . . .	4
3	Прямой вариационный метод. Алгоритм . . . . .	6
4	Программная реализация алгоритма . . . . .	8
5	Результаты численных экспериментов и их анализ . . . . .	9
	Приложение 1. Компьютерный код . . . . .	12
	Список литературы . . . . .	16

# 1. Цели и задачи работы

## Цели работы.

Целями лабораторной работы являются практическое освоение информации, полученной при изучении курса «Компьютерное моделирование в математической физике» по теме «Численное решение стационарного уравнения Шрёдингера», а также развитие алгоритмического мышления и приобретение опыта использования знаний и навыков по математике, численным методам и программированию для решения прикладных задач физико-технического характера.

## Задачи работы.

**Проблема:** электрон находится в одномерной потенциальной яме с бесконечными стенками  $U(x)$ :

$$v(x) = \begin{cases} J_2(x), & x \in (-L, L); \\ \infty, & x \notin (-L, L), \end{cases}$$

где  $U(x) = v(x) * V_0$ ,  $V_0 = 25$  эВ,  $L = 3$  Å,  $J_n(x)$  — функция Бесселя,  $n = 2$ .

- 1) Рассчитать энергию и волновую функцию основного квантового состояния путем разложения искомой волновой функции по базису. Использовать в качестве базисного набора - волновые функции частицы в одномерной прямоугольной яме с бесконечными стенками.
- 2) Вычислить квантовомеханические средние  $\langle p(x) \rangle$  и  $\langle p(x^2) \rangle$ .
- 3) Сравнить результаты с данными, полученными методом пристрелки.

## 2. Одномерное стационарное уравнение Шрёдингера. Математический формализм. Общие свойства решений

Одномерное стационарное уравнение Шрёдингера[1]

$$\hat{H}\psi(x) = E\psi(x) \quad (1)$$

с математической точки зрения представляет собой задачу определения собственных значений  $E$  и собственных функций  $\psi(x)$  оператора Гамильтона  $\hat{H}$ . Для частицы с массой  $m$ , находящейся в потенциальном поле  $U(x)$ , оператор Гамильтона имеет вид

$$\hat{H} = \hat{T} + U(x), \quad (2)$$

где оператор кинетической энергии

$$\hat{T} = -\frac{\hbar^2}{2m} \frac{d^2}{dx^2}, \quad (3)$$

а  $\hbar$  — постоянная Планка. Собственное значение оператора Гамильтона имеет смысл энергии соответствующей изолированной квантовой системы. Собственные функции называются волновыми функциями. Волновая функция однозначна и непрерывна во всем пространстве. Непрерывность волновой функции и её 1-й производной сохраняется и при обращении  $U(x)$  в  $\infty$  в некоторой области пространства. В такую область частица вообще не может проникнуть, то есть в этой области, а также на её границе,  $\psi(x) = 0$ .

Оценим нижнюю границу энергетического спектра. Пусть минимальное значение потенциальной функции равно  $U_{\min}$ . Очевидно, что  $\langle T \rangle \geq 0$  и  $\langle U \rangle \geq U_{\min}$ . Потому, из уравнения (1) следует, что

$$E = \langle H \rangle = \int_{-\infty}^{\infty} \psi^*(x) \hat{H} \psi(x) dx = \langle T \rangle + \langle U \rangle > U_{\min}. \quad (4)$$

То есть, энергии всех состояний  $> U_{\min}$ .

Особый практический интерес представляет случай, когда

$$\lim_{x \rightarrow \pm\infty} U(x) = 0. \quad (5)$$

Потенциал такого типа называется также потенциальной ямой. Для данной  $U(x)$  свойства решений уравнения Шрёдингера зависят от знака собственного значения  $E$ .

Если  $E < 0$ . Частица с отрицательной энергией совершает финитное движение. Оператор Гамильтона имеет дискретный спектр, то есть собственные значения и соответствующие собственные функции можно снабдить номерами. При  $E < 0$  уравнение (1) приобретает вид [1]:

$$\hat{H}\psi_k(x) = E_k\psi_k(x). \quad (6)$$

Квантовое состояние, обладающее наименьшей энергией, называется основным. Остальные состояния называют возбуждёнными состояниями. В силу линейности стационарного уравнения Шрёдингера, волновые функции математически определены с точностью до постоянного множителя. Однако, из физических соображений, волновые функции должны быть нормированы следующим образом

$$\int_{-\infty}^{+\infty} |\psi_k(x)|^2 dx = 1. \quad (7)$$

В дальнейшем будет рассматриваться только дискретный спектр. При этом необходимо пользоваться осцилляционной теоремой.

**Осцилляционная теорема.** Упорядочим собственные значения оператора Гамильтона в порядке возрастания, нумеруя энергию основного состояния индексом «0»:  $E_0, E_1, E_2, \dots, E_k, \dots$ . Тогда волновая функция  $\psi_k(x)$  будет иметь  $k$  узлов (то есть, пересечений с осью абсцисс). Исключения: области, в которых потенциальная функция бесконечна.

### 3. Прямой вариационный метод. Алгоритм

Прямой вариационный метод также называемый методом Ритца [3] представляет собой численный способ решения уравнения Шрёдингера, который базируется на разложении искомой волновой функции по набору базисных функций. Этот метод применим для нахождения приближённых значений собственных энергий и соответствующих волновых функций.

Для приближённого решения задачи волновая функция  $\psi(x)$  в уравнении (1) представляется в виде разложения [4] по конечному набору ортонормированных базисных функций  $\{\phi_k(x)\}$ :

$$\psi(x) \approx \sum_{k=1}^M c_k \phi_k(x), \quad (8)$$

где  $M$  — число базисных функций,  $c_k$  — коэффициенты разложения, которые необходимо найти.

Коэффициенты  $c_k$  вычисляются из матрицы Гамильтона, где элементы матрицы определяются как:

$$H_{mk} = \int_{-\infty}^{\infty} \phi_m(x) \hat{H} \phi_k(x) dx, \quad (9)$$

Раскладывая оператор Гамильтона (2) получаем:

$$H_{mk} = T_{mk} + U_{mk}, \quad (10)$$

где  $T_{mk}$  — кинетическая энергия, а  $U_{mk}$  — потенциальная энергия:

$$T_{mk} = -\frac{1}{2} \int_{-\infty}^{\infty} \phi_m(x) \frac{d^2}{dx^2} \phi_k(x) dx, \quad (11)$$

$$U_{mk} = \int_{-\infty}^{\infty} \phi_m(x) U(x) \phi_k(x) dx. \quad (12)$$

В качестве базиса выбираются собственные функции прямоугольного потенциала, которые имеют вид:

$$\phi_k(x) = \begin{cases} \frac{1}{\sqrt{L}} \sin\left(\frac{k\pi x}{2L}\right), & \text{если } k \text{ чётное,} \\ \frac{1}{\sqrt{L}} \cos\left(\frac{k\pi x}{2L}\right), & \text{если } k \text{ нечётное.} \end{cases} \quad (13)$$

Эти функции автоматически удовлетворяют граничным условиям  $\phi_k(-L) = \phi_k(L) = 0$ .

В результате поиск собственных значений  $E$  и соответствующих им функций  $\psi(x)$  сводится к вычислению собственных значений и собственных векторов матрицы Гамильтона:

$$H\vec{c} = E\vec{c}. \quad (14)$$

## 4. Программная реализация алгоритма

В Приложении представлена программа на языке Python 3.0 [2], реализованная в среде разработки PyCharm Community Edition 2024.2.1, численного решения одномерного стационарного уравнения Шредингера для электрона в одномерной потенциальной яме. Программа реализует алгоритм прямого вариационного метода, позволяющий находить собственные значения и соответствующие им волновые функции. Энергия и длина потенциальной ямы были переведены в атомные единицы Хартри (строки 105-109)

В строках 7-8 определена потенциальная функция.

В строках 11-17 реализована функция вычисляющая базисную волновую функцию  $k$ -го состояния

В строках 30-37 реализована функция вычисляющая матричный элемент по формулам (10, 11, 12), функция реализованная в строках 19-28 является вспомогательной и вычисляет вторую производную для заданной функции.

В строках 44-49 реализовано построение матрицы Гамильтона.

В строках 51-53 реализована функция вычисляющая собственные значения и собственные вектора заданной матрицы.

В строках 55-59 реализована функция вычисляющая волновую функцию по формуле (8).

В строках 61-73 реализованы функции вычисляющие квантовомеханические средние  $\langle p(x) \rangle$ ,  $\langle p(x^2) \rangle$

В строках 75-98 реализована функция выводящая графики волновых функций.

В строках 112-113 задаются размерность сетки и матрицы Гамильтона.

В строках 118-123 вычисляются энергии и волновые функции, и результат записывается в файл.



## 5. Результаты численных экспериментов и их анализ

Волновая функция основного состояния и ее плотность вероятности на на Рис. 1.

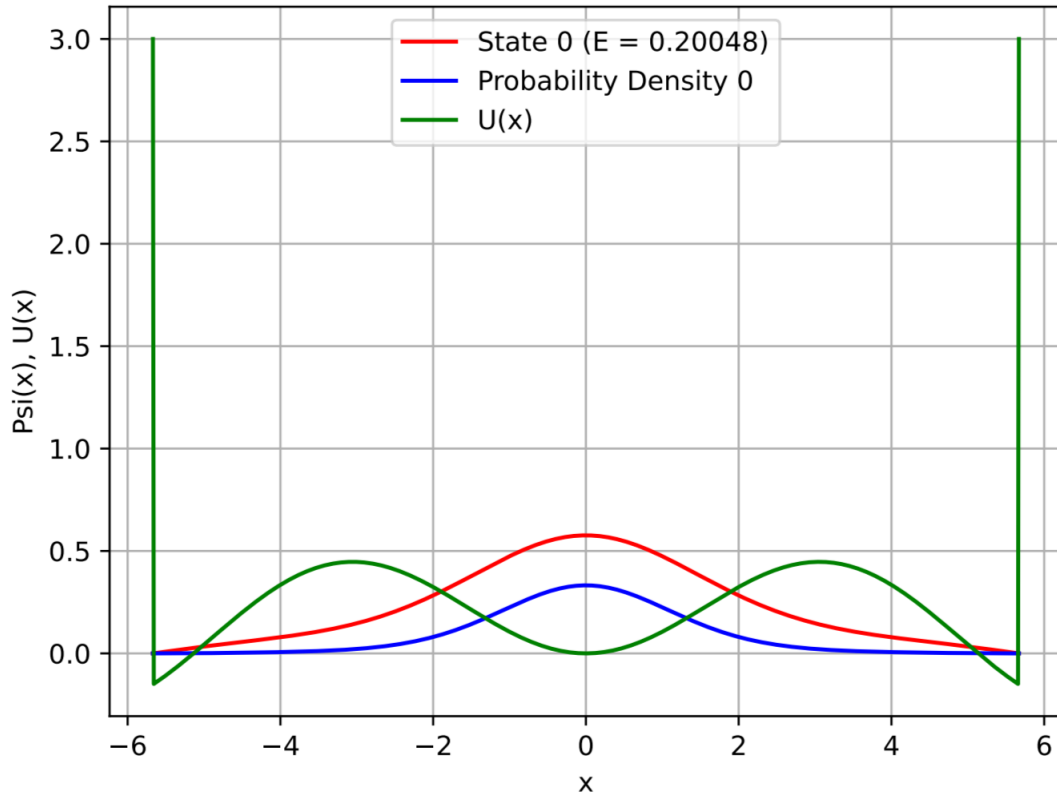


Рис. 1. Волновая функция основного состояния

На Рис. 1 можно видеть, что волновая функция согласно осцилляционной теореме соответствует основному состоянию.

На Рис. 2 представлено сравнение волновых функций основного состояния вычисленных методом пристрелки и методом Ритца.

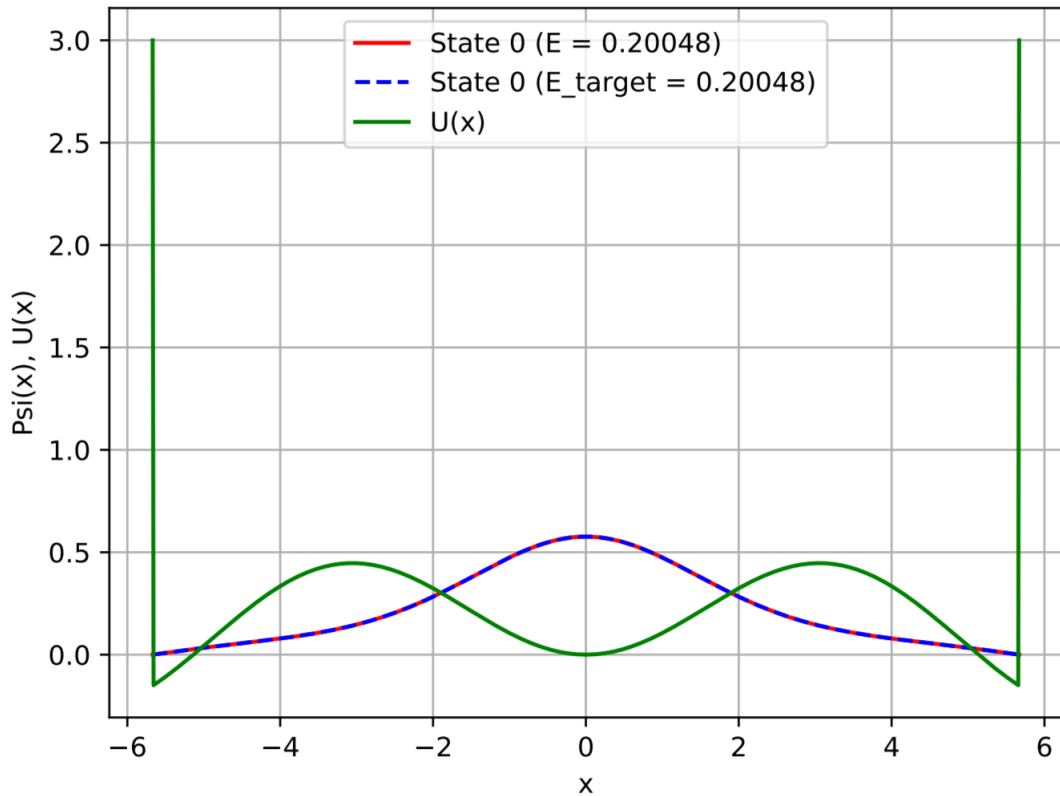


Рис. 2. Волновая функция основного состояния, сравнение методов

Здесь красная линия соответствует функции вычисленной методом Ритца, а синяя линия методом пристрелки. Можно увидеть, что точность методов совпадает.

На Рис. 3 представлены численные значения энергий вычисленных методом Ритца и методом пристрелки, а также квантовомеханические средние  $\langle p(x) \rangle$  и  $\langle p(x^2) \rangle$ :

```

=====
State 0: E = 0.200480, <p_x> = 0.000000e+00, <p_x^2> = 1.551882e-01
State 0: E_target = 0.200481, <p_x> = 0.000000e+00, <p_x^2> = 1.552094e-01
=====
State 1: E = 0.476679, <p_x> = 0.000000e+00, <p_x^2> = 3.352033e-01
State 1: E_target = 0.476680, <p_x> = 0.000000e+00, <p_x^2> = 3.352446e-01
=====
State 2: E = 0.614085, <p_x> = 0.000000e+00, <p_x^2> = 6.514036e-01
State 2: E_target = 0.614087, <p_x> = 0.000000e+00, <p_x^2> = 6.515212e-01
=====

```

Рис. 3. Сравнение энергий

Здесь  $E$  это энергия вычисленная методом Ритца, а  $E_{\text{target}}$  вычисленная методом пристрелки.

### **Заключение.**

Таким образом, было получено численное решение для задачи о частице в одномерной квантовой яме с бесконечными стенками при помощи метода Ритца. Были получены значение энергии и волновая функция основного состояний, а также вычислены квантовомеханические средние  $\langle p(x) \rangle$ ,  $\langle p(x^2) \rangle$ . Полученные волновые функции соответствуют осцилляционной теореме. Сравнивая результаты полученные методом Ритца с методом пристрелки можно сказать, что методы сходятся в точности.

## Приложение 1. Компьютерный код

```
1      import numpy as np
2      from scipy.linalg import eigh
3      import matplotlib.pyplot as plt
4      from scipy.special import jn
5
6
7      def U(x):
8          return V0 * jn(2, x) if abs(x) < L else W
9
10
11     def basis_function(k):
12         result = np.zeros(N)
13         h = (2 * L) / (N - 1)
14         for i in range(N):
15             arg = (np.pi * (k + 1) * (-L + i * h)) / (2 * L)
16             result[i] = np.sin(arg) / np.sqrt(L) if (k + 1) % 2 == 0 else np
17                 .cos(arg) / np.sqrt(L)
18         return result
19
20     def second_deriv(y, h):
21         deriv = np.zeros_like(y)
22         for i in range(len(y)):
23             if i == 0:
24                 deriv[i] = (2 * y[i] - 5 * y[i + 1] + 4 * y[i + 2] - y[i +
25                     3]) / (h * h)
26             elif i == len(y) - 1:
27                 deriv[i] = (-y[i - 3] + 4 * y[i - 2] - 5 * y[i - 1] + 2 * y[i
28                     ]) / (h * h)
29             else:
30                 deriv[i] = (y[i - 1] - 2 * y[i] + y[i + 1]) / (h * h)
31         return deriv
32
33     def h_psi(k):
34         psi_k = basis_function(k)
35         result = np.zeros(N)
36         h = (2 * L) / (N - 1)
37         deriv_psi = second_deriv(psi_k, h)
38         for i in range(N):
39             result[i] = deriv_psi[i] / (-2) + U(-L + i * h) * psi_k[i]
40         return result
41
42     def hamiltonian_element(m, k):
```

```

40         fi_m = basis_function(m)
41         h_fi_k = h_psi(k)
42         return np.trapz(fi_m * h_fi_k, dx=(2 * L) / (N - 1))
43
44     def hamiltonian_matrix():
45         h_matrix = np.zeros((M, M))
46         for i in range(M):
47             for j in range(M):
48                 h_matrix[i, j] = hamiltonian_element(i, j)
49         return h_matrix
50
51     def eigen_solve(h_matrix):
52         eigenvalues, eigenvectors = eigh(h_matrix)
53         return eigenvalues, eigenvectors
54
55     def compute_wave_function(coef):
56         result = np.zeros(N)
57         for k in range(M):
58             result += coef[k] * basis_function(k)
59         return result
60
61     def mean_momentum(Psi, X):
62         hbar = 1.0
63         dPsi_dx = np.gradient(Psi, X)
64         integrand = Psi.conj() * dPsi_dx
65         mean_Px = -1j * hbar * np.trapz(integrand, X)
66         return mean_Px.real
67
68     def mean_square_momentum(Psi, X):
69         hbar = 1.0
70         d2Psi_dx2 = np.gradient(np.gradient(Psi, X), X)
71         integrand = Psi.conj() * d2Psi_dx2
72         mean_Px2 = -hbar**2 * np.trapz(integrand, X)
73         return mean_Px2.real
74
75     def plot_wave_functions(energies, wave_functions):
76         x_vals = np.linspace(-L, L, N)
77         potential = np.array([U(x) for x in x_vals])
78         for i, psi in enumerate(wave_functions):
79             mean_P = mean_momentum(psi.copy(), x_vals)
80             mean_P2 = mean_square_momentum(psi.copy(), x_vals)
81             density_psi = psi ** 2
82
83             plt.plot(x_vals, psi, 'r', label=f"State {i} (E = {energies[i]

```

```

84         plt.plot(x_vals, density_psi, 'b', label=f"Probability Density {
            i}")
85     plt.plot(x_vals, potential, 'g', label="U(x)")
86     plt.xlabel("x")
87     plt.ylabel("Psi(x), U(x)")
88     plt.legend()
89     plt.grid()
90     plt.savefig(f"State {i} probability density.pdf", dpi=300)
91     plt.show()
92
93     print("=====")
94     print("=====",
          file=file1)
95     print(f"State {i}: E = {energies[i]:.6f}, <p_x> = {mean_Px:.6e},
          <p_x^2> = {mean_Px2:.6e}")
96     print(f"State {i}: E = {energies[i]:.6f}, <p_x> = {mean_Px:.6e},
          <p_x^2> = {mean_Px2:.6e}", file=file1)
97     print("=====")
98     print("=====",
          file=file1)
99
100     def normalize_wave_function(psi):
101         dx = (2 * L) / (N - 1)
102         norm = np.sqrt(np.trapz(psi**2, dx=dx))
103         return psi / norm
104
105     clength = 0.5292
106     cenergy = 27.212
107
108     L = 3.0/clength
109     V0 = 25.0/cenergy
110     W = 3.0
111
112     N = 1001
113     M = 21
114
115     h_matrix = hamiltonian_matrix()
116     energies, eigenvectors = eigen_solve(h_matrix)
117
118     wave_functions = [compute_wave_function(eigenvectors[:, i]) for i in
                       range(3)]
119     normalized_wave_functions = [normalize_wave_function(psi) for psi in
                                  wave_functions]
120
121     file1 = open("result.txt", "a")

```

122

123 `plot_wave_functions(energies, normalized_wave_functions)`

## Список литературы

1. Тимошенко Ю.К. Численное решение стационарного уравнения Шредингера: метод пристрелки. Учебное пособие. Воронеж: Научная книга, 2019. 35 с.
2. Доля П.Г. Введение в научный Python. Харьков: ХНУ, 2016. 265 с.
3. Давыдов А. С. Квантовая механика. СПб.: БХВ-Петербург, 2011. 704 с
4. Тимошенко Ю. К. Лекционный материал. 2020.