Санкт-Петербургский национальный исследовательскийуниверситет
информационных технологий, механикии оптики
УЧЕБНЫЙ ЦЕНТР ОБЩЕЙ ФИЗИКИ ФТФ

УНИВЕРСИТЕТ ИТМО

Группа М32011 К работе допущен

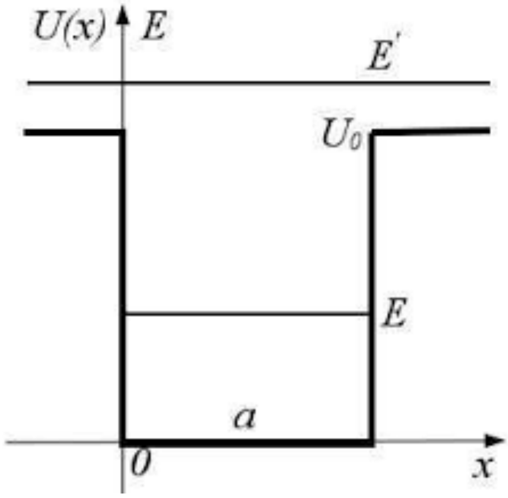
Студент Мирошниченко Александр Работа выполнена

Преподаватель Зинчик Александр Адольфович Отчет принят

Рабочий протокол и отчет моделированию № 1

Поиск связных состояний в случае прямоугольной потенциальной ямы

1. Теория



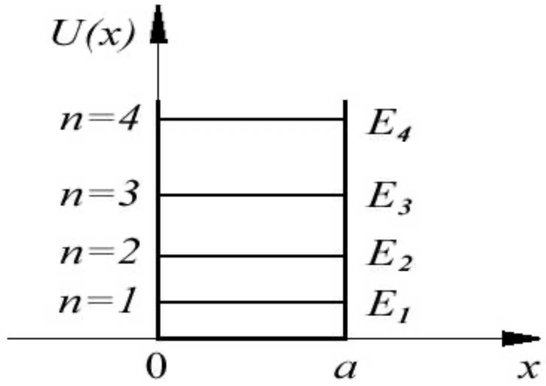
Уравнение Шредингера и потенциальная яма используются для представления квантовой механики частицы массы, ограниченной окружающим постоянным потенциалом.

Для стационарных состояний оно выглядит так:

$$(\frac{d^2}{dx^2} + \frac{2m}{\hbar^2}\left(E - V(x)\right)\psi(x) = 0$$

По условию, потенциал ограничен снизу $-U$, следовательно, спектр энергий ограничен снизу $E \ \geq \ -U$.

Дискретный спектр может быть только при отрицательных энергиях. При положительных энергиях может быть только сплошной спектр. Потенциал является четной функцией x, а, следовательно, мы можем считать, что собственные функции уравнения Шредингера являются либо четными, либо нечетными.



Введём параметр Q:

$$Q = \sqrt{\frac{2\mathrm{ma}^2}{\hbar^2}\mathrm{U}}$$

Откуда:

$$\alpha = \sqrt{\frac{Q^2}{a^2} - k^2},$$

$$E = -\frac{\hbar^2}{2ma}\sqrt{Q^2 - k^2a^2}$$

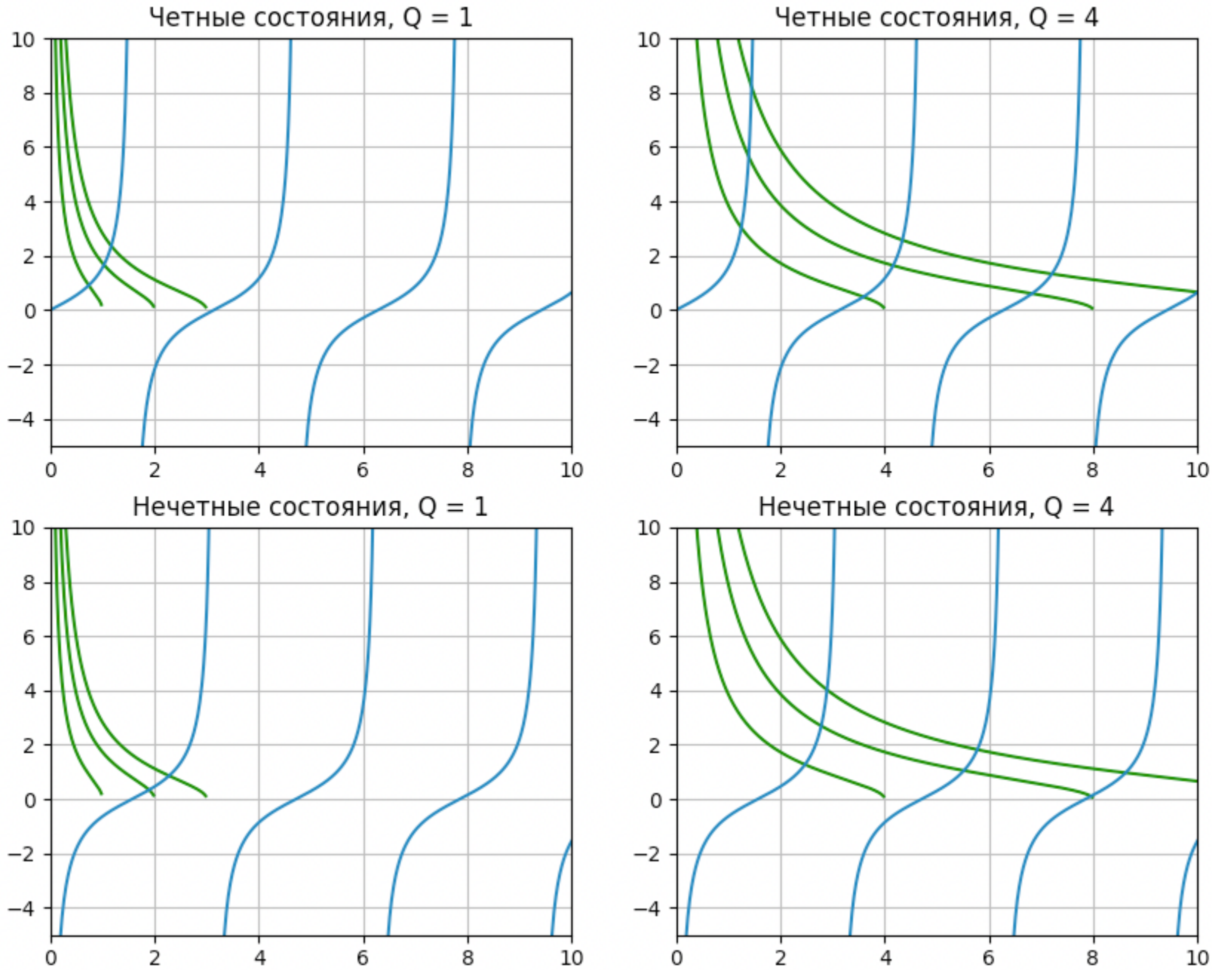
Перепишем уравнения для чётных и нечётных состояний:

$$\begin{cases} tg(ka) = \dfrac{\sqrt{Q^2 - k^2a^2}}{ka} - \text{четные}(1) \\ ctg(ka) = -\dfrac{\sqrt{Q^2 - k^2a^2}}{ka} - \text{нечетные}(2) \end{cases}$$

Решение уравнений будет найдено графически:

- Корни – абсцисса точек пересечения функции $\frac{\sqrt{(Q^2 - k^2a^2)}}{ka}$ для четных и нечетных решений соответственно.

На рисунках ниже изображены кривые, представляющие собой левые и правые части уравнений для четных и нечетных состояний. Видно, что число точек пересечения растет с ростом параметра Q.

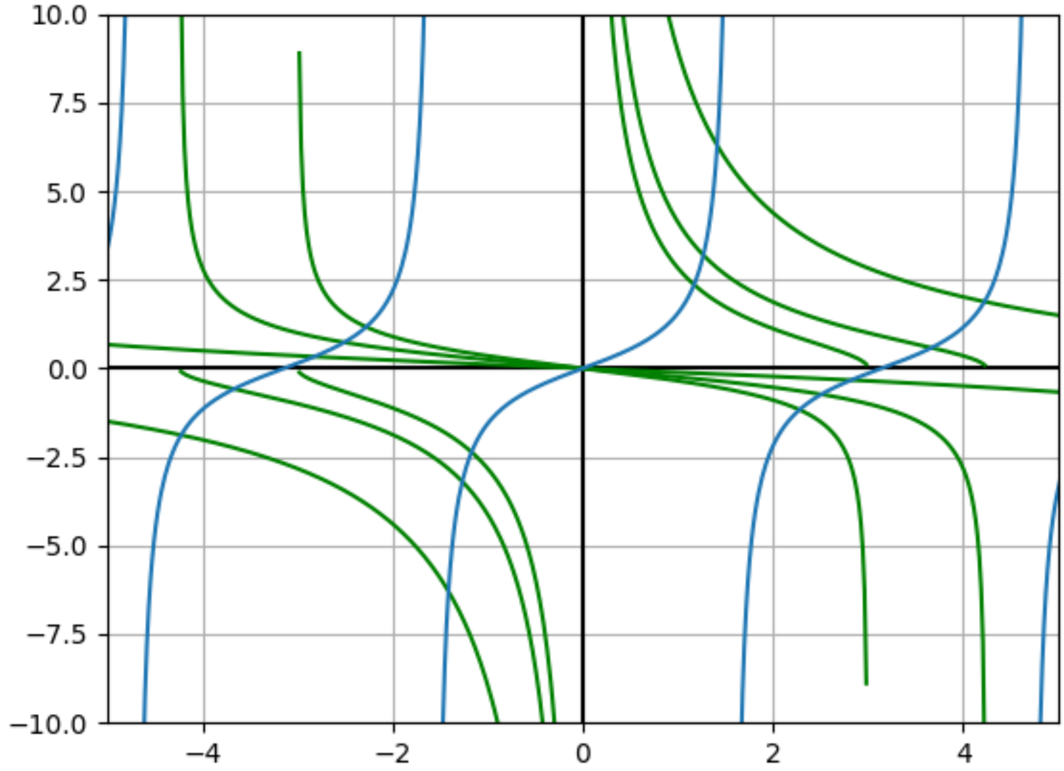


Видно, что при малых значениях параметра Q имеется минимум одно пересечение графиков для четных решений и может не быть ни одного пересечения для нечетных.
Это означает, что в любой яме имеется по крайней мере одно связанное четное состояние.
Рассмотрим число связанных состояний при разных Q:

	Q = 1	Q = 4
Четные состояния	Одно состояние при $ka \approx 0.8$	Два состояния при $k_1 a \approx 1.25, k_2 a \approx 1.4$
Нечетные состояния	Ни одного состояния	Одно состояние при $ka \approx 2.45$

Величина Q является постоянной, зависящей лишь от размеров ямы ($Q^2 \sim Ua^2$)
Ниже на рисунке изображено графическое решение уравнений (1) и (2). Кривые с положительными ординатами относятся к четным состояниям, кривые

отрицательными ординатами — к нечетным состояниям.



Собственные значения находятся как абсциссы точек пересечения двух кривых с тангенсоидой. Кривые (1), (2), зависят от Q, определяемого размерами ямы.
С дальнейшим увеличением размеров ямы растет число связанных состояний растет. В примере были использованы следующие значения Q – [1, 2, 9]

Можно сделать несколько выводов:

- В яме любой ширины и глубины есть по крайней мере одно четное связанное состояние.
- Наименьшей энергией обладает четное состояние
- Значения энергий четных и нечетных состояний чередуются, т.е. уровень энергии нечетного состояния следует за уровнем энергии четного состояния, и наоборот.

Осциляторный потенциал – частица, с потенциальной энергией:

$$V(x) = U(x) = \frac{mw^2x^2}{2} = \frac{kx^2}{2},$$
$$k = mw^2$$

Так как частица движется только вдоль оси x:

$$\frac{d^2\psi}{dx^2} + \frac{2}{\hbar}\left(E - \frac{kx^2}{2}\right)\psi = 0$$

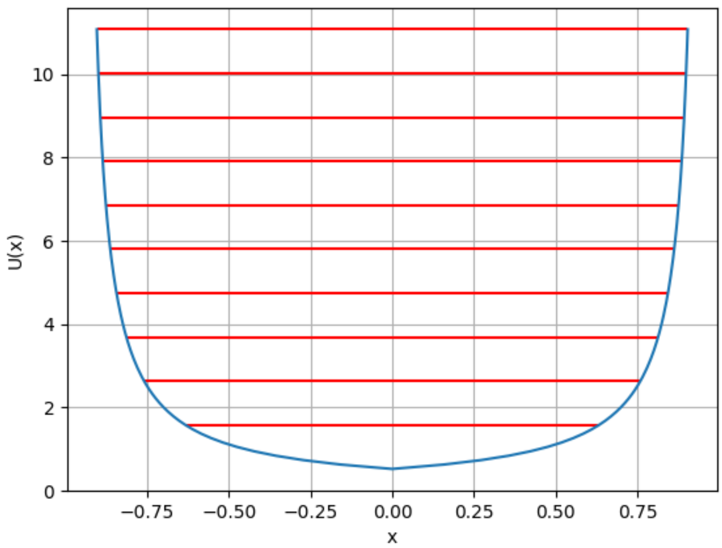
Решением такого уравнения будет $E_n = \left(n + \frac{1}{2}\right)\hbar w$, где n – квантовые числа.

Наименьшее решение - $E_{min} = \frac{\hbar w}{2}$ -> для таких частиц всегда существует собственная E, которая определяется только собственной частотой:

n	***E***
1	$E_1 = \frac{3\hbar w}{2}$
2	$E_1 = \frac{5\hbar w}{2}$

$$\Delta E = E_{n+1} - E_n = \hbar w$$

Можно сделать вывод, что уровни равномерны распределены друг от друга:



Полупроницаемую перегородку, разделяющую всю область на две равные части, можно получить как предельный случай барьера конечной ширины 2ε (между точками $-\varepsilon$ и ε) и конечной высоты U_0.

Кроме двух граничных условий $u(\pm a) = 0$ благодаря наличию барьера мы имеем еще четыре граничных условия, так как функции $u(x)$ и $u'(x)$ должны быть непрерывны в точках $x = \pm\varepsilon$:

$$u = \begin{cases} A_1 \sin k(x+a), & -a \leqslant x \leqslant -\varepsilon, \\ Be^{-\varkappa x} + Ce^{\varkappa x}, & -\varepsilon \leqslant x \leqslant \varepsilon, \\ A_2 \sin k(x-a), & \varepsilon \leqslant x \leqslant a. \end{cases}$$

Требование непрерывности при $x = \pm \varepsilon$ теперь дает

$$
\begin{aligned}
u(-\varepsilon) &= A_1 \sin k\ (a-\varepsilon) = Be^{\varkappa\varepsilon} + Ce^{-\varkappa\varepsilon},\\
u'(-\varepsilon) &= kA_1 \cos k\,(a-\varepsilon) = \varkappa\,(-Be^{\varkappa\varepsilon} + Ce^{-\varkappa\varepsilon}),\\
u(+\varepsilon) &= A_2 \sin k\ (\varepsilon-a) = Be^{-\varkappa\varepsilon} + Ce^{\varkappa\varepsilon},\\
u'(+\varepsilon) &= kA_2 \cos k\ (\varepsilon-a) = \varkappa\,(-Be^{-\varkappa\varepsilon} + Ce^{\varkappa\varepsilon}).
\end{aligned}
$$

Имеем:

$$
k \operatorname{ctg} k\,(a-\varepsilon) = \varkappa\,\frac{-Be^{2\varkappa\varepsilon}+C}{Be^{2\varkappa\varepsilon}+C},
$$

$$
k \operatorname{ctg} k\,(a-\varepsilon) = \varkappa\,\frac{B-Ce^{2\varkappa\varepsilon}}{B+Ce^{2\varkappa\varepsilon}}.
$$

Из тождественности их левых частей следует равенство правых частей. Последние же равны в том и только в том случае, если $B = \pm C$. При $B = C$ мы получаем решения с положительной четностью, если же $B = -C$, то мы получаем решения с отрицательной четностью. Следовательно, стационарные состояния разделяются на два класса, характеризующиеся различными четностями.

Теперь перейдем к пределу $\varepsilon \to 0$, $\chi \to \infty$ так, чтобы $\varepsilon\chi \to \infty$, но величина $\chi^2\varepsilon = \Omega$ оставалась конечной

Имеем $\Omega = 1/2, \Omega -$ коэффициент непроницаемости перегородки

Input

$$\left\{k \cot(k) = -\frac{1}{2},\ 0 < k < 5\right\}$$

Solutions

$k \approx 1.8366 + 0\,i$

$k \approx 4.81584 + 0\,i$

Собственные функции обращаются на перегородке в 0, так что решение имеет вид:

$$u_n^-(x) = \begin{cases} A \sin k_n^- (x + a), & -a \le x < 0 \\ A \sin k_n^- (x - a), & 0 < x \le a \end{cases}$$

$$k_n^- a = n\pi, \qquad n = 1,2,3, \dots$$

$$u_n^-(-x) = -u_n^-(x), \qquad |A|^2 = \frac{1}{a}$$

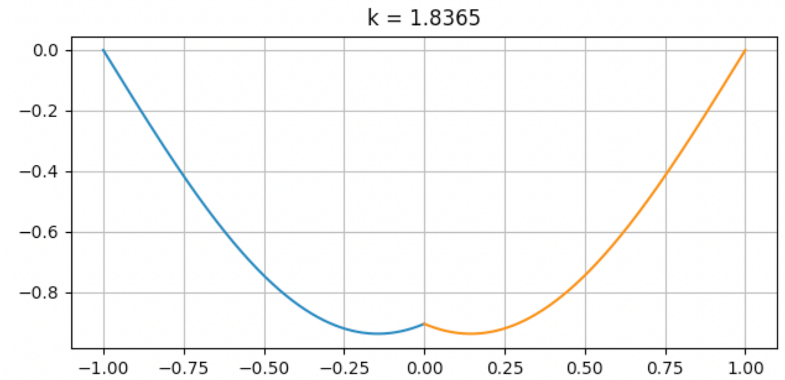
Перезапишем собственные функции:

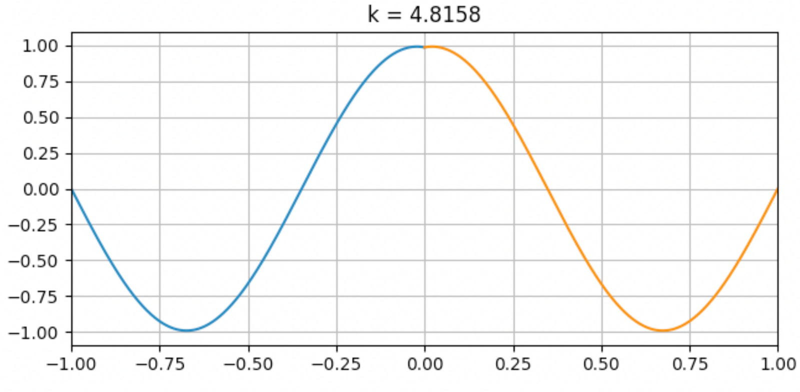
$$u_n^+(x) = \begin{cases} -A \sin k_n^+ (x + a), & -a \leqslant x < 0, \\ +A \sin k_n^+ (x - a), & 0 < x \leqslant +a, \end{cases}$$

$$\left(n - \frac{1}{2}\right)\pi < k_n^+ a < n\pi, \quad n = 1, 2, 3, \dots,$$

$$u_n^+(-x) = u_n^+(x), \quad |A|^2 = \frac{2k_n^+}{2k_n^+ a - \sin 2k_n^+ a}.$$

Их значения при $x = 0$ конечны, а графики имеют изломы





Чем не прозрачней становится перегородка, тем выше поднимаются уровни с положительной четностью; уровни с отрицательной четностью остаются на прежнем месте

2. Код

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

```
##########################################
u = [1, 4, 9]
Q_small = 1
Q_big = 4

delta = 10
t = np.linspace(-2 * math.pi, 4 * math.pi, 1000)

fig, ax = plt.subplots(2, 2)

ax[0, 0].set_title(f'Четные состояния, Q = {Q_small}')
ax[0, 0].set_xlim(0, delta)
ax[0, 0].set_ylim(-delta / 2, delta)
ax[0, 0].grid()

ax[1, 0].set_title(f'Нечетные состояния, Q = {Q_small}')
ax[1, 0].set_xlim(0, delta)
ax[1, 0].set_ylim(-delta / 2, delta)
ax[1, 0].grid()

ax[0, 1].set_title(f'Четные состояния, Q = {Q_big}')
ax[0, 1].set_xlim(0, delta)
ax[0, 1].set_ylim(-delta / 2, delta)
ax[0, 1].grid()

ax[1, 1].set_title(f'Нечетные состояния, Q = {Q_big}')
ax[1, 1].set_xlim(0, delta)
ax[1, 1].set_ylim(-delta / 2, delta)
ax[1, 1].grid()

tan_values = np.tan(t)
cot_values = -1 / np.tan(t)

y2_00 = y2_10 = y2_01 = y2_11 = []

for i in u:
    y2_00 = np.sqrt(i * Q_small ** 2 - t ** 2) / t
    ax[0, 0].plot(t, y2_00, 'g')

    y2_10 = np.sqrt(i * Q_small ** 2 - t ** 2) / t
    ax[1, 0].plot(t, y2_10, 'g')

    y2_01 = np.sqrt(i * Q_big ** 2 - t ** 2) / t
    ax[0, 1].plot(t, y2_01, 'g')

    y2_11 = np.sqrt(i * Q_big ** 2 - t ** 2) / t
    ax[1, 1].plot(t, y2_11, 'g')

tan_values[:-1][np.diff(tan_values) < 0] = np.nan
cot_values[:-1][np.diff(cot_values) < 0] = np.nan

ax[0, 0].plot(t, tan_values)
ax[1, 0].plot(t, cot_values)
ax[0, 1].plot(t, tan_values)
ax[1, 1].plot(t, cot_values)
plt.figure(1)
##########################################
delta = 10
plt.figure(2)
Q = 3
u = [1, 2, 9]
odd_values = []
even_values = []

t = np.linspace(-2 * math.pi, 4 * math.pi, 1000)
tan_values = np.tan(t)
tan_values[:-1][np.diff(tan_values) < 0] = np.nan
```

```python
plt.xlim(-delta / 2, delta / 2)
plt.ylim(-delta, delta)
plt.axvline(x=0, color='k')
plt.axhline(y=0, color='k')

for i in u:
    odd_values = np.sqrt(i * Q ** 2 - t ** 2) / t
    even_values = - (t / np.sqrt(i * Q ** 2 - t ** 2))

    plt.plot(t, odd_values, 'g')
    plt.plot(t, even_values, 'g')

plt.plot(t, tan_values)
plt.grid()
###########################################
plt.figure(3)

h = 1.0545726
w = 1
n = 10

x = []
y = []

for i in range(-n, n + 1):
    temp = (abs(i) + 0.5) * h * w
    y.append(temp)
    x.append(i / temp)

for i in range(n + 1):
    plt.hlines(y[i], x[i], -x[i], colors='red')

x = []
y = []
step = 0.1

for i in range(-n, n):
    start = i
    end = i + 1
    while start < end:
        temp = (abs(start) + 0.5) * h * w
        y.append(temp)
        x.append(start / temp)
        start += step

plt.plot(x, y)
ax = plt.gca()
ax.set_xlabel('x')
ax.set_ylabel('U(x)')
plt.grid()
###########################################
fig, ax = plt.subplots(2)

x1 = np.linspace(-1, 0, 1000)
x2 = np.linspace(0, 1, 1000)
k_values = [1.8365, 4.8158]
a = 1

for i in range(0, len(k_values)):
    y1 = -np.sqrt(2 * k_values[i] / (2 * k_values[i] * a - np.sin(2 * k_values[i] *
a))) * \
         np.sin(k_values[i] * (x1 + a))
    y2 = np.sqrt(2 * k_values[i] / (2 * k_values[i] * a - np.sin(2 * k_values[i] *
a))) * np.sin(k_values[i] * (x2 - a))
    ax[i].plot(x1, y1)
    ax[i].plot(x2, y2)
    ax[i].set_title(f'k = {k_values[i]}')
```

```
    plt.xlim(-1, 1)
    ax[i].grid()

plt.figure(4)
plt.show()
```