Федеральное государственное автономное
образовательное учреждение высшего образования
"Национальный исследовательский университет ИТМО"

УНИВЕРСИТЕТ ИТМО

Дополнительные главы физики
Задание после лекции 13.02.2023
"Основы квантовой физики"

Выполнил:
Лопатенко Г. В., М32021

Преподаватель:
Музыченко Я. Б.

Февраль, 2023

Содержание

1 Основы квантовой теории

Проанализировать формулы Рэлея-Джинса, Вина и Планка.
а) построить графики зависимости испускательной способности от длины волны (от 20 до 2000 нм, шаг 20 нм) для двух лампочек с температурами 2700 К и 3300 К;
б) определить для ламп относительное количество энергии, приходящейся на видимый спектр.

1.1 Построение графиков основных законов

В 1900 г. Рэлей и Джинс, исходя из термодинамической гипотезы о равнораспределении энергии по степеням свободы, получили формулу для излучательной способности:

$$r_w = \frac{\omega^2}{4\pi^2 c^2} kT$$

В теории на лекции подробно разбиралось тождество:

$$r_\lambda = \frac{\omega}{\lambda} \cdot r_w$$

Тогда становится очевидной формула Рэлея-Джинса в длинах волн:

$$r_{\lambda\, Rayleigh} = \frac{2\pi c}{\lambda^4} kT$$

Кстати, формула Планка является надстройкой над этим фактом и предельно сходится на бесконечности:

$$r_{\lambda\, Planck} = \frac{2\pi h c^2}{\lambda^5} \cdot \left(exp\left(\frac{hc}{\lambda kT}\right) - 1 \right)^{-1}$$

А господин Вин в 1896 г. предложил свою аппроксимацию со сдвигом, основываясь на распределении Больцмана:

$$r_w = \frac{h\omega^3}{8\pi^3 c^2} \cdot exp\left(-\frac{h\omega}{2\pi kT}\right)$$

То есть, опять же переходя на длины волн:

$$r_{\lambda\, Wein} = \frac{2\pi h c^2}{\lambda^5} \cdot exp\left(-\frac{hc}{\lambda kT}\right)$$

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

def eval_R_Planck(lambd, temperature):
    return 2 * np.pi * h * c ** 2 / lambd ** 5 / (np.e ** (h * c / (k * temperature * lambd)) - 1)

def eval_R_RayleighJeans(lambd, temperature):
    return 2 * np.pi * c * k * temperature / lambd ** 4

def eval_R_Wien(lambd, temperature):
    return 2 * np.pi * h * c ** 2 / lambd ** 5 / np.e ** (h * c / (k * temperature * lambd))

def plot_scatter(ax, arguments, temperature, mode, color, f):
    ax.scatter(x=arguments,
               y=f(arguments, temperature),
               label=f'{mode} approximation',
               c=color,
               s=10)

h, c, k = 6.626 * 10 ** -34, 299792458, 1.381 * 10 ** -23
T1, T2 = 2700, 3300
arguments = np.linspace(20 * 10 ** -9, 2500 * 10 ** -9, 300)
fig, ax = plt.subplots(1, 1, figsize=(12, 8))
plot_scatter(ax, arguments, T1,
             "Planck, 2700 K", "red",
             eval_R_Planck)
plot_scatter(ax, arguments[250:], T1,
             "RayleighJeans, 2700 K", "blue",
             eval_R_RayleighJeans)
plot_scatter(ax, arguments, T1,
             "Wein, 2700 K", "green",
             eval_R_Wien)
ax.legend()
ax.grid(axis='both')
fig.tight_layout()
```

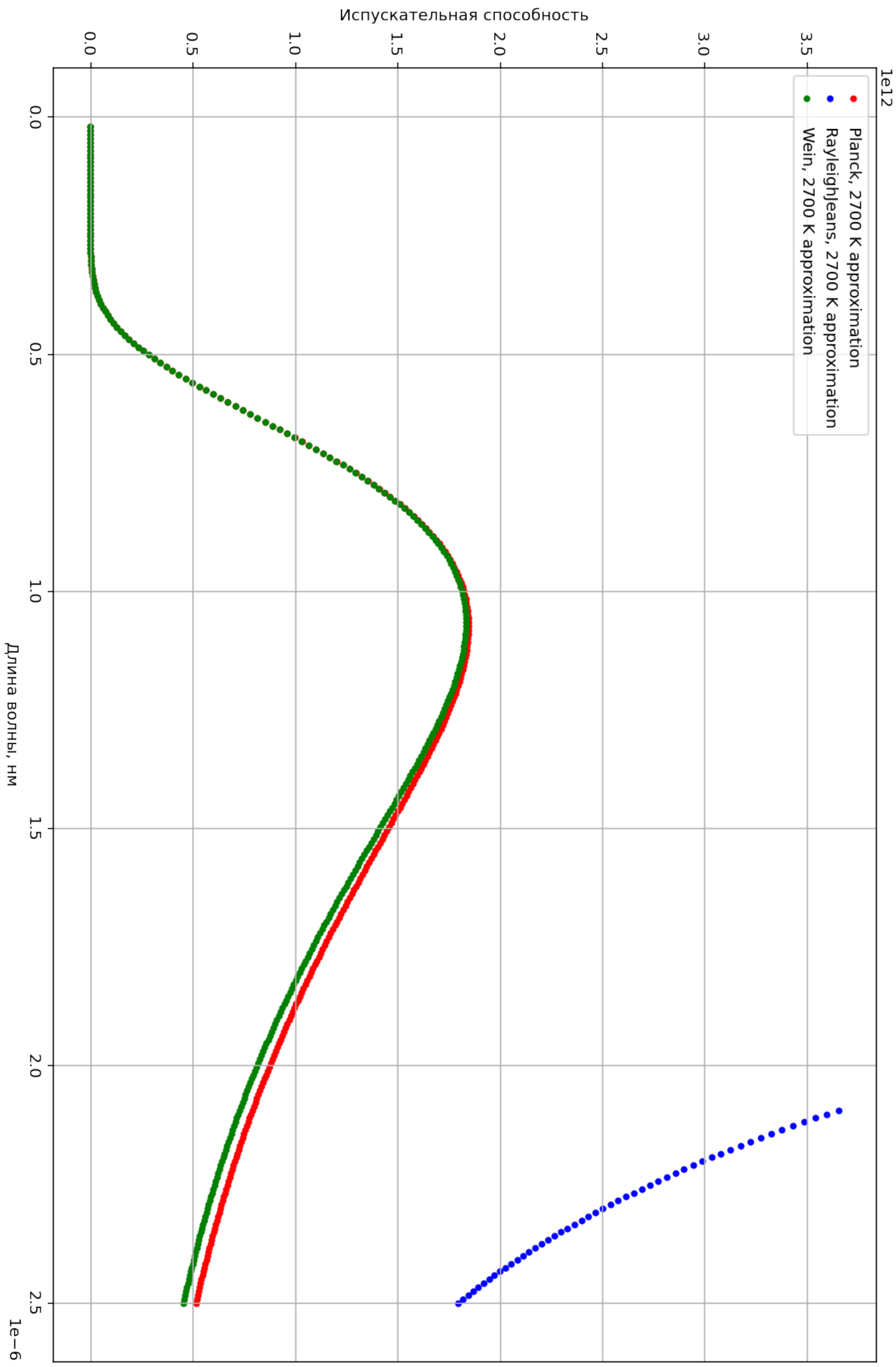

Рис. 1: Графики r_λ при температуре 2700 K

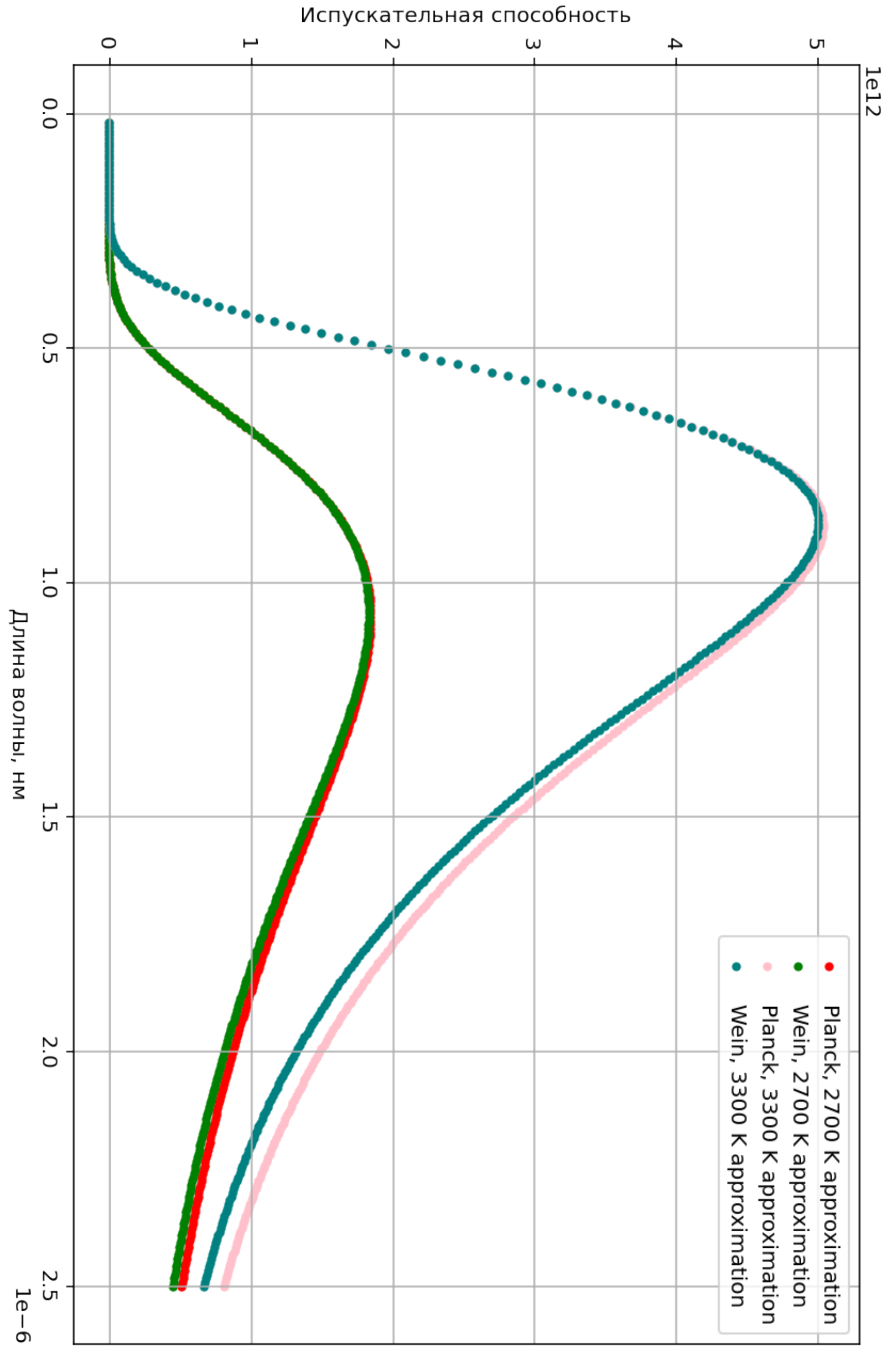

Рис. 2: Графики r_λ при температурах 2700 K и 3300 K

1.2 Относительное количество энергии

Выберем излучатель при температуре $T = 2700K$. Тогда для расчета относительной энергии, приходящейся на видимый спектр, необходимо определить долю площади под кривой зависимости испускательной способности от длины волны на выбранном спектре.

$$\frac{W_0}{W} = \frac{\int\limits_{380nm}^{780nm} \frac{2\pi hc^2}{\lambda^5} \cdot \left(exp\left(\frac{hc}{\lambda kT}\right) - 1\right)^{-1} d\lambda}{\int\limits_{20nm}^{2000nm} \frac{2\pi hc^2}{\lambda^5} \cdot \left(exp\left(\frac{hc}{\lambda kT}\right) - 1\right)^{-1} d\lambda}$$

Конечно, можно посчитать аналитически и взять отношение значения интегралов от известной функции для испускательной способности, но гораздо проще воспользоваться численными методами: в представленном ниже листинге приведен расчет этих интегралов.

```
full_linspace = np.linspace(20 * 10 ** -9, 2000 * 10 ** -9,
    10000)
visible_linpace = np.linspace(380 * 10 ** -9, 780 * 10 ** -9,
    10000)
fstep = full_linspace[1] - full_linspace[0]
vstep = visible_linpace[1] - visible_linpace[0]
T1 = 2700
sum_visible = sum([eval_R_Planck(x, T1) * vstep for x in
    visible_linpace])
sum_full = sum([eval_R_Planck(x, T1) * fstep for x in
    full_linspace])
print(f'Relative energy in visible spectre: {100 * sum_visible
    / sum_full} %')
```

Получили ответ:

```
>>> Relative energy in visible spectre: 12.307838947590378 %
```

Ответ: 12,3%

2 Классика в квантах

По классическим представлениям электрон должен накопить энергию для вылета с катода. Лампой мощностью 100 Вт освещается алюминиевый катод (работа выхода 4,2 эВ). Лампа находится на расстоянии 1 м от катода. Оценить время, необходимое для накопления энергии, большей работы выхода.

Начнем с тривиального: явление фотоэффекта невозможно полно объяснить с точки зрения классической теории электромагнитного излучения. Заметим, что хотя бы на преодоление первой порцией фотонов (хотя мы и не говорим о квантовании) расстояния между излучателем и катодом уйдет время $t_0 = \frac{L}{c}$, при этом предполагается, что фотоны несут энергию (опять навязывается понятие кванта) и не теряют ее на рассеивание. Оценка очень грубая, но при постоянном неупругом соударении фотонов и валентных электронов последние, накопив достаточную энергию, способны покинуть свою орбиту и вылететь из материала, создавая при этом фототок. Тогда так называемое запаздывание будет включать в себя еще и время, которое электрон копит энергию, большую работы выхода, то есть $t_{\text{вых}} = \frac{A_{\text{вых}}}{P}$.

$$t = \frac{L}{c} + \frac{A_{\text{вых}}}{P}$$

Ответ: $t = \frac{1}{299792458} + \frac{4{,}2 \cdot 1{,}6 \cdot 10^{-19}}{100} \approx 3,3356\, ns$

3 Potassium FUN

Potassium has one of the lowest work functions of all metals and so is useful in photoelectric devices using visible light. Light from a source is incident on a potassium surface. Data for the stopping voltage V_0 as a function of wavelength λ is shown below. (*a*) Explain why a graph of V_0 vs. $\frac{1}{\lambda}$ is expected to yield a straight line. What are the theoretical expectations for the slope and y-intercept of this line? (*b*) Using the data below, graph V_0 vs. $\frac{1}{\lambda}$ and show that a straight-line plot does indeed result. Determine the slope a and y-intercept b of this line. Using your values for a and b, determine (*c*) potassium's work function and (*d*) Planck's constant h.

λ (μm)	0.400	0.430	0.460	0.490	0.520
V_0 (V)	0.803	0.578	0.402	0.229	0.083

3.1 Пояснение теоретических аспектов зависимости

Запишем основное тождество фотоэффекта – уравнение Эйнштейна:

$$h\nu = A_{\text{вых}} + \frac{mv_m^2}{2} = A_{\text{вых}} + eU_{\text{з}}$$

Соответсвенно, выразим напряжение запирания:

$$U_{\text{з}} = \frac{h\nu - A_{\text{вых}}}{e} = \frac{hc}{e\lambda} - \frac{A_{\text{вых}}}{e}$$

Линейный характер зависимости напряжения запирания от обратной длины волны во II законе Столетова можно толковать следующим образом: фотоэффект наблюдается с некоторой граничной частоты излучения. А замечание о том, что длина волны λ выражается в виде $\lambda = \frac{c}{\nu}$ говорит о существовании некоторого максимального значения длины волны (которая соответствует минимальной частоте наблюдения фотоэффекта) – это и есть красная граница.

3.2 Представление данных в формате точек на плоскости

```python
lambdas = [0.4, 0.43, 0.46, 0.49, 0.52]
volumes = [0.803, 0.578, 0.402, 0.229, 0.083]
fig3, ax3 = plt.subplots(1, 1, figsize=(9, 6))
ax3.scatter([lam ** -1 for lam in lambdas], volumes, s=70)
ax3.grid(axis='both')
fig3.tight_layout()
```

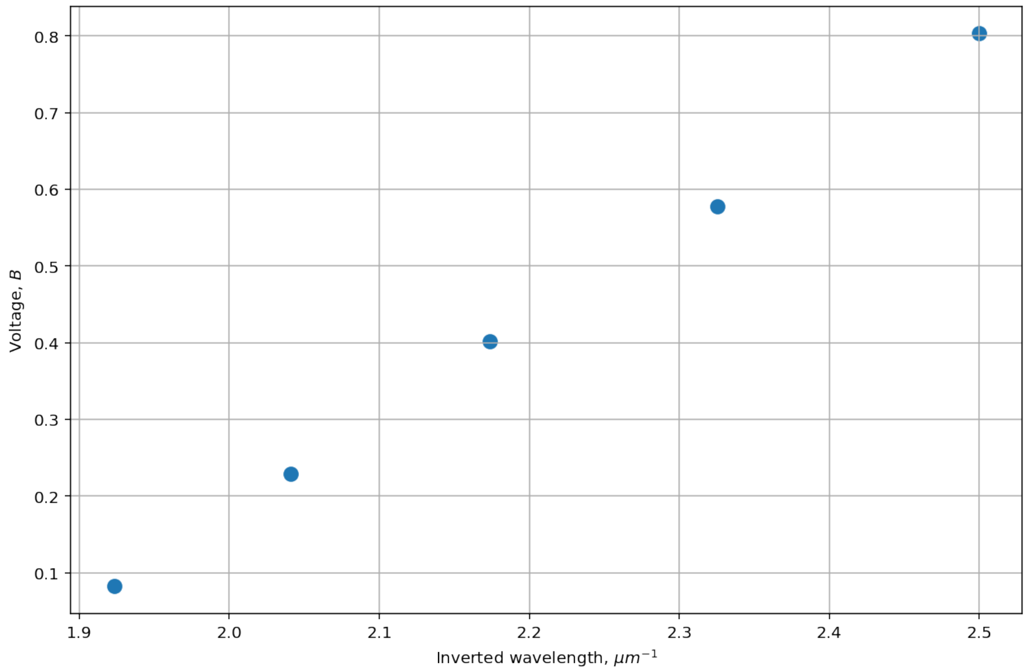

Рис. 3: Данные задачи в точечном формате

3.3 Поиск коэффициентов, линейная регрессия

Заметим, что в матричной форме:

$$X \cdot W = Y \;\rightarrow\; X^T \cdot X \cdot W = X^T \cdot Y \;\rightarrow\; W = (X^T \cdot X)^{-1} \cdot X^T \cdot Y$$

Здесь матрица X принимает вид $\left[\left[\frac{1}{\lambda_1},\, 1\right],\, \left[\frac{1}{\lambda_2},\, 1\right] \ldots \left[\frac{1}{\lambda_n},\, 1\right]\right]$, весовая матрица $W = [a,\, b]$ и Y – вектор значений по запирающей разности потенциалов.

```python
lambdas = [0.4, 0.43, 0.46, 0.49, 0.52]
voltage = [0.803, 0.578, 0.402, 0.229, 0.083]

lambdas_inv = [[lam ** -1, 1] for lam in lambdas]
X, Y = np.array(lambdas_inv), np.array(voltage)
W = np.linalg.inv(np.dot(X.T, X)).dot(X.T).dot(Y)
print(W)
RMSE = sum([val ** 2 for val in X.dot(W) - Y])
print(f'Root Mean Squered Error: {RMSE}')
```

И весовой вектор в итоге $W = [a, b]$:

```
>>> [ 1.24347183 -2.30753275]
>>> Root Mean Squered Error: 8.469118894828288e-05
```

3.4 Поиск работы выхода и постоянной Планка

Обратим внимание на выведенную ранее формулу:

$$U_з = \frac{h\nu - A_{вых}}{e} = \frac{hc}{e\lambda} - \frac{A_{вых}}{e}$$

Очевидно, что в нашем случае весовой вектор можно записать в виде:

$$W = \left[a,\ b\right] = \left[\frac{hc}{e},\ -\frac{A_{вых}}{e}\right]$$

Посчитаем требуемые значения работы выхода и постоянной Планка:

$$h = \frac{e}{c} \cdot a = \frac{1,6 \cdot 10^{-19}}{299792458} \cdot 1,24347183 = 6.636440893822548 \cdot 10^{-28} \mu m$$

$$A_{вых} = e \cdot b = 1,6 \cdot 10^{-19} \cdot 2,30753275 = 3,692052400468156 \cdot 10^{-19} Дж$$

Ответ: $h = 6.636440893822548 \cdot 10^{-34}\ m$
$A_{вых} = 2,30753275$ эВ

ссылка на Google Colab со всеми графиками и вычислениями