Санкт-Петербургский Государственный

Электротехнический Университет "ЛЭТИ"

кафедра физики

Задание №1 по дисциплине

"Физические основы информационных технологий"

Название: Искривление луча в оптическом канале

| Фамилия И.О.: | Герасименко Я.Д. |
| --- | --- |
| группа: | 1303 |
| Преподаватель: | Альтмарк А.М. |
| Итоговый балл: |  |
|  |  |

Крайний срок сдачи: 22.10.23

Санкт-Петербург 2023

***Условие задания***

Найти длину траектории светового луча *S* в прямолинейном дисперсионном оптоволоконном канале, Рис.1, c показателем преломления n1. Оптоволокно окружено средой с показателем преломления n2. Функцию распределения показателя преломления n1(y, ω) можно представить как:

,

где y – поперечная координата, ω – циклическая частота светового луча.

Функцию , функцию Zf(y), описывающую координату z выходного торца волновода, начальный угол ввода луча α в волновод, координату ввода луча в волновод *y*=y0, радиус канала R можно взять в файле FOIT\_IDZ1.xlsx. Все геометрические размеры даются в безразмерных координатах.

Необходимо построить график траектории луча, а также записать ответ *S* в в текстовый файл IDZ1\IDZ1.txt. Помимо текстового файла IDZ1.txt в папке IDZ1 должен находиться Word-файл (Pdf-файл) с отчетом, а также файл с кодом (Python, Mathcad, Mathematica). Для лучшего понимания отчетности смотрите папку “Пример организации яндекс-папки студентов”.

Пример содержания файла IDZ1.txt:

4.53258

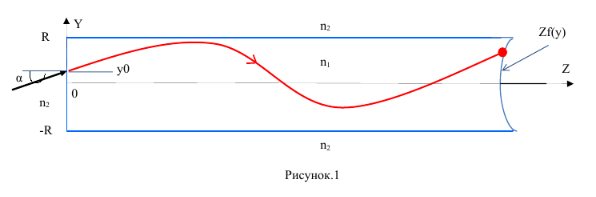


Рисунок.1

***Основные теоретические положения***

Для выполнения работы нам понадобиться воспользоваться с законом Снеллиуса, чтобы определить угол луча после того, как он перейдет из одной среды в другую.

закон Снеллиуса описывает, как изменится направление световой волны при переходе из одной среды в другую Рис.2.

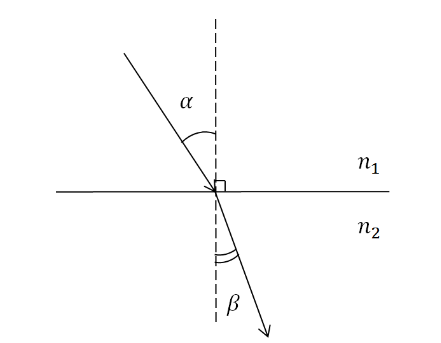


Рис.2

Формула:

n₁ \* sin = n₂ \*

sin

n – показатели преломления

с – скорость света в вакууме

v – скорость света в данной среде.

Показатель преломления – величина, которая описывает насколько сколько измениться скорость световой волны после прохождения через среду, относительно скорости в вакууме.

Ключевой принцип работы оптоволоконного кабеля — полное внутреннее отражение. Он заключается в том, что во время передачи по оптоволоконному кабелю сигналы света отражаются от сердцевины и оболочки во множестве зигзагообразных скачков

Исходные данные

| Вар | R | n2 | f1(y) | Zf(y) | ω \* 10^14, рад/с | y0 | α , град |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 20 | 1.8 | 1 | 1.4 + 0.3\*Cos[0.8\*y]^2 | 12 + 3\*Sin[17.951958020513104\*y] | 3.5 | -0.5 | -42. |

**Выполнение работы:**

Выполнение работы можно разделить на несколько частей:

Входные данные, взятые из таблицы.

**R = 1.8  
n2 = 1  
w = 3.5 \* 10 \*\* 14  
initial\_y = -0.5  
alpha\_deg = -42  
length = 0  
hop = 0.00001  
direction = 1 if alpha\_deg > 0 else -1**

**def f1(y)** - функция для расчета показателя преломления.   
**def Zf(y)** - координата выходного торца оптоволокна..

**def n1(y)** - функция для вычисления показателей среды.

def angle\_refraction(input\_angle, next\_refractive\_index, current\_refractive\_index)- функция, спользуется для вычисления угла преломления света при переходе из одной среды с определенным показателем преломления в другую среду с другим показателем преломления. Функция принимает следующие параметры:

* **input\_angle** - Это угол падения света на границу между двумя средами в радианах.
* **next\_refractive\_index** - Это показатель преломления следующей среды, в которую свет попадает после преломления.
* **current\_refractive\_index -**  Это текущий показатель преломления среды, в которой свет падает.

def calculate\_optical\_path() - функция, вычисляет оптический путь света в среде. Она начинает с начальной точки и двигается в соответствии с законами преломления света. В процессе вычислений записывает координаты точек на этом пути, также строит график оптического путь

**Результат работы программы Рис.3:**

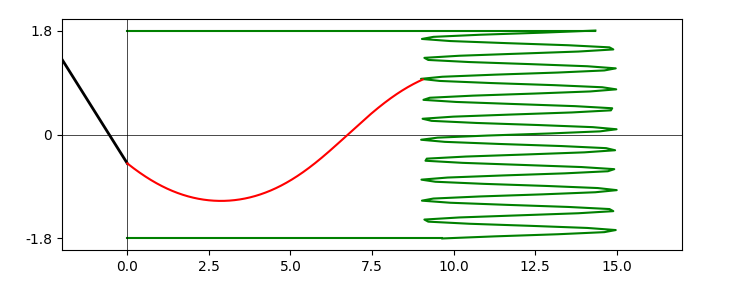


Рис.3

вычисление угла и длины пути Рис.4:



Рис.4

**Вывод:**

В ходе выполнения работы был визуализирована траектория луча в прямолинейном дисперсионном оптоволоконном канале, высчитана его длина.

**ПРИЛОЖЕНИЕ А**

**ПРОГРАММА MAIN.PY**

import math

import matplotlib.pyplot as plt

import numpy as np

R = 1.8

n2 = 1

w = 3.5 \* 10 \*\* 14

initial\_y = -0.5

alpha\_deg = -42

length = 0

hop = 0.00001

direction = 1 if alpha\_deg > 0 else -1

def f1(y):

return 1.4 + 0.3 \* math.cos(0.8 \* y) \*\* 2

def Zf(y):

return 12 + 3 \* math.sin(17.951958020513104 \* y)

def n1(y):

return f1(y) \* (1 - ((0.35 \* 10 \*\* 14) / w) \*\* 2)

def angle\_refraction(input\_angle, next\_refractive\_index, current\_refractive\_index):

angle = (math.sin(input\_angle) \* current\_refractive\_index) / next\_refractive\_index

if angle > 1:

global direction

direction \*= -1

return input\_angle

return math.asin(angle) if -1 <= angle <= 1 else input\_angle

def calculate\_optical\_path():

global length

global direction

n1\_input = n1(initial\_y)

entry\_angle = angle\_refraction(math.radians(alpha\_deg), n1\_input, n2)

print("Угол β на входе в n1 = %.6f градусов" % math.degrees(entry\_angle))

entry\_angle = math.pi / 2 - entry\_angle

Y\_values, Z\_values = [], []

current\_Y, current\_Z, current\_N = initial\_y, 0, n1\_input

while current\_Z < Zf(current\_Y):

Y\_values.append(current\_Y)

Z\_values.append(current\_Z)

next\_Y = current\_Y + math.cos(entry\_angle) \* hop \* direction

next\_Z = current\_Z + abs(math.sin(entry\_angle)) \* hop

next\_N = n1(next\_Y) if R >= abs(next\_Y) else n2

next\_angle = angle\_refraction(entry\_angle, next\_N, current\_N)

length += hop

current\_Y, current\_Z, entry\_angle, current\_N = next\_Y, next\_Z, next\_angle, next\_N

plt.plot(Z\_values, Y\_values, color="red")

plt.figure(figsize=(8, 3))

plt.xlim(-2, 17)

plt.ylim(-2, 2)

plt.plot([0, 9.6545], [-R, -R], color='green')

plt.plot([0, 14.34], [R, R], color='green')

plt.yticks([-R, 0, R], ['-1.8', '0', '1.8'])

plt.axhline(0, color='black', linewidth=0.5)

plt.axvline(0, color='black', linewidth=0.5)

plt.plot([-2, 0], [initial\_y + math.tan(math.radians(alpha\_deg)) \* (-2), initial\_y], color='black', linewidth=2)

Y\_array = np.linspace(-R, R, 100)

Z\_array = [Zf(v) for v in Y\_array]

plt.plot(Z\_array, Y\_array, color="green")

calculate\_optical\_path()

print(length)

print(f"Длина пути = {length}")

with open("IDZ1.txt", "w") as file:

file.write(str(length))

plt.show()