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.: | Андреева Е.А |
| 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:

,

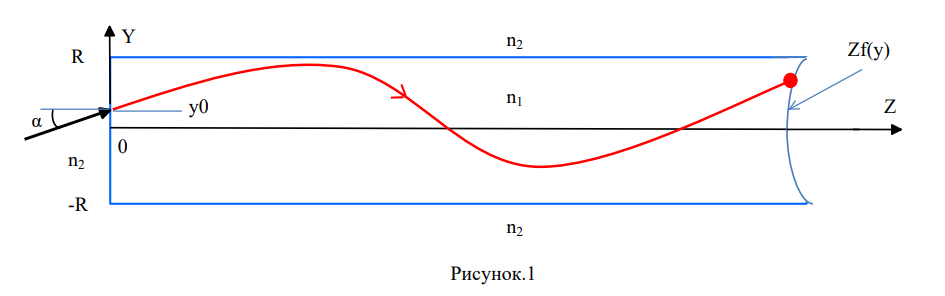
где y – поперечная координата, ω – циклическая частота светового луча.

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



Исходные данные



Основные теоретические положения

Если световые волны достигают границы раздела двух сред и проникают в другую среду, то направление их распространения также изменяется — происходит преломление света*.*

Преломление света — это изменение направления распространения световой волны при переходе из одной прозрачной среды в другую.

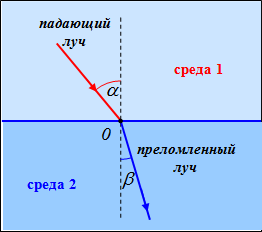


Рисунок 2 – преломление света

Закон преломления гласит:

Падающий луч, перпендикуляр к границе раздела сред в точке падения и преломленный луч лежат в одной плоскости, причем отношение синуса угла падения к синусу угла преломления постоянно для данной пары сред и равно показателю преломления второй среды относительно первой

Здесь  показатель преломления среды, в которой распространяется преломленная волна,  показатель преломления среды, в которой распространяется падающая волна.

**ПРИЛОЖЕНИЕ А**

**ПРОГРАММА MAIN.PY**

from math import \*

import numpy as np

import matplotlib.pyplot as plt

R = 0.7

n2 = 1

omega = 3

y0 = 0.1

alpha = radians(25)

n1\_coeff = (1 - pow((0.35 / omega), 2))

N = 17001

h = 2 \* R / N

length = 0

def f1(y):

return 1.5 - 0.3 \* pow(y, 2)

def Zf(y):

return 20 + 3 \* sin(17.951958020513104 \* y)

def n1(y):

return f1(y) \* n1\_coeff

layers\_n = [n1(i \* h + h / 2 - R) for i in range(N)]

layers\_zf = [Zf(i \* h + h / 2 - R) for i in range(N)]

layers\_coords = [i \* h - R for i in range(N)]

begin\_layer\_index = 0

for i in range(len(layers\_coords) // 2, len(layers\_coords)):

if y0 < layers\_coords[i]:

begin\_layer\_index = i - 1

break

beta = asin(sin(alpha) \* n2 / layers\_n[begin\_layer\_index])

z = h / tan(beta)

length = h / sin(beta)

n1 = layers\_n[begin\_layer\_index]

n1\_index = begin\_layer\_index

n2\_index = n1\_index + 1

up = True

down = False

direction = up

plt.ylim(-R, R)

plt.xlim(0, 25)

plt.ylabel('y', loc='top')

plt.xlabel('z', loc='right')

for elem in layers\_coords:

plt.plot([Zf(elem), Zf(elem) + h], [elem, elem + h], color='blue')

alpha = radians(90) - beta

while z < layers\_zf[n1\_index]:

z += tan(alpha) \* h

length += h / cos(alpha)

if sin(alpha) > layers\_n[n2\_index] / layers\_n[n1\_index] and direction == up:

direction = down

plt.plot([z - tan(alpha) \* h, z], [layers\_coords[n2\_index], layers\_coords[n1\_index]], color='black')

n2\_index -= 2

elif sin(alpha) > layers\_n[n2\_index] / layers\_n[n1\_index] and direction == down:

direction = up

plt.plot([z - tan(alpha) \* h, z], [layers\_coords[n1\_index], layers\_coords[n1\_index + 1]], color='black')

n2\_index += 2

else:

beta = asin(sin(alpha) \* layers\_n[n1\_index] / layers\_n[n2\_index])

if direction == up:

plt.plot([z - tan(beta) \* h, z], [layers\_coords[n2\_index], layers\_coords[n2\_index + 1]], color='black')

n1\_index += 1

n2\_index += 1

alpha = beta

else:

plt.plot([z - tan(beta) \* h, z], [layers\_coords[n1\_index], layers\_coords[n2\_index]], color='black')

n1\_index -= 1

n2\_index -= 1

alpha = beta

plt.show()

print(f'length = {length}')