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.: | Ахримов А.М. |
| группа: | 9303 |
| Преподаватель: | Альтмарк А.М. |
| Итоговый балл: |  |
|  |  |

Крайний срок сдачи: 23.10.21

.

Санкт-Петербург 2021

Условие задания

Найти длину траектории светового луча *S* в прямолинейном оптоволоконном канале, Рис.1. Функцию распределения показателя преломления n1(y) по поперечной координате Y, начальный угол ввода луча α в волновод, длину канала *L*, диаметр канала D можно взять в таблице 1. Ввод луча осуществляется из центральной части канала с координатой *y*=0. Параметры *L* и D даны в безразмерных координатах.

Необходимо построить график траектории луча, а также записать ответ *S* в безразмерных единицах в текстовый файл LR1\result.txt. Помимо текстового файла result.txt в папке LR1 должен находиться Word-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 result.txt:

4.53258

n2

Y

2DDD

n1

Z

α

n2

*L*

Рисунок.1

**Вариант 2**

**Основные теоретические положения**

Оптическая длина пути между двумя точками среды — расстояние, на которое свет (оптическое излучение) распространился бы в вакууме за время его прохождения между этими точками.

Оптическая длина пути в однородной среде с показателем преломления равна произведению геометрической длины пути , пройденного светом, на показатель преломления :

**Преломление света** — явление, при котором луч света, переходя из одной среды в другую, изменяет направление на границе этих сред.

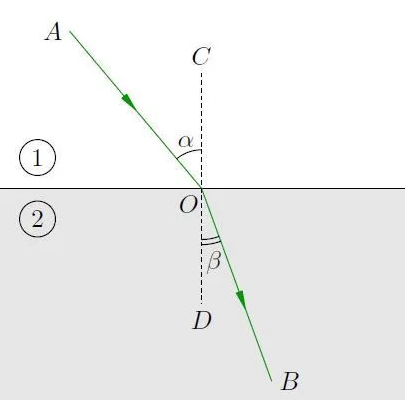


Рисунок 2 – Ход луча через среду

Преломление света происходит по следующему закону: *Падающий и преломленный лучи и перпендикуляр, проведенный к границе раздела двух сред в точке падения луча, лежат в одной плоскости. Отношение синуса угла падения к синусу угла преломления есть величина постоянная для двух сред:*

*где*— *угол падения,* — *угол преломления,*  — *постоянная величина, не зависящая от угла падения.*

**Выполнение работы**

Разобьём ось Y, вдоль которой меняется показатель преломления, и для каждого разбиения посчитаем угол, на который отклоняется луч, проходя через среду. Для каждого разбиения вычисляется функция распределения показателя преломления n1(y), затем по закону преломления света вычисляется синус угла преломления. Геометрическая длина пути вычисляется согласно определению синуса и теореме Пифагора.

Таким образом были получены длина траектории светового луча и следующий график (см. рисунок 3):

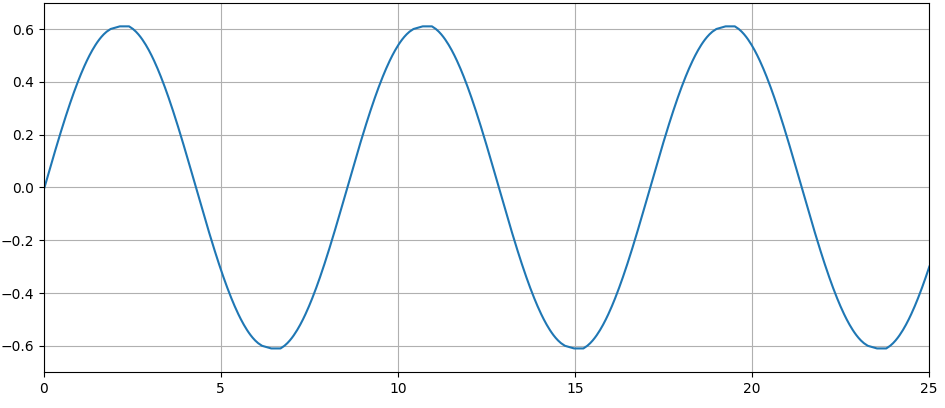


Рисунок 3 – График траектории оптического луча

**ПРОГРАММА MAIN.PY**

**import math**

**import matplotlib.pyplot as plt**

**import numpy as np**

**alpha\_0 = math.radians(90 - 25)**

**delta\_y = 0.0001**

**max\_y = 0.7**

**L = 25**

**def func\_refractive(y):**

**return 1.2 - 0.3 \* y \*\* 2**

**def calculate\_sin\_beta(sin\_alpha, n\_1, n\_2):**

**return sin\_alpha \* n\_1 / n\_2**

**def calculate\_s(sin\_alpha, y):**

**return y/(math.sqrt(1 - sin\_alpha\*\*2))**

**def calculate\_z(sin\_alpha, y):**

**return y / math.sqrt(1 / sin\_alpha \*\* 2 - 1)**

**sin\_alpha = math.sin(alpha\_0)**

**n\_1 = func\_refractive(0)**

**points\_z = []**

**points\_y = []**

**z = 0**

**y = 0**

**s = 0**

**while z < L:**

**n\_2 = func\_refractive(y)**

**sin\_beta = calculate\_sin\_beta(sin\_alpha, n\_1, n\_2)**

**if sin\_beta >= 1:**

**sin\_beta = sin\_alpha**

**delta\_y = - delta\_y**

**y += delta\_y**

**continue**

**s += calculate\_s(sin\_beta, np.abs(delta\_y))**

**z += calculate\_z(sin\_beta, np.abs(delta\_y))**

**n\_1 = n\_2**

**sin\_alpha = sin\_beta**

**points\_z.append(z)**

**points\_y.append(y)**

**if np.abs(y) > max\_y:**

**delta\_y = - delta\_y**

**y += delta\_y**

**plt.axis([0, L, -max\_y, max\_y])**

**plt.grid(True)**

**plt.plot(points\_z, points\_y)**

**plt.show()**

**print(s)**