# Аннотация

Выпускная квалификационная работа изложена на \_\_ страницах, состоит из введения, трех глав, заключения, включает \_\_ рисунка, \_\_ таблиц, библиографический список из \_\_ источников литературы, \_\_ приложения на \_\_ страницах.

Объектом выпускной квалификационной работы является разработка программной системы определения типа поверхности удаленных объектов поляриметрическими методами.

Предметом – математические модели и алгоритмы для программных комплексов вычисления оптических параметров покрытий космических объектов.

Цель выпускной квалификационной работы – разработка новой версии программной системы определения типа поверхности удаленных объектов поляриметрическими методами с использованием современных средств проектирования, которое позволит использовать разработанную систему в исследовательских и учебных целях.

При разработке системы использовались следующие средства разработки:

* *pyhon* 3;
* формат базы данных:*sqlite*;
* *Visual Stuio Code*;
* *git*.

Выпускная квалификационная работа и ее результаты могут быть использованы в исследовательских и учебных целях.

# Задание на дипломное проектирование

# Содержание

|  |  |  |
| --- | --- | --- |
|  | Введение | 5 |
| 1 | Математические модели и алгоритмы прямой и обратной задачи фотометрии | 8 |
| **1.1** | **Модель рассеяния света от поверхности космического объекта и регистрации светового пучка** |  |
| **1.2** | **Применение вектор-функции Стокса** |  |
| **1.3** | **Расчет ветктора Стокса падающего излучения** |  |
| 2 | Современные методи и алгоритмы определения типа покрытия космических объектов |  |
| **2.1** | **Постановка задачи** |  |
| **2.2** | **Определение типа покрытия космических объектов** |  |
| **2.3** | **Классификация типа покрытия** |  |
| **2.4** | **Алгоритм вычисления показателя преломления однородной диэлектроческой поверхностью** |  |
| **2.5** | **Алгоритм вычисления показателя преломления металлических покрытий** |  |
| 3 | Программная система вычисления показателя преломления поверхностей космических объектов и определения материала покрытий STOKES |  |
| **3.1** | **Архитектура программной системы STOKES** |  |
| **3.2** | **Информационная модель программной системы STOKES** |  |
| **3.3** | **Руководство программиста программной системы STOKES** |  |
| **3.4** | **Тестирование программной системы STOKES** |  |
|  | Заключение |  |
|  | Приложение А. Листинг основных программных модулей |  |
|  | Приложение Б. Руководство оператора программной системы STOKES |  |
|  |  |  |

# Введение

Тема выпускной квалификационной работе – исследование алгоритма определения типа поверхности удаленных объектов поляриметрическими методами

В связи с развитием космических технологий и постоянным пребыванием человека в околоземном космическом пространстве, проблема загрязнения окружающей среды становится насущной в отношении не только экосферы Земли, но и околоземного космического пространства. Данная работа посвящена разработке одного из методов решения данной проблемы – идентификации космических объектов с помощью методов фотометрии, поэтому задача разработки и исследование алгоритмов идентификации космических объектов ялвяется актуальной.

Впервые процессы рассеяния и переноса поляризованного света смоделировал математик и физик из Ирландии Джордж Габриель Стокс. В 1852 году он предложил вектор из четырёх компонент, названный вектором Стокса.

С применением интегрального уравнения Фредгольма Р. Азам, О. Френель, Э. Малюс, Д. Брюстер, Ф. Басс, Н. Башара и И. Фукс описали процессы отражения и переноса поляризованного пучка света.

Существенный вклад в реализацию разработки алгоритмов фотометрических задач был внесён отечественными учёными, такими как Т. А. Сушкевич, В. В. Коротаев, Г. Л. Башнина, Г. В Розенберг, С. А. Ухинов.

Учёные РГУ (Рязанский государственный университет) также внесли немалый вклад в разработку методов мониторинга околоземного космического пространства оптическими средствами. А. К Муртазов, А. В. Белошенков, В. И. Курышев и В. В. Куприянов впервые применили алгоритмы решения задач с параметрами рассеяния на космических околоземных объектах с использованием параметров Стокса.

Однако, несмотря на весьма богатую историю изучения вопросов данной сферы, немало пробелов в познании этой области остаётся и по сей день.

Цель выпускной квалификационной работы – разработка новой версии программной системы определения типа поверхности удаленных объектов поляриметрическими методами с использованием современных средств проектирования, которое позволит использовать разработанную систему в исследовательских и учебных целях.

Данная система позволит сократить объем «бумажной» работы, увеличить наглядность представления данных, систематизировать проведение исследований и расчетны задач.

Для достижения поставленной в работе цели необходимо решить следующие задачи:

* изучить математические модели и алгоритмы прямой и обратной задачи фотометрии
* исследовать современные методи и алгоритмы определения типа покрытия космических объектов;
* на основе сделанного анализа спроектировать базу данных;
* разработать удобный и функциональный интерфейс для пользователя;
* реализовать заданные функции программно;
* выполнить экспериментальную проверку разработанной программной системы..

Пояснительная записка к дипломному проекта состоит из введения, трех разделов, заключения и приложений.

В первом разделе описаны математические модели и алгоритмы прямой и обратной задачи фотометрии, а также описаны модель рассеяния света от поверхности космического объекта и регистрации светового пучка.

Второй разрез содержит описание современных методов и алгоритмов определения типа покрытия космических объектов, включая алгоритмы вычисления показателя преломления однородной диэлектроческой поверхностью и показателя преломления металлических покрытий

Третий разрез содержит описание программной системы вычисления показателя преломления поверхностей космических объектов и определения материала покрытий STOKES. В разделе дается краткое описание архитектуры программной системы, ее информационной модели, руководство программиста и описание процесса тестирование программной системы STOKES.

В заключении описана успешность и полнота решения поставленных задач выпускной квалификационной работы, дается оценка перспектив развития вопросов по теме работы.

# 1 Математические модели и алгоритмы прямой и обратной задачи фотометрии

Решение задач подобного рода позволяет эффективно определять тип поверхности космического объекта и её состояние. Оптический мониторинг – один из основных методов исследования космических объектов. На больших же высотах, где радиолокационные методы теряют свою эффективность, такой метод исследования является единственным из доступных.

Актуальность данной работы обосновывается малой степенью изученности вопросов идентификации малых космических околоземных объектов. Решение таких вопросов позволит эффективнее находить и определять космические объекты различной природы происхождения.

Задачами данной работы являются:

1. Провести анализ библиографических данных, посвящённых вопросам фотометрии и электромагнитного излучения, с целью получения необходимой базы для разработки методов решения поставленных задач;
2. Собрать и обработать данные из отобранных источников;
3. Проанализировать обработанные данные;
4. Разработать методы решения поставленных задач с использованием полученных данных;
5. Выявить наиболее эффективные из разработанных алгоритмов решения;
6. Разработать программу реализации отобранных алгоритмов;
7. Реализовать решение поставленных задач с применением разработанной программы;
8. Сделать заключение о проделанной работе и проанализировать полученные результаты.

Объектом исследования данной работы является природа оптических характеристик изучаемых объектов (в данном случае, малых космических объектов околоземного космического пространства). Предметами исследования выступают параметры оптических свойств изучаемых объектов: параметры поляризации отражённого света, параметры электромагнитных потоков.

Мониторинг оптическими средствами позволяет идентифицировать объекты космического пространства с возможностью определения типа поверхности, а также геометрических и габаритных размеров, с целью распознавания и устранения опасных объектов.

Главная задача мониторинга – распознавание формы и типа поверхности и измерение геометрических параметров космического тела. Данная задача может быть разделена на две фотометрических задачи.

1. Прямая задача фотометрии. Представляет собой измерение оптических характеристик изучаемого объекта.
2. Обратная задача фотометрии. По результатам измерения оптических параметров определяются геометрические параметры тела.

С использованием исходных данных изучаемого тела и средствами математического моделирования для измерения доступны следующие параметры:

1. параметры положения в пространстве (ось вращения, ориентация),
2. геометрические параметры (габаритные размеры, форма),
3. параметры структуры (компоненты тела и их взаимное расположение),
4. физические параметры поверхности (оптические характеристики, состояние поверхности).

В ходе данный работы даётся постановка задачи разработки математических методов измерения параметров 4-ой группы методами фотометрии. За исследуемые параметры берутся параметр преломления поверхности, состояние поверхности, материал покрытия.

Методы исследования с помощью фотометрии доказывают свою эффективность при обследовании поверхностей космических объектов и аппаратов. Физические свойства объекта, геометрия и габариты, химическое строение, структура и оптические параметры поверхности исследуются данным методом.

На рисунке 1.1 представлена система из излучателя, объекта и приёмника в сферической системе координат.

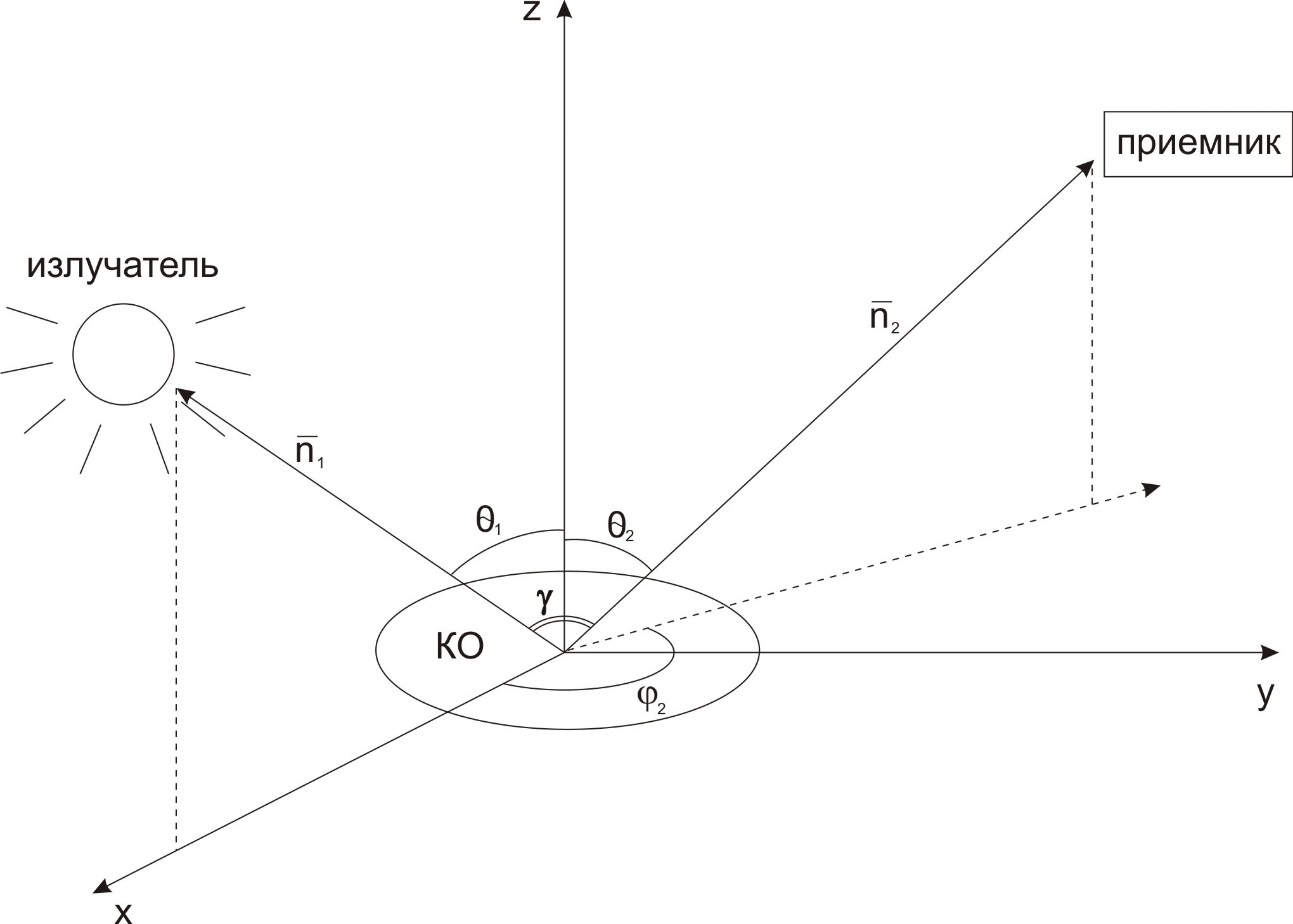


Рис. 1.1. Система из излучателя и приёмника в системе координат.

Начало координат соответствует поверхности космического объекта, расположенного в плоскости XOY. Ось OZ соответствует нормали к плоскости XOY, то есть к поверхности объекта. Угол  соответствует углу между вектором «объект-излучатель»  и проекцией этого вектора на плоскость XOY (коллинеарно направлению оси OX). Угол  соответствует углу между проекцией вектора «объект-приёмник»  на плоскость XOY и осью OX. Углы  и  соответствуют углам между нормалью OZ и векторами  и  соответственно.  – угол между векторами  и .

Модель светового пучка в некоторой среде можно представить как электромагнитное поле с параметрами напряжённости , диэлектрического смещения , магнитной составляющей электромагнитного поля  и вектором магнитной индукции . Уравнения Максвелла позволяют связать эти параметры:



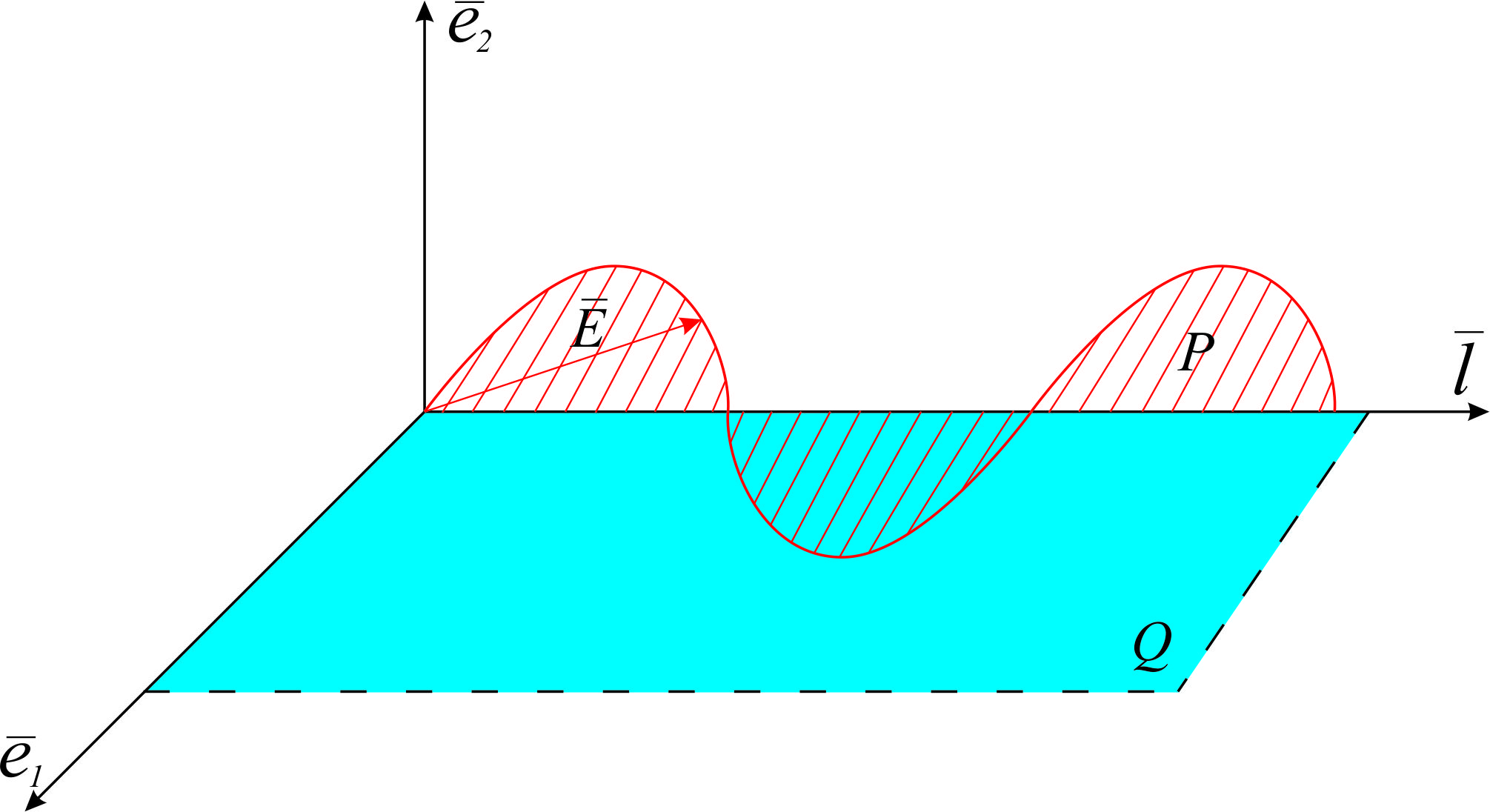
где переменными  обозначена плотность тока,  – плотность заряда по объёму.

Характер колебания вектора напряжённости определяет поляризацию светового поля. Распространение световой волны перпендикулярно направлению колебаний данного вектора, при этом вектор  колеблется гармонически.

Вектор  можно представить через проекции  и  на орты  и . Тогда направление распространения световой волны  будет перпендикулярно данным проекциям. Теперь можно рассмотреть следующие плоскости в системе координат из векторов , , :

1.  – плоскость поляризации, проходит через вектор и направление колебания вектора ;
2.  – плоскость референции, проходит через вектор  и орту .

Теперь вектор напряжённости электромагнитного поля светового потока  можно описать следующим образом:



*Рис. 1.2. Плоскости P, Q и вектор  в системе координат , , .*

|  |  |
| --- | --- |
| , |  |

здесь  – амплитуды ,  – начальные фазы компонент ,  - фазовая скорость светового потока.

В зависимости от траектории вектора напряжённости светового потока вдоль направления распространения световой волны различают линейную, циркулярную или эллиптическую поляризации. Наибольший интерес представляет эллиптическая поляризация, так как остальные виды являются её частными случаями.

На рисунке 1.3 представлены параметры, характеризующие эллипс поляризации в плоскости XOY:

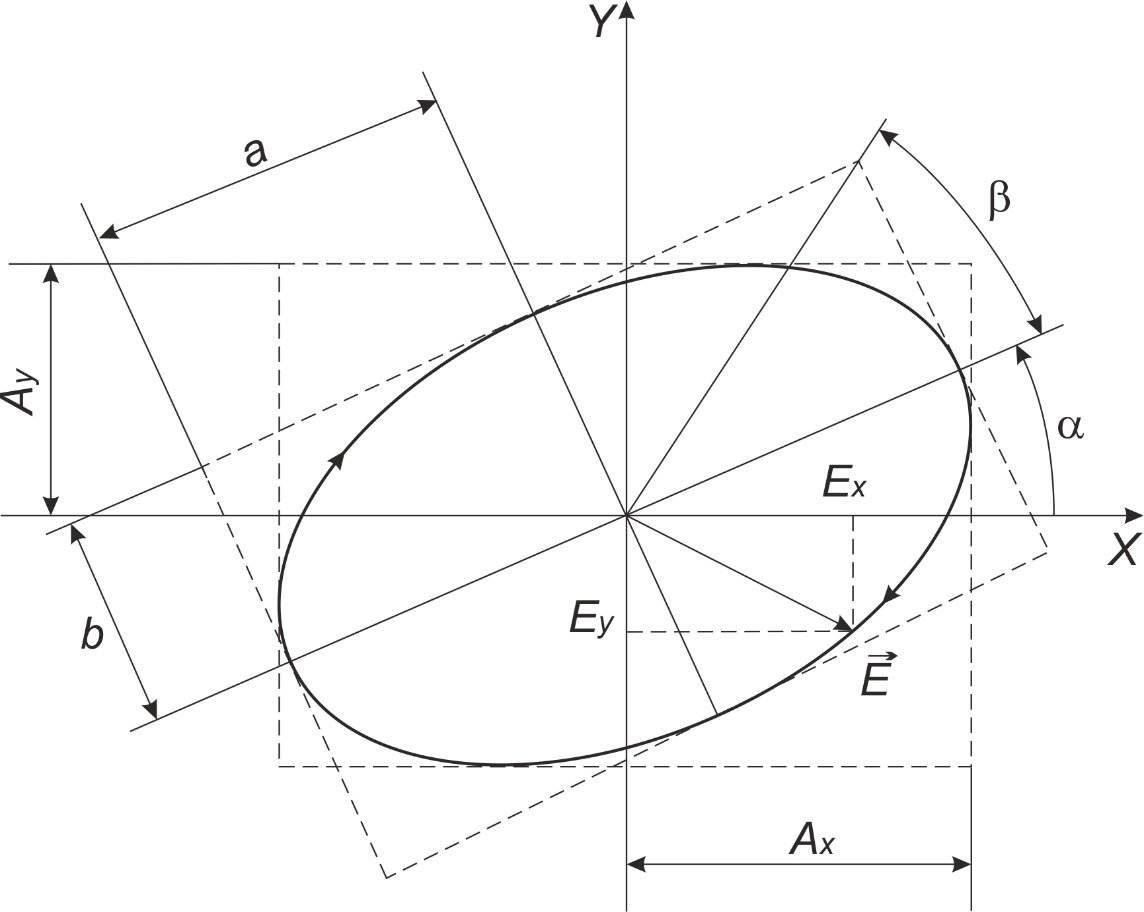


Рис. 1.3. Эллиптическая поляризация и её параметры.

1. Эллиптичность. Выражается коэффициентом эллиптичности :. Здесь  – полуоси эллипса.  – угол эллиптичности. Параметры  и определяют знак эллиптичности, то есть направление вращения вектора напряжённости по отношению к вектору  направления светового потока.
2. Азимутальный угол , обозначает угол между большой полуосью поляризационного эллипса и направлением оси координат OX.
3. Амплитуда колебания эллипса . Переменная  обозначает общую интенсивность светового поля.
4. Параметры амплитуд  составляющих , вектора , равные описанным вокруг эллипса сторонам прямоугольника. Кроме того, имеется угол между диагональю данного прямоугольника и осью OX, выражаемый как .
5. Разность фаз составляющих  в точке отсчёта: .

Напряжённость поля монохроматического светового потока можно выразить так:



Вектор Джонса представляет собой выражение световой волны в начальном положении  и в начальный момент времени :

|  |  |
| --- | --- |
| . |  |

Данный вектор является комплексным. Компоненты пар вектора Джонса описывают поляризационное состояние волны. При этом ,  комплексно сопряжены компонентам , .

Удобнее всего состояние поляризации представить с помощью параметров Стокса – четырёхкомпонентного вектора, компоненты которого представляют собой линейные комбинации координат вектора Джонса (угловыми скобками обозначено усреднённое во времени значение):

|  |  |
| --- | --- |
|  |  |

Понятие вектора Стокса  в четырёхмерном пространстве было введено Дж. Г. Стоксом в 1852 году.

Смысл компонент Стокса с точки зрения физики:

* - интенсивность светового потока;
* - степень поляризации;
* - представление плоскости поляризации;
* - эллиптичность светового потока.

Параметр интенсивности  не может быть меньшим или равным нулю.

Степень поляризации  выражается через разность интенсивностей компонент  и . При этом  достигает максимума при линейной поляризации с  и минимума при .

Параметр , выражающий плоскость поляризации, представляет собой соотношение линейно-поляризованных компонент с азимутальными углами  ( > 0) или  ( < 0).

Эллиптичность светового потока  показывает соотношение левой и правой циркуляции в световой волне. Направление циркуляции определяет знак данного параметра.

Формула 1.4 описывает основные свойства составляющих вектора Стокса:

|  |  |
| --- | --- |
|  |  |

Численно степень поляризация  описывается через связь остальных компонент (формула 1.5):

|  |  |
| --- | --- |
|  |  |

Из формулы следует, что нормированный вектор Стокса выражается как , так как для неполяризованной волны .

При степени поляризации равной единице . Такая световая волна полностью поляризована.

Связь вектора Стокса и Джонса можно выразить по формуле 1.6:

|  |  |
| --- | --- |
|  |  |

Наряду с определением направления распространения волны, параметры Стокса определяют её состояние, что делает данную вектор-функцию базовой для определения свойств поляризованного света.

Для решения фотометрических задач применяется интегральное уравнение, характеризующее рассеяние световых волн исследуемыми поверхностями:



где  – вектор Стокса падающей волны,  – вектор Стокса рассеянной волны,  – матричная яркость компоненты покрытия с индексом  (описывается с помощью матрицы Мюллера),  – единичные орты, направление которых соответствует направлениям на источник и приёмник излучения, описывает нормаль к компоненте поверхности с индексом .

Для описания связи поляризованного излучения, описываемого вектором Стокса, и рассеивающей поверхности, была введена матрица Мюллера (так же именуется матрицей рассеяния, фазовой или поляризационной матрицей). Она служит функциональным оператором в теории рассеяния света. Матрица четвёртого порядка преобразует вектор Стокса падающей волны в параметр Стокса рассеянного излучения.

В результате столкновения с поверхностью вектор напряжённости  в точке с направлением  преобразуется в новый вектор , для которого задаются новые направление  и координаты . Данное преобразование можно описать следующим образом:

.

Таким образом, с помощью матрицы  можно модельно описать воздействие объекта на излучение. При этом матрица состоит из комплексных элементов. В случае многократного взаимодействия излучения с различными объектами конечный вектор будет рассчитываться как результат произведения промежуточных матриц преобразований. В случае единовременных преобразований результирующий вектор рассчитывается как сумма матриц единичных преобразований.

В случае взаимодействия излучения с веществом линейное однородное преобразование функции координат и направления при преобразовании вектора Стокса так же имеет место быть:

.

Здесь  – квадратная матрица преобразования четвёртого порядка:

,

Элементы данной матрицы линейно зависимы от компонент :

,

,

,

,

,

,

,

,

,

,

,

,

,

,

,

.

Несколько разновидностей матриц преобразования уместны при математическом моделировании рассеяния и преобразования потока света:

1. Матрица отражения на границе сред.

Данная матрица получается при замене элементов матрицы : . Здесь  – амплитудные коэффициенты отражения Френеля. В результате получается матрица (Формула 1.7):

|  |  |
| --- | --- |
| . |  |

1. Матрица преломления на границе сред:



Здесь  – амплитудные коэффициенты преломления Френеля.

1. Матрица поляризатора.

Поляризатор, преобразующий падающий световой поток в линейно-поляризованный с вектором , можно описать по формуле 1.8:

|  |  |
| --- | --- |
| . |  |

1. Матрица модулятора.

Матрица описывает модулятор, вносящий между элементами вектора напряжённости  и разность фаз  (формула 1.9):

|  |  |
| --- | --- |
| . |  |

1. Матрица оптически активного кристалла.

Матрица описывает оптически активный кристалл, поворачивающий поляризационную плоскость на угол :

|  |  |
| --- | --- |
| . |  |

1. Матрица дихроичной пластинки.

Матрица описывает действие дихроичной пластинки, которая уменьшает в  и  раз амплитуды элементов  и :



1. Матрица поворота плоскости референции против часовой стрелки на угол :

|  |  |
| --- | --- |
| . |  |

В данной главе были приведены и изучены математические модели поляризованного излучения и его отражения от поверхностей тел. Рассмотренные инструменты послужат фундаментом при усовершенствовании существующих моделей с помощью аналитического моделирования и при описании алгоритмов расчёта характеристик поляризации отражённого излучения и характеристик покрытий объектов.

Параметры Стокса можно выразить через функции от интенсивности *I*, угла  и эллиптичности (формула 2.1):

|  |  |
| --- | --- |
|  |  |

В данной главе решаются следующие 3 задачи.

1. Расчёт компонент вектор-функции Стокса.

Расчёт проводится с применением характеризующих поляризационный эллипс параметров (азимутальный угол, угол эллиптичности, разность фаз составляющих ), а также с использованием метода аналитического моделирования с применением выражений Джонса. С помощью новых параметров вектор-признак Стокса расширяется, что даёт возможность разработки более точных методов определения параметров покрытий.

1. Расчёт матриц оператора рассеяния.

Расчёт проводится для двух видов матриц: рассеяния от металлической и диэлектрической поверхностей. В качестве инструмента расчёта выступают выражения Френеля и Стокса.

1. Усовершенствование установки симуляции физических процессов.

Необходимо модернизировать оптическую установку, моделирующую отражение излучения от разнотипных поверхностей. Структурная схема установки при этом соответствует схеме на рисунке (1.1). В установку встраивается расчётный модуль для вычисления параметров поляризации отражённого излучения с последующей оценкой параметров покрытия.

Компоненты вектора Стокса  представляются вещественными, и для них справедливы неравенства , , что вытекает из физического смысла интенсивности *I*. При стационарном состоянии для компонент вектора Стокса справедливы выражения:



 .

В случае, если состояние нестационарно, имеет место .

Опишем норму на множестве *S*:

, .

Необходимо доказать, что множество *S* – выпуклое.

**Определение**. Множество  называется выпуклым, если , (.

Введём векторы , . Тогда  – линейная комбинация векторов , . В случае  , в случае  .

Так как для параметров Стокса справедливо , то



Отсюда вытекает выпуклость множества *S* вектор-функций Стокса.

**Определение**. Выпуклое замкнутое множество  вещественного нормированного пространства называется конусом, если ,  для .

Необходимо доказать, что *S* – конус:





 при 

**Определение**. Конус  является нормальным, если , где  – универсальная константа.

, тогда , что справедливо, если .

Компоненты плоскости поляризации и эллиптичность светового потока из свойств вектора Стокса близки к нулю, тогда допустимо . Отсюда

, .

Всё вышеописанное говорит о том, что множество *S* – конус.

**Определение**. Конус  вещественного пространства  называется воспроизводящим, если  можно представить в виде разности двух элементов , т.е. .

Возьмём некоторый вектор :

.

Поскольку из выражения выше , то оба слагаемых принадлежат конусу :



**Определение**. Конус называется телесным, если множество внутренних точек конуса  является непустым.

**Определение**. Конус является воспроизводящим тогда и только тогда, когда он телесный.

Из всех приведённых доказательств следует доказанность теоремы:

**Теорема**. Множество вектор-функций Стокса *S* является воспроизводящим, нормальным, телесным конусом.

Рассеяние потока можно выразить через линейный интегральный оператор на воспроизводящем нормальном конусе вектор-функций Стокса в банаховом пространстве , поскольку поляризация меняется после отражения потока от рассеивающей поверхности:

,

здесь – ядро линейного оператора.,  – множество вектор-функций Стокса при начальном состоянии потока,  – множество вектор-функций Стокса при изменённом состоянии потока.

Данное интегральное уравнение переходит в решение интегральных уравнений Фредгольма, которые классифицируются как некорректно поставленные задачи.

Линейный оператор *L* выражает преобразование вектор-функции  из исходного состояния в состояние после отражения , то есть происходит преобразование координат вектора Стокса и меняется поляризация светового потока после отражения. Как результат отражения от *n* поверхностей, имеем систему воспроизводящих конусов  и соответствующую систему операторов .

В предыдущем параграфе был проведён расчёт компонент вектор-функции Стокса с применением дополнительных параметров азимутального угла, угла эллиптичности и разности фаз компонент , описывающих эллипс поляризации.

Для решения данной задачи рассмотрим поляризационный эллипс в системе координат , оси которой представлены малой и большой осями эллипса. Тогда в базисе этой системы . Поскольку интенсивность не зависит от выбора базиса, то , и



Вектор  может быть представлен в :



При переходе из  в базис :



В матричное форме:

.

В результате имеем:

.

В конечном виде компоненты векторов  и  могут быть записаны как:

|  |  |
| --- | --- |
|  |  |

В соответствии с формулами 2.2 и 1.3 компоненты вектора Стокса могут быть выражены следующим образом.

Первая компонента :







Вторая компонента :







Здесь .

Найдем :









Третья компонента :



Четвертая компонента :



Таким образом, была доказаны теоремы:

Теорема 1. Компоненты вектора Стокса для полностью поляризованной световой волны выражаются через параметры :

|  |  |
| --- | --- |
|  |  |

2. Для световой волны с неполной поляризацией :

|  |  |
| --- | --- |
|  |  |

3. При разности начальных фаз компонент  вектора напряженности светового поля  получаем известные формулы частично поляризованного светового потока:



Рассеяние светового потока некоторой поверхностью можно описать линейным уравнением  В диссертационной работе был проведён расчёт матрицы *M* линейного оператора рассеяния металлической поверхностью, в ходе которого для линейного преобразования была получена матрица рассеяния Мюллера:

|  |  |
| --- | --- |
| . | (2.5) |

Здесь ,  – амплитудные коэффициенты Френеля, равные отношению амплитуд падающей и отражённой световых волн (перпендикулярно и параллельно соответственно).

Также была получена собственная матрица линейного оператора рассеяния:

,

и в базисе этой матрицы был представлен вектор Стокса:

.

Аналогичным образом была рассчитана матрица рассеяния от диэлектрической поверхности:

|  |  |
| --- | --- |
| , | (2.6) |

был получен собственный базис:



и через него выражен вектор Стокса:

.

Автором диссертационной работы была решена задача модификации комплексной оптической установки с помощью поляризационных приборов с целью расчёта параметров Стокса, степени поляризации отражённого потока и дополнительных параметров (азимутального угла и угла эллиптичности). Комплексная установка Рязанского государственного университета имени С. А. Есенина для расчёта оптических параметров поверхностей тел и реализации моделирования процессов отражения светового потока была модифицирована.

Установка представляет собой реализацию схемы (рис. 1.1): модель, излучатель и приёмник. Модель может симулировать вращение объекта вокруг двух осей с различной скоростью. В качестве излучателя выступает монохроматор УМ-2 с призмой постоянного отклонения Аббе. Объектив – ахроматический астрономического рефрактора с относительным отверстием 1:10. Источник света – стабилизированная кинолампа. В качестве приёмного устройства выступает фотоэлектрический фотометр с сопротивлением делителя напряжения 3Мом. Установка также включает люксометр для контроля постоянства излучения и лампу накаливания малой мощности в качестве контрольного источника света.

Характеристика величины светового потока и её изменение осуществляется на выходе приёмного устройства путём измерения напряжений:

,

 - напряжение, вызванное световым потоком от модельного объекта,  - фоновое напряжение,  - темновое напряжение ФЭУ-УПТ.

Относительный световой поток выражается:

,

 – напряжение от контрольного источника света.

По формуле  вычислялась среднеквадратичная ошибка ( – коэффициент Стьюдента при  измерениях).При этом относительная ошибка измерения потока составила 1.7%. В случае превышения величины , измерения сбрасывались.

На рисунке 2.1 представлена функциональная схема, адаптированная под наблюдения за космическим мусором околоземного космического пространства, при этом поляризационные элементы находятся между рассеивающим телом и фотоприёмником. Исходя из этого в оптической установке приборы поляризации располагались между приёмным устройством и моделью.

Блок поляризации есть соединение фотометрических приборов в последовательном порядке:

1. модулятор M, осуществляет сдвиг по фазе  между гармониками  и ;
2. анализатор A, вращает поляризационную плоскость на угол ;
3. поляризатор P, выделяет линейно-поляризованную волну в составляющей с поляризационной плоскостью угол  плоскости.

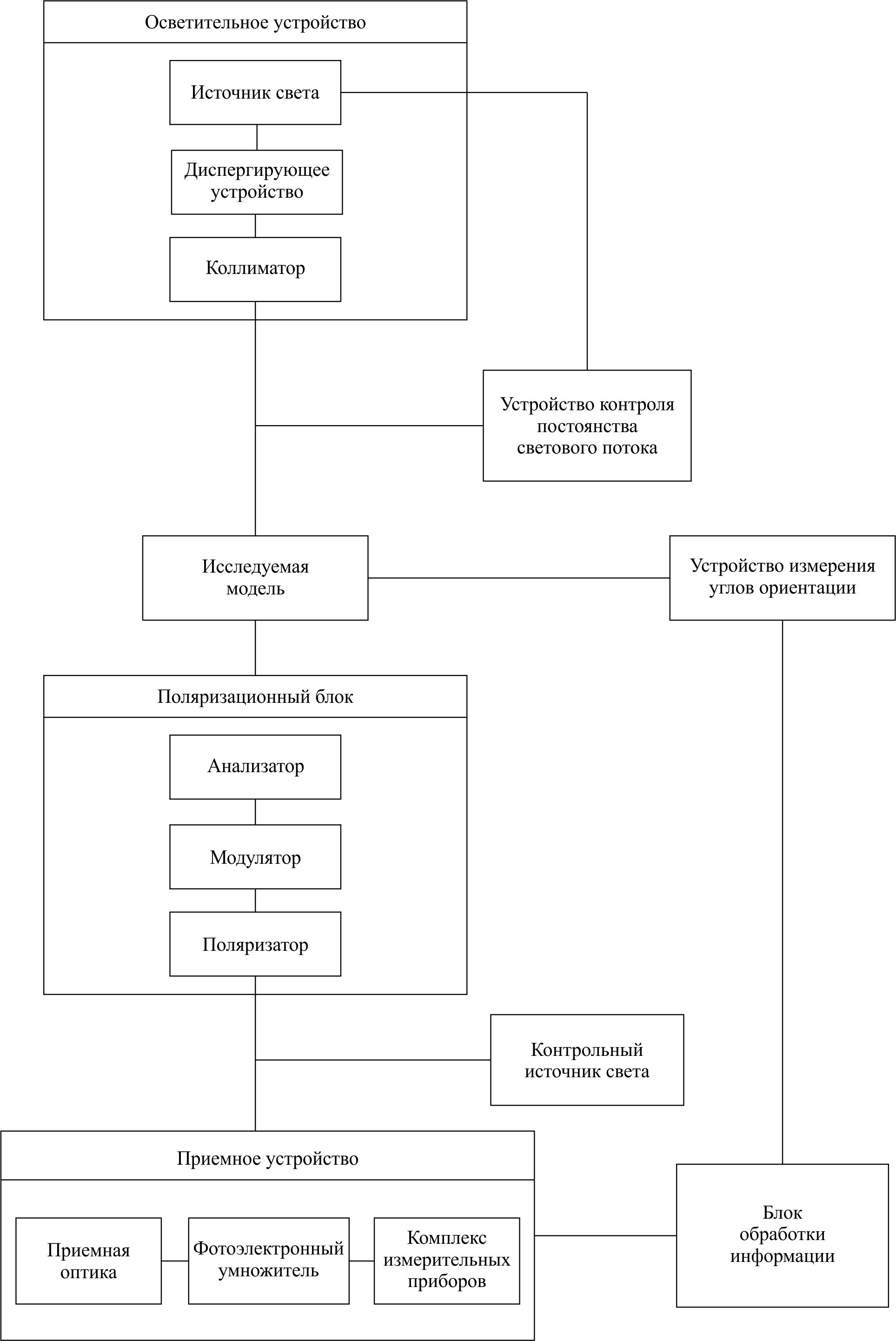


Рис. 2.1. Функциональная блок-схема модифицированной оптической установки.

Данная установка послужила основой для разработки схемы вычисления параметров поляризации отражённого излучения. Метод измерений интенсивности светового потока сложно реализовать технически при измерениях фотометрии, если приборы располагать между объектом и падающим лучом. Поэтому здесь берётся схема, при которой модули располагаются между приёмником и модулью.

# 3 Программная система вычисления показателя преломления поверхностей космических объектов и определения материала покрытий STOKES

## 3.1 Обоснование выбора средств разработки программной системы

### 3.1.1 Обзор Python

Python (далее «Пайтон») представляет собой объектно-ориентированный интерпретируемый язык программирования. Будучи языком общего назначения, Пайтон включает пакет библиотек для различных целей: выполнение математических операций (*math, numpy*), построение графиков (*matplotlib*), работа с базами данных (*sqlite3*), работа с языком HTML (*html*), работа с сетевыми протоколами (*socket*) и др. Существует и множество сторонних библиотек, расширяющих функционал языка: создание GUI (*wx*), создание анимации и игр (*pygame*), разработка веб-приложений (*webapp2*) и др.

Основными чертами языка Пайтон являются следующие.

1. Динамическая типизация.

При такой типизации переменная привязывается к типу данных не во время её объявления (что в принципе отсутствует в языке как таковое), а во время связывания переменной с некоторым значением. Данная особенность, во-первых, избавляет от необходимости объявлять переменную и тип присваиваемых данных перед включением в код, а во-вторых позволяет одну и ту же переменную в рамках одной программы связывать со значениями различных типов.

2. Автоматическое управление памятью (сборщик мусора).

В отличие от других языков программирования, таких как *C/C++* и *Pascal*, в Пайтоне реализован так называемый «сборщик мусора». Переменные автоматически удаляются из памяти, как только они перестают участвовать в выполнении программы.

3. Интроспекция.

Возможность получения типа и структуры данных объекта в процессе выполнения программы. Под объектом подразумеваются в том числе и любые переменные, привязанные к значению конкретного типа, поскольку типы данных на языке Пайтон также представляют собой классы объектов, а данные – представители этих классов. Например, целое число *5* является объектом класса *int* с соответствующей структурой (методами и атрибутами). Один из способов реализации интроспекции – функции *type()* и *dir()*.

4. Реализация обработки исключений.

Данная особенность позволяет предписывать поведение программы в случае возникновения ошибок (исключений). Осуществляется с помощью операторов *try – except – else – finally*.

5. Поддержка многопоточных вычислений.

Позволяет при необходимости разбивать ход выполнения программы на несколько потоков, обрабатываемых одновременно или в заранее описанном порядке. Инструменты реализации предоставляются, например, библиотекой *threading*.

6. Структуры данных высокого уровня.

Будучи высокоуровневым языком программирования, Пайтон реализует высокоуровневые структуры данных – абстракции, описание которых на низкоуровневых языках или машинном коде требует больших усилий. В Пайтоне примерами таких структур являются *списки* и *словари*.

7. Перегрузка операторов.

В отличие от таких популярных языков программирования, как *Java* и *JavaScript*, Пайтон поддерживает перегрузку операторов. Любой встроенный функционал (операторы арифметических операций, возвращаемые встроенными функциями значения и т.п.) может быть описан заново в соответствии с задачей. Например, при описании некоторого класса можно задать метод *\_\_str\_\_*, отвечающий за значение, возвращаемое функцией *print()* в стандартный поток вывода при взятии объекта данного класса в качестве аргумента.

8. Синтаксические особенности.

Отличительной чертой Пайтона, заметной с первого знакомства с языком, являются синтаксические особенности. Отсутствуют, например, операторные и фигурные скобки, определяющие блоки кода; вместо них используются отступы.

Как и многие другие языки программирования, Пайтон поддерживает инкапсуляцию, полиморфизм, методы управления жизненным циклом объекта, метапрограммирование и многое другое.

Как объектно-ориентированный язык программирования, Пайтон поддерживает механизм одиночного и множественного наследования. При описании классов можно задавать так называемые родительские классы. В таком случае объект «дочернего» класса кроме собственных атрибутов и методов также «наследует» атрибуты и методы «родительских» классов.

Одним из главных недостатков языка Пайтон, как и других интерпретируемых языков, является низкое в сравнении с компилируемыми языками быстродействие. Компилятор обрабатывает весь код сразу и переводит его в машинный код, интерпретатор же делает это «на ходу», что, с одной стороны, даёт такие преимущества, как динамическая типизация, а с другой, – значительно (до 100 – 200 порядков по сравнению с компилируемыми языками) уменьшает скорость выполнения программы.

Другим недостатком, в отличие от таких языков, как *Ruby*, является отсутствие возможности модификации или изменения встроенных классов: *object*, *int*, *list*. Но и у такого подхода есть свои положительные стороны – уменьшение ресурсоёмкости программ.

### 3.1.2 Среда разработки Visual Studio Code

Visual Studio Code — редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений. Включает в себя отладчик, инструменты для работы с Git, подсветку синтаксиса, IntelliSense и средства для рефакторинга. Имеет широкие возможности для кастомизации: пользовательские темы, сочетания клавиш и файлы конфигурации.

Распространяется бесплатно, разрабатывается как программное обеспечение с открытым исходным кодом.

### 3.1.3 База данных sqlite

SQLite - это встроенная библиотека, которая реализует автономный, безсерверный, нулевой конфигурации, транзакционный механизм СУБД SQL. Это база данных, которая настроена на нуль, что означает, как и другие базы данных, которые вам не нужно настраивать в вашей системе.

Особенности данной базы данных:

* SQLite не требует отдельного процесса сервера или системы для работы (без сервера).
* SQLite поставляется с нулевой конфигурацией, что означает отсутствие необходимости в настройке или администрировании.
* Полная база данных SQLite хранится в одном кросс-платформенном диске.
* SQLite очень маленький и легкий, менее 400KiB полностью сконфигурированный или менее 250KiB с дополнительными функциями, опущенными.
* SQLite является автономным, что означает отсутствие внешних зависимостей.
* SQLite-транзакции полностью совместимы с ACID, обеспечивая безопасный доступ к нескольким процессам или потокам.
* SQLite поддерживает большинство функций языка запросов, найденных в стандарте SQL92 (SQL2).
* SQLite написан на ANSI-C и предоставляет простой и простой в использовании API.
* SQLite доступен в UNIX (Linux, Mac OS-X, Android, iOS) и Windows (Win32, WinCE, WinRT).

Python получил модуль sqlite3 в версии 2.5, что значит что можно создавать базу данных SQLite в любой версии Python, без необходимости скачивания дополнительных инструментов.

*3.1.4 Система контроля версий git*

*Git* – децентрализованная система управления версиями, то есть система, сохраняющая изменения в одном или нескольких файлах так, чтобы потом можно было восстановить определённые старые версии.

Задачи, решаемые *git*:

* вернуть файлы к прежнему виду;
* вернуть к прежнему состоянию весь проект;
* сравнить изменения с какого-то времени;
* увидеть, кто последним изменял модуль, который дал сбой, кто создал проблему;
* восстановить проект в любом состоянии.

Все файлы проекта хранились с использованием git на репозитории в gtihub.com.

## 3.2 Архитектура программной системы STOKES

Программа для расчета показателя преломления, а также определения типа материала покрытия.



1. 4.3 — Блок-схема комплекса программ расчета поляризационных параметров и оптических характеристик покрытий

Программная система состоит из следующих модулей.

Программа расчета параметров Стокса и коэффициента поляризации отраженного светового потока.

Программа для вычисления азимутального угла, угла эллиптичности и комплексного показателя поляризации светового потока.

Программа для расчета показателя преломления и определения материала покрытия.

Управляющая программа (приложения А, Б, В).

Программы функционирует в любой операционной системе, удовлетворяющей следующим требованиям:

При написании программ использовались следующие средства разработки:

* среда:Visual Studio Code;
* язык программирования: python 3;
* формат базы данных: sqlite3;
* построение графиков: matplotlib.

Разработанная программная система реализован в виде набора следующих программных модулей:

* модуль расчета параметров вектора Стокса;
* модуль работы с базой данных комплекса;
* модуль реализации алгоритма градиентного спуска;
* модуль вычисления нормированной функции Лапласа;
* модуль отображения графиков и диаграмм;
* модуль импорта данных;
* модуль расчета естественного излучения;
* модуль расчета рассеянного излучения;
* модуль расчета поляризации и определения типа покрытия;
* главный управляющий модуль.

Структура основных действий, выполняемых программой (структура программы), представлена на рисунке 4.4.



1. 4.4 — Связь программных модулей комплекса программ для вычисления поляризационных характеристик светового потока и расчета оптических параметров покрытий

## 3.2 Разработка схемы данных программной системы STOKES

### 3.3.1 Спецификация сущностей

На основе проведённого анализа предметной области выделим её основные объекты и сформируем соответствующие им сущности.

С целью наиболее наглядного и правильного обозначение наименований сущностей и соответствующих им атрибутов необходимо выполнение следующих правил:

* Идентификаторы сущностей, атрибутов и ключей должны отражать их семантику (проблемный смысл).
* Следует помнить, что эти идентификаторы выражаются комбинациями существительных, иногда в сочетании с прилагательными. Глаголы в этих именах не используются.
* Идентификатор может представлять собой слитную последовательность слов или их сокращений. Каждое слово (сокращение) в этой последовательности должно начинаться с большой буквы.
* Описания сущностей должны быть представлены таблицами и схемами.

Спецификация полученных сущностей и атрибутов представлена в таблице 3.1.

Таблица 3.1 — Описание сущностей

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Название сущности** | **Определение сущности** |
| 1 | Параметры расчета | Глобальные параметры, задаваемые при расчетах |
| 2 | Интервалы | Интервалы для параметров расчета для контроля того, что значение параметра не выходит за границы интервала |
| 3 | Коэффициенты материалов | Показатели отражения для металлических и диэлектрических поверхностей |
| 4 | Результаты расчета задачи 1, 2 | Хранение реузльтатов расчета задачи 1 и 2 |
| 5 | Результаты расчета задачи 3 | Хранение реузльтатов расчета задачи 3 |
| 6 | Данные признака C1 | Входные данные для задачи 3 по признаку 1 |
| 7 | Данные признака C2 | Входные данные для задачи 3 по признаку 2 |
| 8 | Данные признака C3 | Входные данные для задачи 3 по признаку 3 |

Рассмотрим полученные сущности более подробно (таблица 3.2.).

Сущность «Параметры расчета» имеет атрибуты:

* Код параметра;
* Наименование параметра;
* Значение параметра.

Сущность «Интервалы» имеет атрибуты:

* Имя параметра;
* Минимальное значение;
* Максимальное значение.

Сущность «Коэффициенты материалов» имеет атрибуты:

* Наименование материала;
* Минимальное и максимальное значение действительной части показателя;
* Минимальное и максимальное значение мнимой части показателя.

Сущность «Результаты расчета задачи 1, 2» имеет атрибуты:

* углы α и β;
* значения интенсивностей τ, φ, I для четрыех измерений;
* значения параметров вектора Стокса рассеянного излучения;
* значения параметров вектора Стокса естественного излучения;
* значение параметров угла α1 и β1;
* значение параметра χ.

Сущность «Результаты расчета задачи 3» имеет атрибуты:

* номер признака;
* порядковые номер параметра;
* обозначение и код параметра;
* значение параметра.

Сущность «Данные признака C1» имеет атрибуты:

* значение угла θ;
* значения параметров Q, U, V;
* вычисляемые атрибуты.

Сущность «Данные признака C2» имеет атрибуты:

* значение угла γ;
* значения параметров Q, U, V;
* вычисляемые атрибуты.

Сущность «Данные признака C3» имеет атрибуты:

* значение параметров угла α1 и β1;
* значения параметра С3;
* вычисляемые атрибуты.

Таблица 3.2 — Спецификация сущностей. Ключи и атрибуты

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Название и обозначение сущности | Ключ сущности  и его обозначение | Атрибуты сущности  и их обозначение |
| 1 | Параметры расчета (Пар) | Код параметра (Код) | Наименование параметра (НаимПар)  Значение параметра (ЗначПар)( |
| 2 | Интервалы (Инт) | Имя параметра (ИмяПар) | Минимальное значение (Мин);  Максимальное значение (Макс) |
| 3 | Коэффициенты материалов (Коэф) | Наименование материала (НаимМат) | Минимальное и максимальное значение действительной части показателя  (ДействМин, ДействМакс)  Минимальное и максимальное значение мнимой части показателя (МнимМин, МнимМакс) |
| 4 | Результаты расчета задачи 1, 2 (Расч12) | Угол α (Alfa)  Угол β (Betta) | значения интенсивностей τ, φ, I для четрыех измерений (Tau1..Tau4, Phi1...Phi4, I1...I4);  значения параметров вектора Стокса рассеянного излучения (J, Q, U, V, P);  значения параметров вектора Стокса естественного излучения (J0, Q0, U0, V0, P0);  значение параметров угла α1 и β1 (Alfa1, Beta1);  значение параметра χ (Hi) |
| 5 | Результаты расчета задачи 3 (Расч3) | номер признака (НомПр);  код параметра (КодПар) | порядковый номер параметра  (НомПарам);  обозначение параметра (НаимПарам)  значение параметра  (Знач Парам). |
| 6 | Данные признака C1 (ПрС1) | значение угла θ (Tetta) | значения параметра Q  значения параметра U  значения параметра V; |
| 7 | Данные признака C2  (ПрС2) | значение угла γ (Gamma); | значения параметра Q  значения параметра U  значения параметра V; |
| 8 | Данные признака C3  (ПрС3) | значение параметра угла α1 (Alfa1)  значение параметра и β1 (Beta1) | значения параметра С3 |

Схемы сущностей представляются в традиционной нотации, ключи в схеме выделяются подчеркиванием (таблица 3.3.).

Таблица 3.3. Схемы сущностей

|  |  |  |
| --- | --- | --- |
| № п/п | Название и обозначение сущности | Схема сущности |
| 1 | Параметры расчета (Пар) | **Пар** (Код, НаимПар, ЗначПар) |
| 2 | Интервалы (Инт) | **Инт** (ИмяПар, Мин, Макс) |
| 3 | Коэффициенты материалов (Коэф) | **Коэф** (НаимМат, МинДейств, МаксДейств, МинМним, МаксМним) |
| 4 | Результаты расчета задачи 1, 2 (Расч12) | **Расч12** (Alfa, Betta, Tau1..Tau4, Phi1...Phi4, I1...I4, J, Q, U, V, P, J0, Q0, U0, V0, P0, Alfa1, Beta1, HiДейств, HiМним) |
| 5 | Результаты расчета задачи 3 (Расч3) | **Расч3** (НомПр, КодПар, НомПарам, НаимПарам, ЗначПарам) |
| 6 | Данные признака C1 (ПрС1) | **ПрС1** (Tetta, Q, U, V) |
| 7 | Данные признака C2  (ПрС2) | **ПрС2** (Gamma, Q, U, V) |
| 8 | Данные признака C3  (ПрС3) | **ПрС3** (Alfa1, Beta1, C3) |

### 3.3.3 Спецификация связей

Спецификация связей представляет собой два перечня – список связей и список их идентификаторов .В разработанной программной системе связи между сущностями отсутствуют.

### 3.3.4 Формализация зависимостей

Отношение состоит из двух частей — заголовка отношения и тела отношения. Заголовок отношения — это аналог заголовка таблицы. Заголовок отношения состоит из атрибутов. Количество атрибутов называется степенью отношения. Тело отношения — это аналог тела таблицы. Тело отношения состоит из кортежей. Кортеж отношения является аналогом строки таблицы. Количество кортежей отношения называется мощностью отношения.

Отношения обладают следующими свойствами:

* В отношении нет одинаковых кортежей.
* Кортежи не упорядочены (сверху вниз).
* Атрибуты не упорядочены (слева направо).
* Все значения атрибутов атомарные.

При преобразовании ER-модели в реляционную модель данных использовались следующие правила:

* Степень связи 1:1, Класс принадлежности обязат:обязат => Количество таблиц 1, Первичный ключ (Л или П);
* Степень связи 1:1, Класс принадлежности необязат:обязат => Количество таблиц 2, Первичный ключ (Л, Л или П);
* Степень связи 1:1, Класс принадлежности необязат:необязат => Количество таблиц 3, Первичный ключ (Л, П, Л или П);
* Степень связи 1:М, Класс принадлежности обязат:обязат => Количество таблиц 2, Первичный ключ (Л или П);
* Степень связи 1:М, Класс принадлежности обязат(необязат):обязат => Количество таблиц 3, Первичный ключ (Л, П, П);
* Степень связи М:М, Класс принадлежности обязат(необязат): обязат(необязат) => Количество таблиц 2, Первичный ключ (Л, Л, ЛП);
* М-связей, Класс принадлежности обязат(необязат): обязат(необязат) => Количество таблиц М+1, Первичный ключ (К1, К2, …, КМ, К) [6];

На основании анализа сущностей составим таблицу с предварительных описанием отношений по вышеприведённым правилам (таблицы 3.5).

Таблица 3.5 — Описание предварительных отношений на основе сущностей

|  |  |  |
| --- | --- | --- |
| № п/п | Название и обозначение сущности | Отношение |
| 1 | Параметры расчета (Пар) | **Пар** (Код, НаимПар, ЗначПар) |
| 2 | Интервалы (Инт) | **Инт** (ИмяПар, Мин, Макс) |
| 3 | Коэффициенты материалов (Коэф) | **Коэф** (НаимМат, МинДейств, МаксДейств, МинМним, МаксМним) |
| 4 | Результаты расчета задачи 1, 2 (Расч12) | **Расч12** (Alfa, Betta, Tau1..Tau4, Phi1...Phi4, I1...I4, J, Q, U, V, P, J0, Q0, U0, V0, P0, Alfa1, Beta1, HiДейств, HiМним) |
| 5 | Результаты расчета задачи 3 (Расч3) | **Расч3** (НомПр, КодПар, НомПарам, НаимПарам, ЗначПарам) |
| 6 | Данные признака C1 (ПрС1) | **ПрС1** (Tetta, Q, U, V) |
|  | Данные признака C2  (ПрС2) | **ПрС2** (Gamma, Q, U, V) |
|  | Данные признака C3  (ПрС3) | **ПрС3** (Alfa1, Beta1, C3) |

На основе описания отношений (таблица 3.5, 3.6) определим первичные, внешние и потенциальные ключи отношений. Полученные результаты представим в виде таблицы (таблица 3.7.).

Таблица 3.7. Ключи отношений

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № п/п | Отношение | Первичный ключ | Внешний ключ | Потенциаль-ный ключ |
| 1 | **Пар** (Код, НаимПар, ЗначПар) | Код | - | НаимПар |
| 2 | **Инт** (ИмяПар, Мин, Макс) | ИмяПар | - | - |
| 3 | **Коэф** (НаимМат, МинДейств, МаксДейств, МинМним, МаксМним) | НаимМат | - | - |
| 4 | **Расч12** (Alfa, Betta, Tau1..Tau4, Phi1...Phi4, I1...I4, J, Q, U, V, P, J0, Q0, U0, V0, P0, Alfa1, Beta1, HiДейств, HiМним) | Alfa, Bett | - | - |
| 5 | **Расч3** (НомПр, КодПар, НомПарам, НаимПарам, ЗначПарам) | НомПр, КодПар | - | - |
| 6 | **ПрС1** (Tetta, Q, U, V) | Tetta | - | - |
| 7 | **ПрС2** (Gamma, Q, U, V) | Gamma | - | - |
| 8 | **ПрС3** (Alfa1, Beta1, C3) | Alfa1, Beta1 | - | - |

Все получившиеся отношения не являются подмножественными, не имеют дублирования, следовательно, отношения составлены правильно и являются итоговыми.

### 3.3.5 Определение таблиц базы данных

Далее каждое отношение трансформируется в таблицу (таблица 3.8.). Имена отношений становятся именами таблиц, а имена атрибутов – именами колонок.

Таблица 3.8. Отношения и таблицы

|  |  |  |
| --- | --- | --- |
| № п/п | Отношение | Таблица |
| 1 | **Пар** (Код, НаимПар, ЗначПар) | Constant |
| 2 | **Инт** (ИмяПар, Мин, Макс) | Intervals |
| 3 | **Коэф** (НаимМат, МинДейств, МаксДейств, МинМним, МаксМним) | MaterialRefraction |
| 4 | **Расч12** (Alfa, Betta, Tau1..Tau4, Phi1...Phi4, I1...I4, J, Q, U, V, P, J0, Q0, U0, V0, P0, Alfa1, Beta1, HiДейств, HiМним) | Calculation |
| 5 | **Расч3** (НомПр, КодПар, НомПарам, НаимПарам, ЗначПарам) | Calculation\_Task3 |
| 6 | **ПрС1** (Tetta, Q, U, V) | C1 |
| 7 | **ПрС2** (Gamma, Q, U, V) | C2 |
| 8 | **ПрС3** (Alfa1, Beta1, C3) | C3 |

Более подробная информация о свойствах получённых в таблице 3.8. таблиц представлена в таблице 3.9.

Таблица 3.9. Описание свойств и полей таблиц

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Имя таблицы** | **Имя**  **атрибута** | **Тип** | **Ключевое поле** | **Обязатель- ное поле** |
| Constant | Code | text | да | да |
| Name | text | нет | да |
| Value | real | нет | да |
| Intervals | FieldName | text | да | да |
| FieldMin | real | нет | да |
| FieldMax | real | нет | да |
| MaterialRefraction | MaterialName | text | да | да |
| ReValueMin | real | нет | да |
| ReValueMax | real | нет | да |
| ImValueMin | real | нет | да |
| ImValueMax | real | нет | да |
| Calculation | Alfa | real | да | да |
| Beta | real | да | да |
| Tau1 | real | нет | да |
| Phi1 | real | нет | да |
| I1 | real | нет | да |
| Tau2 | real | нет | да |
| Phi2 | real | нет | да |
| I2 | real | нет | да |
| Tau3 | real | нет | да |
| Phi3 | real | нет | да |
| I3 | real | нет | да |
| Tau4 | real | нет | да |
| Phi4 | real | нет | да |
| I4 | real | нет | да |
| J | real | нет | да |
| Q | real | нет | да |
| U | real | нет | да |
| V | real | нет | да |
| P | real | нет | да |
| J0 | real | нет | да |
| Q0 | real | нет | да |
| U0 | real | нет | да |
| V0 | real | нет | да |
| P0 | real | нет | да |
| Alfa1 | real | нет | да |
| Beta1 | real | нет | да |
| Hi | real | нет | да |
| Calculation\_Task3 | Sign | real | да | да |
| Number | real | нет | да |
| ParameterName | real | нет | да |
| ParameterValue | real | нет | да |
| C1 | Tetta | real | да | да |
| Q | real | нет | да |
| U | real | нет | да |
| V | real | нет | да |
| C2 | Gamma | real | да | да |
| Q | real | нет | да |
| U | real | нет | да |
| V | real | нет | да |
| C3 | Gamma | real | да | да |
| Alfa | real | нет | да |
| Beta | real | нет | да |

### 3.3.6 Схема баз данных

*Схемой реляционной базы* данных называется набор заголовков отношений, входящих в базу данных [11].

Схема призвана визуализировать (наглядно представить) состав отношений (таблиц) и взаимосвязей между ними. Каждое отношение изображается в виде прямоугольника, поделенного на части. Каждая часть символизирует атрибут и содержит его наименование. Стрелки показывают ссылки дочерних отношений на родительские [6].

В схеме данных устанавливаются параметры обеспечения целостности связей в базе данных [6]. Схема базы данных приведена на рисунке 3.2.

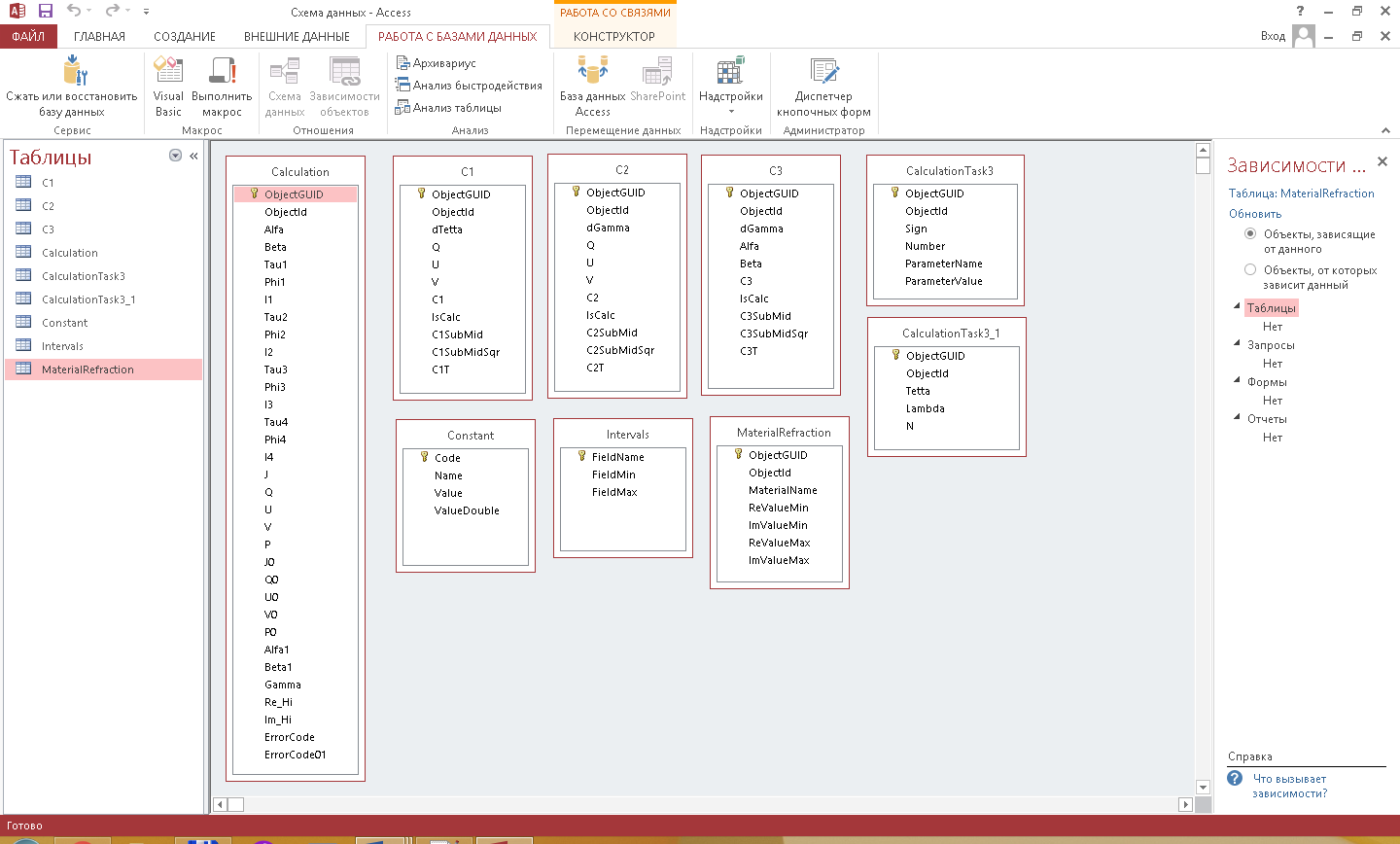


Рисунок 3.4 — Схема базы данных в Microsoft Access

### 3.3.7 Нормализация баз данных

С целью построения реляционной базы без дублирования данных, с обеспечением возможности поддержания целостности при внесении изменений докажем нормальность полученных отношений и в случае необходимости приведём их к четвёртой нормальной форме с помощью алгоритма нормализации.

*Нормализация* — разбиение одной таблицы на две или более, обладающих лучшими свойствами при включении, изменении и удалении данных [P].

*Алгоритм нормализации состоит из следующих этапов*

1. Привести отношение в ***1НФ*** (первая нормальная форма), нужно исключить повторяющиеся группы и многозначные поля (декомпозиция или выравнивание).
2. Чтобы привести отношение в ***2НФ*** (вторая нормальная форма), нужно разбить отношения на проекции для исключения приводимых функциональных зависимостей (от части ключа). Отметим, что поле *В* таблицы функционально зависит от поля *А* той же таблицы в том и только в том случае, когда в любой заданный момент времени для каждого из различных значений поля *А* обязательно существует только одно из различных значений поля *В*. Отметим, что здесь допускается, что поля *А* и *В* могут быть составными.
3. Чтобы привести отношение в ***3НФ***, нужно получившееся отношение разбить на проекции для исключения транзитивных функциональных зависимостей. Функциональная зависимость *R.X  R.Y* называется транзитивной, если существует такой атрибут *Z*, что имеются функциональные зависимости *R.X  R.Z* и *R.Z  R.Y* и отсутствует - *R.Z → R.X*.
4. Чтобы привести отношение в ***НФБК*** (нормальная форма Бойса-Кодда), нужно разбить на проекции для исключения любых функциональных зависимостей, в которых детерминант не является ключом. Детерминант - любой атрибут, от которого полностью функционально зависит некоторый другой атрибут [6].
5. Чтобы привести отношение в ***4НФ*** (четвёртая нормальная форма), нужно разбить его на проекции для исключения многозначных зависимостей. При этом атрибуты (множества атрибутов) *Y* и *Z* многозначно зависятот *Х* (*Х→→Y|Z*), тогда и только тогда, когда из того, что в отношении *R* содержатся кортежи *r1 = (x,y,z1)* и *r2 = (x,y1,z)* следует, что в отношении *R* содержится также и кортеж к *r3 = (x,y,z)*
6. Чтобы привести отношение в ***5НФ*** (пятая нормальная форма), нужно разбить на проекции для исключения любых зависимостей соединения, которые не подразумеваются потенциальными ключами. Имеют место зависимости специального вида, когда отношение не может быть подвергнуто декомпозиции без потерь на две проекции, но может быть декомпозировано на большее число проекций. Такие зависимости называются зависимостями соединения и являются обобщением понятия многозначной зависимости. На практике приводить отношения к 5НФ необязательно.

*Проверка полученных итоговых отношений (таблица 3.6) на принадлежность к 4НФ с помощью алгоритма нормализации показала следующие результаты:*

1. Все приведённые отношения находятся в 1НФ, так как не содержат повторяющиеся группы и многозначные поля, то есть удовлетворяют следующим свойствам:

* в отношении нет одинаковых картежей;
* картежи не упорядочены;
* атрибуты не упорядочены и различаются по наименованию;
* все значения атрибутов атомарные.

2. Все приведённые отношения находятся в 2НФ, так как находятся в 1НФ, и не содержит неключевых атрибутов, зависящих от части сложного ключа. Отметим, что если первичный ключ отношения является простым, то отношение автоматически находится в 2НФ.

3. Все приведённые отношения находятся в 3НФ, так как находятся в 2НФ и все их неключевые атрибуты взаимонезавичимы. Атрибуты называются взаимонезависимыми, если не один из них не является функционально зависимым от другого.

4. Все приведённые отношения находятся в НФБК, так как детерминанты всех функциональных зависимостей являются потенциальными ключами. Если отношение находится в НФБК, то оно автоматически находится в 3НФ. Если отношение имеет два не только первичный, но и потенциальный ключ, и других атрибутов в нём нет, то для анализа отношения можно ограничиться анализом 3НФ и не прибегать к НФБК.

5. Все приведённые отношения находятся в 4НФ, так как находятся в НФБК и не содержит нетривиальных многозначных зависимостей.

Таким образом, все итоговые отношения, полученные в результате тщательно проведённого анализа, находятся в 4НФ и обеспечивают адекватность предметной области, целостность информации, высокую скорость выполнения процедуры обновления данных, а следовательно и гибкую структуру хранимых данных.

## 3.3 Руководство программиста программной системы STOKES

## 3.4 Тестирование программной системы STOKES