Отчёт о модификации проекта «Изображение проекции полиэдра»

И.Д. Соловьев

Е.А. Роганов

А.И. Александров

26.04.2024

Table of Contents

## Постановка задачи

### Точная постановка задачи

Назовём точку в пространстве «хорошей», если её проекция находится строго внутри окружности но строго вне окружности Модифицируйте эталонный проект таким образом чтобы определялась и печаталась следующая характеристика полиэдра: сумма длин рёбер, оба из концов которых — «хорошие» точки.

### Интерпретация задачи и идея решения

В данной задаче требуется рассматривать проэкции вершин полиэдра на плоскость , и складывать длины таких ребер полиэдра, оба конца которых попадают в заданное кольцо, то есть удовлетворяют неравенству:

где — рассматриваемая точка. Также необходимо реализовать графический интерфейс, отображающий заданное кольцо для визуального контроля правильности работы модификации.

## Подробное описание решения задачи на модификацию

1. В класс R3 был добвален метод, проверяющий, является ли переданная точка «хорошей», при помощи подстановки ее и координат в рассмотренное при постановке задачи неравенство.
2. Было принято решение выявлять ребра, оканчивающиеся «хорошими» точками на этапе задания ребер для каждой грани в методе \_\_init\_\_ класса Polyedr
3. Так как некоторые ребра могут принадлежать одной или двум граням, в зависимости от конфигурации полиэдра, то необходио проверять, было ли ранее посчитано рассматриваемое ребро, или нет. Самой простой с точки зрения написания кода проверкой в данном случае является двумерный массив смежности. Это «таблица», по столбцам и строкам которой расположены все вершины полиэдра, ячейки заполнены единицами. Когда вычисляется сумма очередного ребра, ячейка на пересечении строки и столбца с соответствующими вершинами начала и конца ребра заполняется нулем.
4. Также в метод \_\_init\_\_ класса Polyedr был добавлен дополнительный массив вершин, с непреобразованными координатами, чтобы в полученной суммарной длине не учитывался коэффициент гомотетии.
5. Вычисление длины рёбер было реализовно с помощью операции взятия квадратного корня из скалярного произведения.
6. Была добавлена отправка в стандартный вывод информации о длинне ребер, подходящих под условие.
7. В класс Polyedr добавляется метод print\_good\_sum(self), возвращающий искомую сумму длин рёбер с «хорошими» точками на концах.
8. В методе draw класса Polyedr была добавлена отрисовка кольца и проекций точек, при этом учтен коэффициент гомотетии для корректного отображения кольца.
9. Были написаны соответствующие тесты, иллюстрирующие правильность работы всех написанных методов и модификации в целом.

## Команды, использовавшиеся при получении итоговых отчетов в заданных форматах

Здесь pdf.latex и HTML.html5 — переименованные файлы шаблонов.

Команда для получения отчета в формате pdf, с использованием заданного шаблона.

pandoc --template pdf.latex -s --toc --lua-filter ./include-code-files.lua report.md -o report.pdf

Команда для получения отчета в формате html, с использованием заданного шаблона.

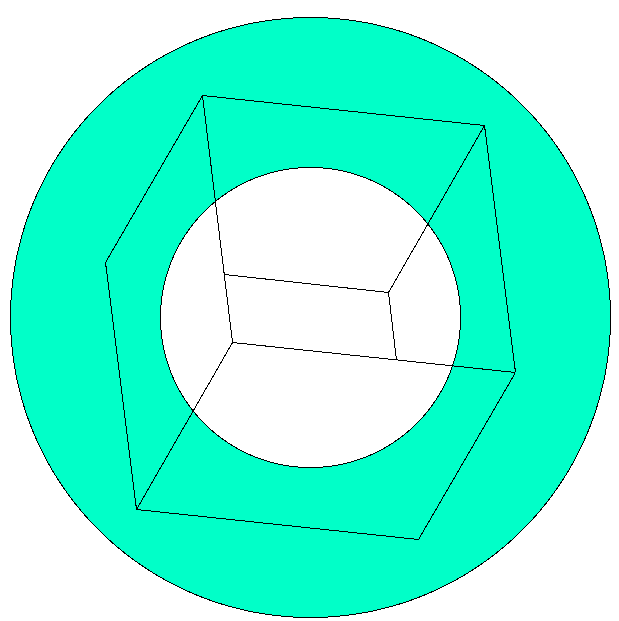
pandoc -o report.html -f markdown -t html -s --toc --lua-filter ./include-code-files.lua --mathjax --template html.html5 report.md

Команда для получения отчета в формате docx.

pandoc -s report.md --toc --lua-filter ./include-code-files.lua -o report.docx

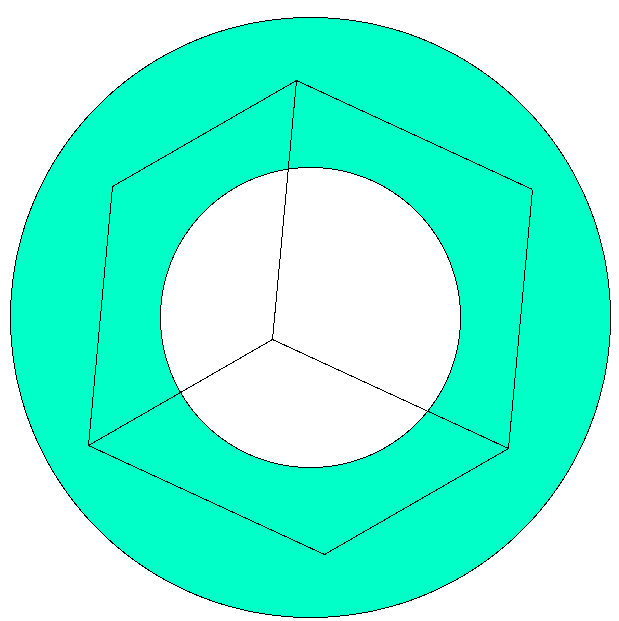
## Примеры тестов для поддтверждения корректности работы модификации

* Коробка, шесть из восьми вершин которой попадают в заданное кольцо.



@classmethod  
def setUpClass(self):  
 fake\_file\_content = """200.0 60.0 -140.0 60.0  
8 5 20  
-1 -1 1  
-1 1 1  
1 1 1  
1 -1 1  
-1 -1 -1  
-1 1 -1  
1 1 -1  
1 -1 -1  
4 1 2 3 4  
4 5 6 2 1  
4 3 2 6 7  
4 3 7 8 4  
4 1 4 8 5"""  
 fake\_file\_path = 'data/poly\_box.geom'  
 with patch('shadow.polyedr.open'.format(\_\_name\_\_),  
 new=mock\_open(read\_data=fake\_file\_content)) as \_file:  
 self.poly\_box = Polyedr(fake\_file\_path)  
 \_file.assert\_called\_once\_with(fake\_file\_path)  
  
  
 # polyedr perimetr  
  
def test\_box\_perimetr(self):  
 self.assertEqual(self.poly\_box.good\_perimetr, 12)

* Куб, шесть из восьми вершин которого попадают в заданное кольцо.



@classmethod  
def setUpClass(self):  
 fake\_file\_content = """200.0 45.0 45.0 30.0  
8 6 24  
-1 -1 1  
-1 1 1  
1 1 1  
1 -1 1  
-1 -1 -1  
-1 1 -1  
1 1 -1  
1 -1 -1  
4 1 2 3 4  
4 5 6 2 1  
4 3 2 6 7  
4 3 7 8 4  
4 1 4 8 5  
4 8 7 6 5"""  
 fake\_file\_path = 'data/poly\_cube.geom'  
 with patch('shadow.polyedr.open'.format(\_\_name\_\_),  
 new=mock\_open(read\_data=fake\_file\_content)) as \_file:  
 self.poly\_cube = Polyedr(fake\_file\_path)  
 \_file.assert\_called\_once\_with(fake\_file\_path)  
  
  
 # double edges (with two facets) can be countable  
  
def test\_cube\_perimetr(self):  
 self.assertEqual(self.poly\_cube.good\_perimetr, 12)