

Московский авиационный институт  
(национальный исследовательский университет)

Институт информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

Курсовая работа по курсу «Компьютерная графика»  
Каркасная визуализация заданной поверхности

Студент: Семенов И.М.  
Преподаватель: Морозов А.В.  
Группа: М8О-306Б-18  
Вариант: 4  
Дата:  
Оценка:  
Подпись:

Москва, 2020

## Условие

Составить и отладить программу, обеспечивающую каркасную визуализацию порции поверхности заданного типа. Должна быть предусмотрена возможность изменения точности аппроксимации и параметров поверхности.

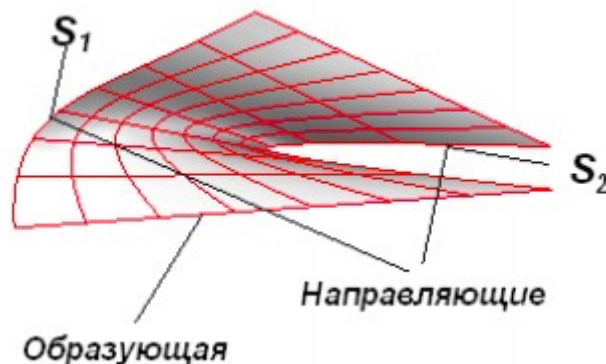
## Вариант 4

Линейчатая поверхность (направляющие — кубические кривые Безье 3D)

## Теоретические сведения

### Линейчатая поверхность. Образующая и направляющая

Линейчатая поверхность получается в результате пространственного движения отрезка прямой переменной длины. Сам отрезок в этом случае называется образующей линией, а траектории его концов — направляющими линиями. Можно считать, что образующая линейно переводит точки одной направляющей линии в точки другой.



Линейчатая поверхность описывается следующим уравнением:

$$Q(u, v) = S_1(u) \cdot (v - 1) + S_2(u) \cdot v$$

Параметрические линии линейчатой поверхности в одном направлении — отрезки, в другом — кривые, линейно трансформирующиеся от одной направляющей к другой. Билинейная поверхность является частным случаем линейчатой.

## Кривая Безье

В этой работе, в качестве направляющих для линейчатой поверхности, используются кривые Безье.

Кривые Безье впервые были предложены в 60-х годах, и применялись тогда для проектирования кузовов автомобилей.

В общем случае, кривая Безье задается векторным уравнением

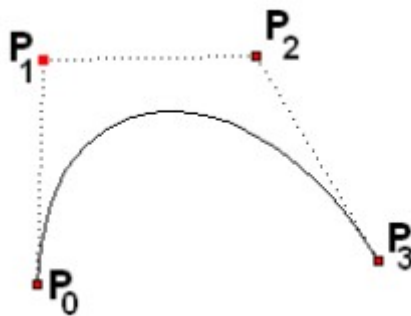
$$P(t) = \sum_{i=0}^N P_i J_{N,i}(t), \text{ где } J_{N,i}(t) = \frac{N!}{i!(N-i)!} t^i (1-t)^{N-i}, \quad 0 \leq t \leq 1,$$

а  $P_i$  — вершины определяющего многоугольника.

Кривые Безье обладают следующими свойствами:

- 1) Степень кривой  $N$  на единицу меньше числа вершин определяющего многогранника. Для двух точек кривая Безье превращается в отрезок прямой.
- 2) Начинается в  $P_0$  и кончается в  $P_N$ .
- 3) Вектор  $P_0P_1$  сонаправлен с вектором касательной  $P'(0)$  к кривой в ее начальной точке  $P_0$ .
- 4) Целиком лежит внутри выпуклой оболочки определяющего многоугольника.

В нашем случае, кривые имеют степень 3 (строятся по 4-м точкам).



## Описание проекта

Для реализации был использован язык C++ и библиотеки OpenGL, GLFW, ImGui, GLM.

После запуска программы, пользователю предоставляется два окна, в одном из которых находятся элементы управления, а во втором отображается непосредственно поверхность.

## Структура проекта

Проект содержит следующие файлы:

- *main.cpp* — создание окон, пользовательского интерфейса, отрисовка поверхности по ее точкам и изменение ее внешнего вида
- *math.cpp*, *math.h* — содержит функцию вычисления факториала для кривой Безье
- *Cube.cpp*, *Cube.h* — класс, содержащий единственную функцию, возвращающую точки полигонов куба.
- *Bezier.cpp*, *Bezier.h* — класс рассчитывающий точки кривой Безье. Также отслеживает изменение параметров кривой, выполняя фактический перерасчет при следующем запросе точек.
- *Surface.cpp*, *Surface.h* — класс, рассчитывающий точки линейчатой поверхности. Содержит две кривых Безье, а заодно и методы доступа к ним, а так же метод, для расчета точек полигонов поверхности.
- *shaders* — папка с вершинными и фрагментными шейдерами.
- *imgui* — папка с библиотекой для работы с интерфейсом.
- *ShaderLoader.cpp*, *ShaderLoader.h* — класс для загрузки шейдеров, использования их и установки uniform переменных.
- *CMakeLists.txt* — инструкции для сборки проекта.

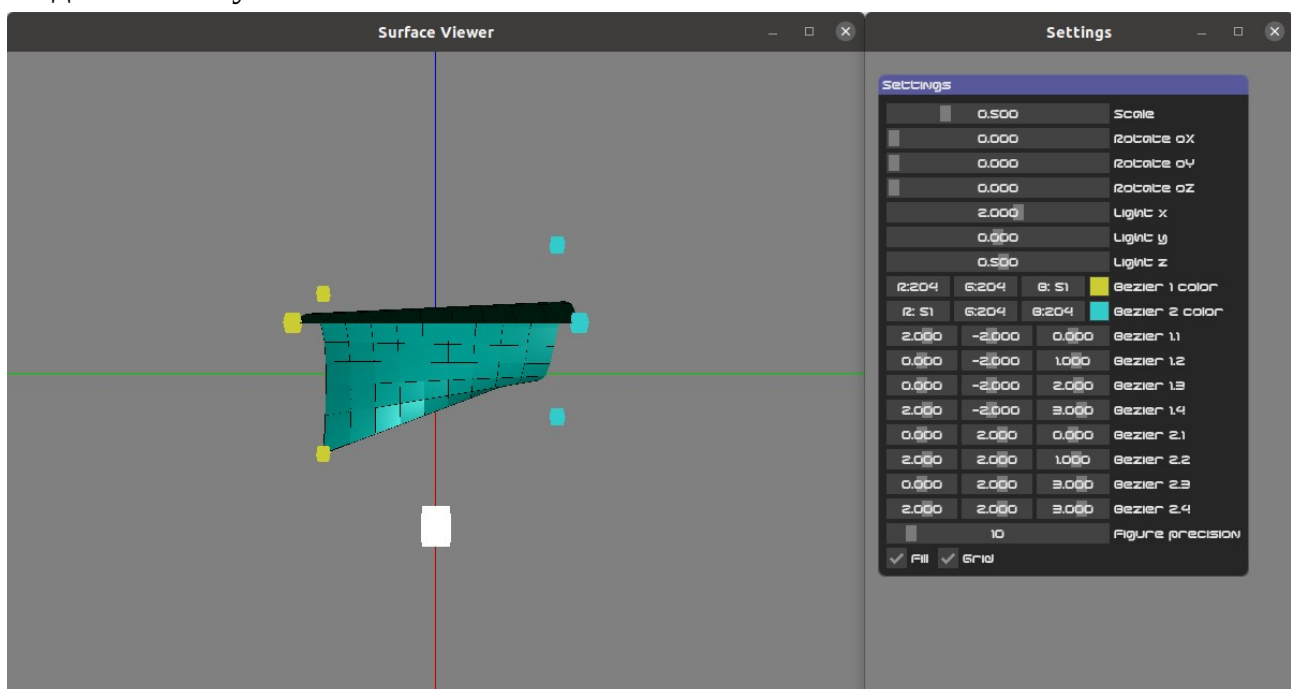
## Описание работы программы

Можно выделить этапы работы программы

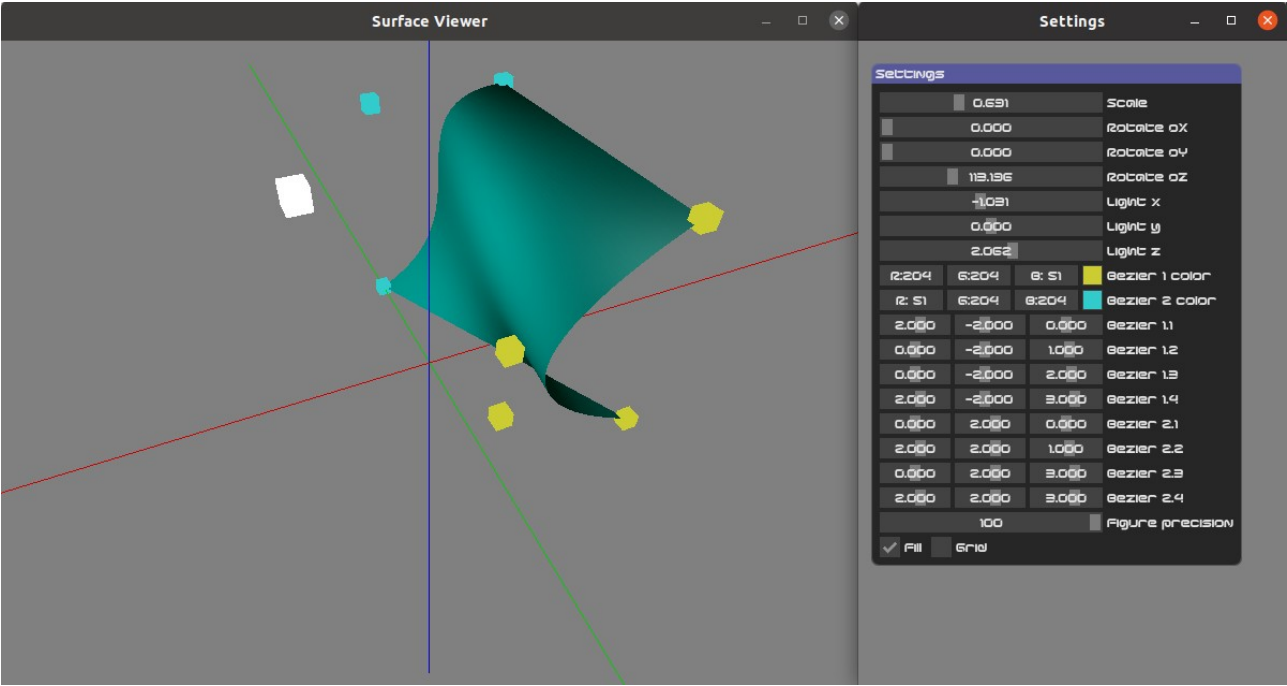
- Сначала происходит инициализация GLFW, и создание окон
- Инициализация GLAD
- Подгрузка и компиляция шейдеров из файлов
- Создание VBO и VAO, буферизация точек поверхности(инициализированной начальными значениями) и куба(источника света)
- Инициализация ImGui
- Работа основного цикла
  - Обработка действий пользователя с помощью элементов ImGui
  - На основе действий пользователя происходит перерасчет точек поверхности(если необходимо), матриц и uniform-параметров для шейдеров.
  - Непосредственно отрисовка поверхности, ее ключевых точек, источника света. Применение освещения к поверхности.

## Демонстрация работы программы

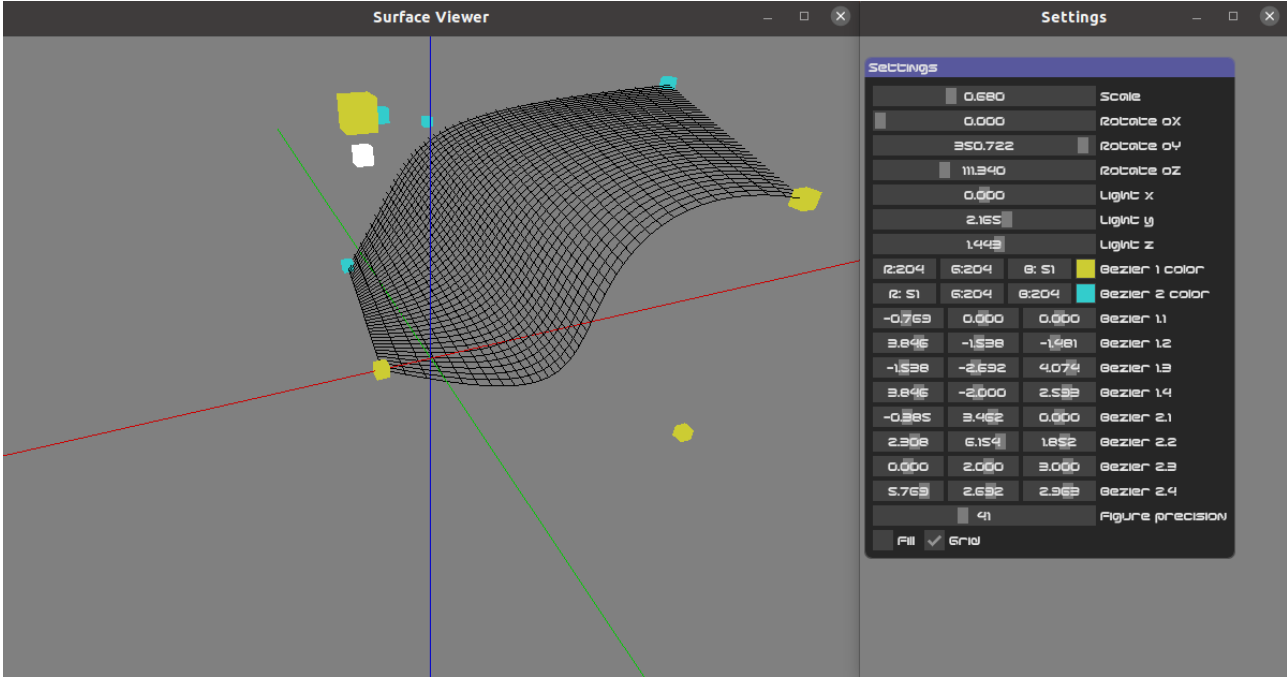
Вид после запуска



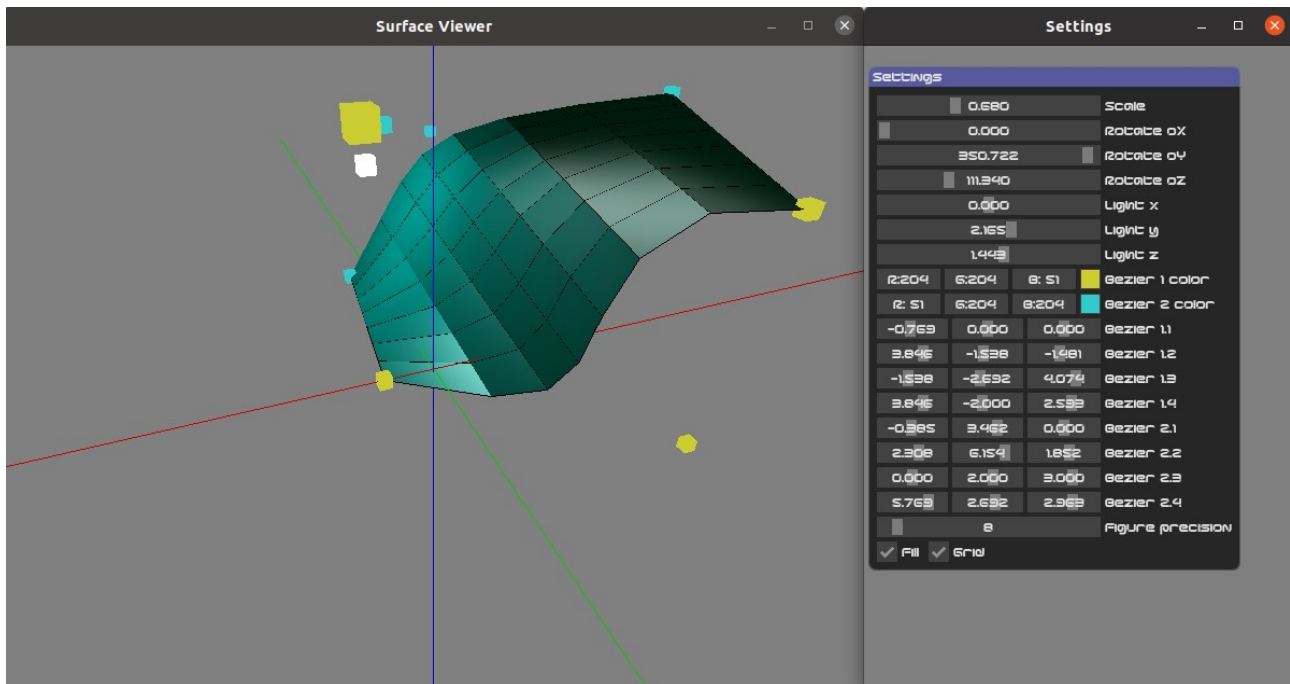
После поворота, перемещения источника света и увеличения точности аппроксимации



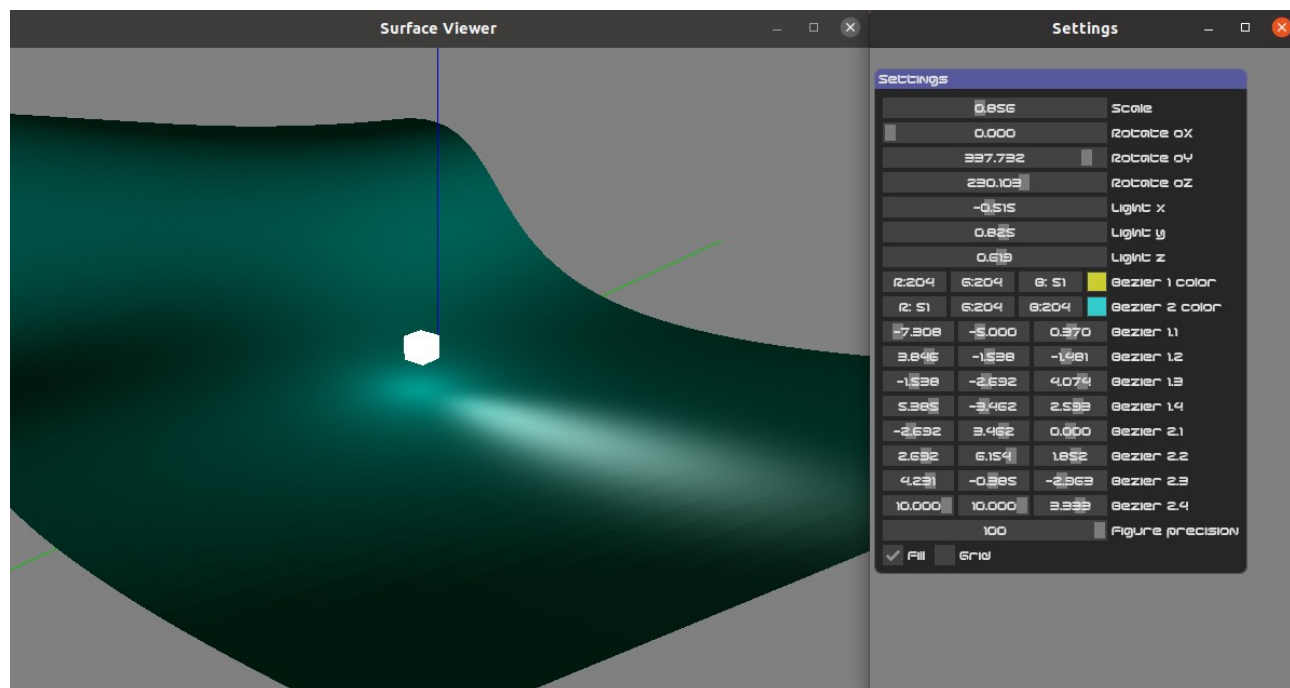
Каркасное представление поверхности



Та же самая конфигурация, но с уменьшенной точностью. Отображаются одновременно каркасная визуализация и полигональная.



Еще одна конфигурация



## **Вывод**

При помощи OpenGL была реализована программа, для моделирования линейчатой поверхности с двумя кривыми Безье в виде направляющих. В данную программу можно достаточно быстро добавить реализации других кривых и поверхностей, а механизм наследования упростит их использование.



