

Московский авиационный институт
(национальный исследовательский университет)

Институт №8 «Информационные технологии и прикладная
математика»

Кафедра 806 «Вычислительная математика и
программирование»

Лабораторная работа №7 по курсу «Компьютерная графика»

Студент: М. А. Инютин
Преподаватель: А. В. Морозов
Группа: М8О-307Б-19
Дата:
Оценка:
Подпись:

Москва, 2022

Построение плоских полиномиальных кривых

Задача: Написать программу, строящую полиномиальную кривую по заданным точкам. Обеспечить возможность изменения позиции точек и, при необходимости, значений касательных векторов и натяжения.

Вариант задания: В-сплайн. $n = 6$, $k = 3$. Узловой вектор равномерный.

1 Описание

Кубическая В-сплайновая кривая определяется уравнением:

$$r(t) = \frac{1-t^3}{6} \cdot P_0 + \frac{3t^3-6t^2+4}{6} \cdot P_1 + \frac{-3t^3+3t^2+3t+1}{6} \cdot P_2 + \frac{t^3}{6} \cdot P_3.$$

Матричная запись параметрического уравнения кривой:

$$r(t) = P \cdot M \cdot T, \quad P = \begin{pmatrix} x(t) \\ y(t) \end{pmatrix}, \quad M = \frac{1}{6} \cdot \begin{pmatrix} 1 & -3 & 3 & -1 \\ 4 & 0 & 6 & 3 \\ 1 & 3 & 3 & -3 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$
$$T = \begin{pmatrix} t^0 \\ t^1 \\ t^2 \\ t^3 \end{pmatrix}, \quad P = (P_0 \quad P_1 \quad P_2 \quad P_3).$$

Матрица M называется базисной матрицей В-сплайновой кривой.

Составная В-сплайновая кривая определяется объединением элементарных В-сплайновых кривых, составленных из 4 точек. То есть для 6 точек В-сплайн составляется из трёх элементарных В-сплайновых кривых. В таком случае уравнение кривой будет иметь вид:

$$r_i(t) = (P_{i-1} \quad P_i \quad P_{i+1} \quad P_{i+2}) \cdot M \cdot \begin{pmatrix} (t-t_i)^0 \\ (t-t_i)^1 \\ (t-t_i)^2 \\ (t-t_i)^3 \end{pmatrix}.$$

2 Исходный код

Файл *Curve.cs* содержит код для генерации В-сплайна и методов изменения её параметров. В программе предусмотрено перемещение, добавление и удаление контрольных точек.

```
1 Matrix4D ConvertToMatrix(Vector4D v1, Vector4D v2, Vector4D v3, Vector4D v4)
2 {
3     List<Vector4D> tmp = new List<Vector4D>(4);
4     tmp.Add(v1);
5     tmp.Add(v2);
6     tmp.Add(v3);
7     tmp.Add(v4);
8     Matrix4D res = new Matrix4D();
9     for (int i = 0; i < 4; ++i)
10    {
11        res[0, i] = tmp[i].X;
12        res[1, i] = tmp[i].Y;
13        res[2, i] = tmp[i].Z;
14        res[3, i] = tmp[i].W;
15    }
16    return res;
17 }
18
19 public void CalcCurve()
20 {
21     CountData = STEPS * (CountPoints - 3);
22     Data = new List<Vector4D>(CountData);
23     for (int i = 0; i < CountPoints - 3; ++i)
24     {
25         Vector4D p0 = Points[i];
26         Vector4D p1 = Points[i + 1];
27         Vector4D p2 = Points[i + 2];
28         Vector4D p3 = Points[i + 3];
29         Matrix4D tmp = ConvertToMatrix(p0, p1, p2, p3) * basisMatrix;
30
31         for (int j = 0; j < STEPS; ++j)
32         {
33             double t = STEP * j;
34             Vector4D tt = new Vector4D(1, t, Math.Pow(t, 2), Math.Pow(t, 3));
35             Data.Add(tmp * tt);
36         }
37     }
38 }
```

Geometry.cs содержит классы векторов и матрицы, методы для работы с ними. В *Misc.cs* заданы константы, используемые во всей программе. Вся работа с интерфейсом, компиляция шейдеров, создание буфера вершин описаны в *MainWindow.cs*.

3 Демонстрация работы программы

Интерфейс программы очень простой, предусмотрена возможность выключить отрисовку контрольных точек и ломаной.

