

Лабораторная работа №7 по курсу компьютерной графики

Построение плоских полиномиальных кривых

Выполнил студент группы М8О-308Б-18 Черемисинов Максим

Условие

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

Вариант: 10

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

Материалы

1. [Статья](#) про В-кривые
2. Серия лекций Мартынюка В.А. про В-кривые с канала ["Робототехника и комплексная автоматизация"](#)

Описание программы

Программа написана на `C++17` с использованием библиотеки `SFML` для работы с графикой.

В-кривая степени k ($2 \leq k \leq n$) с $n - 1$ контрольными точками строится на основе $n + 1$ В-сплайнов степени k (индексация начинается с 0). В-кривая строится с использованием узлового вектора t , который в ситуации, когда он равномерный и однородный, в одном из частных случаев выглядит следующим образом:

$$t = [0, 0, \dots, 0, 1, 2, 3, 3, \dots, 3], 0 \leq t[i] \leq n$$

В его начале и конце k раз повторяется значение 0 и $n - k + 2$ соответственно. Между ними - числа $[1, 2, \dots, n - k + 1]$.

В-кривая для контрольных точек (P_0, P_1, \dots, P_n) вычисляется по формуле:

$$r(t) = \sum_{i=0}^n N_i^k(t) P_i$$

$N_i^k(t)$ - В-сплайн степени k для узлового значения $t[i]$.

$N_i^0(t)$ вычисляется по формуле:

$$N_i^0(t) = \begin{cases} 1, & \text{если } t_i \leq t \leq t_{i+1} \\ 0, & \text{в остальных случаях} \end{cases}$$

$N_i^k(t)$ вычисляется по формуле Кокса-де-Бура:

$$N_i^k(t) = \frac{t - t_i}{t_{i+k} - t_i} N_i^{k-1}(t) + \frac{t_{i+k-1} - t}{t_{i+k-1} - t_{i+1}} N_{i+1}^{k-1}(t)$$

В программной реализации вместо параметра t используется несколько (в зависимости от точности кривой) его числовых значений из соответствующих отрезков вида $[t_i, t_{i+1}]$.

Структура проекта

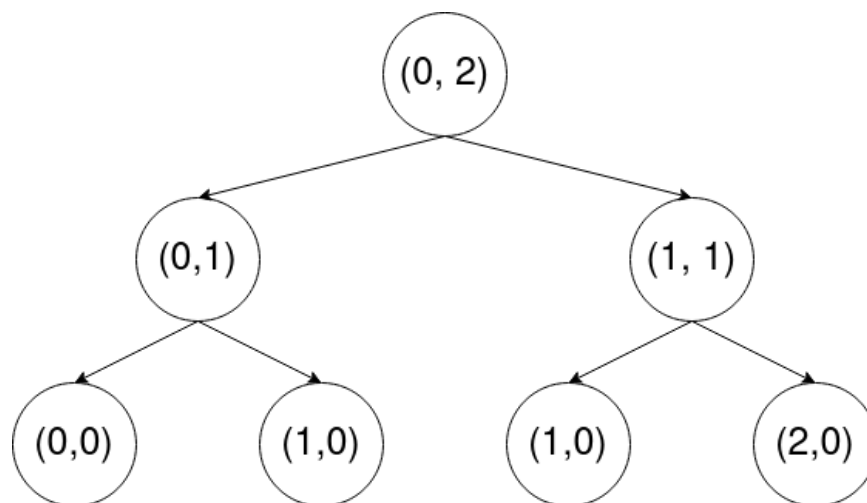
Проект состоит из 8 файлов.

1. `curves_math.h` — кривые, их вычисление
2. `automate.h` — "автоматы", прослойка между кривыми и их отрисовкой в `SFML`
3. `sfml_extra.h` — кнопки, чекбоксы (в `SFML` нет встроенных)
4. `sfml_help.h` — вспомогательные функции для конвертации типов данных из `STL` в типы данных `SFML`

5. `main.cpp` — создание окна, кнопок, чекбоксов
6. `myprint.h` — отладочная печать
7. `Makefile` — для сборки проекта

Реализация вычисления В-кривой

Вычисление N_0^2 можно изобразить следующим образом:



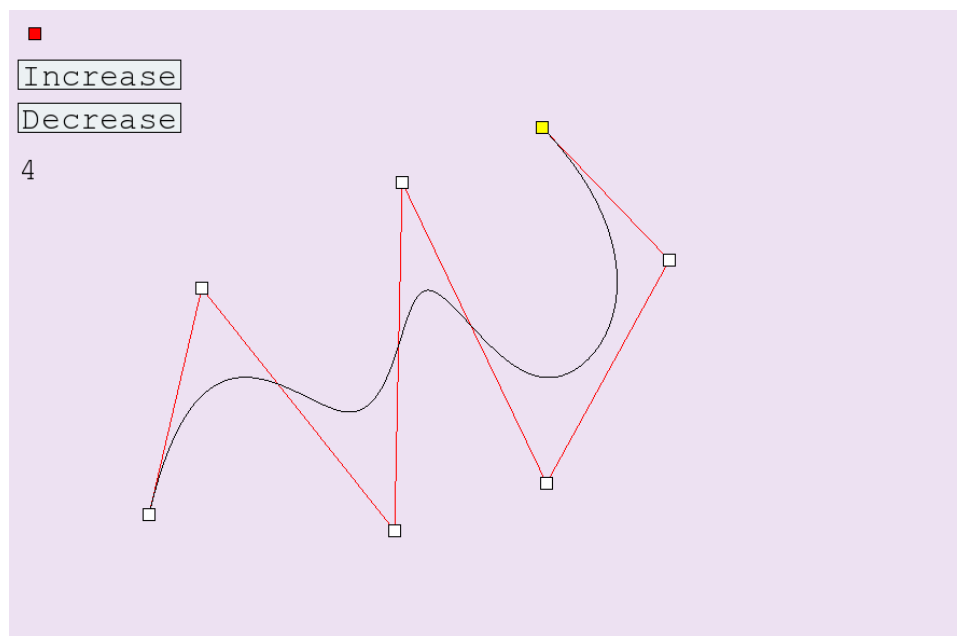
Вычисление происходит с инициализации всех нижних элементов $N_i^0, 0 \leq i \leq n$. После чего вычисляются элементы $N_i^1, 0 \leq i \leq n - 1$ и так далее.

Каждый В-сплайн N_i^j может состоять из нескольких частей, поэтому при объединении двух В-сплайнов проверяется, есть ли уже в результирующем В-сплайне интервал, который собирается быть добавлен.

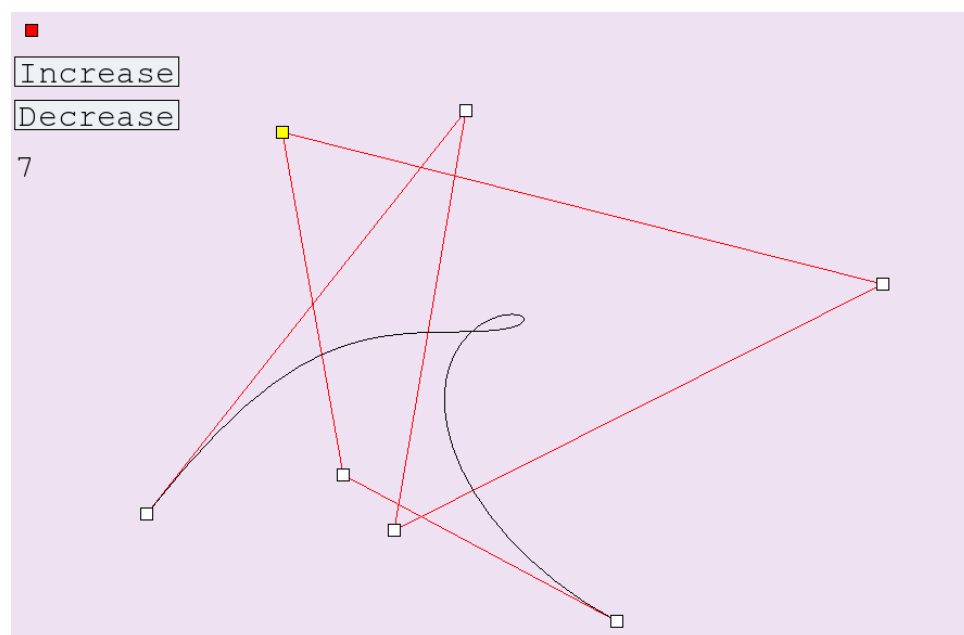
Демонстрация работы

В левом верхнем углу диалогового окна есть чекбокс (рисовать или не рисовать линии между ключевыми точками), две кнопки (повышение и понижение степени В-кривой) и сама степень.

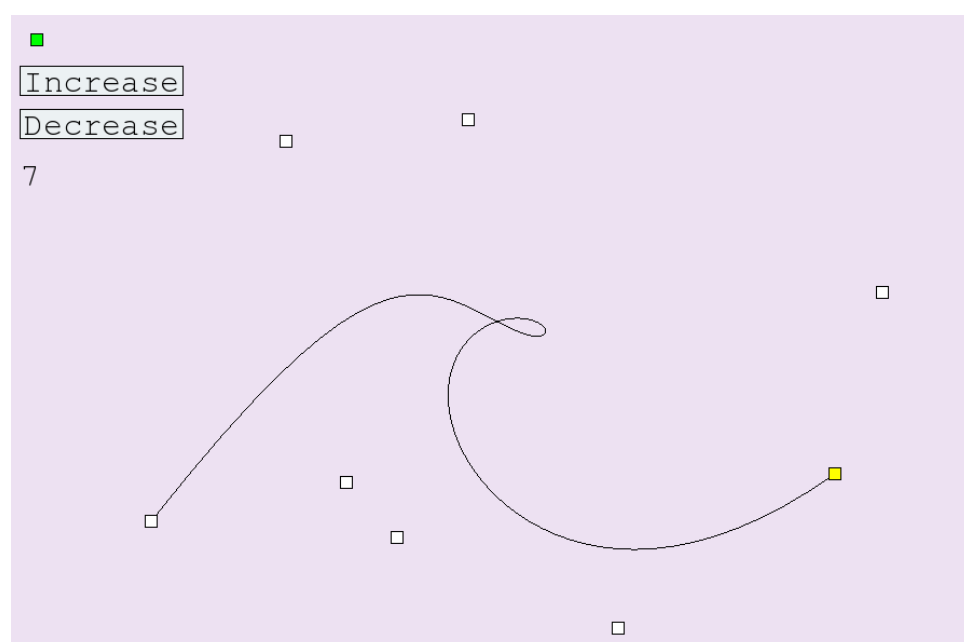
Ключевые точки задаются пользователем на самом окне.



Есть возможность менять положение каждой ключевой точки, в том числе и удалять их. При удалении для новой В-кривой устанавливается максимально возможная степень.



В-кривая без линий между ключевыми точками:



Выводы

В-кривые разрабатывались как замена существующим кривым Безье, главным недостатком которых являлась невозможность их локальной модификации и жесткая привязка степени полинома к количеству точек. В-кривые избавлены от всех этих недостатков: степень может быть задана пользователем, также присутствует локальная модификация кривой (чем ниже степень, тем более локальными получаются модификации).

Данную реализацию В-кривых можно улучшить, если не вычислять кривую полностью заново при изменении одной ключевой точки, а вычислять лишь изменяющиеся её части. Также возможно несколько обобщить реализацию В-кривой, добавив возможность работы с произвольным узловым вектором.