

# Ходограф

Курсова работа по CAGD 2020/2021

Хиндату Сани, фн. 81948

## I. Описание на алгоритъма

Кривата на Безие, генерирана от контролните точки, въведени от потребителя, се изчислява по алгоритъма на de Casteljau.

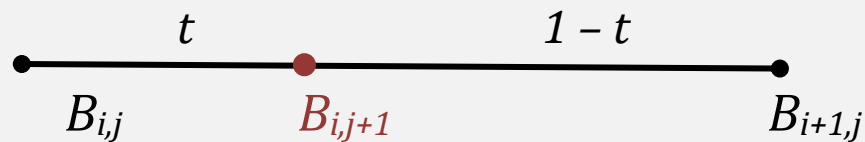
### Алгоритъм на de Casteljau

Вход: масив (полигон) от контролни точки

Изход: точка от кривата на Безие

- ① Разделяме всяка от  $n$  отсечките на полигона в отношение  $t : 1 - t$ , за да получим нова точка.

$$B_{i,j+1} = (1 - t)B_{i,j} + tB_{i+1,j}$$



където  $j$  е номерът на стъпката.

- ② Свързваме новополучените точки и получаваме нов масив (полигон) с дължина  $n - 1$ .
- ③ Върни се ①, докато не се получи само една точка.

Този алгоритъм се извършва за всяко  $t$ , такова че  $0 \leq t \leq 1$ . върнатите точки съставят кривата на Безие. Интервалът  $[0,1]$  е разделен на 200 сегмента, за да се начертае гладко кривата.

Паралелно се извършва и друг алгоритъм, генериращ ходографа.

## Алгоритъм за ходограф

Вход: масив (полигон) от контролни точки

Изход: ходограф

- ① Инициализира се масив за контролните точки на ходографа.
- ② Контролна точка на ходографа се получава като разлика от координатите на две съседни контролни точки на кривата на Безие, т.е. докато  $i \leq n - 1$ , добави в масива:

$$B'_i = n(B_{i+1} - B_i)$$

В тази програма,  $n = 1$ , за по-изгодна визуализация.

- ③ Извършва се *алгоритъма на de Casteljau* с вход масива с контролните точки на ходографа.
- ④ Ходографът на крива на Безие с  $n + 1$  контролни точки е крива на Безие с  $n$  контролни точки.

## II. Работа с програмата

Потребителят работи единствено с левия прозорец и в реално време се генерира ходограф в десния прозорец. Промените на оригиналната крива се отразяват веднага и на ходографа.

- ❖ Добавяне на нова контролна точка: **натискане ляв бутон**
- ❖ Промяна на местоположението на контролна точка: **задържане и влачене на желаната точка с десен бутон**
- ❖ Изчистване на прозореца: **C**

