Науково-практичний звіт на тему

АПРОКСИМАЦІЯ ОПУКЛОЇ ОБОЛОНКИ КРИВОЮ БЕЗЬЄ

С. О. Ярема, студент 3 курсу, групи ІПС-31

Анотація. У роботі запропоновано методи та опис підходу до розв'язання задачі побудови апроксимації опуклої оболонки кубічним сплайном Безьє. Також розглянутий хід роботи з описом того чому була вибрана саме така методологія.

Abstract. The paper offers methods and description of an approach to solving the problem of constructing an approximation of a convex hull by a cubic Bezier spline. The course of work is also reviewed, describing why this particular methodology was chosen.

1 Вступ

Постановка проблеми. В роботі розглядається побудова апроксимації опуклої оболонки сплайном Безьє у 2D. На практиці розв'язки цієї задачі можуть використовуватися в багатьох функціях графічних додатків, зокрема, метод побудови гладкого сплайну Безьє для точок многокутника буде корисним для створення нескінченно збільшуваних векторних зображень. Також коли гладкий сплайн побудований для точок опуклої оболонки це надає можливості для диференціального аналізу побудованої апроксимації.

Криві Безьє широко використовуються в анімації у якості шляхів (траєкторій) руху об'єктів, таким чином поєднавши опуклу оболонку з побудованою гладкою апроксимацією сплайном Безьє можна створювати, наприклад, траєкторії руху патрулів навколо набору об'єктів.

Аналіз останніх досліджень Для побудови опуклої оболонки у двовимірному варіанті без додаткових даних існують достатньо відомі алгоритми Джарвіса, Грехема, Чана, швидкої оболонки. Серед них мій вибір пав на алгоритм Грехема, оскільки він дозволяє побудувати опуклу оболонку з оцінкою $O(n \log n)$ по часу для найгіршого випадку при цьому потребує $M(n)$ пам'яті для збереження відсортованого за полярним кутом списку точок, де n це кількість точок які аналізуються [1][2]. Побудова сплайну Безьє складається з побудови набору кривих Безьє, що були запроваджені в 1962 році П'єром Безьє з автомобілебудівної компанії «Рено», хоча ще в 1959 році використовувались Полем де Кастельє з компанії «Сітроен». Для того щоб можна було контролювати близкість сплайну до справжньої опуклої оболонки я використав кубічні криві Безьє для утворення сплайну. Побудова сплайну виконується лінійним обходом списку вершин опуклої оболонки.

Новизна та ідея. В розглядуваній роботі запропоновано підхід котрий дозволяє будувати сплайн Безьє з регульованою близькістю до опуклого многокутника.

Мета статті. Розробити метод побудови гладкої апроксимації опуклої оболонки набору точок

2 Основна частина.

Задача складається з двох частин: побудови опуклої оболонки та побудови сплайну Безьє.

Постановка задачі побудови опуклої оболонки для набору точок. Нехай заданий список S із N точок на площині. Необхідно виділити упорядкований двозв'язний циклічний список точок $H \subseteq S$ такий, що H буде задавати опуклий многокутник, причому усі точки $p \in S$ будуть знаходитися всередині многокутника заданого H.

2.1. Побудова опуклої оболонки методом Грехема.

Як вже було сказано раніше, даний алгоритм дає найкращу оцінку для найгіршого випадку серед своїх аналогів. Для інтуїтивного сприйняття, основна ідея алгоритму в тому, що точка з мінімальною х координатою буде належати до опуклої оболонки, якщо ж вона не належала б то не могла б знаходитися всередині многокутника опуклої оболонки. Далі ця точка фіксується як центр полярної системи координат і відбувається обхід інших точок проти годинникової стрілки (полярним кутом).

На кроці ми аналізуємо нову точку разом з попередніми двома, якщо ці три точки у порядку за полярним кутом утворюють лівий поворот то ми додаємо нову точку в опуклу оболонку, якщо ж правий поворот то видаляємо з теперішньої опуклої оболонки останню точку і повторюємо видалення доти, доки не виконається умова лівого повороту.

Для виробничих систем дане рішення може бути корисним для обробки деталей першого ступеня щоб відрізати непотрібні частини заготовки.

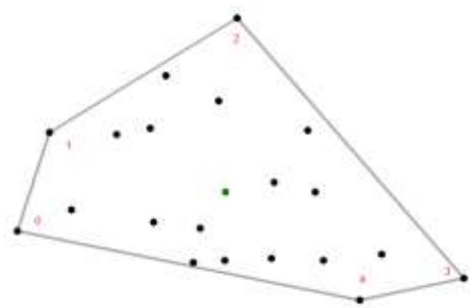

Рис 1. приклад побудованої опуклої оболонки в програмній реалізації

Введемо деякі поняття потрібні для опису алгоритму.

Лівий та правий повороти.

Якщо внутрішній кут $p_1p_2p_3 \geq 0$ то $p_1p_2p_3$ утворюють «правий поворот», інакше $p_1p_2p_3$ утворюють «лівий поворот»

Внутрішній кут визначаємо як:

$$\Delta = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}$$

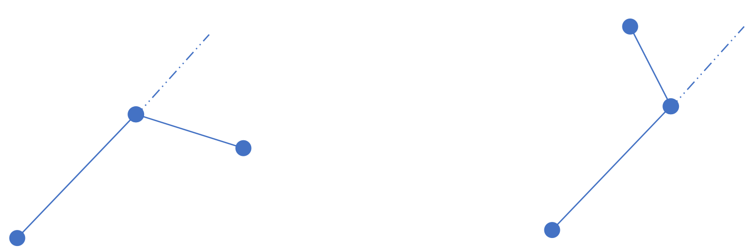

Рис. 2 Правий та лівий повороти

Тобто якщо виконати обхід точок опуклого многокутника за полярним кутом то будуть спостерігатися лише ліві повороти.

2.2. Побудова сплайну Безьє, що апроксимує опуклу оболонку

Оскільки апроксимується саме опукла оболонка була поставлена умова на те щоб сплайн знаходився ззовні опуклого многокутника торкаючись його тільки у вершинах. Також сплайн має бути гладким.

Сплайн буде складатися з набору кривих Безьє, будуватимемо їх для кожної пари суміжних точок. Криву Безьє порядку n задає наступне рівняння:

$$P(t) = \sum_{i=0}^{n} \binom{n}{i} (1-t)^{n-i} t^i P_i \qquad [3, chapter 2]$$

Було вирішено використовувати кубічні криві Безьє, оскільки вони надають більше контролю над формою кривої ніж квадратичні. Також, контрольні точки кубічних кривих допомагають інтуїтивно розуміти як себе буде поводити крива, а саме контрольна точка разом з найближчою початковою точкою утворюють дотичній вектор до кривої у найближчій початковій точці:

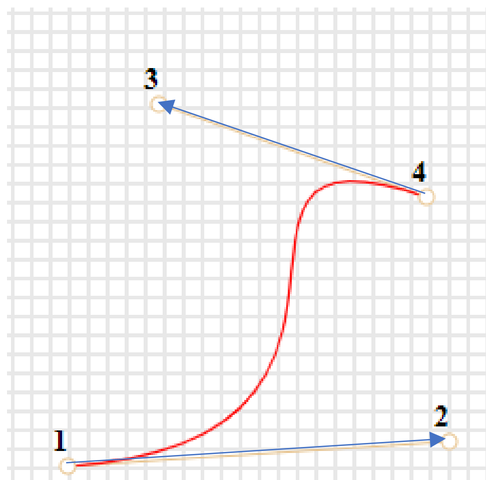

Рис. 3 контрольні точки

За допомогою цієї властивості можна забезпечити гладкість сплайну зробивши напрямні вектори для сусідніх кривих протилежними.

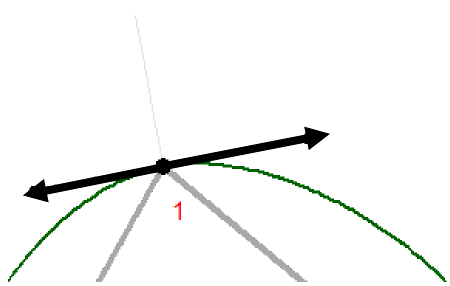

Рис. 4 колінеарні дотичні вектори

За допомогою довжини дотичних векторів можна маніпулювати близкістю кривої до ребра на якому вона побудована.

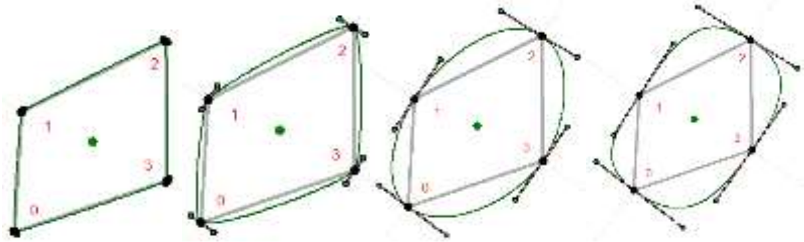

Рис. 5 Зміна форми кривої від довжини дотичного вектора (1/50, 1/10, 1/3, 1/2 довжини ребра на котрому будується крива)

Дільник дробу котрий визначає довжину дотичного вектора назвемо фактором близькості. Фактор близькості має бути більшим або рівним 2 інакше буде можливим виникнення «петлі» на ділянці кривої.

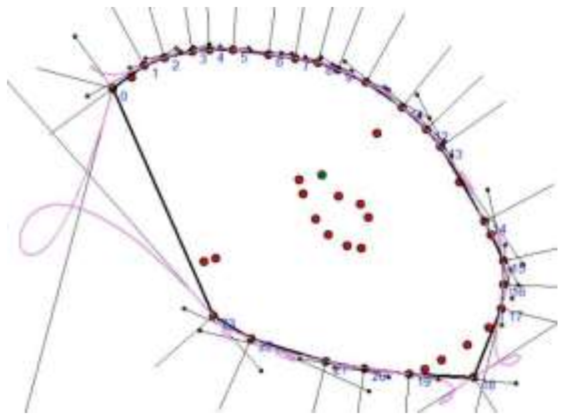

Рис. 6 Утворення багатьох петель при занадто малих факторах близькості

Для того щоб визначати як саме вибирати направляючий вектор дотичної було вирішено використовувати наступні кроки:

1) Для вершини у якій визначаємо контрольну точку будуємо середній вектор нормованих векторів суміжних до неї ребер
2) Будуємо перпендикуляр до отриманого вектора
3) На перпендикулярі відкладаємо вектор потрібної довжини, заданої близькістю та довжиною суміжного ребра
4) Фіксуємо кінець вектора як нову керуючу точку

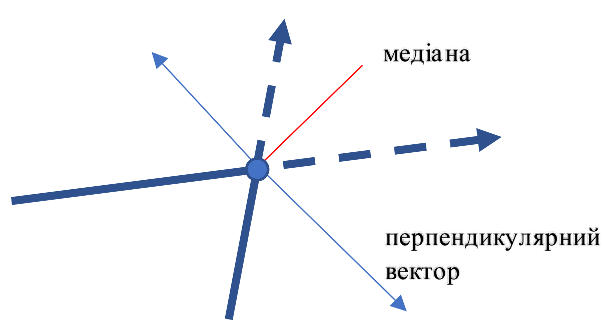

Рис. 7 побудова дотичних векторів

3 Обґрунтування складності

Теорема

Опукла оболонка з n точок на площині може бути знайдена методом Грехема за час $O(n \log n)$ при використанні пам'яті $M(n)$ з використанням лише арифметичних операцій і порівнянь.

Із попереднього обговорення алгоритму зрозуміло, що в ньому використовуються лише арифметичні операції та порівняння. Знаходження крайньої точки та ітерація через відсортовані за полярним кутом точки потребують лінійного часу, тоді як крок сортування, який є визначальним за часом роботи, виконується за час $O(n \log n)$. Для представлення зв'язного списку точок достатньо $M(n)$ пам'яті. [2, ст. 129]

Для побудови сплайну Безьє треба обробити n-1 ребро побудувавши на них відповідно криві Безьє. Для того щоб намалювати криву Безьє з певним кроком малювання d необхідно $\left\lfloor\frac{1}{d}\right\rfloor$ кроків, оскільки ми змінюємо параметр t рівняння кривої у межах [0,1]. Отже для побудови сплайну складність складатиме $O\left(\left\lfloor\frac{1}{d}\right\rfloor h\right) = O(h)$, де h це кількість точок, що увійшли в опуклу оболонку, для малих значень кількості точок константа матиме значний вплив проте асимптотична складність буде лінійною.

Отже загальна складність алгоритму складатиме $O(n \log n)$.

4 Практична частина

Особливістю реалізації є можливість динамічно змінювати параметр близькості сплайну до опуклої оболонки. Реалізація написана на мові Python 3.6, з використанням Tkinter для візуалізації та інтерфейсу, та PIL для створення зображень візуалізації при великій кількості точок.

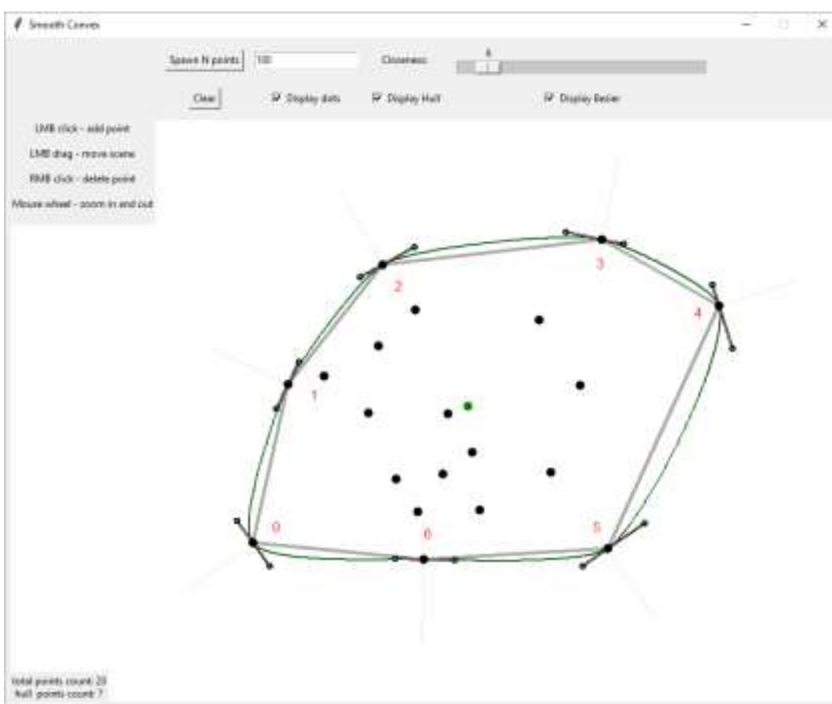

Рис. 8 інтерфейс програмної реалізації

5 Висновки

У роботі запропоновано метод побудови апроксимації опуклої оболонки сплайном Безьє з можливістю контролю близькості кривих до ребер опуклої оболонки. В процесі розробки підходу було протестовано різні варіації того яким саме чином краще створювати контрольні точки кривої Безьє для зручності та кращого візуального вигляду. Було вирішено використовувати алгоритм Грехема для побудови опуклої оболонки, оскільки він дає найкращу оцінку складності в найгіршому випадку для двовимірного випадку $O(n \log n)$. Для побудови сплайну використані кубічні криві Безьє щоб мати більше контролю над напрямом дотичних у стартових та кінечних точках, це зробило можливим маніпуляцію близкістю кривої до ребра опуклої оболонки. Складність побудови сплайну є лінійною. Спосіб утворення контрольних точок описаний у роботі може бути модифікований на більшу розмірність з заміною медіани на середню площину, а також зі зміною алгоритму для створення опуклої оболонки, наприклад, на трьох вимірну варіацію алгоритму Quick Hull.

Для покращення ефективності програми при динамічному додаванні нових точок можна скористатися алгоритмами динамічної підтримки опуклої оболонки. Для великої кількості точок була використана додаткова бібліотека, щоб пришвидшити побудову візуалізації, оскільки саме візуалізація виявилася найбільш часозатратною частиною програми, це спонукає до думки про можливу паралелезацію малювання у розподілений буфер зображення.

Список літератури

1. *Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн.* Алгоритмы. Построение и анализ Introduction to Algorithms. — 2-е изд. — "Вильямс", 2005.

2. *Ф.Препарата, М. Шеймос. Вычислительная геометрия: введение. – М.: Мир, 1989.*

3. *Thomas Sederberg.* Bézier curves