

Аннотация

Рассматривается задача построения гладкой кривой на плоскости по заданным узлам интерполяции. Кривая строится в виде сплайна, составленного из кубических кривых Безье. Показано, что, если потребовать непрерывности первых и вторых производных в промежуточных узлах интерполяции, то при фиксированной параметризации кривых Безье такой сплайн определен однозначно, при этом контрольные точки кривых вычисляются с помощью решения системы линейных уравнений с четырехдиагональной матрицей. Рассматриваются различные способы параметризации кривых Безье, существенно влияющие на форму сплайна. Наилучший сплайн вычисляется как решение оптимизационной задачи: при фиксированном суммарном времени прохождения сплайна минимизировать интеграл от квадрата его второй производной.

Ключевые слова: кривая Безье, сплайн-интерполяция, ленточная матрица, задача оптимизации.

Построение оптимального сплайна Безье

В. В. Борисенко

В компьютерной графике и других разделах прикладной математики часто возникает задача построения гладкой кривой, проходящей через заданные точки — узлы интерполяции. Наиболее популярное решение — использование сплайна, составленного из кривых Безье. Можно рассматривать квадратичные кривые, но наилучшие результаты дает использование кубических кривых Безье.

Кубическая кривая Безье строится по четырем точкам $P_0, P_1, P_2, P_3 \in \mathbf{R}^2$. В классическом определении это отображение \mathcal{B} отрезка $[0, 1]$ в плоскость \mathbf{R}^2 , задаваемое формулой

$$\mathcal{B}(t) = (1-t)^3 P_0 + 3(1-t)^2 t P_1 + 3(1-t) t^2 P_2 + t^3 P_3, \quad 0 \leq t \leq 1. \quad (1)$$

(Кривую Безье порядка n можно также определить рекурсивно через кривые Безье порядка $n-1$.) Точки P_0 и P_3 являются концами кубической кривой Безье, точки P_1 и P_2 называют контрольными. Кривая Безье не проходит через контрольные точки, но они влияют на ее форму. В частности, прямые $P_0 P_1$ и $P_2 P_3$ являются касательными к кривой в ее концах. Целиком кривая Безье содержится внутри выпуклой оболочки точек P_0, P_1, P_2, P_3 , см. рис. 1.

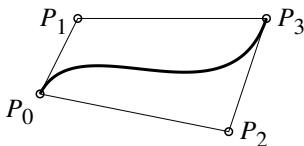


Рис. 1: Кубическая кривая Безье

В дальнейшем мы будем обозначать узлы интерполяции через Q_0, Q_1, \dots, Q_n ; кривую Безье, соединяющую узлы Q_i и Q_{i+1} , обозначим через \mathcal{B}_i , а ее контрольные точки — через $A_i, B_i, i = 0, 1, \dots, n-1$. В частности, на рис. 1 точки P_0, P_3 соответствуют узлам Q_0, Q_1 , а точки P_1, P_2 — контрольным точкам A_0, B_0 .

В большинстве графических редакторов гладкая кривая строится как сплайн, составленный из кубических кривых Безье. При этом пользователю приходится задавать не только узлы интерполяции (т.е. точки, через которые проходит кривая), но и контрольные точки, влияющие на форму

кривой. Сплайн строится так, чтобы первые производные в точках склейки (узлах интерполяции) были непрерывны. Отсюда вытекает, что для двух соседних кривых Безье B_{i-1} и B_i , которые склеиваются в узле Q_i , контрольные точки B_{i-1} и A_i должны быть симметричны относительно точки Q_i . При добавлении нового сегмента сплайна, соединяющего узлы Q_i и Q_{i+1} , пользователь определяет новый узел Q_{i+1} , нажимая на левую клавишу мыши и *не отпуская ее*. Затем при нажатой клавише мыши пользователь перемещает курсор, определяя сразу две симметричные контрольные точки B_i и A_{i+1} , при этом кривая изгибается. В тот момент, когда пользователь находит нужную ему форму кривой, он отпускает клавишу мыши. Таким образом, задается новый узел и две контрольные точки, см. рис 2.

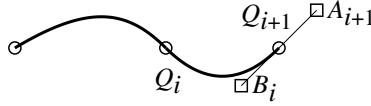


Рис. 2: Построение сплайна в популярных графических редакторах

Такой способ построения сплайна не очень удобен. Неподготовленному пользователю, который не знает, что такое кубическая кривая Безье и ее контрольные точки, поначалу даже трудно понять, что от него требуется. Для него гораздо проще было бы просто отметить узлы интерполяции без всяких контрольных точек, предоставив программе самой построить гладкий сплайн (вычислив контрольные точки незаметно для пользователя). Но даже если пользователь выучит теорию, то все равно построенный сплайн будет не самого высокого качества, поскольку используемый метод обеспечивает непрерывность только первой производной сплайна в точках склейки его сегментов. Вторая производная может иметь разрывы в этих точках. Геометрически это соответствует тому, что функция кривизны кривой имеет разрывы в узлах интерполяции.

Однако имеется простой способ построения сплайна Безье только по узлам интерполяции без указания контрольных точек. При этом построенный сплайн будет иметь непрерывные первую и вторую производные. Зрительно построенная кривая имеет более высокое качество, чем при использовании традиционного метода. Идея метода состоит в том, что контрольные точки вычисляются с помощью решения системы линейных уравнений.

Вычислим первую и вторую производные кривой Безье в ее конечных точках. Дважды дифференцируя по t равенство (1), получаем:

$$\begin{aligned}\dot{B}(t) &= 3[-(1-t)^2 P_0 + (1-3t)(1-t)P_1 + t(2-3t)P_2 + t^2 P_3] \\ \ddot{B}(t) &= 6[(1-t)P_0 - (2-3t)P_1 + (1-3t)P_2 + tP_3]\end{aligned}$$

Подставив значения $t = 0$ и $t = 1$, получим выражения для первой и второй

производных в концах кривой Безье:

$$\dot{B}(0) = 3(P_1 - P_0), \quad \dot{B}(1) = 3(P_3 - P_2), \quad (2)$$

$$\ddot{B}(0) = 6(P_0 - 2P_1 + P_2), \quad \ddot{B}(1) = 6(P_1 - 2P_2 + P_3). \quad (3)$$

Рассмотрим теперь кривую B_i , представляющую собой i -й сегмент сплайна, $0 \leq i < n$. Напомним, что мы обозначили узлы интерполяции буквами Q_i , $0 \leq i \leq n$, а контрольные точки кривых Безье — буквами A_i , B_i , $0 \leq i < n$. Заменяя в равенствах (2) и (3) переменные P_0 на Q_i , P_3 на Q_{i+1} , P_1 на A_i , P_2 на B_i , получим:

$$\dot{B}_i(0) = 3(A_i - Q_i), \quad \dot{B}_i(1) = 3(Q_{i+1} - B_i), \quad (4)$$

$$\ddot{B}_i(0) = 6(Q_i - 2A_i + B_i), \quad \ddot{B}_i(1) = 6(A_i - 2B_i + Q_{i+1}). \quad (5)$$

Будем строить сплайн так, чтобы в каждом промежуточном узле интерполяции Q_i , $i = 1, \dots, n-1$, две кривые B_{i-1} и B_i , которые склеиваются в этом узле, имели одинаковые первые и вторые производные. Получаем два уравнения:

$$\dot{B}_{i-1}(1) = \dot{B}_i(0),$$

$$\ddot{B}_{i-1}(1) = \ddot{B}_i(0).$$

Используя формулы (4) и (5), запишем их в виде

$$Q_i - B_{i-1} = A_i - Q_i,$$

$$A_{i-1} - 2B_{i-1} + Q_i = Q_i - 2A_i + B_i.$$

Перенеся члены с A и B в левую часть, а Q в правую, получим

$$B_{i-1} + A_i = 2Q_i, \quad (6)$$

$$A_{i-1} - 2B_{i-1} + 2A_i - B_i = 0. \quad (7)$$

Равенства (6) и (7) можно рассматривать как уравнения с неизвестными A_j , B_j (контрольные точки кривых), при этом точки Q_j (узлы интерполяции) нам даны. Получаем $2n - 2$ линейных уравнения с $2n$ неизвестными. Для того, чтобы система имела единственное решение, необходимо задать еще 2 уравнения, чтобы число уравнений равнялось числу неизвестных. Два дополнительных уравнения формулируются как краевые условия в начальном и конечном узлах Q_0 и Q_n . Потребуем, чтобы вторая производная в начальной и конечной точках равнялась нулю. Получаем уравнения

$$\ddot{B}_0(0) = 0,$$

$$\ddot{B}_{n-1}(1) = 0,$$

которые через точки записываются как

$$2A_0 - B_0 = Q_0, \quad (8)$$

$$2B_{n-1} - A_{n-1} = Q_n. \quad (9)$$

Итак, мы получили систему из $2n$ линейных уравнений с $2n$ неизвестными. Матрица системы имеет размер $2n \times 2n$, ее элементами являются обычные вещественные числа. Немного непривычно то, что неизвестными A_i, B_i здесь являются не числа, а точки плоскости \mathbf{R}^2 (которые мы отождествляем с векторами \mathbf{R}^2), задаваемые двумя вещественными координатами; столбец свободных членов также состоит не из чисел, а из точек \mathbf{R}^2 . Тем не менее обычный метод Гаусса решения линейной системы здесь работает точно так же, как и в случае числовых неизвестных. Отметим, что матрица системы четырехдиагональная, если упорядочить уравнения следующим образом: сначала записываем краевое условие (8) в начальном узле Q_0 , затем для $i = 1, \dots, n-1$ — уравнения (7) и (6), выражающие равенство вторых и первых производных в узлах Q_i ; последним записываем краевое условие (9) в узле конечном узле Q_n . Система с 4-диагональной матрицей решается за время $O(n)$ методом Гаусса с выбором максимального по модулю разрешающего элемента в столбце (используется вариант метода Гаусса для m -диагональной матрицы, в нашем случае $m = 4$.) Решив систему, мы получим искомые контрольные точки A_i, B_i , которые вместе с узлами интерполяции определяют искомый сплайн. А именно, i -я кривая Безье, соединяющая узлы интерполяции Q_i и Q_{i+1} , задается четырьмя точками Q_i, A_i, B_i, Q_{i+1} , $i = 0, 1, \dots, n-1$.

Пример 1. Пусть требуется построить Безье-сплайн по четырем узлам

$$Q_0 = (0, 0), \quad Q_1 = (2, 2), \quad Q_2 = (3, 1), \quad Q_3 = (4, 1).$$

Составим систему уравнений с неизвестными $A_0, B_0, A_1, B_1, A_2, B_2$. Первое и последнее уравнения представляют собой условия равенства нулю второй производной в краевых узлах Q_0 и Q_3 , уравнения со 2-го по 5-е выражают равенство вторых и первых производных в узлах Q_1 и Q_2 . Получим следующую расширенную матрицу системы:

$$\begin{array}{cccccc|l} 2 & -1 & 0 & 0 & 0 & 0 & (0, 0) \\ 1 & -2 & 2 & -1 & 0 & 0 & (0, 0) \\ 0 & 1 & 1 & 0 & 0 & 0 & (4, 4) \\ 0 & 0 & 1 & -2 & 2 & -1 & (0, 0) \\ 0 & 0 & 0 & 1 & 1 & 0 & (6, 2) \\ 0 & 0 & 0 & 0 & -1 & 2 & (4, 1) \end{array}$$

Решив систему, получим контрольные точки:

$$\begin{aligned} A_0 &= (0.756, 0.956), & B_0 &= (1.511, 1.911), \\ A_1 &= (2.489, 2.089), & B_1 &= (2.711, 1.311), \\ A_2 &= (3.289, 0.689), & B_2 &= (3.644, 0.844). \end{aligned}$$

Искомый сплайн изображен на рис. 3. Кружками отмечены узлы интерполяции, крестиками — контрольные точки.

Описанный метод вычисления сплайна хорошо работает, когда расстояния между узлами интерполяции примерно одинаковы. Однако он требует

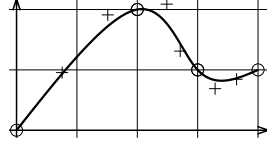


Рис. 3: Сплайн, построенный по узлам $(0, 0)$, $(2, 2)$, $(3, 1)$, $(4, 1)$

модификации, когда расстояния сильно различаются. Дело в том, что параметризация каждой кривой Безье, являющейся сегментом сплайна, выполняется значениями времени t из отрезка $[0, 1]$. Можно рассмотреть следующую аналогию: пусть автомобиль проезжает несколько городов (которые соответствуют узлам интерполяции), мы строим оптимальную траекторию его движения с учетом скорости. Описанный выше метод соответствует тому, что дистанцию между соседними городами автомобиль всегда проезжает ровно за один час независимо от расстояния между ними. Между удаленными друг от друга городами автомобиль едет с большой скоростью, затем, если следующий город расположен близко к предыдущему, он резко тормозит и дальше едет очень медленно. Соответственно траектория движения в таком случае состоит из почти спрямленных линий между удаленными городами и резких поворотов для близких городов (см. рис. 5, кривая в левой части рисунка). Хотелось бы построить траекторию с более плавными поворотами.

Естественное решение — рассматривать параметризацию кривых Безье, составляющих сплайн, не значениями t от нуля до единицы, а значениями из отрезка $[0, l]$, где l — расстояние между соседними узлами интерполяции.

Обозначим $\alpha = 1/l$ и рассмотрим параметризацию кривой Безье значениями из отрезка $[0, l]$:

$$\begin{aligned} \mathcal{A} : [0, l] &\rightarrow \mathbf{R}^2, \\ \mathcal{A}(t) &= \mathcal{B}(\alpha t), \end{aligned}$$

где отображение \mathcal{B} задается четырьмя точками $P_0, P_1, P_2, P_3 \in \mathbf{R}^2$ по формуле (1). Сделав замену $t \rightarrow \alpha t$ в формуле (1) и дважды продифференцировав по t , вычислим первую и вторую производные функции $\mathcal{A}(t)$:

$$\begin{aligned} \dot{\mathcal{A}}(t) &= 3\alpha[-(1 - \alpha t)^2 P_0 + (1 - 3\alpha t)(1 - \alpha t)P_1 + \alpha t(2 - 3\alpha t)P_2 + \alpha^2 t^2 P_3] \\ \ddot{\mathcal{A}}(t) &= 6\alpha^2[(1 - \alpha t)P_0 - (2 - 3\alpha t)P_1 + (1 - 3\alpha t)P_2 + \alpha t P_3] \end{aligned} \quad (10)$$

Подставив значения $t = 0$ и $t = l$, получим выражения для первой и второй производных в концах кривой $\mathcal{A}(t)$:

$$\dot{\mathcal{A}}(0) = 3\alpha(P_1 - P_0), \quad \dot{\mathcal{A}}(l) = 3\alpha(P_3 - P_2), \quad (11)$$

$$\ddot{\mathcal{A}}(0) = 6\alpha^2(P_0 - 2P_1 + P_2), \quad \ddot{\mathcal{A}}(l) = 6\alpha^2(P_1 - 2P_2 + P_3). \quad (12)$$

Пусть теперь \mathcal{A}_i — кривая Безье, соединяющая узлы интерполяции Q_i и Q_{i+1} , параметризованная значениями из отрезка $[0, l_i]$, где $l_i = |Q_i Q_{i+1}|$. Как и ранее, обозначим через A_i и B_i ее контрольные точки. Положим $\alpha_i = 1/l_i$. Кривая \mathcal{A}_i имеет следующие значения первой и второй производных в узлах интерполяции:

$$\dot{\mathcal{A}}_i(0) = 3\alpha_i(A_i - Q_i), \quad \dot{\mathcal{A}}_i(1) = 3\alpha_i(Q_{i+1} - B_i), \quad (13)$$

$$\ddot{\mathcal{A}}_i(0) = 6\alpha_i^2(Q_i - 2A_i + B_i), \quad \ddot{\mathcal{A}}_i(1) = 6\alpha_i^2(A_i - 2B_i + Q_{i+1}). \quad (14)$$

Составим опять уравнения, утверждающие равенство первых и вторых производных соседних кривых Безье в промежуточных узлах интерполяции с учетом другой параметризации. Имеем:

$$\begin{aligned} \dot{\mathcal{A}}_{i-1}(l_{i-1}) &= \dot{\mathcal{A}}_i(0), \\ \ddot{\mathcal{A}}_{i-1}(l_{i-1}) &= \ddot{\mathcal{A}}_i(0). \end{aligned}$$

Применив формулы (13) и (14), получим

$$\begin{aligned} \alpha_{i-1}(Q_i - B_{i-1}) &= \alpha_i(A_i - Q_i), \\ \alpha_{i-1}^2(A_{i-1} - 2B_{i-1} + Q_i) &= \alpha_i^2(Q_i - 2A_i + B_i). \end{aligned}$$

Перенеся члены с A и B в левую часть, а Q в правую, получим уравнения:

$$\begin{aligned} \alpha_{i-1}B_{i-1} + \alpha_iA_i &= (\alpha_{i-1} + \alpha_i)Q_i, \quad (15) \\ \alpha_{i-1}^2A_{i-1} - 2\alpha_{i-1}^2B_{i-1} + 2\alpha_i^2A_i - \alpha_i^2B_i &= \alpha_i^2Q_i - \alpha_{i-1}^2Q_{i-1}. \quad (16) \end{aligned}$$

Здесь индекс i пробегает значения от 1 до $n - 1$. Таким образом, мы получили $2n - 2$ уравнений на контрольные точки A_i , B_i кривых \mathcal{A}_i . Добавим краевые условия — те же самые, как и в предыдущем случае (параметризации кривых без учета расстояний между узлами): вторые производные кривых в крайних узлах интерполяции равны нулю, это уравнения (8) и (9). Чтобы матрица системы имела четырехдиагональную форму, упорядочим уравнения следующим образом: сначала краевое условие (8) в начальном узле Q_0 , затем уравнения (16) и (15) для промежуточных узлов Q_i , $i = 1, \dots, n - 1$, и в конце краевое условие (9) в конечном узле Q_n . Контрольные точки вычисляются как решение полученной системы линейных уравнений.

Пример 2. Построим сплайн Безье по тем же самым узлам, что и в примере 1, но параметризуя кривые Безье с учетом расстояний между узлами интерполяции. Имеем 4 узла:

$$Q_0 = (0, 0), \quad Q_1 = (2, 2), \quad Q_2 = (3, 1), \quad Q_3 = (4, 1).$$

Вычислим расстояния между узлами и коэффициенты α_i :

$$l_0 = |Q_0 Q_1| \approx 2.828, \quad l_1 = |Q_1 Q_2| \approx 1.414, \quad l_2 = |Q_2 Q_3| = 1 \quad (17)$$

$$\alpha_0 = 1/l_0 \approx 0.354, \quad \alpha_1 = 1/l_1 \approx 0.707, \quad \alpha_2 = 1/l_2 = 1 \quad (18)$$

Составим систему уравнений с неизвестными $A_0, B_0, A_1, B_1, A_2, B_2$. Первое и последнее уравнения представляют собой условия равенства нулю второй производной в краевых узлах Q_0 и Q_3 , уравнения со второго по пятое выражают равенство вторых и первых производных соседних кривых A_i в узлах Q_1 и Q_2 . Получим следующую расширенную матрицу системы:

$$\begin{array}{cccccc|c} 2 & -1 & 0 & 0 & 0 & 0 & (0, 0) \\ 0.125 & -0.25 & 1 & -0.5 & 0 & 0 & (0.75, 0.75) \\ 0 & 0.354 & 0.707 & 0 & 0 & 0 & (2.121, 2.121) \\ 0 & 0 & 0.5 & -1 & 2 & -1 & (1.5, 0.5) \\ 0 & 0 & 0 & 0.707 & 1 & 0 & (5.121, 1.707) \\ 0 & 0 & 0 & 0 & -1 & 2 & (4, 1) \end{array}$$

Решив систему, найдем контрольные точки:

$$\begin{aligned} A_0 &= (0.695, 1.202), & B_0 &= (1.390, 2.405), \\ A_1 &= (2.305, 1.798), & B_1 &= (2.589, 1.194), \\ A_2 &= (3.291, 0.863), & B_2 &= (3.645, 0.932). \end{aligned}$$

Искомый сплайн изображен на рис. 4 справа. Для сравнения слева показан сплайн, вычисленный первым методом без учета расстояний между узлами.

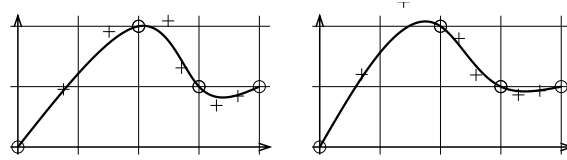


Рис. 4: Сплайны, построенный по узлам $(0, 0)$, $(2, 2)$, $(3, 1)$, $(4, 1)$. Сплайн слева использует традиционную параметризацию кривых Безье $t \in [0, 1]$. Сплайн справа использует параметризацию, учитывающую расстояния между узлами

Два сплайна на рисунке 4 отличаются незначительно, поскольку в данном примере расстояния между узлами интерполяции различаются не очень сильно. На рис. 5 приведен другой пример двух сплайнов, построенных по одним и тем же узлам. Расстояния между узлами в этом примере распределены неравномерно, поэтому разница в форме двух кривых более заметна.

Левый сплайн на рис. 5, вычисленный первым методом без учета расстояний между узлами интерполяции, состоит из более спрямленных отрезков кривых. Вблизи узлов интерполяции кривая совершает довольно резкие повороты. Сплайн, учитывающий расстояния между узлами, на вид более гладкий, длинные отрезки кривых здесь искривлены сильнее и за счет этого повороты вблизи узлов интерполяции получаются более плавными. Разница между сплайнами также наглядно иллюстрируется рисунком 6. Здесь слева изображены узлы интерполяции, соединенные ломаной, правее нарисованы

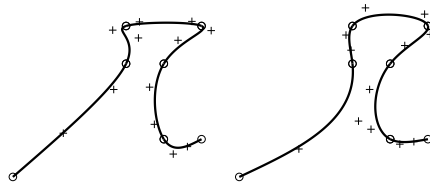


Рис. 5: Сплайны, построенных по одним и тем же узлам двумя разными методами. Сплайн справа использует параметризацию, учитывающую расстояния между узлами

3 сплайна, построенные разными методами по этим узлам. Левый сплайн использует стандартную параметризацию кривых Безье отрезками $t \in [0, 1]$. Сплайн в центре использует рассмотренную параметризацию со временем прохождения сегментов, равным расстоянию между соседними узлами. Наконец, справа изображен сплайн с оптимальной параметризацией, которая рассматривается ниже.

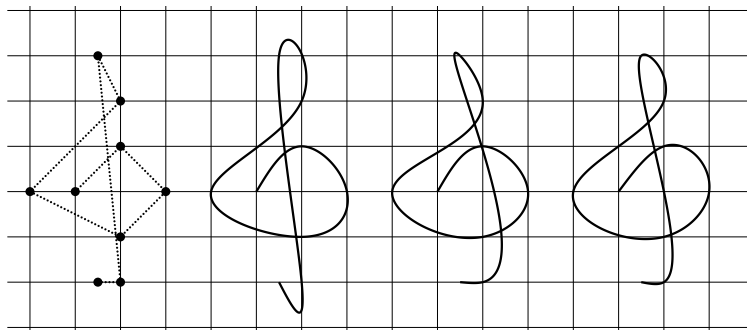


Рис. 6: Три сплайна, построенных по одним и тем же узлам интерполяции. Сплайн слева использует стандартную параметризацию $t \in [0, 1]$, сплайн в центре используют параметризацию расстояниями между соседними узлами. Справа показан *оптимальный сплайн*, минимизирующий интеграл от квадрата второй производной

Вычисление оптимального сплайна

Как видно на рисунках 5 и 6, форма сплайна при заданных узлах сильно зависит от параметризации кривых Безье. Выше мы наряду со стандартной параметризацией рассмотрели параметризацию, при которой кривая между соседними узлами проходит за время, пропорциональное расстоянию между узлами. Априори такое решение представляется естественным и визуально оно дает хорошие результаты, но все же остается открытым

вопрос о выборе наилучшего решения по каким-либо четким критериям. Мы сведем поиск наилучшего сплайна к оптимизационной задаче. Будем искать гладкий сплайн Безье с непрерывными первыми и вторыми производными, который минимизирует интеграл от квадрата второй производной при фиксированном суммарном времени прохождения сплайна. Минимум ищется по всевозможным параметризациям кривых Безье, являющихся сегментами сплайна.

Запишем формально условие задачи. Даны узлы интерполяции $Q_i \in \mathbf{R}^2$, $i = 0, 1, \dots, n$, мы строим сплайн из кривых Безье с непрерывными первыми и вторыми производными в узлах. Кривая Безье A_i , соединяющая узлы Q_i и Q_{i+1} , параметризуется значениями времени из отрезка $[0, t_i]$:

$$\begin{aligned} A_i : [0, t_i] &\rightarrow \mathbf{R}^2, \quad t_i > 0, \quad i = 0, 1, \dots, n-1, \\ A_i(t) &= B_i(\alpha_i t), \quad \alpha_i = 1/t_i. \end{aligned}$$

Кривая Безье B_i задается узлами интерполяции Q_i , Q_{i+1} и контрольными точками A_i , B_i .

Суммарное время прохождения всего сплайна фиксировано, мы полагаем его равным единице:

$$t_0 + t_1 + \dots + t_{n-1} = 1. \quad (19)$$

Сплайн должен иметь непрерывные первые и вторые производные в узлах:

$$\dot{A}_i(t_i) = \dot{A}_{i+1}(0), \quad \ddot{A}_i(t_i) = \ddot{A}_{i+1}(0), \quad i = 0, 1, \dots, n-1.$$

Мы минимизируем интеграл от квадрата второй производной по всему временному отрезку прохождения сплайна:

$$\sum_{i=0}^{n-1} \int_0^{t_i} |\ddot{A}_i(\tau)|^2 d\tau \rightarrow \min. \quad (20)$$

Минимизация выполняется по всем наборам переменных $(t_0, t_1, \dots, t_{n-1})$, удовлетворяющим условию (19); также t_i должны быть положительны. Этот набор переменных задает точку n -мерной гиперплоскости, определяемой уравнением (19).

Интеграл от квадрата второй производной по каждому сегменту сплайна легко вычисляется аналитически. Воспользовавшись формулой (10) для второй производной функции $A_i(t)$, вычислим сначала квадрат второй производной $|\ddot{A}_i(t)|^2$ и затем его интеграл \mathcal{I}_i по отрезку $[0, t_i]$, где $t_i = 1/\alpha_i$. Используем следующие обозначения для координат узлов и контрольных точек:

$$Q_i = (Q_{i1}, Q_{i2}), \quad A_i = (A_{i1}, A_{i2}), \quad B_i = (B_{i1}, B_{i2}), \quad Q_{i+1} = (Q_{i+1,1}, Q_{i+1,2}).$$

Имеем:

$$\begin{aligned}
|\ddot{\mathcal{A}}_i(t)|^2 &= \ddot{\mathcal{A}}_i(t) \cdot \ddot{\mathcal{A}}_i(t) = 36\alpha_i^4 \{ \\
&\quad [t\alpha_i(A_{i1} - 2B_{i1} + Q_{i+1,1}) + (1 - \alpha_i t)(Q_{i1} - 2A_{i1} + B_{i1})]^2 + \\
&\quad [t\alpha_i(A_{i2} - 2B_{i2} + Q_{i+1,2}) + (1 - \alpha_i t)(Q_{i2} - 2A_{i2} + B_{i2})]^2 \}, \\
\mathcal{I}_i &= \int_0^{t_i} |\ddot{\mathcal{A}}_i(\tau)|^2 d\tau = 12\alpha_i^3 [Q_{i1}^2 + Q_{i2}^2 + \\
&\quad 3(A_{i1}^2 + A_{i2}^2 + B_{i1}^2 + B_{i2}^2) + Q_{i+1,1}^2 + Q_{i+1,2}^2 + \\
&\quad Q_{i1}Q_{i+1,1} + Q_{i2}Q_{i+1,2} - \\
&\quad 3(Q_{i1}A_{i1} + Q_{i2}A_{i2} + A_{i1}B_{i1} + A_{i2}B_{i2} + B_{i1}Q_{i+1,1} + B_{i2}Q_{i+1,2})].
\end{aligned} \tag{21}$$

Мы получили классическую задачу на нахождение условного экстремума функции, которая может быть решена аналитически методом множителей Лагранжа. Выражения, однако, получаются очень громоздкими даже в простейшем случае $n = 2$, который соответствует трем узлам. Если преобразовать систему к одному уравнению с переменной α_0 , то получается уравнение, в котором линейная комбинация дробей приравняется к нулю, причем в числителях и знаменателях переменная α_0 встречается в степенях до 7 включительно. Это уравнение можно решить аналитически, воспользовавшись, к примеру, системой компьютерной алгебры Sage. В ответе получается выражение длиной в 21741 символ! Правда, когда мы подставляем в него конкретные числовые значения, то получается уже вполне компактный ответ. Рассмотрим, например, 3 узла, которые образуют египетский треугольник:

$$Q_1 = (0, 0), \quad Q_2 = (4, 3), \quad Q_3 = (4, 0).$$

В этом случае получаем следующий точный ответ:

$$\alpha_0 = S + \frac{84}{625 \cdot S} + \frac{22}{25}, \quad \text{где } S = \left(\frac{24\sqrt{15}}{625} + \frac{2448}{15625} \right)^{1/3}.$$

Вычислим приближенно все параметры, по которым строится сплайн:

$$\begin{aligned}
\alpha_0 &= 1.752999, \quad \alpha_1 = \alpha_0 / (1 - \alpha_0) = 2.328022, \\
t_0 &= 1/\alpha_0 = 0.570451, \quad t_1 = 1 - t_0 = 0.429549, \\
A_0 &= (1.713633, 1.664011), \quad B_0 = (3.427268, 3.328022), \\
A_1 &= (4.431267, 2.752999), \quad B_1 = (4.215634, 1.376500).
\end{aligned}$$

Полученный сплайн изображен на рис. 7.

При $n \geq 3$ (четыре и более узла интерполяции) получить точное аналитическое решение не удастся. Однако можно применить численные методы решения оптимизационной задачи — нахождения условного экстремума функции от многих переменных. Мы минимизируем функцию от набора переменных

$$T = (t_0, t_1, \dots, t_{n-1}) \in \mathbf{R}^n$$

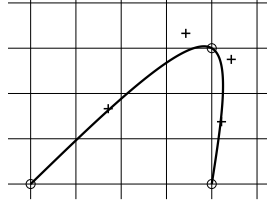


Рис. 7: Оптимальный сплайн, построенный по вершинам египетского треугольника

при условии связи (19):

$$t_0 + t_1 + \dots + t_{i-1} = 1.$$

Точка T , таким образом, принадлежит гиперплоскости, определяемой уравнением (19). Дополнительно мы накладываем также условие положительности всех координат T : $t_i > 0$, $i = 0, \dots, n-1$.

Для заданного набора T и фиксированных узлов интерполяции Q_i , $i = 0, \dots, n$, вычисляется сплайн с данными временами прохождения сегментов между узлами путем решения системы линейных уравнений (15), (16) с учетом краевых условий (8) и (9). (Напомним, что переменные α_i задаются как $\alpha_i = 1/t_i$.) Для найденного сплайна вычисляется интеграл от квадрата второй производной по формулам (20) и (21). Таким образом определяется целевая функция, которую мы минимизируем.

Как показывают эксперименты, в нашем случае прекрасно работает самый простой метод минимизации “Hill Climbing” (Карабканье на холм) с автоматическим выбором шага (при нахождении минимума более уместным было бы название “Сползание в яму”). В качестве начальной точки используется параметризация сплайна, при которой времена прохождения его сегментов пропорциональны расстояниям между соседними узлами:

$$t_i = |Q_i Q_{i+1}| / \sum_{k=0}^{n-1} |Q_k Q_{k+1}|, \quad i = 0, 1, \dots, n-1.$$

Для текущей точки T в методе Hill Climbing рассматриваются n соседних точек, которые в нашем случае также должны быть расположены на гиперплоскости (19). Мы задаем их с помощью следующих векторов сдвига:

$$\vec{v}_i = (-h, \dots, -h, 1, -h, \dots, -h), \quad h = -1/(n-1), \quad i = 0, 1, \dots, n-1,$$

где единица стоит на месте i -й координаты. Вектора \vec{v}_i образуют “звезду” в гиперплоскости (19) с равными углами между любой парой векторов. Также в процессе итерации мы храним текущую величину шага по каждой координате $s_i \in \mathbf{R}$, $s_i > 0$, $i = 0, \dots, n-1$. Первоначально все s_i инициализируются константой h/n , где h — параметр алгоритма (например, $h = 0.01$). Также параметром алгоритма является величина ускорения $Acc = 1.2$.

При выборе следующей точки рассматриваются соседи текущей точки T для каждого направления \vec{v}_i , $i = 0, \dots, n-1$. Для фиксированного \vec{v}_i рассматриваются 4 соседние точки T_k , $k = 0, 1, 2, 3$, задаваемые выражениями

$$T_k = T + a_k s_i \vec{v}_i, \quad k = 0, 1, 2, 3, \quad \text{где} \\ a_0 = Acc, \quad a_1 = 1/Acc, \quad a_2 = -a_0, \quad a_3 = -a_1.$$

Для каждого i вычисляются значения целевой функции в четырех соседних точках T_k , $k = 0, \dots, 3$, и выбирается минимальное из соседних значений T_m . Если оно меньше текущего значения, то текущая точка перемещается к точке T_m , а шаг по направлению \vec{v}_i умножается на ускорение a_m :

$$T = T_m, \quad s_i = s_i a_m.$$

Если все 4 соседних значения больше или равны текущему значению целевой функции, то текущая точка не меняет своего положения, но шаг по текущему направлению \vec{v}_i уменьшается в Acc раз, т.е. происходит замедление передвижения по i -му направлению:

$$s_i = s_i / Acc.$$

В процессе перебора соседних точек также проверяется, что точки не выходят за пределы n -мерного ящика, задаваемого ограничениями $0 < t_i < 1$, $i = 0, \dots, n-1$ (соседние точки, выходящие за его пределы, просто не рассматриваются).

Алгоритм завершается, когда на очередном шаге после перебора всех направлений \vec{v}_i , $i = 0, \dots, n-1$, либо текущая точка сдвинулась и новое значение отличается от предыдущего менее чем на ε , либо, если текущая точка осталась на прежней позиции, величины текущих шагов s_i по всем направлениям \vec{v}_i не превосходят ε . На всякий случай также ограничивается общее число шагов алгоритма, например, не более 1000. Преимущество алгоритма Hill Climbing заключается в том, что даже если он не завершается за максимальное число шагов по указанным критериям, то все равно вычисленная текущая точка будет давать значение целевой функции не хуже, чем в исходной точке.

Программная реализация и численные эксперименты

Программа реализована на языке C++, визуализация использует библиотеку классов Qt и систему разработки прикладных программ Qt Creator. Сплайны, использующие первые два метода параметризации кривых Безье, строятся по точкам, задаваемым кликами мыши, причем узлы интерполяции можно захватывать и перетаскивать с помощью мыши, сплайн при этом меняет свою форму в реальном времени.

Оптимальный сплайн вычисляется и изображается по нажатию на кнопку в оконной форме. В экспериментах использовалось значение $\varepsilon = 10^{-12}$ (более чем достаточная точность, на практике хватило бы и значения 10^{-4}).

Узлы интерполяции выбирались более-менее случайно. Число итераций колебалось в пределах 65-85, причем никакой зависимости от числа узлов замечено не было.

Для 20 узлов оптимальный сплайн вычисляется практически мгновенно (эксперименты проводились на довольно медленном ноутбуке). Для 50 узлов время вычисления оптимального сплайна составило менее 0.5 секунды при 79 итерациях, для 100 узлов — около 1.8 с при 76 итерациях, для 200 узлов — 6.4 с при 73 итерациях. Под одной итерацией понимается перебор соседних точек для всех направлений \vec{v}_i , $i = 0, \dots, n-1$, поэтому время выполнения одной итерации пропорционально n . Эксперименты показали, что количество итераций мало зависит от n . При каждом вычислении целевой функции решается линейная система порядка $2n$ методом Гаусса, при этом используется вариант алгоритма для четырехдиагональной матрицы, работающий за время $O(n)$. Получается, что время вычисления оптимального сплайна равно $O(n^2)$, что хорошо согласуется с результатами эксперимента. Отметим, что точность $\varepsilon = 10^{-12}$, использованная в программе, на практике является явно избыточной, с использованием разумного значения ε программа может быть значительно убыстрена. Но предметом данного исследования являлась прежде всего форма оптимального сплайна, а не скорость его вычисления, поэтому и была выбрана максимально высокая точность.

Сделаем еще два важных замечания.

Замечание 1. Помимо обычного сплайна, соединяющего начальный и конечный узлы через серию промежуточных узлов, можно рассматривать также и замкнутый сплайн, у которого начальный узел является одновременно и конечным, причем условия равенства первой и второй производных распространяются и на начальный/конечный узел. Случай замкнутого сплайна даже более простой, поскольку здесь не нужны краевые условия в конечных узлах (уравнения (8) и (9)). Условия равенства производных записываются в виде уравнений (15), (16), где индекс i пробегает значения от 0 до $n-1$, а индексные выражения надо брать по модулю n . Всего получаем систему из $2n$ уравнений с $2n$ неизвестными и невырожденной матрицей, имеющую единственное решение при заданном наборе параметризаций. Оптимальный сплайн вычисляется аналогично.

Замечание 2. В рассмотренном случае незамкнутого сплайна для его однозначного вычисления требовалось задать краевые условия (два дополнительных уравнения). Мы сформулировали их как условие равенства нулю второй производной в краевых узлах. Это не всегда самое удачное решение. Например, пусть мы хотим построить кривую, максимально близкую к дуге окружности. Но у построенного сплайна с указанными краевыми условиями кривизна кривой в начальном и конечном узлах равна нулю, а у окружности кривизна является константой, отличной от нуля. Поэтому в подобной ситуации более естественно использовать следующее краевое условие, которое должно выполняться для начального и конечного сегментов сплайна. Сформулируем его для начального сегмента.

Условие равенства кривизны. Кривизна кривой Безье, соединяющей первый и второй узлы Q_0 и Q_1 , в этих узлах одинакова с учетом ее знака.

На языке контрольных точек Q_0, A_0, B_0, Q_1 это условие выражается следующим образом: четырехугольник $Q_0A_0B_0Q_1$ является равнобедренной трапецией, т.е. $Q_0Q_1 \parallel A_0B_0$, $|Q_0A_0| = |B_0Q_1|$. Запишем его в векторном виде. Обозначим через \vec{N}_0 нормаль к вектору $Q_1 - Q_0$ (если этот вектор имеет координаты (x, y) , то координаты нормали $(-y, x)$). Тогда условие формулируется так:

$$\vec{N}_0 \cdot (B_0 - A_0) = 0, \quad \vec{N}_0 \cdot (A_0 - Q_0) = \vec{N}_0 \cdot (B_0 - Q_1).$$

Здесь точкой \cdot обозначено скалярное произведение векторов. Аналогично записывается это условие для последнего сегмента сплайна:

$$\vec{N}_{n-1} \cdot (B_{n-1} - A_{n-1}) = 0, \quad \vec{N}_{n-1} \cdot (A_{n-1} - Q_{n-1}) = \vec{N}_{n-1} \cdot (B_{n-1} - Q_n),$$

где \vec{N}_{n-1} — нормаль к вектору $Q_n - Q_{n-1}$.

К сожалению, эти уравнения не выражаются в виде линейных комбинаций неизвестных точек A_i, B_i . Перепишав их через координаты точек и векторов, мы получим 4 линейных уравнения относительно координат контрольных точек $A_0, B_0, A_{n-1}, B_{n-1}$. Перепишем также через координаты и уравнения (15) и (16), $i = 1, \dots, n-1$. Всего получим систему из $4n$ уравнений относительно координат точек A_i, B_i с невырожденной матрицей, только в отличие от рассмотренного случая здесь уже неизвестными являются числа, а не точки; также и столбец свободных членов содержит числа. Решив систему, найдем контрольные точки кривых Безье, составляющих сплайн. Вычисление оптимального сплайна выполняется аналогично.

Заключение

Вернемся к рисунку 6, на котором изображены три сплайна, построенные рассмотренными методами по одним и тем же узлам. Рассмотрим еще раз сплайн со стандартной параметризацией $t \in [0, 1]$ и оптимальный сплайн, они показаны на рис. 8.

Оба сплайна изображают скрипичный ключ по 9-ти узлам. Левый сплайн, возможно, и выглядит более стильно, но кривая правого сплайна бесспорно более плавная и гладкая, а именно это и требуется в большинстве приложений. В графическом редакторе можно даже реализовать слайдер, при перемещении которого из крайней левой позиции в крайнюю правую форма кривой будет плавно меняться от левого сплайна к правому, причем все промежуточные сплайны будут проходить через те же самые узлы и иметь непрерывные первую и вторую производные. И происходить это будет в реальном времени, поскольку набор параметризаций $t_i, i = 0, \dots, n-1$ для оптимального сплайна вычисляется только один раз. Стандартный набор параметризаций из всех единиц (или чисел $1/n$) можно по перемещению

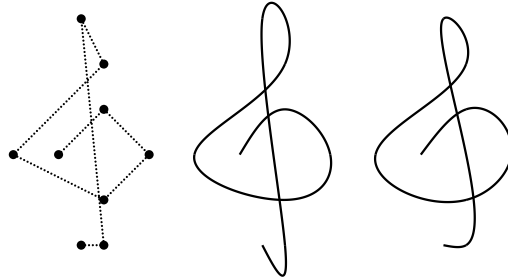


Рис. 8: Сплайны, построенные по одним и тем же узлам интерполяции. Левый сплайн использует стандартную параметризацию. Правый сплайн является оптимальным, минимизирующим интеграл от квадрата второй производной при единичном суммарном времени прохождения сплайна

слайдера плавно преобразовывать в оптимальный набор, а для заданного набора параметризаций сплайн по фиксированным узлам вычисляется мгновенно (за время $O(n)$). На рис. 8 левый сплайн при этом будет плавно деформироваться в правый.

Список литературы

- [1] Bartels, R. H.; Beatty, J. C; and Barsky, B. A. Bézier Curves. Ch. 10 in An Introduction to Splines for Use in Computer Graphics and Geometric Modelling. San Francisco, CA: Morgan Kaufmann, 1998, pp. 211-245.
- [2] A primer on Bézier curves: A free, online book for when you really need to know how to do Bézier things. Retrieved 2016-15-02 from <http://pomax.github.io/bezierinfo/>
- [3] Carl de Boor. A Practical Guide to Splines. Springer-Verlag, 1978, 348 pp. ISBN 3540903569.
- [4] Eugene V. Shikin; Alexander I. Plis. Handbook on Splines for the User. CRC Press. 1995, 96 pp. ISBN 9780849394041.
- [5] Gary D. Knott. Interpolating Cubic Splines. Springer Science & Business Media, 2012, 240 pp. ISBN 1461213207, 9781461213208.
- [6] Jorge Nocedal, Stephen J. Wright. Numerical Optimization. Springer-Verlag, New York, Inc. 1999, 651 pp. ISBN 0-387-98793-2.