Научная визуализация, 2022, том 14, номер 1, страницы 97 - 108, DOI: 10.26583/sv.14.1.08

Об одной методике преобразования сплайн кривой в B-сплайн

А. С. Минкин[1]

Национальный исследовательский центр “Курчатовский институт”, Москва, Россия

[1] ORCID: 0000-0001-8789-0779, amink@mail.ru

Аннотация

Для обеспечения гибкости и удобства работы с геометрическими моделями в CAD системах становится актуальным создание алгоритмов преобразования одних геометрических представлений в другие, среди которых особое место занимают методы их взаимно однозначного и точного преобразования. В данной работе предлагается методика преобразования сплайн кривой в соответствующую ей эквивалентную B-сплайн кривую на основе объединения Безье сегментов с удалением кратных узлов для получения более компактного B-сплайн представления. Приводится обоснование упрощенной версии алгоритма удаления B-сплайн узла. Данный подход позволяет построить B-сплайн кривую на основе информации об отдельных её точках без использования стандартных средств подгонки и сложных схем интерполяции.

Ключевые слова: геометрическое моделирование, параметрические кривые, CAD системы, кубические сплайны, B-сплайн кривые, NURBS, кривые Безье.

1. Введение

Важным этапом проектирования промышленных объектов является создание их геометрической модели, которое обычно производится с использованием специализированного программного обеспечения – CAD систем. Центральной частью CAD системы является геометрическое ядро, позволяющее работать с различными формами кривых и поверхностей для создания двумерных и трехмерных моделей объектов. Как правило, геометрическая модель объекта представлена иерархическим граничным описанием [1], которое состоит в представлении геометрического тела в виде набора фрагментов – граней с общими границами в виде ребер, а для задания криволинейной геометрии используются различные виды параметрических описаний кривых и поверхностей [2-5]. Эффективность работы CAD систем зависит от заложенных в геометрическое ядро алгоритмов работы с различными геометрическими примитивами, в частности, со сплайнами. Подробный обзор и классификация сплайнов приведены в [6], однако рассмотрение всех способов и возможностей их взаимного преобразования выходит за рамки данной статьи. В данной работе рассматриваются свойства кубических сплайнов дефекта 1 и B-сплайнов, предлагается методика преобразования сплайн кривых в B-сплайны без использования специальных средств подгонки B-сплайнов и обсуждаются условия, при которых рассматриваемое преобразование становится возможным.

Если задача состоит в построении математического описания кривых, которое обеспечивает удобный механизм управления их формой, то один из лучших способов достижения указанной цели состоит в использовании рациональных B-сплайнов или NURBS [2,4,5]. B-сплайны являются промышленным стандартом представления составных кривых. Подобно кривым Безье, B-сплайн кривые определяются взвешенной суммой контрольных точек определяющего многоугольника, но, при этом, обладают свойством локальности. NURBS (Non-uniform rational B-spline) представляют собой универсальное математическое описание, охватывающее различные формы кривых и поверхностей, к которым можно отнести прямые, плоскости, конические сечения [7,8], квадрики, поверхности вращения [9] и другие геометрические формы. Кроме того, NURBS представление используется в универсальном формате STEP для обмена данными между CAD системами. Таким образом, NURBS – один из популярных способов параметрического описания кривых и поверхностей, получивший широкое распространение в современных CAD системах, что обусловлено его универсальностью и удобством использования. С одной стороны, NURBS даёт возможность описывать сложные кривые, состоящие из множества параметрических сегментов. С другой стороны, определяющий многоугольник NURBS позволяет управлять формой кривой, но не задаёт явно точек этой кривой, определяя лишь направление её изгиба. В данной работе рассматриваются нерациональные B-сплайны, которые эквивалентны NURBS с единичными весами. В общем случае, B-сплайн кривая может не проходить ни через одну из заданных контрольных точек определяющего многоугольника. В связи с этим, если речь идёт об интерполяции заданных точек, обычно используются сплайны из многочленов невысокого порядка, наибольшее распространение среди которых получили кубические сплайны дефекта 1. По аналогии с физическими сплайнами, для построения математического сплайна используется серия кубических сегментов, каждый из которых проходит через две точки. Кубический сплайн определяется точками, векторами касательных и величинами параметра в концах сегментов, то есть, в отличие от NURBS, явно использует дополнительные сведения о производных в граничных точках.

В общем случае, при неизвестной базовой кривой, задача B-сплайн подгонки решается с использованием более сложных алгоритмов. Для интерполяции множества точек [10], линейных сегментов [11], а также построения кривых перехода [12] могут быть использованы алгоритмы решения линейных систем [13] и различные итерационные схемы [14]. В работе [15] рассмотрена схема интерполяции с подгонкой кривых перехода в виде пары Безье сегментов.

Цель работы состоит в поиске эффективного алгоритма взаимно-однозначного и точного преобразования кривой, заданной в форме кубического сплайна в соответствующее представление в форме B-сплайна. В данной работе рассмотрена методика преобразования кубического сплайна дефекта 1 в B-сплайн на основе объединении Безье сегментов с последующим получением более компактной B-сплайн формы путем удаления единичных узлов. В такой постановке задачи производится переход к представлению кривой в B-сплайн форме за линейное время, в то время как алгоритмы подгонки на основе матричных умножений [16,17], обращений [18,19], а также решения линейных систем [20], имеют, по меньшей мере, квадратичную временную сложность. Алгоритм удаления узла B-сплайна является обратным по отношению к алгоритму его вставки. Существует несколько вариантов алгоритмов вставки узла. Вставка нескольких узлов может быть реализована с помощью алгоритма Осло [21,22], вставка единичного узла – с помощью алгоритма Боэма [23]. Модификация алгоритма Боэма для вставки нескольких узлов рассмотрена в работе [24]. В данной работе удаление узлов используется для получения эквивалентного B-сплайн представления с меньшим числом опорных вершин определяющего многоугольника. В качестве алгоритма удаления узлов, в общем случае, может быть использован любой из рассматриваемых вариантов. В настоящей работе даётся обоснование алгоритма вставки B-сплайн узла и применяется упрощенная версия алгоритма удаления узла для получения наиболее компактного B-сплайн представления исходной сплайн кривой.

В отличие от алгоритмов на основе подгонки B-сплайнов, позволяющих аппроксимировать набор вершин с заранее заданной точностью, предлагаемый алгоритм перехода от сплайнов к B-сплайнам позволяет осуществить эквивалентные преобразования, то есть преобразовать заданную кривую в виде составного кубического сплайна к полностью эквивалентному B-сплайн представлению. Как известно, один и тот же объект может быть представлен с использованием различных B-сплайн описаний. На первом этапе алгоритма выполняется идентификация B-сплайн формы с кратными узловыми значениями, которая точно соответствует заданной сплайн кривой. На втором этапе производится оптимизация полученной B-сплайн формы путем удаления кратных узловых значений, что соответствует более компактному описанию с меньшим числом контрольных точек.

Статья содержит несколько разделов. В разделе 2 дается теоретическая справка, посвященная кубическим сплайнам дефекта 1 и B-сплайнам, определяемым рекурсивными формулами Кокса – Де Бура, а также обсуждаются условия эквивалентности кубических сплайнов Безье сегментам. В разделе 3 дается детальное описание алгоритма перехода от составного кубического сплайна к B-сплайн описанию, а в разделе 4 рассматриваются алгоритмы вставки и упрощенный алгоритм удаления единичного узла для получения эквивалентной B-сплайн формы с меньшим числом узлов. Информация о существующей программной реализации методики приведена в разделе 5. Пример с результатом работы алгоритма рассматривается в разделе 6, за которым следует заключение.

2. Свойства кубических сплайн и B-сплайн кривых

В последние годы были предложены различные формы представления кривых и поверхностей для описания геометрических форм. Среди них особое значение имеют кубические сплайны и B-сплайны.

Согласно [2], кубические сплайны представляют собой кривые наименьшей степени, допускающие точки перегиба и изгиб в пространстве. Рассмотрим интерполяционные кубические сплайны дефекта 1 (разность между степенью и гладкостью сплайна), состоящие из одинаковых сегментов, каждый из которых проходит через две точки. В этом случае, кубическая сплайн кривая может быть описана полиномом 3-й степени с непрерывной второй производной в точках соединения сегментов:

$$P_i(t)=\sum_{j=0}^{3} B_{ij}\left(t-t_i\right)^j,\ t_i \le t \le t_{i+1}, \qquad (1)$$

где B_{ji} – контрольные точки i-го сплайн сегмента, которые определяются по известным координатам точек P_i и P_{i+1}, векторам касательных P_i' и P_{i+1}' и величинам параметра t_i и t_{i+1} в концах сегмента i следующим образом:

$$B_{i0}=P_i, B_{i1}=P_i', B_{i2}=\frac{3\left(P_{i+1}-P_i\right)}{\left(t_{i+1}-t_i\right)^2}-\frac{2P_i'}{t_{i+1}-t_i}-\frac{P_{i+1}'}{t_{i+1}-t_i}, B_{i3}=\frac{2\left(P_i-P_{i+1}\right)}{\left(t_{i+1}-t_i\right)^3}+\frac{P_i'}{\left(t_{i+1}-t_i\right)^2}+\frac{P_{i+1}'}{\left(t_{i+1}-t_i\right)^2}. \qquad (2)$$

Путем несложных преобразований кубический сплайн сегмент можно представить в эквивалентном (2) матричном виде:

$$P_i(\tau)=[F][G],$$

$$[F]=\left[F_{1i}(\tau)\ \ F_{2i}(\tau)\ \ F_{3i}(\tau)\ \ F_{4i}(\tau)\right],[G]^T=\left[P_i\ \ P_{i+1}\ \ P_i'\ \ P_{i+1}'\right],\ 0\le\tau\le 1,$$

$$\tau=\frac{t-t_i}{t_{i+1}-t_i},$$

$$F_{1i}(\tau)=2\tau^3-3\tau^2+1, F_{2i}(\tau)=-2\tau^3+3\tau^2,$$

$$F_{3i}(\tau)=\tau\left(\tau^2-2\tau+1\right)\left(t_{i+1}-t_i\right), F_{4i}(\tau)=\tau\left(\tau^2-\tau\right)\left(t_{i+1}-t_i\right).$$

B-сплайн является промышленным стандартом и используется для описания сложных кривых и поверхностей в большинстве современных CAD систем, что обусловлено следующими свойствами:

- B-сплайн – составная параметрическая кривая, позволяющая при аппроксимации набора точек избежать нежелательных осцилляций, характерных для интерполяционных многочленов высокой степени.

- B-сплайн задаётся вершинами определяющего многоугольника, с помощью которого удобно управлять формой кривой. Локальное изменение узлов не приводит к необходимости заново вычислять всю кривую.
- B-сплайн кривая степени p лежит внутри объединения всех выпуклых оболочек p+1 последовательных вершин определяющего многоугольника.
- Аффинные и проективные преобразования кривых достигаются их применением к вершинам определяющего многоугольника.
- Рациональные B-сплайны (NURBS) позволяют описывать достаточно широкий круг кривых, в частности, кривые Безье и конические сечения.

Перечисленные особенности B-сплайн кривых определяются свойствами B-сплайн базиса.

Пусть $T = (t_0, \ldots, t_r)$ – неубывающая последовательность действительных чисел. Назовем T узловым вектором, а числа t_i – узлами. Обозначим $N_{i,p}(t)$ i-ю нормализованную базисную функцию степени p и определим её следующим образом (формулы Кокса – Де Бура):

$$N_{i,0} = \begin{cases} 1, \text{ if } t_i \le t \le t_{i+1} \\ 0 \text{ иначе} \end{cases}$$

$$N_{i,p}(t) = \frac{t - t_i}{t_{i+p} - t_i} N_{i,p-1}(t) + \frac{t_{i+p+1} - t}{t_{i+p+1} - t_{i+1}} N_{i+1,p-1}(t). \tag{3}$$

Величины t_i – это элементы узлового вектора, удовлетворяющие неравенству $t_0 \le t_i \le t_{i+1} \le t_r$. Чаще всего рассматриваются открытые узловые векторы, для которых число одинаковых элементов в концах на единицу больше, чем степень p базисной функции B-сплайна. Задание p+1 кратных значений в концах узлового вектора позволяет избежать сокращения диапазона параметра t, характерного для B-сплайнов с периодическими узловыми векторами [2], и интерполировать крайние вершины определяющего многоугольника.

Отметим следующие свойства B-сплайн базиса:

1. $N_{i,p}(t) = 0$ для значений t вне интервала $[t_i, t_{i+p+1})$.
2. Для $t \in [t_i, t_{i+1})$ не более p+1 базисных функций принимают ненулевые значения, а именно $N_{i-p,p}(t), \ldots, N_{i,p}(t)$.
3. Базисные функции неотрицательны, т.е. $N_{i,p}(t) \ge 0 \ \forall \ i, p, t$.
4. $\sum_{j=i-p}^{i} N_{j,p}(t) = 1$ для $t \in [t_i, t_{i+1})$.
5. Внутри узлового интервала (t_i, t_{i+1}) $N_{i,p}(t)$ p раз непрерывно дифференцируема. В узле $N_{i,p}(t)$ $p-q$ раз непрерывно дифференцируема, где q – кратность узлового значения.
6. $N_{i,p}(t)$ имеет ровно один максимум (кроме $p = 0$).

Свойства 1 и 2 показывают, что B-сплайн базис является локальным, так как функции $N_{i,p}(t)$ отличны от нуля лишь в определенном интервале, то есть имеют ограниченный носитель. Кратность узловых значений позволяет локально управлять порядком гладкости кривой. В общем случае, согласно свойству 5 базисных функций, задание q кратных узловых значений снижает дифференцируемость B-сплайна до C^{p-q} в узле. Соответственно, задание p кратных узловых значений внутри узлового вектора позволяет интерполировать заданные точки B-сплайн кривой с сохранением её непрерывности. На рис. 1 приведен пример базисных функций B-сплайн кривой при наличии (а) и отсутствии (б) кратных узловых значений. Базисные функции, соответствующие интерполируемым точкам кривой, отмечены на графиках жирными линиями.

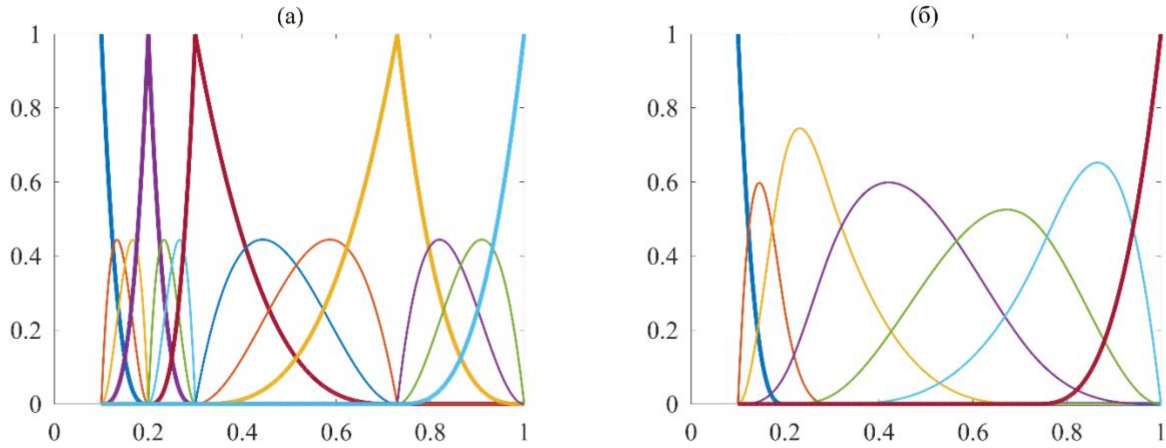

Рисунок 1. Базисные функции модельной B-сплайн кривой:
(а) в случае узлового вектора (0.1,0.1,0.1,0.1,0.2,0.2,0.2,0.3,0.3,0.3,0.73,0.73,0.73,1,1,1,1),
(б) в случае узлового вектора (0.1, 0.1, 0.1, 0.1, 0.2, 0.3, 0.73, 1, 1, 1, 1).

Для задания кривой степени p на множестве узлов T необходимо указать определяющий многоугольник (B_O, ..., B_n). Каждой вершине B_i определяющего многоугольника соответствует базисная функция $N_{i,p}(t)$.Точка B-сплайн кривой вычисляется по значению параметра t следующим образом:

$$P_i(t)=\sum_{j=0}^{n} B_j N_{j,p}(t), t_i \le t \le t_{i+1}, p \le i < r-p.$$

Согласно свойству 2 B-сплайн базиса

$$P_i(t)= \sum_{j=i-p}^{i} B_j N_{j,p}(t), t_i \le t \le t_{i+1}, p \le i < r-p. \tag{4}$$

Каждый сегмент B-сплайн кривой соответствует невырожденному интервалу $t_i \le t \le t_{i+1}$ и определяется p+1 вершинами определяющего многоугольника. Число сегментов составной B-сплайн кривой при отсутствии кратных значений внутри узлового вектора можно получить следующим образом:

$n+1 = ns + p \Rightarrow ns = n - p + 1$,

где n+1 – число контрольных точек определяющего многоугольника. При наличии кратных узловых значений некоторые из этих сегментов можно интерпретировать как вырожденные.

Открытый узловой вектор задаёт параметры сегментов составной B-сплайн кривой, а число его элементов составляет

$nk = ns - 1 + 2 \cdot (p + 1) = n + p + 2$, то есть $r = n + p + 1$.

Задание кратных узловых значений внутри вектора узлов позволяет локально снижать порядок гладкости B-сплайн кривых. В случае B-сплайн кривой третьей степени введение пары кратных узловых значений позволяет *интерполировать* отдельные контрольные точки определяющего многоугольника (рис.1а), совпадающие с точками кривой. Указанное свойство лежит в основе методики преобразования сплайн кривой в B-сплайн форму, рассмотренной в разделе 3 настоящей статьи.

Рассмотрим кубический B-сплайн сегмент с открытым узловым вектором вида $T_1=(t_i,t_i,t_i,t_i, t_{i+1},t_{i+1},t_{i+1},t_{i+1})$. В таком виде B-сплайн кривая эквивалентна кривой Безье, а B-сплайн базис эквивалентен базису Бернштейна [2], что может быть, в общем случае, доказано индукцией по степени полинома. Кубический Безье сегмент, соответствующий узловому вектору T_1, может быть вычислен по p+1=4 контрольным точкам следующим образом:

$$P_i(t)=\left(\frac{t_{i+1}-t}{t_{i+1}-t_i}\right)^3 B_{i0}+3\left(\frac{t-t_i}{t_{i+1}-t_i}\right)\left(\frac{t_{i+1}-t}{t_{i+1}-t_i}\right)^2 B_{i1}+3\left(\frac{t-t_i}{t_{i+1}-t_i}\right)^2\left(\frac{t_{i+1}-t}{t_{i+1}-t_i}\right) B_{i2}+\left(\frac{t-t_i}{t_{i+1}-t_i}\right)^3 B_{i3}. \tag{5}$$

Для $t=t_i$ $P_i(t_i)=B_{i0}$, для $t=t_{i+1}$ $P_i(t_{i+1})=B_{i3}$.

Дифференцированием по параметру *t* получаем

$$P_i'(t)=\frac{1}{(t_{i+1}-t_i)^3}\Big\{-3(t_{i+1}-t)^2 B_{i0}+3\Big[(t_{i+1}-t)^2-2(t_{i+1}-t)(t-t_i)\Big]B_{i1}+$$
$$+3\Big[2(t-t_i)(t_{i+1}-t)-(t-t_i)^2\Big]B_{i2}+3(t-t_i)^2 B_{i3}\Big\}.$$

Для $t=t_i$ $P_i'(t_i)=-\frac{3B_{i0}}{t_{i+1}-t_i}+\frac{3B_{i1}}{t_{i+1}-t_i}$, то есть $B_{i1}=B_{i0}+(1/3)(t_{i+1}-t_i)P_i'(t_i)$, (6)

для $t=t_{i+1}$ $P_i'(t_{i+1})=-\frac{3B_{i2}}{t_{i+1}-t_i}+\frac{3B_{i3}}{t_{i+1}-t_i}$, то есть $B_{i2}=B_{i3}-(1/3)(t_{i+1}-t_i)P_i'(t_{i+1})$, (7)

то есть контрольные точки B_{i1} и B_{i2} могут быть вычислены на основе информации о граничных точках B_{i0} и B_{i3} и производных P_i' и P_{i+1}' в граничных точках, что соответствует аналогичным формулам производных в граничных точках NURBS кривых с открытым узловым вектором [5].

Покажем, что представления (1), (2) эквивалентны (5). Рассмотрим для простоты частный случай кубических сегментов для стандартного диапазона параметров $0 \le t \le 1$. В этом случае, согласно формулам (1) и (2), сплайн кривую можно записать в следующем виде:

$P_i(t)=B_{i0}^{(1)}+B_{i1}^{(1)}t+B_{i2}^{(1)}t^2+B_{i3}^{(1)}t^3, 0\le t\le 1,$ где

$$B_{i0}^{(1)}=P_i, B_{i1}^{(1)}=P_i', B_{i2}^{(1)}=3(P_{i+1}-P_i)-2P_i'-P_{i+1}', B_{i2}^{(1)}=2(P_i-P_{i+1})+P_i'+P_{i+1}'.$$

Уравнение, определяющее Безье сегмент согласно (5) для $0 \le t \le 1$ можно переписать так:

$$P_i(t)=(1-t)^3 B_{i0}^{(2)}+3t(1-t)^2 B_{i1}^{(2)}+3t^2(1-t)B_{i2}^{(2)}+t^3B_{i3}^{(2)}=$$
$$=\left(3B_{i1}^{(2)}-3B_{i2}^{(2)}+B_{i3}^{(2)}-B_{i0}^{(2)}\right)t^3+\left(3B_{i0}^{(2)}-6B_{i1}^{(2)}+3B_{i2}^{(2)}\right)t^2+\left(3B_{i1}^{(2)}-3B_{i2}^{(2)}\right)t+B_{i0}^{(2)}.$$

Используя выражения (6), (7) для граничных производных Безье сегмента и приравнивая коэффициенты при одинаковых степенях, получаем

$$B_{i0}^{(1)}=B_{i0}^{(2)}=P_i,$$
$$B_{i1}^{(1)}=3B_{i1}^{(2)}-3B_{i0}^{(2)}=3\left(B_{i0}^{(2)}+(1/3)P_i'\right)-3B_{i0}^{(2)}=P_i',$$
$$B_{i2}^{(1)}=3B_{i0}^{(2)}-6B_{i1}^{(2)}+3B_{i2}^{(2)}=3B_{i0}^{(2)}-6\left(B_{i0}^{(2)}+(1/3)P_i'\right)+3\left(B_{i3}^{(2)}-(1/3)P_{i+1}'\right)=3(P_{i+1}-P_i)-2P_i'-P_{i+1}',$$
$$B_{i2}^{(1)}=3B_{i1}^{(2)}-3B_{i2}^{(2)}+B_{i3}^{(2)}-B_{i0}^{(2)}=3\left(B_{i0}^{(2)}+(1/3)P_i'\right)-3\left(B_{i3}^{(2)}-(1/3)P_{i+1}'\right)+B_{i3}^{(2)}-B_{i0}^{(2)}=$$
$$=2(P_i-P_{i+1})+P_i'+P_{i+1}'.$$

Таким образом, в указанном случае Безье сегменты эквивалентны кубическим сплайнам дефекта 1, определенным формулами (1) и (2).

3. Постановка задачи. Методика преобразования сплайн кривой в B-сплайн

Пусть исходная сплайн кривая состоит из *ns* сегментов, $(t_0, t_1,..., t_{ns})$ – вектор параметров; $P_0, P_1,..., P_{ns}$ и $P_0', P_1',..., P_{ns}'$ множество точек и векторы касательных по концам сегментов. Необходимо преобразовать сплайн кривую, заданную в виде множества сегментов, в B-сплайн форму. Для достижения указанной цели формируется Безье сегмент, соответствующий каждому сегменту сплайн кривой, а вся совокупность Безье сегментов преобразовывается в B-сплайн кривую на основе интерполяции концевых точек с помощью добавления кратных узловых значения, то есть:

1. Для каждого сплайн сегмента i вычисляется Безье сегмент по формулам:
   $B_{i0}=P_i$, $B_{i1}=B_{i0}+(1/3)(t_{i+1}-t_i)P_i'$, $B_{i2}=P_{i+1}-(1/3)(t_{i+1}-t_i)P_{i+1}'$, $B_{i3}=P_{i+1}$.
2. Пусть B_{ij} – контрольная точка с номером j, соответствующая сегменту i. Предполагая, что $B_{00}=P_0$, $B_{03}=B_{10}=P_1$, $B_{13}=B_{20}=P_2$, …, $B_{i3}=B_{i+1,0}=P_i$, …, $B_{ns-2,3}=B_{ns-1,0}=P_{ns-1}$, $B_{ns-1,3}=P_{ns}$; формируется определяющий многоугольник B-сплайн в следующем виде: $(B_{00},B_{01},B_{02},B_{03},B_{11},B_{12},B_{13},\ldots B_{ns-1,1},B_{ns-1,2},B_{ns-1,3})$, $n=p\cdot ns$, где $n+1$ – число контрольных точек определяющего многоугольника.
3. Для интерполяции концевых вершин B_{03}, B_{13}, …, $B_{ns-2,3}$, $B_{ns-1,3}$ задаётся открытый узловой вектор B-сплайн кривой с кратными узловыми значениями в следующем виде:
   $(t_0, t_0, t_0, t_0, t_1, t_1, t_1, t_2, t_2, t_2, \ldots, t_{ns-1}, t_{ns-1}, t_{ns-1}, t_{ns}, t_{ns}, t_{ns}, t_{ns})$, $nk=p\cdot(ns+1)+2$.

Таким образом, исходная сплайн кривая может быть преобразована в B-сплайн с кратными узловыми значениями. Вычислительная сложность такого преобразования составляет $O(ns)$. Полученный B-сплайн интерполирует часть контрольных точек, соответствующих множеству точек P_0, P_1,…, P_{ns} многосегментной сплайн кривой. Задачу, поставленную таким образом, можно считать решённой, но такое представление кривой в форме B-сплайн является избыточным по числу сегментов и может быть упрощено путем преобразования в более компактное представление, содержащее меньшее число контрольных точек и узлов, соответственно. Соответствующее преобразование может быть сделано с помощью алгоритма удаления узла, который рассматривается далее.

4. Алгоритмы вставки и удаления единичного узла

Алгоритмы вставки и удаления узлов рассмотрены в литературных источниках [5,12]. Ниже приведено альтернативное обоснование алгоритма вставки единичного узла [5].

Пусть $(B_0^{(1)},\ldots,B_n^{(1)})$ – определяющий многоугольник исходной B-сплайн кривой, а $T^{(1)}=(t_0,\ldots,t_r)$ – её узловой вектор. Предположим, что необходимо произвести вставку узла $\bar{t}\in[t_i,t_{i+1})$, $p\le i<r-p$. Поставим задачу получить новое B-сплайн описание без изменения формы кривой. Число элементов узлового вектора является функцией суммы числа контрольных точек и степени кривой, поэтому при неизменной степени число контрольных точек после вставки единичного узла возрастает на единицу. Если при этом $(B_0^{(2)},\ldots,B_{n+1}^{(2)})$ задаёт определяющий многоугольник преобразованной кривой, $N_{j,p}(t)$ и $\bar{N}_{j,p}(t)$ – базисные функции исходной и преобразованной кривой, а

$$T^{(2)}=(\bar{t}_0=t_0,\ldots,\bar{t}_i=t_i,\bar{t}_{i+1}=t,\bar{t}_{i+2}=t_{i+1},\bar{t}_{r+1}=t_r)\ – \tag{8}$$

узловой вектор преобразованной кривой, то

$$\sum_{j=0}^{n}B_j^{(1)}N_{j,p}(t)=\sum_{j=0}^{n+1}B_j^{(2)}\bar{N}_{j,p}(t).$$

Так как форма кривой остаётся неизменной, то, согласно формуле (4), для $\forall t\in[t_i,t_{i+1})$

$$B_j^{(1)}=B_j^{(2)},\ N_{j,p}(t)=\bar{N}_{j,p}(t)\ \text{для } j=0,\ \ldots,\ i-p-1, \tag{9}$$

$$B_j^{(1)}=B_{j+1}^{(2)},\ N_{j,p}(t)=\bar{N}_{j,p}(t)\ \text{для } j=i+1,\ \ldots,\ n, \tag{10}$$

$$\sum_{j=i-p}^{i}B_j^{(1)}N_{j,p}(t)=\sum_{j=i-p}^{i+1}B_j^{(2)}\bar{N}_{j,p}(t). \tag{11}$$

Можно показать, что базисные функции исходной кривой связаны с базисными функциями преобразованной кривой следующим образом:

$$N_{j,p}(t) = \frac{\bar{t} - \bar{t}_j}{\bar{t}_{j+p+1} - \bar{t}_j} \bar{N}_{j,p}(t) + \frac{\bar{t}_{j+p+2} - \bar{t}}{\bar{t}_{j+p+2} - \bar{t}_{j+1}} \bar{N}_{j+1,p}(t) \text{ для } j=i-p, \ldots, i. \quad (12)$$

Идея доказательства данной формулы приведена в [5], доказательство для ненормализованных B-сплайнов в [12], доказательство на основе разделенных разностей в [21]. Ниже по тексту приводится более простое доказательство формулы (12). Легко видеть связь между данной формулой и формулой Кокса–Де Бура (3):

$$N_{j,p}(t) = \bar{N}_{j,p+1}(t). \quad (13)$$

Формула (12) может быть доказана индукцией по p на основе формул (3) и (13) с учётом (8). Для индуктивного перехода получаем

$$N_{j,p+1}(t) = \frac{t - t_j}{t_{j+p+1} - t_j} N_{j,p}(t) + \frac{t_{j+p+2} - t}{t_{j+p+2} - t_{j+1}} N_{j+1,p}(t) = \frac{\bar{t} - \bar{t}_j}{\bar{t}_{j+p+2} - \bar{t}_j} N_{j,p}(t) + \frac{\bar{t}_{j+p+3} - \bar{t}}{\bar{t}_{j+p+3} - \bar{t}_{j+1}} N_{j+1,p}(t) =$$

$$= \frac{\bar{t} - \bar{t}_j}{\bar{t}_{j+p+2} - \bar{t}_j} \bar{N}_{j,p+1}(t) + \frac{\bar{t}_{j+p+2} - \bar{t}}{\bar{t}_{j+p+3} - \bar{t}_{j+1}} \bar{N}_{j+1,p+1}(t).$$

Связь между $B_j^{(1)}$ и $B_j^{(2)}$ для $j=i-p, \ldots, i$ можно получить, подставив формулу (12) в (11):

$$\left(\frac{\bar{t} - \bar{t}_{i-p}}{\bar{t}_{i+1} - \bar{t}_{i-p}} \bar{N}_{i-p,p} + \frac{\bar{t}_{i+2} - \bar{t}}{\bar{t}_{i+2} - \bar{t}_{i-p+1}} \bar{N}_{i-p+1,p} \right) B_{i-p}^{(1)} + \left(\frac{\bar{t} - \bar{t}_{i-p+1}}{\bar{t}_{i+2} - \bar{t}_{i-p+1}} \bar{N}_{i-p+1,p} + \frac{\bar{t}_{i+3} - \bar{t}}{\bar{t}_{i+3} - \bar{t}_{i-p+2}} \bar{N}_{i-p+2,p} \right) B_{i-p+1}^{(1)} +$$

$$+ \ldots + \left(\frac{\bar{t} - \bar{t}_i}{\bar{t}_{i+p+1} - \bar{t}_i} \bar{N}_{i,p} + \frac{\bar{t}_{i+p+2} - \bar{t}}{\bar{t}_{i+p+2} - \bar{t}_{i+1}} \bar{N}_{i+1,p} \right) B_i^{(1)} = \bar{N}_{i-p,p} B_{i-p}^{(2)} + \bar{N}_{i-p+1,p} B_{i-p+1}^{(2)} + \ldots + \bar{N}_{i+1,p} B_{i+1}^{(2)}.$$

Учитывая, что $\bar{t}_{i+1} = \bar{t}$ и используя узловой вектор $T^{(1)}$ вместо $T^{(2)}$, получаем

$$0 = \left(B_{i-p}^{(2)} - B_{i-p}^{(1)} \right) \bar{N}_{i-p,p} + \left(B_{i-p+1}^{(2)} - \frac{t_{i+1} - \bar{t}}{t_{i+1} - t_{i-p+1}} B_{i-p}^{(1)} - \frac{\bar{t} - t_{i-p+1}}{t_{i+1} - t_{i-p+1}} B_{i-p+1}^{(1)} \right) \bar{N}_{i-p+1,p} + \ldots +$$

$$+ \left(B_i^{(2)} - \frac{t_{i+p} - \bar{t}}{t_{i+p} - t_i} B_{i-1}^{(1)} - \frac{\bar{t} - t_i}{t_{i+p} - t_i} B_i^{(1)} \right) \bar{N}_{i-p+1,p} + \left(B_{i+1}^{(2)} - B_i^{(1)} \right) \bar{N}_{i+1,p}.$$

Обозначив через $\alpha_j = (\bar{t} - t_j) / (t_{j+p} - t_j)$ для $j=i-p+1, \ldots, i$, можно выразить координаты новых вершин ($B_0^{(2)}, \ldots, B_{n+1}^{(2)}$) по известным координатам вершин ($B_0^{(1)}, \ldots, B_n^{(1)}$) определяющего многоугольника исходной кривой:

$$B_{i-p}^{(2)} = B_{i-p}^{(1)}$$

$$B_j^{(2)} = \frac{t_{j+p} - \bar{t}}{t_{j+p} - t_j} B_{j-1}^{(1)} + \frac{\bar{t} - t_j}{t_{j+p} - t_j} B_j^{(1)} = (1 - \alpha_j) B_{j-1}^{(1)} + \alpha_j B_j^{(1)} \text{ для } j=i-p+1, \ldots, i. \quad (14)$$

$$B_{i+1}^{(2)} = B_i^{(1)}.$$

Переписывая формулы (14) в обратном порядке и учитывая (9) и (10), получаем последовательность вычислений для удаления единичного узла:

$$B_j^{(1)} = B_j^{(2)} \text{ для } j=0, \ldots, i-p,$$

$$B_j^{(1)} = \begin{cases} \left(B_j^{(2)} - (1 - \alpha_j) B_{j-1}^{(1)} \right) / \alpha_j, \text{ если } \alpha_j > 0 \\ B_{j+1}^{(2)}, \text{ если } \alpha_j = 0 \end{cases} \text{ для } j=i-p+1, \ldots, i.$$

$$B_j^{(1)} = B_{j+1}^{(2)} \text{ для } j=i+1, \ldots, n.$$

В данной постановке задачи представленный алгоритм удаления узла может быть применен дважды для каждого узла двойной кратности с целью получения более компактного B-сплайн описания. Каждая операция удаления узла уменьшает число контрольных точек на единицу, не изменяя число сегментов B-сплайна, что позволяет построить алгоритм перехода к эквивалентному B-сплайн представлению за линейное время, то есть обладает вычислительной сложностью O(ns), где ns – число сегментов исходной кривой в форме кубического сплайна.

Данный алгоритм не содержит операции проверки возможности удаления узла, несмотря на то, что, в общем случае, согласно свойству 5 базисных функций, B-сплайн кривая имеет разрывную первую производную в точках тройной кратности узлового значения. Тем не менее, возможность двукратного удаления узла обоснована тем, что исходный кубический сплайн (дефекта 1) имеет непрерывность второй производной в точках внутреннего соединения, что позволяет отказаться от более сложного алгоритма удаления узлов, рассмотренного в [5]. В целом, методика преобразования кубического сплайна дефекта 1 в B-сплайн может быть реализована в виде алгоритма с вычислительной сложностью O(ns).

5. Программная реализация

Методика преобразования сплайн кривой в B-сплайн реализована в виде пакета программ на языке C++. Алгоритмы итерационного вычисления сплайн и B-сплайн кривых, объединения Безье сегментов и удаления узлов B-сплайна, а также общий алгоритм, реализующий методику преобразования и оптимизации B-сплайн кривой, представлены в виде отдельных функций и программных модулей. Визуализация кривых и базисных функций B-сплайнов реализована в виде сценариев в системе матричных вычислений GNU Octave. Перечисленные программные компоненты находятся в свободном доступе по адресу https://github.com/amo606/bsplfit.

6. Пример работы алгоритмов

Рассмотрим сплайн кривую на рис 2а, с контрольными точками и касательными векторами, представленными в Табл.1.

Таблица 1. Контрольные точки и касательные векторы сплайн кривой

Контрольные точки	Касательные векторы
(1, 1)	(60, 60)
(3.90873, 2.4881)	(12.2619, -0.357143)
(4.91607, 3.31514)	(8.43441, 10.8827)
(7.24717, 5.63957)	(4.07908, 9.08418)
(10, 6)	(22.2222, -22.2222)

Кривая определена 5 контрольными точками и касательными векторами, то есть состоит из 4 сегментов. Узловой вектор представлен следующим набором чисел: (0.1, 0.2, 0.3, 0.73, 1).

Преобразование составной сплайн кривой на рис. 2а в B-сплайн форму на основе объединения Безье сегментов даёт следующее описание B-сплайн кривой:

- Вектор узлов (17 узловых значений):
  (0.1, 0.1, 0.1, 0.1, 0.2, 0.2, 0.2, 0.3, 0.3, 0.3, 0.73, 0.73, 0.73, 1, 1, 1, 1).
- Контрольные точки (13 точек, n=12): (1, 1), (3, 3), (3.5, 2.5), (3.90873, 2.4881), (4.31746, 2.4762), (4.63492,2.95238), (4.91607,3.31514), (6.125,4.87499), (6.6625,4.3375), (7.24717, 5.63957), (7.61429, 6.45715), (8, 8), (10, 6).

Удаление кратных узлов по границам Безье сегментов позволяет получить следующее описание B-сплайн кривой (рис. 2б):

- Вектор узлов (11 узловых значений): (0.1, 0.1, 0.1, 0.1, 0.2, 0.3, 0.73, 1, 1, 1, 1).
- Контрольные точки (7 точек, n=6): (1,1),(3,3),(4,2),(6,5),(7,4),(8,8),(10,6).

B-сплайн кривой на рис. 2а соответствуют базисные функции на рис. 1а, а B-сплайн кривой на рис. 2б соответствуют базисные функции на рис. 1б.

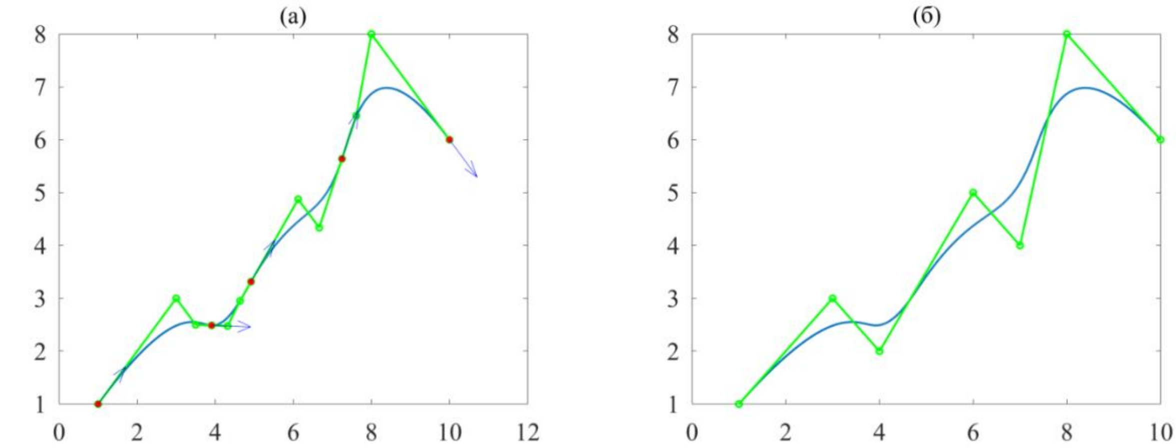

Рисунок 2. Пример кривой, представленной кубическим сплайном и B-сплайном:
(а) Составная сплайн кривая и соответствующий ей B-сплайн, составленный из набора Безье сегментов. Жирными точками обозначены границы сплайн сегментов, стрелками – направления касательных по границам этих сегментов, ломаная линия – определяющий многоугольник B-сплайн.
(б) Оптимальная B-сплайн форма. Результат работы алгоритма удаления кратных узлов.

Таким образом, объединение Безье сегментов, эквивалентных сплайн сегментам, позволяет получить B-сплайн кривую с кратными узлами, которую затем можно упростить с помощью рассмотренного алгоритма удаления узла. Данный алгоритм применяется 6 раз, что уменьшает число контрольных точек B-сплайна с 13 до 7 путем удаления вырожденных сегментов.

7. Заключение

В данной работе рассматривается взаимосвязь между представлениями кривых в форме кубических сплайнов дефекта 1 и B-сплайнов, которые могут быть использованы для описания одной и той же кривой. Представление кривой в виде множества сплайн сегментов позволяет интерполировать набор заранее заданных контрольных точек, однако требует дополнительной информации о производных на границах сегментов. В свою очередь B-сплайн кривые не требуют явного задания производных, но, в общем случае, не интерполируют контрольные точки определяющего многоугольника. В то же время, B-сплайн является промышленным стандартом, удобным для обмена данными между разными CAD системами, что делает актуальной задачу B-сплайн подгонки к заданному набору вершин, а также преобразования альтернативных представлений кривых в форму B-сплайна. Такая задача может быть эффективно решена для кривых, заданных кубическими сплайнами дефекта 1.

Вставка узлов позволяет представить одну и ту же кривую в виде различных B-сплайн форм. В данной работе приведено альтернативное обоснование алгоритмов вставки и удаления узлов B-сплайнов, а также условия эквивалентности кубических сплайнов Безье сегментам третьей степени. B-сплайн кривые могут быть составлены из отдельных Безье сегментов, а сплайн, состоящий из множества сегментов, можно представить в виде эквивалентного B-сплайн описания с кратными узлами, что позволяет интерполировать отдельно взятые вершины. Полученный таким образом B-сплайн можно преобразовать в более компактную форму с помощью алгоритма удаления единичных узлов, который может быть упрощен с учетом свойств кубических сплайнов дефекта 1. Такое представление является оптимальным и будет полностью соответствовать исходному описанию кривой в форме кубического сплайна.

Таким образом, в данной работе предложен оригинальный способ построения В-сплайн кривой на основе её исходного представления в виде кубического сплайна дефекта 1, состоящего из множества сегментов, реализованный в виде алгоритма. Рассмотренный алгоритм позволяет преобразовать сплайн кривую в точно соответствующую ей В-сплайн кривую и представляет собой альтернативу алгоритмам подгонки В-сплайн кривых и применения сложных схем интерполяции. Алгоритм может быть реализован в виде программного модуля в составе геометрического ядра CAD системы.

Список литературы

1. Hiemstra R.R., Shepherd K.M., Johnson M.J., Quan L., Hughes T.J.R., Towards untrimmed NURBS: CAD embedded reparameterization of trimmed B-rep geometry using frame-field guided global parameterization // Comput n Appl Mech Eng. —2020. — Vol.369.— 113227.
2. Роджерс Д., Адамс Дж. Математические основы компьютерной графики. — М.: Мир, 2001.
3. Фокс А., Пратт М. Вычислительная геометрия: применение в проектировании и на производстве. — М.: Мир, 1982.
4. G. Farin. Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide // Academic Press Inc. — 1993.
5. Piegl L.A., Tiller W. The NURBS Book. Second edition. New York: Springer–Verlag.— 1995–1997.
6. Задорожный А.Г., Киселев Д.С. Построение сплайнов с использованием библиотеки OpenGL.— Новосибирск: Изд-во НГТУ, 2019.
7. Lee E. Rational Bezier Representation for Conies // Geometric Modeling. —1986. — Farin G. (ed.), SIAM. — P. 3–27.
8. Blanc C., Schlick C. Accurate parametrization of conics by NURBS // IEEE Computer Graphics and Applications. — 1996. Vol.16, No.6. — P.64–71.
9. Dimas E., Briassoulis D. 3D geometric modelling based on NURBS: a review // Adv Eng Softw. —1999. —Vol.30. — P. 741–751.
10. Wang W., Pottmann H., Liu Y. Fitting B-spline curves to point clouds by curvature-based squared distance minimization // ACM Trans. Graph. —2006. —Vol.25. —P. 214–238.
11. Sun S., Yu D., Wang C., Xie C. A smooth tool path generation and real-time interpolation algorithm based on B-spline curves // Adv Mech Eng. —2018. Vol.10, No.1. — P. 1-14.
12. Голованов Н.Н. Геометрическое моделирование. Учебное пособие. — М.: КУРС: ИНФРА-М, 2016.
13. Ueng W.-D., Lai J.-Y., Tsai Y.-C. Unconstrained and constrained curve fitting for reverse engineering // Int J Adv Manuf Technol. —2007. Vol. 33. — P. 1189–1203.
14. Wang G., Shu Q., Wang J. et al. Research on adaptive non-uniform rational B-spline real-time interpolation technology based on acceleration constraints // Int J Adv Manuf Technol. —2017. —Vol.91. — P. 2089–2100.
15. Zhao H., Zhu L., Ding H. A real-time look-ahead interpolation methodology with curvature-continuous B-spline transition scheme for CNC machining of short line segments // Int J Mach Tools Manuf. — 2013. —Vol. 65. — P. 88–98.
16. Coppersmith D. Winograd S. (1990), Matrix multiplication via arithmetic progressions, J. Symb. Comput. —1990. —Vol. 9 No,3. —P. 251.
17. Cohn H., Kleinberg R., Szegedy B., Umans C. Group-theoretic algorithms for matrix multiplication// 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05). —2005. — P. 379–388.
18. Tveit A. On the complexity of matrix inversion // Mathematical Note. —2003.

19. Krishnamoorthy A., Menon D. Matrix inversion using Cholesky decomposition // Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA).— 2013.— P. 70–72.
20. Pan V. Computer Algorithms for Solving Linear Algebraic Equations // Springer Berlin Heidelberg. —1991. — P.27–56.
21. Cohen E., Lyche T., Riesebfeld R. Discrete B-Splines and Subdivision Techniques in Computer-Aided Geometric Design and Computer Graphics // Comput Graph Image Process.—1980. 14. P. 87-111.
22. Prautzsch H. A short proof of the Oslo algorithm // Comput Aided Geom Des. — 1984.—Vol.1,No.1. — P. 95–96.
23. Boehm W. Inserting new knots into B-spline curves // Comput Aided Des. —1980. — Vol.12, No. 4. — P. 199–201.
24. Boehm W., Prautzsch H. The insertion algorithm // Comput Aided Des. —1985. — Vol.17, No.2. — P. 58–59.

Scientific Visualization, 2022, volume 14, number 1, pages 97 - 108, DOI: 10.26583/sv.14.1.08

One of The Algorithms for Converting Spline Interpolated Curves into B-spline Form

A. S. Minkin[1]

National Research Center «Kurchatov Institute», Moscow, Russia

[1] ORCID: 0000-0001-8789-0779, amink@mail.ru

Abstract

To ensure flexibility and convenience of working with geometric models in CAD systems, algorithms for converting geometric representations are demanded, among which methods of their one-to-one and exact transformation are of great importance. In this paper, we propose a technique for converting a spline curve into a corresponding equivalent B-spline curve based on combining Bezier segments and the removal of multiple knots to obtain a more compact B-spline representation. The justification of the simplified version of the knot-removal algorithm for B-splines is given. This approach makes it possible to construct a B-spline curve based on information about its individual points without using standard fitting tools and complex interpolation schemes.

Keywords: geometric modeling, parametric curves, CAD, cubic splines, B-spline curves, NURBS, Bezier curves.

References

1. Hiemstra R.R., Shepherd K.M., Johnson M.J., Quan L., Hughes T.J.R., Towards untrimmed NURBS: CAD embedded reparameterization of trimmed B-rep geometry using frame-field guided global parameterization // Comput n Appl Mech Eng. —2020. —Vol.369. —113227.
2. Rogers D.F.,Adams J.A. Mathematical elements for computer graphics. 2nd Edition, McGraw-Hill, New York,1990 (Russ. ed.: Rogers D.F.,Adams J.A. Matematicheskie osnovy komp'juternoj grafiki. — M.: Mir, 2001).
3. Faux I.D., Pratt M.J. Computational geometry for design and manufacture, Chichester, West Sussex, John Willey & sons, 1979 (Russ. ed.: Faux I.D., Pratt M.J. Vychislitel'naja geometrija: primenenie v proektirovanii i na proizvodstve. — M.: Mir, 1982).
4. G. Farin. Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide // Academic Press Inc. —1993.
5. Piegl L.A., Tiller W. The NURBS Book. Second edition. New York: Springer–Verlag.—1995–1997.
6. Zadorozhnyi A.G., Kiselev D.S. Postroyenie splaynov s ispol’zovaniem biblioteki OpenGL. —Novosibirsk: NNGU, 2019.
7. Lee E. Rational Bezier Representation for Conies // Geometric Modeling. —1986. — Farin G. (ed.), SIAM. —P. 3–27.
8. Blanc C., Schlick C. Accurate parametrization of conics by NURBS // IEEE Computer Graphics and Applications. —1996. Vol.16, No.6. —P.64–71.
9. Dimas E., Briassoulis D. 3D geometric modelling based on NURBS: a review // Adv Eng Softw. —1999. —Vol.30. —P. 741–751.
10. Wang W., Pottmann H., Liu Y. Fitting B-spline curves to point clouds by curvature-based squared distance minimization // ACM Trans. Graph. —2006. —Vol.25. —P. 214–238.
11. Sun S., Yu D., Wang C., Xie C. A smooth tool path generation and real-time interpolation algorithm based on B-spline curves // Adv Mech Eng. —2018. Vol.10, No.1. —P. 1-14.

12. Golovanov N. Geometric Modeling: The Mathematics of Shapes. CreateSpace Independent Publishing Platform, 2014 (Russ. ed.: Golovanov N. Geometricheskoe modelirovanie. Uchebnoe posobie. — M.: KURS: INFRA-M, 2016).

13. Ueng W.-D., Lai J.-Y., Tsai Y.-C. Unconstrained and constrained curve fitting for reverse engineering // Int J Adv Manuf Technol. —2007. Vol. 33. —P. 1189–1203.

14. Wang G., Shu Q., Wang J. et al. Research on adaptive non-uniform rational B-spline real-time interpolation technology based on acceleration constraints // Int J Adv Manuf Technol. —2017. —Vol.91. —P. 2089–2100.

15. Zhao H., Zhu L., Ding H. A real-time look-ahead interpolation methodology with curvature-continuous B-spline transition scheme for CNC machining of short line segments // Int J Mach Tools Manuf. —2013. —Vol. 65. —P. 88–98.

16. Coppersmith D. Winograd S. (1990), Matrix multiplication via arithmetic progressions // J. Symb. Comput. —1990. —Vol. 9 No,3. —P. 251.

17. Cohn H., Kleinberg R., Szegedy B., Umans C. Group-theoretic algorithms for matrix multiplication// 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05). —2005. —P. 379-388.

18. Tveit A. On the complexity of matrix inversion // Mathematical Note. —2003.

19. Krishnamoorthy A., Menon D. Matrix inversion using Cholesky decomposition // Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA). — 2013. — P. 70-72.

20. Pan V. Computer Algorithms for Solving Linear Algebraic Equations // Springer Berlin Heidelberg. —1991. — P.27–56.

21. Cohen E., Lyche T., Riesebfeld R. Discrete B-Splines and Subdivision Techniques in Computer-Aided Geometric Design and Computer Graphics // Comput Graph Image Process.— 1980. 14. P. 87-111.

22. Prautzsch H. A short proof of the Oslo algorithm // Comput Aided Geom Des. — 1984. —Vol.1,No.1. — P. 95–96.

23. Boehm W. Inserting new knots into B-spline curves // Comput Aided Des.—1980. —Vol.12, No. 4. —P. 199–201.

24. Boehm W., Prautzsch H. The insertion algorithm // Comput Aided Des.—1985. —Vol.17, No.2. —P. 58–59.