электронное научно-техническое издание
НАУКА и ОБРАЗОВАНИЕ
Эл № ФС 77 - 30569. Государственная регистрация №0420900025. ISSN 1994-0408

Аппроксимация профилей лопаток компрессора с помощью кривых Безье

 # 07, июль 2010
автор: Грушин М. А.

УДК.519.6

mihail.grushin@gmail.com

МГТУ им. Н.Э.Баумана

Введение

В практике моделирования лопаточных венцов компрессоров газотурбинных двигателей (ГТД) построение геометрической модели пера лопатки обычно ведётся по её плоским сечениям (см. рис. В.1).

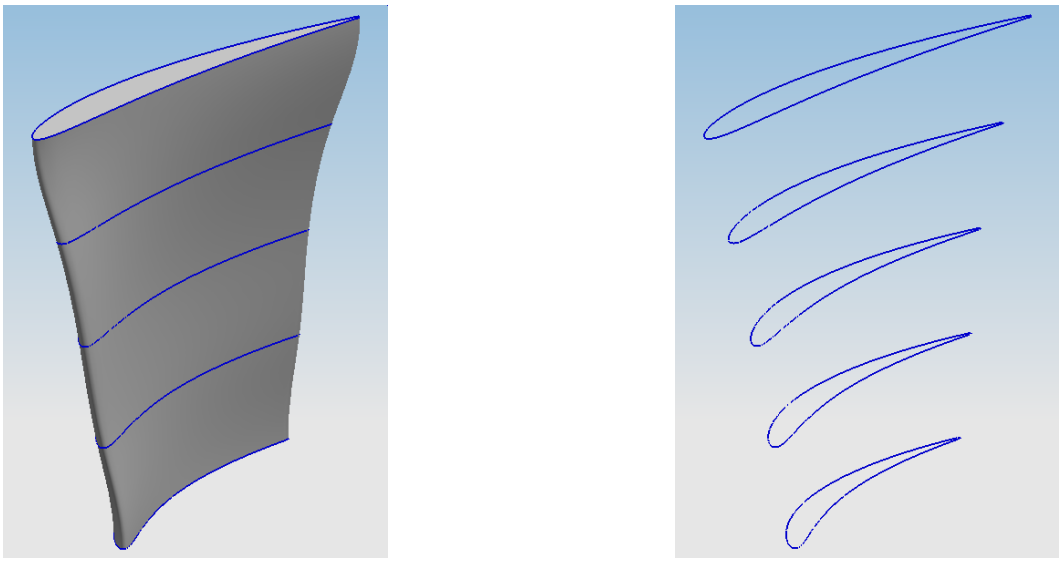

Рис. В.1 Геометрическая модель лопатки и сечения, по которым она получена

При этом сечения получают путём изменения геометрии выбранного исходного симметричного профиля в соответствии с газодинамическими и геометрическими требованиями,

предъявляемыми к лопатке. Плоский исходный профиль в свою очередь представлен координатами точек, обмерянных с поверхности его реального прототипа. В ходе преобразования сечения исходного профиля к требуемому виду меняется его длина (хорда), толщина, радиусы входных и выходных кромок, после чего нужным образом изгибается его средняя линия (рис. В.2).

Получение сечения сводится к вычислению новых значений координат точек исходного профиля, полученных после всех геометрических преобразований.

После получения координат всех точек, начинается этап построения сечения в среде какой-либо графической системы. В данной работе построение ведётся в среде Unigraphics NX. По полученным точкам строится сплайн, затем он сопрягается с окружностями входных и выходных кромок. На этом традиционное построение сечения заканчивается. В статье предлагается несколько иной подход к решению поставленной задачи.

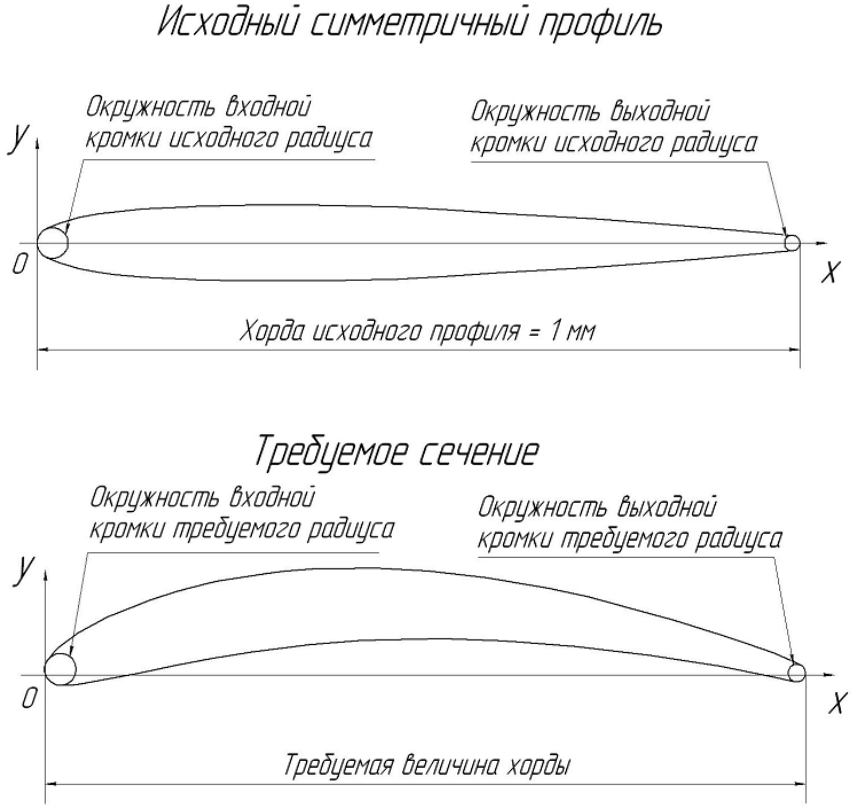

Рис. В.2 Получение требуемого сечения путём изменения геометрии исходного

1. Постановка задачи

Инструментальные погрешности при обмере точек реального исходного профиля приводят к тому, что сплайн, впоследствии проведённый через эти точки, не является гладким и имеет локальные неровности и перегибы, особенно заметные в области сопряжения с кромками на входе и выходе. По этой причине в ходе моделирования лопатки довольно много времени уходит на «выглаживание» сплайна и построение гладких промежуточных кривых, сопрягающих поверхность лопатки с окружностями входной и выходной кромок. Также довольно неудобным является тот факт, что в среде NX невозможно получить аналитическое уравнение сплайна, построенного по заданным точкам.

Нам кажется, что приемлемым решением проблемы является аппроксимация точек исходного симметричного профиля гладкой кривой невысокой степени. При наличии уравнения этой кривой становится возможным получение промежуточных точек исходного профиля в целях повышения точности построения сечения. В данной работе рассмотрен метод аппроксимации с помощью кривой Безье третьего порядка, являющейся достаточно простой для построения, а также обладающей свойствами, необходимыми для простого управления её формой и сопряжением с другими кривыми. Сам процесс аппроксимации сводится к применению методов оптимального поиска, в данном случае метода градиентного спуска, для минимизации функции отклонения кривой от аппроксимируемых точек. Для вычисления отклонений параметрической кривой Безье от точек с заданными координатами используются формулы Кардано. Метод, использованный для аппроксимации, гарантирует отсутствие перегибов и локальных осцилляций кривой, а также гладкость сопряжения с окружностями входной и выходной кромок.

2. Кривые Безье третьего порядка

Как известно, в параметрической форме кубическая кривая Безье описывается следующим уравнением:

$$\mathbf{B}(t) = (1-t)^3\mathbf{P}_0 + 3t(1-t)^2\mathbf{P}_1 + 3t^2(1-t)\mathbf{P}_2 + t^3\mathbf{P}_3, \quad t \in [0,1]$$

Четыре опорные точки P_0, P_1, P_2 и P_3, заданные в 2-х или 3-мерном пространстве определяют форму кривой (рис. 2.1).

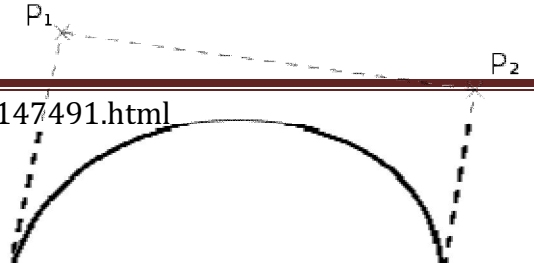

Рис. 2.1. Кубическая кривая Безье.

Линия берёт начало из точки P_0, направляясь к P_1, и заканчивается в точке P_3, подходя к ней со стороны P_2. То есть кривая не проходит через точки P_1 и P_2, они используются для указания её формы и направления. Длина отрезка между P_0 и P_1 определяет, как скоро кривая повернёт к P_3.

Кроме гладкости кривой третьего порядка кривые Безье обладают следующими полезными свойствами.

1. Основа формы кривой повторяет очертания многоугольника.
2. Первая и последняя точки совпадают с соответствующими точками определяющего многоугольника.
3. Векторы касательных к кривой в первой и последней точках по направлению совпадают соответственно с первой и последней сторонами многоугольника.

Теперь вернёмся к нашей задаче. При известном радиусе входной и выходной кромок (соответственно R и r на рис.2.2) и известных углах сопряжения поверхности профиля с ними (α_1 и α_2 на рис. 2.2) можно точно определить координаты лежащих на соответствующих окружностях начальной и конечной точек кривой Безье. С ними совпадут точки P_0 и P_3 определяющего многоугольника (в соответствии со свойством 2). В этих точках кривая будет сопрягаться с окружностями. Но для гладкого сопряжения необходимо достичь G1-непрерывности, т. е. непрерывности по касанию. Таким образом, совпасть должны не только координаты точек на окружностях и кривой Безье, но и направления векторов первой производной в них. Анализируя свойство 3, приходим к выводу, что за направление вектора первой производной в начальной и конечной точках кривой отвечают соседние с ними точки P_1 и P_2. Направление вектора первой производной в точке на окружности нам известно – это направление прямой, касательной к

окружности в этой точке. Значит, для обеспечения равенства направлений векторов опорные точки P_1 и P_2 должны лежать на соответствующих касательных прямых l_1 и l_2 (см. рис. 2.2). Более того, так как поверхность профиля должна быть выпуклой и не иметь перегибов, то в соответствии со свойством 1, определяющий многоугольник также должен быть выпуклым, и эти точки должны лежать выше точек P_0 и P_3.

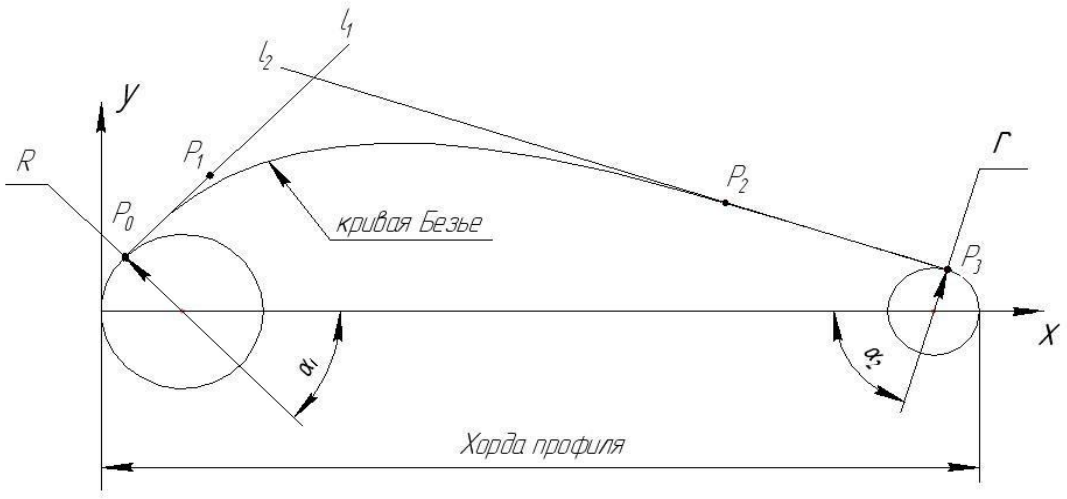

Рис. 2.2. Поверхность исходного профиля.

Итак, на точки наложены все необходимые ограничения для обеспечения нужной формы кривой. Следующий этап – аппроксимация точек исходного профиля. В данной работе рассматривается способ аппроксимации с помощью методов оптимизации.

3. Задача оптимизации

3.1. Составление целевой функции

Как нам уже известно, форму кривой целиком определяют её опорные точки, следовательно, задача получения кривой определённой формы сводится к вычислению координат опорных точек, при которых такая форма будет получена. Если установить зависимость отклонения кривой от аппроксимируемых точек от координат её опорных точек, то с помощью методов оптимизации можно определить такие значения координат опорных точек, при которых отклонение будет иметь минимальную величину.

Целевая функция отклонения кривой от точек имеет вид $E = \frac{1}{2}\sum_{i=1}^{N}(y_i - y_{bi})^2$,

где N – количество исходных точек, y_i - ордината *i*-ой исходной точки, y_{bi} - ордината соответствующей точки на кривой Безье.

Отметим, что только координаты точек P_1 и P_2 доступны для варьирования в целях приближения. Также отметим, что эти точки всегда лежат на прямых, касательных к окружностям в точках сопряжения с кривой. Запишем уравнения этих прямых в параметрической форме:

$\overline{r_1}(t_1) = P_0 \cdot (1-t_1) + P_x \cdot t_1$ для касательной прямой l_1,

$\overline{r_2}(t_2) = P_x \cdot (1-t_2) + P_3 \cdot t_2$ для касательной прямой l_2.

Здесь P_x - точка пересечения прямых l_1 и l_2.

Тогда координаты точек P_1 и P_2 могут быть получены из параметрических уравнений двух прямых как точки, принадлежащие этим прямым. Следовательно, целевую функцию можно представить как функцию от параметров касательных прямых t_1 и t_2: $E = E(t_1, t_2)$.

Подставляя уравнения прямых вместо точек P_1 и P_2 в исходное уравнение кривой Безье третьего порядка, имеем:

$$B(t,t_1,t_2) = (1-t)^3 P_0 + 3t(1-t)^2 \cdot (P_0(1-t_1) + P_x t_1) + 3t^2(1-t) \cdot (P_x(1-t_2) + P_3 t_2) + t^3 P_3 .$$

Тогда с учётом этого выражения целевая функция преобразуется к виду

$$E(t,t_1,t_2) = \frac{1}{2}\sum_{i=1}^{N}(B_y(t,t_1,t_2) - y_i)^2 .$$

Для вычисления значения целевой функции каждой *i*-ой исходной точки определяется такая точка на кривой Безье, которая имеет равную с ней абсциссу, затем вычисляется разность их ординат. За выбор точек на кривой отвечает параметр *t*. Следовательно, для каждого *i* в уравнении целевой функции *t* должно иметь такое значение, при котором выполняется равенство абсцисс. Практически это означает, что необходимо по известному значению $B_x(t,t_1,t_2) = x_i$ определить

соответствующее ему значение *t*, подставив его затем в $B_y(t,t_1,t_2)$. Тогда перед нами встаёт новая проблема: необходимость определения такого *t*, при котором такое равенство выполняется.

Так как для каждого *i* нам известны B_x, то параметр *t* определяется решением кубического уравнения кривой Безье. Для получения аналитического решения уравнения третьей степени воспользуемся формулами Кардано.

Итак, теперь можно однозначно определить значение параметра *t*, которому будет соответствовать точка с координатой x_i. В таком случае из уравнения целевой функции исключается параметр *t*, и оно окончательно принимает вид

$$E(t_1,t_2)=\frac{1}{2}\sum_{i=1}^{N}\left(B_{yi}(t_1,t_2)-y_i\right)^2.$$

3.2. Оптимизация целевой функции

В данной работе для поиска минимума выбран градиентный метод оптимизации с дроблением шага.

Суть градиентного метода состоит в итерационном изменении значений варьируемых параметров функции в антиградиентном направлении для поиска минимума. На каждой итерации значение параметра вычисляется по формуле $\mathbf{X}^{r+1}=\mathbf{X}^r+\lambda^r\mathbf{S}^r$.

Здесь X^r - текущее значение параметра, λ^r - текущее значение шага в направлении S^r,

S^r - вектор направления антиградиента целевой функции в точке X^r.

Итерации повторяются до достижения заданной точности, т. е. достижения целевой функцией определённого значения, либо до тех пор, пока изменение значения функции по отношению к предыдущему значению не будет меньше некоторого значения ε.

Значение шага выбирается по следующему алгоритму.

1. Изначально $\lambda^r=\lambda_0$ и *k*=0.

2. Если $E(t_1^{r+1},t_2^{r+1})<E(t_1,t_2)$ и *k<4*, то *k = k+1.*

3. Если $E(t_1^{r+1},t_2^{r+1})<E(t_1,t_2)$ и *k=4*, то *k = 0,* $\lambda^{r+1}=2\cdot\lambda^r$.

4. Если $E(t_1^{r+1}, t_2^{r+1}) \geq E(t_1, t_2)$, то $k = 0$, $\lambda^{r+1} = \frac{\lambda^r}{2}$.

Метод градиентного спуска является одним из самых быстрых и простых в плане реализации методов оптимального поиска, но также является неустойчивым к наличию в функции локальных экстремумов. При построении графиков целевой функции для каждого из шести типов профилей в пакете MathCAD не было выявлено наличия у неё локальных экстремумов, однако было замечено, что функция имеет «овражный» вид. Тем не менее, значения минимума в «овраге» отличаются весьма несущественно (в пределах двенадцатого знака после запятой) и лежат в области приемлемых величин.

Итого, после достижения оптимизационным алгоритмом заданной величины отклонения, опорные точки располагаются таким образом, что кривая имеет наименьшее достижимое отклонение от точек исходного профиля. Так как опорные точки известны, то из уравнения кривой Безье можно получить любое желаемое количество точек для дальнейшего построения сечения, а сплайн, построенный средствами Unigraphics по полученным точкам, будет иметь гладкость кривой третьей степени. Кроме всего прочего, наличие уравнения кривой упрощает последующий прочностной расчёт.

При практическом применении алгоритму оптимизации требуется в среднем 600 итераций на достижение среднего отклонения от точек, равного 0.001 мм.

4. Создание модуля пользовательского приложения в среде NX

Каждый пользователь, освоив интерактивные средства моделирования, формирования сборочных узлов и оформления конструкторской документации, в процессе решения практических, порой весьма нетривиальных задач сталкивается с потребностью дополнить возможности системы новыми функциями для решения таких задач или автоматизации выполнения повторяющихся процедур. В нашем случае процесс построения сечения можно полностью автоматизировать вместо построения вручную по точкам, полученным из аппроксимации. Сам процесс аппроксимации также можно интегрировать в модуль построения сечения. От пользователя потребуется только задать требуемые параметры сечения.

Для решения подобных задач Unigraphics располагает достаточно развитыми возможностями модуля UG/Open API (Application Program Interface), реализованного на принципах открытой архитектуры и предоставляющего доступ к объектам геометрической модели для программных приложений разработчиков или программ отдельных пользователей. UG/Open даёт возможность программным способом создавать геометрические модели, получать информацию об объектах, формировать сборки, генерировать чертёжную документацию и т. д. Практически все функциональные возможности Unigraphics, доступные пользователю при интерактивном взаимодействии с системой, реализуемы посредством функций API.

UG/Open API (User Function) реализует взаимодействие Unigraphics и программ пользователя, написанных на языке C/C++. Заголовочные файлы (.h) соответствуют требованиям стандарта ANSI C и поддерживают разработку программ с использованием языков C/C++. В зависимости от способа построения программа пользователя может выполняться как внешнее (External) или как внутреннее (Internal) приложение. В первом случае программа запускается средствами операционной системы как независимое приложение или как процесс, порожденный Unigraphics. Так как внешнее приложение не имеет средств графического вывода, ему доступны функции вывода на печатающие устройства и формирования CGI-файла. Во втором случае программа может быть запущена только из текущей сессии Unigraphics. Она загружается в пространство процесса и может быть остановлена только соответствующими командами API, а все результаты работы программы отображаются в графическом окне Unigraphics.

Создание собственного внутреннего приложения задача нетривиальная, и требует определенных усилий. В частности, файл с программным кодом создаётся в среде Microsoft Visual C++ с использованием шаблона для создания модулей Unigraphics NX. Шаблон поставляется вместе с дистрибутивом программы NX и в принципе доступен. Затем в среде Visual C++ создается новый проект, организуются необходимые диалоговые окна, выбирается тип приложения (внутреннее или внешнее) и язык (C/C++). Затем выбирается точка входа в приложение (автоматически при запуске программы, ручной запуск, определённое действие

пользователя) и способ завершения (вместе с Unigraphics, завершение по требованию). Документация по функциям библиотек UG/Open поставляется с дистрибутивом Unigraphics.

В случае если тип приложения – внутренний, точка запуска – вручную, то после компиляции программы получаем файл *.dll. Для запуска приложения из среды Unigraphics необходимо в меню «*Файл –> Выполнить -> Функция пользователя»* выбрать нужный файл.

5. Заключение

Рассмотренный метод аппроксимации кривыми Безье третьей степени показал хорошие результаты при практическом применении. Простота модификации и малое количество вычислений для построения, а также инвариантность аффинных преобразований делают этот тип кривых довольно удобным инструментом при решении задач аппроксимации с наложенными ограничениями на форму кривой. Однако следует обратить внимание на необходимость введения ограничений на значения, принимаемые варьируемыми параметрами в процессе оптимизации, в противном случае кривая может принять неестественную для поверхности профиля форму, хотя условие минимума отклонения при этом будет достигнуто. Точность аппроксимации можно повысить, применяя кривые более высоких степеней и более точные методы оптимального поиска.

Автоматизация процессов построения сечений средствами пользовательских модулей Unigraphics позволяет сократить время моделирования лопаток и повысить эффективность труда инженера-конструктора.

6. Список использованных источников

1. М. Краснов, Ю. Чигишев. Unigraphics для профессионалов. – М: Лори, 2004. - 320с.
2. Роджерс Д., Адамс Дж. Математические основы машинной графики, 2-е издание. – М.: МИР, 2001. – 604 с.
3. База и Генератор Образовательных Ресурсов.