МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ «МИСиС»

КАФЕДРА Инженерной кибернетики

НАПРАВЛЕНИЕ Прикладная математика   ГРУППА БПМ-16-2

КУРСОВАЯ РАБОТА

по курсу **Численные методы**

Тема: Аппроксимация рациональными функциями

Студент Малынковский Олег Витальевич

Руководитель Гопенгауз Владимир Израелевич

Оценка выполнения курсовой работы ____________________________

Москва 2018г.

Оглавление

Введение

Так как обычно вычисление значений аналитически заданной функции происходит с ограниченной точностью (ограниченность разрядной сетки, ошибки округления, всплески, и т.д.), возникает необходимость в повышении точности вычислений. То есть нужна некоторая функция, которая достаточно хорошо аппроксимирует исходную и при этом будет более удобной для расчёта значений. Дробно-рациональную аппроксимацию целесообразно использовать для повышения точности приближения функции f (x), природа которой такова, что она на некоторых участках имеет «всплески». В этом случае удается более точно учитывать особенности поведения функции. Иными словами, функцию представляют в виде дробно-рационального выражения:

$$f(x) = \frac{P_n(x)}{Q_m(x)}$$

, где $P_n(x)$, $Q_m(x)$ – полиномы степеней не меньших n, m ϵ N. Вопрос в том как подобрать такие полиномы.

Проблематика, связанная с классическими конструкциями рациональных аппроксимаций аналитических функций, составляет важное направление на стыке теории приближений, комплексного анализа, вычислительной математики. Развитие этого направления в 18–19 вв. (в основном, в рамках классической теории непрерывных дробей) связано с именами многих выдающихся математиков — от Эйлера, Лагранжа, Гаусса до Чебышева, Маркова, Стилтьеса.

Одним из способов такой аппроксимации стала аппроксимация Паде, названная по имени французского математика Анри Паде. Благодаря развитию вычислительной техники, начиная с 1960-х годов аппроксимации Паде и их обобщения находят новые многочисленные приложения к самым разнообразным опросам физики, механики и других наук.

Аналитический обзор литературы

Аппроксимации Паде — это локально наилучшие рациональные аппроксимации аналитической функции, заданной своим разложением в степенной ряд (или формального степенного ряда).

$$f(z) = \sum_{k=0}^{\infty} c_k z^k \qquad (1)$$

Точнее, аппроксимацией Паде типа (n,m) степенного ряда называется рациональная функция вида

$$[L/M] = \frac{a_0 + \cdots + a_{L-1}z^{L-1} + a_L z^L}{b_0 + \cdots + b_{M-1}z^{M-1} + b_M z^M} \quad , \qquad (2)$$

Разложение которой в ряд Тейлора (с центром в нуле) совпадает с разложение (1) до тех пор, пока это возможно. Более точное определение дадим позже. Отмети, что функция (2) имеет L+1 коэффициентов в числителе и M+1 в знаменателе. Весь набор коэффициентов определяется с точностью до общего множителя, и мы для определённости положим $b_0 = 1$. Такой выбор станет существенной чертой точного определения и мы условимся читать, что в записи (2) он подразумевается.
Теперь мы имеем L+1 свободных параметров в числителе и M в знаменателе формулы (2); всего L+M+1 свободных параметров. Это означает, что в общем случае коэффициенты тейлоровского разложения функции $[L/M]$ при степенях $1, z, z^2, \dots, z^{L+M}$ должны совпадать с соответствующими коэффициентами ряда (1); другими словами, должно выполняться соотношение

$$\sum_{k=0}^{\infty} c_k z^k = \frac{a_0 + \cdots + a_{L-1}z^{L-1} + a_L z^L}{b_0 + \cdots + b_{M-1}z^{M-1} + b_M z^M} + O(z^{L+M+1}) \qquad (3)$$

Пример.

$$f(z) = 1 - \frac{1}{2}z + \frac{1}{3}z^2 + \cdots,$$

$$[1/0] = 1 - \frac{1}{2}z = f(z) + O(z^2),$$

$$[0/1] = \frac{1}{1 + \frac{1}{2}z} = f(z) + O(z^2),$$

$$[1/1] = \frac{1 + \frac{1}{6}z}{1 + \frac{2}{3}z} = f(z) + O(z^3).$$

Умножая (3) на знаменатель дроби, находим, что

$$(b_0 + \cdots + b_{M-1}z^{M-1} + b_M z^M)(c_0 + c_1 z + \cdots) =$$
$$= a_0 + \cdots + a_{L-1}z^{L-1} + a_L z^L + O(z^{L+M+1}) \qquad (4)$$

Сравнивая коэффициенты при $z^{L+1}, z^{L+2}, \dots, z^{L+M}$, получим равенства

$$\begin{gathered} b_M c_{L-M+1} + b_{M-1}c_{L-M+2} + \cdots + b_0 c_{L+1} = 0, \\ b_M c_{L-M+2} + b_{M-1}c_{L-M+3} + \cdots + b_0 c_{L+2} = 0, \\ b_M c_L + b_{M-1}c_{L+1} + \cdots + b_0 c_{L+M} = 0. \end{gathered} \qquad (5)$$

Для полноты мы положим $c_j = 0$ при j < 0. С учётом соглашения $b_0 = 1$ равенства (5) можно переписать в виде системы M линейных уравнений с M неизвестными коэффициентами знаменателя

$$\begin{pmatrix} c_{L-M+1} & c_{L-M+2} & c_{L-M+3} & \cdots & c_L \\ c_{L-M+2} & c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+1} \\ c_{L-M+3} & c_{L-M+4} & c_{L-M+5} & \cdots & c_{L+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_L & c_{L+1} & c_{L+2} & \cdots & c_{L+M-1} \end{pmatrix} \begin{pmatrix} b_M \\ b_{M-1} \\ b_{M-2} \\ \vdots \\ b_1 \end{pmatrix} = - \begin{pmatrix} c_{L+1} \\ c_{L+2} \\ c_{L+3} \\ \vdots \\ c_{L+M} \end{pmatrix} \qquad (6)$$

Отсюда могут быть найдены b_i. Коэффициенты числителя $a_0, a_1, \dots, a_L$ находятся теперь из (4) сравнением коэффициентов при $1, z, z^2, \dots, z^L$:

$$\begin{gathered} a_0 = c_0, \\ a_1 = c_1 + b_1 c_0, \\ a_2 = c_2 + b_1 c_1 + b_2 c_0, \\ \dots \\ a_L = c_L + \sum_{i=1}^{\min(L,M)} b_i c_{L-i} \end{gathered} \qquad (7)$$

Уравнения (6),(7) называются уравнениями Паде; в случае когда система (6) разрешима, они определяют коэффициенты числителя и знаменателя аппроксимации Паде [*L/M*]. Коэффициенты тейлоровского разложения этой функции при $1, z, z^2, \dots, z^{L+M}$ с соответствующими коэффициентами ряда (1). Поскольку исходной точкой наших рассуждений являются коэффициенты ряда $\sum_{k=0}^{\infty} c_k z^k$, то нет необходимости заранее считать эти коэффициенты коэффициентами Тейлора какой-либо функции. Конечно, мы ожидаем, что с помощью аппроксимации Паде ряда $\sum_{k=0}^{\infty} c_k z^k$ можно приблизить функцию f(z), для которой этот ряд является рядом Тейлора, но здесь важно подчеркнуть различие между проблемой сходимости аппроксимации Паде и проблемой их построения. Для решения второй из этих проблем требуются только коэффициенты ряда; способ построения указывают формулы (6),(7).

Каждый степенной ряд имеет круг сходимости |z|<R (при |z|<R ряд сходится при |z|>R расходится). Если $R = \infty$, ряд представляет функцию, аналитическую всюду в комплексной плоскости (такие функции называют целыми); значение функции в любой точке z может быть получено непосредственным суммированием ряда. Если R = 0, то ряд расходится всюду (кроме z=0) и является только формальным. Такой ряд может содержать информацию о функции, но не всегда сразу ясно, как можно использовать эту информацию. В этом случае, когда последовательность аппроксимаций Паде формального степенного ряда сходится к функции $g(z)$ для $z \in \mathcal{D}$, есть основания считать, что $g(z)$ – функция, соответствующая данному ряду. Если данный ряд сходится к функции $f(z)$ в круге |z|<R, $0 < R < \infty$, то последовательность аппроксимаций Паде может сходиться в более широкой области. Часто это даёт практический подход к задачам аналитического продолжения. В этой связи отметим, что метод Вейерштрасса, основанный на переразложении радов, является скорее общим принципом, чем практическим методом. Чтобы продемонстрировать, насколько хорошо срабатывают аппроксимации Паде в естественных ситуациях, рассмотрим следующий пример.
Пример.

$$f(z) = \sqrt{\frac{1+\frac{1}{2}z}{1+2z}} = 1 - \frac{3}{4}z + \frac{39}{32}z^2 - \cdots$$

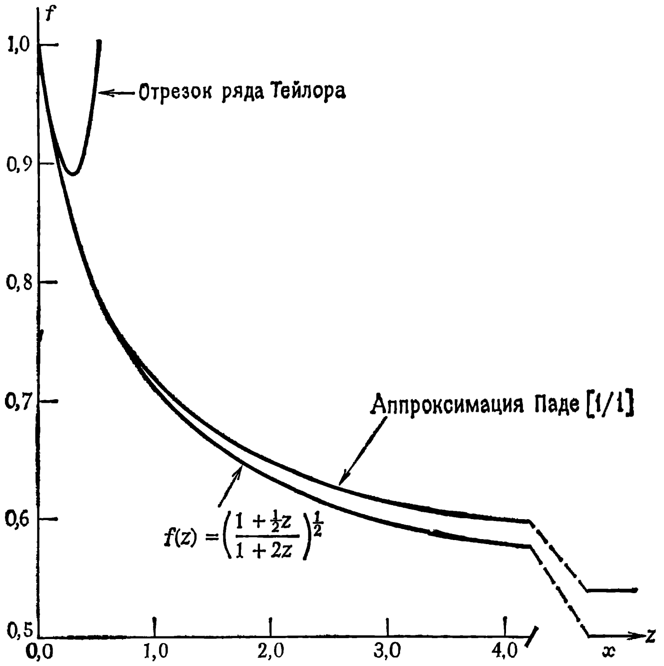


Рис. 1. Графики функций $f(z) = \sqrt{(1+z/2)(1+2z)}$, отрезка ее ряда Тейлора $1 - 3z/4 + 39z^2/32$ и соответствующей аппроксимации Паде [1/1].

Вычислим аппроксимацию Паде [1/1]. Из уравнений (6),(7) находим $b_1 = \frac{13}{8}, a_0 = 1, a_1 = \frac{7}{8}$; На рис. 1 приведены для сравнения графики функции f(z) и [1/1](z) при z>=0. В частности, имеем $f(\infty) = 0.5,\ \ [1/1](\infty) = \frac{7}{13} = 0.54 \dots$, т.е. значение в бесконечности аппроксимация Паде [1/1] даёт с точностью 8%. Этот пример показывает замечательную точность, достигаемую аппроксимацией, которая использует всего три члена разложения.

Следует сразу подчеркнуть одну особенность, связанную, с приближёнными вычислениями аппроксимаций Паде: эти вычисления требуют большей точности, чем можно было бы заранее предположить. Аппроксимации Паде экстраполируют всю последовательность коэффициентов по их конечному числу, поэтому исходные коэффициенты должны быть вычислены достаточно точно.

Из (6) с помощью правила Крамера можно получить явный вид коэффициентов $b_1, b_2, \dots, b_M$ и, следовательно, знаменателя функции (2). Пренебрегая числовым множителем, результат можно записать в виде определителя.

$$Q^{[L/M]}(z) = \begin{vmatrix} c_{L-M+1} & c_{L-M+2} & c_{L-M+3} & \cdots & c_L & c_{L+1} \\ c_{L-M+2} & c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+1} & c_{L+2} \\ c_{L-M+3} & c_{L-M+4} & c_{L-M+5} & \cdots & c_{L+2} & c_{L+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ c_{L-1} & c_L & c_{L+1} & \cdots & c_{L+M-2} & c_{L+M-1} \\ c_L & c_{L+1} & c_{L+2} & \cdots & c_{L+M-1} & c_{L+M} \\ z^M & z^{M-1} & z^{M-2} & \cdots & z & 1 \end{vmatrix} \qquad (8)$$

Обозначение $Q^{[L/M]}(z)$ мы закрепляем за многочленом, который определяется этой формулой. Следует также помнить наше согласие о том, что $c_j = 0$ при $j < 0$. Теперь рассмотрим выражение

$$Q^{[L/M]}(z) \sum_{i=0}^{\infty} c_i z^i =$$

$$= \begin{vmatrix} c_{L-M+1} & c_{L-M+2} & c_{L-M+3} & \cdots & c_L & c_{L+1} \\ c_{L-M+2} & c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+1} & c_{L+2} \\ c_{L-M+3} & c_{L-M+4} & c_{L-M+5} & \cdots & c_{L+2} & c_{L+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ c_{L-1} & c_L & c_{L+1} & \cdots & c_{L+M-2} & c_{L+M-1} \\ c_L & c_{L+1} & c_{L+2} & \cdots & c_{L+M-1} & c_{L+M} \\ \sum_{i=0}^{\infty} c_i z^{M+i} & \sum_{i=0}^{\infty} c_i z^{M+i-1} & \sum_{i=0}^{\infty} c_i z^{M+i-2} & \cdots & \sum_{i=0}^{\infty} c_i z^{i+1} & \sum_{i=0}^{\infty} c_i z^i \end{vmatrix}$$

Вычитая из последней строки первую, умноженную на z^{L+1}, затем вторую умноженную на z^{L+2}, и т.д. вплоть до предпоследней строки, умноженной на

z^{L+M}. Сохраняя только начальные отрезки этих рядов, приходим к определителю:

$$P^{[L/M]}(z) = \begin{vmatrix} c_{L-M+1} & c_{L-M+2} & \cdots & c_{L+1} \\ c_{L-M+2} & c_{L-M+3} & \cdots & c_{L+2} \\ c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+3} \\ \vdots & \vdots & \ddots & \vdots \\ c_{L-1} & c_L & \cdots & c_{L+M-1} \\ c_L & c_{L+1} & \cdots & c_{L+M} \\ \sum_{i=0}^{L-M} c_i z^{M+i} & \sum_{i=0}^{L-M+1} c_i z^{M+i-1} & \cdots & \sum_{i=0}^{L} c_i z^i \end{vmatrix} \qquad (9)$$

Формулы (8),(9) будем использовать дальше в качестве стандартного обозначения.

Теорема 1. Для многочленов, определённых равенствами (8) и (9), справедливо соотношение

$$Q^{[L/M]}(z) \sum_{i=0}^{\infty} c_i z^i - P^{[L/M]}(z) = O(z^{L+M+1}) \qquad (10)$$

Доказательство.

Отметим, что $\deg P^{[L/M]}(z) \le L, \deg Q^{[L/M]}(z) \le M$. Преобразуя левую часть (10), получим

$$Q^{[L/M]}(z) \sum_{i=0}^{\infty} c_i z^i - P^{[L/M]}(z) =$$

$$= \begin{vmatrix} c_{L-M+1} & c_{L-M+2} & \cdots & c_{L+1} \\ c_{L-M+2} & c_{L-M+3} & \cdots & c_{L+2} \\ c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+3} \\ \vdots & \vdots & \ddots & \vdots \\ c_{L-1} & c_L & \cdots & c_{L+M-1} \\ c_L & c_{L+1} & \cdots & c_{L+M} \\ \sum_{i=L+1}^{\infty} c_i z^{M+i} & \sum_{i=L+2}^{\infty} c_i z^{M+i-1} & \cdots & \sum_{i=L+M+1}^{\infty} c_i z^i \end{vmatrix} =$$

$$= \sum_{i=1}^{\infty} z^{L+M+i} \begin{vmatrix} c_{L-M+1} & c_{L-M+2} & \cdots & c_{L+1} \\ c_{L-M+2} & c_{L-M+3} & \cdots & c_{L+2} \\ c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+3} \\ \vdots & \vdots & \ddots & \vdots \\ c_{L-1} & c_L & \cdots & c_{L+M-1} \\ c_L & c_{L+1} & \cdots & c_{L+M} \\ c_{L+i} & c_{L+i+1} & \cdots & c_{L+M+1} \end{vmatrix} \qquad (11)$$

Этим теорема доказана.

Представление остатка в форме (11) иногда бывает полезно при оценках погрешностей, связанных с использованием аппроксимации Паде.

Рассмотрим теперь определитель

$$Q^{[L/M]}(0) = \begin{vmatrix} c_{L-M+1} & c_{L-M+2} & c_{L-M+3} & \cdots & c_L \\ c_{L-M+2} & c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+1} \\ c_{L-M+3} & c_{L-M+4} & c_{L-M+5} & \cdots & c_{L+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_{L-1} & c_L & c_{L+1} & \cdots & c_{L+M-2} \\ c_L & c_{L+1} & c_{L+2} & \cdots & c_{L+M-1} \end{vmatrix}$$

Определители такого вида называются *определителями Ганкеля*. Заметим, что если $Q^{[L/M]}(0) \neq 0$, то система уравнений (6) не вырождения и коэффициенты многочлена $Q^{[L/M]}(z)/Q^{[L/M]}(0)$ представляют её единственное решение. Независимо от этого условие $Q^{[L/M]}(0) \neq 0$ позволяет разделить соотношение (10) на $Q^{[L/M]}(z)$; это даёт

$$\sum_{i=0}^{\infty} c_i z^i - P^{[L/M]}(z) = O(z^{L+M+1})$$

Тем самым доказана наша вторая теорема.

Теорема 2. Если $Q^{[L/M]}(0) \neq 0$, то аппроксимация Паде [L/M] определяется равенством

$$[L/M] = \frac{P^{[L/M]}(z)}{Q^{[L/M]}(z)}, \qquad (12)$$

Где $P^{[L/M]}(z), Q^{[L/M]}(z)$ определены формулами (8),(9).
Набор аппроксимаций записывают в *таблицу Паде*. Часть таблицы Паде функции exp(z) показана в табл.2; эти равенства вытекают из общих формул.

ТАБЛИЦА 1.
ТАБЛИЦА ПАДЕ

M \ L	0	1	2	…
0	[0/0]	[1/0]	[2/0]	…
1	[0/1]	[1/1]	[2/1]	…
2	[0/2]	[1/2]	[2/2]	…
⋮	⋮	⋮	⋮	⋱

ТАБЛИЦА 2.
ЧАСТЬ ТАБЛИЦЫ ПАДЕ ФУНКЦИИ exp (z) [Паде, 1982]

M \ L	0	1	2
0	$\frac{1}{1}$	$\frac{1+z}{1}$	$\frac{2+2z+z^2}{2}$
1	$\frac{1}{1-z}$	$\frac{2+z}{2-z}$	$\frac{6+4z+z^2}{6-2z}$
2	$\frac{2}{2-2z+z^2}$	$\frac{6+2z}{6-4z+z^2}$	$\frac{12+6z+z^2}{12-6z+z^2}$

Определение Бейкера, С-таблица и блочная структура

Чтобы мотивировать обсуждение вопроса о современном определении, анализ которого принадлежит Бейкеру, мы должны понять, какие трудности связаны с основным подходом, обсуждавшимся до сих пор. Классический пример, демонстрирующий недостатки наивного подхода к определению, даёт задача построения аппроксимации Паде [1/1] для функции $1+z^2$. Требуется определить p_0, p_1, q_0, q_1 так, чтобы выполнялось соотношение

$$\frac{p_0+\ p_1 z}{q_0+\ q_1 z} = 1+\ z^2 + O(z^3) \qquad (1)$$

Отсюда вытекает, что $p_0+\ p_1 z=\ q_0+\ q_1 z+\ q_0 z^2 + O(z^3)$, и, следовательно,

$$p_0=\ q_0,\ p_1 = q_1, \text{и } q_0 = 0 \qquad (2)$$

Таким образом аппроксимация [1/1] должна быть равна

$$\frac{0+\ q_1 z}{0+\ q_1 z} = 1 \qquad (3)$$

Однако $1 \neq 1+\ z^2 + O(z^3)$, и мы быстро и справедливо заключаем, что уравнение (1) не имеет решений. В действительности мы получили решение уравнения

$$p_0+\ p_1 z=\ (q_0+\ q_1 z)(1+\ z^2) + O(z^3), \qquad (4)$$

но это не то, что требуется в (1). Отметим, что детерминантные формулы дают также решение (2):

$$P^{[1/1]}(z) = \begin{vmatrix} c_1 & c_2 \\ c_0 z & c_0 + c_1 z \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ z & 1 \end{vmatrix} = -z,$$

$$Q^{[1/1]}(z) = \begin{vmatrix} c_1 & c_2 \\ z & 1 \end{vmatrix} = \begin{vmatrix} 0 & 1 \\ z & 1 \end{vmatrix} = -z.$$

Долгое время уравнение вида (4) принималось в качестве подходящего определения аппроксимации Паде. Точнее, классическое определение Паде или Фробениуса-Паде, состоит в следующем: пусть $p_L(z), q_M(z)$ – многочлены степени L и M соответственно такие, что ($q_M \not\equiv 0$; конечно какждый из многочленов p_L, q_M зависит от пары индексов L, M. Отношение $p_L/\ q_M$ любой пары таких многочленов единственно)

$$q_M(z) f(z) - p_L(z) =\ O(z^{L+M+1}) \qquad (5)$$

тогда $p_L(z)/\ q_M(z)$ называется аппроксимацией Паде функции $f(z)$ (уравнение (5) есть общая форма уравнения (4)). Замечательным является тот факт, что многочлены $p_L(z), q_M(z)$, удовлетворяющие (5) всегда существуют. Однако наш пример показывает, что если $q_M(0) = 0$, то, возможно

$$f(z) \ \neq \frac{p_L(z)}{q_M(z)} + O(z^{L+M+1}).$$

Паде определил индекс дефекта $\omega_{L,M}$ – наименьшее неотрицательное число, для которого

$$f(z) = \frac{p_L(z)}{q_M(z)} + O(z^{L+M-\omega_{L,M}+1}),$$

т.е., $\omega_{L,M}$ – мера недостаточности аппроксимации.

Короче, говоря, рациональная функция $p_L(z)/\ q_M(z)$ может не аппроксимировать функцию $f(z)$ до порядка $L+M$, и при таких обстоятельствах мы предпочитаем говорить, что аппроксимации Паде не существует.

В общей теории рациональной интерполяции хорошо известно, что существуют определённые «выпадающие» значения; например, условие интерполяции значения $f(z_1)$ в узле z_1 может быть не согласовано с другими условиями интерполяции и при таких обстоятельствах считается, что интерполирующей функции не существует. Наш подход полностью согласован с этой точки зрения.

Поскольку требование точности порядка аппроксимации является основным, существенно сохранить его в определении, и поэтому мы пользуемся современным определением, которое было тщательно исследовано Бейкером.

Определние. (Бейкер). Если существуют многочлены $A^{[L/M]}(z), B^{[L/M]}(z)$ степени L и M соответственно такие, что

$$\frac{A^{[L/M]}(z)}{B^{[L/M]}(z)} = f(z) + O(z^{L+M+1}) \qquad (6)$$

и

$$B^{[L/M]}(0) = 1 \qquad (7)$$

то мы по определению полагаем

$$[L/M] = \frac{A^{[L/M]}(z)}{B^{[L/M]}(z)}$$

Форма записи подчёркивает, что и числитель, и знаменатель зависят от каждого из чисел L и M. Замена (6) на соотношение

$$A^{[L/M]}(z) \ - f(z)B^{[L/M]}(z) = \ O(z^{L+M+1})$$

дает полностью эквивалентный вариант определения при условии, что требование (7) сохраняется.

Если $Q^{[L/M]}(0) \neq 0$, то с помощью нового обозначения

$$A^{[L/M]}(z) = \frac{P^{[L/M]}(z)}{Q^{[L/M]}(0)} \quad \text{и} \quad B^{[L/M]}(z) = \frac{Q^{[L/M]}(z)}{Q^{[L/M]}(0)}$$

можно убедиться, что оба рассматриваемых определения совпадают с точностью до несущественного числового множителя. Поэтому вопрос о различии определений иногда считают само собой разумеющимся. Как бы то ни было, случай, когда $Q^{[L/M]}(0) = 0$, требует уточнения терминологии. Обращение $Q^{[L/M]}(0)$ в нуль является очень важным обстоятельством, и поэтому мы вводим для этой величины специальное обозначение.
Определение.

$$C(L/M) = Q^{[L/M]}(0) = \begin{vmatrix} c_{L-M+1} & c_{L-M+2} & \cdots & c_L \\ c_{L-M+2} & c_{L-M+3} & \cdots & c_{L+1} \\ \vdots & \vdots & \ddots & \vdots \\ c_L & c_{L+1} & \cdots & c_{L+M-1} \end{vmatrix} \tag{8}$$

Подведём итог. Если $C(L/M) \neq 0$, то классическое определение Фробениуса-Паде и определение Бейкера эквивалентны. Если $C(L/M) = 0$, то аппроксимации Паде [*L/M*], возможно, не существует. Однако многочлены, удовлетворяющие (5), существуют и определяют рациональную функцию, которая исторически называлась аппроксимацией Паде.

ТАБЛИЦА 1.
C-ТАБЛИЦА

$C(0/0)$	$C(1/0)$	$C(2/0)$	$\cdots$
$C(0/1)$	$C(1/1)$	$C(2/1)$	$\cdots$
$C(0/2)$	$C(1/2)$	$C(2/2)$	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\ddots$

Множество $\{\, C(L/M), L, M = 0,1,2, \dots \}$ удобно изображать в виде таблицы, которую мы будем называть С-таблицей (эту таблицу не следует путать с таблицей Паде).

Пример. Пусть $f(z) = \frac{1 + 2z + z^2 + z^3}{1 + z + z^3}$

Разложение этой функции в ряд Тейлора имеет вид

$$f(z) = 1 + z - z^4 + z^5 - z^6 + 2z^7 - 3z^8 + 4z^9 - \cdots \tag{9}$$

ТАБЛИЦА 2.
C-ТАБЛИЦА ДЛЯ $(1+2z+z^2+z^3)/(1+z+z^2)$

M \ L	0	1	2	3	4	5	6	7	8	9	$\cdots$
0	1	1	1	1	1	1	1	1	1	1	$\cdots$
1	1	1	$\boxed{0}$	$\boxed{0}$	−1	1	−1	2	−3	4	$\cdots$
2	−1	−1	$\boxed{0}$	$\boxed{0}$	−1	$\boxed{0}$	1	−1	−1		$\cdots$
3	−1	−1	1	−1	1	−1	1	−1			$\cdots$
4	1	2	$\boxed{0}$	1	0	0	0	0	0	0	$\cdots$
5	1	4	2	1	0	0	0	0	0	0	$\cdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

Соответствующая C-таблица приведена в табл.2. Наиболее примечательной особенностью таблицы являются квадратные блоки из нулей; нулевой элемент $C(4/4)$ является вершиной бесконечного квадратного блока. Прежде чем доказывать эти утверждения, рассмотрим вопрос о построении табл.1. Чтобы построить табл.2 непосредственно из определения (8) и коэффициентов (9), требуется множество алгебраических операций; покажем как обойтись без этого.

Согласно определению, имеем $C(L/0)=1$, $C(L/1) = c_L$, $C(0/M) = (-1)^{M(M-1)/2}{c^M}_0$. Большинство оставшихся элементов можно найти с помощью тождества

$$C(L/M+1) = \frac{C(L+1/M)C(L-1/M) - C(L/M)^2}{C(L/M-1)} \qquad (10)$$

которое справедливо при условии, что $C(L/M-1) \neq 0$. Равенство (10) называют $\begin{pmatrix} & * & \\ * & * & * \\ & * & \end{pmatrix}$-звёздным тождеством; оно показывает, как связаны элементы C-таблицы, расположение которых схематично изображается звёздочками. Доказательство этого можно получить из теоремы Сильвестра (её мы приводить не будем).

Теорема. Нулевые элементы образуют в С-таблице квадратные блоки, которые окружены со всех сторон ненулевыми элементами.

Доказательство. Левая верхняя вершина блока определяется следующими условиями:

$$C(l/m) = 0, \qquad C(l-1/m) \neq 0, \qquad C(l/m-1) \neq 0$$

(если одно из двух или оба последних условия не выполнены, то l и m переопределяются очевидным образом). Из l-1/m-звёздного тождества

$$C(l-1/m-1)C(l-1/m+1) = C(l-2/m)C(l/m) - C(l-1/m)^2$$

и находим, что

$$C(l-1/m-1)C(l-1/m+1) = -C(l-1/m)^2$$

и, следовательно,

$$C(l-1/m-1) \neq 0, \qquad C(l-1/m+1) \neq 0.$$

Аналогично, из l/m-1-звездного тождества следует, что $C(l+1/m-1) \neq 0$; соответствующая часть С-таблицы показана в табл.3

ТАБЛИЦА 3.
СЛЕДСТВИЯ (4. 12А)

$C(l-1/m+1)$ $\neq 0$	$C(l/m-1)$ $\neq 0$	$C(l+1/m-1)$ $\neq 0$
$C(l-1/m)$ $\neq 0$	$C(l/m)$ $=0$	
$C(l-1/m+1)$ $\neq 0$		

ТАБЛИЦА 4.
ЛЕВАЯ
ГРАНИЦА БЛОКА

$\neq 0$	$\neq 0$	$\neq 0$
$\neq 0$	0	
$\neq 0$	0	
$\neq 0$	0	
$\neq 0$	0	
$\neq 0$	$\neq 0$	$\neq 0$

Предположим, что $C(l/m+1) \neq 0$. Тогда из l/m+1-звездного тождества следует, что $C(l+1/m+1) \neq 0$, и из l+1/m-звездного тождества следует, что $C(l+1/m) \neq 0$.
Таком образом, l/m-блок состоит из одного элемента, и утверждение теоремы для этого случая доказано. Пусть, наоборот $C(l/m+1) = 0$.
Тогда можно найти натуральное k, такое, что

$$C(l/m+j) = 0, \quad j = 0,1,\dots,k-1 \quad \text{и} \quad C(l-1/m+1) \neq 0$$

Используя l-1/m+j-звездное тождество, установим последовательно, что $C(l-1/m+j+1) \neq 0$ при $j = 0,1,\dots,k-1$. Поскольку $C(l-1/m+k) \neq 0$ и $C(l/m+k) \neq 0$, мы заключаем, что $C(l+1/m+k) \neq 0$. Таким образом, мы получили столбец нулей, окаймлённый слева сверху и снизу ненулевыми элементами, как показано в табл.4. Аналогичные рассуждения показывают, что блок является прямоугольником, и если он конечен, то окаймлён со всех сторон ненулевыми элементами; см. табл.5

ТАБЛИЦА 5.
ЧАСТЬ C-ТАБЛИЦЫ, СООТВЕТСТВУЮЩАЯ БЛОКУ

≠0	≠0	≠0	≠0	≠0	≠0
≠0	0	0	0	0	≠0
≠0	0	0	0	0	≠0
≠0	0	0	0	0	≠0
≠0	≠0	≠0	≠0	≠0	≠0

Докажем теперь, что блоки являются квадратами (как табл. 5). Рассмотрим произвольный блок с r строками и s столбцами. Покажем сначала, что r>=s; для этого построим простой пример, который делает общий случай очевидным. Пусть известно, что $C(2/2) = C(3/2) = C(4/2) = 0$; другими словами, имеют место равенства

$$\begin{vmatrix} c_1 & c_2 \\ c_2 & c_3 \end{vmatrix} = \begin{vmatrix} c_2 & c_3 \\ c_3 & c_4 \end{vmatrix} = \begin{vmatrix} c_3 & c_4 \\ c_4 & c_5 \end{vmatrix}$$

Это означает, что некоторая линейная комбинация векторов $\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}, \begin{pmatrix} c_2 \\ c_3 \end{pmatrix}$ обращается в нуль, и то же самое верно для каждой из пар $\begin{pmatrix} c_2 \\ c_3 \end{pmatrix}, \begin{pmatrix} c_3 \\ c_4 \end{pmatrix}$ и $\begin{pmatrix} c_3 \\ c_4 \end{pmatrix}, \begin{pmatrix} c_4 \\ c_5 \end{pmatrix}$. Используя эти линейные комбинации для преобразования определителей, получим

$$\begin{vmatrix} c_0 & c_1 & c_2 \\ c_1 & c_2 & c_3 \\ c_2 & c_3 & c_4 \end{vmatrix} = \begin{vmatrix} * & c_1 & * \\ 0 & c_2 & 0 \\ 0 & c_3 & 0 \end{vmatrix} = 0$$

(звездочки заменяют несущественные элементы) и аналогично

$$\begin{vmatrix} c_1 & c_2 & c_3 \\ c_2 & c_3 & c_4 \\ c_3 & c_4 & c_5 \end{vmatrix} = \begin{vmatrix} c_2 & c_3 & c_4 \\ c_3 & c_4 & c_5 \\ c_4 & c_5 & c_6 \end{vmatrix} = 0$$

Теперь, интерпретируя две предыдущие записи как утверждения о линейной зависимости соответствующих столбцов так же, как и выше, получаем

$$C(2/4) = \begin{vmatrix} 0 & c_0 & c_1 & c_2 \\ c_0 & c_1 & c_2 & c_3 \\ c_1 & c_2 & c_3 & c_4 \\ c_2 & c_3 & c_4 & c_5 \end{vmatrix} = \begin{vmatrix} * & c_0 & c_1 & * \\ 0 & c_1 & c_2 & 0 \\ 0 & c_2 & c_3 & 0 \\ 0 & c_3 & c_4 & 0 \end{vmatrix} = 0$$

Аналогичным образом получаются равенства $C(3/4) = C(3/3)$. После этого очевидно, что если блок *C*-таблицы имеет s столбцов, то число строк в нём не меньше *s*. Другими словами, для блока с r строками и s столбцами имеем *r>=s*.

Для доказательства обратного утверждения *r<=s* воспользуемся теоремой Адамара. Не ограничивая общности, можно считать $c_0 \neq 0$. Рассмотрим *C*-таблицу для функции *g(z) = 1/f(z)*, элементы которой будем обозначать через $C'(*/*)$. Из теоремы Адамара вытекает, что $C(L/M)$ и $C'(M/L)$ обращаются в нуль одновременно. Следовательно, *C*-таблица функции *g(z)* имеет блок из *r'=s* строк и *s'=r* столбцов. Согласно доказанному выше неравенству имеем *r'>=s'*; это означает что *r<=s*, и теорема доказана.

Мы переходим к теореме Паде. Формулировка теоремы была модернизирована Бейкером, но ей содержание осталось по существу без изменений.

Теорема. Элементы [L/M] таблицы Паде, для которых $C(L/M) \neq 0$, единственным образом определяются равенствами

$$[L/M] = \frac{A^{[L/M]}(z)}{B^{[L/M]}(z)} = \frac{P^{[L/M]}(z)}{Q^{[L/M]}(z)}$$

Наоборот, пусть $C(L/M) = 0$ для всех элементов некоторого r x r блока C-таблицы. Этому блоку C-таблицы соответствует (r+1)x(r+1)-блок таблицы Паде, для которого $C(\lambda/\mu) \neq 0, \quad C(\lambda + i/\mu) \neq 0, \quad C(\lambda/\mu + i) \neq 0,$ $i = 1,2,\dots,r, \quad C(\lambda + i/\mu + j) = 0\,,\ i,j = 1,2,\dots,r.$

Аппроксимации Паде для этого блока однозначно определяются условиями

$$[L/M] = [\lambda/\mu], \quad \text{если } L + M \leq \lambda + \mu + r,$$
$$[L/M] \text{ не существует}, \quad \text{если } L + M > \lambda + \mu + r.$$

<u>Доказательство теоремы разобьём на три пункта.</u>

1. Если $C(L/M) = 0$, то уравнения Паде линейно независимы и вместе с условием $b_0 = 1$ однозначно определяют коэффициенты соответствующей аппроксимации. В частности, это замечание исключает возможность того, что многочлены $A^{[L/M]}(z)$ и $B^{[L/M]}(z)$ имеют общие множители.

2. Аппроксимации Паде, соответствующие левой и верхней линиям блока, определяются равенствами

$$[\lambda + i/\mu] = [\lambda/\mu + i] = [\lambda/\mu], i = 1,2, \dots, r.$$

Доказательство основывается на единственности этих аппроксимаций (см. п.1 выше) и блочной структуре С-таблицы. При i=1 нужный рещульатт вытекает из равенств

коэффициент при $z^{\lambda+1}$ в $P^{[\lambda+1/\mu]}(z)$ равен $\pm C(\lambda + 1/\mu + 1) = 0$,
коэффициент при $z^{\mu+1}$ в $P^{[\lambda/\mu+1]}(z)$ равен $\pm C(\lambda + 1/\mu + 1) = 0$.

Далее рассуждаем по индукции; коэффициент при $z^{\lambda+i}$ в $P^{[\lambda+i/\mu]}(z)$ равен $\pm C(\lambda + i/\mu + 1) = 0$; следовательно, $[\lambda + i/\mu] = [\lambda + i - 1/\mu]$, $i = 1,2,..,r$. Аналогичным образом получаются равенства $[\lambda/\mu + i] = [\lambda/\mu + i - 1]$, $i = 1,2,..,r$, и утверждение п.2 доказано.

3.Аппроксимации Паде $[L/M]$, соответствующие блоку С-таблицы, удовлетворяют условиям $[L/M] = [\lambda/\mu]$ при $L + M \leq \lambda + \mu + r$, и $[L/M]$ не существует, если $L + M > \lambda + \mu + r$.

Доказательство. Поскольку аппроксимации Паде $[\lambda + i/\mu]$, $i = 0,1,2,..,r$, соответствующие верхней границе блока, совпадают, то коэффициенты знаменателя Паде удовлетворяют следующим $\mu + r$ уравнениям:

$$\begin{pmatrix} c_{\lambda-\mu+1} & \cdots & c_{\lambda+1} \\ \vdots & & \vdots \\ c_{\lambda} & \cdots & c_{\lambda+\mu} \\ \vdots & & \vdots \\ c_{\lambda+r} & \cdots & c_{\lambda+\mu+r} \end{pmatrix} \begin{pmatrix} q_\mu \\ \vdots \\ q_0 \end{pmatrix} = 0 \qquad (а)$$

и поскольку $[\lambda + r + 1/\mu]$ не лежит в этом блоке, то

$$c_{\lambda+r+1}q_\mu + c_{\lambda+r+2}q_{\mu-1} + \dots + c_{\lambda+r+\mu+1}q_0 \neq 0. \qquad (б)$$

Предположим, что $[L/M]$- некоторая аппроксимация Паде в блоке, не лежащая на верхней и левой границе, т.е. $\lambda + 1 \leq L \leq \lambda + r$, и $\mu + 1 \leq M \leq \mu + r$ и $B^{[L/M]}(0) = 1$. Анализ соответствующих систем уравнений показывает, что в этом случае мы должны иметь равенство $[L/M] = [L - 1/M - 1]$. С учётом утверждения п.2 мы заключаем, что если аппроксимация Паде в блоке существует, то она приводится к $[\lambda/\mu]$, и, таким образом, её знаменатель удовлетворяет системе (а). Таким образом, мы видим, что условия точности порядка аппроксимации выполняются при $L + M \leq \lambda + \mu + r$,так что эти аппроксимации существуют и приводимы. Далее, из (б) вытекает, что условие $L + M > \lambda + \mu + r$ аппроксимации для степени $z^{\lambda+\mu+r+1}$ нарушается и соответствующей аппроксимации Паде не существует. Теперь теорема полностью доказана.

Предполагаемые ошибки

В любой аппроксимационной схеме необходимо иметь возможность оценить погрешность вычислений. При использовании аппроксимации Паде имеются три основных источника ошибок:
(i) коэффициенты c_i известны лишь приближенно;
(ii) точность теряется при вычислении коэффициентов аппроксимаций Паде и их значений;
(iii) сами аппроксимации Паде представляют функцию лишь приближенно, с чем связана погрешность аппроксимации.

В связи с (i) мало что можно сделать. Ограничимся тем знамечанием, что точность вычисления коэффициентов c_i существенна, поскольку аппроксимации Паде сильно реагируют на их малые изменения. Вариации коэффициентов могут дать быструю и очень полезную информацию о точности, с которой найдены аппроксимации Паде.
Потеря точности типа (ii) в практических вычислениях является непростительной. Грубое практическое правило таково: каждый надёжный десятичный знак в ответе требует одного лишнего знака в промежуточных значениях.
Обратимся теперь к проблеме оценки математической точности аппроксимаций. Практически здесь всегда приходится основываться на тех или иных гипотезах, так что говорить приходиться о предполагаемых ошибках. Оценка этих ошибок является наукой в такой же мере наукой как и искусством.
Назовём дефектом комбинацию из полюсов и близко расположенного нуля. Дефекты становятся существенным явлением, когда попадают в рассматриваемую область. В качестве такой области чаще всего удобно выбирать максимальный круг, содержащий начало координат и все особенности функции, которые представляют интерес; другие неполярные особенности удобно оставить за пределами рассматриваемого круга. Тогда дефектом можно считать полюс этой области, вычет которого мал: скажем 0.3% от минимального ожидаемого значения вычета функции. Дефектные аппроксимации Паде мы склонны исключать из рассмотрения. Таким образом, мы ожидаем получить последовательность, которая равномерно ограничена в некоторой области, не содержащей особенностей функции. Если удается получить последовательность, содержащую достаточное количество аппроксимаций, то можно гарантировать сходимость по теореме Монтессу (приводить её мы не станем). В связи с вопрос о наблюдаемых ошибках следует отметить другой важный практический момент. Наличие дефектов часто сопровождается явлением «почти совпадения» нескольких последовательностей аппроксимаций Паде (с $\Delta L = 1$ или $\Delta M = 1$ или $\Delta L = \Delta M = 1$); если дефект не был прямо или косвенно обнаружен, это может привести к заблуждению относительно скорости сходимости. В самом деле, появление дефекта можно понимать, как близость к вырождению, когда

определитель $C(L/M)$ отличен от нуля, но очень мал, другими словами, как близость к появлению блока в таблице Паде. При наличии блока нескольких последовательных аппроксимаций Паде совпадают; при близости к вырождению к вырождению возможно появление нескольких последовательных аппроксимаций Паде, которые очень мало отличаются друг от друга. Все это показывает, как важно проанализировать таблицу Паде в целом, чтобы определить, какие полюсы аппроксимаций Паде обусловлены особенностями функции, а какие являются дефектами, указывающими на ненадёжностью рассматриваемой аппроксимации. В этой связи ясно, что при изолированном вычислении значений аппроксимаций Паде в одной конкретной точке теряется важная информация, которую доставляют структура аппроксимаций Паде и характер из сходимости во всей z-плоскости.

Рассмотрим теперь функцию, которая имеет доминантную особенность вида

$$A(1-\mu z)^{-\gamma} \qquad (1)$$

Предположим, что мы нашли следующие приближенные значения параметров:

$$A'(1-\mu' z)^{-\gamma'} \qquad (2)$$

Три параметра A, μ, γ определяются, вообще говоря, тремя уравнениями из числа тех, что содержатся в условиях точности порядка аппроксимации. Пусть соответствующие равенства справедливы для коэффициентов при степенях z^J, z^{J+1}, z^{J+2}; тогда мы имеем

$$A\binom{-\gamma}{j}\mu' = A'^{\binom{-\gamma'}{j}\mu'^{j}}\left(1+\eta_j\right), j = J, J+1, J+2, \qquad (3)$$

где η_j- малые относительные ошибки. Очевидно, если $\eta_j = 0$, то рассматриваемые значения параметров совпадают, так что η_j должны рассматриваться как источник ошибок в схеме аппроксимации. Мы рассматриваем относительные погрешности, поскольку значение μ определяет масштаб в z-плоскости, и если μ значительно отличается от единицы, то величины коэффициентов быстро меняются вместе с j. При естественных предположениях об аппроксимации мы можем выделить линейную часть погрешностей

$$A' = A + \delta A, \quad \gamma' = \gamma + \delta\gamma, \quad \mu' = \mu + \delta\mu \qquad (4)$$

Из (3) логарифмическим дифференцированием с точностью до членов второго порядка получаем

$$\frac{\delta A}{A} + j\frac{\delta\mu}{\mu} + \delta\gamma\left(\sum_{k=0}^{j-1}\frac{1}{\gamma+k}\right) = \eta_j, \quad j = J, J+1, J+2 \qquad (5)$$

Эти три равенства можно переписать в виде

$$-\frac{\delta\mu}{\mu} = (2\gamma + 2J + 1)\eta_{J+1} - (\gamma + J + 1)\eta_{J+2} - (\gamma + J)\eta_J \quad (6a)$$

$$\delta\gamma = (\gamma + J)(\eta_{J+1} - \eta_J - \frac{\delta\mu}{\mu}) \qquad (6b)$$

$$\frac{\delta A}{A} = \eta_J - J\frac{\delta\mu}{\mu} - \left(\sum_{k=0}^{J-1} \frac{1}{\gamma + k}\right)\delta\gamma \qquad (6c)$$

Из (6a) при условии, что в правой части нет каких-либо необычных сокращений, мы видим, что $\delta\mu/\mu$ – величина порядка $J\eta_J$; из (6b) следует, что $\delta\gamma$- величина порядка $J\ \delta\mu/\mu$; наконец, (6c) показывает, что $\delta A/A$ порядка $J^2 \ln J\eta_J$. Таким образом, мы заключаем, что погрешности оценок параметров находятся между собой в следующих отношениях:

$$\frac{\delta\mu}{\mu} : \delta\gamma : \frac{\delta A}{A} = 1 : J : J \ln J \qquad (7)$$

где *J*- порядок активных членов ряда.

Определение абсолютных значений ошибок сложнее, поэтому сразу без предварительного вывода выпишем

$$A\mu^{L+M}\eta \ \approx \ \frac{C(L/M+1)}{C(L-1/M)} \qquad (8)$$

Основная часть

Качественная постановка задачи

Алгоритм должен аппроксимировать заданную аналитически функцию точнее, чем её разложение в ряд Тейлора.

Математическая постановка задачи

Задана некоторая функция $f(x)$. $f(x) = \sum_{n=0}^{k} \frac{f^{(n)}(a)}{n!}(x-a)^n$ – её разложение в ряд Тейлора.
Нужно получить рациональную функцию (аппроксимацию Паде) вида

$$[L/M](x) = \frac{a_0 + \cdots + a_{L-1}x^{L-1} + a_L x^L}{b_0 + \cdots + b_{M-1}x^{M-1} + b_M x^M}$$

L, M – указывают порядок многочленов дроби.

$$f(x) \approx [L/M](x)$$

Алгоритм решения

1) Необходимо получить разложение функции в ряд Тейлора:
Для нахождения значений производных n-го порядка с точке используем аппарат конечных разностей.

$$\frac{\partial f_i}{\partial x} = \frac{f_{i+1} - f_{i-1}}{2\Delta x}$$

$$\frac{\partial^2 f_i}{\partial x^2} = \frac{f_{i+1} - 2f_i + f_{i-1}}{\Delta x^2}$$

$$\frac{\partial^3 f_i}{\partial x^3} = \frac{f_{i+1} - 3f_i + 3f_{i-1} - f_{i-2}}{\Delta x^3}$$

$$\frac{\partial^4 f_i}{\partial x^4} = \frac{f_{i+2} - 4f_{i+1} + 6f_i - 4f_{i-1} + f_{i-2}}{\Delta x^4}$$

В силу нарастающей погрешности, ограничимся производной 4-го порядка. Так для аппроксимации Паде очень важна исходная точностью коэффициентов, то чем больший порядок мы будем брать для ряда Тейлора, тем хуже будем получаемая аппроксимация Паде. Осталось разделить полученный результат на факториал порядка(если разложение производится в точке x=0, если нет то опять высокая погрешность в силу того, что наша аппроксимация изначально создана для аппроксимации в окрестности нуля)

2) Теперь нужно найти коэффициенты числителя и знаменателя нашей рациональной функции. c_j- коэффициенты ряда Тейлора. Для полноты мы положим $c_j = 0$ при j < 0. С учётом соглашения $b_0 = 1$ можно записать систему M линейных уравнений с M неизвестными коэффициентами знаменателя

$$\begin{pmatrix} c_{L-M+1} & c_{L-M+2} & c_{L-M+3} & \cdots & c_L \\ c_{L-M+2} & c_{L-M+3} & c_{L-M+4} & \cdots & c_{L+1} \\ c_{L-M+3} & c_{L-M+4} & c_{L-M+5} & \cdots & c_{L+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_L & c_{L+1} & c_{L+2} & \cdots & c_{L+M-1} \end{pmatrix} \begin{pmatrix} b_M \\ b_{M-1} \\ b_{M-2} \\ \vdots \\ b_1 \end{pmatrix} = - \begin{pmatrix} c_{L+1} \\ c_{L+2} \\ c_{L+3} \\ \vdots \\ c_{L+M} \end{pmatrix}$$

Решение её получим модифицированным методом Гаусса с поиском максимального по модулю ведущего элемента шага. Коэффициенты числителя $a_0, a_1, \dots, a_L$

$$a_0 = c_0,$$
$$a_1 = c_1 + b_1 c_0 \,,$$
$$\dots$$
$$a_L = c_L + \sum_{i=1}^{\min(L,M)} b_i c_{L-i}$$

Листинг кода

```
    public class Polynomial
    {
        private List<double> _array;

        // Инициализирует новый полином заданной строкой коэффициентов
        public Polynomial(IEnumerable<double> arrayCoefficients) {}

        // Инициализирует новый полином по старшей степени при этом все коэффициенты
будут равны 0
        public Polynomial(int headPow)        {}

        // Доступ к коэффициенту полинома по индексу
        double this[int i]        {}
        // Старшая степень полинома
        public int HeadPow        {}

        // Строка коэффициентов
        public IEnumerable<double> LineCoefficients        {}

        // Выполняет сложение двух полиномов
        public static Polynomial operator +(Polynomial poly1, Polynomial poly2)
        {
            Polynomial maxPoly = maxPowPoly(poly1, poly2);
            Polynomial minPoly = minPowPoly(poly1, poly2);
            Polynomial resultPoly = new Polynomial(maxPoly.LineCoefficients);
            for (int i = 0; i < minPoly.HeadPow; i++)
                resultPoly[maxPoly.HeadPow - 1 - i] = maxPoly[maxPoly.HeadPow - 1 -
i] + minPoly[minPoly.HeadPow - 1 - i];
            return resultPoly;
        }

        // Умножает полином на константу
        public static Polynomial operator *(Polynomial poly1, double k)        {}

        // Перемножает между собой два полинома
        public static Polynomial operator *(Polynomial poly1, Polynomial poly2)
        {
            Polynomial resultPoly = new Polynomial(poly1.HeadPow + poly2.HeadPow -
1);
            resultPoly._array.Select(x => 0);
            for (int i = 0; i < poly1.HeadPow; i++)
                for (int j = 0; j < poly2.HeadPow; j++)
                    resultPoly[i + j] += poly1[i] * poly2[j];
            return resultPoly;
        }

        //Возводит полином в степень
        public Polynomial Power(int p)
        {
            Polynomial result = this;
            for (int i =2; i <= p; i++)
            {
                result = result * this;
            }
            return result;
        }

        // Преобразует полином в строку
        public override string ToString()        {}
```

```
        // Решает полином подставляя x
        public double GetSolution(double x)        {}

        // Полином большего порядка из двух
        private static Func<Polynomial, Polynomial, Polynomial> maxPowPoly = (x, y)
=>
            Math.Max(x.HeadPow, y.HeadPow) == x.HeadPow ? x : y;
        // Полином меньшего порядка из двух
        private static Func<Polynomial, Polynomial, Polynomial> minPowPoly = (x, y)
=>
            maxPowPoly(x, y).Equals(x) ? y : x;
    }

    class Program
    {
        static int Factorial(int n)        {}// Находит факториал числа
        static double Function(double x)
        {
            return Math.Sqrt((1 + 0.5 * x) / (1 + 2 * x));
        }
        //нахождения производных функции n-го порядка для нахождения коэф-то ряда
Тейлора
        static double Derivatives(double x, int k)
        {
            double res = 0;
            double h = 0.005;
            if (k == 1)
            {
                res = (Function(x + h) - Function(x - h)) / (2 * h);
                return res;
            }
            else
            {
                res = (Derivatives(x + h, k - 1) - Derivatives(x - h, k - 1)) / (2 *
h);
                return res;
            }

        }

        static double[] SLAU(double[,] A, double[] B)
        {
            int size = B.Length;
            int[] Memory = new int[size];
            for (int i = 0; i < size; i++)
                Memory[i] = i;//В массиве Memory будем хранить какой столбец какому
корню соотвествуеет, что понадибиться
            //при перестовках столбцов в момента поиска максимального элемента в
текущей нетреугольной части матрицы
            double[] X = new double[size];
            for (int i = 0; i < size; i++)
                X[i] = 0;
            double t = 0;
            bool q = false;
            for (int k = 0; k < size - 1; k++)
            {
                //поиск наибольшего элмента в оставшейся нетреугольной части матрицы
                double max = A[k, k];
                int index_i = k, index_j = k;
                for (int r = k; r < size; r++)
                {
```

```
        for (int w = k; w < size; w++)
        {
            if (A[r, w] > max)
            {
                max = A[r, w];
                index_i = r;
                index_j = w;
            }
        }
    }
    if (max < 1e-15)
    {
        Console.WriteLine("Однозначного решения нет");
        q = true;
        break;
    }
    //обмен строк и столбцов если найден такой элемент
    double temp = 0;
    if ((index_i != k) || (index_j != k))
    {
        //обмен строк
        if (index_i != k)
        {
            for (int j = k; j < size; j++)
            {
                temp = A[k, j];
                A[k, j] = A[index_i, j];
                A[index_i, j] = temp;
            }
            temp = B[k];
            B[k] = B[index_i];
            B[index_i] = temp;
        }

        //обмен столбцов
        if (index_j != k)
        {
            for (int i = 0; i < size; i++)
            {
                temp = A[i, k];
                A[i, k] = A[i, index_j];
                A[i, index_j] = temp;
            }
            int u = Memory[k];
            Memory[k] = Memory[index_j];
            Memory[index_j] = u;
        }

    }
    //прямой ход метода
    for (int i = k + 1; i < size; i++)
    {
        t = A[i, k] / A[k, k];
        B[i] = B[i] - t * B[k];
        for (int j = k; j < size; j++)
            A[i, j] = A[i, j] - t * A[k, j];
    }
}
X[size - 1] = B[size - 1] / A[size - 1, size - 1];
double summa = 0;
for (int k = size - 2; k >= 0; k--)
{
```

```
                for (int r = k + 1; r < size; r++)
                    summa += A[k, r] * X[r];
                //обратный ход
                X[k] = (B[k] - summa) / A[k, k];
                summa = 0;
            }
            //Расставляем корни по местам
            if (!q)
            {
                double[] Ans = new double[size];
                for (int i = 0; i < size; i++)
                    Ans[Memory[i]] = X[i];
                return Ans;
            }
            else
                return new double[] { 0 };
        }

        static void Main(string[] args)
        {
            int L = 1;            int M = 1;            double center = 0.0;
            double[] Coefficients = new double[L + M + 1];
            Polynomial Taylor = new Polynomial(new double[] { Function(center) });
            Polynomial temp = new Polynomial(new double[] { 1, -center });
            for (int i = 1; i < L + M + 1; i++)
                Taylor += temp.Power(i) * Derivatives(center, i) * (1.0 /
Factorial(i));
            Coefficients = Taylor.LineCoefficients.ToArray();
            Array.Reverse(Coefficients);
            double[,] A = new double[M, M];            double[] B = new double[M];
            int index;
            for (int i = 0; i < M; i++)
            {
                for (int j = 0; j <= i; j++)
                {
                    if ((L - M + 1 + i + j) >= 0)
                        A[i, j] = Coefficients[L - M + 1 + i + j];
                    else
                        A[i, j] = 0;
                    A[j, i] = A[i, j];
                }
                if ((L + 1 + i) >= 0)
                    B[i] = -Coefficients[L + 1 + i];
                else
                    B[i] = 0;
            }
            double[] X = SLAU(A, B);
            double[] Denumerator = new double[M + 1];
            for (int i = 0; i < M; i++)
                Denumerator[i] = X[i];
            Denumerator[M] = 1;
            Array.Reverse(Denumerator);
            double[] Numerator = new double[L + 1];
            Numerator[0] = Coefficients[0];
            for (int i = 1; i < L+1; i++)
            {
                Numerator[i] = Coefficients[i];
                for (int j = 1; j <= Math.Min(L, M); j++)
                    if ((i - j)>= 0)
                        Numerator[i] += Denumerator[j] * Coefficients[i - j];
```

```
            }
            Array.Reverse(Numerator);
            Array.Reverse(Denumerator);
            Polynomial Num = new Polynomial(Numerator);
            Polynomial Denum = new Polynomial(Denumerator);
            Console.WriteLine(Num);
            Console.WriteLine("---------------------------------");
            Console.WriteLine(Denum);
            Console.WriteLine("Comparing table:");
            Console.WriteLine("Analytic function      Taylor series      Pade 
approximation");
            for (double i = 0; i < 10.0; i += 0.2)
            {
                Console.Write("{0:N7}              ",Function(i));
                Console.Write("{0:N7}              ",Taylor.GetSolution(i));
                Console.Write("{0:N7}              ",Num.GetSolution(i) / 
Denum.GetSolution(i));
                Console.WriteLine();
            }
```

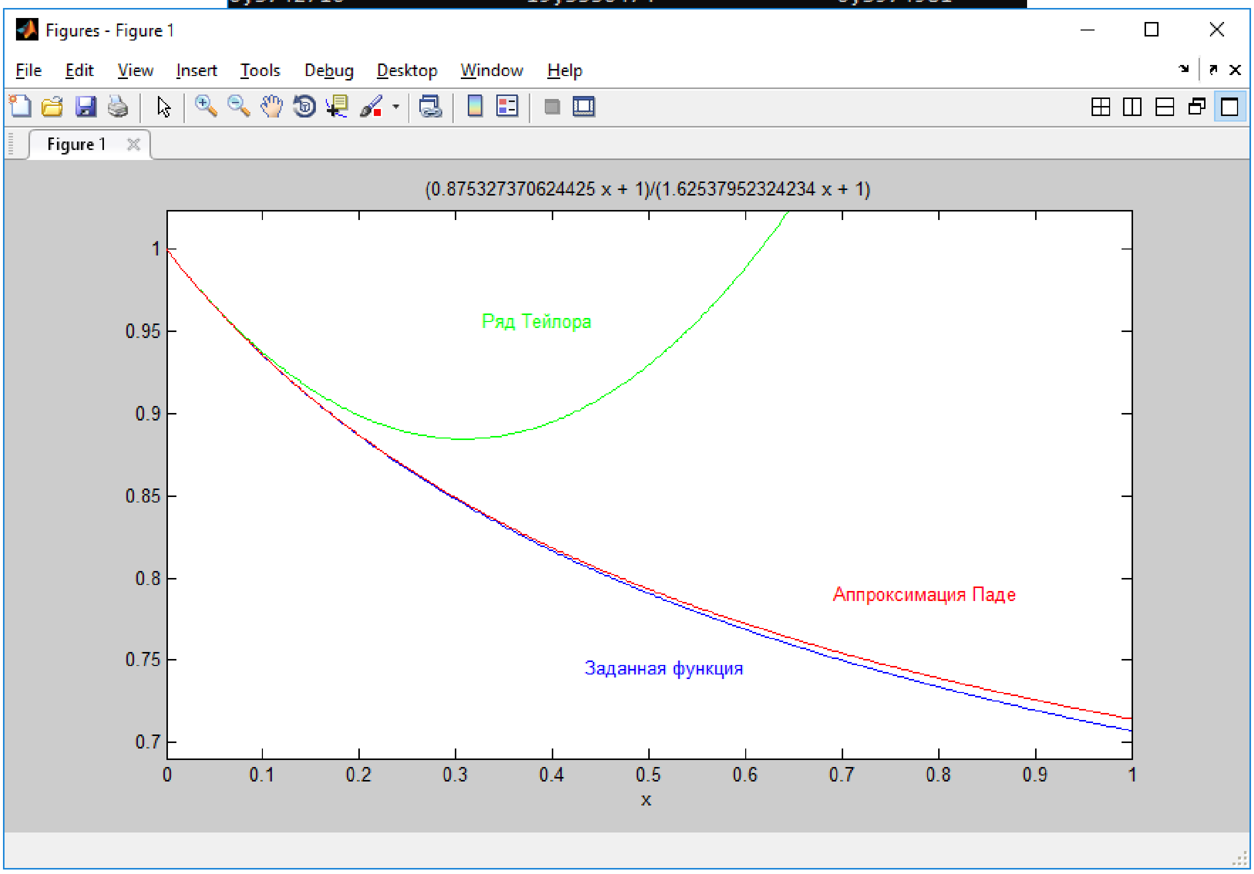
Результат работы программы

$$f(x) = \sqrt{\frac{1 + \frac{1}{2}x}{1 + 2x}}$$

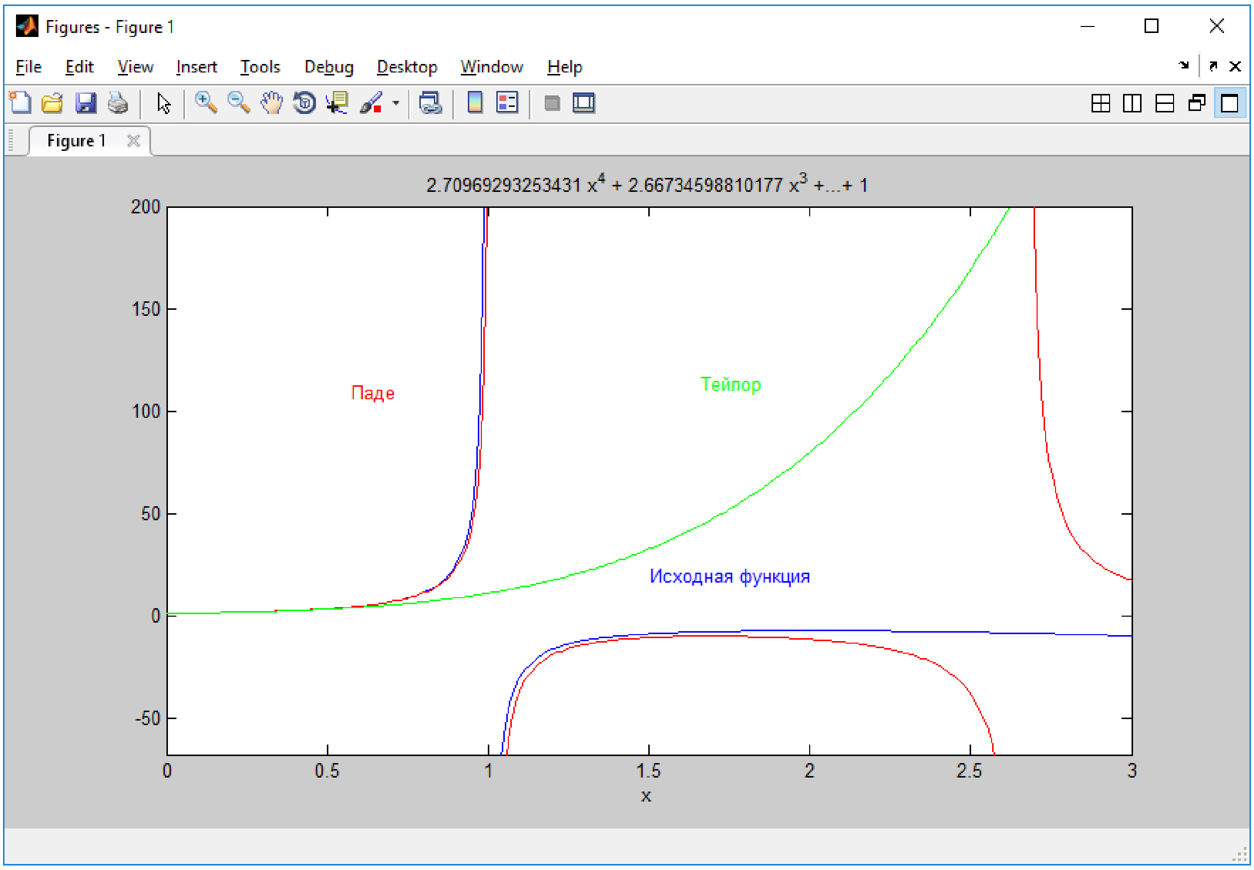
```
C:\WINDOWS\system32\cmd.exe
Taylor series:
1,21911941022901*x^2 + -0,750052152617919*x + 1

0,875327370624425*x + 1
---------------------------------------------------------------
1,62537952324234*x + 1

Comparing table:
Analytic function       Taylor series       Pade approximation
1,0000000               1,0000000            1,0000000
0,8864053               0,8987543            0,8867911
0,8164966               0,8950382            0,8181859
0,7687061               0,9888517            0,7721623
0,7337994               1,1801947            0,7391467
0,7071068               1,4690673            0,7143072
0,6859943               1,8554694            0,6949411
0,6688561               2,3394010            0,6794190
0,6546537               2,9208622            0,6666997
0,6426846               3,5998530            0,6560869
0,6324555               4,3763733            0,6470973
0,6236096               5,2504232            0,6393850
0,6158818               6,2220026            0,6326958
0,6090712               7,2911116            0,6268388
0,6030227               8,4577501            0,6216678
0,5976143               9,7219182            0,6170689
0,5927490               11,0836159            0,6129521
0,5883484               12,5428431            0,6092455
0,5843487               14,0995998            0,6058906
0,5806973               15,7538861            0,6028397
0,5773503               17,5057020            0,6000531
0,5742710               19,3550474            0,5974981
```



$$f(x) = \frac{\exp(x)}{(1-x)}$$



C:\WINDOWS\system32\cmd.exe

```
Taylor series:
2,70969293253431*x^4 + 2,66734598810177*x^3 + 2,50027086051652*x^2 + 2,0000666683646*x + 1

0,144140822696211*x^2 + 0,636682088773976*x + 1
--------------------------------------------------------------
0,370730015981174*x^2 + -1,36338457959062*x + 1

Comparing table:
Analytic function       Taylor series      Pade approximation
1,0000000               1,0000000               1,0000000
1,5267534               1,5256984               1,5267784
2,4863745               2,4401483               2,4860456
4,5552970               4,0274604               4,5458282
11,1277046               6,6757981               10,9279527
```

Видно, что аппроксимация Паде даёт значительно лучшие результаты по сравнению с разложением Тейлора. Во втором примере, у нас была особенность в единице, но полученная рациональная функция прекрасно смогла отобразить это.

Вывод

В ходе выполнения курсовой работы был изучен подход к аппроксимации функции рациональными функциями, в частности аппроксимация Паде. Несмотря на свою идейную простоту (понадобилось только с достаточной точностью разложить функцию в ряд Тейлора) в переразложении исходной функции в отношение двух полиномов, аппроксимация Паде даёт отличный результат в задаче экстраполирования. Также эта аппроксимация учитывает особенности функции (например, полюса) и передаёт их (графический пример этого был приведён выше), также строя таблицы Паде и c-таблицы можно узнать дополнительную информацию о функции.

Список литературы

1) Бейкер Дж., мл., Грейвс-Моррис П. Аппроксимации Паде. Пер. с англ. – М.: Мир, 1986. – 502с., ил.

2) Вержбицкий В.М Основы численных методов: Учебник для вузов – М: Высш. шк., 2005. – 840с.: ил.

3) А. А. Гончар, Рациональные аппроксимации аналитических функций, Совр. пробл. матем., 2003, выпуск 1, 83–106

4) Спецификация языка C#. Стандарт ECMA-334, 5-й выпуск