ISSN 2079-3316 **ПРОГРАММНЫЕ СИСТЕМЫ: ТЕОРИЯ И ПРИЛОЖЕНИЯ** **т. 9**, № 4(39), с. 69–92

УДК 519.652.3

С. В. Знаменский

Численная оценка точности интерполяции несложных элементарных функций

Аннотация. Сравнение точности восстановления элементарных функций по значениям в узлах проведено для алгоритмов интерполяции низкой степени. Результаты тестирования представлены в графическом виде, наглядно демонстрирующем преимущества и недостатки широко используемых кубических интерполяционных сплайнов.

Сравнение выявило, что вопреки распространённому мнению гладкость интерполянт не связана непосредственно с точностью аппроксимации. На рассмотренных 20 разнородных примерах кусочно-квадратичная интерполяция редко и ненамного уступает в точности используемым классическим кубическим сплайнам, зачастую на порядки превосходя в точности многие из них.

В нескольких примерах высокая погрешность интерполяции несложных элементарных функций по фиксированной сетке оказалась практически независимой от степени алгоритма и гладкости интерполянты, в одном из них точнее всех неожиданно оказалась кусочно-линейная интерполяция.

Поставлена задача поиска локального алгоритма интерполяции, точно восстанавливающего рациональные функции второго порядка.

Ключевые слова и фразы: локальная интерполяция, рациональная интерполяция, интерполяция сплайнами, точность восстановления.

Введение

Задача восстановления кусочно-аналитической на отрезке $[a, b]$ вещественной функции $f(x)$ по вектору $\vec{y} = (y_0, \ldots, y_n)$ её $n + 1$ значений $y_i = f(x_i)$ в точках $x_0 < \cdots < x_n, \ldots, n$ вещественной прямой имеет многотысячелетнюю историю [1] и разнообразные обобщения и применения. В ходе разноплановых исследований выросло необозримое в короткой статье множество убедительно стройных теорий интерполяции (см., например, [2–6]), ориентированных на оптимальные алгоритмы для многочисленных приложений.

© С. В. Знаменский, 2018
© Институт программных систем имени А. К. Айламазяна РАН, 2018
© Программные системы: теория и приложения (дизайн), 2018

10.25209/2079-3316-2018-9-4-69-92

Сосредоточимся на доступных с исходным кодом алгоритмах интерполяции, позволяющих восстанавливать простую зависимости по немногим заданным значениям. Такой контекст естественно возникает при анализе труднодоступных экспериментальных данных. Кроме того, выявление слабых мест алгоритмов с этой теряющей популярность точки зрения может привести к появлению новых задач, решение которых могло бы повысить качество интерполяции в других приложениях.

Дальнейшее изложение ориентируется на два практически значимых критерия качества интерполяции:

(1) *Локальность* — зависимость интерполянты только от значений в ближайших l узлах справа и l узлах слева, l—радиус локальности.

(2) *Точность восстановления наиболее простых и употребительных функций.*

Локальность позволяет избежать граничных эффектов связанных с неудачно выбранными граничными условиями, обеспечивает линейную оценку вычислительной сложности и простоту распараллеливания длинной сетки узлов. При обработке изображений локальность обеспечивает чёткость масштабирования фотографий, при поточной обработке сигнала радиус локальности определяет время задержки, а при восстановлении по значениям на сетке кусочно-аналитических функций одной переменной или функций с особыми точками вблизи отрезка интерполяции локализует потери точности восстановления вблизи особых точек.

Точность восстановления наиболее простых функций это объективный и проверяемый признак качества интерполяции. Однако и этот признак таит подводный камень. Колмогоровская сложность очевидно зависит от языка программирования, и нет убедительно однозначного ответа на вопросы, что сложнее:

— Многочлен либо дробно-линейная функция?

— Косинус либо плотность нормального распределения вероятности?

Наложив один график на другой и выделив точки пересечения графиков, нетрудно получить исходные данные для задачи интерполяции, которая имеет два резко различающихся безупречно точных решения.

Мы видим, что, требуя от интерполирующего алгоритма качественного восстановления простой в интуитивном понимании функции, необходимо учитывать подобные двусмысленные постановки задачи интерполяции.

Для выявления влияния и гладкости интерполянты рассмотрим простую формулу локального квадратичного непрерывного (не обязательно

гладкого) интерполирующего сплайна и оценим точность интерполяции на примерах простых элементарных функций, анализируя конкретные ситуации, возникающие при сравнении.

1. Формула кусочно-квадратичной интерполяции

Формулы кусочно-квадратичной интерполяции между узлами с заданными значениями различаются способом получения коэффициента при старшей степени k_i в интерполяционной формуле Ньютона $P_i(x) = y_i + (x - x_i)(m_i + k_i(x - x_{i+1}))$, где $m_i = \frac{(y_{i+1}-y_i)}{(x_{i+1}-x_i)}$ - первая разделённая разность. Этот коэффициент обычно принимается равным второй разделённой разности $\Delta_i = \frac{m_{i+1}-m_i}{x_{i+2}-x_i}$, но более точно может быть получен из значения интерполянты $y_m i = y_i + l_i(m_i + k_i l_i)$ в середине отрезка $[x_i, x_i + 1]$, где $l_i = \frac{x_{i+1}-x_i}{2}$ — половина длины отрезка. Используя для нахождения этого значения интерполяционную формулу Лагранжа по четырём ближайшим узлам, получаем простую кусочно-квадратичную формулу интерполяции:

$$f(x) = y_i + (x - x_i)\left(m_i + k_i(x - x_{i+1})\right), \tag{1}$$

где

$$k_i = \begin{cases} \Delta_0 + \dfrac{x_2 - l_0}{x_3 - x_0}(\Delta_1 - \Delta_0), & i = 0, \\ \Delta_i + \dfrac{l_i - x_{i+2}}{x_{i+2} - x_{i-1}}(\Delta_i - \Delta_{i-1}), & 0 < i < n-1, \\ \Delta_{n-2} + \dfrac{l_{n-1} - x_{n-2}}{x_n - x_{n-3}}(\Delta_{n-3} - \Delta_{n-2}) & i = n-1. \end{cases}$$

Хотя точных ссылок найти не удалось, за тысячелетнюю истории квадратичной интерполяции эти простые формулы несомненно уже в каком-нибудь виде использовались.

2. Сохранение формы при квадратичной интерполяции

Квадратичность интерполянты обеспечивает легко проверяемые точные условия сохранения формы.

Замечание 1. Метод не всегда сохраняет выпуклость, но сохранение выпуклости легко обеспечить, положив $k_i = 0$ для тех $0 < i < n-1$, для которых $\Delta_i \Delta_{i+1} \geq 0$, но вычисленное k_i имеет знак, противоположный знаку $\Delta_i + \Delta_{i+1}$.

Замечание 2. Несохранение монотонности или знака выявляется точным условием $|2A_i^0| \leqslant \left|\frac{m_i}{x_{i+1}-x_i}\right|$. Сохранение знака и монотонности при необходимости может быть легко гарантировано заменой там, где это условие оказывается нарушено, вычисленного значения A_i^0 на взятое с тем же знаком значение $\frac{\pm m_i}{2(x_{i+1}-x_i)}$.

Замечание 3. Сохранение верхзней и нижней границ внутри сегмента образа $[a, b]$ контролируется точным условием

$$-\left(\frac{\sqrt{b-y_i}+\sqrt{b-y_{i+1}}}{x_{i+1}-x_i}\right)^2 \leqslant 2A_i^0 \leqslant \left(\frac{\sqrt{y_i-a}+\sqrt{y_{i+1}-a}}{x_{i+1}-x_i}\right)^2$$

и обеспечивается заменой (там, где это условие оказывается нарушено) вычисленного значения A_i^0 на значение ближайшего ограничения.

Хотя эти простые ограничения сохранения формы являются точными, а соответствующие им дополнения кода несложными, они не включены в испытания этой работы. В рассмотренной постановке задачи они редко что-то меняют, а меняя обычно снижают точность восстановления.

3. Численная оценка качества интерполяции

Численного сравнения результатов интерполяции элементарных функций проводилось по максимальной, средней и средне-квадратичной погрешности.

Интегралы в стандартных формулах для среднего и среднего квадратичного вычислялись по формуле трапеций с 12 равноотстоящими промежуточными узлами на каждом сегменте интерполяции и по этим же значениям строились графики.

Каждый рисунок состоит из двух. На верхнем графики функции и её интерполянт, на нижнем — графики ошибок интерполяции в масштабе, позволяющем их увидеть и сопоставить. Программа, рассчитывающая отклонения и рисующая графики в LaTeX 2_ε TikZ/PGF, реализована на Perl и приложена к статье.

Для сравнения с описанным квадратичным сплайном были использованы все наиболее доступные реализации кубических сплайнов. В сравнение включены практически используемые алгоритмы, реализованные в активно развивающихся библиотеках: открытой GSL (GNU Scientific Library) и многоязычной отечественной AlgLib, на которых

базируется разнообразное современное программное обеспечение. Из них лишь сплайн Стеффена [**9**] (далее S) и негладкий кубический сплайн, см., например, [**10**] (далее C) имеет тот же радиус локальности $l = 2$, что и тестируемый квадратичный Q, и только кусочно линейная интерполяция L имеет меньший радиус локальности.

Для сравнения с результатами для нелокального дважды гладкого кубического сплайна (использовались два типа граничных значений для непериодических задач интерполяции — параболический P, точно восстанавливающий квадратичные полиномы, и естественный N, с нулевыми значениями производной на концах).

Реализованный в GSL алгоритм Х.Акимы (A) [**8**] также включён в сравнение. Другие локальные сплайны, использующие данные из более чем двух соседних узлов с каждой стороны как например [**7**, **11**–**13**], не рассматриваются, поскольку ослабление локальности до $l = 3$ по итогам тестирования в [**14**] ожидаемо допускает гораздо более высокую точность интерполяции аналитических функций.

3.1. Эксперименты на равномерной сетке

Целью экспериментов на равномерной сетке интерполяционных узлов было сравнение поведения различных сплайнов радиуса локальности $l = 2$ на функциях различного поведения, от простого, без локальных экстремумов и перегибов, и кончая более сложным, с чередованием участков убывания-возрастания, а затем и с чередованием направлений выпуклости

3.1.1. *Функции без точек перегиба*

Погрешности интерполяции экспоненты видны на рис. 1, логарифма на рис. 2, простейшей рациональной функции $y = 1/x$ на рис. 3. По трём примерам видно, что при отсутствии точек локальных экстремумов и точек перегиба погрешность P и Q почти совпадают и заметно ниже, чем у остальных сплайнов. В точности та же картина повторяется и с функцией, имеющей график полуокружности на рис. 4, кроме крайних участков где все интерполяционные формулы срабатывают грубо из-за бесконечных производных приближаемой функции в крайних узлах.

Здесь важно отметить, что дважды гладкий кубический сплайн вероятно смог бы продемонстрировать лучшие результаты при удачном выборе краевых условий. Вряд ли возможно это сделать, не зная самой функции. В текущей версии GSL вторая производная на концах задана нулём (N), но гораздо лучший результат P (например, в AlgLib)

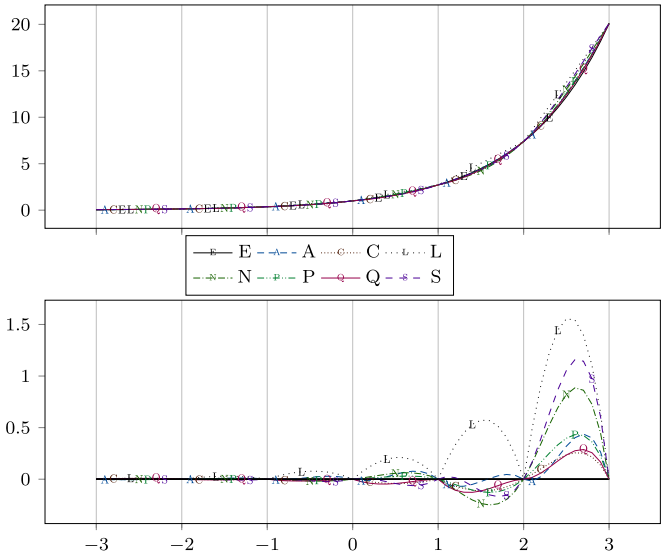

Рисунок 1. Интерполяция e^x на $[-3, 3]$

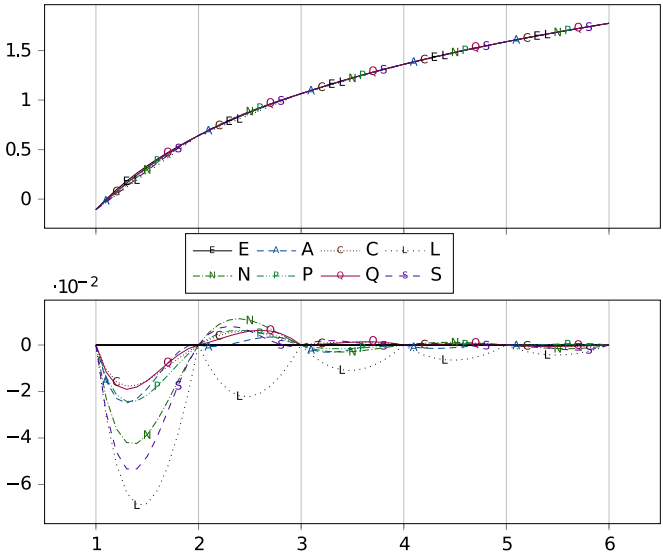

Рисунок 2. Интерполяция $\ln(x - 0.1)$ на $[1, 6]$

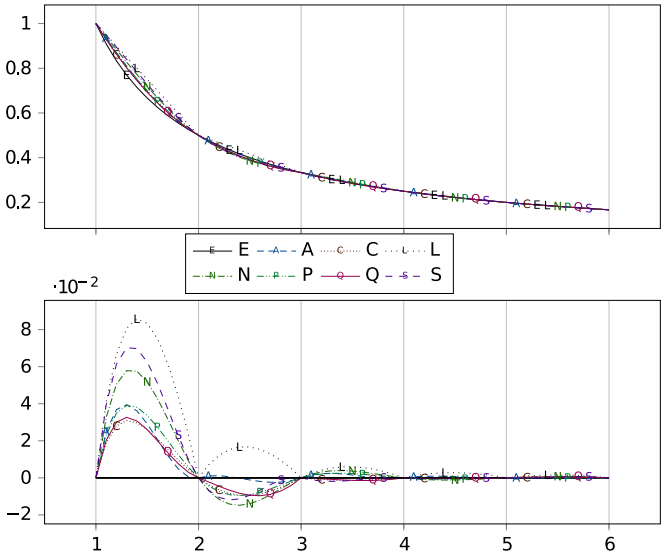

Рисунок 3. Интерполяция $1/x$ на $[1, 6]$

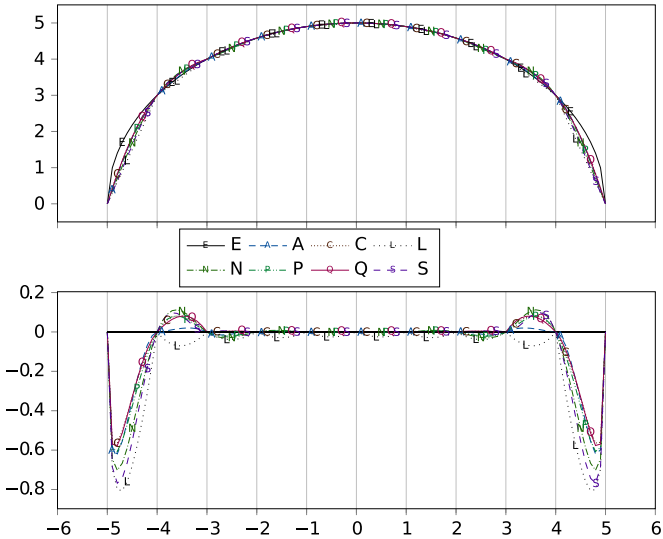

Рисунок 4. Интерполяция $\sqrt{25 - x^2}$ на $[-5, 5]$

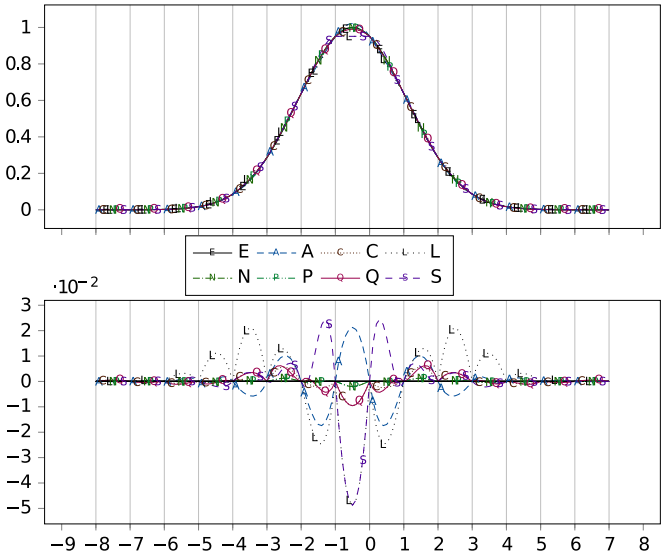

Рисунок 5. Интерполяция функции $e^{-0.1(x+0.5)^2}$ на $[-8, 7]$

получается при параболическом задании $f''(x_0) = \Delta_0$ и $f''(x_n) = \Delta_{n-2}$ по трём крайним точкам. Заметим, что Q на краях определяется по четырём точкам и имеет видимое преимущество.

3.1.2. *Интерполяция плотности нормального распределения*

Классическим пробным камнем для алгоритмов интерполяции является функция плотности вероятности нормального распределения. Общая картина представлена графиками рис. 5–9 и существенно зависит от сетки интерполяционных узлов.

Когда точка максимума не попадает в узел интерполяции (S на рис. 5), то методы, ориентированные на сохранение монотонности, резко теряют в точности, давая повышенную ошибку не только не экстремальном промежутке, но и как минимум на соседних с ним (S на рис. 9). Ряд из них (как например S на рис. 6) пытаются округлить заострённый график возле максимума в узле (S на рис. 5).

Даже при попадании максимума в узел бросается в глаза сложная зависимость результата сравнения от шага (рис. 6 против рис. 7). При уменьшении шага сегменты интерполяции становятся более похожими, рис. 9.

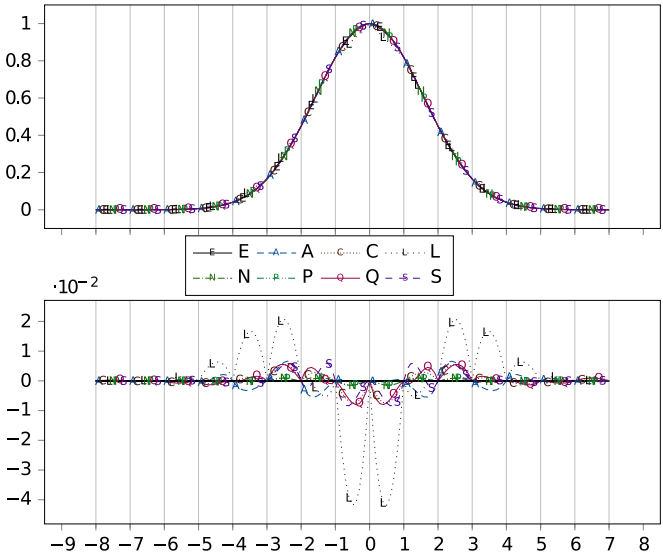

Рисунок 6. Интерполяция функции $e^{-0.1x^2}$ на $[-8, 7]$

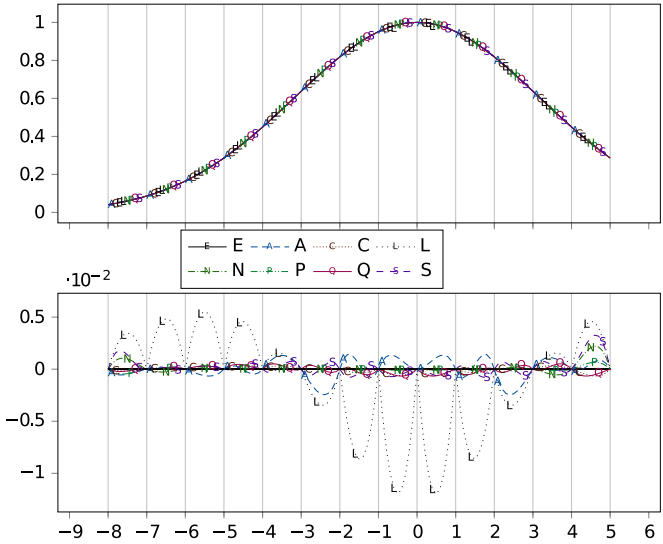

Рисунок 7. Интерполяция функции $e^{-0.05x^2}$ на $[-8, 5]$

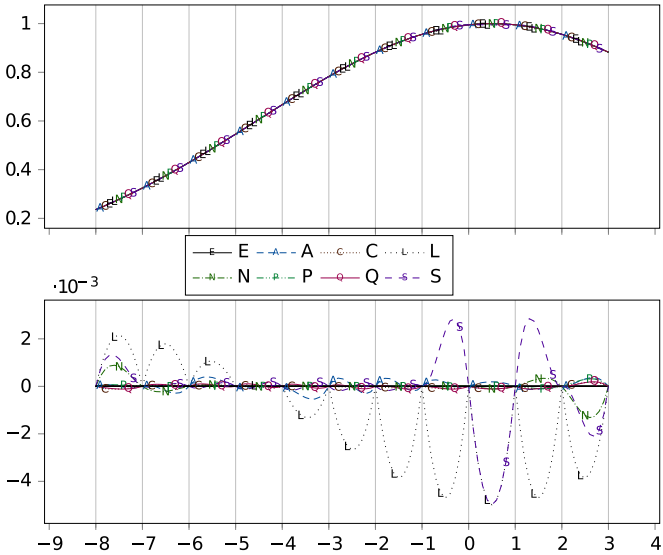

Рисунок 8. Интерполяция функции $e^{-0.01(x-0.5)^2}$ на $[-8, 3]$

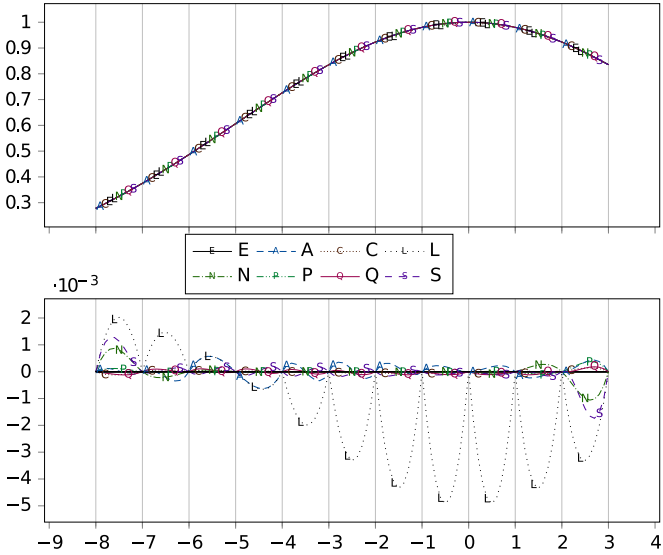

Рисунок 9. Интерполяция функции $e^{-0.01x^2}$ на $[-8, 3]$

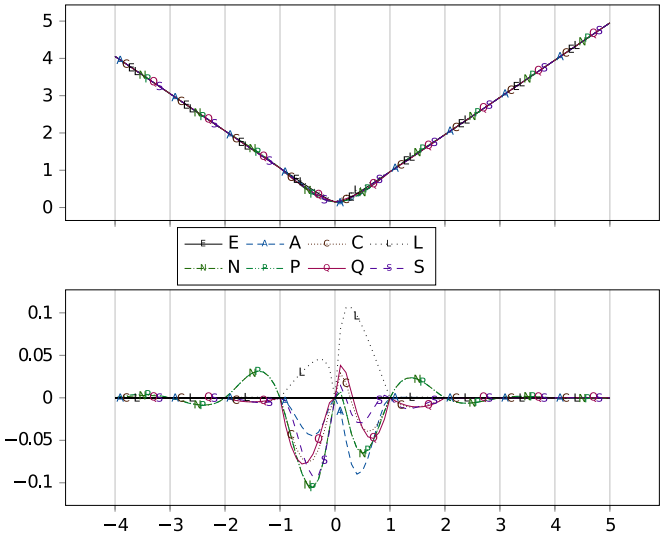

Рисунок 10. Интерполяция функции $\sqrt{0.02 + (x - 0.05)^2}$ на $[-4, 5]$

В рассмотренных примерах P и Q также явно лидируют среди локальных, а естественный кубический сплайн на рис. 5, 6, 8 и 9 также испорчен нулевыми граничными условиями.

3.1.3. *Другие функции с точками перегиба*

На многих рассмотренных примерах естественный кубический сплайн при правильном выборе граничных условий оказывается намного точнее локальных. Однако точность может теряться при крутых изгибах. В примере рис. 10 локальные P и Q заметно точнее.

К сожалению, такой выигрыш интерполяции радиуса локальности $l = 2$ неустойчив к сетке. Существенные различия в соотношении ошибок при разных конфигурациях узлов хорошо видны на рис. 11. Если особые точки не приближены к узлам, то положительный эффект негладкой стыковки в узлах компенсируется невозможностью представления негладкостей вне узлов и преимущество негладкого над естественным теряется на рис. 11.

Тем не менее преимущество негладких сплайнов над гладкими локальными остаётся явным. Оно сохраняется и для синуса, см. рис. 12.

Функции $\operatorname{tg} x$ (рис. 13), $\operatorname{arctg} x$ (рис. 14) и $\sqrt[3]{x}$ (рис. 15), показывают ситуацию большой общей ошибки алгоритмов, иллюстрирующей

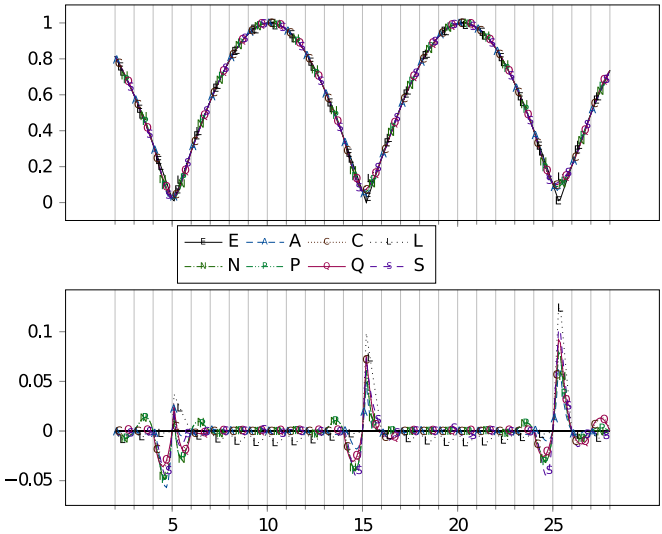

Рисунок 11. Интерполяция функции $|\cos(0.31x)|$ на $[2, 28]$

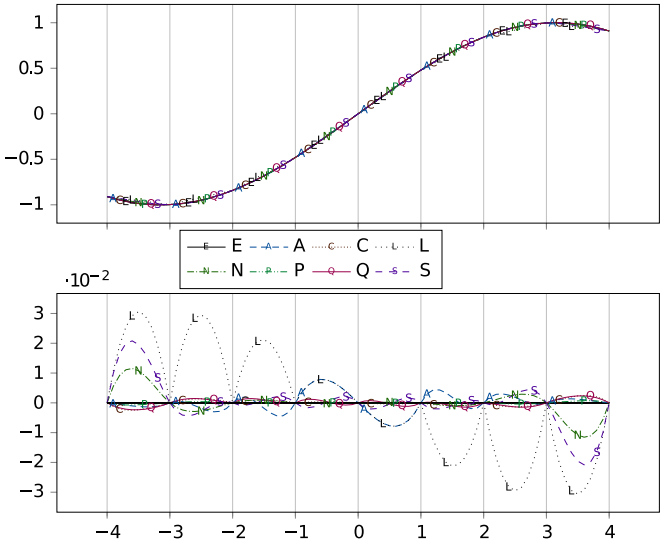

Рисунок 12. Интерполяция функции $\sin(x/2)$ на $[-4, 4]$

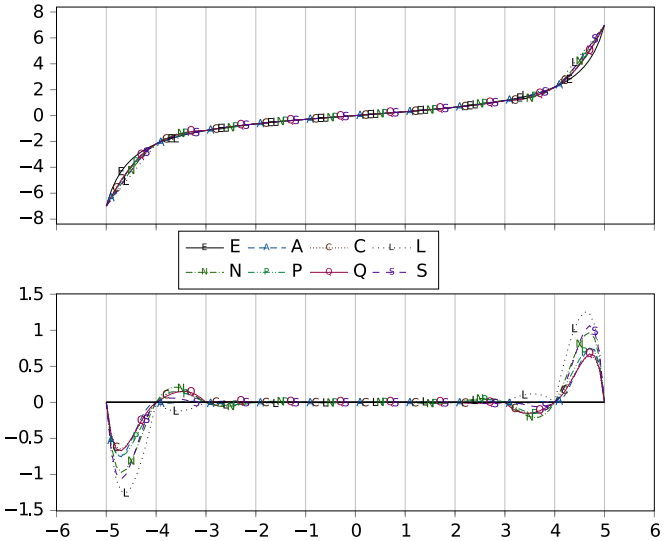

Рисунок 13. Интерполяция функции $\mathrm{tg}(x/6.5)$ на $[-5, 5]$

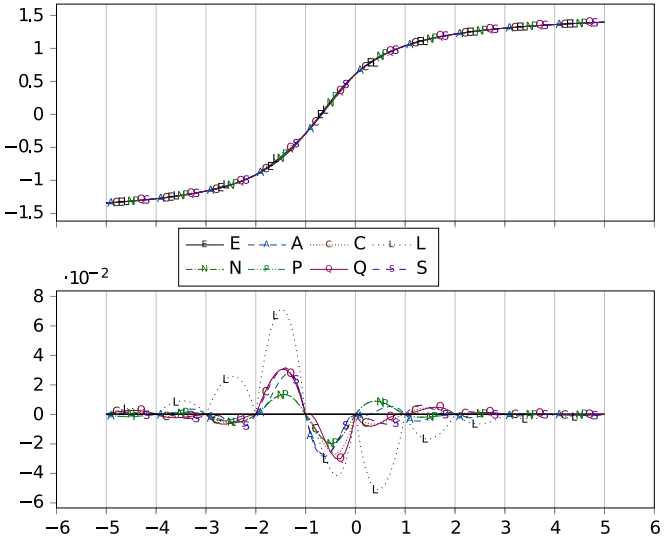

Рисунок 14. Интерполяция функции $\mathrm{arctg}(x + 0.7)$ на $[-5, 5]$

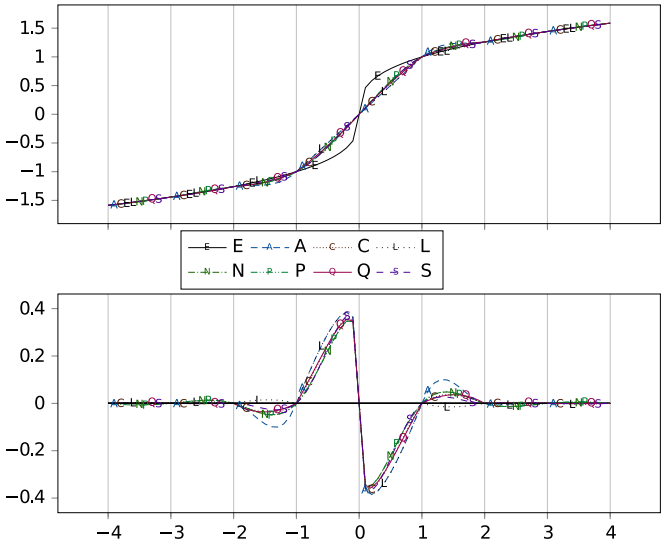

Рисунок 15. Интерполяция функции $\sqrt[3]{x}$ на $[-4, 4]$

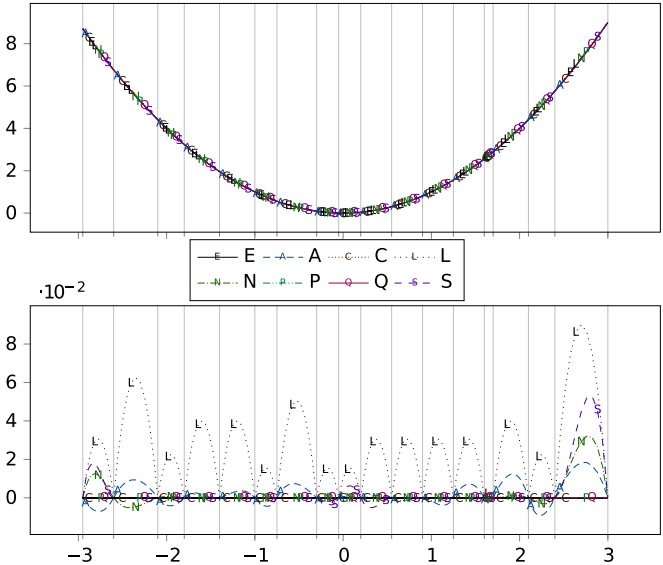

Рисунок 16. Интерполяция функции x^2 по сетке CD 7600

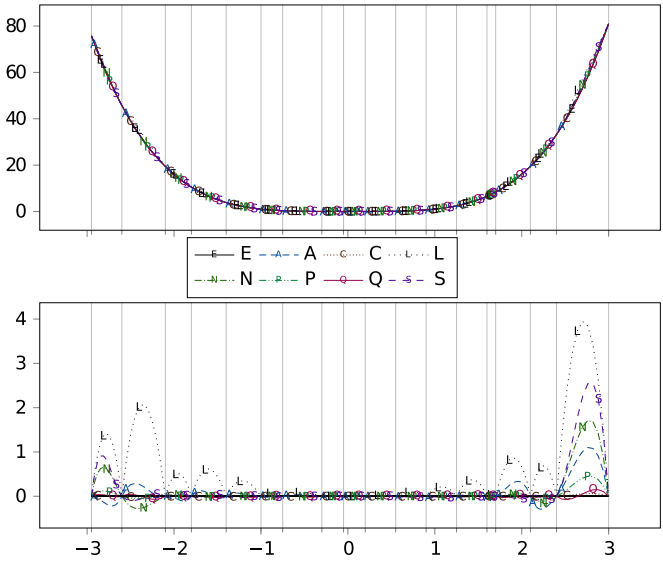

Рисунок 17. Интерполяция функции x^4 по сетке CD 7600

ситуацию попытки приближения функции с бесконечной производной.

3.1.4. *Интерполяция по неравномерной сетке*

Естественно возник вопрос о специфичности сетки с равномерными отсчётами и ином подборе тестовых функций. Поиск в опубликованных статьях ранее использовавшего набора тестов на точность интерполяции элементарных функций выявил лишь тест в [**11**] «Control Data 7600», включающий неравномерную сетку с узлами -2.95, -2.6, -2.1, -1.8, -1.4, -1.0, -0.75, -0.3, -0.05, 0.2, 0.55, 0.9, 1.25, 1.6, 1.7, 2.1, 2.4, 3.0 и 5 тестируемых функций.

Результаты проведения этого теста, представленные на рис. 16–20, чётко подтвердили лидерство P и Q среди сплайнов радиуса локальности $l < 3$. На этом наборе примеров они успешно конкурируют даже с естественными сплайнами, неоднократно уверенно превосходя их в точности.

3.2. Сводные таблицы итогов тестирования

Численные оценки достигнутой в рассмотренных примерах точности аппроксимации представлены в трёх таблицах. Относительные

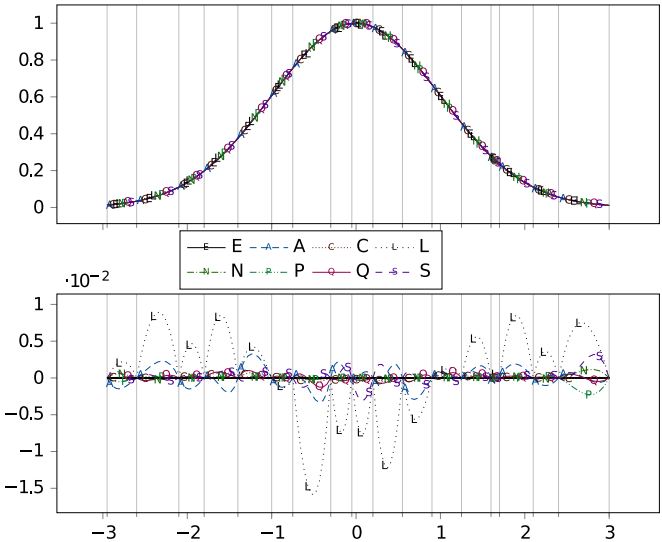

Рисунок 18. Интерполяция функции $\exp(x^2/2)$ по сетке CD 7600

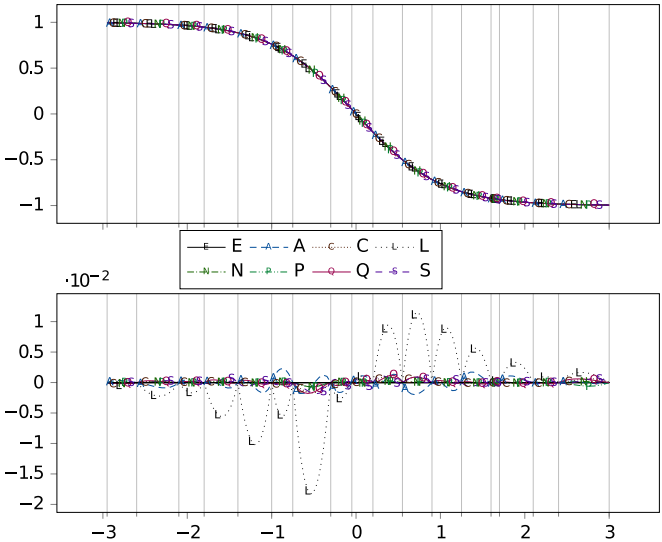

Рисунок 19. Интерполяция функции $\mathrm{th}(x)$ по сетке CD 7600

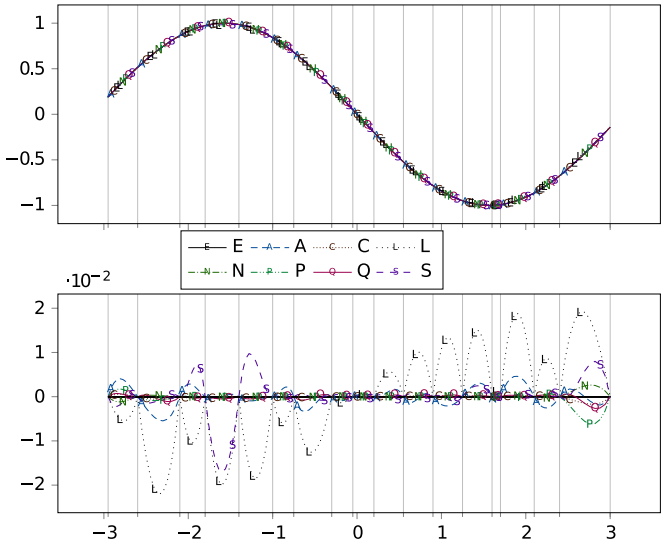

Рисунок 20. Интерполяция функции $\sin(x)$ по сетке CD 7600

максимальные погрешности приведены в таблице 1, относительные средние в таблице 2, а относительные среднеквадратичные — в таблице 3. Цветом выделены значения, **более чем вдвое** превосходящие лучшее из (других) сплайнов рассматриваемой локальности: менее чем вдвое; хуже в 2–5 раз; хуже в 5–10 раз; **хуже в 10 и более раз**.

3.2.1. *Предварительные итоги*

Таблицы, особенно последняя, показывают, что ни в одной испытывавшейся ситуации (сетка узлов, функция) точность квадратичного сплайна не уступила ни одному из других локальных сплайнов хотя бы в два раза, более чем в 20% случаев превзойдя в точности нелокальный естественный сплайн. В большинстве тестов видно не менее чем двукратное превосходство двух крайних справа над остальными сплайнами радиуса локальности $l = 2$.

Внимательное рассмотрение графиков показывает, что когда узлов немного (или функция часто изменяется) точность кусочно-квадратичных интерполянт слабо отличается от точности дважды гладких кубических сплайнов.

При этом ярко выделяются участки, на которых простые функции интерполируются крайне грубо. В рассмотренных примерах это

окрестности особых точек функции и места, где односторонняя или вторая производная многократно превышает средние значения или бесконечна (излом или вертикальная касательная на графике). Это дружные всплески погрешности на крайних участках в примерах рис. 4, 13 и внутри в примерах рис. 13–17.

Заметим, что в проведённом исследовании рассмотрены только линейные алгоритмы интерполяции на полиномиальной основе. Возникла гипотеза, что на этих участках интерполяция подобными алгоритмами затруднена. Для её проверки рассмотрим две ещё более выразительные функции рис. 21, 22.

Хорошо видно, что почти совпадающие ошибки всех алгоритмов достигли и превысили 50%. Хуже того, на последнем рисунке наименьшую погрешность (как максимальную так и среднеквадратичную) показала кусочно-линейная интерполяция. Это показывает, что рассмотренную постановку задачи интерполяции линейные алгоритмы на полиномиальной основе не решат.

Требуется алгоритм интерполяции, точно восстанавливающий все (точнее почти все) рациональные функции второй степени. Он не может быть линейным, поскольку линейная оболочка множества рациональных функций второй (и даже первой степени) бесконечномерна. В публикациях по теории интерполяции идей такого алгоритма не просматривается. Он мог бы повлиять на эффективность различных приложений методов теории аппроксимации, включая нелинейную регрессию и экстраполяцию.

Косвенно подтверждает эту мысль известная возможность существенно более точного, чем полиномами, приближения алгебраических и трансцендентных функций рациональными функциями. Такая возможность эффектно проиллюстрированная например в [**15**].

Выводы

1. Точность восстановления на участках с большими производными крайне низка и не зависит от сложности рассмотренных алгоритмов. Поиск алгоритма, безупречно восстанавливающего рациональные функции второй степени, является новой интересной задачей.
2. Гладкость интерполянты и сохранение монотонности (выпуклости, знака) не гарантируют повышенной точности интерполяции. Задачи интерполяции, в которых эти качества не требуются, могут быть решены более простыми и экономными способами.

Таблица 1. Максимальная относительная погрешность интерполяции (в процентах)

mesh	formula	P	N	A	S	L	C	Q
-3..3	e^x	2.38	4.87	2.41	6.37	8.56	1.41	1.58
1..6	$\ln(x-0.1)$	1.38	2.41	1.42	3.03	3.92	1.01	1.09
1..6	$1/x$	3.91	5.79	3.95	7.03	8.57	3.09	3.26
-5..5	$\sqrt{16-x^2}$	12.14	14.04	12.38	15.36	16.12	11.28	11.53
-8..7	$e^{-0.1(x+0.5)^2}$	**0.20**	**0.20**	2.13	4.88	4.88	0.96	0.96
-8..7	$e^{-0.1x^2}$	**0.15**	**0.15**	0.66	0.83	4.19	0.74	0.80
-8..5	$e^{-0.05x^2}$	0.07	0.22	0.25	0.33	**1.20**	0.07	0.07
-8..3	$e^{-0.01(x-0.5)^2}$	0.03	0.13	0.05	**0.50**	**0.50**	0.02	0.02
-8..3	$e^{-0.01x^2}$	0.04	0.11	0.06	**0.17**	**0.49**	0.01	0.02
-4..5	$\sqrt{0.02+(x-0.05)^2}$	2.17	2.17	1.85	1.92	2.18	1.57	1.60
2..28	$\lvert\cos(0.4x)\rvert$	7.97	7.96	6.37	10.23	12.41	9.07	9.22
-4..4	$\sin(x/2)$	**0.07**	1.15	0.79	2.08	**3.06**	0.25	0.23
-5..5	$\tan(x/3.3)$	10.78	13.89	10.77	15.24	17.96	9.30	9.62
-5..5	$\arctan(x+0.7)$	1.42	1.42	2.06	2.27	5.13	2.21	2.18
-4..4	$\sqrt[3]{x}$	21.91	21.90	24.24	23.27	24.24	22.75	22.38
CD	x^2	0	**0.37**	**0.21**	**0.61**	**1.03**	0	0
CD	x^4	0.59	**2.26**	**1.45**	**3.38**	**5.21**	0.08	0.20
CD	$\exp(x^2/2)$	0.23	0.11	0.33	0.32	**1.58**	0.08	0.13
CD	$\tanh(x)$	**0.07**	**0.07**	0.23	0.18	**1.84**	0.17	0.18
CD	$\sin(x)$	0.62	0.26	0.55	**1.70**	**2.21**	0.17	0.27
-5..5	$\frac{x}{1+5x^2}$	47.85	47.85	59.03	50.71	59.03	52.31	51.83
-15..15	$\sqrt[3]{\cos x}$	75.06	71.73	75.91	70.88	63.22	75.72	76.07

Таблица 2. Средняя относительная погрешность интерполяции (в процентах)

mesh	formula	P	N	A	S	L	C	Q
-3..3	e^x	1.96	3.95	1.66	4.35	8.53	1.46	1.46
1..6	$\ln(x-0.1)$	0.37	0.67	0.31	0.70	1.36	0.29	0.29
1..6	$1/x$	1.71	2.63	1.28	2.73	4.04	1.41	1.41
-5..5	$\sqrt{16-x^2}$	2.00	2.51	1.67	2.54	3.23	1.79	1.79
-8..7	$e^{-0.1(x+0.5)^2}$	**0.10**	**0.10**	1.54	1.78	3.23	0.61	0.64
-8..7	$e^{-0.1x^2}$	**0.11**	**0.11**	0.56	0.61	3.03	0.63	0.65
-8..5	$e^{-0.05x^2}$	**0.01**	0.04	0.12	0.09	**0.66**	0.04	0.05
-8..3	$e^{-0.01(x-0.5)^2}$	0.00	0.02	0.02	**0.11**	**0.25**	0.01	0.01
-8..3	$e^{-0.01x^2}$	0.01	0.02	0.03	0.03	**0.25**	0.01	0.01
-4..5	$\sqrt{0.02+(x-0.05)^2}$	0.72	0.71	0.35	0.36	0.46	0.39	0.42
2..28	$\lvert\cos(0.4x)\rvert$	1.23	1.22	0.71	1.31	1.66	1.09	1.10
-4..4	$\sin(x/2)$	**0.03**	0.34	0.34	0.61	**2.07**	0.12	0.13
-5..5	$\tan(x/3.3)$	8.27	11.03	6.02	9.25	13.52	7.05	7.05
-5..5	$\arctan(x+0.7)$	0.32	0.31	0.41	0.48	1.42	0.48	0.49
-4..4	$\sqrt[3]{x}$	4.90	4.83	6.25	4.51	5.34	4.89	4.89
CD	x^2	0	**0.10**	**0.15**	**0.14**	**0.88**	0	0
CD	x^4	0.24	**0.97**	**0.81**	**1.26**	**3.77**	0.06	0.11
CD	$\exp(x^2/2)$	0.05	0.03	0.25	0.15	**1.07**	0.05	0.07
CD	$\tanh(x)$	**0.01**	**0.01**	0.08	0.04	**0.49**	0.04	0.04
CD	$\sin(x)$	0.08	0.04	0.23	0.31	**1.27**	0.04	0.05
-5..5	$\frac{x}{1+5x^2}$	15.15	15.05	18.95	13.39	15.92	15.15	15.15
-15..15	$\sqrt[3]{\cos x}$	15.41	15.03	14.27	12.13	16.62	15.50	15.70

Таблица 3. Среднеквадратичная относительная погрешность интерполяции (в процентах)

mesh	formula	P	N	A	S	L	C	Q
-3..3	e^x	0.23	**0.48**	0.21	**0.60**	**0.91**	0.15	0.16
1..6	$\ln(x-0.1)$	0.06	0.11	0.06	0.14	**0.20**	0.05	0.05
1..6	$1/x$	0.29	0.44	0.27	0.52	0.68	0.24	0.24
-5..5	$\sqrt{16-x^2}$	0.43	0.52	0.42	0.57	0.65	0.40	0.40
-8..7	$e^{-0.1(x+0.5)^2}$	**0.01**	**0.01**	0.16	**0.25**	**0.31**	0.06	0.06
-8..7	$e^{-0.1x^2}$	**0.01**	**0.01**	0.06	0.07	**0.31**	0.06	0.06
-8..5	$e^{-0.05x^2}$	**0.00**	0.01	**0.01**	0.01	**0.07**	0.00	0.00
-8..3	$e^{-0.01(x-0.5)^2}$	0.00	**0.00**	0.00	**0.02**	**0.03**	0.00	0.00
-8..3	$e^{-0.01x^2}$	0.00	**0.00**	**0.00**	**0.01**	**0.03**	0.00	0.00
-4..5	$\sqrt{0.02+(x-0.05)^2}$	0.11	0.11	0.08	0.08	0.10	0.08	0.08
2..28	$\lvert\cos(0.4x)\rvert$	0.22	0.22	0.17	0.27	0.31	0.23	0.23
-4..4	$\sin(x/2)$	**0.00**	**0.05**	0.04	**0.10**	**0.23**	0.01	0.01
-5..5	$\tan(x/3.3)$	1.17	1.54	1.08	1.58	2.07	1.01	1.03
-5..5	$\arctan(x+0.7)$	**0.05**	**0.05**	0.08	0.09	0.21	0.08	0.09
-4..4	$\sqrt[3]{x}$	0.93	0.93	1.13	0.97	1.10	0.99	0.98
CD	x^2	0	**0.02**	**0.02**	**0.03**	**0.08**	0	0
CD	x^4	**0.04**	**0.15**	**0.10**	**0.22**	**0.42**	0.01	0.01
CD	$\exp(x^2/2)$	0.01	0.01	**0.02**	**0.02**	**0.11**	0.01	0.01
CD	$\tanh(x)$	**0.00**	**0.00**	0.01	0.01	**0.07**	0.01	0.01
CD	$\sin(x)$	**0.02**	0.01	**0.03**	**0.06**	**0.15**	0.01	0.01
-5..5	$\frac{x}{1+5x^2}$	2.68	2.68	3.45	2.76	3.30	2.93	2.92
-15..15	$\sqrt[3]{\cos x}$	2.60	2.53	2.63	2.38	2.58	2.64	2.64

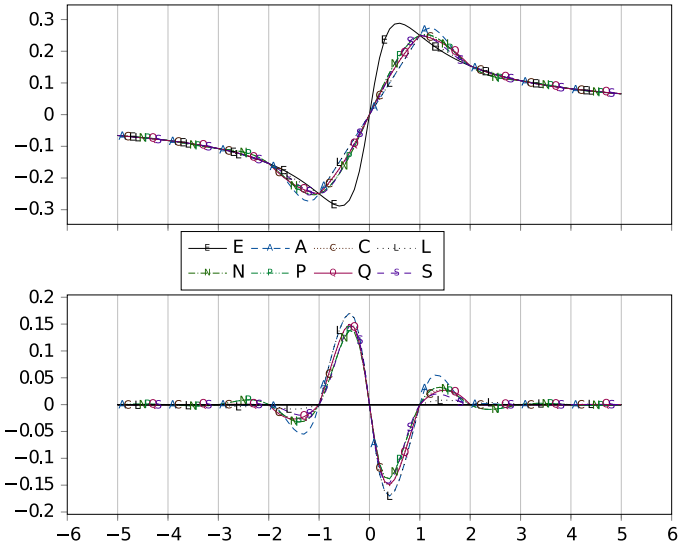

Рисунок 21. Интерполяция функции $\frac{x}{1+5x^2}$

3. При высоких требованиях к локальности сплайна (когда радиус локальности меньше 3) рассмотренный простой квадратичный сплайн сравним по точности с кубическими.

Список литературы

[1] E. Meijering. “A chronology of interpolation: from ancient astronomy to modern signal and image processing”, *Proceedings of the IEEE*, **90**:3 (2002), pp. 319–342. URL doi ↑69

[2] P. Thévenaz, T. Blu, M. Unser. “Interpolation revisited”, *IEEE Transactions on Medical Imaging*, **19**:7 (2000), pp. 739–758. URL doi ↑69

[3] Б. И. Квасов. «Монотонная и выпуклая интерполяция весовыми кубическими сплайнами», *Ж. вычисл. матем. и матем. физ.*, **53**:10 (2013), с. 1610–1621. doi ↑69

[4] К. Ю. Осипенко. «Об оптимальных методах восстановления в пространствах Харди–Соболева», *Матем. сб.*, **192**:2 (2001), с. 67–86. doi ↑69

[5] M. Fuentes. “A high frequency kriging approach for non-stationary environmental processes”, *Environmetrics*, **12**:5 (2001), pp. 469–483. doi ↑69

[6] N. C. Gabrielides, N. S. Sapidis. “C^1 sign, monotonicity and convexity

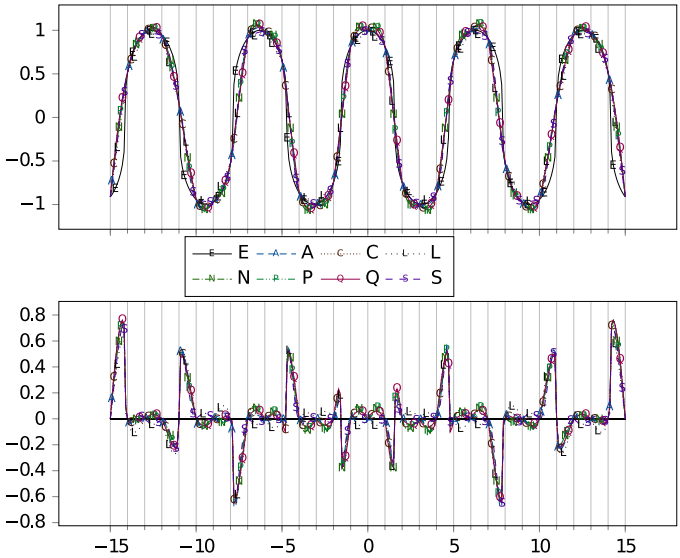

Рисунок 22. Интерполяция функции $\sqrt[3]{\cos(x)}$

preserving Hermite polynomial splines of variable degree", *Journal of Computational and Applied Mathematics*, **343** (2018), pp. 662–707. ↑69

[7] H. Akima. "A method of univariate interpolation that has the accuracy of a third-degree polynomial", *ACM Transactions on Mathematical Software*, **17**:3 (1991), pp. 341–366. ↑73

[8] H. Akima. *A method of smooth curve fitting*, ESSA Tech. Rep. ERL 101-ITS 73, U.S. Government Printing Office, Washington, D.C., 1969. ↑73

[9] M. Steffen. "A simple method for monotonic interpolation in one dimension", *Astron. Astrophys.*, **239** (1990), pp. 443–450. ↑73

[10] O. Niemitalo. *Polynomial interpolators for high-quality resampling of oversampled audio*, 2001. ↑73

[11] T. R. Ellis, D. H. McLain. "Algorithm 514: A new method of cubic curve fitting using local data [E2]", *ACM Transactions on Mathematical Software*, **3**:2 (1977), pp. 175–179. ↑73, 83

[12] A. D. Maude. "Interpolation—mainly for graph plotters", *The Computer J.*, **16**:1 (1973), pp. 64–65. ↑73

[13] J. M. Hyman. "Accurate monotonicity preserving cubic interpolation", *SIAM J. Sci. and Stat. Comput.*, **4**:4 (1983), pp. 645–654. ↑73

[14] H. T. Huynh. "Accurate monotone cubic interpolation", *SIAM J. Numer Anal.*, **30**:1 (1993), pp. 57–100. ↑73

[15] R. Schaback. “Adaptive rational splines”, *Constr. Approx.*, **6**:2 (1990), pp. 167–179. ↑86

Поступила в редакцию 05.10.2018
Переработана 24.10.2018
Опубликована 14.11.2018

Рекомендовал к публикации *д.т.н. А. М. Цирлин*

Пример ссылки на эту публикацию:

С. В. Знаменский. «Численная оценка точности интерполяции несложных элементарных функций». *Программные системы: теория и приложения*, 2018, **9**:4(39), с. 69–92. 10.25209/2079-3316-2018-9-4-69-92
http://psta.psiras.ru/read/psta2018_4_69-92.pdf
Эта же статья по-английски: 10.25209/2079-3316-2018-9-4-93-116

Об авторе:

Foto by A. Yu. Fomenko, CC-BY-SA

Сергей Витальевич Знаменский

Научные интересы мигрировали от функционального анализа и комплексных аналогов выпуклости к основаниям разработки коллаборативного ПО, метрикам сходства и теории интерполяции

0000-0001-8845-7627
e-mail: svz@latex.pereslavl.ru

Sample citation of this publication:

Sergej Znamenskij. “Numerical evaluation of the interpolation accuracy of simple elementary functions”. *Program Systems: Theory and Applications*, 2018, **9**:4(39), pp. 69–92. (*In Russian*).
10.25209/2079-3316-2018-9-4-69-92
http://psta.psiras.ru/read/psta2018_4_69-92.pdf
The same article in English: 10.25209/2079-3316-2018-9-4-93-116