Отчёт по заданию практикума «Исследование ошибок интерполяции полиномом Лагранжа»

Мурат Апишев
great-mel@yandex.ru
МГУ имени М. В. Ломоносова, Москва
28 ноября 2014 г.

Содержание

1 Введение

Зачастую в фундаментальной и прикладной математике возникает необходимость численного решения задачи в силу тех или иных причин. Численные методы являются одним из мощных математических средств решения задачи. Но, помимо разработки математической модели и непосредственно численного метода, требуется также и разработка алгоритма, реализующего данный метод с необходимой точностью. Однако окончательную оценку метода можно дать только после его использования в практических расчетах. К сожалению, зачастую результат решения задачи с помощью численного метода может отличаться от результата, полученного аналитическим методом. Это связано с тем, что в исходных данных могут присутствовать ошибки, что численный метод чаще всего предполагает некоторое отклонение от истинного ответа, и с тем, при выполнении арфметических операций на ЭВМ возникают ошибки округления. В связи с этим в процессе решения задачи прежде всего необходимо оценить возможную ошибку метода. На практике очень часто приходится иметь дело с данными, которые представлены в виде таблиц и задают зависимость одних параметров исследуемого явления от других. Задача состоит в том, чтобы по таким данным восстановить соответствующую аналитическую зависимость. Данная проблема имеет большое практическое значение, поскольку порой расчёт значения функции в точке может быть трудоёмким или попросту неосуществимым процессом. В рамках данной работы рассматривается задача интерполирования функции полиномом Лагранжа, а также проводится анализ ошибок данного численного метода.

2 Постановка теоретической задачи

Пусть на отрезке $[a, b]$ определена некоторая неизвестная функция $f(x)$, значения которой известны в конечном числе несовпадающих точек $x_0, x_1, \dots, x_n$, $\forall i \in 0, \dots, n \implies x_i \in [a, b]$. Точки эти назовём узлами интерполяции и будем считать их упорядоченными по возрастанию, т.е. $a = x_0 < x_1 < \dots < x_n = b$. По данным значениям $f(x_i)$ необходимо построить функцию $F(x)$, называемую интеполирующей функцией, которая бы приближала бы $f(x)$, совпадая с ней в узлах интерполяции.

3 Описание метода интерполяции полиномом Лагранжа

Общее описание Существуют различные подходы решению задачи интерполяции. Одним из наиболее распространённых является интерполяция полиномами. В данном подходе выделяется линейное интерполирование т.н. полиномами Лагранжа, котороеы и будет рассматриваться далее.

Попробуем приблизить $f(x)$ полиномом $L_n(x)$ степени не выше n. Будем искать $L_n(x)$ следующего вида:

$$L_n(x) = \sum_{k=1}^{n+1} f(x_k)\Phi_k(x), \quad \Phi_k(x_j) = \begin{cases} 1, & \mathrm{k = j} \\ 0, & k \neq j \end{cases}$$

Функции $\Phi_n(x)$ строятся достаточно легко. Действительно, функция

$$(x - x_1) \cdot (x - x_2) \cdot \ldots \cdot (x - x_{k-1}) \cdot (x - x_{k+1}) \cdot \ldots \cdot (x - x_n) \cdot (x - x_{n+1})$$

является полиномом степени n, который обращается в ноль для всех x_j не равных x_k. В точке x_k она принимает значение

$$(x_k - x_1) \cdot (x_k - x_2) \cdot \ldots \cdot (x_k - x_{k-1}) \cdot (x_k - x_{k+1}) \cdot \ldots \cdot (x_k - x_n) \cdot (x_k - x_{n+1})$$

Тогда

$$\Phi_k(x) = \frac{(x - x_1) \cdot (x - x_2) \cdot \ldots \cdot (x - x_{k-1}) \cdot (x - x_{k+1}) \cdot \ldots \cdot (x - x_n) \cdot (x - x_{n+1})}{(x_k - x_1) \cdot (x_k - x_2) \cdot \ldots \cdot (x_k - x_{k-1}) \cdot (x_k - x_{k+1}) \cdot \ldots \cdot (x_k - x_n) \cdot (x_k - x_{n+1})} \tag{1}$$

В таком случае, интерполяционный полином Лагранжа примет вид

$$L_n(x) = \sum_{k=1}^{n+1} f(x_k) \prod_{\substack{j=1,2,\ldots n+1 \\ j \neq k}} \frac{x - x_j}{x_k - x_j} \tag{2}$$

Задача метода Алгоритм построение полинома Лагранжа функции одной переменной получает на вход вектор аргументов и вектор соответствующих значений функции $f(x)$ в этих точках. Параметром метода является число точек-аргументов — оно определяет максимальную степень полиномов, в классе которых будет производиться поиск решения. Результатом работы алгоритма должен стать вектор коэффициентов, однозначно характеризующий найденный полином.

Процедура метода Формула метода полностью определяется выражением 2. Алгоритм поиска полинома приведён ниже:

инициализировать вектор коэффициентов L_n нулями
для всех возможных степеней k полинома L_n **выполнять**
  инициализировать знаменатель z единицей
  инициализировать временный полином Φ_k единицей
  для всех возможных степеней j будущего L_n **выполнять**
    если $k \neq j$ **то**
      умножить g на $(x - x_j)$
      прибавить к z $(x_k - x_j)$
  поделить все коэффициенты Φ_k на z
  умножить все коэффициенты Φ_k на значение функции в k-м узле
  прибавить к вектору коэффициентов L_n вектор коэффициентов Φ_k

---

После того, как коэффициенты полинома Лагранжа найдены, поиск приближённого значения функции в новой точке производится с помощью вычисления значения полинома в этой точке. Т.е. нужно вычислить значение полинома $P(x) = a_0 + x(a_1 + x(a_2 + \ldots x(a_{n-1} + a - Nx)))$ в точке $x = x_0$. Это вычисление эффективно производится с помощью схемы Горнера:

$$b_n = a_n, \quad b_{n-1} = a_{n-1} + b_n x_0, \quad \ldots, \quad b_0 = a_0 + b_1 x_0$$

Полученное значение b_0 и будет искомым.

4 Анализ метода

Обоснование метода Интерполяция полиномом Лагранжа позволяет производить достаточно точное приближение аппроксимируемой функции в случае относительно небольшого числа узлов интерполяции. Однако, как и всякий численный метод, она несовершенна и получаемый с её помощью результат сильно зависит всевозможных ошибок. Различные виды ошибок будут рассматриваться далее. Введём неоходимые обозначения:

- $y = f(x)$ — точное, полученное с помощью вычисления по аналитической формуле, значение $f(x)$ в точке x.
- $\hat{y}$ — значение $f(x)$ в точке x, полученное на ЭВМ.
- y_a — точное значение интерполирующей функции $F(x)$ в точке x.
- $\hat{y}_a$ — значение интерполирующей функции $F(x)$ в точке x, полученное на ЭВМ.

Ошибка исходных данных В рамках проводимого численного эксперимента предполагаем, что исходные данные заданы точно, т.е. выполняется соотношение

$$\hat{y} = y = f(x), \forall x \in [a, b]$$

Ошибка численного метода Для погрешности интерполяции полиномами справедлива следующая теорема:

Теорема 1 *Пусть $f(x) \in C^{n+1}[a,b]$, $P_n(x)$ — интерполирующий полином степени n функции $f(x)$ на $[a,b]$. Тогда для $R_n(x) = y - y_a$, $x \in [a,b]$, справедлива следующая оценка:*

$$|R_n(x)| \leq \frac{M_{n+1}}{(n+1)!}|\sigma_{n+1}(x)|,$$

где

$$\sigma_{n+1}(x) = (x - x_0)(x - x_1)\dots(x - x_n),$$
$$M_{n+1} = x \in [a,b] \max|f^{(n+1)}(x)|$$

Зависимость аналитической ошибки метода от количества узлов интерполяции n для задачи аппроксимации функции $f(x) = \exp(x)$ на отрезке $[0, 0.9]$ отражена на графике 1.

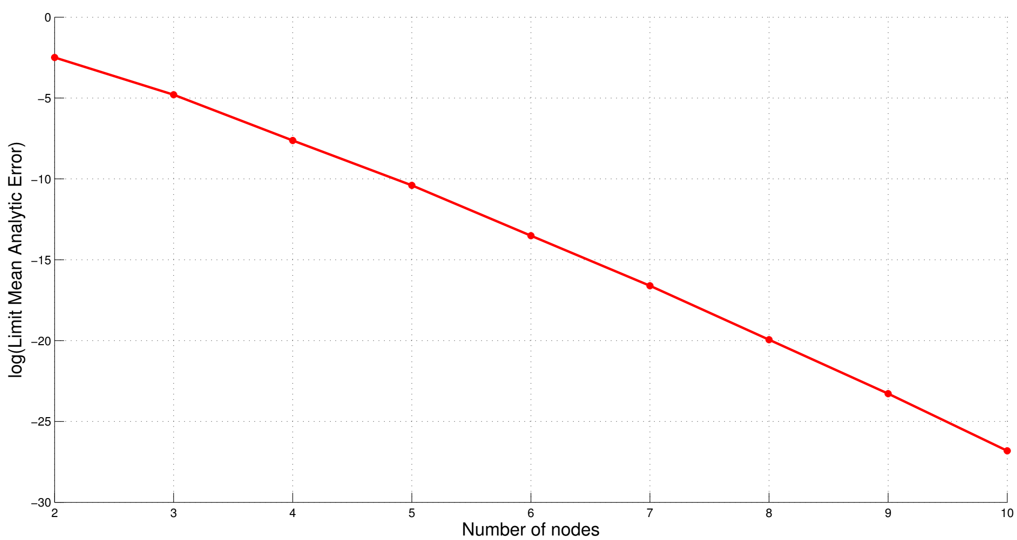

Рис. 1: Зависимость логарифма средней аналитической ошибки от числа узлов интерполяции.

Ошибка округления Введём необходимые обозначения:

- $e(x)$ — абсолютная ошибка (погрешность) вычисления величины x;
- $\varepsilon(x)$ — относительная ошибка (погрешность) вычисления величины x;
- ε_0 — машинный нуль.

Сначала рассчитаем погрешности вычисления коэффициентов полинома Лагранжа. Будем пользоваться нотацией, введённой в алгоритме. Для каждого узла интерполяции k следует рассчитать коэффициенты полинома 1. Опишем процесс подсчёта ошибки вычисления k-го полинома.

Инициализировав Φ_k единицей, будем последовательно умножать его на содержимое каждой пары скобок в числителе. Тогда ошибка 0-го коэффициента будет выражаться следующей рекурентной формулой

$$e^0(c_0) = 0, \quad e^j(c_0) = |c_0 \cdot x_j| \left(\varepsilon^{j-1}(c_0) + \varepsilon(x_j) + \frac{\varepsilon_0}{2}\right)$$

,

а ошибка любого другого i-го коэффициента — следующей

$$e^0(c_i) = 0, \quad e^j(c_0) = |c_i \cdot x_j| \left(\varepsilon^{j-1}(c_i) + \varepsilon(x_j) + \frac{\varepsilon_0}{2}\right) + |c_{i-1} \cdot 1.0| \left(\varepsilon^{j-1}(c_{i-1}) + \varepsilon(1.0) + \frac{\varepsilon_0}{2}\right)$$

Посчитав погрешность вычисления числителя, перейдём нормировочной константе. Инициализировав её так же единицей, будем умножать последовательно на содержимое каждой пары скобок в знаменателе. Тогда погрешность можно будет получить из следующего рекурентного соотношения

$$e^0 z = 0, \quad e^j(z) = |z \cdot (x_k - x_j)| \left(e(z) + \frac{\varepsilon_0 + \frac{\varepsilon_0}{2}|x_k - x_j|}{|x_k - x_j|} + \frac{\varepsilon_0}{2}\right)$$

После того, как погрешности округления при вычислении найдены и для числителя, и для знаменателя, распишем ошибку деления для произвольного i-го коэффициента:

$$e(c_i) = \left|\frac{c_i}{z}\right| \left(\varepsilon(c_i) + \varepsilon(z) + \frac{\varepsilon_0}{2}\right)$$

Последним шагом является умножение полученного Φ_k на значение функции в соответствующем узле и прибавление его к строящемуся полиному Лагранжа. Ошибка округления для i-го коэффициента примет вид

$$e(c_i^{lagrange}) = |c_i \cdot y_i| \left(e(c_i) + \varepsilon_0\right) + |c_i^{lagrange}| \; e(c_i^{lagrange}) + \left(\frac{\varepsilon_0}{2} \left|c_i^{lagrange} + c_i \cdot y_i\right|\right)$$

Теперь посчитаем ошибку погрешности вычисления значения функции в точке с помощью полинома Лагранжа. Ниже описана ошибка погрешности для схемы Горнера:

Для 1-ого шага схемы Горнера имеем:

$$e_{n-1}^r := e(c_n \cdot x + c_{n-1}) = e(c_n \cdot x) + e(c_{n-1}) + \frac{\varepsilon_0}{2}|c_n \cdot x + c_{n-1}| =$$

$$= |c_n \cdot x| \left(\varepsilon(c_n) + \varepsilon(x) + \frac{\varepsilon_0}{2}\right) + e_{n-1}^c + \frac{\varepsilon_0}{2}|c_n \cdot x + c_{n-1}| =$$

$$= |c_n \cdot x| \left(\frac{e_n^c}{|c_n|} + \varepsilon_0\right) + e_{n-1}^c + \frac{\varepsilon_0}{2}|c_n \cdot x + c_{n-1}|.$$

После $n - s$ шагов схемы горнера будут получены значения ошибок $e_i^r, i = \overline{s, n}$. Обозначим $M = (\ldots((c_n \cdot x + c_{n-1})x + c_{n-2})x + \ldots)x + c_s$, тогда для $(n - s + 1)$-ого шага получим:

$$e_{s-1}^r := e(Mx + c_{s-1}) = e(Mx) + e(c_{s-1}) + \frac{\varepsilon_0}{2}|Mx + c_{s-1}| =$$

$$= |Mx| \left(\varepsilon(M) + \varepsilon(x) + \frac{\varepsilon_0}{2}\right) + e(c_{s-1}) + \frac{\varepsilon_0}{2}|Mx + c_{s-1}| =$$

$$= |Mx| \left(\frac{e_s^r}{|M|} + \varepsilon_0\right) + e_{s-1}^c + \frac{\varepsilon_0}{2}|Mx + c_{s-1}|.$$

Таким образом, мы посчитали оценку величины $|\hat{y}_a - y_a|$. Единое аналитическое выражение для ошибки вычислений привести не удалось в связи с тем, что все составляющие формулы являются рекурентными. Необходимо заметить, что ошибка вычислителя получается слишком завышенной при больших абсолютных значениях интерполяции, а также в случае высоких степеней интерполирующего полинома. На графике 2 показана зависимость предельной ошибки вычисления от количества узлов интерполяции при интерполяции функции $f(x) = \exp(x)$ на отрезке $[0, 0.9]$.

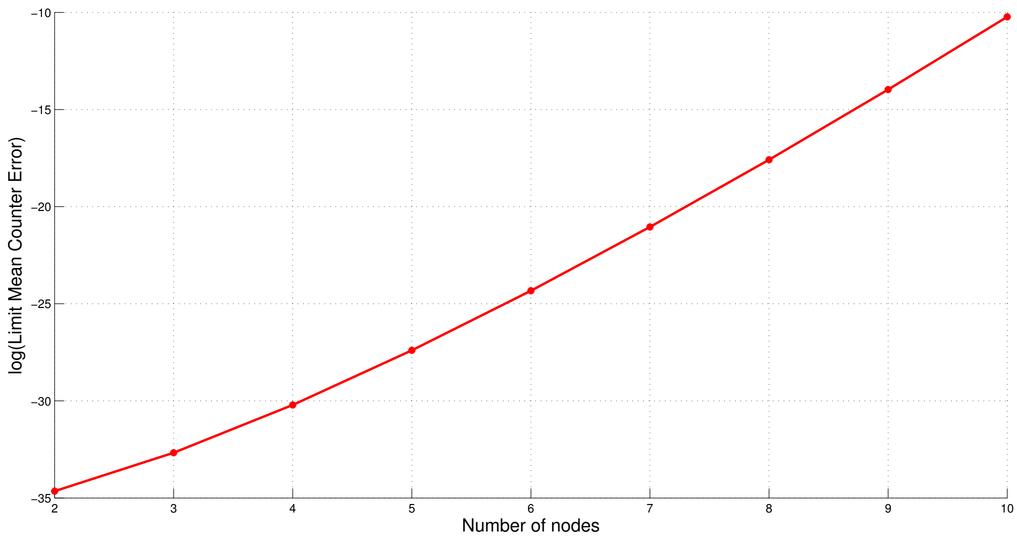

Рис. 2: Зависимость логарифма средней предельной вычислительной ошибки от числа узлов интерполяции.

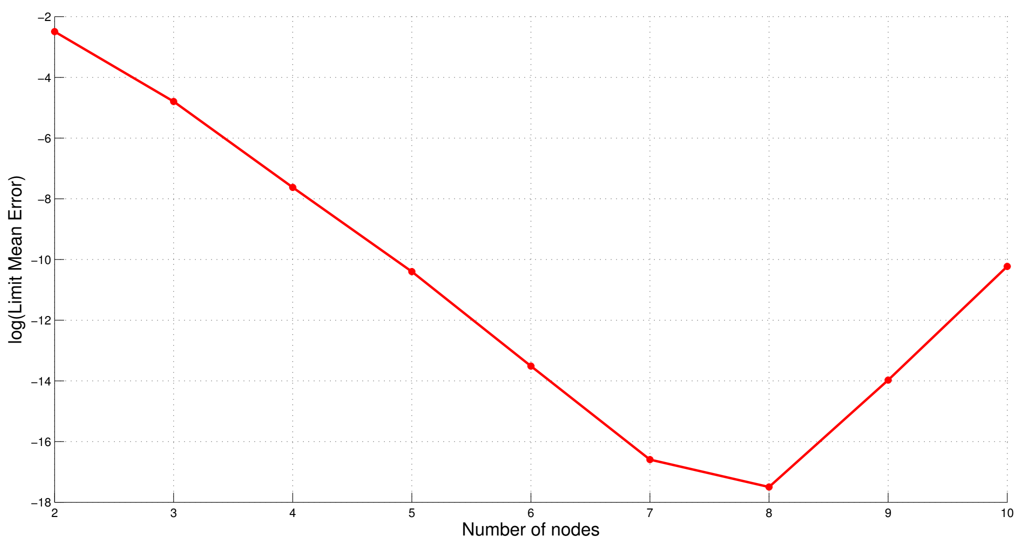

Рис. 3: Зависимость логарифма средней предельной ошибки от числа узлов интерполяции.

Общая ошибка Исходя из вышеупомянутых оценок, можем оценить предельную ошибку метода следующим образом:

$$|\hat{y}_a - y| \leq |\hat{y}_a - y_a| + |y_a - y|.$$

Зависимость суммарной предельной ошибки от количества узлов интерполяции n для задачи интерполяции функции $f(x) = \exp(x)$ на отрезке $[0, 0.9]$ отражена на рис. 3.

5 Модельная задача №1

Теоретическая задача Необходимо проинтерполировать функцию $5\cos(x)$ на отрезке $[-0.8, 0.9]$, и исследовать зависимости различных ошибок от количества узлов интерполяции.

Результаты применения метода Графики ошибок отражены на рис. 4, 5, 6 и 7. Можно заметить, что увеличение числа узлов приводит к уменьшению как средней аналитической ошибки, так и средней предельной ошибки. Средняя ошибка погрешностей вычислений же наоборот, растёт, что вполне закономерно, так как объём вычислений растёт вместе с числом узлов.

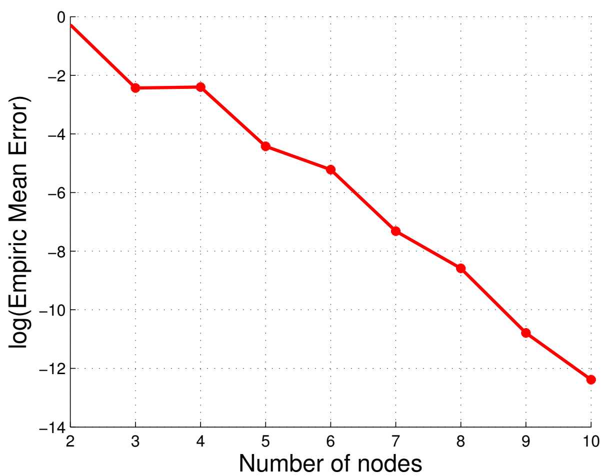

Рис. 4: Зависимость логарифма средней эмпирической ошибки от числа узлов интерполяции.

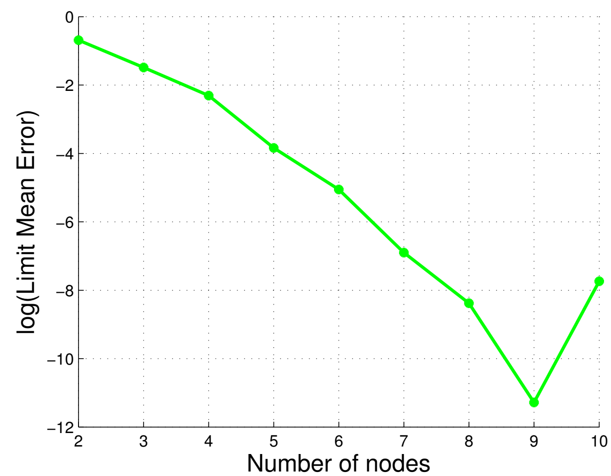

Рис. 5: Зависимость логарифма средней предельной ошибки от числа узлов интерполяции.

6 Модельная задача №2

Теоретическая задача Необходимо проинтерполировать функцию $4x^2 + \exp(2x)$ на отрезке $[0, 1]$, и исследовать зависимости различных ошибок от количества узлов интерполяции.

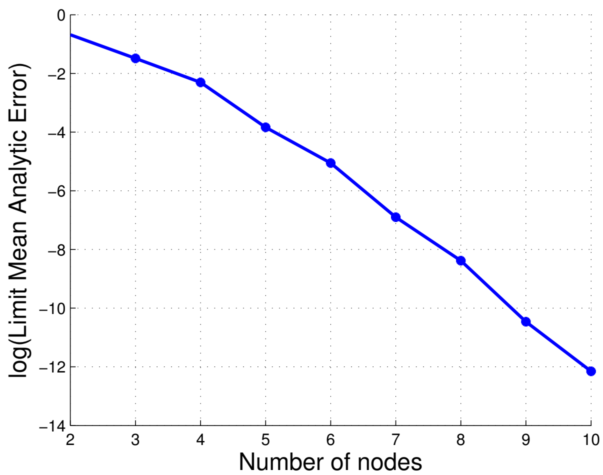

Рис. 6: Зависимость логарифма средней предельной аналитической ошибки от числа узлов интерполяции.

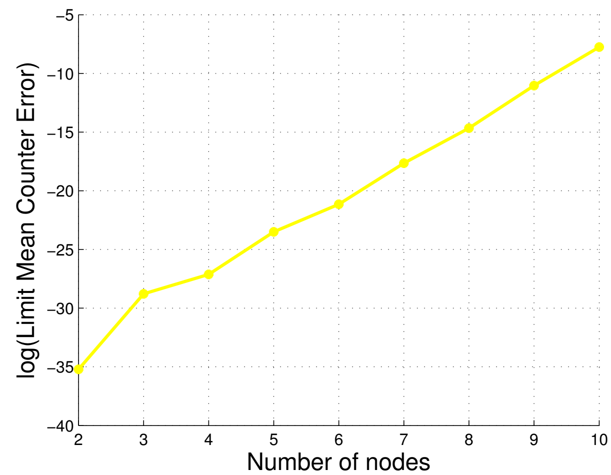

Рис. 7: Зависимость логарифма средней предельной вычислительной ошибки от числа узлов интерполяции.

Результаты применения метода Результаты замеров ошибок показаны на рис. 8, 9, 10 и 11. Видно, что подтверждаются выводы, сделанные по итогам решения предыдущей модельной задачи.

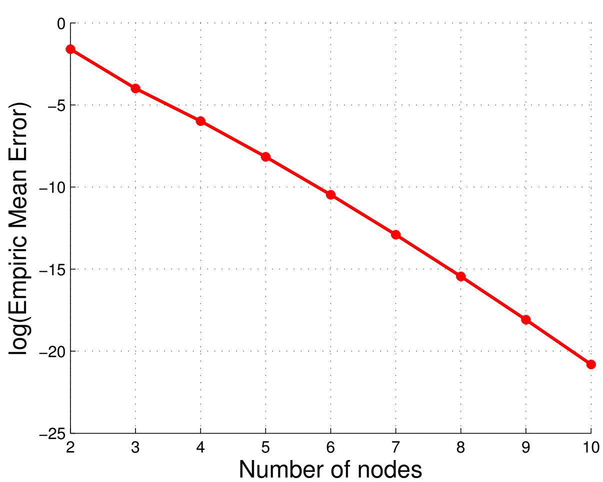

Рис. 8: Зависимость логарифма средней эмпирической ошибки от числа узлов интерполяции.

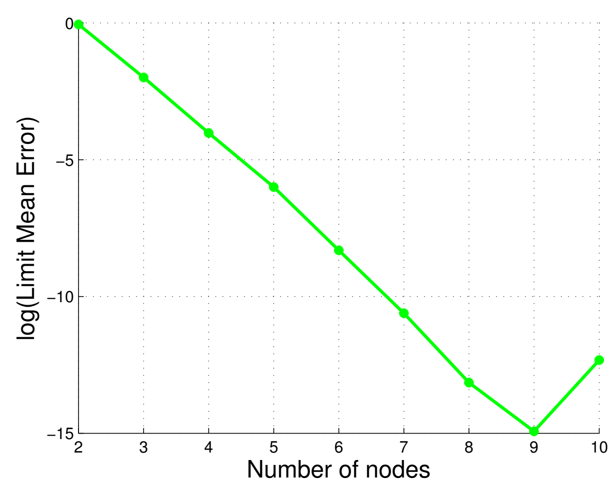

Рис. 9: Зависимость логарифма средней предельной ошибки от числа узлов интерполяции.

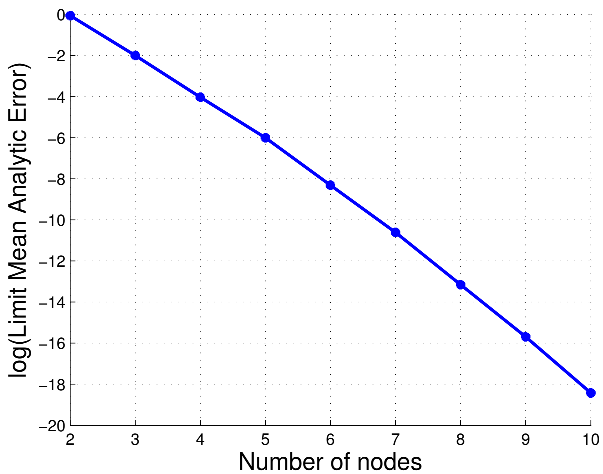

Рис. 10: Зависимость логарифма средней предельной аналитической ошибки от числа узлов интерполяции.

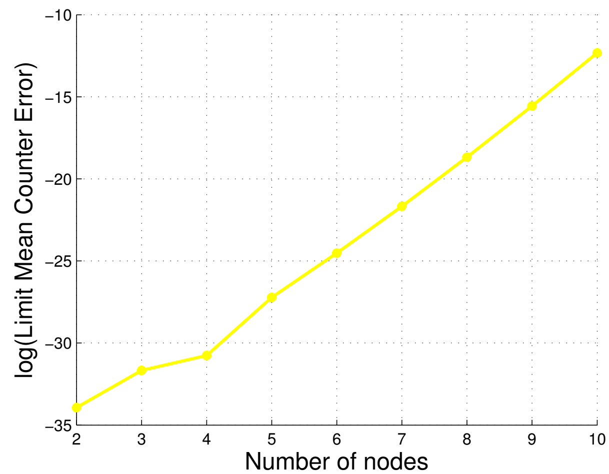

Рис. 11: Зависимость логарифма средней предельной вычислительной ошибки от числа узлов интерполяции.

7 Модельная задача №3

Теоретическая задача Проведём большое число автоматически сгенерированных экспериментов, чтобы исследовать интерполяцию полиномом Лагранжа статистически.

С помощью статистического генератора будем синтезировать функции следующего вида:

$$f(x) = a_0 + a_1 x + \sin(bx) + c e^{dx},$$

где $a_0 \sim \mathrm{U}[0,1],\ a_1 \sim [0,1],\ b \in 0,1,\ldots,9,\ c \in 0,1,\ldots,4,\ d \sim \mathrm{U}[0,3)$.
Интерполяция, соответственно, ведётся на отрезке $[0, 0.9]$ при использовании равномерной сетки узлов интерполяции и числа узлов интерполяции $n = 4$. Указанное семейство функций достаточно разнообразно для того, чтобы адекватно оценить результаты интерполяции.

Исследуем график нормированной ошибки для 20000 запусков статистического генератора. Под нормированной ошибкой понимается выражение

$$\frac{\hat{y} - f(x)}{\frac{M_{n+1}}{(n+1)!}\left|\omega_{n+1}(x)\right| + e(\hat{y})}$$

Результаты применения метода На графике 12 видно, что ошибка густо сконцентрированна у нуля, поэтому можно сказать, что на функциях из заданного семества вероятность получения существенной ошибки при интерполяции полиномом Лагранжа относительно невелика.

8 Параметры вычислителя

Сборка программы производилась с помощью компилятора *Intel C++ Compiler XE 13.0*.

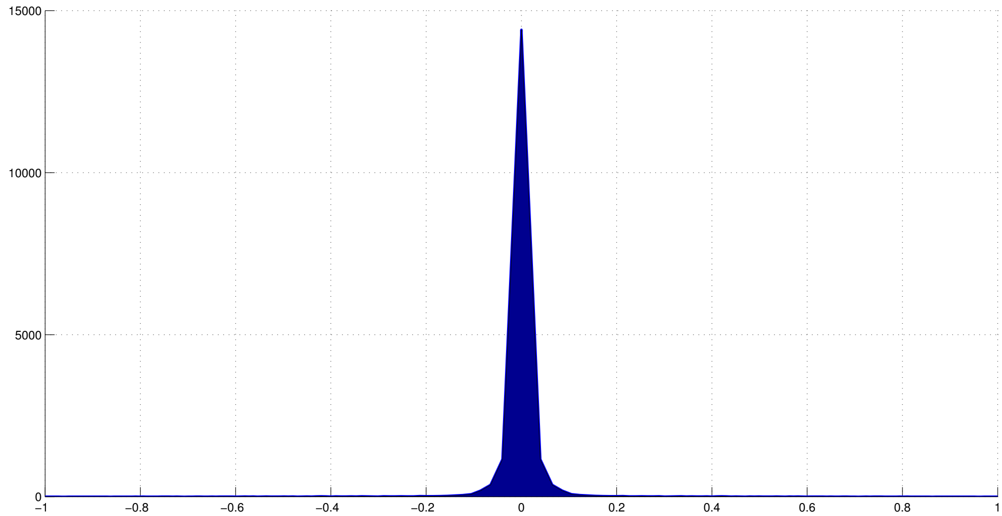

Рис. 12: Плотность нормированной ошибки (ось аргумента — $\sqrt[0.7]{x}$).

Команда компиляции:
`icl /fp:strict /Qstd=c++11 srcmain.cc lagrange.cc`

9 Заключение и выводы

В данной работе был реализован и исследован меод интерполяции полиномами Лагранжа. Исходя из полученных результатов можно сделать следующие выводы:

1. Предельная аналитическая ошибка метода убывает при увеличении числа узлов интерполяции.
2. Предельная ошибка округления метода увеличивается с увеличением количества узлов.
3. Суммарная предельная ошибка, как сумма двух предыдущих, ведёт себя так же, как и большее при данном количестве узлов слагаемое: вначале убывает вслед за анлитической ошибкой, затем, с некоторого момента, начинает возрастать вместе с ошибкой погрешности.
4. Поскольку, несмотря на все оценки, при любых взятых в экспериментах количествах узлов интерполяции эмпирическая ошибка всегда убывала и была существенно меньше, чем предельная ошибка, можно сделать вывод, что полученные предельные оценки являются сильно завышенными.
5. Основным выводом является подтверждение того, что интерполяция полиномами Лагранжа является достаточно точным и удобным методом для приближения функций, заданных на небольшой (порядка 10) сетке точек.