Функции Ферми-Дирака

Н.Н. Калиткин, С.А. Колганов

26 октября 2019 г.

Оглавление

Глава 1

ВВЕДЕНИЕ

§1.1. Определение функций

Функции Ферми-Дирака (далее ФД) возникают в задачах квантовой механики при описании свойств вещества, обусловленных поведением электронов или других фермионов. Как известно, функция распределения электронов в 6-мерном пространстве импульсов $\mathbf{p}$ и координат $\mathbf{r}$ имеет следующий вид:

$$f(\mathbf{p},\mathbf{r}) = \frac{const}{1+\exp\left(\frac{\frac{1}{2}\mathbf{p}^2+\phi(\mathbf{r})-\mu}{T}\right)}. \tag{1.1}$$

Здесь $\phi(\mathbf{r})$ есть потенциал электрического поля, μ – химический потенциал, а T – температура. Константа есть статистический вес частицы, определяемый её спином; для электрона $const = 2$. Все физические формулы написаны в атомной системе единиц; за единицу берутся масса электрона, заряд электрона и постоянная Планка.

При решении квантово-механических задач используются различные моменты фермиевского распределения. Они равны сверткам различных степеней импульса $\mathbf{p}$ с этим распределением по объёму в импульсном пространстве $d\mathbf{p} = 4\pi^2 dp$. Например, электронная плотность в пространстве координат есть свертка нулевой степени импульса:

(1.2)

Плотность кинетической энергии есть свертка квадрата импульса:

(1.3)

Нахождение проводимости требует нахождения потока частиц, т.е. умножения $f(\mathbf{p},\mathbf{r})$ на импульс $\mathbf{p}$; это эквивалентно введению в числитель подынтегрального выражения (2) степени p^3 вместо p^2. Теплопроводность определяется через поток кинетической энергии; это эквивалентно введению в числитель подынтегрального выражения (3) степени p^5 вместо p^4.

В таких свертках принято делать следующую замену переменных:

$$t = p^2/(2T), x = (\mu - \phi(\mathbf{r}))/T.(4) \tag{1.4}$$

Тогда различные моменты с точностью до константы приобретают следующий вид:

$$(1)(1)(5 \tag{1.5}$$

Например, $k = 1/2$ для электронной плотности (2), $k = 1$ для потока частиц, $k = 3/2$ для плотности кинетической энергии (3) и $k = 2$ для плотности потока энергии(электронной теплопроводности). Целые k соответствуют нечетным моментам импульса, а полуцелые – четным.

В математической теории функций ФД рассматриваются произвольные индексы k и комплексные значения x.

Интеграл в (5) сходится при $k > -1$. Способ доопределения функции ФД для индекса $k < -1$ будет показан далее. Заметим, что при целых отрицательных k функции ФД имеют полюс.

В физических задачах нужны только целые и полуцелые индексы и вещественные значения x. Вычислению функций таких индексов в основном посвящена данная книга. Однако многие приведенные далее выражения справедливы для произвольных индексов k.

В квантово-механических моделях атома возникает ещё одна специфическая функция, связанная с вычислением обменной энергии в квазиклассическом приближении. Её называют интегральной функцией ФД:

(1.6)

Её свойства и способы вычисления также будут рассмотрены в этой книге.

§1.2. История

Свойствам функций ФД посвящено лишь несколько работ. Дадим их обзор.

Функции ФД впервые появились на заре развития квантовой механики в работах Паули [1] и Зоммерфельда [2] при описании частично вырожденного электронного газа в металлах. Основные свойства этих функций изложены в статье МакДугала и Стоунера [3]. В ней приведены сходящийся ряд при $x < 0$ и асимптотическое разложение при $x \to +\infty$. Заметим, что в асимптотическом разложении в формуле (5.3) имеется опечатка. Вместо сомножителя $(k - r + 2)$ следует писать $(k - 2r + 2)$.

В работе Калиткина [4] построен ряд, сходящийся при любых значениях x. Однако при $x > 3$ скорость сходимости невелика и фактически этот ряд удобен лишь при $-\infty < x < 1$.

Улучшенное выражение для асимптотического разложения получено в работе Glasser [5]; в нем содержится связь между значениями $I_k(x)$ и $I_k(-x)$.

Интегральную функцию ФД ввел Киржниц [6]. Ее разложение в сходящийся ряд при $x < 0$ и асимптотический ряд при $x \to +\infty$ было построено в работах Каликина, Кузьминой, Луцкого и Колганова.

Ряды, построенные в указанных выше работах, позволяют вычислять функции ФД с высокой точностью при $x < 0$ и при $x > 50$. Остается решить вопрос о практическом вычислении функций ФД в диапазоне $0 < x < 50$.

Функции ФД в указанном диапазоне(и вообще при любом аргументе x) и при заданном k можно найти с точностью ошибок компьютерного округления, непосредственно вычисляя интеграл (1) по каким-либо квадратурным формулам на достаточно подробной сетке. Правда такое вычисление по обычным классическим квадратурным формулам чрезмерно трудоемко и его нецелесообразно использовать как компьютерную подпрограмму. С его помощью можно составить подробные многозначные таблицы этих функций. Первый пример такой таблицы был приведен уже в [3]. Однако такие таблицы имеют огромный объем и также непригодны для создания компьютерных подпрограмм.

Правда, в последние годы для функций ФД полуцелого индекса Калиткиным и Колгановым разработаны экспоненциально сходящиеся квадратуры, радикально уменьшающие трудоемкость вычислений [7-9]. Они уже пригодны для создания попрограмм, хотя их трудоемкость ощутимо больше, чем для вычислений с помощью упомянутых выше рядов.

Поэтому остается актуальной задача построения быстрых компьютерных подпрограмм. Вот почему ряд работ был посвящен вычислению таблиц функций для конкретных индексов и построению для них несложных экономичных аппроксимаций. В них выбирают некоторый разумный вид аппроксимирующих формул с достаточным числом свободных параметров. Эти параметры подбирают для получения наилучшей точности аппроксимации в некоторой норме. Наиболее разумным

представляется получение наилучшей относительной точности в норме C.

Такие аппроксимации для функций целых и полуцелых индексов строились в цитированных работах. Для функций целых индексов аппроксимации с 16 верными десятичными знаками построены в [10], что в полной мере исчерпывает проблему при расчетах с 64-битовыми числами. Для функций полуцелых индексов опубликованные аппроксимации имеют меньшую точность.

В работе МакДугала и Стоунера [3] рассмотрены способы вычисления функций ФД индексов $k = 3/2, 1/2, -1/2$. Но эти способы ещё слишком сложны и трудоемки. Они непригодны для практического использования.

В работе Коди и Тетчера [11] для функций ФД индексов $k = 3/2, 1/2, -1/2$ построен набор аппроксимаций, составленных из трех кусков. Вид этих формул выбран удачно. При $x \leqslant +1$ аппроксимация является отношением двух многочленов от e^x; она точно передает главный член асимптотики при $x \to -\infty$. При $x \geqslant +4$ аппроксимация есть произведение множителя x^{k+1} на отношение многочленов от x^{-2}. Она точно передает главный член асимптотики при $x \to +\infty$. В промежутке $+1 \leqslant x \leqslant +4$ используется отношение двух многочленов одинаковой степени от x. Наборы с небольшим числом коэффициентов обеспечивают невысокую точность. Для набора с наибольшим числом коэффициентов заявлена относительная точность 10^{-12}. К сожалению, фактическая точность существенно хуже. В Табл.1 приведены относительные рассогласованности смежных формул на стыках. Они на много порядков превышают заявленную погрешность, особенно для $k = 3/2$. По-видимому, в табличных коэффициентах имеются опечатки. Поэтому на практике использовать эти формулы невозможно.

В работе Theiler [12] построены прецизионные аппроксимации для функций ФД индексов $k = 3/2, 1/2 - 1/2$. В ней при $x \leqslant -1$ используется классический ряд по степеням e^x. При $x \geqslant 30$ использован асимптотический ряд. Промежуток разбивается на 4 отрезка и на каждом отрезке табулированная функция аппроксимируется полиномами Чебышева. Окончательная относительная погрешность оценивается в 10^{-14}. Сама по себе такая точность хороша. Однако вид аппроксимации выбран менее удачно, чем в работе Коди и Тэтчера. Общее число коэффициентов аппроксимации из-за этого велико. Использование чебышевских многочленов не обеспечивает точность производных. Кроме того, на практике требуются функции и других полуцелых индексов.

Построить единую аппроксимирующую формулу высокой точности вряд ли представляется возможным.

Другой принцип построения формул был предложен в работах Калиткина и Кузьминой [13-15]. В них различные функции ФД выражались через функцию $I_{-1/2}(x)$. Это позволило построить аппроксимацию

из 2 кусков. При построении левой части использовались качественные соображения поведения при $x \to \infty$, а для правой части - асимптотики при $x \to +\infty$. Это позволило при небольшом числе коэффициентов получить относительную погрешность до $3\Delta 10^{-8}$ для функций полуцелых индексов и $7\Delta 10^{-6}$ для интегральной функции ФД.

§1.3. Погрешность округления

В данной работе рассматриваются методы прямого вычисления функций ФД целых и полуцелых индексов, а также интегральной функции ФД с заданной точностью. Желательно вычислять функции ФД с предельно высокой точностью ϵ, допускаемой компьютером. Поэтому опишем точность, которую дают распространенные сейчас процессоры при вычислении с плавающей точкой.

Все операции с плавающей точкой выполняются арифметическим сопроцессором (Floating Point Unit). Его архитектура фактически не развивается с момента принятия в 1985 году стандарта IEEE-754 [ссылка], поэтому предельная разрядность чисел остается на уровне 80 бит. Однако в наиболее распространенных математических обеспечениях для записи в память используются либо 32-битовые числа (single precision), либо 64-битовые числа (double precision). В обоих случаях не полностью используются возможности линейки процессора. В начале 2000-х годов для языка C++ была возможность использовать 80-битовые числа (long double precision); в настоящее время эта возможность не поддерживается и мало где сохранилась. Для суперкомпьютерных вычислений используют 128- битовые числа; но вычисления делаются программными средствами на тех же процессорах. Возможно и вычисление с произвольной разрядностью, но они также выполняются программными средствами. Например, для языка C++ это библиотека boost::multiprecision.

При вычислениях с плавающей точкой побитовая запись числа выглядит следующим образом: знак числа – 1 бит, двоичный порядок числа – p бит, мантисса – m бит. Порядок числа может быть положительным или отрицательным; но при записи в память к нему автоматически добавляется положительная константа, равная по модулю максимально возможному отрицательному порядку, так что в память записывается неотрицательное число. Это эквивалентно тому, что из p разрядов порядка тратится 1 разряд на знак порядка и $p-1$ на модуль порядка.

При записи мантиссы также имеется небольшая "хитрость". Первый разряд мантиссы всегда равен 1, поэтому при записи он отбрасывается. При считывании числа в процессор этот разряд автоматически добавляется. Такой прием позволяет фактически удлинять записываемую мантиссу на 1 бит.

Нетрудно посчитать, что максимально возможный порядок в двоичной системе есть $2^{p-1}-1$; для перевода в десятичную систему его нужно умножить на $lg2$. Разрядность процессора длиннее, чем считанное из памяти число, за исключением long double. После вычисления результат, записанный на процессоре, оказывается длиннее отведенного в памяти места. Поэтому при записи результата в память производится округление. Ошибка такого округления не превышает половины последнего отброшенного разряда. С учетом "спрятанного"первого разряда мантиссы это означает, что относительная ошибка округления есть $\epsilon = 2^{-m-2}$.

Таблица 1.1: Таблица 1. Компьютерные числа с плавающей точкой

точность	бит	p	m	P	$\lg \epsilon$
float	32	8	23	± 38	-7.5
double	64	11	52	± 308	-16.2
long double	80	15	64	± 4932	-19.3
Quadruple double	128	15	112	± 4932	-34.3

В случае long double длина числа равна линейке процессора и нельзя ни спрятать первый разряд мантиссы, ни иметь лишние разряды для округления; в этом случае $\epsilon = 2^{-m}$. В Таблице 1 приведены предельные порядки $P = 2^{p-1} lg 2$ и относительные ошибки единичного округления в форме $lg\epsilon$ для рассмотренных выше случаев вычисления. Видно, что для случая long double представление числа выбрано не вполне удачно; лучше было бы взять $p = 14, m = 65$.

Наиболее частыми являются 64-битовые вычисления, поэтому мы будем ориентироваться на точность 16^10. Вычисление long double кажутся заманчивыми, поскольку они повышают точность по сравнению с double без увеличения машинного времени. Однако разбиение числа на мантиссу и порядок было недостаточно продумано (лучше было бы перекинуть 1 бит на мантиссу), так что увеличение точности не столь значительно. Но 32-битовыми числами не стоит пренебрегать: именно такую точность дают видеокарты, позволяющие сильно ускорять вычисления за счет конвейерной реализации.

Глава 2

СВОЙСТВА ФУНКЦИЙ ФЕРМИ-ДИРАКА

§2.1. Точное решение

Существует единственный индекс $k = 0$, когда интеграл (5) берется в элемен- тарных функциях. Делая в интеграле (5) замену переменных $\tau = \exp t$, легко получаем

$$I_0(x) = ln(1 + e^x). \tag{2.1}$$

Для этой функции выполняется важное соотношение, связывающее функцию положительного и отрицательного аргументов:

$$I_0(x) = I_0(-x) + x, x > 0. \tag{2.2}$$

В его справедливости нетрудно убедиться, подставляя 2.1 в 2.2.

Из 2.1 нетрудно получить асимптотическое поведение это функции: $I_0(x) = e^x$ при $x \to -\infty$, $I_0(x) = x$ при $x \to +\infty$. Видно, что левая и правая асимптоты качественно отличаются друг от друга.

Все прочие функции ФД при $k! = 0$ в элементрных функциях не выражаются. Для них необходимо разрабатывать алгоритмы, имеющие хорошую точность и экономичность. Эти алгоритмы основываются на различных свойствах функций ФД. Поэтому рассмотрим их основные свойства.

§2.2 Связь функций соседних индексов

Пусть $k > 0$. Вычислим производную функции ФД, дифференцируя (1) по :

$$I_k^{'}(x) = \int_0^\infty \frac{d}{dx}\left(\frac{1}{1 + \exp(t - x)}\right) t^k dt \tag{2.3}$$

Поскольку дробь в скобках зависит только от выражения $t-x$, то дифференцирование этой дроби по x эквивалентно дифференцированию этой дроби по t со знаком "минус":

$$I_k^{'}(x) = -\int_0^\infty \frac{d}{dt}\left(\frac{1}{1 + \exp(t - x)}\right) t^k dt \tag{2.4}$$

Берем получившийся интеграл по частям:

$$I_k^{'}(x) = -\frac{t^k}{1 + exp(t - x)}\Big|_0^\infty + \int_0^\infty \frac{d}{dt}(t^k)\frac{dt}{1 + exp(t - x)} \tag{2.5}$$

Поскольку $k > 0$, выражение перед интегралом обращается в нуль при $t = 0$ и при $t \to +\infty$. Раскрывая в последнем интеграле d/dt, получим:

$$I_k^{'}(x) = k\int_0^\infty \frac{t^{k-1}dt}{1 + \exp(t - x)}dt = kI_{k-1}(x) \tag{2.6}$$

Это соотношение связывает производную функции ФД с функцией ФД на единицу меньшего индекса.

Соотношение 2.6 было получено для $k > 0$; при этом в правой части стоит функция с индексом больше -1. Будем считать это соотношение справедливым для любых значений k. Тогда оно доопределяет функции ФД нецелых индексов $k < -1$: оно выражает эти функции через производные функции ФД на единицу большего индекса. Очевидно, для получения функции ФД с индексом $-1 < k < 0$ надо продифференцировать функцию с положительным индексом $0 < k < 1$. Для нахождения функции ФД с индексом $-2 < k < -1$ надо продифференцировать функцию $-1 < k < 0$, т.е. дважды продифференцировать функцию с положительным индексом $0 < k < 1$. Это рекуррентный процесс, т.е. для каждого уменьшения индекса на единицу требуется лишний раз продифференцировать функцию с положительным индексом. Поэтому следует ожидать, что численное нахождение функции ФД с большими отрицательными индексами будет связано с существенной потерей точности.

При $k = 0$ соотношение (2.6) теряет смысл: слева стоит производная от (2), которая положительна и конечна при любом конечном x. Это означает, что $I_{-1}(x) = \infty$ при любом значении x. Соответственно, не существует $I_k(x)$ при любом целом $k < 0$.

§2.3 Ряд для $x < 0$

Пусть $x > 0$. Поскольку в (5) под интегралом $t > 0$, то $\exp x - t < 1$ Тогда в подынтегральном выражении (1) можно преобразовать дробь в сходящийся ряд:

$$\frac{1}{1+e^{t-x}} = \frac{e^{x-t}}{1+e^{x-t}} = \sum_{n=1}^{\infty}(-1)^{n-1}e^{n(x-t)}. \tag{2.7}$$

Умножим последнее выражение на t^k и проинтегрируем. Интеграл от $t^k exp(-nt)$ выражается через $(k+1)$. В итоге получаем следующий ряд [Stoner, McAougal]:

$$I_k(x) = \Gamma(k+1)\sum_{n-1}^{\infty}(-1)^{n-1}\frac{e^{nx}}{n^{k+1}}, x < 0. \tag{2.8}$$

Этот ряд сходится при любых $x < 0$, но сходимость неравномерная: она быстро ухудшается при $x \to -0$. Если $x = 0$, то ряд (5) сходится для индексов $k > -1$, но сходимость при этом крайне медленная. При $x > 0$ ряд расходится для любого k.

Замечание. Почленное дифференцирование ряда (5) с индексом k точно дает ряд для индекса $k - 1$.

§2.4. Всюду сходящийся ряд

Этот ряд был предложен в [Калиткин 68]. Применим к подынтегаль-

ному выражению (1) следующее преобразование:

$$\frac{1}{1+e^{t-x}} = \frac{2e^{-t}}{(1+2e^{-x})-(1+2e^{-t})} = \frac{2e^{-t}}{1+2e^{-x}}\left(1-\frac{1-2e^{-t}}{1+2e^{-x}}\right)^{-1} = \frac{2e^{-t}}{1+2e^{-x}}\sum_{m=0}^{\infty}\left(\frac{1-2e^{-t}}{1+2e^{-x}}\right) \tag{2.9}$$

При допустимых значениях переменных $(-\infty < x < +\infty, 0 < t < +\infty)$ разложение в ряд здесь всегда сходится, так как $|(1-2e^{-t})/(1+2e^{+x})| < 1$. Подставляя последнее разложение в интеграл (1), получаем ряд, сходящийся при любых x:

$$I_k(x) = 2\Gamma(k+1)\sum_{n=0}^{\infty}\frac{b_n^{(k)}}{(1+2e^{-x})^{n+1}}, \tag{2.10}$$

где

$$b_n^{(k)} = \frac{1}{\Gamma(k+1)}\int_0^{\infty}(1-2e^{-t})^n e^{-t}t^k dt = \sum_{p=0}^{n}\frac{(-1)^p 2^p n!}{p!(n-p)!(p+1)^{k+1}}; b_0^(k) = 1. \tag{2.11}$$

Сумма в (2.11) получается раскрытием скобки в подынтегральном выражении по формуле бинома и выражением получившихся интегралов через Γ -функцию.

В подынтегральном выражении для $b_n k$ стоит скобка $(1-2e^{-t})$, которая по модулю не превосходит 1; поэтому всегда $|b_n^{(k)}| \leqslant 1$. Следовательно, ряд (8) сходится при любом x не медленней, чем геометрическая прогрессия со знаменателем $(1+2e^{-x})^{-1}$. На любом ограниченном справа полубесконечном интервале эта сходимость будет равномерной. Практически при $x \leqslant 0$, и даже $x \leqslant 1$, сходимость достаточно быстрая, и ряд (8) удобен для прямых вычислений функций ФД. Этим ряд (8) удобнее ряда (5), который практически пригоден лишь при $x \leqslant -1$. Однако при $x > 1$ скорость сходимости ряда (8) быстро ухудшается.

Отметим одно качественное отличие ряда (8) от ряда (5). Почленное диффиренцирование ряда (8) для индекса k не дает такого же ряда для индекса $k = -1$.

Коэффициенты ряда. Вычисление $b_n^{(k)}$ с помощью суммы (9) легко выполняется для первых членов ряда:

$$b_0^{(k)} = 1; b_1^{(k)} = 1 - 2^{-k}. \tag{2.12}$$

Однако для больших n так вычислять коэффициенты из суммы (9) неудобно. Слагаемые суммы знакопеременны. Члены суммы содержат в себе биномиальные коэффициенты. При больших n центральные коэффициенты на много порядков больше крайних коэффициентов. Поэтому суммирование знакопеременных членов в (9) приводит к потере точности. Уже при n 10 теряется много значащих цифр, а при n 20 потеря точности становится огромной.

Существует способ вычисления $b_n^{(k)}$ без потери точности. Справедливы следующие рекуррентные соотношения, выражающие коэффициенты для индекса k через коэффициенты для индекса $k-1$:

$$b_0^{(k)} = 1; b_n^{(k)} = \frac{1}{n+1}(b_n^{(k-1)} + nb_{n-1}^{(k)}), n > 1. \tag{2.13}$$

Равенство (11) проверяется интегрированием (9) по частям. Предполагая $k > 0$ и умножая все на $\Gamma(k+1)$, проинтегрируем сомножитель e^{-t}:

$$\Gamma(k+1)b_n^{(k)} = -(1-2e^{-t})^n e^{-t}t^k\Big|_0^\infty + \int_0^\infty e^{-t}\frac{d}{dt}[(1-2e^{-t})^n t^k]dt. \tag{2.14}$$

Первое слагаемое равно 0. Преобразуем подынтегральное выражение:

$$e^{-t}\frac{d}{dt}[(1-2e^{-t})^n t^k] = ne^{-t}(1-2e^{-t})^{n-1}(2e^{-t})t^k + k(1-2e^{-t})^n t^{k-1} \tag{2.15}$$

Выражение $(2e^t)$ в первом слагаемом преобразуем так: $(2e^t) = 1 - (1 - 2e^{-t})$. Тогда последний интеграл выражается через коэффициенты следующим образом:

$$n\Gamma(k+1)b_{n-1}^{(k)} - n\Gamma(k+1)b_n^{(k)} + k\Gamma(k)b_n^{(k-1)}. \tag{2.16}$$

Приравнивая полученное выражение величине $\Gamma(k+1)b_n^{(k)}$, доказываем формулу (11).

Поскольку $|b_n^{(k)}| \leqslant 1$, а коэффициенты в правой части (11) суммируются с положительными весами $1/(n+1)$ и $n/(n+1)$, причем сумма весов равна 1, то вычисления по формуле (11) будут устойчивыми. Заметим, что если для некоторого $k-1$ коэффициенты $|b_n^{(k-1)}| \geqslant 0$ при всех n, то коэффициенты $b_n^{(k)}$ также будут неотрицательны. Поскольку при увеличении k на 1 вычисления устойчивы, то можно далее снова увеличить k на 1 и повторить этот процесс неограниченное количество раз, не теряя точности.

Заметим, что, рекуррентно суммируя (11) по n, получаем следующее выражение:

$$b_n^{(k)} = \frac{1}{n+1}\sum_{p=0}^{n} b_p^{(k-1)} \tag{2.17}$$

Вычисление по формуле (12) также устойчиво.

Практически нас интересуют только целые или полуцелые индексы. Для $k = 0$ коэффициенты вычисляются точно:

$$b_n^{(0)} = \frac{1+(-1)^n}{2(n+1)}, n \geqslant 0; \tag{2.18}$$

эта формула легко получается из интеграла (9) заменой переменных $\tau = (1 - 2e^{-t})$. Заметим, что все нечетные коэффициенты (13) равны 0, а четные положительны. Для других целых индексов $k > 0$ коэффициенты вычисляются рекуррентным применением (11). Видно, что все они будут положительными. В принципе, коэффициенты для целых k выражаются через обыкновенные дроби, но эти выражения чрезмерно громоздки.

Для полуцелых индексов k наиболее удобно начать с индекса $k = -1/2$. В этом случае в интеграле (9) удобно сделать замену $t = \tau^2$. Получившиеся интегралы вычисляются по квадратурным формулам. По найденным коэффициентам и рекуррентным формулам (11) легко вычисляются значения коэффициентов для более высоких полуцелых индексов.

Индекс $k = -3/2$. Он требуется в практических приложениях. Формула (11) очевидным образом обращается в сторону уменьшения индекса:

$$b_0^{(k)} = 1; b_n^(k-1) = (n+1)b_n^{(k)} - nb_{n-1}^{(k)}, n > 1. \tag{2.19}$$

Однако эта формула содержит вычитание с большими весами, и вычисления по ней неустойчивы. Выгоднее воспользоваться дифференциальным соотношением (4), полагая в нем $k = -1/2$, и продифференцировать ряд (8) для этого индекса почленно. Это дает

$$I_{-\frac{3}{2}}(x) = -8\sqrt{\pi}e^{-x}\sum_{n=0}^{\infty}\frac{(n+1)b_n^{(-1/2)}}{(1+2e^{-x})^{n+2}} \tag{2.20}$$

Этот ряд также сходится при любых x, а вычисления по этой формуле устойчивы.

Коэффициенты для всех целых и полуцелых индексов от $k = -1/2$ до $k = 4$ приведены в Таблице 1 в виде десятичных дробей. Число значащих цифр рассчитано на получение относительной точности $\epsilon = 10^{-16}$. Число коэффициентов достаточно для вычисления функций ФД при $x \leqslant 1$ с указанной точностью. Для значений $x \leqslant 0$ достаточно суммировать до $n = 35$.

§2.5. Изменение знака аргумента

В [4] получено фундаментальное соотношение, связывающее функции ФД произвольного индекса $k > -1$ от положительного и отрицательного аргументов. Запишем его в несколько преобразованном виде, предполагая $x > 0$:

(2.21)

здесь $\zeta-$ функция от четного аргумента выражается через числа Бернулли:

(2.22)

Заметим, что при сравнении данных формул с [4] надо учитывать одно обстоятельство: определение функции ФД в [4] отличается от (1) делителем $\Gamma(k+1)$.

Таблица 2.1: **Коэффициенты $b_n^{(k)}$ ряда №**

$n\backslash k$	$-1/2$	$1/2$	1
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			
12			
13			
14			
15			
16			
17			
18			
19			
20			
21			
22			
23			
24			
25			
26			
27			
28			
29			
30			
31			
32			
33			
34			

Представление (15) является асимптотическим. Это означает, что при фиксированном x и возрастании N, т.е. увеличении числа членов суммы, точность сначала возрастает до некоторого номера $N(x)$, а при дальнейшем добавлении членов начинает ухудшаться. Оптимальное $N(x)$ монотонно возрастает при $x \to +\infty$. Таким образом, представление (15) позволяет получать высокую точность при положительных $x >> 1$, но при умеренных $x > 0$ его точность невелика.

Нетрудно убедиться, что почленное дифференцирование разложения (15) дает разложение для функции ФД с индексом $k-1$. Однако оптимальное число членов $N(x)$ оказывается разным для функций соседних индексов.

В Таблице 2 приведены значения первых 12 чисел Бернулли и $\zeta-$функции. Такого количества достаточно для обеспечения относительной погрешности 10^{16} при аргументах $x \geqslant 50$.

Таблица 2 **Значения чисел Бернулли и $\zeta-$функции**

Целые индексы. Сам асимптотический ряд (15) был получен ещё в ранних работах [3]. Однако там ещё не была обнаружена связь с функциями от отрицательного аргумента. Для функций произвольного индекса последнее обстоятельство несущественно, поскольку точность асимптотического ряда хороша лишь при больших значениях x, когда величина $I_k(-x)$ весьма мала.

Однако для целых индексов $k \geqslant 0$ ситуация кардинально меняется. В этом случае в произведении по p сомножитель со значением $p = 2k + 2$ равен нулю. Тем самым, сумма по n содержит лишь конечное число слагаемых вплоть до $N = 1 + [k/2]$, где квадратные скобки обозначают целую часть числа. Поэтому соотношение (15) принимает следующий вид:

(2.23)

Самое поразительное то, что для целых индексов соотношение (17) является не асимптотическим, а точным! В этом можно убедиться следующим образом. Для функции $I_0(x)$ оно совпадает с соотношением (3), которое является точным. Полагая $k = 1$ в (4), подставляя в правую часть (3) и интегрируя, получим для функции $I_1(x)$ соотношение между функциями положительного и отрицательного аргументов, которое также оказывается точным. Последовательно повышая таким интегрированием индекс k на единицу, убедимся в том, что соотношение (17) является точным для всех функций целого индекса.

Приведем такие соотношения для нескольких первых индексов:

(2.24)

Этих значений k достаточно для практических приложений.

§2.6. Интегральная функция ФД

Эта функция определена соотношением (6). Исследуем свойства этой функции.

Разложение при $x < 0$. Запишем разложение (5) для индекса $k = -1/2$:

(2.25)

Возводя этот ряд в квадрат и группируя члены с одинаковыми экспонентами, получим

(2.26)

Почленно интегрируя полученный ряд от $-\infty$ до x, получим искомое разложение:

(2.27)

Это разложение сходится при $x < 0$ и расходится при $x > 0$. Сходимость тем быстрее, чем больше модуль x. Ряд (20) знакопеременный. Фактически он пригоден для вычисления функции (19) при $\leqslant -1$, как и ряд (5). Значения коэффициентов a_n приведены в Таблице 3.

Всюду сходящийся ряд. Напишем всюду сходящийся ряд (8) для $k = -1/2$:

(2.28)

Возводя этот ряд в квадрат и группируя члены с одинаковыми знаменателями, получим

(2.29)

Для получения $J()$ надо проинтегрировать (21) от $-\infty$ до x. Будем искать интеграл в виде аналогичного разложения с неизвестными коэффициентами c_n:

(2.30)

Продифференцируем (22) по x и проведем следующие преобразования:

(2.31)

Сдвигая во второй сумме индекс n на 1, получим

(2.32)

Сопоставляя последнее выражение с (21) и приравнивая коэффициенты при одинаковых степенях знаменателя, найдем следующее соотношение между коэффициентами:

(2.33)

Таким образом, коэффициенты формулы (22) определяются из рекуррентных соотношений

(2.34)

Значения коэффициентов n с приведены в Таблице 3.

Формулы (22)-(23) определяют ряд, сходящийся при любых значениях x. Скорость его сходимости достаточно хороша при $x \leqslant 0$, удовлетворительна при $x \leqslant 1$, но быстро ухудшается при $x > 1$.

Разложение при $x \to +\infty$. Запишем асимптотическое разложение (15) для произвольного k в более удобном виде:

(2.35)

Положим здесь $k = -1/2$; тогда $cos(\pi k) = 0$, и остается только сумма. Возведем эту сумму в квадрат и опять сгруппируем члены по одинаковым степеням . Суммирование по n идет не до бесконечности, а до N. Однако мы будем пользоваться разложением (24) только в том случае, если последние члены суммы достаточно малы по сравнению с главными. Тогда можно приближенно записать

(2.36)

Почленно проинтегрируем (25) по x от $-\infty$ до x, учитывая значения $C_0 = 1, A_1(-1/2) = -\pi^2/24, C_1 = -pi^2/12$. Получим следующую сумму:

(2.37)

где j есть константа, возникающая при интегрировании. Значение этой константы приведено в [6-7]:

(2.38)

где $= 0.5772156649015325...$ – константа Эйлера.

Медленно сходящуюся сумму (27) необходимо вычислить с точностью $\epsilon\ 10^{-16}$. Непосредственное суммирование на компьютере требует неприемлемо большого числа членов ряда. Поэтому воспользуемся следующим приемом. Разобъем бесконечную сумму на две:

(2.39)

Первую сумму вычислим непосредственно; при этом суммировать будем с последнего члена, т.к. проведение суммирования в порядке увеличения слагаемых уменьшает ошибки округления. Вторую сумму рассмотрим, как квадратуру средних для интеграла от функции $n^{-2}ln(n)$ в пределах $N + 1/2 \leqslant n \leqslant +\infty$ с шагом $\delta n = 1$. Сам интеграл легко вычисляется точно заменой переменных $\xi = \ln n$ и равен:

(2.40)

Для повышения точности добавим к формуле средних поправки Эйлера-Маклорена, содержащие первую и третью производные подынтегральной функции на левой границе (очевидно, эти поправки на правой границе обращаются в нуль). Получим следующее выражение:

(2.41)

Таблица 2.2: **Коэффициенты разложений функции J(x)**

n	a_n	c_n	C_n
0	0.50000000000000000	0.50000000000000000	1.00000000000000000
1	0.47140452079103173	0.05719095841793650	-0.82246703342411309
2	0.41367513459481287	0.32627341363306145	-3.38226010534730559
3	0.36329931618554523	0.05555337454026419	-56.74866767632004638
4	0.32247788425329943	0.24658846860286468	-2076.439816971693289
5	0.28947176887871823	0.05073748578641944	-133516.6239190830092
6	0.26245962433780035	0.20002927599276679	-13363920.49546855688
7	0.24002748681137623	0.04624864575737019	-1924202279.429788351
8	0.22113507376025146	0.16919747074124367	-376996608458.5720214
9	0.20502010799708259	0.04243339943502982	
10	0.19111818527221525	0.14714269473929523	
11	0.17900522098810093	0.03922350305150957	
12	0.16835758850494667	0.13051578228471081	
13	0.15892459422501767	0.03650451326932291	
14	0.15050930279877994	0.11749388727681387	
15	0.14295500139265471	0.03417682128363331	
16	0.13613550394070195	0.10699573032743079	
17	0.12994810323070083	0.03216234704554120	
18	0.12430837457100294	0.09833733348435542	
19	0.11914629284901028	0.03040122425575591	
20	0.11440329429976415	0.09106388018262407	
21	0.11003002691122073	0.02884749082202021	
22	0.10598460916982465	0.08486060243799427	
23	0.10223126852657950	0.02746553440048496	
24	0.09873926667442291	0.07950240515223343	
25	0.09548204372438338	0.02622743720392143	
26	0.09243653108215864	0.07482385935080225	
27	0.08958259552832193	0.02511104569452402	
28	0.08690258621471895	0.07070053682482372	
29	0.08438096303764077	0.02409857188256913	
30	0.08200398984235426	0.06703696772323861	
31	0.07975947964440647	0.02317557364450819	
32		0.06375862474843785	
33		0.02233020393904351	
34		0.06080644154622106	

Следующая поправка Эйлера-Маклорена есть $O(N^{-7})$; чтобы она не превышала 10^{-16}, достаточно взять $N = 300$. Численный расчет с этими значениями даёт

$$\sum ln = 0.93754825431584388, j = 0.76740941382814898. \tag{2.42}$$

Коэффициенты C_n для $n \geqslant 2$ приведены в Таблице 3. Разложение (26) имеет асимптотическую сходимость, так что суммировать по n можно только до тех пор, пока члены суммы достаточно быстро убывают. Определение оптимального числа членов N является самостоятельной проблемой.

Глава 3

ФУНКЦИИ ЦЕЛОГО ИНДЕКСА

§3.1. Нулевой индекс.

Напомним, что функции ФД $I_k(x)$ целого индекса k существуют при $k \geqslant 0$. При $k = 0$ функция ФД выражается по формуле (2.1): $I_0(x) = log(1 + e^x)$. При целых $k < 0$ функция ФД не существует, так как интеграл (1.5) расходится.

§3.2 Отрицательные аргументы

Классический ряд (2) пригоден для практического вычисления функций при $x \leqslant -1$. Этот ряд знакопеременный, так что ошибка не превышает первого отброшенного слагаемого. Обычно требуется обеспечить некоторую не абсолютную, а относительную точность ε. Главный член суммы в (5) есть e^x. Если в сумме оставляется N слагаемых, то относительная величина отброшенного члена есть $e^{-Nx}/(N+1)^{k+1}$. Поэтому для заданного надо выбирать N из следующего условия:

$$\frac{e^{-Nx}}{(N+1)^{k+1}} \leqslant \varepsilon. \tag{3.1}$$

Можно определить N из уравнения 3.1 каким-либо итерационным процессом. Но при написании программы можно поступить проще: заранее составить таблицу границ , x_N удовлетворяющих уравнению 3.1, и далее отслеживать попадание требуемого значения в эти границы.

Когда N определено, то суммировать отрезок ряда удобнее всего по схеме Горнера:

$$I_k(x) = \Gamma(k+1)e^x\left(\frac{1}{1^{k+1}} - e^x\left(\frac{1}{2^{k+1}} - e^x\left(\frac{1}{3^{k+1}} - e^x\left(...- e^x\left(\frac{1}{N^{k+1}}\right)\right)\right)\right)\right). \tag{3.2}$$

Для 64-битовых вычислений следует полагать $\varepsilon = 10^{-16}$. Тогда для $x = -1$ число слагаемых составляет $N = 37 - 40$; оно слабо зависит от k и слегка убывает при увеличении k. При $x \to -\infty$ число слагаемых N быстро убывает.

Таким образом, это достаточно экономичный способ прямого вычисления функций ФД при $x \leqslant -1$. Однако, при этом остается нерешенной ещё одна проблема: как вычислять функции на отрезке $-1 \leqslant x \leqslant 0$?

Всюду сходящийся ряд. Для произвольных, необязательно целых, индексов k существует всюду сходящийся ряд 2.10. Коэффициенты этого ряда вычислены и приведены в Табл 2.1. Этот ряд сходится при любом x не хуже, чем геометрическая прогрессия со знаменателем $g = (1 + 2e^{-x})^{-1}$. Эта сходимость неравномерная, и при $x \to +\infty$ становится очень медленной. Однако при $x \leqslant 0$ знаменатель $g \leqslant 1/3$, и сходимость становится достаточно быстрой. Таким образом, ряд 2.10 удобен для вычисления функций ФД при $x \leqslant 0$.

Ограничимся конечным числом членов ряда (3) и запишем полученную сумму по схеме Горнера:

$$//gh \tag{3.3}$$

напомним, что для целого индекса $\Gamma(k+1) = k!$. Значения коэффициентов $b_n^{(k)}$ можно непосредственно рассчитывать по формулам (5-6) или брать из Таблицы 2.1, где они приведены с 16-ю десятичными знаками. Такого числа знаков достаточно для вычисления с относительной точностью $\varepsilon = 10^{-16}$ (double precision). Описанный алгоритм прост и экономичен. Поскольку коэффициенты b_n^k, то нигде не возникает вычитаний, и относительные ошибки округления могут накапливаться лишь незначительно, практически не ухудшая точность.

Определим требуемое число членов. Нас интересует относительная погрешность. Поэтому общий множитель $2\Gamma(k+1)g$ можно откинуть. Главный член оставшегося произведения есть $b_0^{(k)} = 1$ и все остальные члены можно сравнивать непосредственно с ним. Все коэффициенты $b_n^{(k)} < 1$, положительны и убывают с увеличением номера n. Это означает, что отброшенная часть ряда сходится быстрее, чем геометрическая прогрессия с первым членом g^{N+1} и знаменателем g . Величина знаменателя возрастает от 0 до 1/3 при возрастании от $-\infty$ до 0. Поэтому сумма отброшенных членов не превышает сумму этой геометрической прогрессии $g^{N+1}/(1-g)$. Нужно, чтобы эта величина не превышала ε .

Эту оценку нетрудно немного усилить. На самом деле величину откинутых членов ряда надо сравнивать не с $b_0^{(k)} = 1$, а с суммой учтенных членов ряда. Их также можно оценить, как сумму геометрической прогрессии с первым членом 1 и знаменателем g. Тогда для сравнения вместо ε надо брать величину $\varepsilon/(1-g)$. Это дает следующую оценку:

$$N \geqslant \frac{ln\varepsilon}{lng} - 1. \tag{3.4}$$

Эта формула дает нецелое значение N. И его надо округлить вверх до целого:

$$N = [\frac{ln\varepsilon}{lng}], \tag{3.5}$$

где квадратные скобки означают целую часть числа. Даже эта оценка завышена, т.к. она не учитывает заметного убывания $b_n^{(k)}$ при возрастании номера n. Однако учет этого эффекта заметно усложнил бы алгоритм, что нецелесообразно.

Наиболее медленная сходимость будет при $x = 0$, когда $\varepsilon = 10^{-16}$; для $g = 1/3$ это дает $N = 33$. При $x \to -\infty$ величина g быстро стремится к 0; при этом число членов N быстро убывает.

§3.3. Положительный аргумент

Для функций ФД существует точное соотношение, связывающее значения функции при положительном и отрицательном аргументах (2.23). Для индексов $k \leqslant 4$ выше были приведены конкретные реализации (2.24) общей формулы. Поскольку для $x \leqslant 0$ алгоритм вычисления построен

выше, эти формулы полностью решают вопрос вычисления функций с целым индексом.

Однако построенные выше ряды для отрицательных значений аргумента при x 0 требуют суммирования большого числа членов ряда, т.е. сравнительно трудоемки. В Главе № будут построены аппроксимации, существенно снижающие трудоемкость при $x \leqslant 0$.

Обсудим точность вычислений при $x > 0$. Функции отрицательного аргумента мы вычисляем с относительной точностью ε. Полиномиальная часть формул (2.23-2.24) вычисляется практически точно: относительная ошибка не превышает ошибки единичного округления. Сами функции ФД при $k > -1$ являются монотонно возрастаюшими по x. Поэтому полиномиальная часть формул (2ю23-2.24) превышает вклад функций отрицательного аргумента тем сильнее, чем больше x. Тем самым, при $x > 0$ относительная погрешность описанного алгоритма будет заведомо меньше ε.

Глава 4

ЭКСПОНЕНЦИАЛЬНО СХОДЯЩИЕСЯ КВАДРАТУРЫ

§4.1. Проблема трудоемкости квадратур

Прямое вычисление функций ФД нецелого индекса требует применения квадратурных формул к интегралу (1.5). При произвольных индексах k (в том числе и целых!) такое интегрирование для получения высокой точности (16 верных десятичных знаков) требует очень подробной сетки ($10^5 - 10^6$ узлов) и является чрезмерно трудоемким. Однако для полуцелых индексов k возможно кардинальное уменьшение трудоемкости при специальном преобразовании подынтегрального выражения. Можно построить квадратуры с экспоненциальной, то есть очень быстрой сходимостью.

Квадратуры с экспоненциальной сходимостью были предложены и исследованы в [ссылки на работу 2014 года, наши работы]. В [ссылка на статью SIAM] было рассмотрено интегрирование периодической функции $u(x)$ на ее периоде $[0, 2\pi]$ по формуле трапеций на равномерной сетке с N интервалами. Была доказана следующая

Теорема 1. Пусть $u(x)$ есть функция, аналитическая в полуплоскости $Imx \geqslant -a, a > 0$. Тогда справедлива следующая мажорантная оценка погрешности формулы трапеций:

$$|\delta| \leqslant \frac{2\pi M}{[exp(aN) - 1]}, \tag{4.1}$$

где $N = max|u(x)|$ на отрезке интегрирования. TODO:добавить жирный кружок в конце Th. Дадим комментарий. Неаналитичность функции $u(x)$ ниже указанной полуплоскости означает, что налинии $Im(x) = -a$ лежит по меньшей мере одна особая точка функции. Тогда, в силу периодичности $u(x)$, эти особые точки будут расположены на линии $Imx = -a$ также с периодом 2π. Тогда, одна из этих особых точек будет лежать непосредственно под вещественным отрезком интегрирования. Тем самым, a есть расстояние от отрезка интегрирования до ближайшей особой точки.

Интуитивно ясно, что теорему можно усилить. По-видимому достаточно требовать аналитичности $u(x)$ в полосе $-a \leqslant Imx \leqslant a$. Кроме того, для $u(x) = const$, погрешность квадратур трапеций равна 0. Это значит, что вычитая из $u(x)$ константу, мы не меняем фактической погрешности. Поэтому оценку постоянной M можно немного улучшить. Например, $u(x)$ вещественно на вещественной оси, то $M = |\max u + \min u|/2$ на вещественном отрезке интегрирования.

Экспоненциальная сходимость гораздо быстрее степенной. Поэтому такие квадратуры могут обеспечить экономичность расчетов. Покажем, как можно построить экспоненциально сходящиеся квадратуры для функций ФД полуцелых индексов.

§4.2. Сходимость квадратур

Рассмотрим задачу вычисления интегралов от функций $u(x)$, имеющих сколь угодно высокие непрерывные производные на отрезке ин-

тегрирования $[a, b]$ (см. напр. [3-5]). Чаще всего на практике берут равномерные или сводящиеся к равномерным сетки $\omega_N = x_n, n = 0, ..., N$ и используют простейшие квадратурные формулы трапеций, средних, Симпсона и т.п. Погрешность подобных формул имеет оценку $const$, где p есть порядок точности формулы, h – шаг интегрирования, $TODO$ Такая сходимость называется **степенной**, поскольку погрешность выражается через степень шага. Она достаточно медленная, и для получения высокой точности требуется большое N . Такие квадратуры довольно трудоемки.

Квадратуры Гаусса-Кристоффеля дают гораздо более быструю сходимость. Например, классическая формула Гаусса для интегрирования на отрезке $[-1, 1]$ с весом $\rho(x) = 1$ имеет погрешность (после упрощения факториальных множителей)

$$\delta \leqslant \sqrt{\frac{\pi}{N}} \frac{b-a}{4} \left(e \frac{b-a}{8N} \right)^{2N} M_{2N}. \tag{4.2}$$

Квадратура Эрмита для отрезка $[-1, 1]$ с весом $\rho(x) = (1-x^2)^{-1/2}$ имеет погрешность

$$\delta \leqslant \sqrt{\frac{\pi}{N}} \left(\frac{e}{2\sqrt{2}N} \right)^{2N} M_{2N}. \tag{4.3}$$

Погрешности (11-12) с точностью до логарифмически малых членов можно записать в следующем виде:

$$\delta \sim \alpha \exp(-\beta N). \tag{4.4}$$

Зависимость от числа узлов является не степенной, а экспоненциальной, поэтому такую сходимость будем называть **экспоненциальной**.

Заметим, что формулы со степенной сходимостью порядка p требуют существования непрерывной производной лишь $p-$ого порядка подынтегральной функции независимо от числа узлов сетки N. Формулы Гаусса-Кристоффеля с N узлами требуют существования $2N-$й непрерывной производной, т.е. при увеличении N надо соответственно повышать гладкость функции.

Трудоемкость формул Гаусса-Кристоффеля несравненно меньше, чем у квадратур со степенной сходимостью. Однако узлы и веса этих квадратур найдены лишь для отдельных отрезков и весов $\rho(x)$ интегрирования. При этом только для квадратур Эрмита эти веса и узлы найдены в виде простых формул для произвольных N. Для остальных случаев узлы и веса точно вычисляются (через радикалы) лишь для $N \leqslant 3$ или $N = 5$. Это сильно ограничивает возможности практического использования таких квадратур.

Далее покажем, что если $u(x)$ четно продолжается через обе границы отрезка, то формула трапеций на равномерной сетке дает экспоненциальную сходимость. При этом коэффициент β в экспоненте определяется расстоянием до ближайшей особой точки в комплексной плоскости. Это открывает новые возможности для построения квадратур малой трудоемкости.

§4.3. Случай экспоненциальной сходимости

Пусть $u^{(p)}(x)$ существуют и непрерывны на $[a;b]$ при любых p. Требуется вычислить интеграл

(4.5)

Введем равномерную сетку *TODO* и воспользуемся формулой Эйлера–Маклорена, базирующейся на формуле трапеций [6,7]:

(4.6)

Если оборвать эту сумму на члене P , то первый отброшенный член будет остаточным. Его величина есть *TODO*. В этом случае формула (15) имеет степенную сходимость.

Пусть $u(x)$ такова, что все её нечётные производные на правой и левой границах одинаковы: $u^(2p-1)(a) = u^(2p-1)(b)$. Тогда в (15) сумма обращается в нуль. Оставшаяся часть квадратур является просто формулой трапеций. Из этого следуют

Утверждение 1. Пусть подынтегральная функция $u(x)$ имеет сколь угодно высокие производные, причем нечетные производные на правой и левой границах одинаковы: $u^(2p-1)(a) = u^(2p-1)(b)$. Тогда формула трапеций на равномерной сетке имеет сходимость выше степенной.
Частный случай. Утверждение 1 справедливо, если $u(x)$ чётно продолжается через обе границы отрезка: $u^(2p-1)(a) = u^(2p-1)(b) = 0$.

Таким образом, установлен класс функций, для которого формула трапеций имеет сверхстепенную сходимость. Остается найти закон этой сходимости. Проведем ее изучение на следующем тестовом примере:

$$U(q,r,c) = \int_0^\pi \frac{(c^2-1)c^r \cos(rx)}{(c^2-2c\cos x+1)^q} dx, c > 1. \tag{4.7}$$

Параметры $r \geqslant 0$, $q \geqslant 1$ берутся целыми. Тогда подынтегральное выражение четно на обеих границах отрезка, его нечетные производные на границах обращаются в нуль, и пример удовлетворяет требованиям Утверждения 1. При $q = 1$ известно точное значение интеграла [8]:

$$U(1,r,c) = \pi. \tag{4.8}$$

При $q! = 1$ интеграл (16) не выражается через элементарные функции от параметров.

Для тщательного численного выявления закономерностей все расчеты проводились с повышенной разрядностью (45 десятичных знаков) с помощью библиотеки языка C++ boost::multiprecision.

Расчеты интеграла (16) при фиксированных параметрах проводились на сетках с разным числом интервалов N. Погрешность расчетов при $q = 1$ определялась непосредственным сравнением с точным ответом (17). На рис.1 показана зависимость погрешности от N при $r = 0$ и различных значениях с в полулогарифмическом масштабе. Каждому значению с соответствует своя линия погрешности. Видно, что при всех значениях с кривые погрешности в этом масштабе являются прямыми. Это означает, что погрешность подчиняется закону

$$ln\delta_N = \alpha - \beta N, \beta = constlnc. \tag{4.9}$$

При других значениях параметров картина была аналогичной. На рис.2 показан случай $q = 1, r = 2$. Опять линии погрешности являются прямыми.

Для $q > 1$ точный ответ неизвестен. В этом случае для получения значений погрешности можно воспользоваться следующими соображениями. При закономерности (18) разности значений U при возрастании N на единицу также должны ложиться на прямую в полулогарифмическом масштабе (это напоминает апостериорную оценку погрешности по методу Ричардсона для квадратур со степенной сходимостью). На рис. 3 приведены графики таких разностей для $q = 2, r = 1$. Они также оказываются прямыми. Все это позволяет сделать эвристическое

Утверждение 2. При выполнении условий Утверждения 1 погрешность формулы трапеций экспоненциально зависит от числа узлов сетки N.

Попробуем выяснить, от чего зависит коэффициент β в (18). Он не должен зависеть от максимумов модулей каких-либо производных $u(x)$, поскольку они входят в суммы формул Эйлера-Маклорена (15) и приводят к степенной сходимости. Поэтому рассмотрим гипотезу о связи β с особыми точками подынтегрального выражения.

Если скобка в знаменателе подынтегрального выражения в (номер формулы теста) обращается в нуль, то подынтегральное выражение имеет полюс порядка q. Это происходит при

$$\frac{c^2 + 1}{2c} = cos(x) = \frac{e^{ix} + e^{-ix}}{2}. \tag{4.10}$$

Это уравнение имеет два решения: $e^{ix} = c$ или $e^{ix} = 1/c$. Следовательно, имеется две цепочки полюсов кратности q в точках

$$x^\star = 2\pi m \pm iln(c), -\infty \leqslant m \leqslant +\infty. \tag{4.11}$$

Из рис № видно, что наименьшее расстояние между каким-либо из полюсов и ближайшей к нему точкой отрезка интегрирования есть $ln(c)$.

Рис.1. Погрешность квадратуры трапеций для (18) при $p = 0$ и $q = 1$. Цифры около линий – величины с . Рис.2. Погрешность квадратуры трапеций для (18) при $p = 2$ и $q = 1$. Цифры около линий – величины с . Рис.3. Погрешность квадратуры трапеций для (16) при $p = 1$ и $q = 2$. Цифры около линий – величины с .

Предварительный просмотр графиков показал, что наклон β $ln(c)$. Для тщательного анализа на рис.4 показано отношение $\beta/ln(c)$ в зависимости от для нескольких значений $q = 1, 2$ и $r = 0, 1, 2$. Видно, что для полюса первого порядка ($q = 1$) при $r = 0$ это отношение с высокой точностью не зависит от и равно 2. При $r = 1$ это отношение равно 2 при $ln(c) \approx 0$; при увеличении с это отношение несколько уменьшается, причем линия в пределах графика блзка к прямой. При $r = 2$ линия также начинается в точке 2, но её наклон ещё немного увеличивается.

Для полюсов второго порядка ($q = 2$) линии для разных r начинаются не со значения 2, а с несколько меньшего значения 1.93. Наклоны линий с разными r также несколько больше, причем даже линия с $r = 0$ уже имеет наклон. Наименьшее отношение в пределах указанного графика 1.70.

Рис.4. Зависимость $\beta/ln(c)$ от величины с для различных p и q . Это позволяет сделать

Утверждение 3. Наклон β в (18) с хорошей точностью пропорционален расстоянию от отрезка интегрирования до ближайшего полюса интегрируемой функции в комплексной плоскости. Коэффициент пропорциональности достигает 2 в случае полюса первого порядка, и несколько уменьшается с увеличением кратности полюса и расстояния до него.

Практические рекомендации. 1. При использовании квадратурных формул со степенной сходимостью удобно сгущать сетки по N последовательно вдвое. Это позволяет использовать обычную процедуру Ричардсона для получения априорной асимптотически точной оценки погрешности. Такое сгущение экономично, поскольку суммарный объем всех расчетов лишь вдвое превышает объем расчетов на последней сетке [4].

Для квадратур с экспоненциальной сходимостью (18) также можно пользоваться процедурой Ричардсона, если сгущать сетки не вдвое, а каждый раз увеличивая N на 1. При этом будет получаться асимптотически точная апостериорная оценка погрешности. Однако такое сгущение сеток экономически невыгодно, поскольку суммарный объем вычислений будет в $\sim N/2$ раз больше, чем расчет на последней сетке.

Поэтому в практических расчетах удобнее увеличивать N в 2 раза. Из (18) нетрудно получить, что при этом $\delta_{2N} \sim \delta_N^2$. Такой закон убывания напоминает сходимость ньютоновских итераций вблизи простого корня: число верных десятичных знаков приблизительно удваивается с увеличением N в 2 раза. Поэтому на практике останавливаются на такой сетке $2N$, когда отклонение от результата на предыдущей сетке стано-

вится меньше $\varepsilon^{2/3}$, где ε – ошибка единичного округления компьютера.

2. Для формулы трапеций на равномерной сетке полезен следующий прием, вдвое уменьшающий трудоемкость вычислений. На сетке с N узлами и шагом h формула трапеций имеет вид

$$U_N = h(\frac{u_0}{2} + u_1 + ... + u_(N-1) + \frac{u_N}{2}). \tag{4.12}$$

При удвоении сетки все узлы предыдущей сетки становятся четными узлами новой сетки, и заново вычислять значения функций в них не надо. Достаточно найти значения функции в новых (нечетных) узлах и вычислить

$$U_{2N} = \frac{1}{2}U_N + \frac{h}{2}(u_1 + u_3 + u_5... + u_{2N-1}). \tag{4.13}$$

где нечетные индексы относятся к узлам новой сетки.

§4.4. Сравнение теории с эвристическими оценками

Теорема 1 с оценкой погрешности (4.1) и эвристические Утверждения 1-3, полученные из численных экспериментов, качественно близки. Однако между ними имеется ряд различий. Обсудим их.

Теорема 1 строго доказана. В ней рассмотрен интеграл от периодической аналитической функции, взятой по полному периоду. Получена мажорантная оценка погрешности, справедливая для любого типа особых точек, включая существенно особые.

При получении эвристических Утверждений 1-3 также неявно предполагалась аналитичность функции. Однако периодичность функции не предполагалась. Правда, в рассмотренном тесте функция была периодической, но интеграл брался только по половине периода. Поэтому на самом деле требовалось лишь равенство нечетных производных на концах отрезка интегрирования (такое обобщение существенно для практических применений). Однако в численных экспериментах рассматривался только простейший случай особых точек - полюса первого и второго порядков. Это более благоприятная ситуация, позволившая получить более сильные оценки, причем асимптотически точные, а не мажорантные.

И Теорема 1, и Утверждения 1-3 дают экспоненциальную сходимость. Однако в Теореме 1 коэффициент перед числом интервалов N равен расстоянию до ближайшей особой точки. В эвристических оценках этот коэффициент почти в 2 раза больше, то есть реальная сходимость гораздо быстрее теоретической оценки (4.1). Однако, при увеличении порядка полюса, этот коэффициент уменьшается. Поэтому возможно, что для существенно особых точек он уменьшится до теоретического значения.

В знаменателе теоретической оценки (4.1) из экспоненты вычитается 1. В эвристических оценках этого не наблюдалось. Скорее всего, вычитание 1 связано с особенностями теоретического вывода. Заметим, что это вычитание существенно лишь, когда особая точка стремится к отрезку

интегрирования. Если расстояние до особой точки не очень мало, то при разумном числе интервалов N экспонента будет существенно превышать 1 и указанный эффект станет незаметным.

Глава 5

АППРОКСИМАЦИИ ФУНКЦИЙ ФД