Лабораторные работы "Математическое и имитационное моделирование"

Лобанов И.С.

17 декабря 2015 г.

1 Лабораторная работа "Вычисления в арифметике с плавающей запятой"

Часть 1. Рассмотрим алгоритм №1, заключающийся в извлечении 52 корней квадратных из числа x и последующего 52-кратного возведения результата в квадрат.

Algorithm 1

function SQRTSQR52(x)
  $y \leftarrow x$
  for $k \leftarrow 1..52$ **do**
    $y \leftarrow \sqrt{y}$
  end for
  $z \leftarrow y$
  for $k \leftarrow 1..52$ **do**
    $z \leftarrow z^2$
  end for
  return z
end function

В силу свойства $a^{bc} = (a^b)^c$ в точной арифметике функция SQRTSQR52 должна быть тождественной, т.е. $\text{SQRTSQR52}(x) = x$ для всех $x \geq 0$. В арифметике с плавающей запятой однако ответ будет значительно отличаться от точного.

Задание 1.1. Реализовать алгоритм и вычислить относительную погрешность возвращаемого функцией SQRTSQR52 значения для x из интервала $[0, 2]$.

Задание 1.2. Объяснить величину наблюдаемой ошибки вычислений и ее зависимость от аргумента. Найти связь ошибки с представлением чисел с плавающей запятой.

Задание 1.3. Переписать алгоритм таким образом, чтобы значения z находились точнее. Найти относительную ошибку результата вычислений по улучшенному алгоритму. *Указание:* Изменить представление чисел $z \to z' + 1$.

Часть 2. Рассмотрим задачу нахождения оценки математического ожидания и дисперсии по выпорке x_n, $n = 1 \dots N$. Несмещенные оценки мат. ожидания $\bar{x}$ и дисперсии σ^2 можно вычислить по формулам:

$$\bar{x} = \frac{1}{N} \sum_{n=1}^{N} x_n,$$

$$\sigma^2 = \frac{1}{N-1} \left(\sum_{n=1}^{N} x_n^2 - \frac{1}{N} \left(\sum_{n=1}^{N} x_n \right)^2 \right).$$

Наивный алгоритм №2 в точной арифметике должен вернуть в точности эти несмещенные оценки.

Algorithm 2

function MeanDisp(N,x)
  $sum_1 \leftarrow sum_2 \leftarrow 0$
  for $n \leftarrow 1..N$ **do**
    $sum_1 \leftarrow sum_1 + x_n$
    $sum_2 \leftarrow sum_2 + x_n^2$
  end for
  $\bar{x} \leftarrow sum_1/N$
  $\sigma^2 \leftarrow (sum_2 - sum_1^2/N)/(N-1)$
  return $\bar{x}, \sigma^2$
end function

Однако в арифметике с плавающей запятой возвращаемые этим алгоритмом значения могут быть на 100% ошибочны.

Задание 2.1. Реализовать алгоритм и вычислить относительную погрешность вычисления среднего и дисперсии на ряде вида $x_n = 10^10 + w_n$, $N = 10^6$, где значения w_n независимы и имеют стандартное нормальное распределение. *Указание:* оценки дисперсии для x и для w совпадают в точной арифметике, а средние значения отличаются на 10^10. Также можно сравнить оценки с точными значениями мат. ожидания и дисперсии для стандартной нормально распределенной величины w, учитывая среднюю ошибку оценки на 10^6 измерениях.

Задание 2.2. Объяснить разницу в точности вычислений для рядов x и w. Найти среднюю относительную погрешность вычислений для рядов вида $x_n = a + bw_n$.

Задание 2.3. Улучшить точность вычисления дисперсии, сначала вычисляя среднее, затем вычисляя дисперсию по формуле

$$\sigma^2 = \frac{1}{N-1} \sum_{n=1}^{N} (x_n - \bar{N})^2.$$

Вычислить относительную ошибку оценки дисперсии по улучшенному алгоритму. Зависит ли результат от порядка членов ряда? Является ли алгоритм вычислительно устойчивым?

Задание 3.3. Реализовать алгоритм №3, вычислить относительную ошибку, проверить устойчивость.

Algorithm 3

function WELFORD(N,x)
  $mean \leftarrow M2 \leftarrow 0$
  for $n \leftarrow 1..N$ **do**
    $delta \leftarrow x_n - mean$
    $mean \leftarrow mean + delta/n$
    $M2 \leftarrow M2 + delta * (x - mean)$
  end for
  return $mean, M2/(N-1)$
end function

Часть 3. Из курса математического анализа известно, что функция $f(x) = e^x$ является голоморфной (аналитической) функцией от x на всей комплексной плоскости (вещественной прямой) и имеет разложение в ряд

$$e^x = \sum_{k=n}^{\infty} \frac{x^n}{n!},$$

сходящийся для всех значений x. Для приближенного вычисления экспоненты можно ограничиться вычислением частичной суммы ряда:

$$e^x \approx S_N = \sum_{k=n}^{N} \frac{x^n}{n!}.$$

В точной арифметике погрешность вычислений оценивается остатком ряда $R(x) = f(x) - S_N(x)$. Остаточный член можно записать, например, в форме Лагранжа:

$$R_N(x) = \frac{x^{N+1}}{(N+1)!} f^{(N+1)}(\theta x),$$

где $\theta \in [0, 1]$. Решив неравенство $|R_N(x)| < \epsilon$ можно найти число N слагаемых, которые необходимо взять для вычисления экспоненты в точке x с абсолютной погрешностью ϵ. В арифметике с плавающей запятой итоговая погрешность вычисления экспоненты будет складываться из ошибок арифметических операций и ошибки отбрасывания остатка ряда.

Задание 3.1. Оцените остаток ряда Маклорена для экспоненты, подберите такое число слагаемых, чтобы остаток ряда был меньше машинной точности.

Задание 3.2. Оцените погрешность частичной суммы при выполнение арифметических операций в арифметике с плавающей запятой.

Задание 3.3. Составьте алгоритм вычисления экспоненты с помощью частичных сумм ряда Маклорена с оптимальным числом слагаемых. Реализуйте алгоритм и вычислите относительные погрешности. *Указание:* для оценки точности можно сравнить результаты вычислений в арифметике одинарной и двойной точности.

Задание 3.4. Относительная точность вычислений быстро возрастает для отрицательных значений аргумента. Улучшите точность вычислений, выразив экспоненту отрицательного аргумента, через экспоненту положительного.

2 Лабораторная работа "Полиномиальные разложения"

При вычислении сложных функций иногда бывает удобно заменить их приближенно на более простые, в качестве которых удобно использовать многочлены. В качестве примера мы рассмотрим функцию $f(x) = \ln x$, однако следует иметь в виду, что в этом случае существуют более эффективные методы ее вычисления. Логарифмическая функция имеет особенность в начале координат, также известно, что она растет медленнее, чем любой многочлен, поэтому логарифм нельзя точно приблизить экспонентой на бесконечном интервале. Однако, логарифм любого значения можно выразить через логарифм более близкого к 1 воспользовавшись тождеством

$$\ln x^2 = 2 \ln x.$$

Тем самым вычисление логарифма произвольного аргумента сводится к его вычислению на малом интервале, содержащем единицу, например, на интервале $[\frac{1}{5}, 5]$. Более того, вычисления достаточно проводить только для для $x > 1$, так как

$$\ln \frac{1}{x} = -\ln x.$$

С помощью преобразования

$$x = \frac{1 + 2u/3}{1 - 2u/3},$$

интервал $x \in [\frac{1}{5}, 5]$ преобразуется в интервал $u \in [-1, 1]$ относительно новой переменной u. На симметричном интервале разложение логарифма содержит только нечетные степени u.

$$\ln x = \ln \left(\frac{1 + 2u/3}{1 - 2u/3} \right) = \frac{4}{3} u + \frac{2^4}{3^4} u^3 + \dots \tag{1}$$

Полученное разложение в ряд и формулы приведения позволяют вычислять логарифм численно, однако разложение по степеням u имеет хорошую точность только при малых u. Для улучшения точности приближения на всем интервале необходимо использовать другие разложения по многочленам.

Произвольный многочлен $p(x)$ степени N можно представить в виде

$$p(x) = a_N x^N + a_{N-1} x^{N-1} + \dots + a_1 x + a_0,$$

где $a_N \neq 0$. В указанном виде многочлен записан как линейная комбинация одночленов x^k или другими словами разложен по мономиальному базису. В общем виде, многочлен можно разложить по базису ϕ_n в виде

$$p(x) = \sum_{j=0}^{N} b_j \phi_j(x).$$

Одним из наиболее важных полиномиальных базисов, является базис из полиномов Чебышева.

Многочлены Чебышева имеют вид

$$T_n(x) = cos(n \cdot \arccos x),$$

для $n = 0, 1, 2, \dots$. Их также можно задать рекуррентно

$$T_0(x) = 1, \quad T_1(x) = x, \quad T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x).$$

Нули многочлена $T_n(x)$ расположены в точках

$$\zeta_j^{(n)} = \cos\frac{\pi(k-1/2)}{n}.$$

Многочлены Чебышева ортогональны относительно скалярного произведения

$$\langle f_1|f_2\rangle = \sum_{j=1}^{n} f_1(\zeta_j^n) f_2(\zeta_j^n). \tag{2}$$

Указанная ортогональность позволяет легко находить коэффициенты разложения по многочленам Чебышева через скалярные произведения. Построенное таким способом разложение, однако, не дает наиболее аккуратного равномерного приближения.

Метод Ланцоша дает другой способ нахождения разложения функции по полиномам Чебышева. Рассмотрим сначала дифференциальное уравнение, решением которого является искомый логарифм

$$x\frac{dy}{dx} - 1 = 0, \quad y(1) = 0.$$

Относительно переменной u уравнение принимает вид

$$(1 - 4u^2/9)\frac{dy}{du} - 4/3 = 0.$$

Для решения этого уравнения разложим производную решения по многочленам Чебышева:

$$\frac{dy}{du} = \sum_{k=0}^{N} a_k T_k(x).$$

Подставляя разложение производной по многочленам в дифференциальное уравнение и замечая, что

$$u^2 T_k(u) = \frac{1}{4}(T_{k-2}(u) + 2T_k(u) + T_{k+2}(u)),$$

мы преобразуем дифференциальное уравнение в алгебраическое

$$\sum_{k=0}^{N+2} \tau_k T_k(u) = 0,$$

для подходящих коэффициентов $\tau_k = \tau_k(a_0, \ldots, a_N)$. Для нахождения разложения по Ланцошу необходимо решить уравнения

$$\tau_0 = \tau_1 = \cdots = \tau_N = 0,$$

найдя тем самым a_k и dy/du. Интегрируя dy/du можно найти разложение для $y(u)$, воспользовавшись тождествами:

$$\int T_k(u)du = \frac{1}{2(k+1)}T_{k+1}(u) - \frac{1}{2(k-1}T_{k-1}(u) + \frac{\sin(\pi k/2)}{k^2-1}.$$

Задание 1. Найдите еще один член разложения (1) по степеням u.

Задание 2. Найдите разложение логарифма как функции u по многочленам Чебышева $T_n(u)$, воспользовавшись значениями в нулях многочлена Чебышева и ортогональностью многочленов Чебышева, относительно скалярного произведения (2).

Задание 3. Найдите разложение по многочленам Чебышева, используя метод Ланцоша (Lanczos).

Задание 4. Реализуйте схему Горнера для вычисления многочленов и алгоритм Clenshaw для вычисления разложения по многочленам Чебышева.

Задание 5. Найдите относительные погрешности различных разложений, объясните различия.

Задание 6. Выразите значения $f(x)$ для произвольного $x > 0$ через $f(x')$ для $x' \in [\frac{1}{5}, 5]$. Составьте алгоритм вычисления $f(x)$ для всех $x > 0$. Теоретически оценить точность вычислений с помощью этого алгоритма.

Задание 7. Реализуйте алгоритм вычисления логарифма на положительной полуоси. Экспериментально найдите относительные погрешности при использовании разложений из заданий 1,2,3 и сравните их с теоретическим предсказанием.

3 Лабораторная работа "Метод Ньютона"

Вычисление функций с максимально возможной в плавающей арифметике точностью является одной из основных задач численных методов. Вычисления, опирающиеся на аппроксимацию функции многочленом, представление в виде ряда или другие разложения, вообще говоря не позволяют вычислять функцию с машинной точностью. Однако, во многих случаях значение функции может быть вычислено с машинной точностью и за малое время с помощью итерационных методов. В настоящей лабораторной работе рассматривается метод Ньютона, позволяющий находить значения функций, заданных неявно.

Теоретический предел точности вычисления функции в приближенной арифметике оценивается с помощью чисел обусловленности, связывающих относительные ошибки аргумента и значения функции. Рассмотрим произвольную функцию

$$y = f(x).$$

Если при вычислениях будет использовано значение аргумента с относительной ошибкой δx, то результат будет найден также с некоторой относительно ошибкой, даже при точном вычислении функции. Из определения относительной ошибки имеем:

$$y(1 + \delta y) = f(x(1 + \delta x)),$$

откуда

$$\delta_y = \frac{f(x(1 + \delta x))}{f(x)} - 1.$$

Числом обусловленности μ называют верхнюю границу отношения относительной ошибки значения функции к значению ошибки аргумента:

$$\left| \frac{\delta y}{\delta x} \right| \leq \mu(f).$$

Если относительная ошибка аргумента мала ($\delta x \to 0$), а функция дифференцируема, то число обусловленности приближенно можно найти следующим образом:

$$\left|\frac{\delta y}{\delta x}\right| = \frac{1}{|\delta x|}\left|\frac{f(x(1+\delta x))}{f(x)} - 1\right| = \frac{1}{|\delta x|}\left|\frac{f(x) + f'(x)x\delta x + o(\delta x)}{f(x)} - 1\right| \to \left|\frac{xf'(x)}{f(x)}\right| \approx \mu(f).$$

Если число обусловленности велико ($\mu >> 1$), то говорят, что задача плохо обусловлена. В арифметике с плавающей запятой получить значение плохо обусловленной функции с машинной точностью невозможно, даже если функция f вычисляется без ошибок.

Для нахождения нулей дифференцируемой функции $f(x)$ можно воспользоваться методом Ньютона. Имея некоторое начальное приближение x_0 для корня x^* уравнения $f(x^*) = 0$, итерации по методу Ньютона определены формулой

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Каждая следующая итерация $x = x_{n+1}$ является решением уравнения

$$f'(x)(x - x_n) + f(x_n) = 0,$$

т.е. абсциссой точки пересечения касательной и оси абсцисс. Таким образом метод Ньютона заключается в аппроксимации функции f в окрестности текущего приближения x_n, и выборе нулей этой аппроксимации в качестве следующего приближения x_{n+1} нулей функции $f(x)$. Если итерации x_n сходятся, то предел x^* последовательности x_n является нолем функции f, т.е. $f(x^*) = 0$.

Задание 1. Вычислить числа обусловленности функции $y = \sqrt{x}$. Найти максимальную точность вычисления квадратного корня в арифметике с плавающей запятой.

Задание 2. Используя метод Ньютона нахождения нулей функции, составить алгоритм вычисления квадратного корня. *Указание:* нахождения квадратного корня y числа x равносильно нахождению корня x уравнения $y = x^2$.

Задание 3. Реализовать алгоритм, найти точность вычисления квадратного корня. Показать, что полученная точность соответствует теоретически возможной.

4 Лабораторная работа "Решение систем линейных уравнений"

Многие задачи численного моделирования сводятся к решению линейных систем вида:

$$AX = B \quad \Leftrightarrow \quad \sum_{m=1}^{N} A_{nm}X_m = B_n,$$

где B – известный вектор с N координатами, A – известная матрица размера $N \times N$, и X – искомый вектор с N координатами. Система имеет единственное решение, если определитель матрицы отличен от ноля. Однако в приближенной арифметике определитель может оказаться неотличимым от ноля, так что система может не иметь решения, даже если система с

формально не вырождена. Числа обусловленности линейной системы определены как Можно показать, что для невырожденных матриц

$$\mu(A) = \|A^{-1}\| \cdot \|A\|,$$

где A^{-1} обозначает обратную матрицу, $\|A\|$ обозначает норму оператора, т.е.

$$\|A\| = \sup_{x \neq 0} \frac{\|Ax\|}{\|x\|}.$$

Если известно сингулярное разложение для матрицы A, то числа обусловленности можно записать в виде:

$$\mu(A) = \frac{\sigma_{max}(A)}{\sigma_{min}(A)},$$

где $\sigma_{max}(A)$ – максимальное сингулярное число матрицы A, $\sigma_{min}(A)$ – минимальное отличное от нуля сингулярное число матрицы A.

Задание 1. Найти числа обусловленности для квадратных матриц с коэффициентами

$$U_{ij} = \begin{cases} 1, & i = j \\ -2, & i > j, \\ 0, & i < j. \end{cases}.$$

Чему равен определитель матрицы? Как числа обусловленности зависят от размера матрицы?

Задание 2. Найти решение системы с матрицей U размера 32×32 и и правой частью вида

$$B_i = \begin{cases} 1, & i \text{ нечетно}, \\ -1/3, & i \text{ четно}, \end{cases}$$

методом обратных подстановок, пользуясь треугольностью матрицы системы. Найти точность решения. Найти невязку. Объясните результат.

Задание 3. Реализуйте вычисление QR разложения матрицы U с помощью преобразования Хаусхолдера. Оцените унитарность матрицы Q из разложения, сравните произведение матриц Q и R и матрицу U. Решите системы из задания 2 с помощью QR разложения. Найдите невязку и точность решения. Объясните результат.

5 Лабораторная работа "Метод конечных разностей"

Численное решение уравнений математической физики и уравнений в частных производных является важной составной частью задач математического моделирования. Основными этапами численного решения таких уравнений являются дискретизация исходного уравнения и решение полученной линейной системы алгебраических уравнений. В настоящей работе мы рассмотрим дискретизацию методом конечных разностей и решение системы методом Якоби.

Продемонстрируем методы на примере решения уравнений Пуассона с граничными условиями Дирихле на круге:

$$\begin{cases} \Delta u(x,y) = f(x,y), \quad (x,y) \in \Omega = \{(x,y)\colon x^2+y^2 \le 1\}, \\ u(x,y) = 0, \text{ для } x^2+y^2 = 1, \end{cases} \tag{3}$$

причем будет решать уравнение для простейшей правой части $f \equiv 1$. Напомним, что оператор Лапласа в декартовых координатах имеет вид

$$\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}.$$

Нетрудно убедиться, что в данном случае явное решение имеет вид:

$$u(x,y) = \frac{x^2+y^2-1}{4}. \tag{4}$$

Решим уравнение (4) численно. Решения начинается с дискретизации уравнения, т.е. с замены уравнения на близкое к нему, но содержащее конечное число переменных. Проведем дискретизацию методом конечных разностей, для чего заменим производные на их центральные конечные разности. Известно, что дважды дифференцируемые функции удовлетворяют следующим тождествам:

$$\frac{\partial u}{\partial x}(x,y) = \frac{u(x+\delta/2,y) - u(x-\delta/2,y)}{\delta} + o(\delta),$$

$$\frac{\partial u}{\partial y}(x,y) = \frac{u(x,y+\delta/2) - u(x,y-\delta/2)}{\delta} + o(\delta),$$

где δ – конечное приращение аргументов, символ $o()$ означает бесконечно малую величину при $\delta \to 0$. Применяя эти формулы дважды, получаем приближение для лапласиана:

$$\Delta u(x,y) = \frac{1}{\delta^2}\Big(u(x+\delta,y) + u(x-\delta,y) + u(x,y+\delta) + u(x,y-\delta) - 4u(x,y)\Big) + o(1).$$

Последняя формула позволяет нам заменить уравнение Пуассона на его дискретный аналог на квадратной решетке. Заметим, что так как задача обладает вращательной симметрией, то переходом к полярной системе координат можно получить более точное решение, однако мы намеренно не будем этого делать, чтобы изучить влияние несогласованности решетки и границы области на точность дискретизации.

Заменим лапласиан в уравнении Пуассона конечными разностями и отбросим бесконечные малые:

$$\frac{1}{\delta^2}\Big(u(x+\delta,y) + u(x-\delta,y) + u(x,y+\delta) + u(x,y-\delta) - 4u(x,y)\Big) = 1. \tag{5}$$

Полученное уравнение не содержит производных, т.е. является алгебраическим, однако по прежнему имеет бесконечное число неизвестных. Чтобы ограничить число неизвестных мы будем рассматривать значения искомой функции только на квадратной решетке. Введем обозначение для значений функции в узлах решетки:

$$u_{kj} = u(k\delta, j\delta).$$

Так как решение ищется только в области $x^2 + y^2 \le 1$, достаточно ограничиться следующим диапазоном изменения индексов: $k, j = -N \dots N$, где $N = [\frac{1}{\delta}]$. На решетке уравнение (6) принимает вид:

$$\frac{1}{\delta^2}\Big(u_{k+1,j} + u_{k-1,j} + u_{k,j+1} + u_{k,j-1} - 4u_{k,j}\Big) = 1, \quad \forall k, j, \text{ таких что } k^2 + j^2 < \delta^{-2}. \tag{6}$$

Вектор $u_{k,j}$ имеет конечное число координат, следователь система (6) есть алгебраическая линейная система с конечным числом неизвестных $u_{k,j}$, решить которую можно, например, методами лабораторной работы "Решение систем линейных уравнений". Однако, мы до сих пор не использовали граничных условий, поэтому решение системы не является единственным. Действительно, система (6) содержит переменные $u_{k,j}$ для $k^2 + j^2 \ge \delta^{-2}$, которые невозможно определить из системы, так как она содержит меньше уравнений, чем переменных. Учтем, что значения $u_{k,j}$ в узлах, лежащих близко к границе области, в силу граничных условий ($u(x, y) = 0$ для $x^2 + y^=1$) близки к нулю. Таким образом, дополняя (6) приближенными граничными условиями, получаем окончательную дискретизацию:

$$\begin{cases} u_{k+1,j} + u_{k-1,j} + u_{k,j+1} + u_{k,j-1} - 4u_{k,j} = \delta^2, & \text{для } k^2 + j^2 < \delta^{-2}, \\ u_{k,j} = 0, & \text{для } k^2 + j^2 \ge \delta^{-2}. \end{cases} \tag{7}$$

Заметим, что несмотря на обозначения с двумя индексами $u_{k,j}$ обозначает вектор. При машинной реализации удобно сопоставлять двойному индексу (k, j) одно число n, указывающее на место хранения индекса в памяти, что можно сделать разными способами, например:

1. $n = 1 + j + N + (k + N) \cdot N$ для реализаций, в которых нумерация индекса n начинается с единицы, и матрица хранится по строкам (FORTRAN, MATLAB,...).
2. $n = (j + N) * N + k + N \cdot N$ для реализаций, в которых нумерация индекса n начинается с нуля, и матрица хранится по столбцам (C, C++,...).

Систему (7) можно переписать в матричном виде:

$$Au = B \quad \Leftrightarrow \quad \sum_{k',j'} A_{k,j;k',j'} u_{k',j'} = B_{k,j}, \tag{8}$$

где матрица имеет вид

$$A_{k,j;k',j'} = \begin{cases} -4, \text{ если } k = k', j = j', \\ 1, \text{ если } |k - k'| = 1, j = j', k^2 + j^2 < \delta^{-2}, \\ 1, \text{ если } |j - j'| = 1, k = k', k^2 + j^2 < \delta^{-2}, \\ 0, \text{ во всех остальных случаях,} \end{cases} \tag{9}$$

и правая часть равна

$$B_{k,j} = \begin{cases} \delta^2, \text{ если } k^2 + j^2 < \delta^{-2}, \\ 0, \text{ во всех остальных случаях.} \end{cases} \tag{10}$$

Так как размер системы быстро растет с уменьшением шага решетки δ, то для ее решения нерационально использовать явные методы, такие как метод Гаусса. Вместо этого обычно применяют итерационные методы, позволяющие получить решение с заданной точностью

используя небольшой объем памяти. Заметим, что матрица системы симметричная и положительно определенная, что позволяет использовать для решения системы эффективные методы, например, метод сопряженного градиента, однако в настоящей работе мы воспользуемся простейшим методом, методом Якоби.

Рассмотрим систему

$$AX^* = B,$$

с матрицей A, обладающей свойством диагонального преобладания:

$$|A_{j;j}| \leq \sum_{j' \neq j} |A_{j;j'}|.$$

Для решения такой системы можно воспользоваться методом Якоби. Искомое решение находится как предел последовательности $X^{(k)} \to X^*$. Первый элемент $X^{(0)}$ может быть выбран произвольно, однако хорошее приближение уменьшает число итераций. Последующие итерации находятся по формуле

$$X_j^{(n+1)} = \frac{1}{A_{j,j}} \left(B_j - \sum_{j \neq j'} A_{j,j'} X_{j'}^{(n)} \right),$$

последовательно вычисляя коэффициенты $X_j^{(n+1)}$, начиная с первого. Можно доказать, что точность приближения оценивается следующим образом для некоторой константы q:

$$\|X^{(k)} - X^{(0)}\| \leq q^k \|X^* - X^{(0)}\|,$$

где l^2 норма считается по формуле:

$$\|X\| = \sqrt{\sum_j |X_j|^2}.$$

Задание 1. Реализовать алгоритм дискретизации уравнения Пуассона с граничными условиями Дирихле (3) методом конечных разностей на квадратной решетке с шагом δ. Составить дискретизованные системы в матричном виде. Пояснить способ получения матрицы. Почему при дискретизации мы не воспользовались хорошо известным тождеством

$$f'(x) = \frac{1}{h}(f(x+h) - f(x)) + o(1),$$

а использовали центральные конечные разности,

Задание 2. Реализовать метод Якоби решения линейных систем. Решить дискретизованное уравнение Пуассона (7) для шага решетки $\delta = 10^{-1}$. Построить график зависимости невязки $R^{(k)} = B - AX^{(k)}$ от номера итерации. Найти скорость сходимости.

Задание 3. Найти решения дискретизованного уравнения Пуассона (7) для шага решетки δ стремящегося к нулю. Построить график ошибки $E(\delta)$ дискретизации в зависимости от шага решетки

$$E(\delta) = \max_{k,j} |u_{k,j} - u(k\delta, j\delta)|,$$

где $u(x, y)$ аналитическое решение уравнения, имеющее вид (4). Показать, что приближенное решение сходится к точному, оценить скорость сходимости.

6 Лабораторная работа "Метод конечных 'элементов"

Несмотря на свою простоту, метод конечных разностей имеет ограниченную применимость для решения уравнений математической физики, так как он дает большую погрешность для областей со сложной границей. Популярной альтернативой методу конечных разностей является метод конечных элементов (а также близкий метод граничных элементов). В данной лабораторной мы изучим метод конечных элементов на примере решения уравнения Пуассона в круге.

Уравнение Пуассона (3) записано в сильной формулировке, которая не подходит для применения метода конечных элементов. Чтобы записать уравнение Пуассона в слабом виде, умножим (3) на произвольную функцию $\phi(x, y)$ и проинтегрируем по области Ω:

$$\iint_\Omega \phi(x,y)\Delta u(x,y)\,dx\,dy = \iint_\Omega \phi(x,y) f(x,y)\,dx\,dy.$$

Если u удовлетворяет (3), то последнее равенство должно выполняться для всех ϕ. Воспользуемся формулой Грина:

$$\iint_\Omega (\phi\Delta u + \nabla\phi\cdot\nabla u)\,dV = \int_{\partial\Omega} \phi(n\cdot\nabla u)\,dS,$$

где $\partial\Omega$ обозначает границу области Ω, $dV = dx\,dy$ – элемент объема, dS – элемент поверхности, n – нормаль к поверхности $\partial\Omega$. Получаем уравнение Пуассона в виде:

$$\int_{\partial\Omega} \phi(x,y)(n(x,y)\cdot\nabla u(x,y))\,dS - \iint_\Omega \nabla\phi(x,y)\cdot\nabla u(x,y)\,dx\,dy = \iint_\Omega \phi(x,y) f(x,y)\,dx\,dy. \quad (11)$$

Вспомним, что функция u подчинена граничным условиям Дирихле, т.е. обращается в ноль на границе. Если ограничиться рассмотрением только ϕ, удовлетворяющих тем же граничным условиям, что и u, то поверхностный интеграл в уравнении (11) обращается в нуль, и мы получаем окончательное выражение для уравнения Пуассона в слабой форме:

$$-\iint_\Omega \nabla\phi(x,y)\cdot\nabla u(x,y)\,dx\,dy = \iint_\Omega \phi(x,y) f(x,y)\,dx\,dy. \quad (12)$$

Следующим шагом проводится дискретизация методом Галеркина. Выберем некий линейно независимый набор из $N < \infty$ функций $e_n(x, y)$, $n = 1..N$, удовлетворяющих краевым условиям Дирихле, и предположим, что при $N \to \infty$ функции e_n образуют базис в пространстве решений. Зафиксируем некоторую размерность $N < \infty$, и обозначим через u_n коэффициенты разложения функции u по первым N базисным функциям e_n, т.е. пусть выполняется тождество:

$$u(x,y) = \sum_{n=1}^{N} u_n e_n(x,y),$$

аналогично введет коэффициенты ϕ_n разложения по базису функции ϕ. Подставляя эти разложение в (12) получаем

$$-\sum_{n=1}^{N}\sum_{k=1}^{N} \phi_n u_k \iint_\Omega \nabla e_n(x,y)\cdot\nabla e_k(x,y)\,dx\,dy = \sum_{n=1}^{N} \phi_n \iint_\Omega e_n(x,y) f(x,y)\,dx\,dy.$$

Равенство может выполняться для всех ϕ только если все коэффициенты перед ϕ_n равны нулю, т.е.

$$-\sum_{k=1}^{N} u_k \iint_\Omega \nabla e_n(x,y) \cdot \nabla e_k(x,y)\,dx\,dy = \iint_\Omega e_n(x,y) f(x,y)\,dx\,dy \quad \forall n.$$

Введем матрицы так называемых интегралов перекрытия:

$$M_{nk} = -\iint_\Omega \nabla e_n(x,y) \cdot \nabla e_k(x,y), \quad Y_n = \iint_\Omega e_n(x,y) f(x,y)\,dx\,dy,$$

получаем дискретизованный вариант уравнения Пуассона

$$\sum_{k=1}^{N} M_{nk} u_k = Y_n \quad \forall n. \tag{13}$$

Отличительной чертой метода конечных элементов является выбор базиса в виде кусочно гладких функций. Выберем некоторую триангуляцию области Ω, т.е. разобьем область на множество треугольников. Базис в пространстве решений выбираем в виде всех таких функций $e_n(x,y)$, что

1. e_n равно единице ровно в одном узле (т.е. общей вершине нескольких треугольников).
2. e_n равно нулю во всех остальных узлах.
3. e_n на каждом треугольнике из триангуляции задается линейной функцией.

Так как e_n либо равно тождественно нулю на треугольнике, либо не равно нулю ровно в одной вершине, то e_n легко задать явно. Без ограничения общности можно считать, что e_n равно единице в вершине $A(x_0,y_0)$ и равно нулю в вершинах $B(x_1,y_1)$, $C(x_2,y_2)$. Тогда e_n имеет вид для (x,y) лежащих внутри треугольника ABC:

$$e_n(x,y) = 1 + \frac{(y_1-y_2)(x-x_0) - (x_1-x_2)(y-y_0)}{S}, \tag{14}$$

где $S = (x_1-x_0)(y_2-y_0) - (x_2-x_0)(y_1-y_0)$ – две площади треугольника ABC. Так как сужение e_n на отдельный треугольник функция линейная, то ее градиент есть функция постоянная на этом треугольнике:

$$\nabla e_n = \frac{1}{S}(y_1-y_2, x_2-x_1).$$

Интегралы перекрытий M_{nk} все равны нулю, кроме ситуации, когда функции e_n и e_k отличны от нуля в узлах, являющихся вершинами одного треугольника, т.о. матрица M является разряженной. Отличные от нуля коэффициенты матрицы A можно найти следующим образом: пусть e_n отлично от нуля в вершине A, а e_k отлично от нуля в точке B, тогда

$$M_{nk} = -\frac{1}{S^2}\iint_{ABC} (y_1-y_2, x_2-x_1)\cdot(y_2-y_0, x_0-x_2)\,dx\,dy = \\ = -\frac{(y_1-y_2)(y_2-y_0)+(x_2-x_1)(x_0-x_2)}{S}.$$

Аналогично можно найти Y_n для данной функции f.

Так как размер N искомого вектора коэффициентов u_n на практике очень велик, то для решения системы (13) обычно используются итерационные методы, позволяющие быстро находить ответ для систем с разряженной матрицей M. Одним из наиболее популярных методов решения систем с симметрической положительно определенной матрицей M является метод сопряженного градиента, являющийся одним из методов подпространств Крылова. Напомним, что матрица называется симметрической, если она совпадает со своей транспонированной, т.е. $M_{jk} = M_{kj}$ для всех k, j. Матрица называется положительно определенной, если для всех $s \neq 0$ выполняется $s^T M s > 0$. Метод сопряженного градиента можно сформулировать в виде следующего алгоритма:

Algorithm 4

function CONJUGATEGRADIENT(M,Y,ϵ)
  $r_0 \leftarrow Y$
  $p_0 \leftarrow r_0$
  $k \leftarrow 0$
  loop
    $\alpha_k \leftarrow \frac{r_k^T r_k}{p_k^T M p_k}$
    $x_{k+1} \leftarrow x_k + \alpha_k p_k$
    $r_{k+1} \leftarrow r_k - \alpha_k M p_k$
    if $\|r_{k+1}\| < \epsilon$ **then return** x_{k+1}
    end if
    $\beta_k \leftarrow \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
    $p_{k+1} \leftarrow r_{k+1} + \beta_k p_k$
    $k \leftarrow k + 1$
  end loop
end function

Задание 1. Реализовать алгоритм дискретизации уравнения Пуассона на круге с граничными условиями Дирихле (3) методом конечных элементов. Можно воспользоваться либо простейшей триангуляцией с узлами в вершинах правильных многоугольников, либо воспользоваться готовой реализацией одного из методов построения триангуляций. Для данной триангуляции вычислять разряженную матрицу M линейной системы уравнений и правую часть Y системы.

Задание 2. Какая триангуляция будет оптимальной? Что такое условия Делоне? Покажите, что матрица M является симметрической и отрицательно определенной. Можно ли для системы с матрицей M применить метод сопряженного градиента?

Задание 3. Реализуйте метод сопряженного градиента или подобный итерационный метод для решения уравнения Пуассона (3) в дискретизации конечными элементами. Оцените скорость сходимости. Постройте график нормы невязки как функции числа итераций.

Задание 4. Сравните решение уравнения методом конечных разностей с аналитическим решением (4). Постройте график зависимости ошибки нахождения решения методом конечных элементов от числа узлов триангуляции. С какой скоростью ошибка стремится к нулю при измельчении разбиения?

Задание 5. Сравните методы конечных элементов и конечных разностей. Какой метод дает лучшую точность при одинаковой размерности пространств приближенных решений? Какой метод требует больше времени на дискретизацию? Какой метод требует больше времени на выполнение одной итерации решения системы (13)? Какой метод лучше подходит для решения задач со сложными граничными условиями? Какие сложности возникнут, если решения уравнения имеют особенность?

7 Лабораторная работа "Волновое уравнение"

Многие задачи математической физики включают в себя время t, для работы с которым нужны специальные методы. В отличии от пространственных переменных x, y, z, временная переменная только одна, однако включающие t уравнения типично имеют параболический или гиперболический, а вместо граничных условий вида условий Дирихле или Неймана по t накладываются начальные условия в виде условий Коши. В качестве примера мы рассмотрим однородное волновое уравнение на круге для единичной скорости:

$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2},\ x^2 + y^2 \leq 1,\ t \geq 0, \tag{15}$$

с граничными условиями

$$u(t; x, y) = 1 \text{ для всех } x^2 + y^2 = 1, \tag{16}$$

и начальными условиями

$$u(0; x, y) = 1,\ \frac{\partial u}{\partial t}(0; x, y) = 0 \text{ для всех } x, y. \tag{17}$$

Заметим, что если бы мы знали зависимость решения u от времени t, то уравнение (15) оказалось равносильным изученному в предыдущих лабораторных работах уравнению (3). Воспользуемся этим наблюдением и результатами предыдущих работ для решения (15). Временная переменная t и пространственные переменные x, y играют разную роль в уравнении (15), поэтому дискретизация по ним производится обычно независимо. Рассмотрим сначала случай, когда дискретизация начинается со времени. Пользуясь известным фактом, что решение u дважды дифференцируемо по t и заменяя производные по t центральными конечными разностями, получаем (15) в следующем виде:

$$\frac{1}{\delta^2}(u(t-\delta; x, y) - 2u(t; x, y) + u(t+\delta; x, y)) = \Delta_{x,y} u(t; x, y), x^2 + y^2 \leq 1,\ t \geq 0, \tag{18}$$

Оператор Лапласа $\Delta_{x,y} = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ был ранее изучен нами, в частности мы знаем, как с помощью конечно-разностной дискретизации или с помощью метода конечных элементов перейти от уравнения $\delta_{x,y} u = f$ перейти к уравнению $H\hat{u} = \hat{f}$, где $\hat{u}$ и $\hat{f}$ обозначают дискретизации u и f. Применяя ту же технику к уравнению (18) получим уравнение на неизвестное $\hat{u}(t)$, которое уже дискретизовано по пространственным переменным, но зависит от времени t непрерывным образом:

$$\frac{1}{\delta^2}\hat{u}(t-\delta) - 2\hat{u}(t) + \hat{u}(t+\delta)) = H\hat{u}(t),\ t \geq 0, \tag{19}$$

Рассматривая только моменты времени $t = t_k = \delta \cdot k$, $k = 0, \ldots$ и обозначая $\hat{u}_k = \hat{u}(\delta \cdot k)$, мы получаем конечно-разностную по времени дискретизацию волнового уравнения:

$$\hat{u}_{k+1} = (2 + \delta^2 H)\hat{u}_k - \hat{u}_{k-1},\ t \geq 0. \tag{20}$$

Это уравнение позволяет находить решение в следующий момент времени, если известны значения в два предыдущих момента. Чтобы найти отправную точку для нахождения решения, обратимся к начальным условиям (17). Первое условия сразу дает нам значение решения в начальный момент времени:

$$\hat{u}_0 = \hat{1}, \tag{21}$$

где $\hat{1}$ – результат дискретизации функции тождественно равной единице (совпадает с 1 для конечно-разностной дискретизации). Чтобы найти значение решения в еще один момент времени, заменим производную в начальных условиях на симметрическую конечную разность с удвоенным шагом, т.е.

$$\frac{1}{2\delta}(u(\delta; x, y) - u(-\delta; x, y)) = 0 \;\Rightarrow\; \hat{u}_1 - \hat{u}_{-1} = 0.$$

Подставляя это ограничение в (20) получаем:

$$\hat{u}_1 = \left(1 + \frac{1}{2}\delta^2 H\right)\hat{u}_0 = \left(1 + \frac{1}{2}\delta^2 H\right)\hat{1}, \tag{22}$$

Таким образом (21) и (22) дают базу для последовательного нахождения приближенного решения $\hat{u}$ во все моменты времени по формуле (20).

Задание 1. Провести дискретизацию волнового уравнения (15) в граничными условиями (16) и начальными условиями (17) методом конечных разностей по времени t и методом конечных разностей или конечных элементов по пространственным переменным x и y.

Задание 2. Реализовать алгоритм вычисления приближенного решения $\hat{u}$ по явным формулам (20) на интервале $t \in [0, 1]$. Построить анимацию решения для шага $\delta = 0.1$.

Задание 3. Из физических соображений ясно, что энергия

$$E[u(t)] = \frac{1}{2}\iint_\Omega \left[\left(\frac{\partial u}{\partial t}\right)^2 + \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2\right] dx\, dy.$$

не изменяется со временем на точном решении u. Оцените значение энергии в каждый момент времени на приближенном решении, и постройте график зависимости энергии от времени. Сохраняется ли энергия? Почему?

Задание 4. Оценить точность вычисления решения. Каким образом точность решения зависит от величины шага δ?

Задание 5. При преобразовании начальных условий к виду (22) мы воспользовались центральной конечной разностью, однако можно было, например, воспользоваться прямой конечной разностью и получить начальное условие в виде $\hat{u}_0 = \hat{u}_1 = \hat{1}$. Какая из дискретизаций начальных условий точнее? Почему?

Задание 6. Ответить на следующие вопросы. Можно ли было провести дискретизацию по времени методом конечных элементов? Какой выигрыш дает использование методов Рунге-Кутты? В чем достоинство неявной схемы?