

Предлагается выполнить одно из заданий, в котором требуется найти решение одного скалярного уравнения.

Чтобы выполнить такое задание требуется построить алгоритм метода, составить программу его реализации на вычислительной машине и провести расчет. При разработке алгоритма надо обращать внимание на ситуации, когда малые погрешности при вводе данных или при расчете могут вызывать большие погрешности вычисленного решения.

Дополнительная информация, которая требуется в задании, выдается преподавателем. После выполнения каждого задания требуется составить отчет по форме:

1. постановка задачи;
2. распечатка программы и результатов счета;
3. анализ погрешности результатов.

1 Тема „Численные методы решения одного уравнения“

Задача о нахождении формул для корней многочленов была одним из важнейших разделов итальянской математики эпохи Ренессанса. Для полиномов 2-й, 3-й, 4-й степени еще несколько столетий назад были найдены алгоритмы, использующие для вычисления корней конечное число квадратных или кубических радикалов, зависящих от коэффициентов многочленов. В тридцатых годах 19 века Галуа доказал невозможность подобных алгоритмов для полиномов 5-й или более высокой степени, даже если допустить в формулах радикалы с произвольным показателем n .

Необходимость найти корни одного нелинейного алгебраического или трансцендентного (неалгебраического) уравнения с одним неизвестным часто появляется в различных прикладных задачах. Иногда задача решения уравнения с одним неизвестным возникает как самостоятельная. Кроме того, изучение этой задачи позволяет увидеть принципы построения методов и алгоритмов решения систем нелинейных уравнений.

Вряд ли когда либо будет существовать метод (алгоритм) вычисления с достаточной точностью всех корней любого нелинейного трансцендентного уравнения

$$f(x) = 0. \quad (1.1)$$

Для каждого алгоритма найдутся нелинейные функции, достаточно замысловатые, чтобы вывести его из строя. Вообще говоря, ответы на вопросы существования и типов

(единственны или кратны, вещественны или комплексны) корней произвольной функции $f(x)$ выходят за рамки возможностей, которые следует ожидать от алгоритмов решения нелинейных уравнений.

Во многих случаях вычислитель из дополнительных соображений знает существует ли корень уравнения, единствен ли он, находится ли он в определенной области. Для целых алгебраических функций (многочленов) существуют теоретические оценки границ области, в которой находятся их корни (см., например, [6]). Существуют также некоторые приемы локализации корней нелинейных уравнений (см. стр. 8).

В заданиях этого раздела требуется найти решения уравнения (1.1) с заданной точностью ε , причем $f(x)$ – вещественная функция вещественного аргумента, производные которой вплоть до некоторого порядка непрерывны в окрестности ее нуля. Кроме того, известно либо начальное приближение к корню x_0 , либо интервал $[a, b]$ существования корня.

Задание 1 "Метод половинного деления (дихотомии, бисекции) решения одного уравнения".

Требуется, используя метод бисекции, вычислить с заданной точностью ε один корень при каждом значении параметра a уравнения

$$f(x, a) = 0, \quad 0 \leq a \leq 1$$

при значениях параметра $a_i = 0.1 \cdot i$, $i = 0, 1, \dots, 10$, и нарисовать график зависимости вычисленного корня от параметра a .

Задание 2 "Метод простой итерации решения одного уравнения".

Требуется, используя метод простой итерации, вычислить с заданной точностью ε один корень при каждом значении параметра a уравнения

$$f(x, a) = 0, \quad 0 \leq a \leq 1$$

при значениях параметра $a_i = i \cdot 0.1$, $i = 0, 1, \dots, 10$, и нарисовать график зависимости вычисленного корня от параметра a .

Задание 3 "Метод Ньютона решения одного уравнения".

Требуется, используя метод Ньютона или его модификации, вычислить с заданной точностью ε один корень при каждом значении параметра a уравнения

$$f(x, a) = 0, \quad 0 \leq a \leq 1$$

при значениях параметра $a_i = i \cdot 0.1$, $i = 0, 1, \dots, 10$, и нарисовать график зависимости вычисленного корня от параметра a .

Методические указания

1. Метод половинного деления (дихотомии, бисекции) Пусть заданная функции $f(x)$ непрерывна и меняет знак на $[x_0, x_1]$. Выполнение условия $f(x_0)f(x_1) < 0$ означает, что на $[x_0, x_1]$ находится не менее одного корня нечетной кратности функции $f(x)$.

Если вычислить значение $f(c)$, где $c = (x_0 + x_1)/2$, то возможны три случая:

1. знак $\text{sgn}(f(c)) = \text{sgn}(f(x_0))$, тогда $f(x)$ опять меняет знак на $[c, x_1]$;
2. знак $\text{sgn}(f(c)) \neq \text{sgn}(f(x_0))$, тогда $f(x)$ опять меняет знак на $[x_0, c]$;
3. $|f(c)| \leq \varepsilon$, тогда значение $x = c$ можно считать приближенным корнем, вычисленным с точностью ε .

Если в первом случае положить $x_0 = c$, а во втором $x_1 = c$, то на отрезке $[x_0, x_1]$ опять находится не менее одного корня нечетной кратности уравнения $f(x) = 0$. Длина этого отрезка уменьшилась в два раза. Метод половинного деления состоит в повторении вычисления значения $f(x)$ в середине отрезка, и определении нового отрезка, содержащего корень. Если на некотором шаге вычислений длина отрезка $[x_0, x_1]$ станет меньше 2ε , то его середину можно принять за приближенное значение корня, вычисленное с точностью ε .

При расчете метод дихотомии прост и надежен, гарантирует точность ответа, сходится для любой непрерывной функции. Если при поиске перемены знака сравнивать $\text{sgn}(f(x_0))$ и $\text{sgn}(f(c))$, то он устойчив к ошибкам округления. Скорость сходимости метода невелика и, если на отрезке имеется более одного простого корня, то неизвестно какой из корней вычислен. Метод неприемлем для поиска корней четной кратности и не обобщается на случай решения системы нелинейных уравнений.

Можно изменить алгоритм метода дихотомии, выбирая в качестве пробной точки c точку пересечения прямой

$$\frac{y - f(x_0)}{f(x_1) - f(x_0)} = \frac{x - x_0}{x_1 - x_0}$$

с осью абсцисс

$$c = x_0 - \frac{f(x_0)(x_1 - x_0)}{f(x_1) - f(x_0)}.$$

Такой метод выбора c часто сходится быстрее метода дихотомии.

2. Метод простой итерации

Суть метода итерации состоит в выборе некоторого начального приближения $x^{(0)}$ и построении алгоритма вычисления последовательности приближений (итераций, от латинского слова “итерацио” – повторение) $x^{(1)}, x^{(2)}, \dots, x^{(n)}, \dots$, сходящейся к искомому решению x^* уравнения (1.1).

Чтобы построить алгоритм вычисления n -ой итерации, можно (1.1) заменить эквивалентным уравнением

$$x = g(x).$$

Тогда n -ая итерация вычисляется через $(n - 1)$ -ую по формуле

$$x^{(n)} = g(x^{(n-1)}), \quad n = 1, 2, \dots \quad (1.2)$$

Если функция $g(x)$ имеет ограниченную производную $|g'(x)| < 1$, то метод простой итерации сходится и является методом первого порядка.

Для практической (вычисляемой) оценки погрешности используется формула

$$\frac{|x^{(n)} - x^{(n-1)}|}{1 - q} \leq \varepsilon, \quad \text{где} \quad q \approx \frac{|x^{(n)} - x^{(n-1)}|}{|x^{(n-1)} - x^{(n-2)}|} \quad (1.3)$$

При выполнении этого условия можно $x^{(n)}$ можно принять за вычисленное с точностью ε приближенное решение уравнения (1.1). В этом случае полезно выдать информацию,

Рис. 1: Здесь изображены случаи, когда $x^{(n)}$ не является приближенным с точностью ε решением при выходе „по шагу итерации“ (первый рисунок), а на втором рисунке при выходе по „по невязке“.

что выход из итерационного процесса произошел „по шагу итерации“. Можно прекратить вычисление итераций „по невязке“, когда будет выполнено условие $|x^{(n)} - g(x^{(n)})| \leq \varepsilon$. Необходимость использования двух условий для выхода из итерационного процесса видна из рис. 1. Слева на рис. 1 изображена ситуация, когда $x^{(n)}$ не является приближенным с точностью ε решением при выходе „по шагу итерации“, а слева — при выходе по „по невязке“.

Успех метода простой итерации зависит от того, насколько удачно выбрана функция $g(x)$, чем меньше соответствующее ей значение константы q , тем быстрее сходится итерационный процесс (1.2).

Пример. Требуется вычислить корни уравнения

$$f(x) \equiv x^2 - a = 0$$

используя метод простой итерации

В качестве $g(x)$ можно выбрать $g(x) = \frac{a}{x}$ или $g(x) = \frac{1}{2} \left(x + \frac{a}{x} \right)$, и, соответственно, записать два итерационных процесса

$$x^{(n+1)} = \frac{a}{x^{(n)}}, \quad x^{(n+1)} = \frac{1}{2} \left(x^{(n)} + \frac{a}{x^{(n)}} \right).$$

Первый из этих процессов не сходится, так как $\max_x |g'(x)| \geq 1$ в окрестности x^* . Второй же сходится очень быстро при любом начальном приближении $x^{(0)} > 0$, так как $|g'(x^*)| = 0$.

Напомним, что при реализации любого итерационного метода надо ограничивать число итераций, так как он может расходиться.

3. Метод Ньютона В случае, когда известно начальное приближение достаточно близкое к искомому решению x^* , эффективным методом вычисления решения уравнения (1.1) ($f(x) = 0$) с заданной точностью ε является метод Ньютона. Идея метода Ньютона (иногда его называют методом линеаризации) заключается в том, что в окрестности имеющегося приближения x^n уравнение (1.1) заменяется вспомогательным линейным.

Для построения этого метода используем разложение $f(x)$ в ряд Тейлора в окрестности точного решения x^*

$$\begin{aligned} 0 &= f(x^*) = f(x^{(n)} + (x^* - x^{(n)})) = \\ &= f(x^{(n)}) + f'(x^{(n)})(x^* - x^{(n)}) + f''(\xi)(x^* - x^{(n)})^2/2. \end{aligned}$$

Отсюда получается приближенное равенство

$$f(x^{(n)}) + f'(x^{(n)})(x^* - x^{(n)}) \approx 0,$$

которое позволяет построить алгоритм вычисления $(n+1)$ -ого приближения с помощью n -ого приближения

$$f(x^{(n)}) + f'(x^{(n)})(x^{(n+1)} - x^{(n)}) = 0 \Rightarrow x^{(n+1)} = x^{(n)} - \frac{f(x^{(n)})}{f'(x^{(n)})}.$$

Такой способ вычисления $n+1$ -ой итерации называется методом Ньютона. Если расстояние $|x^0 - x^*|$ между начальным приближением x^0 и точным решением x^* достаточно мало, то метод Ньютона сходится. Он является методом второго порядка, т.е.

$|x^{(n)} - x^*| \leq C|x^{(n-1)} - x^*|^2$. Действительно, из расчетной формулы и разложение функции $f(x)$ в ряд Тейлора

$$f(x^{(n)}) + f'(x^{(n)})(x^{(n+1)} - x^{(n)}) = 0$$

$$f(x^{(n)}) + f'(x^{(n)})(x^* - x^{(n)}) + \frac{f''(\xi)}{2}(x^* - x^{(n)})^2 = 0.$$

следует

$$-f'(x^{(n)})(x^{(n+1)} - x^{(n)}) + f'(x^{(n)})(x^* - x^{(n)}) + \frac{f''(\xi)}{2}(x^* - x^{(n)})^2 = 0$$

$$\Rightarrow f'(x^{(n)})(-x^{(n+1)} + x^{(n)} + x^* - x^{(n)}) + \frac{f''(\xi)}{2}(x^* - x^{(n)})^2 = 0 \Rightarrow$$

$$\Rightarrow f'(x^{(n)})(x^* - x^{(n+1)}) = -\frac{f''(\xi)}{2}(x^* - x^{(n)})^2$$

Геометрически метод Ньютона можно интерпретировать как метод касательных.

Рис. 2: Слева показано как зависит сходимость метода Ньютона от выбора начального приближения. Если выбрано $x^{(0)}$, изображенное под осью Ох, то метод сходится, а для $x^{(0)}$ над осью Ох — расходится. Справа изображена ситуация, когда при реализации метода Ньютона происходит заикливание.

На рис.2 изображена зависимость сходимости метода Ньютона от выбора $x^{(0)}$, а справа нарисована неприятная ситуация заикливания.

Для метода Ньютона существуют хорошие теоретические оценки, которые требуют вычисление минимума первой производной и максимума второй. Иногда это трудно выполнить, а иногда и невозможно. Поэтому в качестве критерия окончания вычисления итераций обычно пользуются следующими неравенствами:

$$|f(x^{(n)})| \leq \varepsilon \quad \text{— выход по невязке или}$$

$$|x^{(n)} - x^{(n-1)}| \leq \varepsilon \quad \text{— выход по шагу итерации.} \quad (1.4)$$

Одновременное выполнение этих двух условий гарантируют точность ответа в подавляющем большинстве случаев. При практическом расчете разумно выдавать информацию о критерии выхода из итерационного процесса.

Основная трудность в использовании метода Ньютона состоит в выборе хорошего начального приближения. Несколько практических советов как это делать см. на стр.?

Замечание 1. Часто каждое значение функции $f(x)$, корень которой требуется найти, вычисляется с помощью громоздкого расчета. Ясно, что трудно вычислить первую производную такой функции для реализации метода Ньютона. Если заменить $f'(x)$ разностным соотношением, то получится метод Ньютона-Рафсона, скорость сходимости которого будет хуже сходимости метода Ньютона.

При замене $f'(x)$ разностным отношением возможна большая потеря точность, поэтому разумно положить

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

где $h = \begin{cases} \sqrt{\varepsilon}, & \text{если } |x| \leq 1, \\ x\sqrt{\varepsilon}, & \text{если } |x| > 1 \end{cases}$, ε — точность расчета.

Замечание 2. Иногда, для экономии времени расчета можно вычислять производную ни на каждом шаге. Тогда скорость сходимости метода ньютона уменьшиться (см. рис.), зато увеличиться скорость каждой итерации.

Замечание 3. Если нет уверенности, что начальное приближение x^0 достаточно близко к значению корня x^* . Тогда вычисления n -ую итерацию можно вычислять по формуле

$$x^{(n+1)} = x^{(n)} - \alpha \frac{f(x^{(n)})}{f'(x^{(n)})}, \quad (1.5)$$

где параметр α сначала считается равным единице. Если $|f(x^{(n)})| \leq |f(x^{(n-1)})|$, то n -ая итерация считается найденной. Если же $|f(x^{(n)})| > |f(x^{(n-1)})|$, то $\alpha := \alpha/2$ и n -ая итерация пересчитывается по формул (1.4). Значение α делить пополам можно ограниченное число раз, так как метод Ньютона может расходиться.

Пусть заданы начальное приближение $x^{(0)}$, максимально возможное число итераций IT и пусть составлены процедуры-функции для вычисления $f(x)$ и $f'(x)$. Тогда возможен такой вариант алгоритма метода Ньютона:

1. Начало
2. Ввод начального приближения x^0 к искомому значению корня x , IT — максимально возможное число итераций,
3. $it := 0$ — счетчик числа итераций, $x := x^0$,
4. $r := f(x)$, $it := it + 1$
5. $x_1 = x - r/f'(x)$ — следующая итерация метода Ньютона, $y_1 = f(x_1)$,
 $r_1 = |x - x_1|$ — модуль разности между двумя вычисленными итерациями,
6. Пока $|y_1| \geq \varepsilon$ и $|r_1| \geq \varepsilon$ и $it \leq IT$ в цикле вычисляются $x := x_1$, $r := f(x_1)$,
7. после выхода из цикла,

если $it > IT$, тогда печатается строка "вычислить за IT итераций значение
корня с точностью ε не удается."

иначе печатается значение корня x_1 и строка

если $|y_1| \leq \varepsilon$, то "выход по „невязке“ "

если $|r_1| \leq \varepsilon$, то "выход по „шагу итерации“ ".

8. Конец

5. Метод парабол

Метод парабол является трехшаговым, так как для начала расчета требуется знать три итерации $x^{(0)}$, $x^{(1)}$ и $x^{(2)}$. С помощью этих значений строится интерполяционный многочлен Ньютона с разделенными разностями

$$f(x) \approx L_2(x) = f(x^{(n)}) + (x - x^{(n)})f(x^{(n)}, x^{(n-1)}) + (x - x^{(n)})(x - x^{(n-1)})f(x_n, x^{(n-1)}, x^{(n-2)}),$$

$$n = 2, \dots$$

Пусть $z = x - x_n$ — наименьший по модулю корень квадратного уравнения

$$az^2 + bz + c = 0,$$

где $a = f(x^{(n)}, x^{(n-1)}, x^{(n-2)})$, $b = a(x - x^{(n-1)}) + f(x^{(n)}, x^{(n-1)})$, $c = f(x^{(n)})$. Тогда новое приближению к корню вычисляется по формуле

$$x^{(n+1)} = x^{(n)} + z$$

Метод парабол имеет важное достоинство: он позволяет при действительном начальном приближении вычислять комплексные корни вещественной функции $f(x)$. Критерий выхода из итерационного процесса такой же, как в методе Ньютона (1.4).

В методе парабол итерации сходятся медленнее чем в методе Ньютона, а вычисления более громоздкие. Кроме того, возможна большая потеря точности при вычислении разделенных разностей первого и второго порядка. Для страховки от потери точности можно применить прием Гарвика: выбрать не очень малое ε_1 (например, $\varepsilon_1 = 10^k \varepsilon$, $k \geq 1$) и вычислять итерации до выполнения неравенств (1.4) при $\varepsilon = \varepsilon_1$. Далее расчет продолжается, если $|x^{(n)} - x_{(n-1)}|$ убывает. Первое же возрастание этой величины означает рост погрешности вычислений (последняя итерация в этом случае обычно не используется).

4. Локализация корней

При практическом счете обычно трудно или невозможно проводить аналитический анализ функции $f(x)$ для отделения корней. Перечислим некоторые приемы локализации корней:

1. Очень надежным способом отделения корней функции $f(x)$ является графический, особенно после появления большого числа пакетов графических программ.

2. Если вычислителю из каких то соображений известен отрезок $[a, b]$, на котором находятся корни $f(x)$, то можно использовать поиск корня, вычисляя значения $f(x_i)$ в точках $x_i = a + i h$, $i = 0, 1, \dots$ с некоторым постоянным шагом h . Если $f(x_i) f(x_{i+1}) < 0$ то на $[x_i, x_{i+1}]$ находится, по крайней мере, один простой или нечетнократный корень.

При таком способе локализации корней пропускаются четнократные и близкие корни. В этой ситуации иногда помогает построение графика функции с помощью вычисленных значений.

Если известна оценка $\max_{[a,b]} |f'(x)| = M$, то для поиска корня используется переменный шаг (метод Рыбакова), значения $f(x)$ вычисляются в точках $x_0, x_i = x_{i-1} + |f(x_{i-1})|/M$, $i = 1, 2, \dots$. Можно показать, что, если $x_{i-1} < x^*$, то $x_i < x^*$. Так как последовательность x_i монотонная и ограниченная, то $x_i \rightarrow x^*$ при возрастании i . Если выбрать шаг h и вычислять значения $f(x)$ в точках $x_i = x_{i-1} + |f(x_{i-1})|/M + h$, то после перемены знака в точке x_{i+1} корень находится на отрезке $[x_i, x_i + h]$. Этот метод локализации корней может быть более эффективным, если отрезок $[a, b]$ разбить на части и находить M для каждого полученного отрезка.

3. Пусть требуется решать уравнение, содержащее параметр a ,

$$f(x, a) = 0,$$

и при некотором значении $a = a_0$ его решение x_0^* легко находится точно или с заданной точностью ε . Для поиска начального приближения при некотором значении $a = c$ можно использовать продолжение по параметру. Выбирается некоторое значение N и вычисляется шаг $h = (c - a)/N$. Решения x_i^* уравнения $f(x, a) = 0$ разыскиваются с точностью ε для значений параметра $a_i = a + ih$, $i = 1, 2, \dots, N$ и начального приближения x_{i-1}^* . При достаточно малом шаге h этот способ, как правило, приводит к успеху. Иногда объем вычислений можно уменьшить, используя линейную или квадратичную интерполяцию. Например, если вычислены значения x_{i-1}^* и x_i^* , то для $a = a_{i+1}$ с помощью линейной интерполяции в качестве начального приближения можно выбрать $x_{i+1}^0 = 2x_i^* - x_{i-1}^*$.

Замечание. Если функция $f(x)$ является многочленом n -ой степени, то существуют строгие теоремы о корнях многочленов (см., например, [6]). Эти теоремы позволяют найти отрезок $[a, b]$, содержащий корни многочлена. В этом случае легко построить график функции на $[a, b]$.

Список литературы

- [1] Бахвалов Н.С. Численные методы. -М.:Наука,1973.
- [2] Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы. - М.:Наука,1989г.
- [3] Калиткин Н.Н. Численные методы. -М.:Наука,1978
- [4] Самарский А.А., Гулин А.В. Численные методы. -М.:Наука,1989.
- [5] Вержбицкий В.М. Основы численных методов.- М.:Высш.шк.,2002.
- [6] Демидович Б.П., Марон И.Л. Основы вычислительной математики.М.:Наука,1966.
- [7] Дж. Ортега, У. Пул Введение в численные методы решения дифференциальных уравнений.
- [8] Форсайт Дж., Малькольм М., Моулер К. Машинные методы математических вычислений. - М.: Мир, 1960.
- [9] Беленькая Л.Х., Овчинникова С.Н. Вычислительная погрешность при расчетах на ЭВМ, Методические указания.. Ростов -на- Дону, УПЛ РГУ, 1994г., 27 с.
- [10] Воеводин В.В. Вычислительные основы линейной алгебры. -М.:Наука,1977.
- [11] Копченова Н.В., Марон И.А. Вычислительная математика в примерах и задачах.М.: Наука, Гл.ред.физ.-мат.лит., 1972.
- [12] Бахвалов Н.С., Лапин А.В., Чижонков Е.В. Численные методы в задачах и упражнениях. М., "Высшая школа ", 2000.
- [13] Крылов В.И., Бобков В.В. Вычислительные методы. Т.II, М.: Наука, 1977