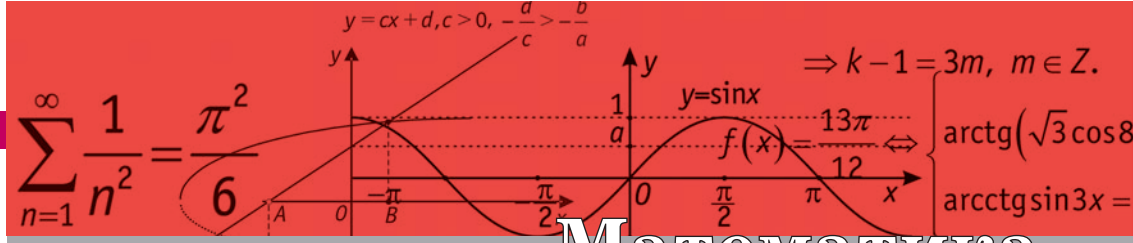


Беклемишев Дмитрий Владимирович
Доктор педагогических наук,
профессор кафедры высшей математики
Московского физико-технического института
(МФТИ).

Приближённые вычисления

В этой статье автор стремится показать основные трудности, которые возникают при приближённых вычислениях, и рассказать о некоторых подходах к их преодолению.

1. Введение

Вещественное число часто определяют как бесконечную десятичную дробь. Можно дать и другие определения, но мы будем исходить из этого, потому что, если мы хотим производить арифметические действия с вещественными числами, их нужно записать цифрами.

Прежде всего, нужно со всей определённостью сказать, что *бесконечности не бывает*. Бесконечных десятичных дробей нет, как нет бесконечных прямых линий, плоскостей, не имеющих толщины, как не бывает абсолютно твёрдых тел или невесомых нитей, о которых говорят физики. Всё это — объекты, не существующие в реальном мире, а применяющиеся для его приближённого и упрощённого описания. Как мы увидим, действия с бесконечными десятичными дробями во многом проще, чем действия при реальных вычислениях.

Для нас важно сейчас то, что нельзя выписать бесконечную десятичную дробь. Можно задать закон, по которому можно найти любой десятичный знак этой дроби, но выписывая цифры, где-то придётся остановиться. Более того, реально число цифр в десятичной дроби не может быть очень большим.

Можно, конечно, как это часто делается, обозначить вещественные числа какими-то символами, вроде π, $\sqrt{2}$ и т. д. и получать результаты вроде $\pi^2\sqrt{2}+2\sqrt{\pi}$, но употребить такой результат можно только для сравнения с ответом в задачнике. Для любых практических целей его нужно приближённо представить конечной десятичной дробью.

Соотношение точных и приближённых вычислений хорошо видно, если сравнить геометрию и черчение. Геометр рассматривает на бесконечной плоскости линии, не имеющие ширины. Линии, проведённые чертёжником на листе бумаги, обязательно имеют конечную длину и ненулевую ширину. На чертеже не параллельные прямые могут не пересекаться, пересечение прямых — точка, которую можно приближённо описать, как ромб с высотой, равной ширине прямой, а длины диаго-

налей этого ромба зависят от угла между прямыми.

Геометр говорит, что диагонали параллелограмма, пересекаясь, делятся пополам. Чертёжник с этим согласен, но у него всё сложнее. Чем ближе начерченный им четырёхугольник к параллелограмму, тем ближе точка пересечения к середине. Хорошо бы добиться того, чтобы середина какого-нибудь геометрического отрезка с концами в нарисованных вершинах четырёхугольника л е жала внутри «точки» пересечения нарисованных диагоналей.

Конечно, тут нет речи о вычислениях, но происхождение трудностей в точности то же самое. Геометрия — идеализация реальных чертежей так же, как операции с вещественными числами — идеализация реальных вычислений, как бы они ни делались, на бумаге или с помощью вычислительной машины.

Ещё одно следствие того, что бесконечности не бывает — это ограниченность памяти для записи промежуточных результатов вычислений. Роль памяти в вычислениях очень ярко в и д н а , когда учишь дошкольника складывать двузначные числа, скажем, 12 и 25. Однозначные он складывать умеет и понимает, что нужно сначала сложить числа десятков $1 + 2$, а затем числа единиц $2 + 5$. Но когда результат 7 получен, число 3 уже забыто, и ничего не получается. Конечно, вычислительная машина складывает лучше и быстрее, но данных и промежуточных результатов использует гораздо больше, и памяти может оказаться недостаточно.

Представьте себя перед классной доской. Вы решаете задачу и, если вычисления не слишком громоздки, не чувствуете ограничений. Но, если задача большая (или доска — маленькая), придётся что-то стирать. Можно ли что-нибудь стереть без ущерба? Если нет, то приходится идти за тетрадкой и переписывать в неё.

У вас «запоминающие устройства» трёх уровней: самое быстрое, но и самое малое по объёму — это то, что вы держите в уме. Доска — более медленное устройство, но по объёму больше. Тетрадка имеет самый большой объём, но требует больших затрат времени.

В вычислительной машине также применяются запоминающие устройства нескольких уровней. Чем выше уровень, тем быстрее можно получать данные из памяти и записывать их в память. Но тем меньше объём этого раздела памяти.

Вычисления должны быть закончены за определённое время, и переписывание данных на более ёмкие носители может слишком их замедлить.

Проблемы с использованием памяти относятся больше к программированию, но ограниченность времени сказывается не только на выборе уровней памяти. Более существенное отличие от обычной математики состоит в ограниченности числа повторений операций, которые мы выполняем. Математика полна рассуждений вроде следующего: «Продолжая дальше таким же образом, мы придём к нужному результату». Возникает вопрос — когда? Не сделать в срок, значит, вообще не сделать.

Но, может быть, за приемлемое время мы можем прийти хотя и не к точному, но удовлетворительному результату. Результат может быть достаточно точным или недостаточно точным.

В этой заметке мы обсудим, к чему при вычислениях приводят перечисленные выше следствия из того, что бесконечности не бывает.

2. Источники ошибок

Округления. Итак, пусть для записи каждого числа мы можем использовать не больше, чем m цифр (десятичных или двоичных — не важно, пусть десятичных). Ясно, что мы сможем выразить только конечное, хотя, может быть, и очень большое количество чисел. Числовая ось превращается в россыпь отдельно стоящих точек. Как распорядиться m цифрами с тем, чтобы записываемые ими числа были расположены на как можно более длинном отрезке и располагались на нём по возможности более равномерно? Об этом можно поговорить в другой раз, а сейчас остановимся на том, к чему приводит необходимость обходиться только конечными десятичными дробями, число знаков в которых ограничено. Будем считать, что больше, чем m десятичных знаков, мы записать не можем.

Когда среди данных или в процессе вычислений появляется число, которое следовало бы записать бо́льшим числом знаков, то это число по определённому правилу заменяют таким числом, которое с помощью m знаков записать можно. Это называется *округлением*. Обычно числа округляются до m значащих цифр так: если $m{+}1$-я цифра меньше 6, то она и все последующие заменяются нулями, в противном случае после замены лишних цифр на нули к числу прибавляется 10^{-k}, если m-я цифра стоит на k-м месте *после* запятой. Если же m-я цифра стоит на l-м месте *перед* запятой, то прибавляется 10^{l-1}.

Почему в процессе вычислений может возникнуть необходимость в округлении? Пусть мы перемножаем две десятичные дроби. Их произведение имеет больше, чем m десятичных знаков, и должно быть округлено.

Если результат арифметической операции, например, умножения, округляется, то он определяется не только сомножителями, но и правилом округления. Поэтому арифметические операции, выполняемые вычислительной машиной, обладают совсем иными свойствами, чем соответствующие математические операции. Например, сложение не обладает свойством ассоциативности: $a + (b + c) \neq (a + b) + c$.

Действительно, пусть мы округляем числа до четырёх значащих цифр. Если к 1234 мы прибавим 0,3, то 1234,3 придётся округлить до 1234. Прибавив затем к этому числу 0,4 и округлив, мы снова получим 1234. Если же мы сначала сложим 0,3 и 0,4, а затем их сумму 0,7 прибавим к 1234, то округление суммы 1234,7 даст 1235.

В последнем примере максимальная ошибка при округлении равна 0,5. Если бы мы округляли число $0{,}1234 = 1234 \cdot 10^{-4}$, то максимальная ошибка составила бы $0{,}5 \cdot 10^{-4}$. Таким образом, ошибка при округлении числа при умножении его на 10 также увеличивается в десять раз. Это и разумно: ошибка при округлении большого числа может быть больше. Нет смысла, скажем, измерять расстояние от Москвы до Петербурга с точностью до метра.

Влияние ошибок округления. Если с помощью вычислительной машины мы хотим произвести какое-

нибудь математическое вычисление, то возникает задача оценить его точность. При этом следует иметь в виду три момента.

1. Ответ на практический вопрос, который мы хотим получить.
2. Решение математической задачи, к которой мы свели этот вопрос.
3. Результат численного решения математической задачи.

Все они различны. Второе, как правило, отличается от первого, даже если математическая задача решена безупречно. Причина в том, что сводя реальную проблему к математической задаче, мы неизбежно упрощаем её. Это различие — важный, но отдельный вопрос. Сейчас мы говорим о различии численного и математического (или, как иногда говорят, *аналитического*) решения.

Трудность в том, что оно неизвестно, в противном случае численное решение не понадобилось бы или было бы совсем простым. Естественный путь — это проследить за всеми выполняемыми операциями и оценить вклад округления при каждой из них в общую ошибку. Этот процесс, называемый *прямым анализом ошибок округления*, является сложным и трудоёмким делом потому, что свойства операций с округляемыми числами сильно отличаются от соответствующих свойств операций с вещественными числами, и приходится рассматривать длинные последовательности таких операций. Кроме того, прямой анализ ошибок округления может приводить к сильно завышенным оценкам возможных ошибок, которые существенно больше того, что можно наблюдать при экспериментах.

Для того, чтобы сделать наглядной причину этого, представим себе, что мы хотим измерить линейкой длиной в двадцать сантиметров шестиметровое бревно. Откладывая её и отмечая очередные 20 сантиметров, мы каждый раз ошибаемся на 2 мм, и на тридцати повторениях этой операции можем накопить ошибку в 6 см. Реально же общая ошибка окажется меньше потому, что ошибка при каждом прикладывании линейки может как увеличивать, так и уменьшать результат, что при сложении приведёт к уменьшению общей ошибки.

Примерно то же самое происходит и с ошибками округления при вычислениях. Поэтому широко применяется такой подход. Допустим, что необходимо получить результат, в котором будет 4 верных десятичных знака. Если производить вычисления, учитывая 10 знаков, то четыре младших (принято говорить, что в десятичной дроби цифры, стоящие правее, младше цифр, стоящих левее) знака будут искажены ошибками округления, а четыре старших останутся чистыми. Их мы и оставим при окончательном округлении результата и получим ответ, который *можем считать* результатом, имеющим требуемую точность. Я нарочно подчеркнул эти слова, так как

никаких оснований, кроме веры, что всё будет хорошо, для этого нет.

Более того, может случиться и случается, что полученный результат не имеет *ни одной* верной цифры. Таким образом, при решении ответственных задач полагаться на оптимистические ожидания нельзя. Задача вычислительной математики — разобраться в причинах таких явлений и научиться производить вычисления так, чтобы подобных ситуаций не возникало, или, по крайней мере, чтобы неверный результат не был воспринят как верный.

Причины затруднений. Перечислим и вкратце разберём эти причины.

- Плохая обусловленность задачи.
- Неустойчивость алгоритма.
- «Катастрофическое падение точности».

Потеря точности. Начнём с последнего. Пусть мы вычитаем одно из другого два близких по величине числа. Каждое из них имеет младшие цифры, «загрязнённые» ошибками округления, и «чистые» старшие цифры. «Чистые» части этих дробей совпадают или почти совпадают, поэтому при вычитании они взаимно уничтожаются, и оказывается, что разность не содержит «чистых» цифр, или содержит их меньше, чем необходимо для достижения нужной точности результата.

Ещё один случай такого рода хорошо виден на следующем примере. Рассмотрим систему

$$\begin{cases} -10^{-4}x + y &= 1, \\ x + y &= 2, \end{cases}$$

и допустим, что арифметические операции производятся с округлением результата до четырёх значащих цифр.

Исключим y, вычитая второе уравнение из первого. Вычисленное без округления решение системы можно записать с помощью формулы суммы геометрической прогрессии

$$x = (1 + 10^{-4})^{-1} = 1 - 10^{-4} + 10^{-8} - \ldots$$
$$y = 2 - x = 1 + 10^{-4} - 10^{-8} + \ldots$$

Округлённое до четырёх значащих цифр решение $x = 1; y = 1$.

Допустим, что кто-то захотел решить систему иначе: исключить x. Для этого нужно умножить первое уравнение на 10^4 и прибавить ко второму. Получится система

$$\begin{cases} -x + 10^4 y &= 10^4, \\ 10001y &= 10002. \end{cases}$$

После этого полученные числа округляются до четырёх значащих цифр. Второе уравнение превращается в $10^4 y = 10^4$. Заметим, что число 2 после прибавления к нему 10^4 и округления исчезло, и если бы в исходной системе там стояло 3, а не 2, результат был бы точен.

Из полученного второго уравнения $y = 1$, и в силу первого уравнения $x = 0$. Найденное решение есть $x = 0$, $y = 1$. Это далеко от истинного решения.

Если бы мы увеличили точность, округляя числа до пяти или больше значащих цифр, то получили бы при любом способе решения верный ответ

$$x = \frac{10000}{10001}, \qquad y = \frac{10002}{10001},$$

округлённый до соответствующего числа знаков.

Из этого примера видны два важных обстоятельства.

Во-первых, получение совершенно неверного решения по причине округлений возможно даже при совсем простых вычислениях.

Во-вторых, этого явления можно избежать, если организовать вычисления

иначе, или увеличить число учитываемых десятичных знаков.

Человек-вычислитель в этом примере, конечно, не будет исключать x. Другое дело машина, которая выполняет заранее предписанную программу. Если там сказано, что нужно сначала исключить первую переменную, она так и сделает. Поэтому настоящие программы, решающие системы линейных уравнений при помощи последовательного исключения неизвестных, содержат предварительный просмотр коэффициентов, на основе которого выбирается переменная, подлежащая исключению.

Число учитываемых десятичных знаков определяется конструкцией вычислительной машины, и его дальнейшее увеличение возможно только путём составления специально для этого предназначенных программ, причём время счёта по таким программам значительно увеличивается. Практического значения этот приём не имеет.

Неустойчивость алгоритма. Теперь рассмотрим понятие устойчивости алгоритма. Под *алгоритмом* понимается полный и однозначный перечень действий, которые должен проделать вычислитель для получения искомого результата. Алгоритм, изложенный на языке, который может быть понят программным обеспечением вычислительной машины, превращается в программу, если к нему добавить процедуры ввода и вывода данных и другие необходимые дополнения.

Алгоритм может быть правильным, то есть давать в любых случаях точный результат при условии точного выполнения арифметических операций. При этом, однако, может оказаться, что при вычислениях с округлениями он приводит (или может привести) к результату, ошибка которого столь велика, что делает результат совершенно бесполезным. Такой алгоритм называется *численно неустойчивым*.

То, что в приведённом выше примере решения системы линейных уравнений возникла катастрофическая потеря точности, свидетельствует о неустойчивости алгоритма исключения. Для того, чтобы сделать его устойчивым, в него нужно ввести изменения, предотвращающие подобные ситуации, о чём и было сказано выше.

Часто неустойчивость объясняется тем, что алгоритм состоит из многих последовательных этапов вычислений, причём каждый следующий этап использует результат предыдущего. При этом ошибка, содержащаяся в результате предыдущего этапа, увеличивает ошибку, совершаемую на следующем этапе.

Выше мы обсуждали измерение бревна при помощи маленькой линейки. Этот алгоритм в значительно меньшей мере устойчив, чем измерение рулеткой. Если линейка будет прикладываться так, что каждый раз ошибка совершается в сторону увеличения, ошибка будет накапливаться.

Обусловленность. Перейдём к обсуждению понятия обусловленности задачи. Начнём со следующего примера. Пусть мы решаем систему из двух линейных уравнений графически. Как известно, каждое линейное урав-

нение на координатной плоскости определяет прямую линию. Точка пересечения прямых, которые определяются уравнениями системы, имеет координаты, удовлетворяющие обоим уравнением, то есть являющиеся решением системы. Но это — в математике. Если мы попробуем начертить прямые, чертёж, конечно, будет содержать отклонения, по меньшей мере, порядка толщины проводимых линий. Если мы учитываем толщину линий и ошибки при их построении, то их «точка пересечения» должна рассматриваться как пятно, форму которого приближённо можно описать как параллелограмм. И координаты любой математической точки из этого пятна с одинаковым основанием могут быть приняты за решение.

Посмотрим, от чего зависят размеры пятна. От точности чертежа, конечно. Но если она фиксирована, они зависят от угла между прямыми. Если угол уменьшается, то пятно вытягивается. Это означает, что при фиксированной точности вычислений точность, с которой может быть получено решение, может быть существенно различной для двух внешне сходных задач.

Про ту из этих задач, решение которой определяется с бóльшей ошибкой, говорят, что она хуже обусловлена.

Пусть нам нужно найти значение функции $y = f(x)$ при $x = a$, причём число a не может быть представлено в машине точно. Результат его округления обозначим a'. Естественно в качестве приближённого значения для $f(a)$ принять вычисленное значение $f(a')$. Допустим даже для упрощения, что $f(a')$ вычислено точно. В какой мере допустима такая замена? Для этого нужно, чтобы разность $f(a') - f(a)$ по абсолютной величине не превосходила допустимую ошибку результата, которую мы обозначим ε. Число $|a' - a|$ обозначим δ. Читатель, изучивший понятия предела функции и непрерывности функции, несомненно, узнает своих старых знакомых, но здесь они действуют в других ролях. Значение ε фиксировано, а δ не может быть сделано «сколь угодно малым». Поэтому непрерывности функции совсем не достаточно: нужно, чтобы при значениях x, близких к a, значение функции не менялось слишком сильно. Чем сильнее его изменения, тем хуже обусловлена задача.

В качестве примера рассмотрим задачу извлечения квадратного корня из числа a. Пусть сначала $a \geqslant 2$. Его округлённое значение a' можно считать бóльшим единицы, тогда

$$|\sqrt{a'} - \sqrt{a}| = \frac{|a' - a|}{\sqrt{a'} + \sqrt{a}} < |a' - a| = \delta.$$

Мы видим, что задача обусловлена прекрасно: погрешность результата меньше, чем погрешность данных. Дело меняется, если речь идёт об извлечении корня из малого числа. Пусть для упрощения $a' < a$. Тогда

$$|\sqrt{a'} - \sqrt{a}| = \frac{|a' - a|}{\sqrt{a'} + \sqrt{a}} \geqslant \frac{|a' - a|}{2\sqrt{a}} = \frac{\delta}{2\sqrt{a}}.$$

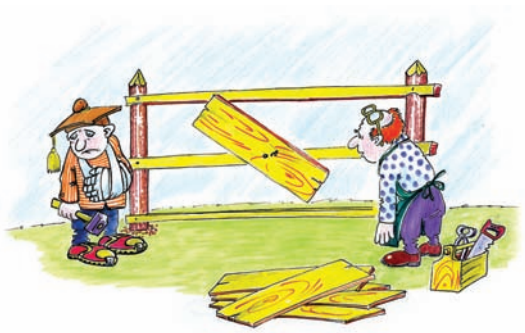
Задача обусловлена плохо: если $a = 0{,}01$, а $\delta = 0{,}005$, то $|\sqrt{a'} - \sqrt{a}| \geqslant 0{,}25\sqrt{a}$. Мы видим, что погрешность результата может по абсолютной величине оказаться больше, чем 25% этого результата.

Для того, чтобы сделать наглядной причину этого различия обусловленности в зависимости от a, нарисуйте график функции $y = \sqrt{x}$ и посмотрите, как пологая при больших x кривая круто спускается к нулю при малых x.

Как добиться удовлетворительного результата при решении плохо обусловленной задачи? Ответ один — нужно уточнить постановку задачи, вернувшись к исходной проблеме реаль-

ного мира и её математическому описанию. Действительно, плохая обусловленность означает, что малое изменение исходных данных существенно влияет на вычисленный результат, и дело тут не в методе решения, а в самой задаче.

Чтобы подчеркнуть это, рассмотрим ещё один пример. Математик вызвался помочь плотнику и получил задание прибить доску двумя гвоздями. Математик уверен, что две точки определяют прямую, и, чтобы не ходить далеко, забивает оба гвоздя поблизости от середины.



Результат неудовлетворительный: доска заметно качается, так как небольшое усилие на её конце создаёт большие силы, приложенные к точкам крепления, и гвозди немного сгибаются. В свою очередь, их малое смещение вызывает значительное перемещение концов доски.

Задача определения положения отрезка прямой по двум точкам оказалась плохо обусловленной, если расстояние между точками мало по сравнению с длиной отрезка: малое изменение исходных данных (положения точек) приводит к значительным изменениям полученного решения.

Хорошая или плохая обусловленность — неотъемлемое качество любой математической задачи, которое обязательно должно учитываться при любых рассуждениях о точности её численного решения.

Можно заметить, что во всех рассуждениях о точности решений употребляются слова совсем не из математического лексикона: «малое», «значительное», «неудовлетворительный»... Это и понятно, мы говорим сейчас не о математике, а о совсем другой науке. Если есть какое-то утверждение, например, начинающееся с слов «если a мало...», то оно справедливо настолько, насколько мы согласны считать данное конкретное значение a малым.

Математиками придумана шуточка, что математик отличается от инженера тем, что математик не знает, много это или мало — погрешность в 1%, а инженер знает, что мало. Эта шуточка верна только в своей первой части, если речь идёт о хорошем инженере. Он в любом случае должен знать, велика ли погрешность. Это требует высокой квалификации и в практически важных случаях влечёт за собой серьёзную ответственность.

3. Невязка и обратная устойчивость

Оценка точности. Как же оценить точность полученного решения? Для простоты и определённости будем говорить об уравнении $f(x) = 0$ и его решении, которое является одним числом, но сказанное ниже верно (с некоторыми усложнениями) и для любых других задач. Естественно сравнить полученное решение x_1 с истинным решением x_0. Величина $|x_1 - x_0|$ называется *ошибкой* или *погрешностью*. Но эта характеристика точности полученного решения применима больше для теоретических рассуждений, так как x_0 нам обычно неизвестно. Можно сделать «проверку»: подставить x_1 в уравнение

и посмотреть, получится ли нуль. Нуля, естественно, не получится, результатом подстановки будет число $d = f(x_1)$, которое называется *невязкой*.

Ясно, что малая невязка — это хорошо, но в какой мере малая невязка гарантирует малую погрешность? Ответ на этот вопрос совсем не однозначен. Для плохо обусловленной задачи малая невязка может быть и при большой погрешности.

Например, точное решение системы линейных уравнений

$$\begin{cases} x + y = 1, \\ x + 1{,}001y = 1{,}001 \end{cases}$$

— это $x_0 = 0$, $y_0 = 1$. Однако пара чисел $x_1 = 1$, $y_1 = 0$ даёт невязку в первом уравнении 0, а во втором 0,001. Система плохо обусловлена: её уравнения определяют прямые, которые пересекаются под очень малым углом (вспомним сказанное о графическом решении систем). Изменив на 0,1% коэффициенты при x и y во втором уравнении, можно получить систему, точным решением которой является пара чисел $x_1 = 1$, $y_1 = 0$.

Очень часто способ решения задачи состоит в уменьшении невязки, и его результатом является приближённое решение, дающее малую невязку. Не ограничиться ли этим решением, не интересуясь, насколько оно близко к теоретическому точному решению? Это зависит от цели, с которой решается задача. К этому вопросу мы ещё вернёмся, а сейчас рассмотрим ещё один подход к оценке точности решения.

Обратный анализ ошибок. Он основан на следующем тезисе: приближённое решение задачи является точным решением некоторой другой задачи. «Эту задачу мы решить не смогли, но зато решили другую!» У математика это вызовет только насмешливую улыбку. Но вычислитель не столь высокомерен, он заинтересуется, насколько решённая задача близка к исходной.

Пусть нам нужно решить уравнение $3x = 1$, округляя числа до трёх десятичных знаков. Точное решение $x_0 = 1/3$ будет представлено десятичной дробью $x_1 = 0{,}333$. Это число является точным решением уравнения $3x = 0{,}999$, свободный член которого отличается от свободного члена исходного уравнения на 0,001. Новое уравнение по отношению к исходному называется *возмущённым*.

Представим себе, что коэффициент и свободный член исходного уравнения — физические величины, полученные измерениями, точность которых $\pm 1\%$. В этом случае наше решение — всё, о чём можно было мечтать. Действительно, при такой точности измерения нет никакой гарантии того, что свободный член равен 1, а не 0,999, и нет никакого смысла увеличивать точность, приближаясь к теоретическому решению 1/3.

Если возмущённая задача достаточно близка к исходной, то решение считается удовлетворительным.

Оценка точности вычисленного решения исходной задачи с помощью количественной оценки отличия возмущённой задачи от исходной называется *обратным анализом ошибок округления*.

Уточним сказанное. Пусть решается задача с исходными данными a, и е ё решение без ошибок округления есть x. Производя округления, мы получаем результат x_1. Здесь a, x и x_1 могут обозначать не отдельные числа, а целые наборы чисел, как это имеет место, например, при решении систем уравнений. Если x_1 является точным решением такой же задачи с исходными данными $a + d$ и d мало, то алгоритм решения задачи называется *обратно устойчивым*.

Разработчики алгоритмов всегда заботятся о том, чтобы созданный ими алгоритм был обратно устойчивым. Это основной способ достижения достаточной точности решения.

4. Как решить неразрешимую задачу

Как было сказано, в ряде случаев нет смысла искать приближения к точному решению, добиваясь близости к теоретическому решению, а можно ограничиться приближённым решением, дающим малую невязку. Это соображение имеет далеко идущие следствия. Именно при таком подходе точное решение отходит в тень, и неважно, существует оно или нет. Если задача не имеет решения, то получить нулевую невязку в принципе невозможно, но и в лучшем случае её получение возможно только в принципе. Разница невелика. Это открывает путь к «решению» задач, не имеющих решения.

Султан был в гневе. Он созвал своих мудрецов, и сказал, что через три дня казнит всех, если задача не будет решена. Возразить, что решения не существует, никто не решился. Что было делать? Было найдено решение, дающее минимальную из всех возможных невязок. За три дня султан остыл, принял предложенное решение, и всё кончилось благополучно. С тех пор так и повелось.

В современном мире это выглядит так. Пусть из физических соображений можно считать, что в некоторой области их изменения величины y и x связаны линейной зависимостью вида $y = kx + b$, а коэффициенты должны быть установлены экспериментально. Экспериментальные данные представляют собой m точек на координатной плоскости $(x_1; y_1),\ldots,(x_m; y_m)$. Если эти пары значений действительно связаны искомой зависимостью, то подстановка их в уравнение приводит нас к системе из m линейных уравнений для двух неизвестных k и b:

$$y_i = kx_i + b, \qquad i = 1,\ldots,m.$$

Любая пара различных точек $(x_i; y_i)$ и $(x_j; y_j)$ определяет прямую. Но другая пара точек определяет другую прямую, и у нас нет оснований выбрать какую-нибудь одну из всех этих прямых.

Не опровергают ли такие экспериментальные данные исходные «физические соображения»? Однозначного ответа тут быть не может, он зависит от того, насколько мы склонны доверять этим данным. В какой мере исходные данные совместны с гипотезой о том, что y — линейная функция от x,

решается с помощью статистического анализа.

Пусть точность исходной информации допускает существование линейной зависимости между переменными. В этом случае то, что в действительности нужно — это найти такую прямую $y = k_0x + b_0$ на координатной плоскости, которая, может быть, не проходит ни через одну пару экспериментальных точек, или даже ни через одну из точек, но в каком-то смысле как можно более близко расположена ко всем точкам.

Обычно в этой задаче удалённость точки от прямой измеряют не расстоянием, а разностью ординат $y_i - kx_i - b$, и выбирают прямую так, чтобы сумма квадратов всех таких разностей была минимальна. Это означает, что должна быть минимальна сумма квадратов всех m чисел, составляющих невязку в этой задаче. Этот минимум — *«остаточная сумма квадратов»* — одна из исходных величин в упомянутом выше статистическом анализе.

Коэффициенты k_0 и b_0 уравнения так построенной прямой дают некоторое решение стоящей перед нами задачи, которое отнюдь не является решением системы линейных уравнений (вообще не имеющей решений). Любопытно отметить, что прямая $y = k_0x + b_0$ проходит через центр тяжести системы, состоящей из всех заданных точек.

Можно считать числа k_0 и b_0 *обобщённым решением* системы или, как говорят, *псевдорешением*.

Метод нахождения обобщённых решений, основанный на минимизации суммы квадратов невязок, получаемых во всех уравнениях, называется *методом наименьших квадратов*.

Практические вопросы, приводящие к математическим задачам, не имеющим решений, встречаются часто. Типичным является поиск компромисса между несовместимыми требованиями. Сразу приходят на ум конфликтные политические ситуации, но они редко могут быть формализованы и сведены к математическим задачам. Более реальный пример — противоречие между весом и прочностью конструкции.

Метод наименьших квадратов может применяться гораздо шире. Общую область его применения можно охарактеризовать как «подгонку данных».

Пусть, например, какая-то практическая задача сводится к системе уравнений, не обязательно линейных или даже алгебраических. В любом случае имеются в наличии исходные данные («известные») и результаты («неизвестные»). Уже много раз говорилось, что известные величины известны не точно, а с некоторой неопределённостью, вносимой ошибками измерений и другими источниками. Но и неизвестные не являются полностью неизвестными: исходя из постановки задачи и просто из здравого смысла, можно указать, в каких пределах могут находиться значения неизвестных. Если в результате расчётов будет найдено, что температура больного равна 137,73°, то дальше мусорной корзины такие результаты не пойдут, сколь бы солидными расчёты ни выглядели.

Таким образом, различие между известными и неизвестными становится не качественным, а количественным — оно в размере допустимого интервала значений. Особенно это относится к таким областям науки, где теория продвинута слабо, а экспериментальная база недостаточна.

Для того, чтобы использовать все данные, которыми мы располагаем об известных и неизвестных величинах, можно, не делая между ними принципиального различия, попытаться со-

кратить норму невязки. Вполне возможно, что абсурдный ответ, полученный прямым расчётом, был результатом неустойчивости в расчётах, и изменением данных в допустимых пределах можно получить приемлемый результат. Конечно, такой поворот дела вызывает необходимость пересмотра данных и подхода к задаче, но в любом случае это продвижение.

Ещё один вопрос, в котором возникает необходимость в подгонке данных — это согласование имеющихся данных с поступающими новыми. Представим себе, что данные о каком-то наборе объектов получены путём расчёта и результатов измерений других величин. Через некоторое время измерения повторяются, причём, возможно, добавляются новые объекты. Новые данные не совпадают со старыми, но и те, и другие измерения не заслуживают безусловного доверия, так как, естественно, содержат ошибки. То, что в этом случае необходимо, — это выбор компромиссного решения, каким может оказаться уменьшение суммы квадратов поправок, вносимых в старые данные.

Именно в такой задаче — согласовании новых данных геодезических измерений со старыми — и был впервые применён К. Ф. Гауссом метод наименьших квадратов.

5. Итерационные процессы

Перейдём теперь к последней проблеме приближённых вычислений, упомянутой во введении, а именно, к невозможности бесконечно повторять какие-либо действия.

Есть известная геометрическая задача — найти середину отрезка с помощью циркуля и линейки. Её решение даёт математически точный результат, точный до тех пор, пока мы имеем дело с прямыми, не имеющими ширины. Посмотрим, как выглядит практическое решение подобной задачи.

Пусть нам нужно найти середину бревна. Для практических задач характерно, что приступая к их решению, мы можем выбрать один из многих способов решения, и при выборе приходится учитывать имеющиеся в нашем распоряжении средства. В данном случае есть два хороших способа: измерить бревно рулеткой и разделить длину пополам или можно натянуть вдоль бревна тесёмку и сложить её вдвое.

Конечно, для этой задачи бревно можно рассматривать как отрезок, но даже если мы сможем приспособить что-то вместо циркуля, геометрический способ явно не выдерживает конкуренции с двумя названными выше.

Полезно посмотреть на ещё один способ, который поначалу выглядит не очень выигрышно. Пусть мы делим пополам отрезок AB. Возьмём какой-нибудь отрезок, неважно, больший или меньший половины AB, и отложим его от концов AB. Получим точки A_1 и B_1, причём середина нового отрезка A_1B_1 совпадает с серединой исходного. Поступим с отрезком A_1B_1 так же, получим отрезок A_2B_2 и будем повторять это действие.

Длина отрезка будет с каждым повторением уменьшаться, и с точки зрения математика, в пределе концы отрезка A_kB_k сойдутся к середине отрезка AB. Но за конечное число повторений середина достигнута не будет, если случайно не повезёт.

Такого рода вычислительные процессы, когда строится последовательность, имеющая пределом искомый результат, называются *итерационными*. Как отнесётся плотник к итерационно-

му процессу деления бревна? Как ни странно, совсем не плохо. При некотором опыте и глазомере уже второй отрезок A_1B_1 окажется настолько мал, что разделив его опять на глаз, он получит достаточно точный результат. Какой результат можно считать достаточно точным — отдельный вопрос, а сейчас важно остановиться на важнейшей характеристике итерационных процессов — скорости сходимости.

Бесконечности нет, и нет смысла гнаться за точным результатом. Гораздо важнее, сколько потребуется повторений для получения приемлемого результата. Если бы плотнику потребовалось 5 или 6 итераций, он пошёл бы в бытовку за рулеткой. Но если процесс сходится на второй итерации, можно обойтись первой попавшей под руку планкой. Вывод такой: эффективность итерационного метода определяется двумя факторами — затратами труда (или времени) на каждую итерацию и числом итераций, необходимых для получения приемлемого результата.

Ещё один пример, более относящийся к вычислительной математике. Пусть мы хотим найти корень многочлена $p(x)$. Для этого можно воспользоваться методом *бисекции* или, проще говоря, деления пополам. Именно, если мы знаем отрезок числовой оси $[a;b]$ такой, что на его концах значения многочлена имеют разные знаки: $p(a)\cdot p(b)<0$, то (как видно из графика (им можно строго доказать), на этом отрезке есть хоть один корень многочлена. Он отстоит от середины отрезка c не больше, чем на половину длины отрезка. Значит, если мы примем c за приближённое значение корня, погрешность будет не больше, чем $(b-a)/2$. Если $p(c)=0$, то корень найден, но это было бы исключительным везением. Если $p(c)\neq 0$, то либо $p(a)\cdot p(c)<0$, либо $p(c)\cdot p(b)<0$. Пусть, например, выполнено первое неравенство. Тогда на отрезке $[a;c]$ имеется корень, и он нам известен с погрешностью, равной $(b-a)/4$.

Ясно, что деление отрезка пополам может повторяться, и после каждого повторения погрешность уменьшается вдвое. Таким образом, после десяти повторений она уменьшится больше, чем в 1000 раз. Если нам этого достаточно, мы говорим, что процесс сошёлся и принимаем середину последнего отрезка за приближённое значение корня многочлена.

Скорость сходимости этого процесса характеризуется как *линейная* — на каждом шагу погрешность умножается на постоянное число, меньшее 1. В нашем случае это $1/2$. Такая сходимость считается сравнительно медленной. Хорошей считается, например, *квадратичная* скорость сходимости, когда погрешность очередного шага убывает примерно как квадрат погрешности предыдущего.

Каждый итерационный процесс начинается с какого-то начального приближения, в предыдущем примере это — середина отрезка $[a;b]$. Необходимое число итераций зависит не только от самого процесса, но и от начально-

го приближения. Если процесс сходится при любом начальном приближении, то говорится, что имеет место *глобальная сходимость*. Это очень важное достоинство, и ниже мы увидим, с чем это связано.

Метод бисекции, с точки зрения математики, точен и эффективен: в пределе при бесконечном повторении деления пополам отрезок стянется в точку, которая является корнем многочлена. Посмотрим, каким выглядит он, и вообще итерационные процессы, при приближённых вычислениях.

Как было сказано во введении, при реальных вычислениях возможно только конечное число повторений, поэтому итерационный метод вычисления и при точных вычислениях может дать только приближённое решение. Но это никоим образом не является его недостатком по сравнению с другими методами, поскольку точные вычисления невозможны, а при приближённых вычислениях любой метод не даёт точного решения.

В чём сказывается неточность вычислений в итерационных методах? Допустим, что мы оцениваем свой результат, сравнивая очередное приближение x_k с предыдущим x_{k-1}. С ростом числа итераций k растёт число цифр, которые совпадают в двух последовательных приближениях. Сначала совпадающих цифр нет, потом совпадает одна цифра, затем две и так далее.

Если бы мы могли получить приближение, получающееся округлением точного решения, то у этого приближения и следующего за ним совпали бы все цифры. Но этого не происходит. Как вы помните, младшие цифры «загрязнены» ошибками округления. При продолжении итераций они меняются беспорядочно, и стабилизации не происходит. Это значит, что расстояние от полученного приближения до точного решения стало сравнимым с ошибкой округления. Ближе этого к точному решению мы подойти не сможем.

Примерно так же ведёт себя в общем случае и невязка. Сначала она уменьшается, но затем её уменьшение останавливается, и меньше некоторой определённой величины её сделать не удаётся.

В методе бисекции, описанном выше, по мере того, как концы отрезка приближаются к корню, значения многочлена на концах отрезка становятся всё меньше по абсолютной величине и когда-нибудь окажутся сравнимыми с ошибкой округления. В этом случае знак числа становится недостоверным, и дальнейшего уточнения не происходит. Это особенно заметно при вычислении кратных корней. Если корень кратный, то график многочлена касается оси x, и в точке, отстоящей от корня на δ, значения многочлена много меньше — они имеют величину порядка δ^2 для корня кратности 2. Если кратность корня больше, этот эффект ещё усиливается: δ^k для корня кратности k. Задача нахождения кратного корня многочлена плохо обусловлена, и тем хуже, чем больше кратность корня.

Выше, при обсуждении обусловленности, когда мы вычисляли $\sqrt{a}$, мы фактически искали корень уравнения $x^2 - a = 0$, который является кратным при $a = 0$. Мы видели там, что задача плохо обусловлена не только в случае кратного корня, но и тогда, когда отыскивается корень, близкий к другому корню.

Отдельный вопрос, не накапливаются ли ошибки в процессе повторения расчётов. Чтобы в этом разобраться, подумаем, какое влияние оказывает выбор начального приближения на процесс расчётов. Это понятно: если на-

чальное приближение далеко от точного решения, итераций потребуется много, если близко — то меньше. Если при наших поисках корня многочлена мы начали с большого отрезка, то для получения отрезка длины менее 0,001 его придётся делить пополам большее число раз. Но любой отрезок, полученный после какого-то числа делений, мог бы быть принят за начальный. Поэтому точно ли он найден — неважно, важно только то, что на его концах значения многочлена имеют разные знаки.

Вот здесь выясняется роль глобальной сходимости. Если она имеет место, то с какой бы ошибкой ни было вычислено очередное приближение, его можно считать началом нового процесса вычисления, и эта ошибка не влияет ни на сходимость, ни на полученное решение.

Другое дело, если существуют такие начальные значения, для которых процесс не сходится. Тогда может произойти следующее: после большого числа итераций вычисленное очередное приближение окажется искажённым настолько, что начинающийся с него процесс не сходится.

Если это произошло, то для вычислителя, следящего за невязкой, процесс может выглядеть так: сначала он сходится и невязка убывает. После какой-то итерации сходимость замедляется, затем невязка начинает расти, очередное приближение становится совсем не похожим на предыдущее. Вычисления приходится прекращать.

6. Заключение

В короткой заметке невозможно дать полное описание огромной области науки, тем более, что основные её проблемы и методы решения лежат далеко за пределами школьной программы. Здесь подробнее говорилось о методе наименьших квадратов и об итерациях. Это — важные темы, но не были упомянуты многие не менее, и даже более важные. Для того, чтобы говорить о них, потребовалась бы бóльшая подготовка по математике. Пока придётся ограничиться этим.

Моя задача выполнена, надеюсь, мне удалось рассказать о некоторых трудностях, лежащих в начале этой науки, и о некоторых путях их преодоления. Мне хотелось также показать, что вычисления — не просто подстановка чисел в математическую формулу, а интересная, трудная и важная область деятельности.