ЛЕКЦИЯ 2

СИСТЕМЫ ЛИНЕЙНЫХ УРАВНЕНИЙ. МЕТОД ИСКЛЮЧЕНИЯ

В этой лекции в первом приближении будет рассмотрен вопрос о решении систем линейных алгебраических уравнений. Системы линейных уравнений часто встречаются в вычислительной практике. Особенно часто — как составная часть задачи. Так, например, можно сказать (это будет не очень далеко от истины), что в задачах математической физики мы вообще никаких уравнений кроме линейных не решаем: решение сложных нелинейных задач сводится к многократному решению линейных систем. Забегая вперед, я хочу подчеркнуть следующее. О задаче решения линейных систем как о единой задаче имеет смысл говорить лишь до тех пор, пока число уравнений невелико. В таком случае это есть сравнительно простая задача *. Чем больше уравнений в системе, тем более специальную, специфическую структуру она обычно имеет, и тем более необходимо для решения этой системы использовать ее специальные свойства. Некоторые виды таких систем и специфические способы их решения мы рассмотрим позже.

1. Метод исключения

Простейший способ решения линейных систем по существу всем известен из школьного курса. Обычно, когда школьники решают систему линейных уравнений, они исключают какое-либо неизвестное, получают систему меньшего порядка. Потом исключают еще одно неизвестное и т. д. — пока не доберутся до одного уравнения с одним неизвестным. Оформив несколько более культур-

* Стандартные пограммы для решения линейных систем есть на всех вычислительных машинах.

но этот процесс, мы и получим метод исключения. Я поясню его сначала на простом числовом примере. Пусть дана система 4 уравнений:

$$
\begin{aligned}
2x_1 + 3x_2 + 6x_3 + 8x_4 &= 7 \qquad [1] \\
3x_1 + 7x_2 + 3x_3 + 6x_4 &= 3 \qquad [2] \\
2x_1 + 4x_2 + 7x_3 + 7x_4 &= 2 \qquad [3] \\
2x_1 + 5x_2 + 3x_3 + 7x_4 &= 3 \qquad [4]
\end{aligned}
\tag{1}
$$

Из первого уравнения можно выразить x_1 через x_2, x_3, x_4. Если подставить это выражение в остальные уравнения, то получится система 3 уравнений с 3 неизвестными. Практически удобнее провести это исключение, комбинируя уравнения. Например, так. Разделим первое уравнение на коэффициент при x_1. Получим:

$$x_1 + \frac{3}{2}x_2 + 3x_3 + 4x_4 = \frac{7}{2} \qquad [1']$$

Теперь из уравнения [2] вычтем [1′], умноженное на 3, из [3] вычитаем [1′], умноженное на 2 и т. д. В результате новые уравнения [2′—4′] составляют отдельную систему с 3 неизвестными (x_2, x_3, x_4), а вся система приобретает вид:

$$
\begin{aligned}
&x_1 + \frac{3}{2}x_2 + 3x_3 + 4x_4 = \frac{7}{2} && [1'] \\
&\frac{5}{2}x_2 - 6x_3 - 6x_4 = -\frac{15}{2} && [2'] \\
&x_2 + x_3 - x_4 = -5 && [3'] \\
&2x_2 - 3x_3 - x_4 = -4. && [4']
\end{aligned}
$$

Исключая аналогично из уравнений [3′] и [4′] 2 неизвестное (x_2), с помощью уравнения [2′] (уравнение [1′] просто переписывается), мы получим систему:

$$
\begin{aligned}
&x_1 + \frac{3}{2}x_2 + 3x_3 + 4x_4 = \frac{7}{2} \\
&x_2 - \frac{12}{5}x_3 - \frac{12}{5}x_4 = -3 \\
&\frac{17}{5}x_3 + \frac{7}{5}x_4 = -2 \\
&\frac{9}{5}x_3 + \frac{19}{5}x_4 = 2.
\end{aligned}
$$

Наконец, на последнем шаге получается следующая система, эквивалентная исходной:

$$
\begin{aligned}
& x_1 + \frac{3}{2}x_2 + 3x_3 + 4x_4 = \frac{7}{2} \\
& x_2 - \frac{12}{5}x_3 - \frac{12}{5}x_4 = -3 \\
& x_3 + \frac{7}{17}x_4 = -\frac{10}{17} \\
& \frac{52}{17}x_4 = \frac{52}{17}.
\end{aligned}
\tag{2}
$$

Это по-прежнему система 4 уравнений с 4 неизвестными, но она очень легко решается: из последнего уравнения находим x_4, подставляем его в 3 уравнение и находим x_3, затем x_2 из 2 уравнения и, наконец, x_1 из 1-го ($x_4 = 1$; $x_3 = -1$; $x_2 = -3$; $x_1 = 7$). Таким образом, можно сказать, что метод исключения состоит в переходе от заданной системы к эквивалентной ей «треугольной системе», которая легко решается.

2. Трудности в методе исключения

В рассмотренном примере мы использовали простейшую схему метода исключения: x_1 исключается с помощью первого уравнения, x_2 — с помощью второго и т. д. Ясно, что такая схема не всегда приводит к цели. Так, например, если в 1 уравнение x_1 совсем не входит (коэффициент при нем равен 0), то мы не сможем сделать даже первого шага. В этом случае можно, конечно, взять уравнение, содержащее x_1, и ему присвоить номер 1.

Аналогичная неприятность (обращение в 0 нужного нам коэффициента) подстерегает нас на каждом шаге метода исключения.

Подчеркнем, что это может случиться в системах вполне благополучных на вид. Например, в такой, у которой все коэффициенты отличны от нуля:

$$
\begin{aligned}
& x_1 + 2x_2 + 3x_3 + 4x_4 = 2 \\
& x_1 + 3x_2 + x_3 + 2x_4 = -1 \\
& 2x_1 + 3x_2 + 8x_3 + 7x_4 = 10 \\
& 2x_1 + 5x_2 + 3x_3 + 7x_4 = 3.
\end{aligned}
\tag{3}
$$

Здесь обращается в 0 коэффициент при x_3 после 2 шагов исключения.

Таким образом, на каждом шаге метода исключения может оказаться необходимым выбор уравнения, с помощью которого мы будем исключать очередное неизвестное. Поскольку уже использованные (переставленные на первые места) уравнения мы больше не используем, выбор наш на каждом шаге становится все более бедным. Естественно возникает вопрос, не окажется ли, что на каком-то шаге очередное неизвестное (x_k) не входит ни в одно из оставшихся уравнений (с номерами k, $k+1,\dots,n$). Можно доказать, что если заданная система имеет единственное решение, такой случай невозможен. Таким образом, если бы мы занимались алгеброй, то вопрос о решении системы линейных уравнений можно было бы считать исчерпанным: для всякой системы, имеющей (единственное) решение, указан рецепт его нахождения. Реально дело обстоит сложнее, поскольку все вычисления производятся с конечной точностью.

3. Трудности, связанные с приближенностью вычислений. Исключение с выбором главного элемента

При практических вычислениях с конечным числом десятичных знаков мы можем иногда по методу исключения получить результат, далекий от точного решения. Это происходит в тех случаях, когда (при выбранном порядке исключения) некоторые из «ведущих» коэффициентов * очень малы (хотя и не равны нулю). Чтобы избежать потери точности, можно для исключения очередного неизвестного использовать то уравнение, в котором это неизвестное имеет наибольший коэффициент **. Так видоизмененный способ называется методом исключения с выбором главного элемента. Эффект, достигаемый выбором главного элемента, я покажу на «модельном» примере системы 2 уравнений.

Пусть дана система:

$$\begin{aligned} \varepsilon x_1 + x_2 &= 1 \qquad [1] \\ x_1 + 2x_2 &= 4. \qquad [2] \end{aligned} \tag{4}$$

Возьмем $\varepsilon = -10^{-5}$ и предположим, что все вычисления производятся с сохранением 4 значащих цифр (или, как коротко говорят, с 4 знаками). Решим сначала эту систему по простейшей схеме метода исключения, т. е. исключим x_1 с помощью первого уравнения:

$$\begin{aligned} x_1 - 10^5 x_2 &= -10^5 \qquad [1'] \\ (2 + 10^5)x_2 &= 4 + 10^5. \qquad [2'] \end{aligned}$$

Если вычисления производятся с 4 знаками, то мы получим из [2′]: $x_2 = 1$; подставляя в [1′], получим $x_1 = 0$. Теперь поступим по-другому: исключим x_1 с помощью второго уравнения (это и будет для нашего примера исключение с выбором главного элемента). Получим

---

* Ведущий коэффициент — коэффициент при исключаемом неизвестном в уравнении, с помощью которого это неизвестное исключается.

** Имеется в виду, конечно, наибольший по абсолютной величине.

$$x_1 + 2x_2 = 4 \qquad [1'] \qquad (= [2])$$

$$(1+2\cdot10^{-5})x_2 = 1+4\cdot10^{-5} \qquad [2'] \qquad (= [1] + 10^{-5}\cdot[2])$$

Отсюда: $x_2 = 1$; $x_1 = 2$ (вместо $x_1 = 0!$). Легко проверяется, что точные значения x_1 и x_2 отличаются от $x_1 = 2$, $x_2 = 1$ меньше чем на 10^{-4}, так что второй раз мы решили систему достаточно точно. Возвращаясь к общему методу, заметим, что до сих пор мы исключали неизвестные в их первоначальном порядке $(x_1, x_2, \ldots)$ и меняли, при необходимости, лишь порядок уравнений. Возможны, конечно, и другие варианты метода исключения. Например, такой. На очередном шаге выбирается не уравнение, а неизвестное, которое в очередном уравнении стоит с наибольшим коэффициентом. Этот способ называется методом исключения с выбором главного элемента по строке; в отличие от него наш первоначальный выбор называют выбором главного элемента по столбцу *.

4. Всегда ли метод исключения с выбором главного элемента дает разумный результат?

На этот естественный вопрос, по-видимому, нет четкого ответа. Практика показывает, что если система «не очень плохая» (что это значит — надо определять в каждом конкретном случае), то метод исключения с выбором главного элемента «хорошо работает». Однако я хочу обратить ваше внимание на то, что метод исключения с выбором главного элемента можно «обмануть», умножив одно из уравнений на очень большое число. Покажем это на том же модельном примере (4). Мы убедились, что в этой системе (при вычислениях с 4 знаками) нельзя исключать x_1 с помощью первого уравнения, куда это неизвестное «почти не входит». Умножив первое уравнение на $N = 10^6$, мы получим систему, формально (с точки зрения алгебры) эквивалентную исходной:

$$-10x_1 + 10^6 x_2 = 10^6$$

$$x_1 + 2x_2 = 4.$$

Выбирая главный элемент в столбце $\begin{pmatrix} -10 \\ 1 \end{pmatrix}$, мы должны будем как раз исключать x_1 с помощью первого уравнения! Аналогично можно изменить порядок исключения при выборе главного элемента по строке, разделив одно из неизвестных, скажем x_1, на большое число N (т. е. введя новое неизвестное $y_1 = \frac{x_1}{N}$).

* Поскольку выбирается наибольший элемент в строке или столбце матрицы коэффициентов.

Таким образом, умножение на число отдельных уравнений или неизвестных не есть безобидное действие с точки зрения вычислительной математики: вычислительный алгоритм (в частности, программа для машины) может при этом сработать по-другому и дать совсем другой результат. При практическом решении систем линейных уравнений желательна поэтому разумная нормировка уравнений и неизвестных. Обычно при подходящем выборе единиц измерения коэффициенты в уравнениях бывают одного порядка. Если это оказалось не так, то стоит перед решением сделать вот что: отнормировать уравнения так, чтобы в каждом уравнении максимальный коэффициент был порядка 1 *.

5. О трудоемкости метода исключения для общей системы n линейных уравнений

Решение общей системы n линейных уравнений с n неизвестными по методу исключения состоит из 2 этапов.

I. Процесс исключения: сведение заданной системы к эквивалентной ей «треугольной» (уравнения «треугольной» системы есть некие комбинации исходных уравнений).

II. Решение треугольной системы. Забегая вперед, замечу, что основную часть вычислительной работы поглощает I этап — процесс исключения неизвестных. Его трудоемкость еще несколько увеличивается, если нужно производить выбор главного элемента **. Чтобы подсчитать число арифметических действий, опишем еще раз ход вычислений для простейшей схемы метода исключения. Итак, пусть задана система

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = f_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = f_2 \\ \cdots\cdots\cdots \qquad\qquad \cdots\cdots\cdots \quad \cdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = f_n\,. \end{cases} \tag{5}$$

Как обычно, первый индекс у коэффициента — номер уравнения, а второй — номер неизвестного. Опишем процесс исключения неизвестных. При этом для сокращения записи будем писать только коэффициенты перед неизвестными и свободные члены. Исходная система тогда будет иметь вид:

* Стандартные программы обычно не включают в себя нормировку.

** Для вычислительной машины даже перестановка двух уравнений есть «работа»: пересылка числа часто требует времени, сравнимого со временем арифметических операций.

$$\left(\begin{array}{c} a_{11}a_{12}\dots a_{1n} \\ a_{21}a_{22}\dots a_{2n} \\ \dots\dots\dots \\ a_{n1}a_{n2}\dots a_{nn} \end{array}\right) \quad \left|\begin{array}{c} f_1 \\ f_2 \\ \dots \\ f_n \end{array}\right| .$$

Поступаем следующим образом:

1) первое уравнение делим на a_{11}: $[1] \to [1']$;

2) полученное уравнение $1'$ умножаем на a_{21} и вычитаем его из второго уравнения;

3) уравнение $1'$ умножаем на a_{31} и вычитаем его из 3-го уравнения и т. д.

Схематически это выглядит так:

$$\left(\begin{array}{c} a_{11}\, a_{12}\dots a_{1n} \\ a_{21}\, a_{22}\dots a_{2n} \\ \dots\dots\dots \\ a_{n1}\, a_{n2}\dots a_{nn} \end{array}\right) \left|\begin{array}{c} f_1 \\ f_2 \\ \dots \\ f_n \end{array}\right| \longrightarrow \left(\begin{array}{c} 1 \quad b_{12}\dots b_{1n} \\ a_{21}\, a_{22}\dots a_{2n} \\ \dots\dots\dots \\ a_{n1}\, a_{n2}\dots a_{nn} \end{array}\right) \left|\begin{array}{c} g_1 \\ f_2 \\ \dots \\ f_n \end{array}\right| \longrightarrow$$

$$\left(\begin{array}{c} 1 \quad b_{12}\dots b_{1n} \\ 0 \quad a'_{22}\dots a'_{2n} \\ a_{31}\, a_{32}\dots a_{3n} \\ \dots\dots\dots \\ a_{n1}\, a_{n2}\dots a_{nn} \end{array}\right) \left|\begin{array}{c} g_1 \\ f'_2 \\ f_3 \\ \dots \\ f_n \end{array}\right| \longrightarrow \left(\begin{array}{c} 1 \quad b_{12}\dots b_{1n} \\ 0 \quad a'_{22}\dots a'_{2n} \\ 0 \quad a'_{32}\dots a'_{3n} \\ \dots\dots\dots \\ a_{n1}\, a_{n2}\dots a_{nn} \end{array}\right) \left|\begin{array}{c} g_1 \\ f'_2 \\ f'_3 \\ \dots \\ f_n \end{array}\right| \longrightarrow$$

$$\longrightarrow \qquad \longrightarrow \left(\begin{array}{c} 1 \quad b_{12}\, b_{13}\dots b_{1n} \\ 0 \quad a'_{22}\, a'_{23}\dots a'_{2n} \\ 0 \quad a'_{32}\, a'_{33}\dots a'_{3n} \\ \dots\dots\dots \\ 0 \quad a'_{n2}\, a'_{n3}\dots a'_{nn} \end{array}\right) \left|\begin{array}{c} g_1 \\ f'_2 \\ f'_3 \\ \dots \\ f'_n \end{array}\right| .$$

Следующий шаг аналогичен первому. Первое уравнение теперь трогать не будем, а со вторым и последующим поступаем как с заданной системой на первом шаге:

1) второе уравнение делим на a'_{22} ($[2'] \to [2'']$);

2) полученное уравнение $2''$ умножаем на a'_{31} и вычитаем его из третьего уравнения и т. д. (всего $n-2$ комбинации).

В результате получим:

$$\left(\begin{array}{cccc} 1 & b_{12} & b_{13} \dots & b_{1n} \\ 0 & 1 & b_{23} \dots & b_{2n} \\ 0 & 0 & a''_{33} \dots & a''_{3n} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & a''_{n3} \dots & a''_{nn} \end{array}\right) \quad \left|\begin{array}{c} g_1 \\ g_2 \\ f''_3 \\ \dots \\ f''_n \end{array}\right| .$$

Продолжая эту процедуру, через n шагов мы получим:

$$\begin{pmatrix} 1 & b_{12} & b_{13} & \dots & b_{1n} \\ 0 & 1 & b_{23} & \dots & b_{2n} \\ 0 & 0 & 1 & \dots & b_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \quad \begin{vmatrix} g_1 \\ g_2 \\ g_3 \\ \dots \\ g_n \end{vmatrix} . \tag{6}$$

Эта треугольная система эквивалентна исходной системе (5). Раньше, чем подсчитывать число произведенных действий, заметим, что особенно точный подсчет не имеет смысла. Дело в том, что вычислительная машина значительную часть времени тратит на логические операции, число и длительность которых зависит от конструкции машины и от того, как написана программа. Поэтому подсчитаем лишь число умножений и делений (выполняемых медленнее, чем сложение и вычитание). На первом шаге исключения мы произвели n делений при нормировке первого уравнения, затем по n умножений для каждого из следующих. Итого: $n+(n-1)n=n^2$ умножений и делений. На втором шаге мы имеем дело с системой $(n-1)$-го порядка — $(n-1)^2$ умножений и делений. Всего

$$N_{\text{искл.}} = n^2+(n-1)^2+\dots+1 =$$
$$=\frac{1}{6}n(n+1)(2n+1) \approx \frac{1}{3}n^3+\frac{1}{2}n^2. \tag{7}$$

Теперь рассмотрим решение треугольной системы (6). Записав ее полностью:

$$\begin{aligned} x_1 + b_{12}x_2 + \dots + b_{1n}x_n &= g_1 \\ x_2 + \dots + b_{2n}x_n &= g_2 \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ x_n &= g_n, \end{aligned} \tag{8}$$

мы видим, что нахождение x_n совсем не требует вычислений. Далее: $x_{n-1}=g_{n-1}-b_{n-1,n}x_n$ — одно умножение. Для x_{n-2} нужно два умножения и т. д. Итого:

$$1+2+\dots+(n-1)=\frac{n(n-1)}{2}\approx\frac{1}{2}n^2 \quad \text{умножений.}$$

Всего решение системы n уравнений по простейшей схеме метода исключения требует следующего количества умножений и делений:

$$N \approx \frac{1}{3}n^3+n^2. \tag{9}$$

С ростом n эта величина быстро растет.

6. Пример: трехдиагональная система

При решении краевых задач для дифференциальных уравнений часто приходится дело иметь с линейными системами, бóльшая часть коэффициентов в которых равна нулю. Простейшая (и очень часто встречающаяся) система такого рода имеет следующий вид:

$$\begin{array}{lr} b_0x_0 + c_0x_1 & = f_0 \\ a_1x_0 + b_1x_1 + c_1x_2 & = f_1 \\ \quad a_2x_1 + b_2x_2 + c_2x_3 & = f_2 \\ \dots\dots\dots\dots\dots\dots\dots\dots & \\ \qquad a_nx_{n-1} + b_nx_n & = f_n. \end{array} \tag{10}$$

В матрице коэффициентов этой системы отличны от 0 лишь коэффициенты, стоящие на главной диагонали и 2 соседних с ней:

$$\begin{pmatrix} b_0 & c_0 & 0 & 0 & \dots\dots & 0 & 0 \\ a_1 & b_1 & c_1 & 0 & \dots\dots & 0 & 0 \\ 0 & a_2 & b_2 & c_2 & \dots\dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots a_{n-1} & b_{n-1} & c_{n-1} \\ 0 & 0 & 0 & 0 & \dots 0 & a_n & b_n \end{pmatrix}. \tag{11}$$

Ясно, что такую систему глупо решать по стандартной программе: машина будет большую часть времени оперировать с нулями. Но сам метод исключения здесь вполне пригоден.

Особенно простые и красивые формулы получаются, если можно использовать простейшую схему метода исключения (*без* выбора главного элемента). Чтобы их вывести, заметим сначала, что на каждом шаге исключения нам придется исключать определенное неизвестное x_k лишь из одного уравнения. По окончании процесса исключения мы получим треугольную систему очень специального вида:

$$\begin{pmatrix} 1 & p_0 & 0 & \dots & 0 \\ 0 & 1 & p_1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix} \qquad \begin{vmatrix} r_0 \\ r_1 \\ \dots \\ r_n. \end{vmatrix}. \tag{12}$$

Из этой системы легко последовательно (рекуррентно) найти все x_i:

$$x_n = r_n; \quad x_k = r_k - p_kx_{k+1} \qquad (k = n-1, \dots, 0)\,. \tag{13}$$

Теперь проследим, как возникают коэффициенты p_k и r_k в процессе исключения. На k-ом шаге процесса исключения картина такова. Первые $k-1$ уравнений уже в окончательном виде, k-ое уравнение изменилось при исключении x_{k-1}. Все остальные — не тронуты:

$$\begin{pmatrix} 1 & p_0 & 0 & \dots\dots\dots & 0 & 0 & \dots \\ \dots & \dots & \dots & \dots & & & \\ 0 & 0 \dots & 1 & p_{k-1} & 0 & 0 & \dots \\ 0 & 0 \dots & 0 & b'_k & c_k & 0 & \dots \\ 0 & 0 \dots & 0 & a_{k+1} & b_{k+1} & c_{k+1} & \dots \\ \dots & \dots & \dots & \dots & & & \end{pmatrix} \quad \begin{vmatrix} r_0 \\ \dots \\ r_{k-1} \\ f'_k \\ f_{k+1} \\ \dots \end{vmatrix}. \tag{14}$$

Дальше можно писать только две строки. Отнормировав уравнение $[k']$, получим

$$\begin{pmatrix} 0 \dots\dots & 1 & p_k & 0 \dots\dots & 0 \\ 0 \dots\dots & a_{k+1} & b_{k+1} & c_{k+1} \dots & 0 \end{pmatrix} \quad \begin{vmatrix} r_k \\ f_{k+1} \end{vmatrix}.$$

Исключив x_k из уравнения $[k+1]$, получим:

$$\begin{pmatrix} 0 \dots 1 & p_k & 0 & \dots\dots\dots & 0 \\ 0 \dots 0 & b'_{k+1} & c_{k+1} & \dots\dots & 0 \end{pmatrix} \quad \begin{vmatrix} r_k \\ f'_{k+1} \end{vmatrix}.$$

Здесь $b'_{k+1}=b_{k+1}-a_{k+1}\,p_k$, $f'_{k+1}=f_{k+1}-a_{k+1}r_k$. Поделив на b'_{k+1}, получим

$$p_{k+1}=\frac{c_{k+1}}{b_{k+1}-a_{k+1}\cdot p_k}; \qquad r_{k+1}=\frac{f_{k+1}-a_{k+1}\cdot r_k}{b_{k+1}-a_{k+1}\cdot p_k}. \tag{15}$$

Итак, решение системы (10) простейшим методом исключения сводится к следующему:

I этап

Находятся рекуррентно в порядке $k=0,1,\dots,n-1$ p_{k+1} и r_{k+1} по формулам (15).

При этом $p_0=\dfrac{c_0}{b_0}$, $\quad r_0=\dfrac{f_0}{b_0}$.

II этап

Находятся рекуррентно в порядке $k=n-1,\dots,1,0$ неизвестные x_k по формуле (13)

$$x_k=r_k-p_k x_{k+1}. \tag{16}$$

При этом $x_n=r_n$. Формулы (15) и (16) очень часто употребляются и носят выразительное название «прогонка» (I этап — «прямая прогонка», II этап — «обратная прогонка»). Различная «направленность» (по индексам) этих двух этапов не есть, конечно, особое свойство этих формул — это общее свойство метода исключения. А вот простота вычислений специфична для системы (10). Заметьте, что с ростом n количество действий здесь растет пропорционально n (а не $\sim\frac{1}{3}n^3$, как в общем случае). Поэтому без труда решаются трехдиагональные системы даже при $n\sim 100-200$. В заключение еще раз напомним, что прогонка не есть какой-то новый способ решения. Это есть лишь удобная формульная запись самого простого варианта метода исключения.

7. Метод исключения и определители

Идея исключения неизвестных очень естественна и элементарна. Поэтому особенно интересно взглянуть на метод исключения с более «ученой» точки зрения *. Вы знаете, что в алгебре существуют универсальные формулы для решения любой невырожденной системы линейных уравнений. По этим формулам (называемым формулами Крамера) каждое неизвестное представляется в виде отношения двух определителей n-го порядка:

$$x_k = \Delta_k/\Delta. \tag{17}$$

В свою очередь определитель n-го порядка может быть задан явной формулой. Точный вид ее для нас сейчас не важен. Важно лишь следующее: в этой формуле $n!$ $(=1\cdot 2\cdot\ldots\cdot n)$ слагаемых, а каждое слагаемое состоит из n множителей. Что произойдет, если решение линейной системы находить прямо по формулам Крамера и определители при этом считать «в лоб» по задающим их формулам?

Всего нужно вычислить $n+1$ определитель. Вычисление каждого из них требует $n\cdot n!$ умножений. Итого $(n+1)n\cdot n!\approx\approx(n+2)!$ умножений. Прикинем, что это означает, скажем, для $n=20$. Воспользовавшись приближенной формулой $k!\approx\approx\sqrt{2\pi k}\,(k/e)^k$, получим: $22!>(22/e)^{22}>8^{22}>10^{19}$. Предположим, что мы располагаем машиной, делающей 10^5 арифметических действий в секунду. Тогда нам понадобится более 10^{14} секунд или более 3 миллионов лет непрерывной работы **. Реально, по стандартной программе система 20 порядка решается за доли секунды! Каким образом получается, что прекрасные общие формулы непригодны для практического использования? На этот вопрос я не могу дать ясного ответа. Замечу только следующее. По первоначальному смыслу неизвестные (и уравнения)

---

* Всерьез алгебраическую структуру метода исключения мы обсудим позже, во 2 концентре. Здесь же затрагивается лишь арифметическая сторона дела (число необходимых арифметических операций).

** 1 год $\approx 3\cdot 10^7$ сек.

в системе линейных уравнений равноправны. Формулы Крамера сохраняют это равноправие (при перестановке неизвестных соответствующие определители могут лишь поменять знак). При решении системы методом исключения мы устанавливаем некоторый вполне определенный порядок неизвестных. Нарушив имевшуюся симметрию, мы чрезвычайно выигрываем в скорости!

Решив систему по методу исключения, мы тем самым (косвенно) вычислим отношение определителей (17). Легко сообразить, что попутно можно найти и каждый из них в отдельности. Самый интересный — определитель системы Δ *. Вспомним, что определитель не меняется, если к одной из его строк добавляется другая, умноженная на любое число. Значит, в процессе исключения мы меняем определитель системы лишь при нормировке уравнений: при делении строки на очередной ведущий коэффициент определитель делится на то же число **. В полученной треугольной системе определитель равен произведению диагональных элементов, т. е. 1. Обозначим ведущий элемент k-го шага c_k (в простейшей схеме исключения $c_1 = a_{11}$, $c_2 = a'_{22}$ и т. д.). Тогда $\frac{\Delta}{c_1 \cdot c_2 \cdot \ldots \cdot c_n} = 1$ или $\Delta = c_1 \cdot c_2 \cdot \ldots \cdot c_n$. Значит, мы можем попутно в процессе исключения найти определитель Δ — для этого нужно $\approx \frac{1}{3} n^3 + \frac{1}{2} n^2$ умножений и делений. Несмотря на простоту, вывод этот все-таки удивителен. Подумайте сами. Определитель, скажем, 10-го порядка

$$\Delta = \begin{vmatrix} a_{1,1} & a_{1,2} & \ldots & a_{1,10} \\ \cdots & \cdots & \cdots & \cdots \\ a_{10,1} & a_{10,2} & \ldots & a_{10,10} \end{vmatrix}$$

есть функция 100 независимых переменных $a_{i,k}$. Формула, задающая эту функцию в явном виде, содержит более 3 миллионов слагаемых, в каждом из которых перемножаются 10 из 100 переменных. Тем не менее значение этой функции при заданных значениях аргументов можно вычислить менее, чем за 1000 арифметических действий. Попробуйте, глядя на разложение определителя, до этого додуматься!

8. Решение системы линейных уравнений и обращение матриц

Теперь мы немного заглянем на более высокий уровень и сопоставим задачу решения системы линейных уравнений

$$Ax = f \tag{18}$$

---

* Т.е. определитель, составленный из коэффициентов системы.

** Если уравнения переставляются, то нужно будет еще учесть изменение знака Δ при перестановке.

с задачей обращения матрицы A. Заметим, прежде всего, что при известной обратной матрице A^{-1} решение (18) есть: $\boldsymbol{x}=A^{-1}\boldsymbol{f}$. Умножение матрицы на вектор требует немного вычислений. Значит, зная A^{-1}, мы можем решить систему (18) с любой правой частью $\boldsymbol{f}$ сравнительно легко.

Заметим дальше, что столбцы обратной матрицы сами являются решениями системы (18) при специальном выборе правых частей. Скажем, первый столбец отвечает вектору $\boldsymbol{f}=$ $=(1,0,\ldots,0)$ — это есть прямое следствие равенства $AA^{-1}=I$ (I — единичная матрица).

Таким образом, знание A^{-1} эквивалентно знанию решений системы (18) при n различных правых частях. Поэтому на первый взгляд кажется, что обращение матрицы A должно быть в n раз более трудоемким, чем решение системы (18). Небольшое размышление показывает, однако, что это не так. Дело в том, что основная работа в процессе исключения производится над коэффициентами системы и не зависит от правых частей. В результате оказывается, что обращение матрицы лишь в фиксированное число раз — приблизительно втрое — более трудоемко, чем решение отдельной системы. Поэтому, если нужно решить одну систему **несколько** раз (>3—4) при разных правых частях, может оказаться разумным использовать обратную матрицу.

Алгоритм обращения матрицы легко извлечь из того же метода исключения, и мы позже это проделаем. А пока заметим, что число умножений $N\sim n^3$ для обращения матрицы с некоторой точки зрения является совершенно удивительным. В самом деле. Рассмотрим произведение двух матриц n-го порядка $C=AB$. **Для нахождения** элементов C согласно определению нужно произвести n^3 умножений — по n на каждый из элементов c_{ik}:

$$c_{ik}=\sum_{\alpha=1}^{n} a_{i\alpha}b_{\alpha k}.$$

Интуитивно обращение матрицы кажется гораздо более трудной задачей, чем перемножение двух матриц. Между тем оно требует не большого количества действий! Стоит все же реабилитировать нашу интуицию. Обращение матриц, конечно, *есть* более сложная задача, но в *алгоритмическом* смысле. Алгоритмы здесь нетривиальны, для данной матрицы не все алгоритмы пригодны и т. д. Таким образом, наша интуитивная настороженность при обращении матриц — правильна. Но не следует все же забывать и вторую сторону дела: обращение матрицы невысокого порядка — не столь уж трудоемкая процедура.

Примечания ко 2 лекции

1) (к п. 1). Убедимся, что после k шагов метода исключения очередное неизвестное x_{k+1} обязательно входит в одно из оставшихся $n-k$ уравнений. Для этого нам понадобится следующее утверждение из теории систем линейных уравнений. Если система при заданных правых частях имеет единственное решение, то она обязательно имеет решение при любых правых частях.

Предположим теперь, что x_{k+1} не входит в (преобразованные) уравнения $[k+1],\ldots,[n]$. Неизвестные $x_1, x_2,\ldots,x_k$ из этих уравнений уже исключены. Стало быть, мы имеем систему $n-k$ уравнений с $n-k-1$ неизвестным. Ясно, что такая (переопределенная) система не может иметь решения при *любых* правых частях (легко дать этому и формальное доказательство). Полученное противоречие и доказывает утверждение.

2) (к п. 3). Приведенный в тексте модельный пример хорош лишь своей простотой. Однако ситуация, когда потеря точности происходит за счет одного очень малого коэффициента — исключительна. Гораздо чаще можно получить неверный ответ за счет наличия многих умеренно малых ведущих коэффициентов. Примеры такого рода сложнее и мы построим их позже.

3) (к п. 5). Выбор главного элемента по столбцу (т. е. бо́льшего из n чисел) требует на первом шаге $\sim n$ арифметических и логических действий. Всего необходимое количество действий имеет порядок n^2 и уже при $n \sim 5-6$ замедление, связанное с выбором главного элемента, становится несущественным *.

4) (к п. 8). Некоторой сенсацией была надавняя работа Strassen'a **, где был предложен алгоритм перемножения матриц, требующий $\sim n^\alpha$, $\alpha<3$ действий. На сегодня открыт вопрос: каково минимальное α? Нельзя ли его сделать близким к 2? Независимо от практической ценности подобных алгоритмов их изучение очень интересно для теории.

5) Некоторые литературные указания.

Геометрические аспекты решения систем линейных уравнений (в смысле геометрии n-мерного пространства), в частности, вопрос о правильной нормировке, рассмотрены в книге:

К. Ланцош. Практические методы прикладного анализа. Физматгиз, 1961.

Много различных методов для решения систем линейных урав-

---

* Иногда на каждом шаге используют выбор наибольшего элемента во всей матрице. Такой выбор требует $\sim \frac{1}{3} n^3$ арифметических действий (скажем, вычитаний). Это уже по порядку сравнимо с основными действиями процесса исключения.

** V. Strassen, Gaussian elimination is not optimal. Numer. Math. 13, 354—356, 1969. Переведено в сб. «Математика», № 14: 3, 1970.

нений, обращения матриц и нахождения собственных значений описаны в книгах:

В. В. Воеводин. Численные методы алгебры. М., «Наука», 1966.

Д. К. Фаддеев, В. Н. Фаддеева. Вычислительные методы линейной алгебры. М., Физматгиз, 1963.

Некоторые идеи, обсуждающиеся в этих книгах, мы рассмотрим позже.