Абрамов С. А., Рябенко А. А.

О строго невырожденных числовых матрицах

Аннотация

Как для заданной невырожденной числовой вещественной матрицы, в элементах которой после десятичной точки присутствует лишь конечное число цифр, проверить, останется ли эта матрица невырожденной после произвольного добавления цифр в младшие разряды некоторых (явно указанных) ее элементов? Выясняется, что эта задача алгоритмически разрешима. Обсуждается компьютерная реализация предлагаемого алгоритмического решения.

1. Добавление цифр к элементам числовой матрицы

Пусть известно конечное количество цифр после десятичной точки в каждом элементе вещественной числовой $n \times n$-матрицы A. Количество известных цифр может не совпадать для разных элементов. Не исключается, что для некоторых из элементов матрицы такое конечное представление оказывается точным, — значащие цифры после десятичной точки известны полностью.

Но относительно некоторых элементов изначально нет информации, все ли цифры после десятичной точки выписаны. Может быть так, например, что полная десятичная запись какого-то из элементов требует бесконечного количества цифр. Или, может быть, в заданной записи элемента не хватает лишь конечного количества цифр. Мы считаем, что такого рода элементы как-то отмечены в матрице (далее говорим об *отмеченных элементах*).

Пусть вычисление определителя матрицы A, выполненное исходя из заданного конечного представления элементов, дало ненулевое значение. Вопрос — верно ли, что матрица остается невырожденной при любом добавлении цифр в младшие разряды отмеченных элементов матрицы A?

Рассмотрим два примера, в каждом из них фигурирует некоторая невырожденная 2×2-матрица $A = (a_{ij})$.

Пусть

$$A = \begin{pmatrix} 1.0 & -3.333 \\ -0.3 & 0.99 \end{pmatrix}, \tag{1}$$

для выделенного элемента $a_{1,1} = 1.0$ известна только одна цифра после точки и нет гарантии, что нам известны все цифры. Можно убедиться, что если здесь добавить в конец $a_{1,1}$ цифру 1, получив $a_{1,1} = 1.01$, то матрица станет вырожденной. Если же считать известными в $a_{1,1}$ две нулевые цифры после точки: $a_{1,1} = 1.00$, то матрица станет вырожденной при добавлении бесконечного количества девяток, дающем $a_{1,1} = 1.00999... = 1.01$.

Пример матрицы, которая остается невырожденной при любом добавлении цифр к каждому элементу:

$$\begin{pmatrix} 1.40 & -7.8 \\ 9.954 & 0.46 \end{pmatrix}. \tag{2}$$

К этим двум примерам мы еще вернемся в разделе 3.

Пусть элемент a_{ij} исходной матрицы имеет после десятичной точки m цифр (несколько последних из этих цифр могут быть нулями). Добавление цифр к этому элементу может быть осуществлено так: берется какая-то десятичная дробь β, которую мы представим условно в виде

$$0.\beta_1\beta_2\dots,$$

и к a_{ij} прибавляется $10^{-m}\beta$ (со знаком "минус", если $a_{ij} < 0$). В результате цифры $\beta_1, \beta_2, \dots$ дроби β будут добавлены в конец числа a_{ij}. Допускается, что дробь β, начиная с какого-то места, имеет только нулевые цифры. Фактически, значение β — произвольное вещественное число из отрезка $[0, 1]$, и значению $\beta = 1$ соответствует дробь $0.999\dots$. Последнее использовано нами выше при рассмотрении матрицы (1) после замены 1.0 на 1.00.

2. Распознавание строгой невырожденности

Определение. Пусть в каждом из элементов невырожденной числовой вещественной матрицы A присутствует лишь конечное число цифр после десятичной точки. Назовем эту матрицу *строго невырожденной* относительно указанного набора U ее отмеченных элементов, если эта матрица остается невырожденной после произвольного добавления цифр в младшие разряды элементов, входящих в U. Если U содержит все элементы матрицы A, то матрицу A будем называть строго невырожденной без ссылки на какой-либо набор ее отмеченных элементов.

Задача распознавания (т.е. задача, где предполагаемый ответ — это *да* или *нет*) строгой невырожденности данной матрицы оказывается алгоритмически разрешимой.

Теорема. *Существует алгоритм, распознающий строго невырожденные матрицы (строгая невырожденность может рассматриваться и относительно набора элементов, и обобщенно).*

Касаясь алгоритма, укажем здесь, что элементарный подход работает, когда в матрице отмечен лишь один элемент, а при большем числе таких элементов можно применить метод цилиндрической декомпозиции при элиминации кванторов [3, 1, 2, 4], являющийся вариантом теоремы Тарского [5] (в [8, разд. 3.8] эта теорема именуется теоремой Тарского-Зайденберга).

Подробно разбираемый в следующем разделе пример демонстрирует схему алгоритма.

3. Реализация в Maple

Для упомянутого в разделе 2 алгоритма решения обсуждаемой задачи нами подготовлена реализация в среде компьютерной алгебры Maple 2024 [6]. Используется реализация метода цилиндрической декомпозиции, предоставляемая Maple-пакетом *QuantifierElimination* (см. [7]).

Пусть для матрицы (1) требуется определить, является ли ее невырожденность строгой относительно элементов $a_{1,2}, a_{2,1}, a_{2,2}$. Чтобы избежать округления чисел при вычислении определителя, в Maple элементы матрицы записываем рациональными числами. Отмечаем элементы так: если $a_{i,j}$ — положительное число, то прибавляем к нему $T_{i,j} \cdot 10^{-m}$, где T — неопределенное имя, m — количество известных цифр после десятичной точки элемента $a_{i,j}$. Если же $a_{i,j} < 0$, вычитаем $T_{i,j} \cdot 10^{-m}$. Таким образом, в сессии Maple выполняем присваивание

$$> A := \begin{bmatrix} 1 & -3333 \cdot 10^{-3} - T_{1,2} \cdot 10^{-3} \\ -3 \cdot 10^{-1} - T_{2,1} \cdot 10^{-1} & 99 \cdot 10^{-2} + T_{2,2} \cdot 10^{-2} \end{bmatrix}:$$

Следующей командой вычисляем определитель. Этому полиному от всех $T_{i,j}$ с рациональными коэффициентами дается имя Dt:

$> Dt := LinearAlgebra\text{:-}Determinant(A);$

$$Dt := -\frac{99}{10000} + \frac{1}{100}T_{2,2} - \frac{3333}{100000}T_{2,1} - \frac{3}{10000}T_{1,2} - \frac{1}{100000}T_{1,2}T_{2,1}$$

Далее получаем список S параметров, от которых зависит определитель Dt:

$> S := [indets(Dt)[\,]]:$

$$S := [T_{1,2}, T_{2,1}, T_{2,2}]$$

Составляем соответствующее выражение с квантором: существуют такие значения параметров $T_{1,2}, T_{2,1}, T_{2,2}$, что выполняется $Dt = 0$ и значения параметров находятся в отрезке $[0, 1]$. В Maple это выполняется следующим образом:

$> exists(S, \mathrm{And}(Dt = 0,\ seq([i \geq 0,\ i \leq 1][\,], i = S)));$

$$\exists\Big([T_{1,2}, T_{2,1}, T_{2,2}], -\frac{99}{10000} + \frac{1}{100}T_{2,2} - \frac{3333}{100000}T_{2,1}$$

$$-\frac{3}{10000}T_{1,2} - \frac{1}{100000}T_{1,2}T_{2,1} = 0 \wedge$$

$$0 \leq T_{1,2} \wedge T_{1,2} \leq 1 \wedge 0 \leq T_{2,1} \wedge T_{2,1} \leq 1 \wedge 0 \leq T_{2,2} \wedge T_{2,2} \leq 1\Big)$$

Процедура *QuantifierElimination:-QuantifierEliminate* определяет, выполняется ли записанное выше выражение:

$> QuantifierElimination\text{:-}QuantifierEliminate(\%);$

$$true$$

Ответ *true* означает, что возможно добавление цифр к отмеченным элементам, обращающее определитель в 0, т.е. невырожденность (1) относительно $a_{1,2}, a_{2,1}, a_{2,2}$ не является строгой.

Эта последовательность команд нами оформлена в виде процедуры *StronglyNonsingular*. Аргумент процедуры — квадратная числовая матрица с отмеченными указанным выше способом элементами (т.е. каждый (i, j)-й элемент матрицы — полином от параметра $T_{i,j}$, степени не выше 1, с коэффициентами-рациональными числами). При этом, в отличие от рассмотренной выше последовательности команд, процедура возвращает true, если матрица является строго невырожденной, и false иначе.

Запишем в Maple матрицу (2), отметив в ней все элементы:

$$> A := \begin{bmatrix} 14 \cdot 10^{-2} + T_{1,1} \cdot 10^{-2} & -78 \cdot 10^{-1} - T_{1,2} \cdot 10^{-1} \\ 9954 \cdot 10^{-2} + T_{2,1} \cdot 10^{-2} & 99 \cdot 46 \cdot 10^{-2} + T_{2,2} \cdot 10^{-2} \end{bmatrix}:$$

Получаем, что эта матрица строго невырождена:

$> StronglyNonsingular(A);$

$$true$$

Мы провели эксперименты с 5×5-матрицей со случайно выбранными элементами, для которых известны значения двух цифр после десятичной точки. В этой матрице были отмечены два элемента, затем три и четыре элемента. Относительно этих элементов матрица оказалась строго невырожденной. Время работы[1] процедуры *QuantifierElimination:-QuantifierEliminate* — 0.182 для двух отмеченных элементов, 7.091 для трех и 640.624 для четырех.

Были проведены эксперименты и с нестрого невырожденной 5×5-матрицей с элементами, для которых известны значения двух цифр после десятичной точки. Так же были отмечены два элемента, затем три и четыре элемента, относительно которых матрица нестрого невырождена. Время работы процедуры *QuantifierElimination:-QuantifierEliminate* — 0.051 для двух отмеченных элементов, 0.362 для трех и 0.257 для четырех.

[1]В секундах. Вычисления проводились в Maple 2024, Ubuntu 8.04.4 LTS, AMD Athlon(tm) 64 Processor 3700+, 3GB RAM

Сессия Maple с примерами и процедурой *StronglyNonsingular* доступна по ссылке

http://www.ccas.ru/ca/_media/test_of_strongly_nonsingular.mw

и pdf-вариант этой сессии:

http://www.ccas.ru/ca/_media/test_of_strongly_nonsingular.pdf

Авторы признательны М.Кауэрсу, Г.Погудину и Д.Хмельнову за советы по теме публикации. Авторы также благодарят компанию Maplesoft (Ватерлоо, Канада) за консультации и дискуссии.

Литература

1. Basu S., Pollack R., Coste-Roy M.-F. Algorithms in real algebraic geometry // Algorithms and Computation in Mathematics, vol. 10, Springer, 2006.
2. Caviness B.F., Johnson J.R. (eds.) Quantifier elimination and cylindrical algebraic decomposition // Texts & Monographs in Symbolic Computation, Springer, 1998.
3. Collins G.E. Quantifier elimination for real closed fields by cylindrical algebraic decomposition // In Proc. 2nd GI Conf. Automata Theory and Formal Languages. New York: Springer-Verlag, pp. 134–183, 1975.
4. Davenport J. and Heintz J. Real quantifier elimination is doubly exponential // J. Symb. Comput. 5, pp. 29–35, 1988.
5. Tarski A. A decision method for elementary algebra and geometry // Santa Monica CA: RAND Corp., 1948.
6. Maple online help: http://www.maplesoft.com/support/help/
7. Tonks Z. A Poly-algorithmic quantifier elimination package in Maple // In Jürgen Gerhard and Ilias Kotsireas, editors, Maple in Mathematics Education and Research, pp. 176–186, 2020. Springer International Publishing.
8. Верещагин Н.К., Шень А. Языки и исчисления. М: МЦНМО, 2000.

Сведения об авторах

Абрамов Сергей Александрович, д.ф-м.н., гл.н.с.
Федерального исследовательского центра “Информатика и управление” РАН,
проф. факультета вычислительной математики и кибернетики
МГУ им. М.В. Ломоносова
sergeyabramov@mail.ru
Рябенко Анна Андреевна, к.ф-м.н., ст.н.с.
Федерального исследовательского центра “Информатика и управление” РАН
anna.ryabenko@gmail.com