

## КОМПЬЮТЕРНОЕ РЕШЕНИЕ ФУНДАМЕНТАЛЬНОЙ ПРОБЛЕМЫ ИСКЛЮЧЕНИЯ

**Ключевые слова:** *ведущие элементы, оптимизация пивотирования, элементарные матричные преобразования, ортогонализация.*

В настоящей работе рассматривается не общая «задача исключения», как она описана в [1], а проблема корректного решения систем линейных алгебраических уравнений посредством последовательного исключения неизвестных, хотя выводы применимы и к общей задаче исключения. Впервые такой метод был предложен Гауссом. Как известно, Гаусс проводил обработку геофизических наблюдений, в которой приходилось решать системы линейных алгебраических уравнений со многими десятками неизвестных, а при ручных вычислениях матрица таких систем просто физически необозрима, поэтому для обеспечения устойчивости вычислений Гаусс применял только сокращенную стратегию выбора ведущих элементов — в очередном по порядку следования столбце, называя «главным элементом» максимальный по абсолютной величине элемент столбца. С появлением компьютеров принцип выбора «главных элементов» «механически» распространился на всю матрицу, однако не дал существенных преимуществ перед столбцовой стратегией выбора ведущих элементов. Общее разочарование тогда выразил Уилкинсон: «Мы не утверждаем, что выбор наибольшего элемента в качестве ведущего является наилучшим, и в действительности это часто будет неверно. Мы лишь утверждаем, что до сих пор не предложен никакой другой практически приемлемый процесс... С другой стороны, выбор главного элемента по всей матрице никогда не является очень плохим.» [2]. В действительности же все тестовые примеры для программ, выбиравших главные элементы по всей матрице, в сущности не выходили за рамки ручных вычислений, предполагающих, как правило, отсутствие существенной разницы в порядках исходных данных, и тем самым игнорировали особенности компьютерных вычислений и поэтому не могли выявить слабые стороны метода главных элементов.

В чисто математическом плане исключение и, следовательно, треугольное разложение матрицы системы следуют сразу же из условия невырожденности этой матрицы. Однако в вычислительном плане на стандартных компьютерах с их ограниченной разрядностью в ограниченном, но все же значительном диапазоне порядков даже при невырожденности матрицы системы приходится требовать компьютерной невырожденности.

**Определение.** Система линейных алгебраических уравнений компьютерно невырожденна (в отношении методов исключения), если существует хотя бы один процесс исключения, в котором получающаяся треугольная матрица оказывается невырожденной.

В компьютерной арифметике для полной (вырождающей) потери цифровой информации существует только две возможности: полное взаимоуничтожение (в результате вычитания близких чисел) и полное «затирание» информации (вытеснение исходной информации относительно большим слагаемым). В вычислительной теории обе эти возможности принято «скрывать» в одном символическом компьютерном уравнении  $I + \varepsilon = I$  с так называемыми «исчезающими  $\varepsilon$ » — сверхмалыми относительно других компьютерных чисел. Первая возможность «вскрывается»,

если переписать это уравнение в виде

$$1 - (1 + \varepsilon) = 0, \quad (1)$$

а вторая выявляется, если при переписывании использовать типичный для программирования операторный подход, при котором адрес первого операнда обычно является и адресом результата, и тогда приобретает значимость следующий его вид:

$$\varepsilon + 1 = 1. \quad (2)$$

(Само уравнение  $1 + \varepsilon = 1$  при операторном подходе означает просто безопасное сохранение исходной информации, тогда как символические уравнения по своему назначению должны обращать внимание программистов на ключевые опасные ситуации в компьютерных вычислениях. «Соккрытие» их влияет на «азы» компьютерной арифметики, особенно (2).) Ситуация (1) моделирует полное обнуление из-за чрезмерной относительной «близости» компьютерных чисел, чем и выражается вырождение ведущего элемента. Ситуация (2) — подавления и утраты информации — возникает из-за чрезмерной относительной удаленности компьютерных чисел и создает предпосылки для катастрофического обнуления на последующем шаге процесса исключений.

Именно такое катастрофическое уничтожение компьютерной информации допускает гауссов принцип главных элементов, что наглядно демонстрирует приводимый ниже контрпример. Он только перестановкой строк отличается от примера в [3], построенного авторами на идее Хемминга, для решения которого по методу Гаусса разработаны «искусные» программы, приведенные в той же книге. Однако рассмотренный ниже контрпример неразрешим даже этими усовершенствованными программами.

**Контрпример.**

$$\begin{cases} 2x_1 + x_2 + x_3 = 2e4 \\ x_1 + 2e - 4x_2 + 3e - 4x_3 = 6 \\ x_1 - e - 4x_2 + e - 4x_3 = 1. \end{cases} \quad (3)$$

При проведении всех действий с тремя десятичными знаками эту невырожденную систему нельзя разрешить методом исключения Гаусса с помощью частичного или полного выбора главных элементов: первый же этап исключения приводит к вырожденной системе

$$\begin{cases} 2x_1 + x_2 + x_3 = 2e4 \\ -5e - 1x_2 + 5e - 1x_3 = -e4 \\ -5e - 1x_2 + 5e - 1x_3 = -e4. \end{cases}$$

Кроме того, если заменить гауссов метод треугольных исключений на треугольные исключения методом вращений или методом отражений, которые и были восприняты как не нуждающиеся в выборе главных элементов, то произойдет такая же вычислительная катастрофа! Так может быть, эта система (3) вообще компьютерно неразрешима методами исключений? В действительности она прекрасно решается, если в методе исключения Гаусса отказаться от гауссова принципа главных элементов. В [3] даже указано, что можно удовлетворительно решить систему, если в качестве первого ведущего элемента взять  $a_{21}$ , однако никаким «принципом» этот выбор не удалось оправдать (тогда как отказ вообще от какого бы то ни было принципа выбора может сделать процесс Гаусса несостоятельным [2]). Таким образом, система контрпримера компьютерно невырожденна. И это, к сожалению, позволяет категорически утверждать, что метод исключений Гаусса с выбором главных элементов не гарантирует корректного решения невырожденных систем линейных уравнений в компьютерной арифметике, не обеспечивает и треугольной разложимости невырожденной матрицы сис-

темы вопреки общепринятому мнению, выраженному, в частности, в [2] и [3] (впрочем, в [4] эти же авторы косвенно признали неуниверсальность алгоритма главных элементов, приведя свой контрпример); в общем, не решает численно проблему исключения — фундаментальную проблему вычислительной математики, долгое время ошибочно считавшуюся решенной, и для опровержения упомянутых устоявшихся ошибочных утверждений достаточно и одного этого контрпримера.

В данной работе предлагается новый принцип выбора ведущего элемента (его обоснование приведено ниже), согласно которому первым ведущим элементом должен стать  $a_{31}$ . После первого шага исключений имеем

$$\begin{cases} x_1 - e - 4x_2 + e - 4x_3 = 1 \\ 3e - 4x_2 + 2e - 4x_3 = 5 \\ x_2 + x_3 = 2e4, \end{cases}$$

и далее по этому же принципу

$$\begin{cases} 3e - 4x_2 + 2e - 4x_3 = 5 \\ x_2 + 3.3e - 1x_3 = 3.3e3. \end{cases}$$

Отсюда  $x_3 = e4$ ,  $x_2 = e4$ ,  $x_1 = 1$ . Сравнивая с решением при большем числе разрядов  $x_1 = 1.0010$ ,  $x_2 = 1.0004e4$ ,  $x_3 = 9.9940e3$ , видим, что решение получилось верным практически на все три знака. (Впрочем, абсолютная достоверность результата прямо следует из того факта, что система контрпримера отличается от хорошо обусловленной системы лишь масштабированием и перестановкой уравнений, которая также была представлена и решалась в [3], ее решение  $x_1 = x_2 = x_3 = 1.00$  отличается от решения контрпримера лишь масштабными множителями.) Аналогично можно решить систему (3) и другими методами исключения, для этого необходимо только применить такую же устойчивую последовательность исключений (об этом также речь пойдет ниже).

Данная работа ограничивается теоретическим решением вопроса компьютерной осуществимости методов исключения как методов разложения матрицы с участием треугольного сомножителя и для практического анализа вычислительных погрешностей не обращается к существующей теории ошибок, поскольку автор солидарен с авторами [1], которые скептически относятся к теории ошибок округления для анализа вычислительных ошибок алгоритмов (и подчеркивают это во втором издании данной книги). К тому же скрупулезный анализ ошибок гауссова алгоритма вычислений [5] (не обнаруживший никаких опасностей для компьютерного решения систем!) только подтверждает бесполезность этой теории в данном случае. В то же время точность предлагаемых алгоритмов не может быть хуже «главных элементов», поскольку в отличие от них теперь постоянно соблюдается чрезвычайно бережное отношение к числовой информации.

Из контрпримера ясно, что корректный метод исключения должен основываться на двумерном поиске ведущего элемента по всей матрице, но для простоты изложения придется отказаться от необходимых перестановок строк и столбцов, считая их уже выполненными. Каждый шаг процесса исключения Гаусса рассмат-

ривает преобразуемую часть матрицы как окаймленную сверху и слева:  $\begin{bmatrix} a_{ij} & \mathbf{d}_i \\ \mathbf{c}_j & \mathbf{A}_{ij} \end{bmatrix}$ ,

причем на ведущей позиции этого окаймления располагается «главный элемент» столбца окаймления:  $|a_{ij}| \geq \|\mathbf{c}_j\|_\infty$  для обеспечения устойчивости прямого хода исключений (для удобства сохранен доперестановочный номер строки  $i \neq j$  без ее перенумерации). Само преобразование этой матрицы, реализующее гауссово исключение, в матричной форме имеет вид

$$\mathbf{L}_j^{-1} \cdot \begin{bmatrix} a_{ij} & \mathbf{d}_i \\ \mathbf{c}_j & \mathbf{A}_{ij} \end{bmatrix} = \begin{bmatrix} a_{ij} & \mathbf{d}_i \\ 0 & \mathbf{A}_{ij} - a_{ij}^{-1} \mathbf{c}_j \mathbf{d}_i \end{bmatrix},$$

где  $\mathbf{L}_j^{-1}$  — элементарная матрица гауссова исключения, представляющая собой унитреугольную матрицу с единственным поддиагональным  $j$ -м столбцом —  $a_{ij}^{-1} \mathbf{c}_j$ . Очевидно, что «стержнем» гауссова преобразования является не один элемент  $a_{ij}$ , а весь выбираемый вектор  $(a_{ij}, \mathbf{d}_i)$ , который целиком отходит к формируемой верхней треугольной матрице, так что для обеспечения устойчивости «обратного хода» важно, чтобы элемент  $a_{ij}$  был «главным» в этой строке:  $|a_{ij}| \geq \|\mathbf{d}_j\|_\infty$ . Однако и тот, и другой подходы к обеспечению устойчивости совершенно игнорируют операционную сторону гауссовых исключений, характерную именно для компьютерных вычислений, а не ручных. В связи с этим надо обратить внимание на аддитивную форму всех элементов результирующей матрицы  $\mathbf{A}_{ij} - a_{ij}^{-1} \mathbf{c}_j \mathbf{d}_i$ , так что их компьютерное вычисление чревато опасностью, о которой предупреждает символическая формула (2) и подтверждает контрпример.

Избежать катастрофического затирания информации (2) при гауссовом исключении и тем самым гарантировать треугольную разложимость любой компьютерно невырожденной матрицы можно, если на каждом шаге процесса исключения выбирать ведущий элемент в преобразуемой части матрицы так, чтобы элементы вычитаемой диады  $a_{ij}^{-1} \mathbf{c}_j \mathbf{d}_i$  были по абсолютной величине минимальными из возможных, т.е. оптимальный ведущий элемент (кратко — пивот) должен удовлетворять критерию

$$\min_{i,j} \frac{\|\mathbf{c}_j\|_\infty \cdot \|\mathbf{d}_j\|_\infty}{|a_{ij}|}. \quad (4)$$

Действительно, это условие одновременно делает недопустимыми обе аварийные ситуации: (1) и (2), поскольку именно внося минимальные изменения в преобразуемую часть матрицы, мы предельно уходим от возможности полных (вырождающих) взаимоуничтожений, а значит, избегаем их вследствие предположения о компьютерной невырожденности.

Для этого **оптимального пивотирования** надо перебрать все главные элементы в строках и столбцах, чтобы опробовать их на роль пивота  $a_{ij}$ , а в остальной части строки  $\mathbf{d}_i$  и столбца  $\mathbf{c}_j$ , пересекающихся в испытуемом элементе, надо определить максимальные по абсолютной величине элементы, чтобы вычислить функционал критерия (4). Таким образом, в каждой строке и в каждом столбце преобразуемой части матрицы необходимо выделять не только главные элементы, но и «подглавные», главные в остальной части того же вектора, характеризуя относительное превосходство главных элементов. Все это можно сделать за один обход всех элементов преобразуемой части матрицы, тем не менее найденное оптимальное пивотирование представляется все же излишне сложным (достаточным, но не необходимым).

Функционал критерия (4) нацеливает на наибольшее относительное превалирование ведущего элемента одновременно и в своей строке, и в своем столбце, тогда как для обеспечения незатирания информации при вычислении  $\mathbf{A}_{ij} - a_{ij}^{-1} \mathbf{c}_j \mathbf{d}_i$  необходимо и достаточно максимальности одного относительного превалирования: либо по строке, либо по столбцу, что и определяет более простые и также корректные критерии выбора относительных пивотов — строкового:

$$\min_{i,j} \frac{\|\mathbf{d}_j\|_\infty}{|a_{ij}|} \quad (5)$$

и столбцевого

$$\min_{i,j} \frac{\|\mathbf{c}_j\|_\infty}{|a_{ij}|}, \quad (6)$$

определяя тем самым **относительное пивотирование**.

Действительно, если строка  $(a_{ij}, \mathbf{d}_i)$  (или столбец  $\begin{pmatrix} a_{ij} \\ \mathbf{c}_j \end{pmatrix}$ ) выбрана так, что

в остальной части преобразуемой матрицы строковые (или столбцевые) главные элементы оказались в  $\mathbf{A}_{ij}$ , то это уже исключает возможность катастрофического затирания информации в матрице  $\mathbf{A}_{ij} - a_{ij}^{-1} \mathbf{c}_j \mathbf{d}_i$ . Если же строковые главные элементы находятся также и в столбце  $\mathbf{c}_j$  (или столбцевые в строке  $\mathbf{d}_i$ ), то избежать затирания информации можно посредством перестановки строк (или столбцов), помещая на первое место ту строку (или тот столбец), где наибольшее относительное превалирование главного элемента. Эта процедура выбора заметно проще, чем при оптимальном пивотировании, поскольку здесь только в строках (или только в столбцах) нужно определять два наибольших по абсолютной величине элемента.

Нетрудно заметить полную аналогию алгоритмов относительного и оптимального пивотирования алгоритмам частичного и полного перебора в гауссовых главных элементах. Выбор просто главного элемента по всей матрице был выбором абсолютного пивота и оставался по сути одномерным поиском — с вытягиванием всей матрицы в один составной столбец, поэтому его поисковые возможности не отвечали двумерному объекту — матрице. Оптимальное пивотирование по сравнению с относительным также остается двумерным поиском, поэтому его применение специфическое. (Например, в случае разреженных матриц «оптимальный выбор порядка строк и столбцов крайне важен для экономии памяти и работы, но представляет собой чрезвычайно трудную задачу» [5]. Для ее решения может быть полезно оптимальное пивотирование, если функционал его критерия переопределить, превращая его в функционал трудоемкости: полагая все не равные нулю элементы равными единице и применяя затем 1-норму; в случае стреловидных матриц любой ориентации это автоматически дает оптимальный результат.)

Сложность этих двух видов алгоритмов пивотирования в арифметических операциях для полнозаполненных  $n \times n$ -матриц отличается в пять раз. На первом шаге определение в строках (или в столбцах) абсолютных величин  $\max$  и  $\text{submax}$  требует  $n(n-1)$  операций; затем вычисление величин функционалов относительного пивотирования занимает  $n$  операций, а определение их минимума составляет  $n-1$  операций; итого  $O(n^2)$  операций. Это составляет примерно половину общего числа операций исключения на том же шаге. Значит, по сравнению с гауссовым алгоритмом главных элементов трудоемкость треугольного исключения при относительном пивотировании возрастает в 1,5 раза; тогда как при оптимальном пивотировании на получение информации о  $\max$  и  $\text{submax}$  по всем строкам и столбцам надо на первом шаге  $2n(n-1)$  операций; затем осуществляется  $n^2$  вычислений функционала по две операции на каждое, и определение минимума занимает  $n^2 - 1$  операций; итого  $O(5n^2)$  операций на первом шаге. Таким образом, полная трудоемкость алгоритма с оптимальным пивотированием в 3,5 раза больше гауссова алгоритма с главными элементами.

В методе исключения Гаусса по столбцам наиболее подходящим представляется строковое относительное пивотирование, поскольку гауссовы преобразования исключения основаны на переборе строк (уравнений). И хотя при этом в нижней треугольной матрице  $\mathbf{L}$  могут быть элементы больше единицы по абсолютной величине, зато в верхней треугольной матрице  $\mathbf{U}$  максимально возможное превалирование диагональных элементов обеспечивает наибольшую возможную устойчивость обратного хода.

Столбцевое относительное пивотирование больше всего подходит для альтернативного гауссова метода исключения — для исключений по строкам: которые осуществляются правыми элементарными матрицами  $\mathbf{U}_1^{-1}$  совершенно идентично:

$$\begin{bmatrix} a_{ij} & \mathbf{d}_i \\ \mathbf{c}_j & \mathbf{A}_{ij} \end{bmatrix} \mathbf{U}_1^{-1} = \begin{bmatrix} a_{ij} & \mathbf{0} \\ \mathbf{c}_j & \mathbf{A}_{ij} - a_{ij}^{-1} \mathbf{c}_j \mathbf{d}_i \end{bmatrix},$$

где единственной  $i$ -й строкой в верхней унитреугольной матрице  $\mathbf{U}_1^{-1}$  является  $-a_{ij}^{-1} \mathbf{d}_i$ . Столбцовое относительное пивотирование ведется в пространстве аргументов, где оси координат заданы вектор-столбцами матрицы  $\mathbf{A}$ , и выбор столбцов составит полную нижнюю треугольную матрицу  $\mathbf{L}$  в разложении  $\mathbf{A} = \mathbf{LU}$  (опять отвлекаясь от перестановочных матриц), а матрица  $\mathbf{U}$  будет верхней унитреугольной. Теперь обеспечивается наивысшая возможная устойчивость прямого хода (начинающегося, предположим, с точных данных и, стало быть, в максимальной мере сберегающего точность формируемого промежуточного решения), а в матрице  $\mathbf{U}$  могут быть элементы больше единицы по абсолютной величине (но если они формировались по возможности точно на прямом ходе, то и их вклад в накапливающуюся погрешность минимален).

Решение контрпримера альтернативным методом исключений при столбцовом относительном пивотировании дает разложение

$$\mathbf{LU} = \begin{bmatrix} 1 & & & \\ -e-4 & 2e-4 & & \\ 2e-4 & e-4 & 5e-1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 2 \\ & 1 & 5e3 \\ & & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 2 \\ -e-4 & e-4 & 1 \\ 2e-4 & 3e-4 & 1 \end{bmatrix}$$

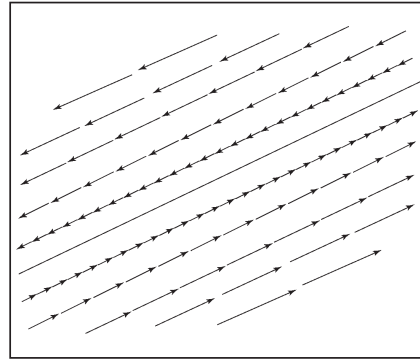
и далее  $\mathbf{Ly} = \begin{bmatrix} 2e4 \\ 1 \\ 6 \end{bmatrix}$ ,  $\mathbf{Uz} = \mathbf{y} = \begin{bmatrix} 2e4 \\ 1.5e4 \\ 1 \end{bmatrix}$ ,  $\mathbf{z} = \begin{bmatrix} x_2 \\ x_3 \\ x_1 \end{bmatrix} = \begin{bmatrix} e4 \\ e4 \\ 1 \end{bmatrix}$ .

Почему-то считается, что альтернативный метод исключений не может дать ничего нового по сравнению с гауссовым. Это опровергает приведенное более устойчивое альтернативное решение и представляется естественным, если вычислитель всегда будет иметь на выбор или для контроля два разных (если матрица  $\mathbf{A}$  несимметричная) однотипных метода исключения: классический Гаусса и альтернативный, это может расширить его возможности конструирования алгоритмических схем.

Однако, кроме классического и альтернативного методов, есть еще два метода исключения, использующих, как и два первых, аналогичные пространственные преобразования, но с иными элементарными унитреугольными матрицами:  $\mathbf{R}_i^{-1}$  — одностроковыми левыми и  $\mathbf{C}_j^{-1}$  — одностолбцовыми правыми. Исключения, которые они позволяют производить, неявные: исключения ортогональных векторов из состава ортогонализуемых исходных, — это известные «методы ортогонализации», они продуцируют разложения матрицы в произведение треугольной и ортогональной матриц:  $\mathbf{A} = \mathbf{RQ}$  или  $\mathbf{A} = \mathbf{QC}$ . Процессы неявных исключений оказываются именно неявными процессами, т.е. нуждающимися в итерационной переортогонализации, так что название «неявные» неслучайно.

Целесообразно обратить внимание на геометрические свойства, объединяющие все четыре вида элементарных унитреугольных преобразований: левых, правых, явных и неявных, тем более, что алгебраически их всех охватывает одна формула:  $\mathbf{I} + \mathbf{uv}^T$ , где один из векторов:  $\mathbf{u}$  или  $\mathbf{v}$ , — координатный орт ( $\mathbf{L}_j = \mathbf{I} + \mathbf{u}_j \mathbf{e}_j^T$ ,  $\mathbf{U}_i = \mathbf{I} + \mathbf{e}_i \mathbf{v}_i^T$ ,  $\mathbf{R}_i = \mathbf{I} + \mathbf{e}_j \mathbf{v}_i^T$ ,  $\mathbf{C}_j = \mathbf{I} + \mathbf{u}_j \mathbf{e}_j^T$ ). В общем случае в преобразованиях с матрицей  $\mathbf{I} + \mathbf{uv}^T$  один вектор из  $\mathbf{u}$  и  $\mathbf{v}$  определяет выделенную ось, а компоненты другого — коэффициенты пропорционального перемещения вдоль конкретной оси, — этот еще один вид аффинных преобразований из-за весьма широкой распространенности должен наконец получить естественное геометрическое название «перекос» (рис. 1). Причем в явных методах исключений перекосу подвергаются все координатные оси вдоль одной выделенной (в методе Гаусса — строки, в альтернативном — столбцы), а в неявных методах подвергается перекосам одна выделенная ось вдоль остальных.

После этого освещения геометрии неортогональных методов исключения (ортогональные изначально имеют именно геометрические названия: методы вращений, методы отражений, причем отражения являются частным случаем перекосов при  $\mathbf{v}^T \mathbf{u} = -2$  и  $\mathbf{v} = -\mathbf{u}$ ) ясно, что критерии относительного пивотирования (5) и (6) — это требования минимальности перекоса выбираемого вектора от очередной координатной оси — минимальности по относительной характеристике их близости, а значит, по направлению, т.е. предпочтение отдается с учетом прямоугольного координатного ориентира (да ведь другого надежного ориентира и нет). Отсюда в неявных методах исключения — процессах ортогонализации — следует искать необходимое упорядочение ортогонализуемых векторов в последовательность устойчивой ортогонализации (без чего, в частности, в контрпримере невозможна ни ортогонализация строк, ни ортогонализация столбцов), предпочитая всякий раз тот вектор, который нуждается в минимальных перекосах, чтобы вместе с построенными ортогональными составить прямоугольную координатную систему.



Şuñ. 1. Изображение аффинного преобразования «перекос»

Говоря об ортогонализации, будем иметь в виду 2-норму. Выбор первого вектора  $\mathbf{q}_1 = \mathbf{a}_1$  поясним позже, а далее на  $k$ -м шаге ортогонализации ( $k = 2, \dots, n - 1$ ) исходные векторы  $\mathbf{a}_m$  (перенумерованные, так что  $m = k, \dots, n$ ) получают новые координаты в виде скалярных произведений  $\langle \mathbf{a}_m, \mathbf{q}_i / \|\mathbf{q}_i\| \rangle$ ,  $i = 1, \dots, k - 1$ , так что коэффициенты ортогонального (неортонормированного) разложения

$$\mathbf{a}_m = \sum_{i=1}^{k-1} \frac{\langle \mathbf{a}_m, \mathbf{q}_i \rangle}{\|\mathbf{q}_i\|^2} \mathbf{q}_i + \mathbf{q}_k \text{ как раз и являются элементами соответствующей элементарной унитарной матрицы. Ортогонализация вектора осуществляется вычитанием его проекции, и минимальность перекоса (относительной длины проекции) оказывается критерием близости } \mathbf{a}_m \text{ к «прямоугольной координатности»:$$

$$\min_m \frac{\sum_{i=1}^{k-1} \langle \mathbf{a}_m, \mathbf{q}_i / \|\mathbf{q}_i\| \rangle^2}{\|\mathbf{a}_m\|^2} = \min_m \sum_{i=1}^{k-1} \cos^2 \{ \mathbf{a}_m, \mathbf{q}_i \}. \quad (7)$$

Это лишь другой нормировкой отличается от критериев относительного пивотирования (5) и (6), в которых, пользуясь эквивалентностью векторных норм, можно заменить  $\infty$ -нормы  $\min_{i,j} \frac{\|\mathbf{d}_j\|_\infty}{\|a_{ij}, \mathbf{d}_j\|_\infty}$  и  $\min_{i,j} \frac{\|\mathbf{c}_j\|_\infty}{\|a_{ij}\|_\infty}$ , например, 2-нормами:

$$\min_{i,j} \frac{\|\mathbf{d}_j\|_2}{\|a_{ij}, \mathbf{d}_j\|_2} \text{ и } \min_{i,j} \frac{\|\mathbf{c}_j\|_2}{\|a_{ij}\|_2}, \text{ получится выражение (7).}$$

Здесь, поскольку преобразование представляет собой переход к новому базису  $\mathbf{Q}$  под  $\mathbf{d}_i$  понимаются пробные строки нижнестроковой унитарной матрицы  $\mathbf{R}$ , а под  $\mathbf{c}_j$  — пробные столбцы верхнестолбцевой унитарной матрицы  $\mathbf{C}$ , поэтому  $a_{ij} = 1$  и формально должно быть не первой компонентой, а последней. Как видно, даже неизвестность «главного элемента» не мешает относительному пивотированию в любой норме: оптимальный выбор всегда производится по остальным

компонентам минимизацией отношения нормы подвектора, образуемого ими, к норме всего вектора. В частности, так следует выбирать и первый вектор  $\mathbf{q}_1$ . Производимая таким образом в контрпримере ортогонализация строк

$$\mathbf{RQ} = \begin{bmatrix} 1 & & \\ 2 & 1 & \\ 1 & 2.5e-4 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -e-4 & e-4 \\ 0 & 1 & 1 \\ e-8 & 5e-5 & -5e-5 \end{bmatrix} = \begin{bmatrix} 1 & -e-4 & e-4 \\ 2 & 1 & 1 \\ 1 & 2e-4 & 3e-4 \end{bmatrix}$$

и ортогонализация столбцов

$$\mathbf{QC} = \begin{bmatrix} 1 & 0 & -e-4 \\ 2e-4 & e-4 & 4e-1 \\ -e-4 & 2e-4 & -2e-1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 2 \\ & 1 & 6e3 \\ & & 1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 2 \\ 2e-4 & 3e-4 & 1 \\ -e-4 & e-4 & 1 \end{bmatrix}$$

дают прекрасные результаты при решении системы (2).

Это «классический» алгоритм ортогонализации Шмидта [2], позволяющий формульно построить унитарную матрицу ортогонализующих перекосов. Но факторизовать ее вовсе не обязательно посредством элементарных унитарных верхнестолбцовых матриц  $\mathbf{C}_j$  (при ортогонализации столбцов) или нижнестроковых  $\mathbf{R}_i$  (при ортогонализации строк), как делается в алгоритмах неявных (ортогонализационных) исключений Шмидта. Факторизовать ее можно и посредством элементарных унитарных матриц, используемых в явных исключениях: нижнестолбцовых  $\mathbf{L}_j$  или верхнестроковых  $\mathbf{U}_i$ . Это будет покоординатный процесс (в новых координатах), на каждом шаге которого из всех ортогонализуемых векторов будут вычитаться проекции очередного ортогонализованного, что совершенно аналогично поэлементным исключениям в явных методах — Гаусса и альтернативного. На языке алгоритмов эта факторизация означает принципиально другую схему ортогонализации:

$$\text{for } m = k \text{ to } n \text{ do } a_m = a_m - \left\langle a_m, q_{k-1} / \|q_{k-1}\|^2 \right\rangle q_{k-1}. \quad (8)$$

Ее (операторная) запись подразумевает возможность итерационных уточнений, существенно ужесточающих процесс ортогонализации и поэтому позволяющих получать значительно лучшую точность.

Такая схема ортогонализации эмпирически была найдена Райсом [6], но его название «модифицированная ортогонализация Грама-Шмидта» больше соответствует алгоритму расчленения формул Шмидта  $\mathbf{q}_k = \mathbf{a}_m - \sum_{i=1}^{k-1} \frac{\langle \mathbf{a}_m, \mathbf{q}_i \rangle}{\|\mathbf{q}_i\|^2} \mathbf{q}_i$  на простые рекурсивные разности [5]:

$$q_k = a_m; \text{ for } i = 1 \text{ to } k-1 \text{ do } q_k = q_k - \left\langle q_k, q_i / \|q_i\|^2 \right\rangle q_i.$$

Эта модификация также подразумевает возможность итерационного уточнения каждой разности, что оказывается лучше, чем общая переортогонализация для формул Шмидта:

$$q_k = a_m; q_k = q_k - \sum_{i=1}^{k-1} \frac{\langle q_k, q_i \rangle}{\|q_i\|^2} q_i.$$

Тем не менее выбор оптимального вектора  $\mathbf{a}_m$  по указанному выше критерию относительного пивотирования и для модифицированной схемы остается очень трудоемким, тогда как ортогонализация Райса (8) предельно проста для применения относительного пивотирования

$$\min_{k \leq m \leq n} \left\| \left\langle \mathbf{a}_m, \mathbf{q}_{k-1} / \|\mathbf{q}_{k-1}\| \right\rangle / \|\mathbf{a}_m\| \right\| = \min_{k \leq m \leq n} |\cos \{ \mathbf{a}_m, \mathbf{q}_{k-1} \}|.$$

Очевидно, что это также лишь перефразировка критериев (5) и (6) в новых ортогонализированных координатах (теперь это просто двумерный случай  $\{ \mathbf{q}_{k-1}, \mathbf{q}_k \}$ ).



Для ортогонализации Шмидта можно построить точную аналогию в явных методах исключений: либо гауссово исключение, но не по столбцам, а степенями по строкам: во второй строке только одного элемента  $a_{21}$ , в третьей двух элементов:  $a_{31}$  и  $a_{32}$  и т.д.; либо альтернативное исключение степенями по столбцам: в элементе  $a_{12}$ , затем  $a_{13}$  и  $a_{23}$  и т.д. Очевидна нерациональность такой схемы исключений, как и ее неявного аналога — ортогонализации Шмидта.

И наконец, кроме этих неявных (ортогонализационных) методов исключения есть еще методы исключений, «синтезирующие» в себе явные качества с ортогональностью, это явные исключения посредством ортогональных преобразований методами отражений и вращений. (В [1] эти треугольные исключения (левые) названы модификациями метода исключений Гаусса.) Как показал тот же контрпример, они также нуждаются в пивотировании. Оказывается, что относительное пивотирование прекрасно подходит и для этих явных исключений, объясняется это тем, что формулы этих преобразований также представимы в виде суммы главной части с малой поправкой в случае удачного пивотирования.

Так, метод отражений (левых) преобразует столбцы матрицы в разности

$$\left[ \mathbf{I} - \frac{2}{\|\mathbf{w}\|^2} \mathbf{w}\mathbf{w}^T \right] \cdot [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n] = \left[ \alpha \mathbf{e}_1, \mathbf{a}_2 - \frac{\langle \mathbf{w}, \mathbf{a}_2 \rangle}{\langle \mathbf{w}, \mathbf{a}_1 \rangle} \mathbf{w}, \dots, \mathbf{a}_n - \frac{\langle \mathbf{w}, \mathbf{a}_n \rangle}{\langle \mathbf{w}, \mathbf{a}_1 \rangle} \mathbf{w} \right],$$

где  $\mathbf{w} = \mathbf{a}_1 - \alpha \mathbf{e}_1$  с  $\alpha = \pm \|\mathbf{a}_1\|$  — исключение в первом столбце. Очевидно, чтобы не допустить аддитивного затирания информации (2) под первой строкой этой матрицы, малость вычитаемых столбцов (пропорциональных  $\mathbf{w}$ ) можно обеспечить только близостью направления вектора-пivота  $\mathbf{a}_1$  к первой координатной оси, — а эту геометрию обеспечивает столбцовое относительное пивотирование.

Исключения в первом столбце по порядку сверху вниз методом вращений будут придавать его ведущему элементу последовательные значения

$$a_{11}^{(i)} = \pm \sqrt{\sum_{k=1}^i a_{k1}^2}. \text{ Остальные элементы под первой строкой примут вид}$$

$$\frac{a_{11}^{(i-1)}}{a_{11}^{(i)}} a_{ij} - \frac{a_{i1}}{a_{11}^{(i)}} a_{ij} = \frac{a_{11}^{(i-1)}}{a_{11}^{(i)}} \left( a_{ij} - \frac{a_{i1} \cdot a_{1j}}{a_{11}^{(i-1)}} \right). \text{ В поправочном слагаемом в скобке}$$

мы узнаем элементы той же диады, которая прежде обозначалась  $\mathbf{c}_j \mathbf{d}_i$  (произведение первого поддиагонального столбца на первую наддиагональную строку), а так как  $|a_{11}^{(i)}| \leq |a_{11}^{(i-1)}| \leq |a_{11}^{(i)}|$ , то применимы рассуждения (4)–(6), т.е. если выполнены условия критериев оптимального или относительного пивотирования, то успех применения метода вращений обеспечен.

Итак, теоретически обосновано, что оптимальное и относительное пивотирование — универсальный метод, при условии компьютерной невырожденности гарантирующий всем основным методам успешное компьютерное решение фундаментальной проблемы исключения.

#### СПИСОК ЛИТЕРАТУРЫ

1. Фаддеев Д.К., Фаддеева В.Н. Вычислительные методы линейной алгебры. — М.: Физматгиз, 1956. — 656 с.
2. Уилкинсон Дж.Х. Алгебраическая проблема собственных значений. — М.: Наука, 1970. — 564 с.
3. Форсайт Дж., Молер К. Численное решение систем линейных алгебраических уравнений. — М.: Мир, 1969. — 167 с.
4. Форсайт Дж., Малькольм М., Молер К. Машинные методы математических вычислений. — М.: Мир, 1980. — 279 с.
5. Деммель Дж. Вычислительная линейная алгебра. Теория и приложения. — М.: Мир, 2001. — 430 с.
6. Rice J.R. Experiments with Gram-Schmidt orthogonalization // Math. Comput. — 1966. — 20, N 94. — P. 325–328.

Поступила 04.12.2006  
После доработки 08.01.2008