

АЛГЕБРАИЧЕСКАЯ ПРОБЛЕМА СОБСТВЕННЫХ ЗНАЧЕНИЙ

В главе VI рассмотрены методы нахождения собственных значений и собственных векторов квадратных матриц. В § 1 изложены необходимые сведения по линейной алгебре, рассмотрена устойчивость проблемы собственных значений и даны простые (но сравнительно медленные) численные методы решения. Наиболее быстрые численные методы нахождения всех собственных значений и собственных векторов эрмитовых матриц разобраны в § 2, а неэрмитовых матриц — в § 3. В § 4 изложены методы, которые оказываются более выгодными при определении не всех, а некоторых собственных значений и собственных векторов.

§ 1. Проблема и простейшие методы

1. Элементы теории. Напомним некоторые сведения из курса линейной алгебры. Если A — квадратная матрица n -го порядка и $Ax = \lambda x$ при $x \neq 0$, то число λ называется *собственным значением* матрицы, а ненулевой вектор x — соответствующим ему *собственным вектором*. Перепишем задачу в виде

$$(A - \lambda E)x = 0, \quad x \neq 0. \quad (1)$$

Для существования нетривиального решения задачи (1) должно выполняться условие

$$\det(A - \lambda E) = 0. \quad (2)$$

Этот определитель является многочленом n -й степени от λ ; его называют *характеристическим многочленом*. Значит, существует n собственных значений — корней этого многочлена, среди которых могут быть одинаковые (кратные). В принципе можно вычислить характеристический многочлен и найти все его корни.

Если найдено некоторое собственное значение, то, подставляя его в однородную систему (1), можно определить соответствующий собственный вектор. Будем нормировать собственные векторы*). Тогда каждому простому (не кратному) собственному

*) Нормировкой (на единицу) вектора x называют умножение его на $\|x\|^{-1}$; нормированный вектор имеет единичную длину.

значению соответствует один (с точностью до направления) собственный вектор, а совокупность всех собственных векторов, соответствующих совокупности простых собственных значений, линейно-независима. Таким образом, если все собственные значения матрицы простые, то она имеет n линейно-независимых собственных векторов, которые образуют базис пространства.

Кратному собственному значению кратности p может соответствовать от 1 до p линейно-независимых собственных векторов. Например, рассмотрим такие матрицы четвертого порядка:

$$A = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ 0 & 0 & a & 0 \\ 0 & 0 & 0 & a \end{bmatrix}, \quad C_4 = \begin{bmatrix} a & 1 & 0 & 0 \\ 0 & a & 1 & 0 \\ 0 & 0 & a & 1 \\ 0 & 0 & 0 & a \end{bmatrix}, \quad B = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & a & 1 & 0 \\ 0 & 0 & a & 1 \\ 0 & 0 & 0 & a \end{bmatrix}. \quad (3)$$

У каждой из них характеристическое уравнение принимает вид $\det(A - \lambda E) = (a - \lambda)^4 = 0$, а следовательно, собственное значение $\lambda = a$ и имеет кратность $p = 4$. Однако у первой матрицы есть четыре линейно-независимых собственных вектора

$$e_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad e_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}, \quad e_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad e_4 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}; \quad (4)$$

это легко проверить, поочередно подставляя векторы (4) в равенство (1). У второй же матрицы имеется только один собственный вектор e_1 . В самом деле, пусть ее собственный вектор x имеет компоненты x_i ; тогда уравнение (1) примет для нее вид

$$(G_4 - \lambda E)x = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_3 \\ x_4 \\ 0 \end{pmatrix} = 0, \quad \lambda = a,$$

откуда $x_2 = x_3 = x_4 = 0$, а $x_1 = 1$ в силу условия нормировки. Вторую матрицу называют простой *жордановой* (или классической) *подматрицей*. Третья матрица имеет так называемую каноническую жорданову форму (по диагонали стоят либо числа, либо жордановы подматрицы, а остальные элементы равны нулю). Ее собственными векторами являются e_1 и e_2 ; в этом легко убедиться при помощи выкладки, аналогичной только что сделанной.

Таким образом, если среди собственных значений матрицы есть кратные, то ее собственные векторы не всегда образуют базис. Однако и в этом случае собственные векторы, соответствующие различным собственным значениям, являются линейно-независимыми.

Задача на собственные значения легко решается для некоторых простых форм матрицы: диагональной, трехдиагональной, треугольной или почти треугольной. Например, определитель треугольной (в частности, диагональной) матрицы равен произведению диагональных элементов. В этом случае $A - \lambda E$ тоже треугольная или диагональная матрица. Поэтому *собственные значения треугольной (диагональной) матрицы равны диагональным элементам*. Легко проверить, что диагональная матрица имеет n собственных ортонормированных векторов $e_i = \{0, \dots, 0, 1, 0, \dots, 0\}^T$, соответствующих собственным значениям $\lambda_i = a_{ii}$; наоборот, матрица с такими собственными векторами диагональна.

Многие численные методы решения задач на собственные значения основаны на приведении матрицы к одной из перечисленных выше простых форм при помощи преобразования подобия. Матрица $G = F^{-1}AF$ называется *подобной* матрице A . Пусть η , y суть собственное значение и собственный вектор матрицы G ; тогда $\eta y = Gy = F^{-1}AFy$, что после умножения слева на матрицу F дает $\eta(Fy) = A(Fy)$. Отсюда видно, что η и Fy суть собственное значение и собственный вектор матрицы A . Следовательно, *преобразование подобия не меняет собственных значений матрицы и по определенному закону преобразует ее собственные векторы*.

Особенно удобны преобразования подобия при помощи унитарных матриц *). Если ортонормированный базис преобразовать унитарной матрицей, то он останется ортонормированным. Если подобно преобразовать эрмитову матрицу при помощи унитарной, то она остается эрмитовой; в самом деле,

$$B = U^H A U, \quad B^H = (U^H A U)^H = U^H A^H (U^H)^H = U^H A U = B.$$

Если A — нормальная матрица, то при подобном унитарном преобразовании она остается нормальной; читателям предлагается проверить это.

Известно, что для любой матрицы A есть такое унитарное преобразование, что $U^H A U$ является верхней треугольной матри-

) Напомним принятую терминологию. Матрица B называется эрмитово сопряженной к матрице A , если она получена из нее транспонированием (зеркальным отражением от главной диагонали) с последующей заменой элементов комплексно-сопряженными, т. е. $B = A^H$, если $b_{ik} = a_{ki}^$. Матрица эрмитова, если она эрмитово сопряжена самой себе: $A = A^H$, и косоэрмитова, если она удовлетворяет соотношению $A = -A^H$. Вещественная эрмитова матрица называется симметричной, а косоэрмитова — кососимметричной. Унитарной называется матрица, обратная своей эрмитово сопряженной: $U^H = U^{-1}$; вещественные унитарные матрицы называют ортогональными. Матрица называется нормальной, если она перестановочна со своей эрмитово сопряженной, т. е. $AA^H = A^H A$. Легко видеть, что эрмитовые, косоэрмитовые и унитарные матрицы являются частными случаями нормальных.

цей; если A — нормальная матрица, то это унитарное преобразование приводит ее к диагональной форме.

Непосредственно для практических вычислений теорема Шура ничего не дает, ибо неизвестен способ нахождения такого унитарного преобразования. Но одно косвенное следствие является важным. После указанного преобразования нормальная матрица A становится диагональной; тогда ее новые собственные векторы образуют ортонормированный базис e_i . Следовательно, собственные векторы исходной нормальной матрицы получаются из ортонормированного базиса e_i унитарным преобразованием и сами образуют ортонормированный базис.

Это существенно, ибо в практике вычислений часто встречаются нормальные матрицы, особенно их такие частные случаи, как эрмитовы, косоэрмитовы и унитарные матрицы. Ортогональные же преобразования обеспечивают наибольшую устойчивость алгоритма по отношению к ошибкам округления. Действия с неортогональными базисами и преобразованиями при больших порядках матрицы нередко приводят к «разболтке» счета (это уже отмечалось в главе II в связи с вопросами аппроксимации).

Не всякую матрицу с кратными собственными значениями можно подобно преобразовать к диагональной форме, но ее заведомо можно преобразовать к канонической жордановой форме. Если же матрица имеет только простые собственные значения, то существует преобразование подобия (не обязательно унитарное), приводящее ее к диагональной. В самом деле, такая матрица имеет n линейно-независимых собственных векторов. Матрица F , столбцами которой являются координаты этих векторов, преобразует базис e_i в базис из собственных векторов. Значит, преобразование подобия с матрицей F приводит A к диагональной форме.

2. Устойчивость. Для исследования устойчивости проблемы собственных значений надо наряду с матрицей A рассмотреть эрмитово сопряженную к ней матрицу A^H . Поскольку при транспонировании матрицы ее определитель не меняется, а замена всех матричных элементов комплексно сопряженными величинами приводит к замене определителя тоже комплексно сопряженным числом, то $\det(A^H - \lambda^* E) = [\det(A - \lambda E)]^*$. Отсюда видно, что если λ_i есть собственное значение матрицы A , то $\det(A^H - \lambda_i^* E) = 0$, то есть λ_i^* есть собственное значение матрицы A^H . Следовательно, *собственные значения эрмитово сопряженных матриц комплексно сопряжены друг другу.*

Обозначим собственные векторы матриц A и A^H соответственно через x_i и y_i . Докажем, что *собственные векторы сопряженных матриц, соответствующие различным (точнее, не комплексно-сопряженным друг другу) собственным значениям, взаимно*

ортогональны. Для этого напишем тождества

$$A\mathbf{x}_i = \lambda_i \mathbf{x}_i, \quad A^H \mathbf{y}_j = \lambda_j^* \mathbf{y}_j.$$

Скалярно умножим *) первое равенство слева на \mathbf{y}_j , а второе — справа на \mathbf{x}_i и вычтем одно из другого; получим

$$(\mathbf{y}_j, A\mathbf{x}_i) - (A^H \mathbf{y}_j, \mathbf{x}_i) = (\mathbf{y}_j, \lambda_i \mathbf{x}_i) - (\lambda_j^* \mathbf{y}_j, \mathbf{x}_i).$$

Выражение в левой части этого равенства равно нулю. Вынося λ из скалярных произведений правой части этого равенства, получим $(\lambda_i - \lambda_j) (\mathbf{y}_j, \mathbf{x}_i) = 0$ или

$$(\mathbf{y}_j, \mathbf{x}_i) = 0 \quad \text{при} \quad \lambda_i \neq \lambda_j, \quad (5)$$

что и требовалось доказать.

Отсюда следует, что у эрмитовых матриц собственные значения вещественны, а собственные векторы образуют ортогональную систему (поскольку $\mathbf{y}_j = \mathbf{x}_j$).

Рассмотрим устойчивость проблемы собственных значений. Для простоты ограничимся случаем, когда собственные векторы матрицы образуют базис, а данное собственное значение — простое.

Если немного изменить матричные элементы, то поправки к собственному значению и соответствующему вектору с точностью до величин второго порядка малости удовлетворяют линеаризованному уравнению

$$A \delta \mathbf{x}_i + \delta A \cdot \mathbf{x}_i \approx \delta \lambda_i \cdot \mathbf{x}_i + \lambda_i \delta \mathbf{x}_i. \quad (6)$$

Разложим поправку $\delta \mathbf{x}_i$ по невозмущенным собственным векторам. Вектор $\mathbf{x}_i + \delta \mathbf{x}_i$ определен с точностью до множителя; подберем этот множитель так, чтобы диагональный коэффициент разложения обратился в нуль:

$$\delta \mathbf{x}_i = \sum_{j=1}^n \xi_{ij} \mathbf{x}_j, \quad \xi_{ii} = 0.$$

Подставляя это разложение в (6) и умножая слева на различные собственные векторы сопряженной матрицы, получим

$$(\mathbf{y}_i, \mathbf{x}_i) \delta \lambda_i \approx (\mathbf{y}_i, \delta A \cdot \mathbf{x}_i), \quad \xi_{ij} (\lambda_j - \lambda_i) (\mathbf{y}_j, \mathbf{x}_j) \approx (\mathbf{y}_j, \delta A \cdot \mathbf{x}_i).$$

Поскольку вариация матрицы может быть любой, то максимумы правых частей обоих последних равенств равны $\|\mathbf{x}\| \cdot \|\mathbf{y}\| \times \times \max |\delta a_{kl}|$. Тогда максимально возможные ошибки собственного значения и компонент собственного вектора не превышают (с точ-

) Напомним, что для комплексных векторов скалярное произведение равно $(\mathbf{a}, \mathbf{b}) = \sum_{k=1}^n a_k^ b_k$.

ностью до отброшенных в ходе выкладок бесконечно малых более высокого порядка) следующих значений:

$$|\delta\lambda_i| \lesssim \kappa_i \max_{k,l} |\delta a_{kl}|, \quad |\xi_{ij}| \lesssim \frac{\kappa_j}{|\lambda_i - \lambda_j|} \max_{k,l} |\delta a_{kl}|. \quad (7)$$

Здесь через κ_i обозначен так называемый i -й коэффициент перекоса матрицы

$$\kappa_i = \frac{\sqrt{(x_i, x_i)(y_i, y_i)}}{(x_i, y_i)} = \frac{1}{\cos \varphi_i}, \quad (8)$$

где φ_i есть угол между соответствующими векторами данной матрицы и эрмитово сопряженной к ней.

Заметим, что для эрмитовой матрицы все коэффициенты перекоса равны единице, поскольку соответствующие векторы ортогональны. А для типичной жордановой клетки

$$A = \begin{bmatrix} a & 1 \\ 0 & a \end{bmatrix}, \quad x_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad A^n = \begin{bmatrix} a^n & 0 \\ 1 & a^n \end{bmatrix}, \quad y_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

выполняется условие $(x_1, y_1) = 0$, т. е. коэффициент перекоса обращается в бесконечность. Очевидно, что для любых матриц $|\kappa_i| \geq 1$.

Выводы из оценки (7) можно сформулировать следующим образом. Собственное значение устойчиво относительно вариации матричных элементов, если соответствующий ему коэффициент перекоса мал; если этот коэффициент перекоса очень велик, то устойчивость может быть плохой. Собственный вектор устойчив по матричным элементам, если все коэффициенты перекоса матрицы невелики, а данное собственное значение — простое.

Значит, все собственные значения эрмитовых матриц мало чувствительны к погрешностям матричных элементов. А собственные значения жордановых подматриц могут быть очень чувствительны к погрешностям. Проиллюстрируем последнее на примере неэрмитовой матрицы 20-го порядка:

$$A = \begin{bmatrix} 20 & 20 & 0 & 0 & \dots & 0 & 0 \\ 0 & 19 & 20 & 0 & \dots & 0 & 0 \\ 0 & 0 & 18 & 20 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 2 & 20 \\ \varepsilon & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \quad (9)$$

где через ε обозначено малое возмущение нулевого углового элемента. Характеристическое уравнение этой матрицы имеет вид

$$\det(A - \lambda E) = \prod_{k=1}^{20} (k - \lambda) - 20^{19}\varepsilon = 0. \quad (10)$$

При $\varepsilon = 0$ младший коэффициент характеристического многочлена есть $a_0 = 20! \approx 2,5 \times 10^{18}$, а наименьшее по модулю собственное значение $\lambda_{20} = 1$. Если же положить $\varepsilon = 20^{-19} \times 20! \approx 5 \times 10^{-7}$, то коэффициент a_0 обращается в нуль, а тогда $\lambda_{20} = 0$. Таким образом, и коэффициенты и сами корни характеристического многочлена могут быть очень чувствительны к малым погрешностям матричных элементов, что означает слабую устойчивость задачи. Это согласуется со сделанным в § 2 главы V замечанием о том, что корни многочлена высокой степени нередко чувствительны к погрешностям коэффициентов.

Но для эрмитовых матриц собственные значения хорошо устойчивы по матричным элементам. Даже для неэрмитовых матриц опасна вариация не любого коэффициента; например, к возмущениям элементов главной диагонали собственные значения мало чувствительны.

3. Метод интерполяции. Если мы найдем характеристический многочлен, то все его корни нетрудно вычислить, например, методом парабол. В методе парабол для нахождения одного корня обычно требуется около 10 раз вычислить многочлен. Поэтому важно найти способ быстрого вычисления характеристического многочлена.

Те методы решения проблемы собственных значений, которые позволяют определить характеристический многочлен за конечное число действий, называются прямыми. Методы, в которых характеристический многочлен определяется как предел некоторого итерационного процесса, называются итерационными. Это разделение носит несколько условный характер, ибо даже если характеристический многочлен найден за конечное число действий, то его корни приходится определять итерационным процессом. Однако оно имеет практический смысл, поскольку нахождение характеристического многочлена высокой степени гораздо более трудоемко, чем отыскание его корней.

Простейшим прямым методом является *метод интерполяции* (предложенный, по-видимому, Ш. Е. Микеладзе в 1948 г.) Известно, что многочлен n -й степени однозначно определяется своими значениями в $n + 1$ узле. Произвольно выберем $n + 1$ значение $\lambda^{(k)}$ в качестве таких узлов. Вычислим в них значение $f(\lambda^{(k)}) = \det(A - \lambda^{(k)} E)$ и построим по этим значениям интерполяционный многочлен Ньютона при помощи формул (2.6) и (2.8). В силу единственности этот многочлен будет характеристическим. Он при этом получается в форме многочлена с заданными коэффициентами, так что дальнейшие вычисления для нахождения его корней потребуют малого числа действий.

Описанный алгоритм несложен и легко программируется на ЭВМ. В нем следует использовать стандартную программу вычисления определителя методом исключения (глава V, § 1, п. 3). При этом характеристический многочлен определяется примерно за $\frac{2}{3}n^4$ арифметических действий, из которых половину составляют сложения и половину — умножения. Видно, что для мат-

риц невысокого порядка $n \leq 10$ нахождение характеристического многочлена методом интерполяции требует не более 0,5 сек на ЭВМ БЭСМ-4, что вполне приемлемо.

Если известны границы, в которых расположены собственные значения, то целесообразно размещать узлы интерполяции $\lambda^{(k)}$ в этих границах, причем приблизительно равномерно. Это уменьшает ошибки округления, возникающие при нахождении разделенных разностей в формуле Ньютона, т. е. улучшает устойчивость алгоритма. Для определения границ спектра можно воспользоваться оценкой $|\lambda_i| \leq \|A\|$, справедливой для любой нормы матрицы (это следует из того, что спектральная норма $\|A\|_\lambda = \max |\lambda_i|$ есть наименьшая из норм матрицы). Хотя эта оценка в среднем завышена, но она достаточно разумна для тех матриц, с которыми приходится встречаться на практике, и тех норм (см. § 2 главы I), которые просто вычисляются.

Метод интерполяции прост и применим для матриц произвольной структуры, а также для более сложных проблем. Например, общую задачу $\det [p_{ik}(\lambda)] = 0$, где каждый элемент матрицы есть некоторый многочлен от λ , решают практически только этим методом; разумеется, число узлов по λ выбирают в соответствии со степенью результирующего многочлена. Лишь в частном случае этой задачи — так называемой обобщенной проблемы собственных значений $\det (A - \lambda B) = 0$, разработаны более экономичные методы.

Однако, чем выше порядок матрицы, тем менее выгоден метод интерполяции. Во-первых, число выполняемых арифметических действий возрастает с ростом порядка очень быстро — как n^4 . Во-вторых, при составлении интерполяционного многочлена Ньютона вычисляются разделенные разности, что при высоких порядках приводит к большой потере точности. Поэтому при $n \gtrsim 10$ (а в случае кратных или близких собственных значений и при меньших n) метод интерполяции дает плохие результаты. Для матриц высокого порядка применяют более сложные, но зато более устойчивые и экономичные методы, изложенные в следующих параграфах.

Существуют прямые методы Леверье, А. Н. Крылова, А. М. Данилевского, Самуэльсона и Ланцоша, позволяющие вычислить все коэффициенты характеристического многочлена произвольной матрицы примерно за n^3 арифметических действий. Они экономичней метода интерполяции.

Однако в § 2 главы V отмечалось, что корни многочлена высокой степени могут быть очень чувствительны к погрешностям коэффициентов. Кроме того, и коэффициенты и сами корни характеристического многочлена нередко слабо устойчивы по матричным элементам, как было показано в п. 2. Поэтому указанные выше экономичные методы оказались достаточно устойчивыми только для матриц невысокого порядка $n \leq 10$, а при наличии кратных или близких собственных значений — для еще меньшего порядка. Но при таких порядках матрицы экономия по сравнению с методом интерполяции невелика и не оправдывает применения этих довольно сложных и капризных методов.

4. Трехдиагональные матрицы. В интерполяционном методе мы находили явное выражение для характеристического многочлена только затем, чтобы иметь способ экономного вычисления этого многочлена при заданных λ . Однако для трехдиагональных матриц (даже очень высокого порядка) есть способ быстрого вычисления $\det(A - \lambda E)$ без нахождения явного выражения характеристического многочлена. Это существенно, ибо матрицы общего вида можно привести к трехдиагональной форме преобразованием подобия.

Рассмотрим этот способ. Обозначим главный минор m -го порядка матрицы $A - \lambda E$ через $D_m(\lambda)$. Разложим такой минор по элементам последней строки; в ней всего два ненулевых элемента (рис. 30), так что получим

$$D_m(\lambda) = (a_{mm} - \lambda) D_{m-1}(\lambda) - a_{m, m-1} B_{m, m-1}(\lambda), \quad (11)$$

где через $B_{m, m-1}(\lambda)$ обозначен минор, дополняющий элемент $a_{m, m-1}$. Этот минор содержит в последнем столбце только один ненулевой элемент $a_{m-1, m}$, поэтому его целесообразно разложить по элементам последнего столбца:

$$B_{m, m-1}(\lambda) = a_{m-1, m} D_{m-2}(\lambda). \quad (12)$$

Подставляя (12) в (11), найдем рекуррентное соотношение, выражающее минор высшего порядка через низшие:

$$D_m(\lambda) = (a_{mm} - \lambda) D_{m-1}(\lambda) - a_{m, m-1} a_{m-1, m} D_{m-2}(\lambda). \quad (13)$$

Для начала расчета по рекуррентной формуле надо задать два первых минора. Удобно формально положить

$$D_{-1}(\lambda) = 0, \quad D_0(\lambda) = 1. \quad (14)$$

Подставляя эти значения в (13) и вычисляя $D_1(\lambda)$ и $D_2(\lambda)$, легко убедиться, что результат получается правильным. Следовательно, такой способ начала счета приемлем.

Однократный расчет величины определителя по формулам (11) — (14) требует всего $5n$ арифметических действий, причем среди них нет делений, и вычисления очень быстрые и устойчивые. Таким образом, имеется быстрый способ нахождения характеристического многочлена при заданном значении λ .

Имеется способ нахождения характеристического многочлена, более экономичный при многократных вычислениях. Преобразуя рекуррентное соотношение (13), можно получить коэффициенты характеристического многочлена в форме Горнера. Однократное же вычисление многочлена по схеме Горнера требует всего $2n$ действий. Однако устойчивость этого процесса для высоких порядков матрицы, по-видимому, хуже, а формулы расчета более сложны.

$D_{m-2}(\lambda)$		0		0
0		$a_{m-1, m-2}$		$a_{m-1, m}$
0		$a_{m, m-1}$		$a_{mm} - \lambda$

Рис. 30.

Еще один несложный способ вычисления характеристического многочлена заключается в следующем. Вычтем заданное значение λ из диагональных элементов a_{ii} , а затем найдем определитель получившейся трехдиагональной матрицы по формулам прямого хода прогонки (5.12)—(5.13). Однако этот способ менее экономичен и устойчив, чем расчет по формуле (13).

Корни многочлена $D_n(\lambda)$ удобнее всего находить методом парабол (см. § 2 главы V). Этот метод для многочленов не слишком высокой степени ($n \lesssim 50$) достаточно устойчив и позволяет найти все корни с 5—7 верными знаками, даже если среди корней есть кратные. В библиотеках многих ЭВМ имеются стандартные программы вычислений всех корней многочлена методом парабол.

Иногда для нахождения всех корней характеристического многочлена употребляют метод Ньютона, но детали такого алгоритма хуже отработаны. В основном метод Ньютона применяют к частичной проблеме собственных значений.

Заметим, что при любом способе вычислений для нахождения всех корней требуется удалять уже полученные корни, т. е. переходить к вспомогательной функции $G(\lambda) = D_n(\lambda) / \prod_{i=1}^k (\lambda - \lambda_i)$.

Поскольку явного выражения характеристического многочлена мы не выписываем, то для вычисления $G(\lambda)$ надо находить отдельно числитель и знаменатель при требуемых значениях λ . Это немного увеличивает объем расчетов.

Описанным способом все собственные значения трехдиагональной матрицы находятся довольно легко, причем для этого требуется всего около $50n^2$ арифметических действий*), т. е. способ экономичен. Поэтому для трехдиагональных матриц этот способ является основным.

5. Почти треугольные матрицы. Для такой матрицы также можно написать формулы, позволяющие легко вычислить определитель при заданном значении λ . Это удобно делать методом исключения Гаусса, учитывая большое количество нулей в матрице определителя.

Используем формулы метода исключения (5.3)—(5.5). Для определенности будем считать нашу матрицу верхней почти треугольной. Тогда видно, что $c_{mk} = 0$ при $m > k + 1$, а каждый цикл исключения сводится всего лишь к вычитанию двух строк. Достаточно при этом пометить изменяющиеся величины штрихом, опуская верхний индекс цикла. После этого формулы k -го цикла примут вид

$$c_k = \frac{a_{k+1, k}}{a'_{kk}}, \quad a'_{k+1, i} = a_{k+1, i} - c_k a'_{ki}, \quad k+1 \leq i \leq n, \quad (15)$$

причем $a'_{k+1, k} = 0$. Последовательно полагая $k = 1, 2, \dots, n-1$, аннулируем все поддиагональные элементы. После этого определитель легко вычисляется

*) Метод парабол обычно сходится менее чем за 10 итераций, а одна итерация требует $5n$ действий. Эти цифры мы будем использовать при описании других методов.

по формуле Чжо (5.8):

$$\det A = \prod_{k=1}^n a'_{kk}, \quad a'_{11} = a_{11}. \quad (16)$$

Поскольку нас интересует $\det(A - \lambda E)$, то для его вычисления надо в формулах (15) — (16) вместо нештрихованных величин a_{kk} всюду подставить $a_{kk} - \lambda$. Этот способ позволяет вычислить определитель за n^2 арифметических действий.

Как и для трехдиагональной матрицы, корни характеристического многочлена можно находить методом парабол. Тогда нахождение всех корней потребует около $10n^3$ действий. Видно, что метод оказывается не быстрым, но довольно простым и устойчивым. Дальше мы увидим, что есть заметно более быстрые способы нахождения собственных значений почти треугольной матрицы, основанные на преобразовании матрицы к трехдиагональной форме. Но они более сложны и менее устойчивы.

6. Обратные итерации. Если собственное значение известно, то собственный вектор удовлетворяет системе (1). Но любой численный метод дает вместо точного собственного значения λ_i приближенное значение $\tilde{\lambda}_i$, так что $\det(A - \tilde{\lambda}_i E) \neq 0$, хотя отличается от нуля очень мало. В таком случае задача $(A - \tilde{\lambda}_i E) \mathbf{x} = \mathbf{0}$ при использовании приближенного собственного значения имеет только тривиальное решение $\mathbf{x} = \mathbf{0}$. Поэтому в численных расчетах находить собственные векторы непосредственно из системы (1) нельзя.

Для нахождения собственных векторов удобен метод *обратной итерации*, заключающийся в следующем. Выберем наудачу вектор \mathbf{b} и рассмотрим линейную неоднородную систему

$$(A - \tilde{\lambda}_i E) \mathbf{x} = \mathbf{b}. \quad (17)$$

Определитель этой системы отличен от нуля, так что она имеет единственное решение. Покажем, что найденный из нее вектор \mathbf{x} окажется почти равным собственному вектору \mathbf{x}_i , соответствующему данному собственному значению λ_i .

Для простоты ограничимся случаем, когда матрица n -го порядка имеет n линейно-независимых собственных векторов \mathbf{x}_j — например, матрица нормальная (для случая произвольных матриц ниже приведен численный пример). Тогда собственные векторы образуют базис, по которому можно разложить векторы \mathbf{x} и \mathbf{b} :

$$\mathbf{x} = \sum_{j=1}^n \xi_j \mathbf{x}_j, \quad \mathbf{b} = \sum_{j=1}^n \beta_j \mathbf{x}_j. \quad (18)$$

Подставляя это разложение в систему (17), перенося все члены влево и учитывая, что $A \mathbf{x}_j = \lambda_j \mathbf{x}_j$, получим

$$\sum_{j=1}^n [\xi_j (\lambda_j - \tilde{\lambda}_i) - \beta_j] \mathbf{x}_j = \mathbf{0}. \quad (19)$$

Поскольку собственные векторы линейно-независимы, то их линейная комбинация обращается в нуль только в том случае, когда

все ее коэффициенты равны нулю. Поэтому из (19) следует

$$\xi_j = \frac{\beta_j}{\lambda_j - \tilde{\lambda}_i}. \quad (20)$$

Видно, что если $\lambda_j \approx \tilde{\lambda}_i$, то коэффициент ξ_j будет очень большим; в противном случае он невелик. Рассмотрим следствия из этого в трех основных случаях.

Первый случай — собственное значение λ_i простое. Тогда из всех коэффициентов ξ_j , $1 \leq j \leq n$, только один коэффициент ξ_i оказывается очень большим. Это означает, что найденный вектор \mathbf{x} почти совпадает с собственным вектором \mathbf{x}_i (с точностью до нормировочного множителя), что и требовалось доказать. Заметим, что поскольку найденный вектор \mathbf{x} оказывается очень большим, то его обычно нормируют.

Из (20) видно, что при обратной итерации (т. е. при переходе от \mathbf{b} к \mathbf{x}) компонента β_i усиливается по сравнению с другими компонентами β_j примерно во столько раз, во сколько погрешность данного собственного значения меньше разности соседних собственных значений. Поэтому чем точнее найдено $\tilde{\lambda}_i$ (очевидно, хорошая точность особенно важна при наличии близких собственных значений), тем ближе \mathbf{x} будет к \mathbf{x}_i . Если собственные значения найдены слишком грубо, или случайно вектор \mathbf{b} выбран неудачно, так что β_i очень мало, то разница между \mathbf{x} и \mathbf{x}_i может оказаться заметной. Тогда подставляют найденный вектор \mathbf{x} в правую часть уравнения (17) вместо \mathbf{b} и организуют итерационный процесс

$$(A - \tilde{\lambda}_i E) \mathbf{x}^{(s)} = \mathbf{x}^{(s-1)}, \quad \mathbf{x}^{(0)} = \mathbf{b}. \quad (21)$$

Обычно он сходится настолько быстро, что двух итераций вполне достаточно. Напомним, что на каждой итерации обязательно надо нормировать найденные $\mathbf{x}^{(s)}$, чтобы не получать в расчетах слишком больших чисел, вызывающих переполнение на ЭВМ.

Замечание 1. Очень эффективен один простой способ выбора \mathbf{b} . В качестве его компонент в декартовых координатах возьмем последовательные многоразрядные псевдослучайные числа γ_k (см. § 4 главы IV). Тогда вероятность того, что β_i окажется очень малым, будет ничтожна.

Второй случай — собственное значение λ_i кратно; например, $\lambda_1 = \lambda_2 = \dots = \lambda_p$, $1 < p \leq n$. Напомним, что в этом случае собственные векторы $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p$ определены неоднозначно; любая их линейная комбинация удовлетворяет уравнению

$$A \left(\sum_{j=1}^p \alpha_j \mathbf{x}_j \right) = \sum_{j=1}^p \alpha_j A \mathbf{x}_j = \lambda_1 \sum_{j=1}^p \alpha_j \mathbf{x}_j$$

и является собственным вектором. Т. е. они порождают p -мерное

подпространство, любой базис которого можно взять в качестве системы искоемых собственных векторов.

Теперь из (20) следует, что большими оказываются коэффициенты $\xi_1, \xi_2, \dots, \xi_p$, причем степень их усиления одинакова; остальные коэффициенты остаются малыми. Значит, найденный из (17) вектор \mathbf{x} будет приближенно линейной комбинацией $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p$, а тем самым — искомым собственным вектором. Если точность полученного приближения недостаточна, то обратную итерацию повторяют снова по формуле (21).

Чтобы найти все собственные векторы для кратного собственного значения, возьмем столько линейно-независимых векторов $\mathbf{b}^{(k)}$, какова кратность корня. Обратными итерациями получим столько же векторов $\mathbf{x}^{(k)}$, которые и будут искомыми; они будут линейно-независимыми, поскольку преобразование (17) невырожденное. Остается только ортогонализировать найденные векторы, если это требуется по условиям задачи.

Напомним, что в качестве декартовых координат векторов $\mathbf{b}^{(k)}$ целесообразно брать псевдослучайные числа; здесь это имеет то дополнительное преимущество, что векторы автоматически получаются линейно-независимыми.

Третий случай — когда матрица имеет кратные корни, но число ее собственных векторов меньше n — выходит за рамки нашего доказательства. Однако метод обратных итераций здесь также применим в той форме, которая описана для кратных корней. Разница лишь в том, что если p -кратному собственному значению соответствуют всего q собственных векторов ($q < p$), то из полученных обратной итерацией векторов $\mathbf{x}^{(k)}$ только q будут линейно-независимыми. Это выясняется при их ортогонализации: первые q векторов ортогонализуются «без приключений», а при ортогонализации следующих векторов их компоненты обращаются почти в нуль (в пределах погрешности расчета).

Каков объем расчетов в методе обратной итерации? Нахождение собственного вектора требует (при одной итерации) не более $2/3n^3$ действий, так что для нахождения всех их надо около n^4 арифметических действий. Таким образом, при больших порядках матрицы метод неэкономичен, но при $n \leq 10$ вполне удовлетворителен. Особенно употребителен этот метод из-за своей простоты, универсальности и хорошей устойчивости алгоритма.

В некоторых частных случаях расчеты существенно упрощаются и ускоряются. Наиболее важен случай трехдиагональной матрицы. При этом линейная система уравнений (17) для определения компонент собственных векторов также будет трехдиагональной, и ее решают экономичным методом прогонки по несложным формулам (5.10) — (5.12). Для вычисления одного собственного вектора в этом случае требуется $10n$, а для всех — $10n^2$ арифметических действий.

Для почти треугольной матрицы в методе обратных итераций требуется решать линейную систему с почти треугольной матрицей, что делается специальным вариантом метода исключения. Если учесть, что случайный вектор в правой части (17) можно задавать уже после приведения матрицы в методе исключения Гаусса к треугольной форме, то нахождение каждого собственного вектора требует $3/2n^2$ действий (тот же прием для трехдиагональной матрицы позволяет сократить число действий до $7n$). А для нахождения всех собственных векторов требуется соответственно $3/2n^3$ арифметических действий.

Отметим одну существенную деталь. Поскольку $\det(A - \tilde{\lambda}_i E) \approx 0$, то при нахождении собственных векторов в формулах прямого хода метода исключений (прогонки) на главной диагонали появится хотя бы один очень малый элемент. Чтобы формально можно было вести расчёт, диагональные элементы не должны обращаться в нуль; для этого надо, чтобы погрешность собственного значения была не слишком мала, т. е. составляла бы 10—15 последних двоичных разрядов числа на ЭВМ. Если корни характеристического многочлена находят методом парабол (или секущих), то такая погрешность получается естественно, ибо из-за ошибок округления эти методы перестают сходиться в очень малой окрестности корня. Но если корни определялись методом Ньютона, то при этом могли быть найдены верно все знаки собственного значения; тогда, чтобы избежать деления на нуль, приходится специально вносить в λ_i небольшие погрешности.

Пример. Возьмем жорданову подматрицу C_4 четвертого порядка (3) и приближенное собственное значение $\tilde{\lambda} = a - \varepsilon$. В качестве b выберем вектор с единичными декартовыми координатами. Тогда уравнение (17) примет вид

$$\begin{bmatrix} \varepsilon & 1 & 0 & 0 \\ 0 & \varepsilon & 1 & 0 \\ 0 & 0 & \varepsilon & 1 \\ 0 & 0 & 0 & \varepsilon \end{bmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} \equiv \begin{pmatrix} \varepsilon x_1 + x_2 \\ \varepsilon x_2 + x_3 \\ \varepsilon x_3 + x_4 \\ \varepsilon x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}. \quad (22a)$$

Последовательно находим компоненты вектора x :

$$\begin{aligned} x_4 &= \varepsilon^{-1}, & x_3 &= -\varepsilon^{-2} + \varepsilon^{-1}, \\ x_2 &= \varepsilon^{-3} - \varepsilon^{-2} + \varepsilon^{-1}, \\ x_1 &= -\varepsilon^{-4} + \varepsilon^{-3} - \varepsilon^{-2} + \varepsilon^{-1}. \end{aligned} \quad (22б)$$

Затем нормируем вектор, умножив все компоненты на $-\varepsilon^{-4}$:

$$x_1 = 1 + O(\varepsilon), \quad x_2 = O(\varepsilon), \quad x_3 = O(\varepsilon^2), \quad x_4 = O(\varepsilon^3). \quad (22в)$$

Полученный вектор $x = e_1 + O(\varepsilon)$ приближенно равен собственному вектору жордановой матрицы (см. п. 1), что нам и требовалось. Попутно заметим, что в промежуточных выкладках (22б) возникали высокие обратные степени погрешности ε , чего не бывает у матриц с n собственными векторами. Это показывает, что случай матриц, содержащих жордановы подматрицы высокого порядка, труден для численных расчетов на ЭВМ, ибо в них легко возникают переполнения.

§ 2. Эрмитовы матрицы

1. Метод отражения. Существуют экономичные и устойчивые методы нахождения всех собственных значений матриц высокого порядка*). Они основаны на приведении матрицы преобразованием подобия к трехдиагональной или другим простым формам, для которых проблема собственных значений решается легко.

Сейчас мы рассмотрим *метод отражений*, который позволяет подобно преобразовать произвольную матрицу к почти треугольной форме за $(10/3)n^3$ арифметических действий, а эрмитову матрицу к трехдиагональной форме — всего за $4/3n^3$ действий. Поскольку для трехдиагональной матрицы все собственные значения находятся очень экономично, то для эрмитовых матриц метод отражения является самым быстрым из известных методов решения полной проблемы собственных значений. Рассмотрим его.

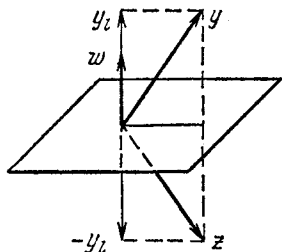


Рис. 31.

Произведем в n -мерном векторном пространстве отражение относительно некоторой гиперплоскости, проходящей через начало координат. Преобразование полностью определяется заданием нормали w к гиперплоскости. Эта нормаль есть нормированный вектор-столбец

$$(w, w) \equiv w^H w = \sum_{i=1}^n w_i^* w_i = 1, \quad (23)$$

где w^H есть вектор-строка, эрмитово сопряженный к столбцу. Возьмем произвольный вектор y и разложим его на две составляющие: параллельно нормали $y_l = w(w, y)$ и перпендикулярно ей. При отражении вектора его перпендикулярная составляющая остается неизменной, а параллельная — меняет знак (рис. 31), поэтому отраженный вектор z отличается от исходного на удвоенную величину параллельной компоненты

$$z = y - 2w(w, y). \quad (24)$$

Это преобразование вектора можно записать в канонической форме умножения на *матрицу отражения* R :

$$z = Ry, \quad R = E - 2ww^H, \quad (25)$$

*) Практически все описанные в §§ 2—3 методы хорошо применимы к матрицам, порядок которых не превышает сотни. Для матриц произвольного типа с $n > 100$ удовлетворительных методов решения общей проблемы собственных значений пока нет.

где умножение столбца w справа на строку той же длины w^H дает, по правилам умножения прямоугольных матриц, квадратную матрицу. Заметим, что равенства (24)—(25) в координатной форме записываются следующим образом:

$$z_i = y_i - 2w_i \sum_{j=1}^n w_j^* y_j, \quad (26)$$

$$R_{ij} = \delta_{ij} - 2w_i w_j^*.$$

Исследуем свойства матрицы отражения. Эта матрица эрмитова, что непосредственно вытекает из следующей цепочки преобразований:

$$R^H = (E - 2w w^H)^H = E - 2(w^H)^H w^H = E - 2w w^H = R. \quad (27)$$

Возведем матрицу отражения в квадрат:

$$R^2 = (E - 2w w^H)(E - 2w w^H) = E - 4w w^H + 4w w^H w w^H.$$

Преобразуем последний член правой части, используя ассоциативность умножения матриц и условие нормировки (23):

$$w w^H w w^H = w (w^H w) w^H = w w^H.$$

Тогда последний член сократится с предпоследним, и мы получим

$$R R = E, \text{ или } R = R^{-1}, \quad (28)$$

т. е. матрица отражения равна своей обратной. А сравнивая (27) и (28), убедимся, что $R^H = R^{-1}$, так что матрица отражений унитарна.

Последнее свойство для нас наиболее важно, поскольку для эрмитовых матриц наиболее выгодны унитарные преобразования подобия. В § 1 было показано, что они сохраняют эрмитовость матрицы. Поэтому если мы унитарным преобразованием подобия приведем матрицу к верхней почти треугольной форме, то в силу эрмитовости она будет трехдиагональной.

Заметим, что произведение любого числа унитарных матриц есть также унитарная матрица. В самом деле, если матрицы U, V, \dots, W унитарны, то

$$(UV\dots W)^{-1} = W^{-1}\dots V^{-1}U^{-1} = W^H\dots V^H U^H = (UV\dots W)^H.$$

Поэтому если мы применяем к эрмитовой матрице последовательность унитарных преобразований подобия, то она эквивалентна одному результирующему унитарному преобразованию подобия, и эрмитовость матрицы сохраняется.

Покажем, что для произвольной матрицы A можно подобрать такую конечную последовательность отражений, которая приводит матрицу к верхней почти треугольной форме. Для этого очередное отражение должно уничтожить самый длинный ненулевой столбец в нижней части матрицы A . Действие первых двух отражений показано на рис. 32, где жирными точками обозначены ненулевые элементы матрицы, а кружками — нулевые; третье отражение обращает в нуль обведенные элементы третьего столбца.

Будем считать, что уже уничтожен $q-1$ столбец, и разобьем матрицу A на клетки, как показано на рис. 32. Квадратная клетка A_1 есть верхняя почти треугольная матрица, а в прямоугольной клетке A_2 только последний столбец отличен от нуля.

Сделаем отражение при помощи вектора $\omega^q = \{\omega_i^q\}$, у которого первые q компонент нулевые:

$$\omega_i^q = 0 \text{ при } i \leq q$$

(далее верхний индекс вектора будем обычно опускать). Тогда видно, что если матрицу отражения разбить на клетки того же размера, что у матрицы A , то недиагональные клетки будут нулевыми:

$$\left[\begin{array}{c} \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \\ \bullet \bullet \bullet \bullet \bullet \end{array} \right] = \left[\begin{array}{c|c} A_1 & A_2 \\ \hline A_3 & A_4 \end{array} \right] \left. \begin{array}{l} \} q \\ \} n-q \end{array} \right\} \quad R = R^{-1} = \left[\begin{array}{c|c} E & 0 \\ \hline 0 & W \end{array} \right], \quad (29)$$

$$W_{ij} = \delta_{ij} - 2\omega_i \omega_j^*, \quad q+1 \leq i; j \leq n.$$

Рис. 32.

Из курса линейной алгебры известно, что если матрицы одинаковым образом разбиты на клетки, то они перемножаются по таким же формулам, как если бы эти клетки были обыкновенными элементами. Тогда искомое преобразование подобия принимает следующий вид:

$$B = R^{-1}AR = \left[\begin{array}{c|c} E & 0 \\ \hline 0 & W \end{array} \right] \cdot \left[\begin{array}{c|c} A_1 & A_2 \\ \hline A_3 & A_4 \end{array} \right] \cdot \left[\begin{array}{c|c} E & 0 \\ \hline 0 & W \end{array} \right] = \left[\begin{array}{c|c} A_1 & A_2 W \\ \hline W A_3 & W A_4 W \end{array} \right]. \quad (30)$$

Левая верхняя клетка результирующей матрицы B имеет нужную нам форму — верхнюю почти треугольную. Поскольку у клетки A_3 только последний столбец отличен от нуля, то у клетки $B_3 = W A_3$ элементы последнего столбца равны

$$b_{iq} = a_{iq} - \alpha \omega_i, \quad q+1 \leq i \leq n, \quad (31)$$

где

$$\alpha = 2 \sum_{j=q+1}^n \omega_j^* a_{jq}; \quad (32)$$

остальные же элементы этой клетки равны нулю.

Нам надо так подобрать вектор ω , чтобы обратились в нуль все элементы столбца (31), кроме верхнего. Очевидно, для этого надо положить

$$\omega_i = a_{iq}/\alpha \text{ при } q+2 \leq i \leq n \quad (33)$$

и найти, чему равна постоянная α . Заметим, что умножение вектора ω на множитель $e^{i\varphi}$ не меняет матрицы отражения; тогда из (32) следует, что α также определена только с точностью до этого множителя. Его всегда можно выбрать так, чтобы α была вещественной положительной, что будем предполагать выполненным.

Из (31) следует, что

$$\omega_{q+1} = (a_{q+1, q} - b_{q+1, q})/\alpha. \quad (34)$$

Подставляя (33)—(34) в условие нормировки (23), получим

$$\alpha^2 = \sum_{i=q+1}^n |a_{iq}|^2 + |b_{q+1, q}|^2 - (b_{q+1, q}^* a_{q+1, q} + b_{q+1, q} a_{q+1, q}^*). \quad (35)$$

Подстановка тех же выражений в формулу (32) дает

$$\alpha^2 = 2 \sum_{i=q+1}^n |a_{iq}|^2 - 2b_{q+1, q}^* a_{q+1, q}. \quad (36)$$

Последнее слагаемое в правой части этого равенства должно быть вещественным, поскольку остальные члены вещественны. Поэтому должно выполняться соотношение $\arg b_{q+1, q}^* = \pi k - \arg a_{q+1, q}$, где k — любое целое число. Для улучшения счетной устойчивости алгоритма выгодно полагать $k = \pm 1$: тогда последний член в правой части (36) будет положительным, и величина α никогда не станет близкой к нулю. Таким образом,

$$\arg b_{q+1, q} = \pi + \arg a_{q+1, q}. \quad (37)$$

Учитывая этот выбор аргумента и приравнявая друг другу правые части равенств (35) и (36), получим

$$|b_{q+1, q}| = \left(\sum_{i=q+1}^n |a_{iq}|^2 \right)^{1/2}. \quad (38)$$

Заменяя в (36) сумму при помощи равенства (38) и учитывая (37), упростим выражение для α :

$$\alpha = [2 |b_{q+1, q}| (|b_{q+1, q}| + |a_{q+1, q}|)]^{1/2}. \quad (39)$$

Формулы (37)—(39) и (33)—(34) полностью определяют матрицу очередного отражения. Эти формулы составлены так, что для вещественной матрицы A при вычислениях не возникает комплексных величин, а формула (37) для вычисления аргумента принимает при этом вид

$$\text{sign } b_{q+1, q} = - \text{sign } a_{q+1, q}.$$

Последовательно полагая $q = 1, 2, \dots, n-2$, определяя соответствующие векторы ω^q и производя отражения, мы приведем произвольную матрицу A к верхней почти треугольной форме. Если исходная матрица A была эрмитова, то результирующая матрица будет трехдиагональной.

Рассмотрим, как экономно организовать вычисления. Формулы для определения матрицы отражения не требуют большого объема

расчетов. Основное число действий уходит на перемножение матричных клеток в формуле (30). Заметим, что клетка A_1 не меняется, а в клетке $B_3 = WA_3$ имеется только один ненулевой элемент $b_{q+1, q}$, уже вычисленный при нахождении матрицы отражения; следовательно, эти клетки не нужно специально вычислять. При нахождении остальных двух клеток умножение на матрицу отражения W надо выполнять специальным образом; например, умножим A_4 на W справа, тогда

$$A_4 W = A_4 (E - 2\omega\omega^H) = A_4 - 2(A_4\omega)\omega^H. \quad (40)$$

Вместо того, чтобы перемножать две матрицы, мы пользуемся ассоциативностью умножения и сводим вычисление к двукратному умножению матрицы на вектор, что примерно в n раз быстрее. Умножение на W слева выполняется аналогично. Заметим, что если матрица A эрмитова, то возможна дополнительная экономия: тогда $B_2 = B_3^H$ и $B_4 = B_4^H$, так что клетку B_2 можно вообще не вычислять, а в клетке B_4 достаточно найти только нижнюю половину элементов.

Устойчивость численного алгоритма теоретически исследована недостаточно. Однако практика вычислений показала, что преобразования унитарными матрицами достаточно устойчивы. Поэтому основное, на что надо обращать внимание, — это чтобы ошибки округления не сказались бы на унитарности матриц отражения. Для контроля следует проверять выполнение условия нормировки (23); если оно соблюдается с очень высокой точностью (верны почти все двоичные разряды), то устойчивость обычно хорошая.

Когда матрица A приведена к трехдиагональной (или верхней почти треугольной) форме, то для этой формы собственные значения λ_i и собственные векторы y_i находятся легко. Найденные собственные значения одновременно являются собственными значениями исходной матрицы A . Для нахождения собственных векторов x_i исходной матрицы надо применить преобразование отражения

$$x_i = (R_1 R_2 \dots R_{n-2}) y_i. \quad (41)$$

Вычисления по этой формуле также надо делать экономично, выполняя каждое умножение на очередную матрицу R_q как два умножения на вектор по формуле (24).

Подсчет числа операций показывает, что для эрмитовых матриц метод отражения позволяет найти все собственные значения примерно за $((4/3)n^3 + 50n^2)$ арифметических действий, а все собственные векторы — еще за $(2n^3 + 10n^2)$ действий. Это самый быстрый из известных методов. Его скорость настолько велика, что позволяет на ЭВМ класса БЭСМ-6 вести расчет для матриц порядка $n \sim 100$; фактическую границу его применимости определяет устойчивость, которая при расчете с обычной точностью

Аналогично, матрица $C = U_{kl}^H B$ отличается от матрицы B только элементами k -й и l -й строк:

$$\begin{aligned} c_{kl} &= b_{kl}\alpha + b_{li}\beta^*, & c_{li} &= -b_{kl}\beta + b_{li}\alpha, & 1 \leq i \leq n, \\ c_{ji} &= b_{ji} & \text{при } j \neq k, l & \text{ и } 1 \leq i \leq n. \end{aligned} \quad (44)$$

Следовательно, матрица $C = U^H A U$ отличается от матрицы A лишь двумя строками и двумя столбцами. Формулы для вычисления элементов этих строк и столбцов написать нетрудно, но в этом нет необходимости; удобнее программировать на ЭВМ непосредственно формулы (43)—(44). Заметим, что если матрица A эрмитова, то матрица C также будет эрмитова; тогда в изменившихся столбцах и строках достаточно вычислить только половину элементов и тем самым вдвое уменьшить объем расчетов.

Найдем такую последовательность элементарных вращений, которая приводит произвольную (неэрмитову) матрицу A к верхней почти треугольной форме.

Можно так подобрать угол поворота в матрице U_{kl} , чтобы уничтожить элемент $c_{l, k-1}$, расположенный непосредственно перед

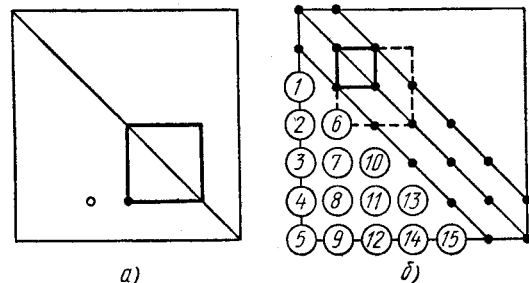


Рис. 33.

левым нижним углом подматрицы плоского поворота (рис. 33, а). Из формул (43) и (44) видно, что для этого надо положить

$$\alpha = \frac{|a_{k, k-1}|}{\sqrt{|a_{k, k-1}|^2 + |a_{l, k-1}|^2}}, \quad \beta = \frac{\alpha a_{l, k-1}}{a_{k, k-1}}. \quad (45a)$$

Сами углы вычислять нет необходимости, ибо в формулы для преобразования матричных элементов они не входят. Отметим, что для вещественных матриц величина β тоже будет вещественной; тогда формулы (45) удобнее записать следующим образом:

$$\alpha = \frac{a_{k, k-1}}{\sqrt{a_{k, k-1}^2 + a_{l, k-1}^2}}, \quad \beta = \frac{a_{l, k-1}}{\sqrt{a_{k, k-1}^2 + a_{l, k-1}^2}}. \quad (45b)$$

Теперь будем аннулировать те элементы матрицы и в том порядке, как это указано цифрами на рис. 33, б. Первый элемент уничтожается при помощи матрицы U_{23} , обозначенной на рисунке сплошным квадратом. Второй уничтожается вращением U_{24} , обозначенным пунктирным квадратом. При втором вращении в матрице A меняются элементы вторых и четвертых строк и столбцов. Значит, аннулированный элемент «1», лежащий в третьей строке, так и останется равным нулю.

Продолжая эти рассуждения, можно убедиться, что однажды уничтоженный элемент при такой последовательности исключения будет оставаться равным нулю. Поэтому после окончания всех исключений матрица станет верхней почти треугольной матрицей ($a_{ij} = 0$ при $i > j + 1$). Это справедливо для произвольной (неэрмитовой) матрицы.

Если исходная матрица A эрмитова, то благодаря сохранению эрмитовости при унитарном преобразовании подобия она приводится к трехдиагональной форме. В этом случае для экономии времени при каждом вращении достаточно

вычислять только изменившиеся элементы нижней половины матрицы (уже обратившиеся в нуль элементы в дальнейшие расчеты не включают).

Для полученной трехдиагональной (или почти треугольной) матрицы можно вычислять собственные значения и собственные векторы способами, изложенными в § 1. Найденные собственные значения будут одновременно собственными значениями исходной матрицы. А собственные векторы x_i исходной матрицы связаны с собственными векторами трехдиагональной матрицы соотношением

$$x_i = U_{23}U_{24} \dots U_{n-1} \cdot n y_i. \quad (46)$$

Проще всего вычислять их, последовательно умножая требуемый вектор y слева на матрицы вращения. Структура матриц такова, что при умножении на U_{kl} меняются только k -я и l -я компоненты вектора

$$\begin{aligned} x_k &= \alpha y_k - \beta^* y_l, & x_l &= \beta y_k + \alpha y_l, \\ x_j &= y_j & \text{при } j &\neq k, l. \end{aligned} \quad (47)$$

Предварительное перемножение самих матриц вращения потребовало бы большего числа действий (это особенно невыгодно, если нужна только часть собственных векторов).

На приведение эрмитовой матрицы к трехдиагональной форме и нахождение всех собственных значений в методе вращений требуется около $2n^3 + 50n^2$ арифметических действий и n^2 ячеек оперативной памяти. Для нахождения каждого собственного вектора надо затратить еще $3n^2$ действий. Собственные значения и собственные векторы в этом методе определяются устойчиво (если унитарность U_{kl} не нарушена ошибками округления).

3. Итерационный метод вращений. Несмотря на свою быстроту, описанные выше прямые методы не вполне удовлетворительны. Так, их алгоритм состоит из разнородных частей: преобразования исходной матрицы, вычисления корней многочлена, нахождения собственных векторов обратными итерациями. Кроме того, их формулы не упрощаются для некоторых употребительных специальных форм матриц (например, ленточных); тем самым они невыгодны для таких матриц. Поэтому разработан и используется ряд итерационных методов, в общем случае более медленных, но обладающих какими-то частными преимуществами.

Для эрмитовых матриц наиболее известен итерационный метод вращений, предложенный Якоби в 1846 г.; но в численных расчетах он начал использоваться только после появления работы [52]. Метод основан на подборе такой бесконечной последовательности элементарных вращений, которая в пределе преобразует эрмитову матрицу A в диагональную. При этом используются преобразования вращения с матрицами (42) такого же типа, как и для прямого метода вращений, но последовательность поворотов и их углы подбираются совершенно иным способом.

Рассмотрим, как действует элементарное вращение на сферическую норму матрицы (точнее, квадрат этой нормы):

$$S = \|A\|_E^2 = \sum_{i,j=1}^n |a_{ij}|^2. \quad (48)$$

Для определенности рассмотрим сначала умножение справа, $B = AU_{kl}$. Из формулы (43) видно, что при этом элементы k -го и l -го столбцов меняются так, что попарные суммы квадратов модулей сохраняются:

$$|b_{ik}|^2 + |b_{il}|^2 = |a_{ik}|^2 + |a_{il}|^2, \quad 1 \leq i \leq n;$$

элементы остальных столбцов остаются неизменными. Отсюда следует, что $\|AU\|_E = \|A\|_E$, т. е. сферическая норма матрицы A не меняется при умножении справа на матрицу вращения. Аналогичное утверждение легко доказать для умножения на матрицу U или U^H слева *). Описываемый метод основан на сохранении сферической нормы при вращениях.

Разобьем сумму, входящую в сферическую норму (48), на диагональную и недиагональную части:

$$S_1 = \sum_{i=1}^n |a_{ii}|^2, \quad S_2 = \sum_{\substack{i,j=1 \\ i \neq j}}^n |a_{ij}|^2. \quad (49)$$

При элементарном преобразовании вращения $U_{kl}^H A U_{kl}$ недиагональные элементы a_{ik} , a_{il} и a_{ki} , a_{li} при $i \neq k, l$ меняются так, что попарные суммы квадратов их модулей сохраняются; это легко видеть из формул (43)—(44). Кроме этих элементов вне диагонали есть еще один меняющийся элемент — это a_{kl} . Поэтому величина S_2 меняется при элементарном вращении настолько, насколько изменится $|a_{kl}|^2$. Будем подбирать вращения так, чтобы S_2 уменьшалась.

Чтобы максимально уменьшить S_2 за одно вращение, подберем угол поворота так, чтобы аннулировать элемент a_{kl} . Для простоты ограничимся вещественными эрмитовыми (т. е. симметричными) матрицами. Тогда $a_{kl} = a_{lk}$ — вещественные числа, и матрицы вращений U_{kl} тоже вещественны. Из формул (44) и (43) с учетом вещественности всех величин следует

$$c_{kl} = b_{kl}\alpha + b_{ll}\beta = a_{kl}(\alpha^2 - \beta^2) + (a_{ll} - a_{kk})\alpha\beta.$$

Полагая $c_{kl} = 0$ и вспоминая условие нормировки (42), получим систему уравнений для определения параметров поворота

$$\begin{aligned} \alpha^2 + \beta^2 &= 1, \\ a_{kl}(\alpha^2 - \beta^2) &= (a_{kk} - a_{ll})\alpha\beta. \end{aligned} \quad (50)$$

Возводя второе уравнение (50) в квадрат и исключая из него β^2 при помощи первого уравнения, получим биквадратное уравнение

*) Это является частным случаем общего утверждения, которое мы не доказываем: сферическая норма любой матрицы не меняется при умножении с любой стороны на унитарную матрицу.

для определения α :

$$\alpha^4 - \alpha^2 + a_{kl}^2 [4a_{kl}^2 + (a_{kk} - a_{ll})^2]^{-1} = 0.$$

Можно выбрать любой из четырех корней этого уравнения, тогда β определится однозначно. Для определенности положим

$$\alpha = \sqrt{\frac{1}{2} \left(1 + \frac{1}{\sqrt{1+\mu^2}} \right)}, \quad \text{где } \mu = \frac{2a_{kl}}{a_{kk} - a_{ll}}, \quad (51)$$

$$\beta = (\text{sign } \mu) \sqrt{\frac{1}{2} \left(1 - \frac{1}{\sqrt{1+\mu^2}} \right)}.$$

Сами углы поворота находить не требуется.

Итак, при каждом вращении S_2 уменьшается, а S_1 соответственно увеличивается, поскольку $S = S_1 + S_2$ сохраняется. Если подобрать такую последовательность вращений, чтобы $S_2 \rightarrow 0$, то все недиагональные элементы после достаточного числа поворотов станут пренебрежимо малыми и матрица A преобразуется в диагональную. *Диагональные элементы полученной диагональной матрицы и будут искомыми собственными значениями.* Но уничтожить все недиагональные элементы за конечное число поворотов нельзя, ибо, в отличие от прямого метода вращений, здесь при очередном повороте ранее уничтоженный элемент снова может стать ненулевым.

Какой именно недиагональный элемент целесообразно аннулировать при очередном повороте? Конечно, если уничтожать максимальный по модулю внедиагональный элемент, то скорость убывания S_2 будет наибольшей. При ручных расчетах это наилучший способ. Но на ЭВМ перебор элементов матрицы для определения максимального элемента требует неприемлемо большого числа действий. А если аннулировать элементы в заранее определенном порядке — циклом, то сходимость будет очень медленной. Наиболее выгодным оказалось уничтожение так называемого *оптимального элемента*.

Составим суммы квадратов модулей внедиагональных элементов строк:

$$r_i = \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|^2. \quad (52)$$

Выберем из этих сумм наибольшую, а в ней выберем наибольший по модулю элемент; его называют *оптимальным*. Поиск оптимального элемента сводится к перебору двух строк (строки сумм r_i , а в выбранной сумме — строки $|a_{ij}|$), т. е. требует малого числа действий. Суммы (52) вычисляются тоже экономично, ибо при каждом вращении из них меняются только две — r_k

и r_l , причем их можно вычислять по таким формулам:

$$\begin{aligned} r'_k &= r_k + a_{kk}^{\prime 2} - a_{kk}^{\circ} - a_{kl}^{\circ}, \\ r'_l &= r_l + r_k - r'_k; \end{aligned} \quad (53)$$

штрихи относятся к значениям после вращения.

Доказательство сходимости. Оптимальный элемент составляет не менее $1/(n-1)$ части суммы (52) своей строки, а эта сумма — не менее $1/n$ части S_2 . Следовательно, за одно вращение недиагональная часть сферической нормы уменьшается не менее чем на $\frac{2}{n(n-1)}$ долю своей величины (ибо уничтожаются два симметричных элемента). Значит, за N вращений S_2 убывает не медленнее, чем

$$\left[1 - \frac{2}{n(n-1)}\right]^N \approx \exp(-2N/n^2),$$

и тем самым стремится к нулю при $N \rightarrow \infty$. Следовательно, процесс Якоби с выбором оптимального (и тем более максимального) элемента всегда сходится.

Исследуем сходимость вблизи решения, считая собственные значения простыми. Пусть все внедиагональные элементы уже малы, $a_{ij} \sim \varepsilon$. Тогда из формул (51) следует, что угол поворота имеет тот же порядок малости: $\beta \sim \varepsilon$, $\alpha \approx 1 - O(\varepsilon^2)$. Подстановка в формулы (43)–(44) показывает, что при этом неуничтожаемые внедиагональные элементы меняются на $O(\varepsilon^2)$. Значит, за один цикл вращений*) все внедиагональные элементы станут $\sim \varepsilon^2$, что означает квадратичную сходимость.

Итак, вдали от решения сходимость не хуже линейной, а вблизи решения — квадратичная, т. е. быстрая. Это позволяет получать все собственные значения с высокой точностью. Обычно процесс сходится за 6–8 циклов вращений, или за $(3 \div 4)n^2$ элементарных вращений. Интересно, что чем больше кратных собственных значений, тем быстрее сходится метод.

Поскольку собственные векторы диагональной матрицы суть e_i , то собственными векторами матрицы A будут столбцы матрицы $U = \prod U_{kl}$. Заметим, что если внедиагональные элементы $a_{ij} = O(\varepsilon)$, то $S_2 = O(\varepsilon^2)$, так что диагональные элементы отличаются от собственных значений на $O(\varepsilon^2)$. Поэтому для нахождения собственных значений достаточно положить $\varepsilon \approx 10^{-6}$, чтобы получить правильно 10 знаков. Но чтобы получить с той же точностью

*) Циклом будем называть процесс с последовательным перебором всех поддиагональных элементов, или условно — $n(n-1)/2$ последовательных поворотов при другом способе выбора аннулируемых элементов.

собственные векторы, надо или вычислять их по формулам

$$x_i = \left(\prod U_{kl} \right) y_i, \quad y_i = \{y_{ij}\},$$

$$y_{ii} = 1 \text{ и } y_{ij} = \frac{a_{ji}}{a_{ii} - a_{jj}} \text{ при } j \neq i \quad (54)$$

или делать еще один цикл вращений.

Хотя теоретически в методе Якоби могут накапливаться ошибки, фактически устойчивость и точность очень высоки. Кратные собственные значения получаются столь же точно, как и простые, а собственные векторы практически ортогональны друг другу.

Метод Якоби с выбором оптимального элемента требует обычно около $30n^3$ арифметических действий и n^2 ячеек памяти для нахождения всех собственных значений. Для нахождения всех собственных векторов требуется еще около $20n^3$ действий. Таким образом, этот метод раз в 10 медленнее метода отражений. Основное его достоинство — надежность и единообразие вычислений, что позволяет легко запрограммировать метод. Итерационный метод вращений применяется там, где важна точность, надежность и простота расчета и менее существен объем вычислений.

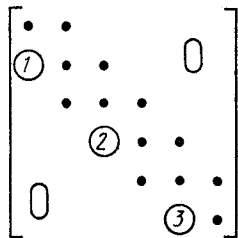


Рис. 34.

З а м е ч а н и е. Этот метод можно ускорить в 1,5—2 раза, не теряя его достоинств. У произвольных матриц недиагональная часть сферической нормы в среднем много больше диагональной, $S_2/S_1 \sim n$, а у трехдиагональных в среднем $S_2/S_1 \sim 2$. Значит, трехдиагональная матрица является выгодным начальным приближением для итерационного метода Якоби. Поэтому целесообразно предварительно привести исходную эрмитову матрицу к трехдиагональной форме при помощи прямого метода вращений и затем первым ходом метода Якоби аннулировать все нечетные или все четные поддиагональные элементы (рис. 34). После этого можно переходить на обычный вариант итерационного метода вращений с выбором оптимального элемента.

§ 3. Неэрмитовы матрицы

1. Метод элементарных преобразований. В принципе можно привести преобразованием подобия неэрмитову матрицу к почти треугольной форме при помощи отражений или вращений. Для матриц такой формы задача на собственные значения решается сравнительно быстро и устойчиво способом, описанным в § 1. Однако существует втрое более быстрый (хотя несколько менее устойчивый) метод элементарных преобразований; он позволяет привести произвольную матрицу к трехдиагональной форме всего за $2n^3$ арифметических действий. Для неэрмитовых матриц это самый быстрый из известных методов.

Метод является двухходовым. Первым ходом матрица приводится к верхней почти треугольной форме, а вторым — к трехдиагональной форме. Каждый ход состоит из последовательности элементарных преобразований подобия, напоминающих отражения; преобразования первого хода поочередно обращают в нуль столбцы в нижней части матрицы, а преобразования второго хода — строки в верхней части матрицы.

Первый ход. На его q -м шаге для преобразования подобия используется матрица следующего вида:

$$N = \left[\begin{array}{c|c} E_q & 0 \\ \hline 0 & N_q \end{array} \right] \cdot \dots \cdot \left. \begin{array}{c} q \\ n - q \end{array} \right\} \quad N_q \equiv N_q(v) = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ v_{q+2} & 1 & 0 & \dots & 0 \\ v_{q+3} & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ v_n & 0 & 0 & \dots & 1 \end{bmatrix}. \quad (55)$$

Исследуем свойства этой матрицы. Нетрудно проверить, что $N^H N \neq E$, так что эта матрица *не унитарна* (именно поэтому элементарные преобразования менее устойчивы по отношению к ошибкам округления). Изменим знаки всех компонент v_i , т. е. возьмем матрицу $N(-v)$; непосредственным перемножением легко убеждаемся, что $N(v)N(-v) = E$. Следовательно, обратная к (55) матрица определяется просто:

$$N^{-1}(v) = N(-v). \quad (56)$$

Аналогично методу отражений, матрицы (55) применяются для обращения в нуль нижней части q -го столбца, если уже аннулированы предыдущие столбцы (рис. 32). Разобьем матрицу A на клетки тех же размеров, что и в (55), и запишем преобразование подобия на данном шаге

$$\begin{aligned} B &= N^{-1}AN = \left[\begin{array}{c|c} E & 0 \\ \hline 0 & N_q(-v) \end{array} \right] \cdot \left[\begin{array}{c|c} A_1 & A_2 \\ \hline A_3 & A_4 \end{array} \right] \cdot \left[\begin{array}{c|c} E & 0 \\ \hline 0 & N_q(v) \end{array} \right] = \\ &= \left[\begin{array}{c|c} A_1 & A_2 N_q(v) \\ \hline N_q(-v) A_3 & N_q(-v) A_4 N_q(v) \end{array} \right] = \left[\begin{array}{c|c} B_1 & B_2 \\ \hline B_3 & B_4 \end{array} \right]. \quad (57) \end{aligned}$$

У клетки A_3 , а следовательно, и у клетки $B_3 = N_q(-v)A_3$ только последний столбец является ненулевым; элементы этого столбца результирующей матрицы получаются поэлементным перемножением клеток:

$$b_{q+1,q} = a_{q+1,q}, \quad b_{iq} = a_{iq} - v_i a_{q+1,q} \quad \text{при } q+2 \leq i \leq n. \quad (58)$$

Поэтому, чтобы обратить в нуль все элементы клетки B_3 , кроме углового элемента $b_{q+1,q}$, надо положить

$$v_i = \frac{a_{iq}}{a_{q+1,q}} \quad \text{при } q+2 \leq i \leq n. \quad (59)$$

Последняя формула определяет матрицу искомого элементарного

преобразования. Она существенно проще, чем формулы для нахождения нужной матрицы отражения в п. 2.

Само преобразование (57) очень несложно. Благодаря специальной структуре матрицы N умножение на нее выполняется так же быстро, как умножение на вектор. Например, поэлементно перемножая матрицы, найдем клетку $B_2 = A_2 N_q(v)$:

$$b_{ij} = a_{ij} \text{ при } q+2 \leq j \leq n, 1 \leq i \leq q, \\ b_{i, q+1} = a_{i, q+1} + \sum_{j=q+2}^n a_{ij} v_j \text{ при } 1 \leq i \leq q; \quad (60)$$

при умножении справа на N_q меняется только первый столбец клетки A_2 , а остальные столбцы клетки B_2 равны соответствующим столбцам клетки A_2 . Произведение $C_4 = A_4 N_q(v)$ вычисляется также по формулам (60), только с одним отличием: первый индекс элементов принимает значения $q+1 \leq i \leq n$. Умножение слева на N_q приводит к другим выражениям; так, для четвертой клетки $B_4 = N_q(-v) C_4$ поэлементное перемножение дает

$$b_{q+1, j} = c_{q+1, j} \text{ при } q+1 \leq j \leq n, \\ b_{ij} = c_{ij} - v_i c_{q+1, j} \text{ при } q+1 \leq j \leq n, q+2 \leq i \leq n, \quad (61)$$

т. е. меняются почти все элементы клетки. Формулы (58)–(61) полностью определяют очередной шаг первого хода. Они экономичны, так что метод элементарных преобразований позволяет привести произвольную матрицу к почти треугольной форме всего за $(5/3)n^3$ арифметических действий, т. е. вдвое быстрее, чем в методе отражений.

Но для эрмитовых матриц метод элементарных преобразований невыгоден, ибо при неунитарных преобразованиях эрмитовость не сохраняется. Тем самым результирующая матрица будет почти треугольной, а не трехдиагональной, как в методе отражений; вдобавок выигрыша в скорости по сравнению с методом отражений в этом случае нет.

Однако расчет по полученным формулам еще недостаточно устойчив. Если в ходе расчета на очередном шаге возникает малый ведущий элемент $a_{q+1, q}$, то согласно формулам (59) компоненты v_i будут велики. При вычислении остальных клеток матрицы элементы умножаются на эти компоненты, и погрешность сильно возрастает. Чтобы сделать метод устойчивым, выбирают *главный* (т. е. наибольший по модулю) элемент аннулируемого столбца

$$|a_{r_q}| = \max_i |a_{iq}| \text{ при } q+2 \leq r, i \leq n \quad (62)$$

и перестановкой $(q+1)$ -й и r -й строк делают его ведущим. Тогда будет выполняться неравенство $|v_i| \leq 1$, и погрешность практически не будет нарастать. Формально перестановку двух строк

преобразовании подобия (57) с матрицей N она останется нижней почти треугольной. В самом деле, клетка A_1 при этом преобразовании не меняется: В клетке A_2 ненулевым был только левый нижний элемент $a_{q, q+1}$; из формул (60) видно, что в клетке B_2 тоже только он будет отличен от нуля, причем $b_{q, q+1} = a_{q, q+1}$. Клетка B_4 также имеет нужную форму; в этом нетрудно убедиться, произведя поэлементное умножение.

Мысленно транспонируем все преобразования первого хода; согласно правилам матричной алгебры для этого надо транспонировать все матрицы, а во всех произведениях изменить порядок перемножения матриц на обратный. При транспонировании $N^{-1}(v)$ получим матрицу

$$M = \left[\begin{array}{c|c} E_q & 0 \\ \hline 0 & M_q \end{array} \right], \quad M_q = \left[\begin{array}{ccccc} 1 & -v_{q+2} & -v_{q+3} & \dots & -v_n \\ 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \hline 0 & 0 & 0 & \dots & 1 \end{array} \right], \quad (65)$$

причем $M^{-1}(v) = M(-v)$. Тогда предыдущий вывод принимает такую форму: если для верхней почти треугольной матрицы производится преобразование подобия при помощи матрицы (65), то результирующая матрица остается верхней почти треугольной.

Применим цепочку преобразований подобия $M^{-1}AM$ к матрице, полученной в результате первого хода. На каждом шаге элементарную матрицу $M(v)$ будем подбирать так, чтобы аннулировать элементы правой половины очередной строки (см. рис. 35, где точками обозначены ненулевые элементы, кружками — элементы, обращаемые в нуль на первом ходе, крестиками — обращаемые в нуль на начальных шагах второго хода; элементы, аннулируемые на очередном шаге, обведены). Благодаря такому выбору результирующая матрица должна стать нижней почти треугольной. Но в силу сказанного выше она одновременно остается верхней почти треугольной. Тем самым она будет трехдиагональной, что и требовалось.

Все формулы второго хода легко получить, применяя описанное выше транспонирование к формулам первого хода. Например, пусть аннулированы первые $q-1$ строк, и надо аннулировать q -ю строку. Тогда подбор элементов искомого преобразования производится аналогично формуле (59):

$$v_j = \frac{a_{q, j}}{a_{q, q+1}} \text{ при } q+2 \leq j \leq n, \quad (66)$$

и т. д. Поскольку на втором ходе в клетке A_3 имеется только

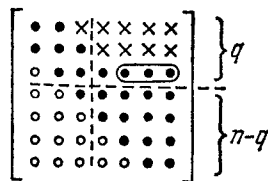


Рис. 35.

один ненулевой элемент, а в клетке A_4 около половины элементов — нулевые, то преобразование подобия почти треугольной матрицы к трехдиагональной форме требует всего $n^3/3$ арифметических действий, т. е. является очень быстрым.

Однако, если на некотором шаге второго хода ведущий элемент $a_{q, q+1}$ окажется очень малым, то расчет становится неустойчивым. А улучшать устойчивость перестановкой строк и столбцов здесь уже нельзя, ибо такая перестановка нарушает структуру матрицы (она перестает быть верхней почти треугольной). Поэтому, чтобы ослабить влияние ошибок округления, нередко производят расчет второго хода и вычисление характеристического многочлена трехдиагональной матрицы с двойной точностью. В отдельных случаях и это не помогает; тогда производят какую-либо перестановку столбцов исходной матрицы и такую же перестановку строк и повторяют расчет с самого начала. Правда, на практике срывы процесса довольно редки.

Всего двухходовой метод элементарных преобразований требует $2n^3$ действий для приведения матрицы к трехдиагональной форме, около $60n^2$ действий для нахождения всех собственных значений и собственных векторов трехдиагональной матрицы, и еще $2n^3$ действий для преобразования этих векторов в собственные векторы исходной матрицы. Это всего лишь в 7—8 раз больше, чем нужно для решения очень простой задачи — линейной системы того же порядка!

Таким образом, метод элементарных преобразований является очень быстрым и в большинстве случаев устойчивым.

2. Итерационные методы. Существует много методов, основанных на бесконечной последовательности преобразований подобия, приводящей матрицу к некоторым специальным формам, для которых полная проблема собственных значений легко решается. Итерационные методы сложнее прямых, а для матриц произвольного вида заметно уступают прямым методам по скорости (и зачастую — по устойчивости). Но поскольку известные прямые методы не совсем удовлетворительны, то пренебрегать итерационными методами не следует. Ниже даны краткие сведения о наиболее известных итерационных методах; подробное изложение их алгоритмов имеется, например, в монографии [5, 41].

Метод обобщенных вращений (развитый Эберлейн и В. В. Воеводиным в 1962—1965 гг.) основан на преобразовании матрицы к квазидиагональной форме, когда по главной диагонали расположены клетки, порядки которых равны кратности соответствующих собственных значений, а все остальные элементы матрицы равны нулю (разумеется, приближенно, ибо процесс итерационный). Если все собственные значения простые, то процесс сходится к диагональной матрице.

Для клеток, соответствующих кратным собственным значениям, надо находить собственные значения и собственные векторы специальным алгоритмом, т. е. в метод включаются дополнительные процедуры. Это неудобство имеется во всех итерационных методах. Но поскольку порядок таких клеток обычно невелик, то это не вызывает серьезных затруднений.

Шаг процесса состоит из двух частей. На первом полушаге делается элементарное преобразование матрицей типа N или M , в которой только одна из компонент v отлична от нуля; ее величина подбирается так, чтобы как

можно сильнее уменьшить $\|A\|_E$. Поскольку для любой матрицы $\|A\|_E^2 \geq \sum_{i=1}^n |\lambda_i|^2$, причем только для нормальных матриц имеет место равенство,

то такое преобразование приближает матрицу к нормальной. Второй полушаг — это вращение типа Якоби; для вещественной матрицы угол поворота φ определяется из условия

$$\operatorname{tg} 2\varphi = (a_{kl} + a_{lk}) / (a_{kk} - a_{ll}).$$

Процесс организован так, что полный шаг для эрмитовых матриц точно совпадает с циклическим вариантом метода Якоби. Значит, вычисления в общем случае требуют более $50n^3$ арифметических действий, т. е. метод довольно медленный. Зато он является одним из наиболее устойчивых.

Ортогональный степенной метод (предложенный В. В. Воеводиным в 1962 г.) основан на преобразовании матрицы к квазитреугольной форме, когда на главной диагонали стоят клетки, а ниже их все элементы равны нулю. У таких матриц собственные значения равны собственным значениям диагональных клеток, но собственные векторы определяются сложнее и значительно менее точно.

Ортогональный степенной метод устойчив и всегда сходится. Скорость сходимости линейная, со знаменателем типа $\mu = \max |\lambda_i / \lambda_{i+1}|$, где λ_i — собственные значения, расположенные в порядке возрастания модулей (причем кратные значения считаются за одно). Следовательно, требуемое число итераций довольно велико, особенно если среди собственных значений есть близкие. Одна итерация требует $(10/3)n^3$ арифметических действий, так что метод оказывается весьма медленным.

Треугольный степенной метод (предложен Бауэром в 1957 г.) также основан на преобразовании матрицы к квазитреугольной форме. Сходимость его тоже линейная, но одна итерация требует всего $(5/3)n^3$ действий, а при небольшом усложнении алгоритма — даже $(2/3)n^3$ действий. Зато этот метод менее устойчив, чем ортогональный степенной метод, особенно если собственные значения комплексные, или в расчетах появляются матрицы с близкими к нулю главными минорами. Зачастую для сохранения устойчивости приходится видоизменять алгоритм.

LR-алгоритм (предложен Рутисхаузером и Бауэром в 1955 г.) тоже содержит преобразование матрицы к квазитреугольной форме. Он разработан только для вещественных матриц с вещественными собственными значениями. Метод всегда сходится, причем вблизи решения квадратично; одна итерация требует $(7/3)n^3$ действий. Таким образом, по скорости этот метод превосходит ортогональный степенной; зато он уступает ему по устойчивости.

QR-алгоритм (предложен В. Н. Кублановской и Френсисом в 1961 г.) основан на преобразовании матрицы к квазитреугольной форме. По устойчивости и характеру сходимости он аналогичен ортогональному степенному методу. Этот метод очень выгоден для верхних почти треугольных матриц: в ходе преобразований их структура не разрушается, и благодаря этому одна итерация требует всего $6n^2$ арифметических действий (т. е. время расчета уменьшается в $n/2$ раз по сравнению с общим случаем). Детали этого алгоритма хорошо отработаны, и существуют основанные на нем стандартные программы.

LR-алгоритм (предложен Рутисхаузером в 1955 г.) рассчитан только на вещественные матрицы с вещественными собственными значениями. Он близок к треугольному степенному методу, не очень устойчив и сходится медленно (построены даже примеры заикливания процесса). Зато для почти треугольных матриц он требует всего $2n^2$ действий на одну итерацию, а для ленточных матриц дает еще большую экономию.

3. Некоторые частные случаи. *Косоэрмитова* матрица A умножением на i превращается в эрмитову матрицу $B = iA$; для эрмитовой же матрицы проблема собственных значений решается

гораздо легче, чем для неэрмитовых. Для комплексных матриц этот способ наиболее удобен. Для вещественных косоэрмитовых (кососимметричных) матриц он несколько менее выгоден, ибо после умножения на i приходится все остальные действия выполнять с комплексными числами.

Обобщенная проблема собственных значений

$$Ax = \lambda Bx \quad (67)$$

особенно легко решается, если матрицы A и B эрмитовы и одна из них — положительно определенная. Будем считать, что положительно определена вторая матрица (если положительно определена матрица A , то задачу (67) надо переписать в виде $Bx = \lambda^{-1}Ax$).

Разложим матрицу B методом квадратного корня (см. главу V, § 1) в произведение двух треугольных $B = S^H S$; благодаря положительной определенности матрицы B в этом разложении отсутствует диагональная матрица. Тогда можно переписать исходную задачу (67) в следующем виде:

$$Cy = \lambda y, \quad \text{где } y = Sx, \quad C = (S^H)^{-1} A S^{-1}. \quad (68)$$

Вычисление треугольной матрицы S , ее обращение и нахождение матрицы C выполняются за $4n^3$ действий. Легко проверить, что C — эрмитова матрица; таким образом, задача свелась к хорошо изученной.

Замечание. Запись задачи (67) в виде $(B^{-1}A)x = \lambda x$ невыгодна, ибо матрица $B^{-1}A$ не будет, вообще говоря, эрмитовой.

Нормальную матрицу A по теореме Шура можно привести к диагональной форме унитарным преобразованием подобия. Например, если уничтожены все элементы нижней половины, то в силу нормальности матрицы полученная верхняя треугольная матрица будет диагональной.

Чтобы реализовать эту идею, были предложены разные варианты итерационного метода вращений: циклическое аннулирование поддиагональных элементов, или уменьшение поддиагональной части сферической нормы. Однако они оказались неудачными: были построены примеры, в которых эти процессы сходились к недиагональным матрицам.

Удовлетворительным оказался довольно искусственный вариант. Рассмотрим преобразование $V_{kl}^H A U_{kl}$, производимое унитарными матрицами вращения (оно не является преобразованием подобия). Если углы поворота справа и слева разные, то их можно подобрать так, что на каждом повороте недиагональная часть сферической нормы уменьшается, а бесконечная цепочка преобразований приводит матрицу к диагональной форме: $V^H A U \rightarrow D$, где $V = \prod V_{kl}$, $U = \prod U_{kl}$.

Теперь рассмотрим преобразование подобия $B = U^H A U$, выполненное найденной матрицей поворота. Можно доказать, что в матрице B равны нулю (разумеется, приближенно) все недиагональные элементы, кроме тех, которые лежат на пересечении строк и столбцов, для которых диагональные элементы вспомогательной матрицы D равны между собой по модулю: $b_{ik} = 0$, если $|d_{ii}| \neq |d_{kk}|$.

Сделаем такую перестановку столбцов и строк с одинаковыми номерами (а это — преобразование подобия), чтобы в матрице D равные по модулю диагональные элементы заняли соседние места вдоль диагонали. Такая перестановка приводит матрицу B к квазидиагональной форме, после чего задача на собственные значения легко решается (если размеры клеток невелики).

§ 4. Частичная проблема собственных значений

1. Особенности проблемы. Во многих задачах интересны не все собственные значения, а только небольшая их часть.

Матрицы очень высоких порядков обычно получаются при конечно-разностном решении задач на собственные значения для дифференциальных уравнений. В этом случае достаточно вычислить только несколько низших собственных значений, соответствующих малому числу нулей собственной функции (а высокие собственные значения матрицы все равно плохо аппроксимируют соответствующие собственные значения дифференциального оператора в силу свойств конечно-разностных методов). В задачах диффузии нейтронов имеют физический смысл только одно или два собственных значения, и т. д.

В этих случаях решать полную проблему собственных значений невыгодно. Обычно применяют итерационные процессы, сходящиеся к одному собственному значению и собственному вектору. Большинство этих процессов для особенно важных на практике ленточных матриц записывается очень экономно.

Для описанных ниже процессов нужно задавать нулевое приближение. Если оно удачно выбрано, то число итераций заметно уменьшается. Зачастую хорошее нулевое приближение можно получить из физических соображений.

2. Метод линеаризации. Запишем задачу на собственные значения (1) через компоненты собственного вектора:

$$F_i(\mathbf{x}, \lambda) \equiv \sum_{k=1}^n a_{ik}x_k - \lambda x_i = 0, \quad 1 \leq i \leq n. \quad (69)$$

Задачу (69) можно рассматривать как систему уравнений с $n+1$ неизвестным $x_1, x_2, \dots, x_n, \lambda$; эта система нелинейна благодаря наличию членов λx_i . Для решения нелинейной системы целесообразно применить метод Ньютона. Давая всем переменным малые приращения $\delta x_i, \delta \lambda$ и линеаризуя уравнения (69) относительно приращений, получим

$$\sum_{k=1}^n a_{ik} \delta x_k - \lambda^{(s)} \delta x_i - x_i^{(s)} \delta \lambda = -F_i(\mathbf{x}^{(s)}, \lambda^{(s)}), \quad 1 \leq i \leq n; \quad (70)$$

здесь индекс s обозначает номер итерации. Система (70) содержит n

уравнений, линейных относительно неизвестных приращений. Этих неизвестных $n+1$; но поскольку собственный вектор определен с точностью до множителя, то, не нарушая общности, можно положить или $\delta x_1 = 0$, или $\delta x_n = 0$. После этого число уравнений будет равно числу неизвестных приращений.

Напомним, что ньютоновский процесс вблизи решения сходится квадратично. Число итераций зависит от выбора нулевого приближения. При удачном приближении достаточно 3—5 итераций. А если за 10 итераций процесс не сошелся, то скорее всего он не сойдется, и надо изменить нулевое приближение. Заметим, что при неудачном нулевом приближении процесс иногда сходится не к искомому собственному значению, а к другому.

Матрица линейной системы (70) отличается от матрицы A по структуре мало — только добавлением одного ненулевого столбца. Поэтому для матрицы A общего вида на одну итерацию требуется $2/3n^3$ арифметических действий, для почти треугольной матрицы — всего $2n^2$ действий, а для ленточной матрицы даже $m^2n/2$ действий (где m — ширина ленты).

Метод линеаризации успешно применяется к матрицам порядка $n \approx 100 - 1000$. Он особенно выгоден для трехдиагональных матриц; для них получение одного собственного значения и собственного вектора требует обычно $\sim 50n$ арифметических действий.

3. Степенной метод (счет на установление) применяется для получения наибольшего по модулю собственного значения. Пусть $|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots$. Построим такой итерационный процесс:

$$\mathbf{x}^{(s+1)} = A\mathbf{x}^{(s)}. \quad (71)$$

Он не сходится в обычном смысле. Разложим нулевое приближение по собственным векторам матрицы: $\mathbf{x}^{(0)} = \sum_i \xi_i \mathbf{x}_i$. Тогда

легко убедиться, что $\mathbf{x}^{(s)} = \sum_i \lambda_i^s \xi_i \mathbf{x}_i$ и при достаточно большом

числе итераций $\mathbf{x}^{(s)} \approx \lambda_1^s \xi_1 \mathbf{x}_1$, т. е. вектор $\mathbf{x}^{(s)}$ сходится к собственному вектору по направлению. Очевидно, при этом $\mathbf{x}^{(s+1)} \approx \lambda_1 \mathbf{x}^{(s)}$.

Процесс сходится линейно со знаменателем $q \approx |\lambda_2/\lambda_1|$. Считается, что процесс практически сошелся, если отношения соответствующих координат векторов $\mathbf{x}^{(s+1)}$ и $\mathbf{x}^{(s)}$ с требуемой точностью одинаковы и не меняются на последних итерациях. При этом для более точного получения собственного значения целесообразно положить

$$|\lambda_1| \approx \frac{|\mathbf{x}^{(s+1)}|}{|\mathbf{x}^{(s)}|} = \sqrt{\frac{(\mathbf{x}^{(s+1)}, \mathbf{x}^{(s+1)})}{(\mathbf{x}^{(s)}, \mathbf{x}^{(s)})}}. \quad (72)$$

Отметим, что при расчетах на ЭВМ на каждой итерации после вычисления λ_1 вектор $\mathbf{x}^{(s+1)}$ надо нормировать, чтобы не получать переполнений или исчезновений чисел.

Формально при $\xi_1 = 0$ итерации сходятся к следующему собственному значению. Однако из-за ошибок округления ξ_1 не может быть точно нулем, а при малом ξ_1 процесс по-прежнему сходится к первому собственному значению, только за большее число итераций.

Если наибольшее собственное значение кратное, но соответствующий элементарный делитель матрицы линеен, то итерации сходятся обычным образом. Но если $\lambda_1 \neq \lambda_2$, а их модули равны или если элементарный делитель матрицы нелинеен (жорданова клетка), то процесс не сходится.

Если $|\lambda_1| \approx |\lambda_2|$, то сходимость очень медленная; этот случай нередко встречается в простейших итерационных методах решения разностных схем для эллиптических уравнений (глава XII). Тогда сходимость можно ускорить процессом Эйткена (см. главу IV, § 1).

Одна итерация для матрицы общего вида требует $2n^2$ арифметических действий, а для ленточной матрицы — $2mn$ действий. Из-за медленной сходимости степенной метод применяют только к матрицам, содержащим очень много нулевых элементов (и даже к ним — довольно редко).

В математической литературе описана вариация степенного метода, имеющая квадратичную сходимость: $\mathbf{x}^{(s)} = A_s \mathbf{x}^{(0)}$, где $A_s = A_{s-1} A_{s-1}$ и $A_0 = A$. Однако если матрица A имеет много нулевых элементов, то ее степени уже такими не будут. Поэтому этот вариант обычно не экономичен.

4. Обратные итерации со сдвигом. Напишем итерационный процесс, обратный по отношению к степенному процессу:

$$\mathbf{x}^{(s+1)} = A^{-1} \mathbf{x}^{(s)}. \quad (73)$$

Очевидно, он сходится в указанном в п. 3 смысле к наибольшему по модулю собственному значению матрицы A^{-1} , т. е. к наименьшему по модулю собственному значению матрицы A (ибо собственные значения матриц A и A^{-1} обратны друг другу). Все, что говорилось в предыдущем пункте о характере сходимости, разумеется, справедливо и в этом случае; сходимость будет довольно медленной.

Однако здесь положение можно существенно улучшить *методом сдвига*, который заключается в следующем. Пусть нам приближенно известно некоторое, не обязательно наименьшее, собственное значение $\tilde{\lambda}_i$. Тогда так называемая *сдвинутая матрица* $(A - \tilde{\lambda}_i E)$ будет иметь собственные значения $\lambda - \tilde{\lambda}_i$. У этой матрицы интересующее нас собственное значение $\lambda_i - \tilde{\lambda}_i$ будет намного меньше по модулю, чем остальные. Поэтому обратные итерации со сдвинутой матрицей (которые мы запишем в несколько иной форме)

$$(A - \tilde{\lambda}_i E) \mathbf{x}^{(s+1)} = \mathbf{x}^{(s)}, \quad (74a)$$

будут быстро сходиться и определяют требуемое нам собственное значение $\lambda_i - \tilde{\lambda}_i$. Напомним, что после каждой итерации надо нормировать вектор, чтобы избежать переполнений. С учетом этого вместо (74a) получим последовательность формул

$$(A - \tilde{\lambda}_i E) \mathbf{y}^{(s)} = \mathbf{x}^{(s)},$$

$$\lambda_i^{(s)} - \tilde{\lambda}_i = \left\langle \frac{x_k^{(s)}}{y_k^{(s)}} \right\rangle, \quad \mathbf{x}^{(s+1)} = \frac{\mathbf{y}^{(s)}}{\|\mathbf{y}^{(s)}\|}. \quad (74б)$$

Здесь индекс k относится к компонентам векторов, а скобки $\langle \dots \rangle$ означают некоторое усреднение по всем компонентам: например, среднеарифметическое.

Если исходное приближение было хорошим, то иногда процесс сходится за несколько итераций; тогда выгодно непосредственно решать линейную систему (73). Если же требуемое число итераций велико, то лучше обратить матрицу $(A - \tilde{\lambda}_i E)$. Выгодней всего при решении линейной системы (74) методом исключения Гаусса использовать полученные на первой же итерации вспомогательные коэффициенты c_{mk} (см. главу V, § 1, п. 1) на каждой последующей итерации; но это не предусмотрено в обычных стандартных программах.

Если сдвиг постоянный, то итерации сходятся линейно. Можно получить квадратичную сходимость, если уточнять сдвиг в ходе расчета следующим образом:

$$(A - \lambda_i^{(s)} E) \mathbf{y}^{(s)} = \mathbf{x}^{(s)},$$

$$\lambda_i^{(s+1)} = \lambda_i^{(s)} + \left\langle \frac{x_k^{(s)}}{y_k^{(s)}} \right\rangle, \quad \mathbf{x}^{(s+1)} = \frac{\mathbf{y}^{(s)}}{\|\mathbf{y}^{(s)}\|}. \quad (75)$$

Для матриц, имеющих ортогональную систему собственных векторов (например, эрмитовых матриц), сходимость вблизи корня будет даже кубической. Заметим, что допускать слишком точное совпадение $\tilde{\lambda}_i$ с собственным значением нельзя, ибо матрица системы (75) становится плохо обусловленной; об этом уже говорилось в § 1, п. 6 в связи с нахождением собственных векторов. Поэтому, когда в ходе итераций у величины $\lambda_i^{(s)}$ устанавливаются (т. е. перестают меняться) 5—7 знаков, то итерации следует прекращать.

З а м е ч а н и е 1. Переменный сдвиг собственного значения (75) нельзя включать с первой итераций; сначала надо получить грубую сходимость итераций с постоянным сдвигом.

З а м е ч а н и е 2. Обратные итерации особенно удобны, если матрица заранее приведена преобразованием подобия к почти треугольной форме. Тогда одна обратная итерация выполняется

методом исключения с выбором главного элемента всего за $2n^2$ действий. Теоретически для ленточных матриц возможна еще большая экономия, но преобразование подобия почти треугольной матрицы к трехдиагональной форме не всегда устойчиво.

Выводы. Обратные итерации с постоянным и особенно с переменным сдвигом — очень эффективный метод расчета. Для нахождения собственных векторов этот метод считается наиболее точным. Сходимость при хорошем подборе $\tilde{\lambda}$ настолько быстрая, что метод пригоден и для близких или случайно равных по модулю собственных значений (ибо после сдвига они хорошо различаются), и даже при наличии у матрицы нелинейного элементарного делителя.

ЗАДАЧИ

1. Доказать, что если матрица n -го порядка имеет n собственных векторов $e_i = \{\delta_{in}\}$, $1 \leq k \leq n$, то она диагональна.
2. Найти собственные векторы треугольной матрицы, считая все собственные значения простыми.
3. Доказать, что нормальная матрица при унитарном преобразовании подобия остается нормальной.
4. Показать, что если матрица A ленточная, то преобразование подобия матрицами отражения (30) не сохраняет ее структуры.
5. Какие элементы необходимо вычислять в формулах (43) — (44) при преобразовании подобия матрицами вращения для эрмитовой матрицы A ?
6. Доказать, что сферическая норма произвольной матрицы не меняется при умножении с любой стороны на унитарную матрицу.
7. В итерационном методе вращений вывести для определения параметров поворота комплексных матриц формулу, аналогичную (51).
8. Показать, что в итерационном методе вращений формулы (54) определяют собственные векторы с точностью $O(\epsilon^2)$, где ϵ — максимум модулей внедиагональных элементов; если же в этих формулах положить $y_i = e_i$, то точность ухудшается до $O(\epsilon)$.
9. Получить все формулы расчета матричных элементов для второго хода метода элементарных преобразований.
10. Написать формулы восстановления собственных векторов исходной матрицы по собственным векторам трехдиагональной матрицы в методе элементарных преобразований.
11. Показать, что если матрица A ленточная, то элементарное преобразование подобия (57) разрушает ее структуру.
12. а) Какой вид примут формулы метода линеаризации (70), если недостающее уравнение получать из условия нормировки собственного вектора $\sum_{i=1}^n |x_i|^2 = 1$? б) Как построить экономичный алгоритм решения полученной при этом линейной системы, если матрица A является трехдиагональной?
13. Доказать, что метод обратных итераций с переменным сдвигом (75) сходится квадратично вблизи простого собственного значения.