Курс «Вычислительные алгоритмы теории автоматического управления».

Лекция 2. Методы анализа детерминированных линейных стационарных систем. Основы матричных вычислительных алгоритмов.

Уравнения состояния линейных стационарных систем и типовые вычислительные задачи. Вычислительные аспекты умножения матриц. Решение линейных уравнений. Положительно определенные матрицы. Разложение Холесского. Исключение Гаусса и LU разложение. Вычисление обратной матрицы. Нормы векторов и матриц. Число обусловленности матрицы. Сингулярное разложение. Задача наименьших квадратов и SVD метод. Псевдообращение матриц. Вычисление функций от матриц. Матричная экспонента.

Уравнения состояния детерминированных линейных стационарных систем и типовые вычислительные задачи.

Рассмотрим автономную линейную систему следующего вида:

$$\dot{x} = Ax + Bu; y = Cx + Du; x \in \Re^n, u \in \Re^m, y \in \Re^p,$$

где A, B, C, D - действительные матрицы размерностей $(n \times n), (n \times m), (p \times n), (p \times m)$ соответственно.

Обычно предполагается, что начало координат $x0 = 0$ является положением равновесия для свободной, или неуправляемой системы, для которой $u(t) \equiv 0$. При этом считается, что более общая система с ненулевым положением равновесия $(\overline{x}, \overline{u})$ во многих случаях может быть приведена к такому же виду с помощью параллельного переноса осей координат. Для такого приведения необходимо уметь решать линейное уравнение, которое определяет точку $\overline{x} \neq 0$ при значении $\overline{u} \neq 0$, то есть уравнение вида $Ax = b$.

Анализ динамики линейной стационарной системы начинается с анализа ее устойчивости. Анализ устойчивости можно осуществлять с помощью прямого метода вычисления собственных значений матрицы A или с помощью матричного уравнения Ляпунова.

После анализа устойчивости обычно необходимо вычислить матричную передаточную функцию объекта вида:

$$W(s) = C \cdot (s \cdot E - A)^{-1} \cdot B + D,$$

где s - комплексная переменная преобразования Лапласа, $E \in \Re^{(n \times n)}$ - единичная диагональная матрица.

Частотный анализ передаточной функции $W(s)$ сводится к исследованию амплитудно – частотной характеристики $W(i \cdot \omega)$ при изменении частоты ω в интервале $(-\infty, \infty)$.

Для исследования BIBO устойчивости объекта необходимо исследовать вырожденность передаточной функции $W(s)$.

Любой синтез системы начинается с оценки ее управляемости и наблюдаемости, то есть вычисления рангов матриц $F = (B, AB, A^2B, ..., A^{n-1}B)$ и $H = (C, A^TC, ..., (A^T)^{n-1}C)$. Если ранги матриц управляемости и наблюдаемости не являются полными, то в этом случае необходимо провести декомпозицию Калмана уравнений объекта с целью выделения управляемой и наблюдаемой частей объекта. То есть представить уравнения объекта с помощью невырожденного линейного преобразования в виде:

$$\begin{pmatrix} \dot{\tilde{x}}_{c0} \\ \dot{\tilde{x}}_{cuo} \\ \dot{\tilde{x}}_{uco} \\ \dot{\tilde{x}}_{ucuo} \end{pmatrix} = \begin{pmatrix} \tilde{A}_{co} & 0 & \tilde{A}_{13} & 0 \\ \tilde{A}_{21} & \tilde{A}_{cuo} & \tilde{A}_{23} & \tilde{A}_{24} \\ 0 & 0 & \tilde{A}_{uco} & 0 \\ 0 & 0 & \tilde{A}_{43} & \tilde{A}_{ucuo} \end{pmatrix} \begin{pmatrix} \tilde{x}_{c0} \\ \tilde{x}_{cuo} \\ \tilde{x}_{uco} \\ \tilde{x}_{ucuo} \end{pmatrix} + \begin{pmatrix} \tilde{B}_{co} \\ \tilde{B}_{cuo} \\ 0 \\ 0 \end{pmatrix} u,\ y = \begin{pmatrix} \tilde{C}_{co} & 0 & \tilde{C}_{uco} & 0 \end{pmatrix} \cdot \tilde{x} + Du,$$

где вектор $\tilde{x}_{co}$ является управляемым и наблюдаемым; вектор $\tilde{x}_{cuo}$ является управляемым, но не наблюдаемым; вектор $\tilde{x}_{uco}$ является наблюдаемым, но не управляемым, и вектор $\tilde{x}_{ucuo}$ является не управляемым и не наблюдаемым. Более того, исходная модель состояния эквивалентна, при нулевых начальных условиях, управляемой и наблюдаемой ее редуцированной части: $\dot{\tilde{x}}_{co} = \tilde{A}_{co}\tilde{x}_{co} + \tilde{B}_{co}u,\ \tilde{y} = \tilde{C}_{co}\tilde{x}_{co} + Du$, и имеет передаточную функцию равную: $\tilde{W}(s) = \tilde{C}_{co}(sE - \tilde{A}_{co})^{-1}\tilde{B}_{co} + D$.

Если управляемая часть объекта управления является стабилизируемой, то можно переходить к решению задачи синтеза системы управления.

Задача синтеза линейной обратной связи по состоянию заключается в поиске такой матрицы K коэффициентов обратной связи $u = -K \cdot x$, при которой будет выполняться следующее соотношение:

$$\det(s \cdot E - A + B \cdot K) = \varphi(s),$$

где $\varphi(s)$ - желаемый полином замкнутой системы. Решение данной задачи для SISO систем с помощью формулы Аккермана требует обращения матрицы управляемости F, а для MIMO систем процедуры нахождения соответствующей псевдообратной матрицы, которая обычно реализуется с помощью различных итеративных алгоритмов.

При этом, для определения управления вида $u = -K \cdot x$, обеспечивающего необходимую динамику замкнутой системы, обычно требуется вычисление обратной матрицы вида $(F \cdot F^T)^{-1}$ при $m > 1$ и, F^{-1} при $m = 1$, где F - соответствующая матрица управляемости. Аналогичные задачи возникают и при синтезе наблюдателя (идентификатора). Таким образом, численное решение задач анализа и синтеза линейных стационарных систем в основном базируется на алгоритмах вычислительной матричной алгебры.

Вычислительные аспекты умножения матриц.

Основные примитивные процедуры с матрицами, как вычислительными структурами данными, были рассмотрены ранее в курсе ОТАУ. Напомним их в кратком виде.

- транспонирование $\Re^{n\times m} \to \Re^{n\times m}; C = A^T \Rightarrow c_{ij} = a_{ji}$;

- сложение $\Re^{n\times m}, \Re^{n\times m} \to \Re^{n\times m}; C = A + B \Rightarrow c_{ij} = a_{ij} + b_{ij}$;

- умножение матрицы на число $\Re, \Re^{n\times m} \to \Re^{n\times m}; C = \alpha A \Rightarrow c_{ij} = \alpha a_{ij}$;

- умножение матрицы на матрицу $\Re^{n\times r} \otimes \Re^{r\times m} \to \Re^{n\times m}; C = AB \Rightarrow c_{ij} = \sum_{k=1}^{r} a_{ik}b_{kj}; i = 1{:}n;\ j = 1{:}m$.

Рассмотрим построение алгоритма умножения матрицы на вектор на основе базовых операций. Пусть $A \in \Re^{n\times m}$. Нужно найти $z = Ax$. Стандартный способ вычисления результата задается следующей формулой $z_i = \sum_{j=1}^{m} a_{ij}x_j$.

Алгоритм. Версия с доступом по строкам.

$$
\begin{array}{l}
function\ z = matvec(A,x) \\
\quad n = rows(A);\ m = cols(A); \\
\quad z(1:n,1) = 0; \\
\quad for\ i = 1:n \\
\quad\quad for\ j = 1:m \\
\quad\quad\quad z(i) = z(i) + A(i,j)\cdot x(j); \\
\quad\quad end\ \%\ for\ j \\
\quad end\ \%\ for\ i \\
end\ \%matvec
\end{array}
$$

Аналогичную версию алгоритма можно составить с учетом доступа по столбцам.

Минимизации записи алгоритма с использование блочных разбиений матрицы.

Анализ внутренних циклов показывает, что доступ к элементам матрицы A можно осуществлять как по строкам, так и по столбцам. Чтобы яснее охарактеризовать такой способ доступа будем использовать язык блочных разбиений матриц. С точки зрения строк матрица представляет собой набор векторов-строк: $A \in \Re^{n\times m} \leftrightarrow A = \begin{pmatrix} a_1 \\ .. \\ a_k \\ .. \\ a_n \end{pmatrix}, a_k \in \Re^m$. То есть, в этом случае функция matvec будет устроена следующим образом:

$$
\begin{array}{l}
z(1:n;1) = 0; \\
for\ i = 1:n \\
\quad z(i) = z(i) + a(i,:)\cdot x(:,1); \\
end\ \%\ for\ i
\end{array}
$$

где $a_i = a(i,:)$.

При разбиении по столбцам матрица A рассматривается как набор векторов-столбцов. То есть:

$A \in \Re^{n\times m} \leftrightarrow A = (a_1, a_2, ..., a_m), a_k \in \Re^n$

Тогда функцию с доступом по столбцам можно реализовать процедурой:

$$
\begin{array}{l}
z(1:n;1) = 0; \\
for\ j = 1:m \\
\quad z(i) = z(i) + x(j,1)\cdot a(j,i); \\
end\ \%\ for\ j
\end{array}
$$

В вычислительной практике более предпочтительными считаются алгоритмы, выбирающие элементы массивов по столбцам.

Рассмотрим часто встречающиеся в алгоритмах расчета систем задачу модификации вектора:

$$z = y + A\cdot x, x \in \Re^n, y \in \Re^m, A \in \Re^{n\times m}\ .$$

Тогда алгоритм ее решения в блочной схеме доступа можно реализовать следующим образом:

```
function z = gaxpy(A, x, y)
  m = cols(A); z = y;
  for j = 1:m
    z = z + x(j) · A(:, j);
  end % for j
end %gaxpy
```

Обозначение $A(:, j)$ означает представление матрицы в виде набора векторов-столбцов.

Рассмотрим теперь процедуру умножения матриц. Пусть $A \in \Re^{n \times r}, B \in \Re^{r \times m}$ и необходимо вычислить $C = A \cdot B$. Стандартная процедура заключается в последовательном вычислении элементов матрицы C в порядке слева направо и сверху вниз.

```
function C = matmatr(A, B)
  n = rows(A); r = cols(A); m = cols(B)
  C(1:n, 1:m) = 0;
  for i = 1:n
    for j = 1:m
      for k = 1:r
        C(i, j) = C(i, j) + A(i, k) · B(k, j);
      end % for k
    end % for j
  end % for i
end %matmatr
```

Теперь предположим, что матрицы A, B, C разбиты на столбцы

$A = (a_1, ..., a_m), a_k \in \Re^n$, $B = (b_1, ..., b_m), b_k \in \Re^r$, $C = (c_1, ..., c_n), c_k \in \Re^m$

Тогда алгоритм можно записать в более компактном виде, используя вышеприведенную функцию:

```
z = gaxpy
  C(1:n, 1:m) = 0;
  for j = 1:m
    C(:, j) = gaxpy(A, B(:, j), 0(:, j));
  end % for j
```

В приведенном алгоритме, по сути, используется следующее соотношение $AB = \sum_{k=1}^{r} a_k b_{k,j}^T$.

Эффективность массовых матричных алгоритмов умножения можно значительно увеличить, используя индивидуальные особенности матриц. Например, рассмотрим ленточные матрицы.

Будем говорить, что матрица $A \in R^{n*m}$ имеет нижнюю ширину ленты p , если $a_{ij} = 0$ для $i > j + p$, и верхнюю ширину ленты q , если из $j > i + q$ следует $a_{ij} = 0$.

Пример. Матрица 8*5 с нижней шириной ленты 1 и верхней шириной 2 имеет вид:

$$\begin{bmatrix} \times & \times & \times & 0 & 0 \\ \times & \times & \times & \times & 0 \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Основные типы ленточных матриц, применяемых в вычислительных процедурах, приведены в нижележащей таблице.

Тип матрицы	Нижняя ширина ленты	Верхняя ширина ленты
диагональная	0	0
верхняя треугольная	0	$n-1$
нижняя треугольная	$m-1$	0
трехдиагональная	1	1
верхняя двухдиагональная	0	1
нижняя двухдиагональная	1	0
верхняя хессенбергова	1	$n-1$
нижняя хессенбергова	$m-1$	1

Рассмотрим алгоритм матричного умножения $C = A \cdot B$, где $A, B \in \Re^{n \times n}$ верхние треугольные матрицы. Матрица C, в этом случае, также будет являться верхней треугольной матрицей.

$$\begin{aligned} &C(1:n,1:n) = 0; \\ &for\ i = 1:n \\ &\quad for\ j = 1:n \\ &\quad\quad for\ k = i:j \\ &\quad\quad\quad C(i,j) = C(i,j) + A(i,k) \cdot B(k,j); \\ &\quad\quad end\ \%\ for\ k \\ &\quad end\ \%\ for\ j \\ &end\ \%\ for\ i \end{aligned}$$

Проведем количественную оценку достигнутой в этом алгоритме экономии процессорного времени. Для этого введем понятие флопа.

Флоп – это одна операция с плавающей точкой. Так скалярное произведение двух векторов размерностью n содержит $2 \cdot n$ флопов, поскольку состоит из n умножений и n сложений.

Умножение матрицы $A \in \Re^{n \times m}$ на вектор, содержит $2 \cdot m \cdot n$ флопов. Умножение матриц $C = A \cdot B$, где $A \in \Re^{n \times r}, B \in \Re^{r \times m}$, содержит $2 \cdot m \cdot n \cdot r$ флопов.

Для оценки количества флопов обычно суммируют арифметические затраты для наиболее глубоких вложений операторов алгоритма.

Найдем оценку объема вычислительной работы для приведенного выше алгоритма умножения треугольных матриц. Для этого используем следующие соотношения:

$$\sum_{p=1}^{q} p = \frac{q(q+1)}{2} = \Theta(\frac{q^2}{2}),\ \sum_{p=1}^{q} p^2 = \frac{q^3}{3} + \frac{q^2}{2} + \frac{q}{6} = \Theta(\frac{q^3}{3})\ .$$

На основе этих равенств получим следующую оценку вычислительной работы алгоритма умножения двух треугольных матриц: $\sum_{i=1}^{n}\sum_{j=1}^{n} 2(j-i+1) = \sum_{i=1}^{n}\sum_{j=1}^{n-i+1} 2j \approx \sum_{i=1}^{n} i^2 = \Theta(\frac{n^3}{3})$.

Следует сразу сказать, что оценка количества флопов, это грубый подход к измерению эффективности программ. Такой подход игнорирует затраты на индексацию, обмены с памятью, другими устройствами и прочие многочисленные издержки. Кроме того флоп не учитывает различную длительность выполнения таких операций, как сложение, умножение и деление. Флопы дают лишь первое приближенное измерение эффективности алгоритма.

Блочные матрицы.

В последнее время все шире используют параллельные вычисления на нескольких процессорах. Для таких систем удобно использовать так называемые блочные алгоритмы. Структуру матриц для таких алгоритмов также удобно представлять в виде блочных матриц.

Разбиение матрицы $A \in \Re^{n\times m}$ на блоки выглядит следующим образом:

$$A = \begin{pmatrix} A_{11} & \dots & A_{1q} \\ \dots & \dots & \dots \\ A_{p1} & \dots & A_{pq} \end{pmatrix} \begin{matrix} n_1 \\ \dots \\ n_p \end{matrix}.$$
$$m_1 \dots\dots\ m_q$$

Здесь $m_1 + \dots + m_q = m;\ n_1 + \dots + n_p = n$, A_{ij} означает (i, j) блок или подматрицу размерности $(n_i \times m_j)$. В этом случае, говорят, что $A \in \Re^{n\times m}$ есть (p,q) блочная матрица.

Если матрица B разбита на блоки, согласовано с матрицей A, то их сумма $C = A + B$ есть $(p \times q)$ блочная матрица $C_{ij} = A_{ij} + B_{ij}, i = 1,2,\dots, p;\ j = 1,2,\dots, q$.

Рассмотрим теперь задачу умножения блочных матриц.

***Теорема**. Если*

$$A = \begin{pmatrix} A_{11} & \dots & A_{1q} \\ \dots & \dots & \dots \\ A_{p1} & \dots & A_{pq} \end{pmatrix} \begin{matrix} n_1 \\ \dots \\ n_p \end{matrix},\ B = \begin{pmatrix} B_{11} & \dots & B_{1t} \\ \dots & \dots & \dots \\ B_{q1} & \dots & B_{qt} \end{pmatrix} \begin{matrix} r_1 \\ \dots \\ r_q \end{matrix}$$
$$r_1 \dots\dots\ r_q \qquad\qquad m_1 \dots\dots\ m_t$$

, и произведение $C = A \cdot B$ разбито на блоки следующим образом: $C = \begin{pmatrix} C_{11} & \dots & C_{1r} \\ \dots & \dots & \dots \\ C_{p1} & \dots & C_{pr} \end{pmatrix} \begin{matrix} n_1 \\ \dots \\ n_p \end{matrix}$, *то* $C_{ij} = \sum_{k=1}^{q} A_{ik} B_{kj}; i = 1{:}\ p;\ j = 1{:}\ t$

$$m_1 \dots\dots\ m_t$$

Существует еще один подход к умножению матриц, основанный на так называемом принципе «разделяй и властвуй». Предварительно рассмотрим случай умножения 2*2 блочных матриц

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}.$$

В обычном алгоритме вычисления $C_{ij} = A_{i1}B_{j1} + A_{i2}B_{2j}$ имеется 8 умножений и 4 сложения. Штрассен предложил способ вычисления C с использованием 7 умножений и 18 сложений блоков.

$$
\begin{aligned}
P_1 &= (A_{11} + A_{22})(B_{11} + B_{22}),\\
P_2 &= (A_{21} + A_{22})B_{11},\\
P_3 &= A_{11}(B_{12} - B_{22}),\\
P_4 &= A_{22}(B_{21} - B_{11}),\\
P_5 &= (A_{11} + A_{12})B_{22},\\
P_6 &= (A_{21} - A_{11})(B_{11} + B_{12}),\\
P_7 &= (A_{12} - A_{22})(B_{21} + B_{22}),\\
C_{11} &= P_1 + P_4 - P_5 + P_7,\\
C_{12} &= P_3 + P_5,\\
C_{21} &= P_2 + P_4,\\
C_{22} &= P_1 + P_3 - P_2 + P_6.
\end{aligned}
$$

Можно показать, что метод Штрассена будет обходиться примерно в 7/8 арифметических затрат стандартного алгоритма. Метод Штрассена может быть реализован с помощью рекурсивной процедуры, так как алгоритм его реализующий может быть применен для нахождения каждого из произведений блоков половинного размера /Голуб с21/.

Решение линейных уравнений.

Пусть задано линейное матричное уравнение $Ax = b$, где $A \in \Re^{n\times n}; x, b \in \Re^n$. Уравнение имеет единственное решение, когда матрица A невырожденна.

Теорема (условие невырожденности) */Уоткинс с23/. Пусть матрица A квадратная матрица. Следующие шесть условий эквивалентны, то есть если любое из них выполнено, то выполняются они все.*

1. A^{-1} *- существует.*
2. *Не существует отличного от нуля вектора* y *такого, что* $Ay = 0$.
3. *Столбцы матрицы* A *линейно независимы.*
4. *Строки матрицы* A *линейно независимы.*
5. $\det(A) \neq 0$.
6. *Для любого заданного вектора* b *имеется точно один такой вектор* x, *что* $Ax = b$.

Если условия, приведенные в теореме, не выполняются, то говорят, что матрица A вырождена или необратима. Если матрица A невырождена, то решение уравнения определяется соотношением $x = A^{-1} \cdot b$. Однако решать, таким образом, уравнение $Ax = b$ не очень эффективно. Для достаточно больших задач экономия в вычислении и хранении, достигаемая за счет отказа от использования матрицы A^{-1}, весьма существенна.

Рассмотрим предварительно линейные системы с треугольной матрицей коэффициентов. Такие системы решаются быстро и эффективно и вычислительные затраты на их решение невелики.

Пусть матрица $G=(g_{ij})$ является нижней треугольной, если $g_{ij}=0$ при $i<j$, то есть имеет вид:

$$G=\begin{pmatrix} g_{11} & 0 & \dots & 0 \\ g_{21} & g_{22} & \dots & 0 \\ \cdot & \cdot & \dots & 0 \\ g_{n1} & g_{n2} & \dots & g_{nn} \end{pmatrix}.$$

Аналогично у верхней треугольной матрицы $g_{ij}=0$ при $i>j$.

Теорема */Уоткинс с38/* *. Пусть $G=(g_{ij})$ треугольная матрица. Матрица G является невырожденной тогда и только тогда, когда $g_{ii}\neq 0; i=1,2,\dots,n$*

Рассмотрим систему $Gy=b$, где $G=(g_{ij})$ - невырожденная нижняя треугольная матрица. Тогда решение такой системы будет даваться следующей формулой $y_i=g_{ii}^{-1}(b_i-\sum_{j=1}^{i-1}g_{ij}y_j)$. Такой алгоритм решения нижних треугольных систем называется прямым исключением, так как матрица G выбирается строка за строкой и i-я строка используется на i - ом шаге. Трудоемкость (флопов) такого алгоритма оценивается с помощью следующего соотношения:

$$\sum_{i=1}^{n}\sum_{j=1}^{i-1}2=2\sum_{i=1}^{n}(i-1)=n(n-1)\approx\Theta(n^2).$$

Выведем столбовую версию прямой подстановки. Разобьем систему $Gy=b$ на блоки следующим образом $\begin{pmatrix} g_{11} & 0 \\ \hat{h} & \hat{G} \end{pmatrix}\cdot\begin{pmatrix} y_1 \\ \hat{y} \end{pmatrix}=\begin{pmatrix} b_1 \\ \hat{b} \end{pmatrix}$. Здесь $\hat{h},\hat{y},\hat{b}$ векторы длиной $(n-1)$, а $\hat{G}$ нижняя треугольная матрица размером $(n-1)\times(n-1)$. Приведенную блочную систему можно записать в следующем виде: $\begin{cases} g_{11}y_1=b_1 \\ \hat{h}y_1+\hat{G}\hat{y}=\hat{b} \end{cases}$. Тогда процесс решения можно описать следующим рекурсивным алгоритмом: решить $\hat{G}\hat{y}=\hat{b}$ относительно $\hat{y}$, где $y_1=b_1/g_{11}$, $\tilde{b}=\hat{b}-\hat{h}y_1$.

Для систем с верхней треугольной матрицей алгоритм решения будет аналогичным.

<u>Положительно определенные матрицы. Разложение Холесского.</u>

Если матрица $A\in\Re^{n\times n}$ вещественная, симметричная и удовлетворяет условию $x^TAx>0$ для всех ненулевых $x\in\Re^n$, то матрица A называется положительно-определенной.

Теорема*. Если матрица A положительно определенная, то она невырожденная.*

Доказательство. Действительно, если матрица A вырожденная, то существует ненулевой вектор $y\in\Re^n$ такой, что $Ay=0$. Но тогда будет выполняться соотношение $y^TAy=0$, то есть матрица A не является положительно определенной.

Теорема. *Пусть $M\in\Re^{n\times n}$ произвольная невырожденная матрица и $A=M^TM$. Тогда матрица A положительно определенная.*

Доказательство. Матрица A является симметрической. Действительно, имеет место соотношение $A^T = (M^T M)^T = M^T (M^T)^T = M^T M = A$. Пусть теперь $y = Mx$ для некоторого ненулевого вектора $x \in \Re^n$. Тогда имеет место следующее выражение $x^T Ax = y^T y > 0$, так как матрица M и, соответственно, матрица A являются невырожденными.

***Теорема (о разложении Холесского)**. Пусть A положительно определенная матрица. Тогда существует единственное представление матрицы A в виде произведения $A = G^T G$, где G есть верхняя треугольная матрица, у которой все элементы g_{ii} главной диагонали положительны. Матрица G называется множителем Холесского матрицы A.*

Пусть необходимо решить систему $Ax = b$, где A положительно определенная матрица. Тогда, в соответствии с теоремой можно записать $G^T Gx = b$. Пусть $y = Gx$. Очевидно, что вектор y является решением системы $G^T y = b$, где G^T - нижняя треугольная матрица. Такую систему легко решить методом прямой подстановки. Найдя вектор y, можно решить верхнюю треугольную систему $Gx = y$ относительно вектора x обратной подстановкой в правую часть вектора y. Если множитель Холесского G известен, то общее число флопов для поиска решения x составит $2 \cdot n^2$.

Метод Холесского с внешним произведением.

Вариант метода с внешним произведением получается разбиением выражения $A = G^T \cdot G$ в виде:

$$\begin{pmatrix} a_{11} & b^T \\ b & \hat{A} \end{pmatrix} = \begin{pmatrix} g_{11} & 0 \\ s & \hat{G}^T \end{pmatrix} \cdot \begin{pmatrix} g_{11} & s^T \\ 0 & \hat{G} \end{pmatrix}.$$

Приравнивая блоки, получим следующие три равенства: $a_{11} = r_{11}^2; b^T = r_{11} s^T; A = ss^T + \hat{G}^T \hat{G}$. Четвертое равенство $b = sr_{11}$ является лишним, так как дублирует второе соотношение. Отсюда можно получить следующую процедуру вычисления r_{11}, s^T и $\hat{G}$ (а, следовательно, G): решить уравнение $\hat{A} = \hat{G}^T \hat{G}$ относительно $\hat{G}$, где
$r_{11} = \sqrt{a_{11}}$, $s^T = r_{11}^{-1} b^T$, $\hat{A} = \hat{A} - ss^T$.

Данный алгоритм сводит задачу нахождения множителя Холесского для матрицы размеров $(n \times n)$ к задаче для $((n-1) \times (n-1))$ матрицы $\hat{A}$. Эта задача может быть редуцирована к задаче для $((n-2) \times (n-2))$ матрицы посредством того же алгоритма и т.д. В конечном итоге задаче редуцируется к тривиальному случаю матрицы 1*1. Процедура называется методом Холесского с внешним произведением, потому что на каждом шаге внешнее произведение ss^T вычитается из оставшейся подматрицы. Такая процедура легко может быть реализована в виде рекурсивного алгоритма. Число флопов, необходимых для реализации процедуры Холесского, равно $\approx \frac{n^3}{3}$.

Исключение Гаусса и LU- разложение

Рассмотрим теперь общую задачу решения линейных уравнений $Ax = b$ методом гауссова исключения. Приводимый ниже алгоритм дает единственное решение всякий раз, когда матрица A невырожденная. Основным принципом рассматриваемой процедуры заключается в том, чтобы преобразовать систему $Ax = b$ в эквивалентную систему $Ux = y$ с верхней треугольной матрицей коэффициентов. Система $Ux = y$ легко решается с использованием метода обратной подстановки, при условии, что матрица U является невырожденной. При этом, под понятием «две системы эквивалентны» будем понимать, что они имеют одни и те же решения. Введем следующие элементарные операции:

1. Сложить кратное одного уравнения с другим.
2. Переставить два уравнения.
3. Умножить уравнение на ненулевую константу.

***Утверждение**. Если система $\hat{A}x = \hat{b}$ получена из системы $Ax = b$ с помощью элементарных операций 1,2 и 3, то эти системы эквивалентны.*

Для проведения элементарных операций удобно представлять систему $Ax = b$ с помощью расширенной матрицы $\{A|b\}$. Каждое уравнение системы $Ax = b$ соответствует строке такой матрицы. Соответственно, элементарные операции над уравнениями равносильны следующим элементарным операциям над строками матрицы $\{A|b\}$. Будем вначале рассматривать такую матрицу A, которая обладает свойством (все главные миноры матрицы A отличны от нуля), позволяющим приводить ее к треугольной форме, используя лишь элементарную операцию типа 1. Приведение к треугольной форме в этом случае выполняется за $(n-1)$ шагов. На первом шаге соответствующее кратное первой строки вычитается из каждой другой строки так, чтобы получить нули в позициях $(2,1);(3,1);...;(n,1)$ преобразованной матрицы. Для этого необходимо, чтобы $a_{11} \neq 0$. Соответствующий множитель для i - ой строки будет равен $m_{i1} = a_{i1} / a_{11}, i = 2,...,n$. Коэффициенты преобразованной матрицы будут иметь вид $a_{ij}^{(1)} = a_{ij} - m_{i1}a_{1j}, j = 2,...,n; i = 2,...,n$, $b_i^{(1)} = b_i - m_{i1}b_1, i = 2,...,n$. Матрица $\{A|b\}$, при этом, приводится к виду:

$$\left[\begin{array}{c|ccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ \hline 0 & a_{22}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right].$$

При выполнении вычислений нет необходимости записывать нули в первый столбец. Поэтому там обычно хранятся множители $m_{21},...,m_{n1}$. Поэтому, после первого шага, массив содержащий матрицу $\{A|b\}$ примет вид:

$$\left[\begin{array}{c|ccc|c} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ \hline m_{21} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} & b_2^{(1)} \\ \vdots & \vdots & & \vdots & \vdots \\ m_{n1} & a_{n2}^{(1)} & \dots & a_{nn}^{(1)} & b_n^{(1)} \end{array}\right].$$

После $(n-1)$ шагов система будет приведена к виду $\{U \mid y\}$, где U - верхняя треугольная матрица. Очевидно, что матрица U будет невырожденной, так как матрица A является невырожденной. Поэтому система $Ux = y$ может быть решена относительно вектора x обратной подстановкой. Общая стоимость решения системы $Ax = b$ этим методом будет равна $\frac{2}{3}n^3$ флопов.

Более точное рассмотрение преобразования вектора b в вектор y дает важную интерпретацию гауссова исключения. Выразим преобразованные компоненты b через компоненты вектора y, используя соответствующие множители, для разных шагов алгоритма.

$$b_i^{(1)} = b_i - m_{i1}y_1, i = 2,3,...,n;$$
$$b_i^{(2)} = b_i^{(1)} - m_{i2}y_2, i = 3,4,...,n;$$
$$....$$
$$b_i^{n-(1)} = b_i^{(n-2)} - m_{i,(n-1)}y_{n-1}, i = n.$$

Если рассмотреть все компоненты вектора y, то есть $y_1, y_2, ..., y_n$, то эти формулы можно записать в виде $\sum_{j=1}^{i-1} m_{ij}y_j + y_i = b_i; i = 1,2,...,n$. Таким образом, вектор y является решением линейной системы $Ly = b$, где L - нижняя треугольная матрица, которая имеет следующий вид:

$$\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{21} & 1 & 0 & \ddots & \vdots \\ m_{31} & m_{32} & 1 & \ddots & \\ \vdots & & & \ddots & 0 \\ m_{n1} & m_{n2} & m_{3n} & \dots & 1 \end{bmatrix}.$$

Такую матрицу L еще часто называют нижней уни - треугольной матрицей, так как все элементы ее главной диагонали равны единице. Таким образом, чтобы решить систему $Ax = b$ ее нужно привести к виду $Ux = y$, где U - верхняя треугольная матрица, а y является решением нижней уни - треугольной системы $Ly = b$. Тогда можно записать $LUx = b = Ax$, то есть $A = L \cdot U$. Конечная преобразованная матрица $\hat{A}$ имеет вид:

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ m_{21} & m_{22} & m_{23} & \cdots & m_{2n} \\ m_{31} & m_{32} & m_{33} & \cdots & m_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ m_{n1} & m_{n2} & m_{n3} & & m_{nn} \end{bmatrix},$$

где содержится вся информация о матрицах L,U .

***Теорема (о $L\cdot U$ разложении)**. Пусть $A\in\Re^{n\times n}$ невырожденная матрица. Тогда матрицу A можно разложить единственным образом в произведение матриц $A=L\cdot U$, так что L является нижней уни - треугольной, а U - верхней треугольной матрицами.*

Рассмотрим теперь решение системы $Ax=b$ в отсутствии предположения о невырожденности ведущих главных подматриц. После $(k-1)$ - го шага алгоритма гауссова исключения, массив, содержащий вначале матрицу A примет вид

$$\left[\begin{array}{cccc|ccc} u_{11} & u_{12} & \cdots & u_{1,k-1} & u_{1k} & & \cdots & u_{1n} \\ m_{21} & u_{22} & \cdots & u_{2,k-1} & u_{2k} & & \cdots & u_{2n} \\ \vdots & \ddots & \ddots & \vdots & \vdots & & & \vdots \\ m_{k-1,1} & & & u_{k-1,k-1} & u_{k-1,k} & & \cdots & u_{k-1,n} \\ \hline m_{k1} & \cdots & & m_{k,k-1} & a_{k,k}^{(k-1)} & a_{k,k+1}^{(k-1)} & \cdots & a_{n,k}^{(k-1)} \\ & & & & a_{k+1,k}^{(k-1)} & a_{k+1,k+1}^{(k-1)} & \cdots & a_{k+1,n}^{(k-1)} \\ \vdots & & & \vdots & \vdots & \vdots & & \vdots \\ m_{n1} & \cdots & & m_{n,k-1} & a_{n,k}^{(k-1)} & a_{n,k+1}^{(k-1)} & \cdots & a_{n,n}^{(k-1)} \end{array}\right].$$

Чтобы вычислить множители k - го шага, необходимо выполнить деление на $a_{kk}^{(k-1)}$. Если значение $a_{kk}^{(k-1)}=0$, то применим предварительно построчные операции типа 2 (перестановка строк), чтобы получить в позиции (k,k) не равный нулю элемент. Чтобы уменьшить ошибку при делении, выбирается элемент, обладающий наибольшим значением по модулю. Такой способ перестановки называется выбором главного элемента. После $(n-1)$ - го шага разложение завершается. Остается сделать заключительную проверку: если $a_{kk}^{(n-1)}=0$, то матрица A вырождена и, тогда необходимо выбрать соответствующий элемент. Следует иметь в виду, когда производится перестановка строк, то переставляются и множители, соответствующие этим строкам. Это необходимо учитывать при получении окончательного решения.

***Теорема (о $L\cdot U$ разложении)**. Пусть $A\in\Re^{n\times n}$ невырожденная матрица. Тогда матрицу A можно разложить единственным образом в произведение матриц $A=L\cdot U$, так что L является нижней уни - треугольной, а U - верхней треугольной матрицами.*

В пакетах MatLab и MatCad имеются соответствующие подпрограммы ($lu(A)$), реализующие процедуру $L \cdot U$ разложения. Для уравнений большой размерности используются пакеты линейной алгебры LINPACK и LAPACK (свободный доступ к пакетам http: // www. netlib. org).

Вычисление обратной матрицы.

Рассмотрим снова систему $Ax = b$. Положив, что $X = A^{-1}$, можно записать следующее матричное уравнение $AX = E$. Перепишем это уравнение в блочной форме в следующем виде $A \cdot [x_1, x_2, ..., x_n] = [e_1, e_2, ..., e_n]$, где $x_1, x_2, ..., x_n$ и $e_1, e_2, ..., e_n$ - столбцы соответственно матриц X, E. То есть матричное уравнение $AX = E$ эквивалентно системе уравнений $Ax_i = e_i, i = 1, 2, ..., n$.

Решая эти n систем методом гауссова исключения, с выбором главного элемента, получим матрицу $X = A^{-1}$. Общая стоимость такого алгоритма равна $\frac{8}{3}n^3$ флопов. Известный из курса линейной алгебры метод алгебраических дополнений для вычисления обратной матрицы дается формулой $A^{-1} = \frac{1}{\det(A)} adj(A)$, где $adj(A)$ - алгебраическое дополнение матрицы A. В этом методе требуется вычислить много детерминантов. Если детерминанты вычисляются классическим образом, то затраты составят $n!$ флопов.

Рассмотрим алгоритмы обращения блочных матриц. Предварительно рассмотрим схему вычисления определителя блочной матрицы размера 2x2.

Лемма Шура. *Пусть заданы матрицы* $A \in \Re^{n \times n}; B \in \Re^{n \times m}; C \in \Re^{m \times n}; D \in \Re^{m \times m}$. *Тогда, если* $\det A \neq 0$, *имеет место равенство:* $\det \begin{pmatrix} A & B \\ C & D \end{pmatrix} = \det(A) \cdot \det(D - CA^{-1}B)$. *Если* $\det D \neq 0$, *то имеет место равенство:* $\det \begin{pmatrix} A & B \\ C & D \end{pmatrix} = \det(D) \cdot \det(A - BD^{-1}C)$.

Доказательство.

Из условия леммы вытекает, что матрица A является квадратной и обращаемой (инвертируемой). Применим операции над строками и столбцами к матрице $Z = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$, чтобы подматрицы нижнего левого угла и верхнего правого угла были равны нулю. Для этого умножим первую строку на $C \cdot A^{-1}$ и вычтем из второй строки, а затем вычтем первый столбец, умноженный на $A^{-1} \cdot B$ из второго столбца:
$\begin{pmatrix} E_m & 0 \\ -CA^{-1} & E_n \end{pmatrix} \cdot \begin{pmatrix} A & B \\ C & D \end{pmatrix} \cdot \begin{pmatrix} E_m & -A^{-1}B \\ 0 & E_n \end{pmatrix} = \begin{pmatrix} A & 0 \\ 0 & D - CA^{-1}B \end{pmatrix}$. Вычислив детерминант модифицированной матрицы, найдем:

$$\det \begin{pmatrix} A & B \\ C & D \end{pmatrix} = \det \begin{pmatrix} A & 0 \\ 0 & D - CA^{-1}B \end{pmatrix} = \det(A) \cdot \det(D - CA^{-1}B).$$

Обращение блочной матрицы.

Если подматрицы A и D являются несингулярными, то: $Z^{-1} = \begin{pmatrix} A & 0 \\ 0 & D \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} & 0 \\ 0 & D^{-1} \end{pmatrix}$.

В общем случае, если матрицы A и $P = D - CA^{-1}B$ являются несингулярными (невырожденными), получим:

$$Z^{-1} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} A^{-1} + A^{-1}BP^{-1}CA^{-1} & -A^{-1}BP^{-1} \\ -P^{-1}CA^{-1} & P^{-1} \end{pmatrix};$$

Если матрицы D и $F = A - BD^{-1}C$ являются несингулярными, получим:

$$Z^{-1} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}^{-1} = \begin{pmatrix} F^{-1} & -F^{-1}BD^{-1} \\ -D^{-1}CF^{-1} & D^{-1} + D^{-1}CF^{-1}BD^{-1} \end{pmatrix}.$$

Нормы векторов и матриц.

Нормы векторов.

Чтобы изучать возмущения векторов и матриц, нужно уметь измерять их. Для этого вводятся нормы векторов и матриц. Векторная норма в $\Re^n$ есть функция сопоставляющая каждому вектору $x \in \Re^n$ неотрицательное вещественное число $\| x \|$, которое называется нормой вектора x, такое, что выполняются следующие условия при всех $x, y \in \Re^n$ и всех $\alpha \in \Re$:

$$\| x \| > 0, x \neq 0 \text{ и } \| 0 \| = 0;$$
$$\| \alpha x \| = | \alpha | \cdot \| x \|;$$
$$\| x + y \| \leq \| x \| + \| y \|.$$

Пример. Евклидова норма определяется как $\| x \|_2 = (\sum_{i=1}^{n} | x_i \|^2)^{1/2}$.

***Теорема (Неравенство Коши-Шварца)**. Для любых $x, y \in \Re^n$ имеет место соотношение*

$$| \sum_{i=1}^{n} x_i y_i | \leq (\sum_{i=1}^{n} x_i^2)^{1/2} (\sum_{i=1}^{n} y_i^2)^{1/2}.$$

Доказательство.

Для любого вещественного t имеем $0 \leq \sum_{i=1}^{n} (x_i + t y_i)^2 = \sum_i x_i^2 + 2t \sum_i x_i y_i + t^2 = c + bt + at^2$, где

$a = \sum_{i=1}^{n} y_i^2$, $b = 2\sum_{i=1}^{n} x_i y_i$ и $c = \sum_{i=1}^{n} x_i^2$. Так как $c + bt + at^2 \geq 0$ для всех вещественных t, этот многочлен не может иметь разных вещественных корней, то есть $b^2 - 4ac \leq 0$ или $(b/2)^2 \leq ac$. Отсюда, подставляя исходные значения a, b, c, получим утверждение теоремы.

Из результата доказанной теоремы несложно показать, что для любых $x, y \in \Re^n$ будет выполняться соотношение (неравенство треугольника) $\| x + y \|_2^2 \leq \| x \|_2 + \| y \|_2$. Обобщение

евклидову норму можно ввести понятие p - нормы: $\| x \|_P = (\sum_{i=1}^{n} | x_i |^P)^{1/p}$. При $p = 1$ норма определяется следующим соотношение $\| x \|_1 = \sum_{i=1}^{n} | x_i |$. При $p = \infty$ норма будет определяться выражением $\| x \|_\infty = \max_{1 \le i \le n} | x_i |$. Довольно часто, в конкретных примерах, вводится понятие А – нормы для вектора $x \in \Re^n$ с помощью следующего выражения $\| x \|_A = (x^T A x)^{1/2}$.

В частности, если матрица A – положительно определенная матрица и G ее множитель Холесского, то есть $A = G^T \cdot G$. Тогда $\| x \|_A = \| Gx \|_2$.

Матричные нормы.

Норма матриц рассматривается, в основном, в отношении квадратных матриц $A \in \Re^{n \times n}$.

Функция $\| . \|_m : \Re^{n \times n} \to \Re$ называется матричной нормой, если для любых матриц $A, B \in \Re^{n \times n}$, выполняются следующие пять аксиом:

$\| A \|_m \ge 0$;

$\| A \|_m = 0$, тогда и только тогда, если $A = 0$;

$\| c \cdot A \|_m = | c | \cdot \| A \|_m$ для всех постоянных $c \in \Re$;

$\| A + B \|_m \le \| A \|_m + \| B \|_m$;

$\| A \cdot B \|_m \le \| A \|_m \cdot \| B \|_m$.

Однако пространство $\Re^{n \times n}$ это не только пространство высокой векторной размерности, оно имеет целый ряд специфических операций, присущих именно матричной форме представления данных.

Пример.

Норма Фробениуса определяется следующим выражением $\| A \|_F = (\sum_{i=1}^{n} \sum_{j=1}^{n} | a_{ij} |^2)^{1/2}$. С другой стороны $\| A \|_{\max} = \max_{1 \le i, j \le n} | a_{ij} |$ не является матричной нормой.

В общем случае, чтобы определить понятие нормы $\| . \|_m$ в матричном пространстве $\Re^{n \times n}$ следует следовать следующему условию.

Определение.

Пусть имеется некоторое определение нормы $\| . \|_a$ *в пространстве векторов* $\Re^n$. *Тогда определим* $\| . \|_m$, *как векторную матричную норму в пространстве* $\Re^{n \times n}$ *следующим образом:* $\| A \|_m = \max_{\|x\|_a = 1} \| Ax \|_a$, *где* $\| . \|_a$ *определяет некоторую норму в векторном пространстве* $\Re^n$.

Можно показать, что матричная форма может быть вычислена также следующим образом:

$$\| A \|_m = \max_{\|x\|=1} \| Ax \|_a = \max_{\|x\|_a=1} \frac{\| Ax \|_a}{\| x \|_a}.$$

Каждая векторная норма в $\Re^n$ определяет матричную норму в пространстве $\Re^{n\times n}$. В частности, любая матричная норма, индуцированная векторной нормой $\|.\|$, определяется с помощью соотношения $\| A \|_M = \max_{x\neq 0} \frac{\| Ax \|_v}{\| x \|_v}$. Другим названием такой нормы служит операторная норма. Число $\| A \|_M$ в геометрическом смысле можно интерпретировать как максимум растяжения, вызываемого оператором A. Обычно индексы v и M не различаются. Поэтому приведенное соотношение часто записывают в виде $\| A \| = \max_{x\neq 0} \frac{\| Ax \|}{\| x \|}$.

Теорема */Уоткинс с132/. Векторная норма и индуцированная ею матричная норма удовлетворяют неравенству* $\| Ax \| \leq \| A \| \cdot \| x \|$ *для всех* $A \in \Re^{n\times n}$ *и* $x \in \Re^n$. *Это неравенство является точным в следующем смысле: для каждой матрицы* $A \in \Re^{n\times n}$ *существует отличный от нуля вектор* $x \in \Re^n$, *для которого имеет место равенство.*

Теорема. *Индуцированная норма является нормой.*

Следствие. $\| A \| = \max_{|x|=1} \frac{\| Ax \|}{\| x \|}$

Рассмотрим некоторые примеры матричных норм.

Для $1 < p < \infty$ норма, индуцированная векторной p нормой, называется p матричной нормой и определяется соотношением $\| A \|_p = \max_{x\neq 0} \frac{\| Ax \|_p}{\| x \|_p}$. При значениях $p = 1$ и $p = \infty$ нормы имеют следующий вид $\| A \|_1 = \max_{1\leq j\leq n} \sum_{i=1}^{n} | a_{ij} |$, $\| A \|_\infty = \max_{1\leq i\leq n} \sum_{j=1}^{n} | a_{ij} |$. Поэтому матричную 1-норму часто называют нормой суммирования по столбцам. Соответственно матричную ∞ - норму называют нормой суммирования по строкам.

Соответственно можно определить спектральную норму $\|.\|_2$ в матричном пространстве $\Re^{n\times n}$ как: $\| A \|_2 = \sigma_1(A)$, где $\sigma_1(A)$ является наибольшим сингулярным числом (значением) матрицы A.

Заметим, что Фробениусова норма $\| A \|_2 = | tr(AA^*) |^{1/2} = (\sum_{i,j=1}^{n} | a_{ij} |^2)^{1/2}$ является абсолютной, так как она является одновременно Эвклидовой нормой для матрицы A, представленной, как вектор

в пространстве $\Re^{n^2}$. Так как $tr(AA^*)$ является суммой собственных значений матрицы $A\cdot A^*$, и эти собственные значения представляют собой квадраты сингулярных чисел матрицы A, то получим: $\| A\|_2=\sqrt{\sigma_1(A)^2+...+\sigma_n(A)^2}$.

Число обусловленности матрицы.

Число обусловленности матрицы A широко используется в анализе чувствительности линейных систем $Ax=b$. Рассмотрим вариацию параметра δb. Тогда можно записать следующее соотношение $A\delta x=\delta b$, где δx - соответствующая вариация решения. Из уравнения $\delta x=A^{-1}\delta b$ и свойств индуцированной матричной нормы вытекает следующее неравенство $\| \delta x\|<\| A^{-1}\|\cdot\| \delta b\|$. Подобным образом из уравнения $Ax=b$ следует неравенство $\| b\|<\| A\|\cdot\| x\|$ или $\frac{1}{\| x\|}\le\| A\|\frac{1}{\| b\|}$. Тогда для относительного изменения решения $\frac{\| \delta x\|}{\| x\|}$ можно записать следующее ограничение $\frac{\| \delta x\|}{\| x\|}\le\| A\|\cdot\| A^{-1}\|\frac{\| \delta b\|}{\| b\|}$. Множитель $k(A)=\| A\|\cdot\| A^{-1}\|$ - называется числом обусловленности матрицы A. Несложно заметить, что выполняется соотношение $k(A)=k(A^{-1})$. Из полученного соотношения видно, что если $k(A)$ не очень велико, то малые значения $\frac{\| \delta b\|}{\| b\|}$ приводят к малым значениям $\frac{\| \delta x\|}{\| x\|}$. Поэтому, если $k(A)$ не очень велико, то говорят, что матрица A хорошо обусловлена. В противном случае, когда $k(A)$, то предполагают, что матрица A плохо обусловлена. Обычно, для оценки числа обусловленности матрицы, используют 1-, 2- и ∞ - нормы, которые определяются равенством $k_p(A)=\| A\|_p\| A^{-1}\|_p, 1\le p\le\infty$.

Пример. Пусть компоненты вектора b заданы с точностью до 4- го знака, то есть $\frac{\| \delta b\|}{\| b\|}\approx 10^{-4}$. Если $k(A)\le 10^2$, то имеем $\frac{\| \delta x\|}{\| x\|}\approx 10^{-2}$. Однако, если $k(A)\approx 10^4$, то получим $\frac{\| \delta x\|}{\| x\|}\approx 1$. То есть в этой задаче граница между плохо и хорошо обусловленными матрицами определяется соотношением $10^2\le k_{gr}(A)<10^4$

Пример. Пусть $A=\begin{pmatrix}1000 & 999\\ 999 & 998\end{pmatrix}$. Соответственно $A^{-1}=\begin{pmatrix}-998 & 999\\ 999 & -1000\end{pmatrix}$. Отсюда найдем $k_1(A)=k_\infty(A)=1999^2=3.996\times 10^6$. Соответственно $k_2(A)\approx 3.992\times 10^6$. То есть матрица A будет плохо обусловленной для большинства норм.

В пакете MatLab для вычисления числа обусловленности используется подпрограмма $cond(A,i), i$ - номер нормы.

Определим максимальный и минимальный коэффициенты растяжения для оператора A с помощью следующих соотношений $Mag_{\max}(A) = \max_{x \neq 0} \frac{\| Ax \|}{\| x \|}$ и $Mag_{\min}(A) = \min_{x \neq 0} \frac{\| Ax \|}{\| x \|}$.

Очевидно, что коэффициент максимального растяжения совпадает с индуцированной матричной нормой $\| A \|$. Тогда можно показать, что число обусловленности равно: $k(A) = \frac{Mag_{\max}(A)}{Mag_{\min}(A)}$ и, оно будет определено для всех невырожденных матриц A. То есть $k(A)$ - есть просто отношение максимального растяжения к минимальному .

Рассмотрим теперь влияние возмущений матрицы A, то есть $(A + \delta A)\hat{x} = b$. Условия, гарантирующие, что такая возмущенная система имеет единственное решение, определяются следующим утверждением.

Теорема */Уоткинс с149/. Если матрица* A *является невырожденной и выполняется следующее неравенство* $\frac{\| \delta A \|}{\| A \|} < \frac{1}{k(A)}$, *то и* $A + \delta A$ *невырожденная.*

То есть число обусловленности $k(A)$ дает представление о расстоянии между матрицей A и ближайшей вырожденной матрицей: если $A + \delta A$ вырождена, то значение $\frac{\| \delta A \|}{\| A \|}$, по крайней мере, не меньше $\frac{1}{k(A)}$. Если матрица A невырожденная и выполняются следующие соотношения $\frac{\| \delta A \|}{\| A \|} < \frac{1}{k(A)}$, $b \neq 0$, $Ax = b$ и $(A + \delta A)(x + \delta x) = b$, то можно записать следующее неравенство

$$\frac{\| \delta x \|}{\| x \|} \leq \frac{k(A) \frac{\| \delta A \|}{\| A \|}}{1 - k(A) \frac{\| \delta A \|}{\| A \|}}$$

Если матрица A хорошо обусловлена и величина $\frac{\| \delta A \|}{\| A \|}$ достаточно мала, то можно получить следующую оценку $\frac{\| \delta x \|}{\| x \|} \leq k(A) \frac{\| \delta A \|}{\| A \|}$.

Рассмотрим теперь случай, когда возмущения воздействуют и на вектор b и на матрицу A.

Теорема*. Если матрица* A *невырожденная и выполняются следующие соотношения* $\frac{\| \delta A \|}{\| A \|} < \frac{1}{k(A)}$, $b \neq 0$, $Ax = b$ *и* $(A + \delta A)(x + \delta x) = b + \delta b$, *то справедливо следующее неравенство*

$$\frac{\| \delta x \|}{\| x \|} \leq \frac{k(A)\left(\frac{\| \delta A \|}{\| A \|} + \frac{\| \delta b \|}{\| b \|}\right)}{1 - k(A) \frac{\| \delta A \|}{\| A \|}}.$$

Сингулярное разложение (SVD-singular value decomposition).

Пусть матрица $A \in \Re^{n\times m}; n,m>0$. Определим область значений матрицы A как подпространство в $\Re^n$, определяемое соотношением $F(A)=\{Ax \mid x \in \Re^m\}$. Очевидно, что ранг матрицы A - это размерность подпространства $F(A)$.

Теорема (SVD-теорема) */Уоткинс с283/. Пусть* $A \in \Re^{n\times m}$ *- ненулевая матрица ранга* r *. Тогда ее можно представить в виде произведения* $A = U \cdot \Sigma \cdot V^T$, *где* $U \in \Re^{n\times n}$ *и* $V \in \Re^{m\times m}$ *ортогональные, а* $\Sigma \in \Re^{n\times m}$ *- неквадратная «диагональная» матрица вида*

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & \dots & \dots & .. \\ 0 & .. & .. & .. & .. \\ 0 & \dots & \sigma_r & 0 & .. \\ \dots & & & 0 & .. \\ 0 & 0 & \dots & .. & 0 \end{pmatrix},\ \sigma_1 \ge \sigma_2 \ge \dots \ge \sigma_r > 0$$

Матрицы U, V обладают свойствами только частичной однозначности, элементы $\sigma_1 \ge \sigma_2 \ge \dots \ge \sigma_r$ матрицы Σ определены однозначно и называются сингулярными числами матрицы A. Столбцами матрицы U являются ортонормированные векторы, называемые левыми сингулярными векторами матрицы A, а столбцы матрицы V называются правыми сингулярными векторами.

Теорема (геометрическая интерпретация SVD – теоремы)*. Пусть* $A \in \Re^{n\times m}, m<n$ *- ненулевая матрица ранга* r *. Тогда в пространстве* $\Re^m$ *найдется ортонормированный базис* $v_1, v_2, \dots, v_m$, *а в пространстве* $\Re^n$ *найдется ортонормированный базис* $u_1, u_2, \dots, u_m$. *Кроме того, существуют числа* $\sigma_1 \ge \sigma_2 \ge \dots \ge \sigma_r > 0$ *такие, что:*

$$Av_i = \begin{cases} \sigma_i u_i, i = 1,2,\dots r \\ 0, i = r+1,\dots,m \end{cases} \text{ и } A^T u_i = \begin{cases} \sigma_i v_i, i = 1,2,\dots r \\ 0, i = r+1,\dots,n \end{cases}.$$

SVD – разложение дает ортонормированные базисы для четырех базовых подпространств $F(A) = span\{u_1,\dots,u_r\}, N(A) = span\{v_{r+1},\dots v_m\}$,
$F(A^T) = span\{v_1,\dots,v_r\}, N(A^T) = span\{u_{r+1},\dots v_n\}$.

Следствие*. Пусть* $A \in \Re^{n\times m}$ *ненулевая матрица ранга* r, *а* $\sigma_1 \ge \sigma_2 \ge \dots \ge \sigma_r > 0$ *ее сингулярные числа с правыми и левыми сингулярными векторами* $v_1, v_2, \dots, v_r$ *и* $u_1, u_2, \dots, u_r$ *соответственно. Тогда имеет место соотношение* $A = \sum_{j=1}^{r} \sigma_j u_j v_j^T$.

В пакете MatLab для вычисления, как сингулярных чисел, так и сингулярного разложения матриц можно воспользоваться командой *svd* .

Рассмотрим связь спектральной нормы матрицы $\| A\|_2$ и числа обусловленности $k_2(A)$ с сингулярным ее разложением. Вспомним, что спектральная норма определена, как индуцированная евклидовой векторной нормой: $\| A\|_2 = \max\limits_{x\neq 0} \dfrac{\| Ax\|_2}{\| x\|_2}$. Это определение имеет смысл и для неквадратных матриц.

***Теорема**. Пусть матрица $A \in \Re^{n\times m}$ имеет сингулярные числа $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0$. Тогда $\| A\|_2 = \sigma_1$.*

***Следствие**. Пусть $A \in \Re^{n\times n}$ - невырожденная матрица с сингулярными числами $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n > 0$. Тогда $k_2(A) = \dfrac{\sigma_1}{\sigma_n}$.*

Задача численного определения ранга матрицы.

При отсутствии в данных ошибок округления и различных неопределенностей сингулярное разложение позволяет определить ранг матрицы. К сожалению, наличие ошибок, делает такую возможность определения ранга достаточно сложной.

Пример. Пусть задана матрица $A = \begin{pmatrix} 1/3 & 1/3 & 2/3 \\ 2/3 & 2/3 & 4/3 \\ 1/3 & 2/3 & 3/3 \\ 2/5 & 2/5 & 4/5 \\ 3/5 & 1/5 & 4/5 \end{pmatrix}$. Матрица A имеет ранг 2, так как третий столбец является суммой первых двух. Однако, если использовать стандартную функцию $rank(A)$, например пакета MatLab, то из-за ошибок округления, получим, что матрица будет иметь ранг 3. Если теперь использовать SVD разложение в арифметике с плавающей точкой, то получим $\sigma_1 = 2.5987; \sigma_2 = 0.3682; \sigma_3 = 8.6614 \cdot 10^{-17}$. То есть ранг матрицы опять будет равен 3. Однако, одно из сингулярных чисел мало. Поэтому имеет смысл считать его равным нулю.

Рассмотренный пример показывает, что имеет смысл введения понятия численного ранга. То есть, если матрица имеет k «больших» сингулярных чисел, а остальные сингулярные числа «малы», то имеет смысл считать ее численный ранг равным k. Для этого задают пороговое значения ε, определяющее уровень неопределенности данных в матрице. Обычно в стандартных процедурах MatLab и MatCad используются некоторые, принятые по умолчанию, пороговые значения, которые ограничивают неопределенности в задании исходных данных. При необходимости эти значения могут быть изменены пользователем.

Пусть матрица $A \in \Re^{n\times m}$ имеет ранг $r < \min\{n, m\}$. Тогда можно показать, что для $\forall \varepsilon > 0$ существует матрица полного ранга $A_\varepsilon \in \Re^{n\times m}$ ($rank(A_\varepsilon) = \min\{n, m\}$), такая, что $\| A - A_\varepsilon\|_2 < \varepsilon$. То есть, каждой матрице A неполного ранга соответствует полноранговая матрица A_ε, сколь угодно близкая к ней.

***Теорема**. Пусть матрица $A \in \Re^{n\times m}$, имеет полный ранг, то есть $rank(A) = r = \min\{n,m\}$. Пусть $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0$ - сингулярные числа матрицы A, а матрица $B \in \Re^{n\times m}$ удовлетворяет условию $\| A - B \|_2 < \sigma_r$. Тогда матрица B также имеет полный ранг.*

Таким образом, если матрица A имеет полный ранг, то все матрицы, достаточно близкие к A, также имеют полный ранг. То есть, множество матриц полного ранга есть открытое плотное подмножество в пространстве $\Re^{n\times m}$. Его дополнение, множество матриц неполного ранга, является, следовательно, замкнутым и нигде не плотным. Тем не менее, имеет смысл говорить, что матрица имеет неполный численный ранг, если она близка к неполноранговой матрице с точностью до небольшого возмущения ε. То есть численный ранг матрицы A будет равен k тогда и только тогда, когда $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_k >> \varepsilon \geq \sigma_{k+1} \geq ....$

<u>Задача наименьших квадратов и SVD разложение.</u>

Пусть матрица $A \in \Re^{n\times m}$, $r = rank(A)$ и $b \in \Re^n$. Рассмотрим систему уравнений $Ax = b$ с неизвестным $x \in \Re^m$. Если $n > m$, то система переопределена и нельзя, вообще говоря, найти единственное решение. Исходя из этого, обычно, ищется вектор $x \in \Re^m$, который минимизировал бы $\| b - Ax \|_2$. Такая постановка поиска решения называется задачей наименьших квадратов.

Решение задачи наименьших квадратов также не всегда единственно. Поэтому рассматривается дополнительная задача: из всех векторов $x \in \Re^m$, обеспечивающих минимум нормы $\| b - Ax \|_2$ найти тот, у которого $\| x \|_2$ минимальна.

Пусть известно точное $SVD(A) = U \cdot \Sigma \cdot V^T$, где $U \in \Re^{n\times n}$ и $V \in \Re^{m\times m}$ ортогональны, а матрица $\Sigma = \begin{pmatrix} \hat{\Sigma} & 0 \\ 0 & 0 \end{pmatrix}, \hat{\Sigma} = diag\{\sigma_1,...,\sigma_r\}$, $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0$. Поскольку U ортогональна, то $\| b - Ax \|_2 = \| U^T(b - Ax) \|_2 = \| U^T b - \Sigma(V^T x) \|_2$. Введем обозначение $c = U^T b = \begin{pmatrix} \hat{c} \\ d \end{pmatrix}$ и $y = V^T x = \begin{pmatrix} \hat{y} \\ z \end{pmatrix}$, где $\hat{c}, \hat{y} \in \Re^r$. Тогда $c - \Sigma y = \begin{pmatrix} \hat{c} \\ d \end{pmatrix} - \begin{pmatrix} \hat{\Sigma} & 0 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} \hat{y} \\ z \end{pmatrix} = \begin{pmatrix} \hat{c} - \hat{\Sigma}\hat{y} \\ d \end{pmatrix}$. Отсюда найдем $\| b - Ax \|_2^2 = \| c - \Sigma y \|_2^2 = \| \hat{c} - \hat{\Sigma}\hat{y} \|_2^2 + \| d \|_2^2$.

Это выражение принимает минимальное значение тогда и только тогда, когда $\hat{y} = \hat{\Sigma}^{-1}\hat{c}$, то есть $y_i = c_i / \sigma_i, i = 1,2,...,r$. Очевидно, что вектор z может быть выбран произвольным образом, но решение с минимальной нормой $\| x \|_2$ получим при $z = 0$. При этом норма минимальной невязки равна $\| d \|_2$. Алгоритм вычисления решения можно описать следующим образом.

1. Вычисляем $SVD(A) = U \cdot \Sigma \cdot V^T$.
2. Полагаем $\hat{y} = \hat{\Sigma}^{-1}\hat{c}$.
3. Если $r < m$, то выбираем произвольным образом вектор $z \in \Re^{m-r}$ (для минимизации решения, полагаем $z = 0$).

4. Полагаем $y = \begin{pmatrix} \hat{y} \\ z \end{pmatrix} \in \Re^m$.

5. Полагаем $x = Vy$.

На практике ранг матрицы A точно не известен. В этом случае, обычно пользуются понятием численного ранга. То есть, все «малые» сингулярные числа необходимо положить равными нулю. То есть для вычисления вектора c используется только r первых столбцов матрицы U. Если ищется лишь решение с минимальной нормой, то достаточно вычислить только r первых столбцов матрицы V.

Псевдообращение матриц.

Псевдообращение, или обобщенное обращение Мура-Пенроуза, является обобщением обычного обращения матрицы. Любая матрица $A \in \Re^{n \times m}$ имеет псевдообратную. Точно так же, как и решение квадратной невырожденной системы $Ax = b$ можно выразить через обратную матрицу A^{-1} в виде $x = A^{-1} \cdot b$, решение с минимальной нормой задачи наименьших квадратов с матрицей коэффициентов $A \in \Re^{n \times m}$ можно записать с помощью псевдообратной матрицы A^+ в виде $x = A^+ \cdot b$.

Для матрицы (оператора) $A \in \Re^{n \times m}$ ранга r действие матрицы A полностью описывается диаграммой:

$$A: \begin{cases} v_1 \xrightarrow{\sigma_1} u_1 \\ v_2 \xrightarrow{\sigma_2} u_2 \\ \dots \\ v_r \xrightarrow{\sigma_r} u_r \end{cases}; \quad \begin{matrix} v_{r+1} \\ \dots \\ v_m \end{matrix} \Big\} \to 0$$

Здесь $v_1, v_2, .., v_m$ и $u_1, u_2, .., u_n$ - полные ортонормированные множества правых и левых сингулярных векторов соответственно, а $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r > 0$ - ненулевые сингулярные числа матрицы A. В матричной форме $A = U \cdot \Sigma \cdot V^T$. Введем псевдообращение матрицы A, как матрицу $A^+ \in \Re^{n \times m}$, однозначно определяемую диаграммой:

$$A^+: \begin{cases} u_1 \xrightarrow{\sigma_1^{-1}} v_1 \\ u_2 \xrightarrow{\sigma_2^{-1}} v_2 \\ \dots \\ u_r \xrightarrow{\sigma_r^{-1}} v_r \end{cases}; \quad \begin{matrix} u_{r+1} \\ \dots \\ u_m \end{matrix} \Big\} \to 0$$

Очевидно, $rank(A^+) = rank(A)$. Кроме того, сужения операторов $A: span\{v_1, \dots, v_r\} \to span\{u_1, \dots, u_r\}$ и $A^+: span\{u_1, \dots, u_r\} \to span\{v_1, \dots, v_r\}$ являются

настоящими обращениями друг для друга. Чтобы понять, как матрица A^+ выглядит, в общем случае, заметим, что выражения $A^+u_i = \begin{cases} v_i\sigma_i^{-1}, i=1,...,r \\ 0, i=r+1,...,n \end{cases}$ можно представить в виде простого матричного соотношения $A^+ = V\cdot\Sigma^+\cdot U^T$. Это и есть SVD матрицы A^+ в матричной форме.

Если считать, что очень малые сингулярные числа отличны от нуля, то псевдообратную матрицу можно определить с помощью следующих рассуждений.

Пусть $A\in\Re^{n\times m}$ и $B=A^+\in\Re^{m\times n}$. Тогда псевдообратную матрицу можно определить с помощью следующих соотношений: $B\cdot A\cdot B=B$, $A\cdot B\cdot A=A$, $(B\cdot A)^T=B\cdot A$, $(A\cdot B)^T=A\cdot B$.

Вычисление функций от матриц. Матричная экспонента.

Термин “матричная функциях” может определяться по разному. Будем рассматривать определение матричной функции в следующем смысле. Пусть задана некоторая скалярная функция f и квадратная матрица $A\in\mathbb{C}^{n\times n}$ и будем определять $f(A)$ как матрицу такой же размерности, как матрица A.

Когда функция $f(t)$ является полиномом или дробно – рациональной функцией со скалярными коэффициентами и скалярным аргументом, естественно, для определения функции $f(A)$ просто подставить матрицу A вместо скаляра t. При этом, вместо деления на скаляр t использовать обращение матрицы. Кроме того, надо заменить скалярные константы α на αE, где E - единичная диагональная матрица, соответствующего размера. Например:

$$f(t)=\frac{1+t^2}{1-t}\Rightarrow f(A)=(E-A)^{-1}(E+A^2),$$

если $E\in\Lambda(A)$. Здесь $\Lambda(A)$ означает множество диагональных матриц, имеющих размер, равный размеру матрицы A. Так как матричная рациональная функция коммутативна, то не имеет значения, как будет записано выражение: $(E-A)^{-1}(E+A^2)$ или $(E+A^2)(E-A)^{-1}$. Если функция $f(t)$ допускает разложение в виде степенного ряда, например:

$$\log(1+t)=t-\frac{t^2}{2}+\frac{t^3}{3}-\frac{t^4}{4}+...,\ |t|<1,$$

то тогда можно подставит матрицу A вместо t:

$$\log(1+A)=A-\frac{A^2}{2}+\frac{A^3}{3}-\frac{A^4}{4}+...,\ \rho(A)<1.$$

Здесь $\rho(A)$ определяется, как спектральный радиус, и условие $\rho(A)<1$ необходимо для сходимости матричного ряда. Однако, при использовании такого подхода, имеется несколько трудностей:

- необходимо определить критерии, позволяющие оценить, для каких функций $f(t)$ возможна замена скалярного аргумента на матрицу;
- возможная замена скалярного аргумента на матрицу часто возможна только для очень ограниченного множества матриц, то есть необходимо каждый раз проверять, удовлетворяет ли матричный аргумент необходимым условиям;

- для многозначных функций, таких как логарифм или квадратный корень, необходимо каждый раз проверять все возможные значения $f(A)$, чтобы выделить подходящие величины.

Определение матричной функции с использование жордановой канонической формы.

Любую матрицу $A \in \mathbb{C}^{n\times n}$ можно преобразовать к жордановой канонической форме вида: $T^{-1}AT = J = diag(J_1, J_2, ..., J_p)$, где

$$J_k = J_k(\lambda_k) = \begin{pmatrix} \lambda_k & 1 & ... & ... & 0 \\ 0 & \lambda_k & ... & ... & 0 \\ . & . & ... & ... & ... \\ . & . & ... & ... & 1 \\ 0 & . & ... & ... & \lambda_k \end{pmatrix} \in \mathbb{C}^{m_k \times m_k},\ m_1 + m_2 + ... + m_p = n.$$

Напомним, что жорданова форма является единственной, по составу, жордановых ячеек (блоков) J_k, но преобразующая матрица T может быть не единственной.

Обозначим через $\lambda_1, \lambda_2, ..., \lambda_p$ различные собственные значения матрицы A и через m_k их соответствующие геометрические кратности, которые определяют размеры соответствующих жордановых ячеек. Пусть существует функция f, которая определена на спектре матрицы A, то есть все величины $f^{(j)}(\lambda_k)$, $j = 1, 2, ..., m_k - 1; k = 1, 2, ..., p$ существуют. Здесь $f^{(j)}(.)$ означает j - ую производную функции f.

Определение.

Пусть функция f определена на спектре матрицы $A \in \mathbb{C}^{n\times n}$ и пусть матрица A имеет следующую каноническую форму: $J = T^{-1}AT = diag(J_1, J_2, ..., J_p)$. Тогда:

$f(A) = Tf(J)T^{-1} = Tdiag(f(J_k))T^{-1}$, *где*

$$f(J_k) = \begin{pmatrix} f(\lambda_k) & f'(\lambda_k) & ... & \frac{f^{(m_k-2)}(\lambda_k)}{(m_k-2)!} & \frac{f^{(m_k-1)}(\lambda_k)}{(m_k-1)!} \\ 0 & f(\lambda)_k & ... & \frac{f^{(m_k-3)}(\lambda_k)}{(m_k-3)!} & \frac{f^{(m_k-2)}(\lambda_k)}{(m_k-2)!} \\ . & . & ... & ... & ... \\ 0 & 0 & ... & \frac{f''(\lambda_k)}{2!} & f'(\lambda_k) \\ 0 & 0 & ... & 0 & f(\lambda_k) \end{pmatrix} \in \mathbb{C}^{m_k \times m_k}$$

Пример.

Заданы функция $f(x)=x^3$ и жорданова ячейка $J=\begin{pmatrix} 1/2 & 0 \\ 0 & 1/2 \end{pmatrix}$. Отсюда найдем:

$$f(J)=J^3=\begin{pmatrix} f(\frac{1}{2}) & f'(\frac{1}{2}) \\ 0 & f(\frac{1}{2}) \end{pmatrix}=\begin{pmatrix} 1/8 & 3/4 \\ 0 & 1/8 \end{pmatrix}.$$

Запишем жорданову ячейку в виде: $J_k=\lambda_k E_{m_k}+I_{1,m_k}$, где I_{1,m_k} косая матрица размера m_k, с верхней диагональю в виде единиц. Заметим, что: $I_{m_k,m_k}=(I_{1,m_k})^{m_k}=0$.

Пример.

Задана косая матрица: $I_{1,3}=\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$. Тогда $I_{2,3}=I_{1,3}^2=\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$; $I_{3,3}=I_{1,3}^3=\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$.

Тогда функцию $f(J_k)$ можно представить в виде конечного ряда:

$$f(J_k)=f(\lambda_k)E+f'(\lambda_k)I_{1,m_k}+...+\frac{f^{(m_k-1)}(\lambda_k)}{(m_k-1)!}I_{m_k-1,m_k}$$

Элементарные матричные функции.

Полиномиальные матричные функции.

Любая квадратная матрица A может быть умножена сама на себя, то есть $A^mA^n=A^{m+n}$. Кроме того, если матрица A является несингулярной, то можно определить обратную матрицу A^{-1} с помощью соотношения $AA^{-1}=A^{-1}A=E$. Следовательно, можно записать соотношения: $A^0=A^{1-1}=AA^{-1}=E$ и $A^{-n}=(A^{-1})^n$. Таким образом, с помощью полученных соотношений, можно построить любую полиномиальную матричную функцию.

Пример.

Пусть $f(x)=x^2+5x+4$ и $A=\begin{pmatrix} 1 & 1 \\ 2 & 3 \end{pmatrix}$. Тогда: $f(A)=A^2+5A+4E$, или:

$$f(A)=\begin{pmatrix} 1 & 1 \\ 2 & 3 \end{pmatrix}^2+5\cdot\begin{pmatrix} 1 & 1 \\ 2 & 3 \end{pmatrix}+4\cdot\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}=\begin{pmatrix} 12 & 9 \\ 18 & 30 \end{pmatrix}.$$

Рассмотрим теперь случай полинома $f(\lambda)$ с комплексными коэффициентами: $a_n, a_{n-1}, ..., a_0$:

$$f(\lambda)=a_n\lambda^n+a_{n-1}\lambda^{n-1}+...+a_1\lambda+a_0.$$

Тогда можно определить матричный полином $f(A)$ для матрицы $A \in \mathbb{C}^{n\times n}$ как:

$$f(A) = a_n A^n + a_{n-1}A^{n-1} + ... + a_1 A + a_0 E .$$

Матричная экспонента.

Пусть A имеет размер $n \times n$ и является вещественной или комплексной матрицей. Экспонециал матрицы A, обозначаемый как матрица e^A, является матрицей размера $n \times n$, которая определяется матричным степенным рядом: $e^A = E + \sum_{k=1}^{\infty} \frac{A^k}{k!}$, полученным из функции $f(z) = e^z$, которая является аналитической в $\mathbb{C}$. Можно также записать:

$$\cos(A) = E + \sum_{k=1}^{\infty} (-1)^k \frac{A^{2k}}{(2k)!}; \ \sin(A) = E + \sum_{k=1}^{\infty} (-1)^{k-1} \frac{A^{2k-1}}{(2k-1)!} .$$

Тогда, на основе формулы Эйлера получим:

$$e^{i\cdot A} = \cos(A) + i \cdot \sin(A) .$$

Утверждение 1.

Пусть A и P являются комплексными матрицами размера $n \times n$, и пусть матрица P является обратимой. Тогда: $e^{P^{-1}AP} = P^{-1}e^A P$.

Доказательство.

Если $m \geq 0$, то можно записать: $(P^{-1}AP)^m = P^{-1}A^m P$. Тогда:

$$e^{P^{-1}AP} = P^{-1}(E + A + \frac{A^2}{2!} + ...)P = P^{-1}e^A P .$$

Утверждение 2.

Пусть $A \in \mathbb{C}^{n\times n}$. Тогда имеют место следующие соотношения:

- если 0 обозначает нулевую матрицу соответствующего размера, то тогда $e^0 = E$, где: $e^0 = E$ единичная диагональная матрица;
- $A^m e^A = e^A A^m$ - для любого целого m;
- $(e^A)^L = e^{A^L}$;
- если $AB = BA$, то тогда $Ae^B = e^B A$ и $e^{A+B} = e^A e^B = e^B e^A$.

Следует учесть, что если матрицы A и B не являются коммутативными, то в общем случае: $e^{A+B} \neq e^A e^B$.

Утверждение 3.

Пусть $A \in \mathbb{C}^{n\times n}$ и заданы величины $s, t \in \mathbb{C}$. Тогда: $e^{A(s+t)} = e^{As} e^{At}$.

Доказательство.

$$e^{As}e^{At} = (E + As + \frac{A^2s^2}{2!} + ...)(E + At + \frac{A^2t^2}{2!} + ...) = \sum_{j=0}^{\infty}\sum_{k=0}^{\infty}\frac{A^{j+k}s^j t^k}{j!k!}.$$

Обозначим $n = j + k$, тогда $j = n - k$. Отсюда получим:

$$e^{As}e^{At} = \sum_{n=0}^{\infty}\sum_{k=0}^{\infty}\frac{A^n s^{(n-k)}t^k}{(n-k)!k!} = \sum_{n=0}^{\infty}\frac{A^n}{n!}\sum_{k=0}^{\infty}\frac{n!}{(n-k)!K!}s^{(n-k)}t^k = \sum_{n=0}^{\infty}\frac{A^n(s+t)^n}{n!} = e^{A(s+t)}.$$

Утверждение 4.

Пусть $A \in \mathbb{C}^{n\times n}$ и задана вещественная величина $t \in \Re$. Определим $f(t) = e^{At}$. Тогда:

$f'(t) = Ae^{At}$.

Доказательство.

$$f'(t) = \lim_{h\to 0}\frac{e^{A(t+h)} - e^{At}}{h} = e^{At}(\lim_{h\to 0}\frac{e^{Ah} - E}{h}) = e^{At}(\lim_{h\to 0}\frac{1}{h}[Ah + \frac{A^2h^2}{2!} + ...]) = e^{At}A = Ae^{At}.$$

Матричные функции в среде Matlab.

$\exp m$ - матричная экспонента e^A;

$\log m$ - матричный логарифм $\log m(A)$ - определяется как главная часть логарифма от мнимого числа и имеет аргумент в диапазоне от $-\pi$ до π;

$sqrtm$ - положительный корень квадратный из матрицы $\sqrt{A} = A^{1/2}$;

$mpower$ - степень матрицы X^Y (если оба аргумента X и Y являются матрицами, то среда формирует ошибку).