DOI 10.37539/2949-1991.2025.28.5.041
УДК 519.623

Щипицын Анатолий Георгиевич,
доктор технических наук, профессор,
эксперт Российской Академии Естествознания,
г. Москва

ПРАКТИЧЕСКИ УДОБНЫЙ МЕТОД ОБРАЩЕНИЯ МАТРИЦЫ

Аннотация: Предложен точный метод обращения матрицы произвольного размера с использованием алгоритма решения системы алгебраических уравнений. Приведён текст исходного модуля программы обращения матрицы с определением средней арифметической абсолютной вычислительной погрешности компонент обратной матрицы, а также указан способ анализа влияния близкого к нулю определителя исходной матрицы на погрешность вычисления обратной матрицы.

Ключевые слова: исходная матрица, обратная матрица, обращение матрицы, система линейных алгебраических уравнений, программа.

Введение

Вообще говоря, методы обращения матрицы с позиций математических описаний разработаны и сводятся к двум основным: точным (например, [1, 2, 3]), погрешности которых зависят только от характеристик компьютера, и итерационным (например, [4]), погрешности которых являются методическими и зависят, в частности от количества итераций приближения к решениям и, естественно, ещё и от характеристик компьютера. Поэтому приоритетными являются точные методы, один их которых и подлежит дальнейшему рассмотрению.

Указанное в заголовке статьи словосочетание «практически удобный метод» следует понимать так, что в данном случае описанный метод удобен при необходимости использования его для решения какой-либо сложной задачи и когда «под рукой» имеется разработанная ранее программа решения системы линейных алгебраических уравнений и «остаётся только добавить» блок обращения матрицы к этой программе. Такую «добавку» пользователь выбирает по своему критерию удобства, например, эрудицией во владении математическим аппаратом. В данном случае относительно сложная задача - это определение оценок переменных состояния динамической системы процедурой линейной дискретной оптимальной фильтрации, для реализации которой необходима вложенная процедура обращения матрицы большого размера. Указанное выше «практическое удобство» заключается в данном случае в разработанном и отлаженном конкретном исходном модуле программы обращения матрицы, который приведён в конце текста статьи и к которому даны комментарии.

Математическое описание

Исходная, то есть обращаемая квадратная невырожденная матрица A размера nxn имеет компоненты:

$$a_{ij}\ ,\ i,j=1,\dots n. \tag{1}$$

Задача заключается в определении матрицы X, обратной к матрице A, то есть выполнить операцию

$$X=A^{-1} \tag{2}$$

Для решения этой задачи используем матричное уравнение

$$XA=E, \tag{3}$$

где E - единичная матрица размера nxn, X - матрица размера nzn с компонентами:

$$x_{ij}, \ i,j=1,\dots n, \tag{4}$$

которые подлежат определению путём решения уравнения (3). Запишем матричное уравнение (3) в скалярной форме:

$$\sum_{j=1}^{n} x_{ij}a_{jk} = D_{ik}, \ i,k=1,\dots n, \tag{5}$$

где D_{ik} - символ Кронеккера [5]:

$$D_{ik} = 1, \text{ если } i=k; \ D_{ik} = 0, \text{ если } i \neq k. \tag{6}$$

Преобразуем систему уравнений (5) к эквивалентной системе линейных алгебраических уравнений, для чего введём переобозначения для неизвестных:

$$x^{c}_{n(i-1)+j} = x_{ij}, \ i,j=1,\dots n, \tag{7}$$

переобозначения:

$$a^{c}_{[I+(k-1)n],[j+(k-1)n]} = a^{T}_{ij}, \ i,j,k=1,\dots n, \tag{8}$$

$$b_{n(i-1)+k} = D_{ik}, \ i,k=1,\dots n \tag{9}$$

В правых частях выражений (8) введены обозначения a^{T}_{ij} компонент транспонированной матрицы для матрицы A, то есть:

$$a^{T}_{ij} = a_{ji}, \ i,j=1,\dots n. \tag{10}$$

Использовав переобозначения (7), (8), (9), перепишем систему (5) в матричной форме

$$A^{c}X^{c} = B, \tag{11}$$

где A^{c} - матрица размера NxN с ненулевыми компонентами (8), X^{c} - матрица-столбец размером Nx1 с компонентами (7), B - матрица-столбец размером Nx1с единицами и нулями, стоящими на местах согласно символу Кронеккера (6), где

$$N = n^2. \tag{12}$$

Скалярный эквивалент матричной записи (11) имеет вид

$$\sum_{q=1}^{N} a^{c}_{pq}x^{c}_{q} = b_{q}, \ p = 1,\dots,N. \tag{13}$$

Решая систему (13) относительно x^{c}_{q} и возвращаясь к прежним обозначениям (7), получаем компоненты обратной матрицы для матрицы A:

$$x^{*}_{ij} = x^{c}_{n(i-1)+j}, \ i,j=1,\dots n. \tag{14}$$

Путём подстановки полученных решений x^{*}_{ij} в (5), проверяем их правильность: если каждое их уравнений системы (5) обращается в тождество с точностью до вычислительных погрешностей, то решения x^{*}_{ij} правильные. А чтобы найти оценку вычислительных погрешностей, составим выражения:

$$D^{R}_{ik} = \sum_{j=1}^{n} x^{*}_{ij}a_{jk}, \ i,k=1,\dots n, \tag{15}$$

$$D^{R}_{0} = (1/N)\sum_{j,k=1}^{n} |D^{R}_{ik} - D_{jk}| \, . \tag{16}$$

Принимая в качестве меры точности решения задачи среднюю арифметическую допустимую величину абсолютной погрешности D^*, проверяем выполнение условия

$$D^{R}_{0} \leq D^*. \tag{17}$$

Если это условие выполняется, то решения задачи поучены с требуемой точностью. А если не выполняется, то возможны три варианта: 1) если величина D^{R}_{0} выходит за пределы допустимого для компьютера наибольшего значения, то есть имеет место переполнение, то причину следует искать в вырожденности исходной матрицы, при котором определить этой матрицы либо равен нулю, либо близок к нулю; 2) если переполнения не происходит, но величина $D^{R}_{0} > D^*$, то это означает, что решения не правильные и ошибки следует искать в операторах программы; 3) если порядок величины D^{R}_{0} соизмерим с порядком величины D^*, но условие (17) не выполняется, то это означает, что компьютерная погрешность больше требуемой допустимой погрешности и в этом варианте остаётся только уменьшить D^*, то есть ослабить требования к точности решений.

Программа обращения матрицы

Программа разработана в среде PascalABC.NET 3.8 [6]. В таблице приведены только входные и выходные переменные математического описания и соответствующие им идентификаторы программы. После таблицы даны комментарии к программе, после которых приведён её исходный модуль.

Таблица 1

	Входная информация				Выходная информация		
Символ текста	n	N	a_{ij}, i,j=1,…n	D^*	x^*_{ij}, i,j=1,…n	----	D^R_0
Идентификатор программы	NN	N	AV[I,J]	DZ	XV[I,J] NN	Определитель исходной матрицы D0	DD0

После описания меток, типов переменных и ввода исходной информации: допустимая вычислительная погрешность, символ Кронеккера, размер исходной матрицы (для примера), компоненты исходной матрицы (для примера), введены операторы с указанными слева метками, которые выполняют следующие действия:

1: вычисление компонент транспонированной матрицы (10);
2,3,4: преобразования согласно формул (8), (9);
4,...,12: решение системы (13);
13: вычисление определителя исходной матрицы;
14: проверка правильности решений системы (13);
15: вычисление компонент обратной матрицы согласно формул (14);
16,17,18: проверка правильности вычислений компонент обратной матрицы согласно формул(15), (16);
19,20: вывод компонент обратной матрицы;
21: вывод определителя исходной матрицы;
22: вывод сообщения об удовлетворении или неудовлетворении условия (17).

Текст исходного модуля программы:

```
PROGRAM AG2511;
LABEL 0,1,2,3,4,5,6,7,8,9,10, 11,12,13,14,15,16,17,18 ,19,20,21,22,23;
VAR N,NN,I,J,K,P,Q:INTEGER;
VAR DZ,D0,DD0,D20:REAL;
VAR D,AV,AT:ARRAY[1..30,1..30]OF REAL;
VAR XV,DD:ARRAY[1..30,1..30]OF REAL;
VAR AA,B:ARRAY[0..900,1..900]OF REAL;
VAR BB.X,F:ARRAY[0..900]OF REAL;
VAR A:ARRAY[0..900,1..900,1..900]OF REAL;
VAR LST:TEXT;
BEGIN
ASSIGN(LST,'AG251.DOCX');
REWRITE(LST);
WRITELN(LST,'SOURCE INFORMATION AG2511');
DZ:=1E-12;WRITELN(LST,'0.0__ COMPUTATIONAL ERROR DZ=',DZ);
WRITELN(LST,'0.2___THE KRONECKER SYMBOL:');
FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
```

```
BEGIN
D[I,J]:=0;
END;
FOR I:=1 TO NN DO
BEGIN
D[I,I]:=1;
END;
NN:=6;WRITELN(LST,'0.3___THE SIZE OF THE ORIGINAL MATRIX NN=',NN);
N:=SQR(NN);WRITELN(LST,'THE SIZE OF THE EXTENDED MATRIX N=',N);
WRITELN(LST,'0.4___ELEMENTS OF THE ORIGINAL MATRIX:');
AV[1,1]:=1; AV[1,2]:=2; AV[1,3]:=3; AV[1,4]:=0.1; AV[1,5]:=0.2; AV[1,6]:=0.3;
AV[2,1]:=0.4; AV[2,2]:=2; AV[2,3]:=0.5; AV[2,4]:=0.6; AV[2,5]:=0.7; AV[2,6]:=0.8;
AV[3,1]:=0.5; AV[3,2]:=0.4; AV[3,3]:=0.3; AV[3,4]:=0.2; AV[3,5]:=0.1; AV[3,6]:=0;
AV[4,1]:=0.1; AV[4,2]:=0.2; AV[4,3]:=0.3; AV[4,4]:=4; AV[4,5]:=0.4; AV[4,6]:=0.5;
 AV[5,1]:=9; AV[5,2]:=8; AV[5,3]:=7; AV[5,4]:=6; AV[5,5]:=5; AV[5,6]:=4;
 AV[6,1]:=0.9; AV[3,2]:=0.8; AV[6,3]:=0.7; AV[6,4]:=0.6; AV[6,5]:=0.7; AV[6,6]:=6;
WRITELN(LST,'THE ORIGINAL SIZE MATRIX     ' , NN,' X ',NN);
WRITELN(LST);
FOR I:=1 TO NN DO
BEGIN
WRITELN(LST,AV[I,1]:6:2,'  ',AV[I,2]:6:2,'  ',AV[I,3]:6:2,'  ',
AV[I,4]:6:2,'  ',AV[I,5]:6:2,'  ',AV[I,6]:6:2);
END;
WRITELN(LST,'RESULTS OF AG2511');
1:FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
BEGIN
AT[I,J]:=AV[J,I];
END;
2:FOR P:=1 TO N DO
FOR Q:=1 TO N DO
BEGIN
AA[P,Q]:=0;
END;
FOR P:=1 TO N DO
BEGIN
BB[P]:=0;
END;
3:FOR K:=1 TO NN DO
FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
BEGIN
AA[((K-1)*NN+I),((K-1)*NN+J)]:=AT[I,J];
END;
4:FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
BEGIN
BB[(I-1)*NN+J]:=D[I,J];
```

```
END;
5:FOR P:=1 TO N DO
FOR Q:=1 TO N DO
BEGIN
A[0,P,Q]:=AA[P,Q];
END;
FOR P:=1 TO N DO
BEGIN
B[0,P]:=BB[P];
END;
6:FOR K:=1 TO N-1 DO
FOR I:=K+1 TO N DO
FOR J:=K+1 TO N DO
BEGIN
A[K,I,J]:=A[K-1,I,J]-(A[K-1,I,K]/A[K-1,K,K])*A[K-1,K,J];
END;
7:FOR K:=1 TO N-1 DO
FOR I:=K+1 TO N DO
BEGIN
B[K,I]:=B[K-1,I]-(A[K-1,I,K]/A[K-1,K,K])*B[K-1,K];
END;
8:X[N]:=B[N-1,N]/A[N-1,N,N];
9:I:=N-1;
10:X[I]:=B[I-1,I]/A[I-1,I,I];
FOR J:=I+1 TO N DO
BEGIN
X[I]:=X[I]-(A[I-1,I,J]/A[I-1,I,I])*X[J];
END;
11:I:=I-1;
12:IF(I>=1)THEN GOTO 10;
13: D0:=1;
FOR I:=1 TO NN DO
BEGIN
D0:=D0*A[I-1,I,I];
END;
14:FOR I:=1 TO N DO
BEGIN
F[I]:=0;
END;
FOR I:=1 TO N DO
FOR J:=1 TO N DO
BEGIN
F[I]:=F[I]+A[0,I,J]*X[J];
END;
15:FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
BEGIN
XV[I,J]:=X[(I-1)*NN+J];
```

```
END;
16:FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
BEGIN
DD[I,J]:=0;
END;
17:FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
FOR K:=1 TO NN DO
BEGIN
DD[I,J]:=DD[I,J]+XV[I,K]*AV[K,J];
END;
DD0:=0;
  18:FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
BEGIN
DD0:=DD0+ABS(DD[I,J]-D[I,J]);
END;
DD0:=DD0/N;
19:FOR I:=1 TO NN DO
FOR J:=1 TO NN DO
BEGIN
WRITELN(LST,'XV[',I:2,' ',J:2,']=',XV[I,J]);
END;
WRITELN(LST);
20:WRITELN(LST,'THE INVERSE SIZE MATRIX    ' , NN,' X ',NN);
WRITELN(LST);
FOR I:=1 TO NN DO
BEGIN
WRITELN(LST,XV[I,1]:10:4,'  ',XV[I,2]:10:4,'  ',XV[I,3]:10:4,'  ',
XV[I,4]:10:4,'  ',XV[I,5]:10:4,'  ',XV[I,6]:10:4);
END;
21: WRITELN(LST,'DETERMINANT OF THE ORIGINAL MATRIX:');
WRITELN(LST,'D0=',D0:10);
WRITELN(LST,'DD0=',DD0,'     DZ=',DZ);
22:IF(DD0<=DZ)THEN
WRITELN(LST,'THE SOLUTIONS ARE ACCURATE')
ELSE  IF(DD0>DZ)THEN
WRITELN(LST,'THE SOLUTIONS ARE NOT ACCURATE');
23:CLOSE(LST);
END.
```

Пример вывода результатов работы программы:

```
PROGRAM AG2511___ ОБРАЩЕНИЕ МАТРИЦЫ
SOURCE INFORMATION AG2511
COMPUTATIONAL ERROR DZ=1E-12
THE SIZE OF THE ORIGINAL MATRIX NN=6
THE SIZE OF THE EXTENDED MATRIX N=36
THE ORIGINAL SIZE MATRIX     6 X 6
```

1.00	2.00	3.00	0.10	0.20	0.30
0.40	2.00	0.50	0.60	0.70	0.80
0.50	0.80	0.30	0.20	0.10	0.00
0.10	0.20	0.30	4.00	0.40	0.50
9.00	8.00	7.00	6.00	5.00	4.00
0.90	0.00	0.70	0.60	0.70	6.00

RESULTS OF AG2511

THE INVERSE SIZE MATRIX 6 X 6

-0.2184	-0.6989	1.5835	-0.0881	0.0732	0.0627
0.0078	0.3821	1.0149	0.0031	-0.0740	-0.0022
0.4134	-0.0771	-0.9973	0.0323	0.0148	-0.0230
-0.0074	-0.0823	0.3180	0.2709	-0.0162	-0.0005
-0.1957	0.8334	-3.6806	-0.2236	0.2142	-0.2255
0.0081	0.0248	0.2764	0.0084	-0.0361	0.1863

DETERMINANT OF THE ORIGINAL MATRIX:

D0= -198.4176

THE SOLUTIONS ARE ACCURATE

DD0=7.92009687000661E-17 DZ=1E-12

Заключение

1. Рассмотрен метод обращения матрицы, в основе которого использовано известное уравнение, выражающее равенство единичной матрице произведения исходной матрицы на обратную. Согласно рассматриваемому методу, это матричное уравнение преобразовано к эквивалентному матричному уравнению, в котором компоненты обратной матрицы записаны в виде столбца неизвестных, матрица коэффициентов представлена в виде блочной матрицы, на главной диагонали которой расположены транспонированные исходные матрицы, а остальные компоненты этой блочной матрицы - нули, правая часть эквивалентного матричного уравнения - это столбец, составленный из компонентов единичной матрицы, то есть состоящий из единиц и нулей согласно символу Кронеккера. Полученное матричное уравнение решается относительно своих неизвестных любым численным методом решения системы линейных алгебраических неизвестных.

2. На основе описанного метода разработана программа обращения матрицы, согласованная со вложенной в неё программой решения системы линейных алгебраических уравнений, вход в которую - это величина требуемой вычислительной точности, порядок и компоненты исходной матрицы, а её выход - это компоненты обратной матрицы, величина определителя исходной матрицы и величина вычислительной погрешности. Программа написана в среде PascalABC.NET 3.8 и в силу ограничений этой среды позволяет обращать матрицы до 30-го порядка включительно. В случае равенства нулю определителя исходной матрицы программа фиксирует в величине определителя переполнение и отсутствие решения задачи.

3. Численное моделирование на основе разработанной программы при задании исходной матрицы порядка 30 позволяет сделать следующие выводы: 1) при приближении определителя исходной матрицы к нулю вычислительная погрешность обращения матрицы увеличивается; 2) наиболее точно осуществляется обращение исходной матрицы, у которой диагональные и недиагональные компоненты отличаются друг от друга в пределах одного порядка величины; 3) текст исходного модуля программы для обращения исходной матрицы 30-го порядка занимает 41 кб, а время её обращения на стационарном компьютере с частотой 2,9 ГГц и 64-х разрядной сеткой составляет 18 секунд.

Список литературы:

1. Курош А.Г. Курс высшей алгебры.Издание 9-е. - М.: Издательство «Наука», 1968. - 432 с.

2. Бронштейн И.Н., Семендяев К.А. Справочник по математике для инженеров и учащихся втузов. - М.: Издательство «Наука», 1980. - 976 с.

3. Меньшова И.В. Матричная и векторная алгебра : учебно-методическое пособие / И. В. Меньшова, А. Н. Семакин Москва : Изд-во МГТУ им. Н. Э. Баумана, 2023. - 88 с.:: учебно-методическое пособие / И.В. Меньшова, А.Н.

4.Архипенков С.М. Метод последовательных приближений для обращения матриц больших размерностей // Модели и методы агроинформатики в научных исследованиях и рыночных условиях хозяйствования: Сб. науч. тр. / РАСХН. Сиб.Отделение. СибНИИЭСХ. – Новосибирск, 1992 – С. 182–187.

5.Лурье А.И Аналитическая механика. - М.: Физматгиз, 1961. - 824 с.

6. Осипов А.В. PascalABC.NET: Введение в современное программирование. - Ростов-на-Дону. Интернет издание, 2019. - 572 с.