УДК 519.6

МЕТОД *CR* – ФАКТОРИЗАЦИИ МАТРИЦ

С.Е. Саух
Институт проблем моделирования в энергетике им. Г.Е. Пухова
НАН Украины

Запропоновано метод стовпцево-рядкової CR – факторизації матриць, який відрізняється від методу LU – факторизації адаптивністю до множини динамічно обираємих провідних елементів, що дозволяє відмовитися від виконання дій з упорядкування рядків та стовпчиків та суттєво пришвидшити розв'язок систем алгебраїчних рівнянь великої розмірності.

Одним из наиболее часто используемых вариантов метода Гаусса является метод треугольного разложения или LU – факторизации [1, 2]. В этом случае матрица $\boldsymbol{A}$ представляется в виде произведения нижнетреугольной матрицы $\boldsymbol{L}$ и верхнетреугольной матрицы $\boldsymbol{U}$. Необходимыми условиями осуществимости LU – факторизации являются невырожденность матрицы $\boldsymbol{A}$ и неравенство нулю элементов u_{kk}. Осуществимости треугольного разложения невырожденной матрицы $\boldsymbol{A}$ всегда можно добиться переупорядочив ее строки и столбцы так, чтобы выполнялись условия $u_{kk} \neq 0$. Для этого используют матрицы перестановок $\boldsymbol{P}$ и $\boldsymbol{Q}$ такие, что $\boldsymbol{P} \cdot \boldsymbol{A} \cdot \boldsymbol{Q} = \boldsymbol{L} \cdot \boldsymbol{U}$. Особенности реализации метода Гаусса в условиях конечно-разрядных вычислений требуют такого выбора ведущих элементов a_{ij}, при котором ограничивается рост абсолютных значений элементов матриц $\boldsymbol{L}$ и $\boldsymbol{U}$ [3-8]. В случае разреженных матриц большой размерности во внимание принимается влияние перестановок на уровень заполненности ненулевыми элементами матриц $\boldsymbol{L}$ и $\boldsymbol{U}$, который может существенно превысить уровень заполненности ненулевыми элементами исходной матрицы $\boldsymbol{A}$. Поэтому выбор ведущих элементов основывается на сочетании требований как обеспечения численной устойчивости метода Гаусса, так и минимизации уровеня заполненности ненулевыми элементами матриц $\boldsymbol{L}$ и $\boldsymbol{U}$. В соответствии с этими требованиями в современных программных средствах реализуются различные стратегии выбора ведущих элементов [3-6].

Хранение ненулевых элементов разреженных матриц большой размерности в строчных или связанных списках существенно затрудняет перестановки строк и столбцов [3-6]. Заметим, что реализация перестановок не связана с выполнением фактических пересылок значений ненулевых элементов в памяти вычислительного устройства, а связана с выполнением операций поиска и изменения значений строчных и столбцовых индексов. Кроме того, в общем случае списки столбцовых индексов в строках и списки строчных индексов в столбцах подлежащих перестановкам не совпадают по длине, что усложняет реализацию изменений их индексов. Наши исследования показывают соизмеримость объемов операций необходимых для реализации перестановок и осуществления собственно $LU-$факторизации матриц. Мы предлагаем новый метод столбцово-строчной факторизации или, кратко, $CR-$факторизации, который принципиально отличается от метода $LU-$факторизации адаптивностью к выбору ведущих элементов, что позволяет вовсе отказаться от перестановок строк и столбцов в процессе формирования факторных матриц и ускорить процесс их вычисления.

Метод $\boldsymbol{CR}-$***факторизации матриц***. Базовым соотношением метода является выражение

$$\boldsymbol{A} = \boldsymbol{C}_j \boldsymbol{R}_i + \boldsymbol{A}\langle\langle i, j\rangle\rangle, \tag{1}$$

в котором исходная $n \times n$ матрица $\boldsymbol{A}$ представляется суммой произведения $n \times 1$ вектор-столбца $\boldsymbol{C}_j$ на $1 \times n$ вектор-строку $\boldsymbol{R}_i$ и $n \times n$ матрицы $\boldsymbol{A}\langle\langle i, j\rangle\rangle$.

Если элемент a_{ij} матрицы $\boldsymbol{A}$ не равен нулю, то, полагая равными нулю элементы $i-$й строки и $j-$го столбца матрицы $\boldsymbol{A}\langle\langle i, j\rangle\rangle$, из соотношения (1) с точностью до некоторого множителя можно определить все элементы столбца $\boldsymbol{C}_j$ и строки $\boldsymbol{R}_i$. Неоднозначность «с точностью до некоторого множителя» в определении элементов столбца $\boldsymbol{C}_j$ и строки $\boldsymbol{R}_i$ обусловлена невозможностью найти из одного уравнения вида $c_{ij} \cdot r_{ij} = a_{ij}$ две искомые величины c_{ij} и r_{ij}.

Для устранения неоднозначности в определении величин c_{ij} и r_{ij} следует принять одно из возможных допущений, например, вида

$$c_{ij} = 1, \tag{2}$$

$$r_{ij} = 1, \tag{3}$$

$$|c_{ij}| = |r_{ij}|. \tag{4}$$

Заметим, что допущения (2) и (3) аналогичны допущениям принимаемым в отношении диагональных элементов одной из факторных матриц $\boldsymbol{L}$ и $\boldsymbol{U}$ в методе $LU-$факторизации. Допущение вида (4) также имеет место в специальном варианте метода $LU-$факторизации симметричных положительно определенных матриц известного как метод LL^T- факторизации Холесского или метод квадратного корня [7, 8].

Соотношение (1) определяет частичную одношаговую столбцово-строчную факторизацию матрицы $\boldsymbol{A}$ относительно ведущего элемента a_{ij}. Здесь столбец $\boldsymbol{C}_j$ и строка $\boldsymbol{R}_i$ являются множителями или факторами, а матрица $\boldsymbol{A}\langle(i,j)\rangle$ является остаточной матрицей, элементы которой, кроме тех, что расположены в $i-$й строке и $j-$м столбце, образуют активную $(n-1)\times(n-1)$ подматрицу.

Анализируя структуру матрицы $\boldsymbol{A}\langle(i,j)\rangle$ в соотношении (1), замечаем, что найденные элементы столбца $\boldsymbol{C}_j$ и строки $\boldsymbol{R}_i$ естественно расположить в матрице $\boldsymbol{A}\langle(i,j)\rangle$ на месте нулевых элементов ее $i-$й строки и $j-$го столбца. Такое совмещение элементов столбца $\boldsymbol{C}_j$ и строки $\boldsymbol{R}_i$ с элементами матрицы $\boldsymbol{A}\langle(i,j)\rangle$ приводит к формированию совмещенной матрицы вида

$$\boldsymbol{A}_{CR}\langle(i,j)\rangle = \boldsymbol{A}\langle(i,j)\rangle \oplus (\boldsymbol{C}_j, \boldsymbol{R}_i), \tag{5}$$

где символ $\oplus$ обозначает операцию совмещения.

Таким образом соотношения (1) – (5) определяют первый шаг столбцово-строчной факторизации матрицы $\boldsymbol{A}$ относительно ведущего элемента a_{ij} без перестановок строк и столбцов. Следующий аналогичный шаг факторизации осуществляется в отношении ведущего элемента, выбираемого в активной части матрицы $\boldsymbol{A}\langle(i,j)\rangle$, совпадающей с аналогичной частью матрицы $\boldsymbol{A}_{CR}\langle(i,j)\rangle$.

Если в качестве ведущего выбран элемент $a_{pq}-c_{pi}r_{jq}\neq 0$ находящийся на пересечении $p-$й строки и $q-$го столбца матрицы $\boldsymbol{A}\langle(i,j)\rangle$, то, приняв во внимание тождество (1), можем записать

$$\boldsymbol{A}\langle(i,j)\rangle = \boldsymbol{C}_q\boldsymbol{R}_p + \boldsymbol{A}\langle(i,j),(p,q)\rangle, \tag{6}$$

где $n\times 1$ вектор-столбц $\boldsymbol{C}_q$ и $1\times n$ вектор-строка $\boldsymbol{R}_p$ содержат по нулевому элементу $c_{iq}=0$ и $r_{pj}=0$, а $n\times n$ матрица $\boldsymbol{A}\langle(i,j),(p,q)\rangle$ имеет нулевыми две строки и два столбца с номерами i, p и j, q соотвественно.

После подстановки (6) в (1) получаем

$$A = C_j R_i + C_q R_p + A\langle (i,j),(p,q)\rangle \tag{7}$$

или в совмещенной форме

$$A_{CR}\langle (i,j),(p,q)\rangle = A\langle (i,j),(p,q)\rangle \oplus (C_j, R_i, C_q, R_p). \tag{8}$$

Основываясь на выражениях (1) и (6) определяющих частичную одно- и двухшаговую солбцово-строчную факторизацию, получаем формулу полной $CR-$факторизации невырожденной $n\times n$ матрицы A для случая, когда координаты (i,j) динамически выбираемых в процессе факторизации ведущих элементов образуют некоторое упорядоченное множество P:

$$A = \sum_{(i,j)\in P} C_j \cdot R_i\,. \tag{9}$$

Здесь суммирование осуществляется в порядке следования элементов в множестве P. Заметим, что в отличие от (1) и (7) правая часть этого выражения не содержит матричных слагаемых вида $A\langle P\rangle$, поскольку в процессе их формирования за n шагов устанавливались равными нулю значения элементов тех n строк и n столбцов, на пересечении которых выбирались ведущие элементы. Именно на этих местах размещались вектор-столбцы C_j и вектор-строки R_i формируя совмещенную матрицу $A_{CR}\langle P\rangle$. Поэтому $A\langle P\rangle \equiv 0$.

Очевидно выражение (9) может быть представлено в матричном виде

$$A = C\langle P\rangle \cdot R\langle P\rangle, \tag{10}$$

где матрицы $C\langle P\rangle$ и $R\langle P\rangle$ составлены соответственно из вектор-столбцов C_j и вектор-строк R_i, расположенных в том порядке, в котором их индексы встречаются в множестве P. Однако, порядок размещения вектор-столбцов C_j и вектор-строк R_i в совмещенной матрице вида

$$A_{CR}\langle P\rangle = C_{CR} \oplus R_{CR}. \tag{11}$$

соответствует номерам их индексов j и i, т.е. вместо матриц $C\langle P\rangle$ и $R\langle P\rangle$ в матрице $A_{CR}\langle P\rangle$ фактически совмещаются две матрицы $C_{CR}\langle P\rangle$ и $R_{CR}\langle P\rangle$.

Матрицы $C\langle P\rangle$ и $R\langle P\rangle$ в общем случае не приобретают треугольных форм. Только, когда в процессе $CR-$факторизации ведущие элементы выбираются последовательно в координатах $P\langle (1,1),(2,2),\dots,(n,n)\rangle$, тогда формируемые вектор-столбцы C_j и вектор-строки R_i образуют нижнетреугольную и верхнетреугольную матрицы $\boldsymbol{C}\langle\boldsymbol{P}\rangle=\boldsymbol{L}$ и $\boldsymbol{R}\langle\boldsymbol{P}\rangle=\boldsymbol{U}$, которые располагаются в совмещенной матрице $\boldsymbol{A}_{CR}\langle\boldsymbol{P}\rangle$ так, что $\boldsymbol{C}_{CR}\langle\boldsymbol{P}\rangle=\boldsymbol{C}\langle\boldsymbol{P}\rangle=\boldsymbol{L}$ и $\boldsymbol{R}_{CR}\langle\boldsymbol{P}\rangle=\boldsymbol{R}\langle\boldsymbol{P}\rangle=\boldsymbol{U}$.

Основная структурная особенность матриц $\boldsymbol{C}\langle\boldsymbol{P}\rangle$ и $\boldsymbol{R}\langle\boldsymbol{P}\rangle$ общего вида заключается в том, что они могут быть в принципе преобразованы к треугольному виду путем перестановок строк и столбцов.

Результаты экспериментальных исследований. Очевидно выполняемые в процессе перестановок строк и столбцов операции выборки-сравнения индексов строчных и столбцовых списков ненулевых матричных элементов не могут быть непосредственно соотнесены с арифметическими операциями выполняемыми в процессе факторизации над ненулевыми элементами. Поэтому количество операций выборки-сравнения следует оценивать экспериментально по затратам времени на выполнение таких операций и оценивать их удельный вес в общих затратах времени на факторизацию матриц. Описание результатов и условий проведения экспериментов представлены ниже.

Вычисления выполнялись на компьютере Intel P4 (Chipset Intel 865 PE, FSB 800 MHz, CPU 3.0GHz with HT, Dual Channel Memory 1 GB: 2x512MB, DDR400), работающем под управлением операционной системы Microsoft Windows XP.

В основу экспериментальных исследований был положен программный код, написанный автором на языке C++ в среде Microsoft Visual Studio.Net. В нем реализованы последовательно три алгоритма решения системы линейных алгебраических уравнений вида $\boldsymbol{A}\cdot\boldsymbol{X}=\boldsymbol{B}$ с вектором правой части $\boldsymbol{B}=\boldsymbol{A}\cdot\boldsymbol{1}$, где $\boldsymbol{1}-$ единичный вектор. Первый алгоритм реализует метод $CR-$факторизации матрицы $\boldsymbol{A}$ с улучшенной обобщенной стратегией Марковица выбора ведущих элементов, описанной в работе [4]. Второй алгоритм реализует метод $CR-$факторизации той же матрицы $\boldsymbol{A}$, использующий уже сформированное первым алгоритмом множество $\boldsymbol{P}$ последовательности выбора и координат ведущих элементов. Третий алгоритм отличается от второго только тем, что он реализует метод $LU-$факторизации. Отметим, что в обоих реализациях метода $CR-$факторизации матрицы $\boldsymbol{A}$ используется предположение (3).

Исполнение написанного программного кода позволяет оценить суммарные затраты времени вычислений на поиск решения $\boldsymbol{X}$ по заданному вектору $\boldsymbol{B}$ и дифференцировать затраты времени вычислений на $CR-$ и $LU-$факторизацию. При этом обеспечивается выполнение равенства $nnz(\boldsymbol{C}_{CR} \oplus \boldsymbol{R}_{CR}) = nnz(\boldsymbol{L} \oplus \boldsymbol{U})$, т.е. совпадение количеств ненулевых элементов в факторных матрицах $\boldsymbol{C}$, $\boldsymbol{R}$ и $\boldsymbol{L}$, $\boldsymbol{U}$ полученных методами $CR-$ и $LU-$факторизации.

Для обеспечения корректности экспериментальных исследований было максимально ослаблено влияние процедур управления расположением ненулевых матричных элементов в рабочих массивах. Прежде всего размеры массивов ненулевых элементов, их строчных и столбцовых индексов были выбраны равными $45 \cdot 10^6$, т.е. настолько большими, чтобы не возникало потребности инициировать работу специальных процедур «уборки мусора» в них [4, 5]. К тому же во всех тестах применялась улучшенная обобщенная стратегия Марковица, где поиск ведущего элемента ограничивался множеством S_p ненулевых элементов, расположенных в p строках активной подматрицы с минимальным количеством таких элеменов, а ведущим выбирался элемент a_{ij} с наименьшей ценой Марковица, но тот, абсолютное значение которого не более чем в τ раз меньше максимального по абсолютной величине элемента $a_{\boldsymbol{max}} \in S_p$.

Таблица 1. Тестовые матрицы, их размерность n, количество ненулевых элементов в исходных матрицах $nnz(\boldsymbol{A})$ и в совмещенных факторных матрицах $nnz(\boldsymbol{C} \oplus \boldsymbol{R})$, ошибки ε и время решения $T_{\boldsymbol{CR}}$ тестовых систем.

название	n	$nnz(\boldsymbol{A})$	$nnz(\boldsymbol{C} \oplus \boldsymbol{R})$	ε	$T_{\boldsymbol{CR}}, c$
ASIC_680k	682862	3871773	8220972	4.57E-08	1024.531
ASIC_680ks	682712	2329176	6170719	9.51E-09	47.953
ASIC_320ks	321671	1827807	5082048	2.89E-12	54.875
scircuit	170998	958936	2408645	8.39E-11	2.328
circuit_4	80209	307604	707803	1.92E-09	1.359
bayer01	57735	277774	7283567	4.33E-08	40.453
mark3jac060	27449	170695	28775871	3.42E-11	2345.125
ex35	19716	228208	2474829	6.82E-08	7.360
poisson3Da	13514	352762	10697886	1.10E-13	162.281
circuit_3	12127	48137	76708	2.07E-12	0.141
cry10000	10000	49699	501518	4.04E-06	0.828
gemat11	4929	33185	77616	2.21E-13	0.047
lns_3937	3937	25407	568884	1.60E-12	1.719
psmigr_1	3140	543162	6534812	9.40E-13	122.531
orani678	2529	90158	336551	3.07E-15	0.469
adder_trans_01	1814	14579	20541	2.11E-16	0.016
orsirr_1	1030	6858	57892	1.42E-13	0.063
orsirr_2	886	5970	45146	1.37E-13	0.031

Вычислительные эксперименты выполнялись с матрицами, взятыми с веб-сайтов [9, 10]. Общие характеристики тестовых матриц и результаты экспериментирования с ними представлены в таблицах 1-2, где

$$\varepsilon = \sqrt{\frac{\sum_{z=1}^{n} (x_z - 1)^2}{n}}.$$

В зависимости от тестируемой матрицы параметры p и τ устанавливались равными: в тесте ASIC_680k – $p = 100$ и $\tau = 0.0005$, в тесте mark3jac060 – $p = n$ и $\tau = 0.001$. Во всех остальных тестах значения параметров были $p = 1$ и $\tau = 1$.

Таблица 2. Затраты времени на факторизацию (T_{CR-F}, T_{LU-F}) и решение (T_{CR-S}, T_{LU-S}) тестовых систем уравнений методами $CR-$ и $LU-$факторизации; относительное замедление процесса $LU-$факторизации матриц $\delta T_F = \left(\frac{T_{LU-F}}{T_{CR-F}} - 1 \right) \cdot 100\%$.

название	T_{CR-F}, с	T_{LU-F}, с	δT_F, %	T_{CR-S}, с	T_{LU-S}, с
ASIC_680k	1004.328	1539.047	53.2	0.172	0.156
ASIC_680ks	46.656	49.204	5.5	0.188	0.140
ASIC_320ks	54.875	59.921	9.2	0.171	0.141
scircuit	2.000	3.907	95.4	0.125	0.078
circuit_4	1.172	1.562	33.3	0.016	0.032
bayer01	39.062	51.563	32.0	0.094	0.094
mark3jac060	903.953	1106.750	22.4	0.297	0.281
ex35	7.016	10.578	50.8	0.047	0.031
poisson3Da	159.109	204.328	28.4	0.110	0.110
circuit_3	0.093	0.109	17.2	0.000	0.016
cry10000	0.781	1.171	49.9	0.000	0.016
gemat11	0.031	0.047	51.6	0.000	0.000
lns_3937	1.625	2.313	42.3	0.015	0.000
psmigr_1	120.437	147.562	22.5	0.063	0.063
orani678	0.437	0.656	50.1	0.016	0.000
adder_trans_01	0.015	0.016	6.7	0.000	0.000
orsirr_1	0.047	0.078	66.0	0.000	0.000
orsirr_2	0.047	0.063	34.0	0.000	0.000

Описанные условия проведения экспериментов позволили избежать инициализации процедур «уборки мусора» в массивах. Однако, в алгоритме реализующем метод $LU-$факторизации мы применили одну из известных процедур перестановки строк и столбцов, минимизирующую объем требуемых для этого операций выборки-сравнения [11]. Такая процедура приводит не только к изменению значений индексов переставляемых строк и столбцов в массивах строчных и столбцовых индексов, но и сопровождается перестановками отдельных элементов в тех же массивах, а таже в массиве ненулевых элементов. В результате перестановок элементов в массивах их размещение на текущем шаге реализации метода $LU-$факторизации становится отличным от того размещения, которое можно наблюдать в случае реализации метода $CR-$факторизации. Такие различия в размещении элементов усиливаются необходимостью размещения вновь возникающих ненулевых элементов, что порождает различные динамические процессы переразмещения элементов строк и столбцов. Таким образом нам не удалось достичь полного равенста условий проведения экспериментов с программными реализациями методов $CR-$ и $LU-$факторизации. Однако мы минимизировали расхождения в условиях настолько насколько это было возможно.

Из полученных результатов следуют такие выводы:

1. Затраты времени на осуществление перестановок строк и столбцов в активных подматрицах оказываются сопоставимыми с затратами времени на факторизацию матриц. Реализация перестановок в методе $LU-$факторизации приводит к существенному замедлению вычислительного процесса в среднем на величину $\delta T_F = 37.3\%$ для данного множества тестов, что следует из таблицы 2, где представлены значения $\delta T_F = \left(\frac{T_{LU-F}}{T_{CR-F}} - 1 \right) \cdot 100\%$.
2. Затраты времени на вычисление решений систем уравнений вида $\boldsymbol{C}\langle \boldsymbol{P} \rangle \cdot \boldsymbol{R}\langle \boldsymbol{P} \rangle \cdot \boldsymbol{X} = \boldsymbol{B}$, полученных методом $CR-$факторизации, больше затрат времени на вычисление решений систем уравнений вида $\boldsymbol{L} \cdot \boldsymbol{U} \cdot \boldsymbol{X} = \boldsymbol{B}$, формируемых методом $LU-$факторизации. Увеличение вычислительных затрат обусловлено особенностями алгоритмов решения систем уравнений вида $\boldsymbol{C}\langle \boldsymbol{P} \rangle \cdot \boldsymbol{V} = \boldsymbol{B}$ и $\boldsymbol{R}\langle \boldsymbol{P} \rangle \cdot \boldsymbol{X} = \boldsymbol{V}$ с нетреугольными матрицами $\boldsymbol{C}\langle \boldsymbol{P} \rangle$ и $\boldsymbol{R}\langle \boldsymbol{P} \rangle$. Однако, в целом суммарное время решения исходных систем уравнений составляет $T_F + T_S$ и является существенно большим в случае использования метода $LU-$факторизации.

Заключение. В отличие от метода $LU-$ факторизации матриц с выбором ведущих элементов, где в общем случае обязательно выполняются операции перестановки строк и столбцов, метод $CR-$факторизации матриц не требует выполнения таких операций, что ускоряет процесс факторизации. Ускорение оказывается особенно существенным в случае затрудненного доступа к матричным элементам, который обусловлен использованием специальных форматов размещения в массивах ненулевых элементов разреженных матриц.

Очевидно метод $CR-$факторизации матриц может использоваться не только, как эффективный прямой метод решения больших систем уравнений, но и как метод быстрого построения предобусловливателей для итерационного решения систем [12-14]. В этом случае построение предобусловливателей посредством неполной $CR-$факторизации матриц может осуществляться без какого-либо перемещения ненулевых элементов в массивах. Таким образом предлагаемый метод $CR-$факторизации матриц может быть положен в основу алгоритмов и программ ускоренного решения систем линейных алгебраических уравнений большой размерности.

1. www.cise.ufl.edu/research/sparse/codes/
2. www.srcc.msu.su/num_anal/lib_na/cat/cat552.htm
3. Писанецки С. Технология разреженных матриц. – М.: Мир, – 1988. – 410 с.
4. Эстербю О., Златев З. Прямые методы для разреженных матриц. – М.: Мир, – 1987. – 120 с.
5. Икрамов Х.Д. Вычислительные методы линейной алгебры. – Знание, 1989. – 48 с.
6. Davis T.A. Direct Methods for Sparse Linear Systems. – Philadelphia.: SIAM, – 2006. – 226 p.
7. Самарский А.А., Гулин А.В. Численные методы. – М.: Наука, – 1989. – 432 с.
8. Джордж А., Лю Дж. Численное решение больших разреженных систем уравнений. – М.: Мир, – 1984. – 334 с.
9. www.cise.ufl.edu/research/sparse/matrices/index.html
10. http://math.nist.gov/MatrixMarket/matrices.html
11. www.netlib.org/y12m/index.html
12. Saad Y. Iterative Methods for Sparse Linear Systems, University of Minnesota, Department of Computer Science and Engineering, Minneapolis, MN, 2000. – 448 p.
13. Ортега Дж. Введение в параллельные и векторные методы решения линейных систем. – М.: Мир, – 1991. – 386 с.
14. Saukh S.Ye. Incomplete Cholesky factorization in fixed memory // Parallel Processing and Applied Mathematics: 5th International Conference, PPAM 2003, Czestochowa, Poland, September 7-10, 2003. Revised Papers. – Springer-Verlag Heidelberg: Lecture Notes in Computer Science. –Volume 3019 / 2004, pp. 1042 - 1051.

Материал поступил в редколлегию 10.05.2007