

МАТЕМАТИКА

УДК 512.64, 519.64

Член-корреспондент АН Армении А. Б. Нерсисян, Х. М. Бешликян

Численная реализация нового метода решения
уравнений теплицева типа

(Представлено 18/IX 1990)

Известно (см., например, ⁽¹⁾), что решение линейной алгебраической системы с теплицевой $n \times n$ -матрицей, при условии невырожденности главных подматриц, можно реализовать посредством алгоритма Л—Т (Н. Левинсона—В. Тренча), требующего выполнения $6n^2 + O(n)$ ($n \rightarrow \infty$) аддитивных и мультипликативных операций вместо $2/3n^3 + O(n^2)$, если иметь в виду метод Гаусса для систем общего вида. Попытка отказа от этого ограничения на ведущие миноры (при невырожденности самой матрицы) привела к модификациям алгоритма Л—Т, что, так или иначе, усложняет его (см., например, ^(1, 2)).

В сообщении ⁽³⁾ был предложен совершенно иной подход. Его суть заключается в следующем.

Пусть дана комплекснозначная невырожденная матрица $T_n = \|t_{ij}\|$, $i, j = 1, 2, \dots, n$ и требуется вычислить обратную матрицу $T_n^{-1} = R_n = \|r_{ij}^n\|$, $i, j = 1, 2, \dots, n$. Задав произвольным комплексным числом $\varphi \neq 0$, дополним матрицу T_n до φ -косоциркулянта* размеров $(2n - 1) \times (2n - 1)$ посредством формул (см. ⁽¹⁾)

$$t_{-k} = \varphi t_{n-k+1}, \quad k = \pm(n+1), \dots, \pm(2n-1) \quad (1)$$

Предположим, что $\det T_{2n-1} \neq 0$. Как известно (см. ⁽¹⁾), матрицу $T_{2n-1}^{-1} = R_{2n-1} = \|r_{ij}^{2n-1}\|$, $i, j = 1, 2, \dots, 2n-1$, можно выписать в явном виде через дискретное преобразование Фурье. Применение быстрого преобразования Фурье (БПФ) решает эту задачу с вычислительными затратами порядка $O(n \ln n)$ ($n \rightarrow \infty$) операций.

Дальнейшие шаги связаны с обратной рекурсией по $k = 2n-2, 2n-3, \dots, n+1, n$, где $T_k = \|t_{ij}\|$, $i, j = 1, 2, \dots, k$.

* Дополнение теплицевой матрицы до циркулянта ($\varphi = 1$) производилось ранее при решении уравнения с ленточной матрицей в ⁽⁴⁾ (см. также ⁽¹⁾).

Рассмотрим сначала невырожденные матрицы общего вида $A_k = [a_{ij}]$, $i, j = 1, 2, \dots, k$ и соответственно обозначим $A_k^{-1} = R_k = [r_{ij}^k]$. Нетрудно установить (хотя бы прямой проверкой) следующую связь между R_k и R_{k-1} :

$$r_{ij}^{k-1} = r_{ij}^k - r_{ik}^k (r_{kk}^k)^{-1} r_{kj}^k, \quad i, j = 1, 2, \dots, k \quad (2)$$

С другой стороны, в изучаемом теплицевом случае ($a_{ij} = t_{i-j}$) известна формула В. Тренча (см. (1))

$$r_{i+1, j+1}^k = r_{ij}^k + r_{i+1, i}^k (r_{ii}^k)^{-1} r_{1, j+1}^k - r_{ik}^k (r_{ii}^k)^{-1} r_{kj}^k, \quad (3)$$

$$i, j = 1, 2, \dots, k-1,$$

сводящая задачу определения R_k к вычислению ее «граничных» значений. Наконец, учитывая персимметричность скалярной матрицы R_k ($r_{i, j}^k = r_{k-j+1, k-i+1}^k$), приходим к соотношениям

$$r_{i, k-1}^{k-1} = r_{i+1, k}^k - r_{k, k-1}^k (r_{kk}^k)^{-1} r_{1k}^k, \quad (i = 1, 2, \dots, k-1) \quad (4)$$

$$r_{k-1, i}^{k-1} = r_{k, i+1}^k - r_{k1}^k (r_{kk}^k)^{-1} r_{k-i, k}^k.$$

Формулы (3) и (4) позволяют рекуррентно найти $T_{2n-1}^{-1} = R_{2n-1}$ при невырожденности матриц T_k ($k = 2n-2, \dots, n+1$). Назовем этот путь алгоритмом А и отметим следующие его характеристики:

1. Алгоритм работает при условии $\det T_k \neq 0$, $k = n+1, n+2, \dots, 2n-1$. Ясно, что при близком к машинному нулю значении одного из этих определителей может произойти авост (или, по крайней мере, резкое накопление ошибок). Если для данного значения φ это условие не выполнено, то алгоритм надо повторить при другом значении. Эвристические соображения показывают, что при этом вероятность повторения переполнения очень мала. В многочисленных экспериментах, проведенных нами, авост ни разу не встречался даже при первом выборе φ .

2. Вычислительные затраты на обсчет «граничных значений» матрицы T_n^{-1} по алгоритму (4) составляют $6n^2 + O(n)$ аддитивных и мультипликативных операций, а требуемая память— $4n-2$ ячейки. Заметим, что в алгоритме Л—Т на это требуется $4n^2 + O(n)$ операций при той же памяти (см. (1)). На выполнение всего алгоритма А требуется $10n^2 + O(n)$ операций (если для T_{2n-1}^{-1} применить БПЭ). Однако матрицу T_n^{-1} можно представить по формуле Гохберга—Семенцула (1, 2) и, таким образом, для решения уравнения с матрицей T_n можно ограничиться $10n^2 + O(n)$ операциями, в то время как по алгоритму Л—Т требуется $6n^2 + O(n)$.

3. Алгоритм А полностью векторизован и может быть распараллелен применением $4n-2$ процессоров. При этом алгоритм (4) будет выполнен за время $(n-1)t_0$, где t_0 —время выполнения операции вида

$a + bc$. Если при этом через t_p обозначить время выполнения алгоритма при использовании p процессоров, то коэффициент эффективности будет равен (см. (5))

$$k_* \lim_{n \rightarrow \infty} t_1 / 4n t_{4n-2} \rightarrow 0,75 \quad (n \rightarrow \infty). \quad (5)$$

Аналогичная форма алгоритма Л—Т обладает примерно такими же характеристиками и распараллеливается с $k_* \rightarrow 0,5$ (1).

4. Простое устройство графа алгоритма (4) позволяет легко наметить архитектуру систолического массива из p вычислителей для реализации параллельного счета на основе СБИС (см. (5), $p \leq 4n-2$).

5. Алгоритм А может быть успешно реализован на ЭВМ конвейерного типа, что и было продемонстрировано (см. ниже) в обсчетах на матричном процессоре МП ЕС 2700 ЕрНИИММ (6).

6. Формулы (2) и (3) сохраняют свою силу и в случае теплицевых матриц $T = \|t_{ij}\|$ с элементами t_k из некоммутативной алгебры с единицей. В частности, это относится к блочно-теплицевым матрицам. При таком обобщении надо при переходе от (2), (3) к (4) отказаться от свойства персимметричности, в результате чего формулы (4) примут вид, пригодный для определения четырех граничных векторов:

$$r_{ki}^k, r_{ik}^k, r_{ii}^k, r_{ii}^k \quad (i = 1, 2, \dots, k; \quad k = 2n-2, \dots, n+1, n).$$

7. В случае интегральных уравнений Фредгольма второго рода на конечном отрезке с ядром, зависящим от разности аргументов, аналоги алгоритма А различной точности строятся по той же схеме (см. (3)).

Вычислительный эксперимент был проведен для ряда матриц с разными свойствами. В качестве контрольного выбран метод Гаусса с постолбцовым выбором ведущего элемента (алгоритм G). В частности, были обсчитаны следующие тест-матрицы из сборника (7):

T1, T2 — симметричные трехдиагональные матрицы вида $t_0 = a$, $t_1 = t_{-1} = -1$, $t_k = 0$ ($|k| \geq 2$) при $a = 2$ и $a = 1$ соответственно. Отметим, что все миноры четного порядка матрицы T2 равны нулю, вследствие чего алгоритм Л—Т для нее не проходит.

T3 — плотная симметрическая матрица вида $t_0 = 0$, $t_k = 1$ ($|k| \geq 1$).

T4, T5 — плотные матрицы вида $t_k = (k + a)^{-1}$, $|k| \geq 0$ при $a = 0,1$ и $a = 1,1$ соответственно. Это несимметрические, плохо обусловленные при больших размерах, матрицы типа известной матрицы Гильберта.

Время работы арифметического процессора (в секундах) при обсчетах по сравниваемым алгоритмам приведено на диаграммах (рис. 1 и 2). Как и следовало ожидать, преимущество алгоритма А в скорости при работе на ЭВМ ЕС 1045 растет с ростом размерности n (рис. 1). При использовании матричного процессора МП ЕС 2700 время резко сокращается в обоих случаях (рис. 2). Заметим, что при

$n < 150$ алгоритм G быстрее по той причине, что он организован в матричном процессоре на микропрограммном уровне. При аналогичной реализации алгоритма A его преимущества были бы ощутимей, особенно при небольших n .

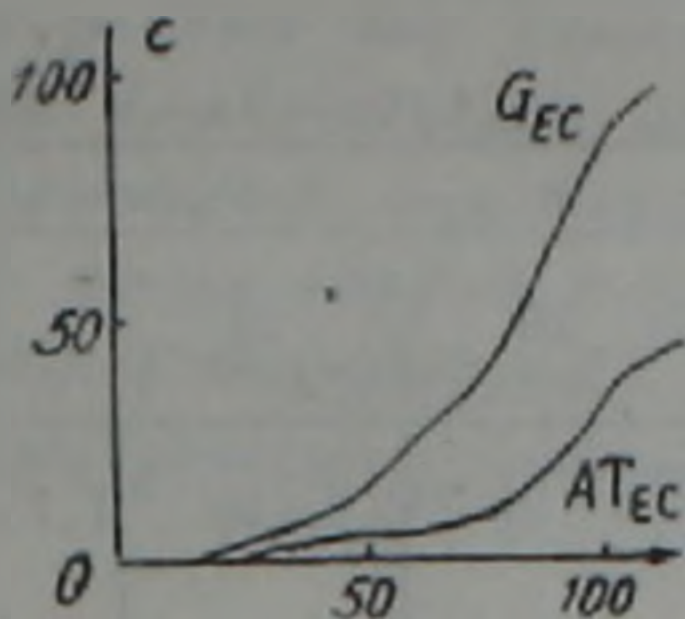


Рис. 1

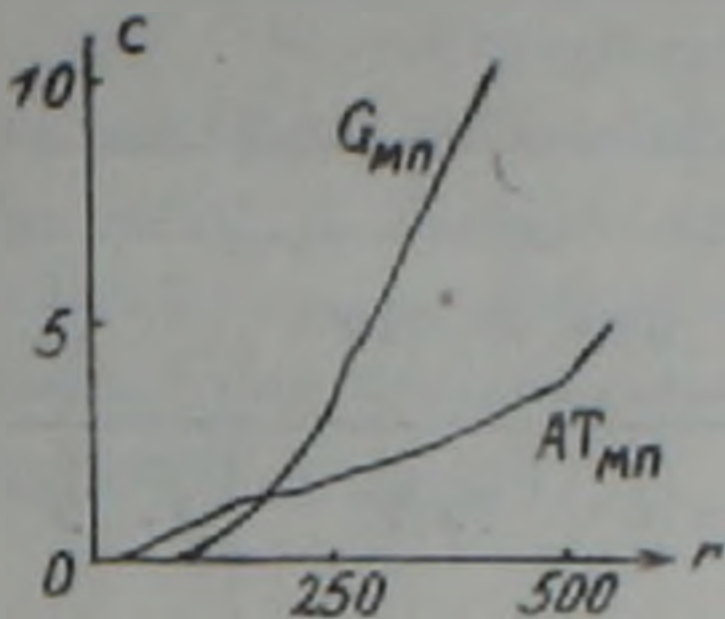


Рис. 2

Результаты применения алгоритма A при $\varphi=1$ к матрице $T2$ более чем на порядок хуже, чем при обсчете по алгоритму G (табл. 1). Возможно, при ином выборе φ (см. ниже) результаты были бы лучше. В случае матрицы $T3$ (табл. 2) при том же $\varphi=1$ точность алгоритма G лишь в несколько раз лучше.

Совершенно неожиданная ситуация наблюдается в случае матриц $T4$ и $T5$ (табл. 3). Здесь с ростом n точность алгоритма G резко ухудшается (а для матрицы $T5$ уже при $n=15$ алгоритм G практически непригоден)). На этом фоне результаты применения алгоритма A превосходны в случае матрицы $T4$ и вполне удовлетворительны для матрицы $T5$.

Таблица 1

Алгоритм	n				
	25	45	66	85	90
AT2 ($\varphi=1$)	0,0018	0,002	0,003	0,012	0,015
GT2	0,0001	0,0003	0,0003	0,0003	0,0005

Таблица 2

Алгоритм	n				
	6	30	50	80	90
AT3 ($\varphi=1$)	4×10^{-5}	3×10^{-4}	1×10^{-3}	3×10^{-3}	5×10^{-3}
GT3	1×10^{-5}	1×10^{-4}	2×10^{-4}	8×10^{-4}	1×10^{-3}

На примере матрицы $T1$ была выявлена зависимость результатов применения алгоритма A от выбора параметра φ . Оказалось, что при $\varphi=1+10\%$ его точность ($\delta=2 \times 10^{-5}$ при $n=32$ и $\delta=9 \times 10^{-4}$ при

$n = 90$) примерно такая же, как и у алгоритма G ($\delta = 1 \times 10^{-5}$ при $n = 32$ и $\delta = 5 \times 10^{-4}$ при $n = 90$). При $\varphi = 9 + 7i$ алгоритм A уже гораздо менее точен ($\delta = 9 \times 10^{-2}$ при $n = 32$ и $\delta = 5 \times 10^{-1}$ при $n = 90$), а при $\varphi = 1$ он уже практически непригоден (в интервале $32 \leq n \leq 90$ ошибка держится на уровне $\delta \approx 1,08 - 1,09$).

Таблица 3

Алгоритм	n						
	5	10	15	20	25	30	40
$A74$ ($\varphi = 9 + 7i$)	1×10^{-5}	2×10^{-5}	4×10^{-5}	$1,3 \times 10^{-4}$	$2,8 \times 10^{-4}$	$1,7 \times 10^{-4}$	$2,4 \times 10^{-4}$
$G74$	1×10^{-5}	$1,6 \times 10^{-5}$	2×10^{-5}	$3,5 \times 10^{-5}$	4×10^{-5}	3×10^{-1}	6×10^{-1}
$A75$ ($\varphi = 1$)	0	1×10^{-3}	3×10^{-3}	1×10^{-2}	3×10^{-2}	3×10^{-2}	6×10^{-2}
$G75$	1×10^{-5}	5×10^{-4}	$1,6 \times 10^{-3}$	$4,7 \times 10^{-3}$	5×10^{-2}	$2,4 \times 10^3$	∞

Заключение. Приведенные результаты свидетельствуют о том, что алгоритм A является неплохой альтернативой алгоритмам типа $L-T$, если *a priori* неизвестна информация об обратимости ведущих подматриц. Особенно обнадеживают результаты, приведенные в табл. 3. Сильная зависимость результатов счета от выбора φ требует выработки оптимальной стратегии этого выбора.

В то же время можно искусственно построить такую невырожденную теплицеву матрицу T_n для которой алгоритм A ни при каком φ не пройдет. Хотя такие случаи в практике вряд ли встретятся, их также можно избежать. Например, можно расширять T_n до T_m ($m > 2n - 1$), получив в свое распоряжение, кроме φ , новые свободные параметры. Это удобно еще потому, что для применения БПФ желателен выбор размерности m в виде степени двойки (см. (1)).

Имея в виду весь набор методов обращения матриц теплицева типа и аналогичных интегральных уравнений, желательно разработать механизм автоматического выбора алгоритмов. Например, можно начать с наиболее быстрого алгоритма $L-T$ и, в случае его сбоя или сомнений в его точности, продолжить обсчет по другим алгоритмам (скажем, по мере возрастания их высоты). При этом необходимо иметь в виду и методы ортогонального (8) и блочно-ортогонального (9) разложений, не требующие каких-либо ограничений на главные подматрицы, но несколько более медленные (в скалярном случае (8) вычисление T_n^{-1} требует $20n^2 + O(n)$ ($n \rightarrow \infty$) аддитивных и $23n^2 + O(n)$ мультипликативных операций).

Институт математики Академии наук Армении
Ереванский научно-исследовательский
институт математических машин

Տոպլիցյան տիպի նավասարումների լուծման նոր եղանակի
թվային իրականացում

Տոպլիցյան մատրիցաներով համակարգերի համար ⁽³⁾ աշխատանքում առաջարկվել էր լուծման նոր եղանակ, որը հնարավորություն էր տալիս շրջանցել հայտնի մոտեցումներին բնորոշ սահմանափակումները:

Տվյալ աշխատանքում այդ եղանակը գտել է թվային իրականացում: Ինչպես ցույց են տալիս բերված հաշվարկների արդյունքները, առաջարկվող ալգորիթմը կարելի է հաջողությամբ կիրառել: Համապատասխան ալգորիթմ կարելի է կառուցել ինտեգրալ հավասարումների դեպքում:

ЛИТЕРАТУРА — ԳՐԱԿԱՆՈՒԹՅՈՒՆ

- ¹ В. В. Воеводин, Е. Е. Тыртышников, Вычислительные процессы с теплицевыми матрицами, Наука, М., 1987. ² G. Heinig, Zeitschrift d. TH Karl-Marx-Stadt, v. 21, № 3, p. 326—333 (1983). ³ А. Б. Нерсисян, II Всесоюзная конференция по актуальным проблемам информатики и вычислительной техники «Информатика—87», Тезисы докладов, Ереван, 20—22 октября 1987, с. 157—158, 1987. ⁴ А. К. Jain, IEEE Trans., ASSP-26, № 2, p. 121—126 (1978). ⁵ В. В. Воеводин, Математические модели и методы в параллельных процессах, Наука, М., 1986. ⁶ М. А. Семерджян, Ж. С. Налбандян, Л. Х. Гаспарян, Принципы работы и система программного обеспечения МП ЕС 2700, Наука, М., 1988. ⁷ В. Н. Фаддеева, Л. Ю. Колотилина, Вычислительные методы в линейной алгебре: набор матриц для тестирования, ч. 1, ЛОМИ, Л., 1982. ⁸ G. Gubenko, SIAM J. Sci. Stat. Comput., v. 8, p. 734—740 (1987). ⁹ Г. В. Азекян, А. Б. Нерсисян, Деп. в Арм. НИИНТИ, № 64—Ар89, 1988.