

УДК 519.6

## СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИТЕРАЦИОННОГО МЕТОДА GMRES С ПРЯМЫМ LU РАЗЛОЖЕНИЕМ ПРИ РЕШЕНИИ СИСТЕМ ЛИНЕЙНЫХ УРАВНЕНИЙ БОЛЬШИХ РАЗМЕРНОСТЕЙ

Богданов А.Е., Торшина О.А.

ФГБОУ ВО «Магнитогорский государственный технический университет имени Г.И. Носова»,  
Магнитогорск, e-mail: olganica@mail.ru

При выборе численных методов для решения систем линейных алгебраических уравнений в первую очередь руководствуются такими параметрами, как точность (позволяет ли метод получить решение с требуемой точностью) и скорость (в зависимости от того, как быстро необходимо получить решение, оценивается единственный релевантный показатель – время на конкретной машине или возможность масштабируемости метода на большой вычислительный кластер). Обычно приходится выбирать компромисс между этими параметрами. Наравне с вышеперечисленным для выбора конкретного метода рассматривается структура матрицы (симметричная, плотная или разреженная, пластинчатая) и собственные значения (все ли они положительны, их сгруппированность, разброс величин). В том числе немаловажную роль играет реализация метода (не всегда удается найти для подходящего метода хорошую реализацию). Когда речь идет о системах уравнений больших размерностей, возникает вопрос о сходимости методов для таких систем. В связи с этим для осуществления оптимального выбора численного метода для решения систем линейных алгебраических уравнений возникает задача оценки различных методов для конкретной проблемы. В данной статье производится сравнительный анализ итерационного метода GMRES с прямым LU разложением при решении систем линейных алгебраических уравнений больших размерностей на плотных несимметричных матрицах квадратной формы. Для сравнения сходимости осуществляется нахождение погрешности полученных решений, высчитывается невязка решений рассматриваемых методов для всех выбранных в рамках статьи размерностей.

**Ключевые слова:** численные методы, системы линейных алгебраических уравнений, GMRES, LU, итерационный, прямой

## COMPARATIVE ANALYSIS OF THE GMRES ITERATIVE METHOD WITH DIRECT LU DECOMPOSITION WHEN SOLVING THE SYSTEMS OF LINEAR EQUATIONS OF LARGE DIMENSIONS

Bogdanov A.E., Torshina O.A.

Magnitogorsk State Technical University named after G.I. Nosov, Magnitogorsk,  
e-mail: olganica@mail.ru

When choosing numerical methods for solving systems of linear algebraic equations, first of all, they are guided by such parameters as accuracy (does the method allow to obtain a solution with the required accuracy) and speed (depending on how quickly a solution is needed, the only relevant indicator is evaluated: time on a specific machine or the possibility of scalability of a method on a large computing cluster). Usually, it is necessary to choose a compromise between these parameters. The structure of the matrix (symmetric, dense or discharged, lamellar) and eigenvalues (whether they are all positive, their grouping, dispersion of values) are considered, equally with the above parameters, to select a specific method. Also, the important role is played by the implementation of the method (it is not always possible to find a good implementation for a suitable method). When it comes to systems of equations of large dimensions, the question on the convergence of methods for such systems appears. In this regard, to implement the optimal choice of a numerical method for solving systems of linear algebraic equations, the problem of evaluating various methods for a specific problem comes up. This article presents a comparative analysis of the iterative GMRES method with direct LU decomposition when solving systems of linear algebraic equations of large dimensions on dense asymmetric square matrices. For comparison of convergence, the error of the obtained solutions is found; the discrepancy between the solutions of the considered methods is calculated for all dimensions selected within the article.

**Keywords:** numerical methods, systems of linear algebraic equations, GMRES, LU, iterative, direct

Рассмотрим численные методы GMRES и прямое LU разложение, произведем сравнительный анализ полученных данных методами решений СЛАУ больших размерностей на плотных несимметричных матрицах квадратной формы.

Цель исследования: выявить зависимость между сходимостью рассматриваемых численных методов и размерностью СЛАУ.

### Материалы и методы исследования

Вычисления проводились на программной платформе .NET Framework 4.5.1, в качестве IDE использовалась Microsoft Visual Studio 2017. Для реализации метода GMRES за основу был взят github проект участника Dan Lavrushko [1]. LU декомпозиция выполнена с помощью библиотеки Math.NET Numerics [2].

*Постановка задачи*

Рассмотрим задачу решения СЛАУ  $A \cdot x = b$  итерационным методом GMRES [3] и LU разложением (декомпозицией) [4], где:  $A$  – невырожденная, плотная, несимметричная матрица размером  $m \times m$ ;  $x$  – вектор неизвестных размером  $m$ ;  $b$  – свободные члены размером  $m$  (1).

$$\begin{cases} a_{11} \cdot x_1 + a_{12} \cdot x_2 + \dots + a_{1m} \cdot x_m = b_1 \\ a_{21} \cdot x_1 + a_{22} \cdot x_2 + \dots + a_{2m} \cdot x_m = b_2 \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ a_{m1} \cdot x_1 + a_{m2} \cdot x_2 + \dots + a_{mm} \cdot x_m = b_m \end{cases} \cdot \quad (1)$$

*Метод GMRES*

Для численного решения произвольных СЛАУ с заданной точностью на невырожденных, плотных и несимметричных матрицах распространено применение итерационного обобщенного метода минимальных невязок (GMRES).

Метод подразумевает использование алгоритма итераций Арнольди [5] с целью нахождения приближенных собственных векторов матрицы посредством построения ортонормированного базиса подпространства Крылова [6] для осуществления аппроксимации решения с минимальной невязкой.

Сам алгоритм метода GMRES был предложен в 1986 г. Юсефом Саадом и Мартином Х. Шульцем. GMRES стал обобщением метода MINRES Криса Пейджда и Майкла Сондерса, также является частным случаем метода DIIS Питера Пулайе, применимым к нелинейным системам.

Из сложности процесса Арнольди, минимизации невязки и матрично-векторного произведения складывается сложность метода GMRES.

Увеличение объема требуемой памяти для покрытия стоимости вычисления собственных векторов на каждой итерации составляет  $O(n^2)$ , где  $n$  – номер итерации. Очевидно, что многие вычислительные системы не в состоянии удовлетворить такие потребности алгоритма, поэтому для устранения проблемы производится перезапуск вычислений после заданного значения  $k$  итераций, начиная с  $x_k$  в качестве начального предположения. Такой подход признан модификацией исходного GMRES алгоритма и обозначается с припиской Restarted. При использовании Restarted GMRES возникает существенный недостаток: перезапущенное подпространство часто близко к более раннему подпространству, поэтому метод страдает от стагнации конвергенции.

Решением проблемы служит повторное использование подпространства Крылова методами GCROT и GCRODR, которые относятся к типу GCRO.

Сложность решения задачи минимизации невязки (минимальных квадратов) составляет  $O(m^2)$ , где  $m$  – размер матрицы.

Сложность получения решения матрично-векторным произведением составляет  $O(nm)$ .

Математическое описание алгоритма GMRES. На каждом  $k$ -м шаге решение  $x_k$  рассматривается в виде  $x_0 + Vy_k$ , где столбцы  $v_k$  матрицы  $V$  находятся с помощью процесса Арнольди для построения ортонормированного базиса  $\{v_1, v_2, \dots, v_k\}$  подпространства Крылова  $K_k(A, v_1)$ , вектор  $y_k$  представляет собой решение системы  $H_k y_k = \beta_k e_1$ , где  $H_k$  – верхняя матрица Хессенберга размера  $k \times k$ , элементы которой формируются в процессе ортогонализации Арнольди,  $\beta_k = \|r_0\|_2$ ,  $r_0 = b - Ax_0$ , и, наконец,  $e_1$  – канонический вектор  $(1 \ 0 \ \dots \ 0)^T$  размерности  $k$ .

Ортонормированный базис  $\{v_1, v_2, \dots, v_k\}$  подпространства  $K_k(A, v_1)$  вычисляется с помощью ортогонализации Арнольди по формулам

$$\begin{aligned} \omega &= Av_j, \\ h_{ij} &= (\omega, v_i), \\ \omega &= \omega - h_{ij} v_i, \\ h_{(i+1)j} &= \|\omega\|, \\ v_{j+1} &= \frac{\omega}{h_{(i+1)j}}, \\ j &= \overline{1..k}, \quad i = \overline{1..j}. \end{aligned} \quad (2)$$

После завершения вычислений процессом Арнольди получаем ортонормированный базис подпространства Крылова  $V_k = \{v_1, v_2, \dots, v_k\}$  и матрицу, расширенную последним вычисленным вектором  $v_{k+1}$ :  $\overline{V}_k = \{v_1, v_2, \dots, v_k, v_{k+1}\}$ .

$H_k$  – верхняя Хессенбергова матрица, элементы которой равны коэффициентам ортогонализации (3).

$$H_k = \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} & \dots & h_{1k} \\ h_{21} & h_{22} & h_{23} & h_{24} & \dots & h_{2k} \\ 0 & h_{32} & h_{33} & h_{34} & \dots & h_{3k} \\ \vdots & & \ddots & \ddots & & \vdots \\ 0 & \dots & 0 & h_{k-1,k-2} & h_{k-1,k-1} & h_{k-1,k} \\ 0 & \dots & 0 & 0 & h_{k,k-1} & h_{kk} \\ 0 & \dots & & & 0 & h_{k+1,k} \end{pmatrix} \cdot \quad (3)$$

Выполнив несложные преобразования, можно получить выражение для невязки

на  $k$ -й итерации:  $r_k = \beta e_1 - H_k y_k$ . Очевидно, что для задачи  $\|r_k\|_2 \rightarrow \min$  минимум достигается при  $H_k y_k = \beta e_1$ .

Для решения последней системы матрица приводится к верхнему треугольному виду с помощью  $k$  вращений Гивенса  $R_k = \Omega_k \Omega_{k-1} \dots \Omega_1$  (4).

$$\Omega_i = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & c_i & s_i & & \\ & & & -s_i & c_i & & \\ & & & & & 1 & \\ & & & & & & \ddots & \\ & & & & & & & 1 \end{pmatrix}, \quad (4)$$

где  $s_i = \frac{h_{i+1,i}}{\sqrt{h_{ii}^2 + h_{i+1,i}^2}}$ ,  $c_i = \frac{h_{ii}}{\sqrt{h_{ii}^2 + h_{i+1,i}^2}}$ .

*Метод LU разложения*

LU разложение основывается на методе последовательного исключения Гаусса. Метод LU разложения подразумевает представление исходной матрицы  $A$  в виде произведения нижнетреугольной матрицы  $L$  на верхнетреугольную матрицу  $U$  (5). Традиционно, либо матрица  $L$ , либо  $U$  нормализуются до получения единичной диагонали. Несмотря на то, что метод LU разложения предназначен для матриц любой (прямоугольной) формы, чаще всего он применяется к квадратным матрицам  $A \cdot x = b$ .

$$A = \begin{pmatrix} 1 & & & & \\ L_{21} & 1 & & & \\ L_{31} & L_{32} & 1 & & \\ \dots & \dots & \dots & 1 & \\ L_{m1} & \dots & \dots & L_{m,m-1} & 1 \end{pmatrix} \times \begin{pmatrix} U_{11} & U_{12} & \dots & \dots & U_{1m} \\ & U_{22} & U_{23} & \dots & \dots \\ & & U_{33} & \dots & \dots \\ & & & \dots & \dots \\ & & & & U_{mm} \end{pmatrix}. \quad (5)$$

Хорошим свойством LU декомпозиции является то, что для хранения опорных точек достаточно  $O(m)$  требуемой памяти, так как элементы матрицы  $L$  выше главной диагонали равны нулю (сама матрица  $L$  имеет единичную диагональ, в соответствии

с классической схемой нормализации), поэтому должны быть сохранены только субдиагональные элементы. Точно так же с матрицей  $U$  – элементы ниже главной диагонали равны нулю, поэтому должны быть сохранены только главная диагональ и супердиагональные элементы (6).

$$A \rightarrow \begin{pmatrix} U_{11} & U_{12} & U_{13} & \dots & U_{1m} \\ L_{21} & U_{22} & U_{23} & \dots & U_{2m} \\ L_{31} & L_{32} & U_{33} & \dots & U_{3m} \\ \dots & \dots & \dots & \dots & \dots \\ L_{m1} & \dots & \dots & L_{m,m-1} & U_{mm} \end{pmatrix}. \quad (6)$$

Таким образом, LU представление матрицы  $A$  занимает столько же места, сколько исходная матрица. Исходную матрицу не требуется хранить, так как вся информация о ней расположена в полученных треугольных  $L$  и  $U$  матрицах.

Различают две реализации алгоритма: наивную и блочно-матричную факторизацию.

Наивная реализация LU разложения плохо масштабируется в зависимости от размера СЛАУ. Когда размерность матрицы  $A$  превышает  $64 \times 64$ , она не вписывается в типичную кэш-память ЦП L1, вследствие чего алгоритм затрачивает большую часть времени исполнения на ожидание поступления данных из кэш-памяти L2/L3 или ОЗУ.

Блочно-матричный подход является лучшим решением для решения СЛАУ больших размерностей. При его применении большая матрица разбивается на мелкие блоки и выполняется пошаговая факторизация. Алгоритм LU разложения при таком подходе будет состоять из нескольких шагов.

На первом шаге от входящей матрицы  $m \times m$  производится отделение ведущей подматрицы  $m \times k$  (7).

$$\left( \begin{array}{c|c} A_1 & \begin{array}{c} A_{21} \\ A_{22} \end{array} \end{array} \right), \quad (7)$$

где  $A_1 \in R^{m \times k}$ ,  $A_{21} \in R^{(m-k) \times k}$ ,  $A_{22} \in R^{(m-k) \times (m-k)}$ .

На следующем шаге производится LU факторизация наименьшей матрицы  $m \times k$ , в то время как остальные части остаются неизменными (8).

$$\left( \begin{array}{c|c} L_1 U_1 & \begin{array}{c} A_{21} \\ A_{22} \end{array} \end{array} \right), \text{ with } L_1 = \begin{pmatrix} L_{11} \\ \dots \\ L_{12} \end{pmatrix}. \quad (8)$$

Сравнительный анализ итерационного метода GMRES с прямым LU разложением

| Размерность | GMRES                | LU                   | Невязка              |
|-------------|----------------------|----------------------|----------------------|
|             | Погрешность          | Погрешность          |                      |
| 100×100     | 2,0116088480075E-12  | 2,11060429475097E-12 | 2,0925483568135E-11  |
| 200×200     | 1,83254249558738E-13 | 5,04210124070489E-13 | 1,3944401189292E-13  |
| 300×300     | 1,95004482511071E-12 | 2,47861082279823E-12 | 3,47103679132488E-10 |
| 400×400     | 2,76235503824824E-13 | 1,1413797766996E-12  | 1,38888900380607E-13 |
| 500×500     | 4,48703350072223E-12 | 1,3246281079378E-11  | 1,64552815817842E-11 |
| 600×600     | 5,9363410406452E-13  | 2,72143384747093E-12 | 7,18314296932476E-14 |
| 700×700     | 1,42739638197509E-12 | 7,78304694683747E-12 | 2,00772731773213E-12 |
| 800×800     | 1,61541324545316E-12 | 8,03706747369854E-12 | 8,83848549904087E-12 |
| 900×900     | 5,80869227195123E-12 | 4,70100645431032E-11 | 3,67883501439792E-12 |
| 1000×1000   | 1,78459029722083E-12 | 1,50275970448393E-11 | 2,58693066967908E-12 |
| 2000×2000   | 8,9116957833455E-11  | 8,4298279402325E-10  | 4,32651248161164E-09 |
| 3000×3000   | 1,02300509221517E-10 | 2,25055203080853E-09 | 3,241424906264E-10   |
| 4000×4000   | 2,11655434842488E-10 | 5,10361178511771E-09 | 1,20999033015323E-09 |
| 5000×5000   | 2,45441089565206E-11 | 7,38593135178026E-10 | 9,10804764941986E-12 |
| 6000×6000   | 4,77423520551148E-11 | 1,13189903713514E-09 | 6,47268016962244E-10 |
| 7000×7000   | 9,69911363063168E-10 | 1,10518722842222E-08 | 1,12652671191427E-07 |
| 8000×8000   | 2,66089964273607E-11 | 1,36072345874094E-09 | 1,36598510280805E-11 |
| 9000×9000   | 4,99308489261675E-11 | 2,23105000993159E-09 | 1,92463156523104E-10 |

Следующий шаг – обновление части матрицы результатами исходной LU. Верхний блок  $A_{21}$  умножается на  $inv(L_{11})$ , а конечная матрица  $A_{22}$  получает более сложное обновление (9).

$$\left( \begin{array}{c|c} L_1 U_1 & \begin{array}{c} A_{21} \\ A_{22} \end{array} \end{array} \right) \rightarrow \left( \begin{array}{c|c} L_1 U_1 & \begin{array}{c} L_{11}^{-1} \cdot A_{21} \\ A_{22} - L_{12} \cdot (L_{11}^{-1} \cdot A_{21}) \end{array} \end{array} \right) \rightarrow \left( \begin{array}{c|c} L_1 U_1 & \begin{array}{c} U_{21} \\ \tilde{A}_{22} \end{array} \end{array} \right). \quad (9)$$

В результате имеется  $k$  столбцов L-фактора в  $L_1$ , первые  $k$  строк U-фактора в матрицах  $U_1$  и  $U_{21}$ , а оставшаяся часть, подлежащая обработке,  $A$ , сохраняется в конечных  $mk$ -строках/столбцах. На этом этапе остается только применить алгоритм к конечным строкам/столбцам  $A$ , пока не получится факторизация полной матрицы.

В целом подход имеет хороший потенциал для распараллеливания, потому что взаимодействия с большими  $m$  и  $k$  могут быть эффективно разделены между различными ядрами.

Частными случаями LU разложения являются: метод прогонки (используется для трехдиагональных матриц), факторизация Дулитла (если у матрицы L на главной диагонали все единицы), факторизация Крота (если у матрицы U на главной диагонали все единицы). Когда  $U = L^T$  (или  $L = U^T$ ), это называется разложением Холецкого.

**Результаты исследования и их обсуждение**

Программно реализовав описанные выше методы, был проведен численный эксперимент, результаты которого представлены в таблице (в соответствии с таблицей).

Для сравнения поведения методов на СЛАУ больших размерностей на каждом шаге сгенерирована произвольная плотная несимметричная матрица квадратной формы  $A$  и массив свободных членов  $b$  (1).

Решение итерационным методом GMRES находилось с заранее заданной точностью  $1E-3$ .

Погрешность решения рассмотренными численными методами [7] для каждой размерности СЛАУ находилась по формуле

$$\Delta = \|A \cdot x - b\|. \quad (10)$$

Для проведения сравнительного анализа была найдена максимальная невязка между решениями, полученными представленными методами:

$$\max \|x_i^{GMRES} - x_i^{LU}\|, i = \overline{1..m}. \quad (11)$$

Полученные результаты могут быть использованы при решении задач, поставленных в работах [8–10].

**Выводы**

Опираясь на результаты невязок, можно сделать вывод, что оба метода сошлись для

сгенерированных СЛАУ всех размерностей. Анализируя значения невязок, можно сказать, что наиболее заметна зависимость решений от сгенерированных коэффициентов в исходной матрице. Зависимость между размерностью исходных СЛАУ и значениями невязок решений не обнаружена.

#### Список литературы

1. Biswa Nath Datta Numerical Linear Algebra and Applications. SIAM, 2010. 529 p.
2. Frank R. Tortellini. Recipes for Linear Algebra Computation: LU Factorization Using C++. Amazon Digital Services LLC, 2013. 47 p.
3. Gene H. Golub Matrix Computations (Johns Hopkins Studies in the Mathematical Sciences). Johns Hopkins University Press, 2013. 784 p.
4. J. Douglas Faires Numerical Methods. Cengage Learning, 2012. 608 p.
5. Raphael Couturier Designing Scientific Applications on GPUs. CRC Press, 2013. 498 p.
6. Silvia Gazzola Silvia Noschese, Paolo Novati, Lothar Reichel. Arnoldi decomposition, GMRES, and preconditioning for linear discrete ill-posed problems. Applied Numerical Mathematics. 2019. № 137. [Electronic resource]. URL: <https://www.sciencedirect.com/science/article/pii/S0168927419300479?via%3Dihub> (date of access: 25.03.2019).
7. Takahiro Katagiri, Pierre-Yves Aquilanti, Serge Petiton A Smart Tuning Strategy for Restart Frequency of GMRES(m) with Hierarchical Cache Sizes. International Conference on Vector and Parallel Processing. Springer, Berlin, Heidelberg. 2018. P. 314–318.
8. Торшина О.А. Формула асимптотики собственных чисел оператора Лапласа-Бельтрами с потенциалом на проективной плоскости // Современные методы теории функций и смежные проблемы: материалы конференции. 2003. С. 258–259.
9. Дубровский В.В., Торшина О.А. Дискретность спектра задачи Неймана // Вестник Магнитогорского государственного университета. 2004. № 5. С. 130–131.
10. Кадченко С.И., Закирова Г.А., Рязанова Л.С., Торшина О.А. Обратная спектральная задача определения неоднородности упругого стержня // Актуальные проблемы современной науки, техники и образования. 2018. Т. 9. № 2. С. 42–45.