



Math-Net.Ru

Общероссийский математический портал

И. К. Марчевский, В. В. Пузикова, Сравнение эффективности решателей разреженных систем линейных алгебраических уравнений на основе методов BiCGStab и FGMRES, *Труды ИСП РАН*, 2018, том 30, выпуск 1, 195–214

DOI: [https://doi.org/10.15514/ISPRAS-2018-30\(1\)-13](https://doi.org/10.15514/ISPRAS-2018-30(1)-13)

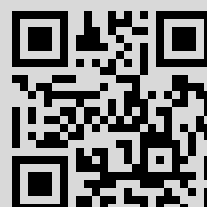
Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочитали и согласны с пользовательским соглашением

<http://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 85.15.114.142

30 июля 2021 г., 17:02:20



Сравнение эффективности решателей разреженных систем линейных алгебраических уравнений на основе методов BiCGStab и FGMRES

И.К. Марчевский <iliamarchevsky@mail.ru>

В.В. Пузикова <vvvp@dms-at.ru>

МГТУ им. Н.Э. Баумана,

105005, Россия, г. Москва, ул. 2-я Бауманская, дом 5

Аннотация. В данной работе проведено сравнение эффективности решателей разреженных систем линейных алгебраических уравнений, построенных на основе одних из наиболее быстрых итерационных методов, метода BiCGStab (метода бисопряженных градиентов со стабилизацией) и метода FGMRES (гибкого метода обобщенных минимальных невязок). В работе представлены оценки трудоемкости выполнения одной итерации рассматриваемых методов. Получено условие, которому должна удовлетворять размерность подпространства Крылова в методе FGMRES для того, чтобы трудоемкость одной итерации данного метода была меньше трудоемкости одной итерации метода BiCGStab. Кроме того представлена модификация метода FGMRES, позволяющая останавливать алгоритм до очередного рестарта в случае достижения заданной точности. На языке C++ на основе представленных алгоритмов методов BiCGStab и FGMRES (в том числе с ILU- и многосеточным предобуславливанием) разработаны решатели разреженных систем линейных алгебраических уравнений. Сравнение эффективности разработанных решателей проводилось на разностных аналогах уравнений Гельмгольца и Пуассона. Системы были взяты из тестовой задачи о моделировании обтекания кругового профиля, совершающего вынужденные поперечные колебания. Разностная схема для решения задачи строится на прямоугольной структурированной сетке интегро-интерполяционным методом LS-STAG – методом погруженных границ с функциями уровня. Вычислительные эксперименты показали, что метод FGMRES на задачах такого класса демонстрирует более высокую скорость сходимости по сравнению с методом BiCGStab. Время проведения расчета при использовании метода FGMRES сократилось более чем в 6.5 раз без предобуславливания и примерно в 3 раза с предобуславливанием. Реализация модифицированного алгоритма метода FGMRES также сравнивалась с аналогичным решателем из библиотеки Intel® Math Kernel Library. Вычислительные эксперименты показали, что разработанная реализация метода FGMRES позволила получить ускорение по сравнению с использованием Intel® MKL в 3.4 раза без предобуславливания и в 1.4 раза при использовании ILU-предобуславливания.

Ключевые слова: разреженные системы линейных алгебраических уравнений; методы крыловского типа; метод BiCGStab; метод FGMRES; предобуславливание; многосеточный метод; уравнение Гельмгольца; уравнение Пуассона; метод погруженных границ LS-STAG; библиотека Intel Math Kernel Library

DOI: 10.15514/ISPRAS-2018-30(1)-13

Для цитирования: Марчевский И.К., Пузикова В.В. Сравнение эффективности решателей разреженных систем линейных алгебраических уравнений на основе методов BiCGStab и FGMRES. Труды ИСП РАН, том 30, вып. 1, 2018 г., стр. 195-214. DOI: 10.15514/ISPRAS-2018-30(1)-13

1. Введение

Численное моделирование используется для решения множества научных и промышленных задач. При этом значительная часть вычислительных ресурсов тратится на решение разреженных систем линейных алгебраических уравнений (СЛАУ) большого размера, возникающих при дискретизации соответствующих дифференциальных или интегро-дифференциальных уравнений. Для решения таких систем, как правило, используются различные итерационные методы [1].

Отметим, что существуют также прямые методы решения разреженных систем, в частности решатель PARDISO [2]. Автор этого метода совместно с коллективом разработчиков реализовал его в программе "PARDISO 5.0.0 Solver Project", которая согласно информации на сайте данного проекта [3] представляет собой потокобезопасное, высокопроизводительное, надежное, эффективное для памяти и простое в использовании программное обеспечение для решения больших разреженных симметричных и несимметричных линейных систем уравнений в многопроцессорных системах с общей памятью и распределенной памятью. Кроме того, данный решатель реализован в библиотеке высокооптимизированных математических алгоритмов Intel® Math Kernel Library (MKL), в документации [4] которой он позиционируется как высокопроизводительный параллельный прямой решатель для систем с разреженными матрицами, оказывающийся эффективнее итерационных решателей при решении систем с числом уравнений менее ста тысяч. Тем не менее, даже при соблюдении указанного ограничения на размер СЛАУ возникают случаи, когда использование прямого решателя PARDISO приводит к увеличению времени счета в 9-10 раз по сравнению с итерационными методами [5]. По этой причине в данной работе далее будут рассматриваться исключительно итерационные методы решения СЛАУ.

Затраты вычислительных ресурсов при проведении моделирования можно существенно сократить за счет выбора для решения СЛАУ итерационных методов, обладающих высокой скоростью сходимости и низкой трудоемкостью на проведение одной итерации. К сожалению, несмотря на то, что возможность выбора итерационного метода для решения СЛАУ имеется

во многих коммерческих программных пакетах, применяемых при решении широкого класса задач механики сплошной среды (ANSYS [6], NASTRAN [7], FLUENT [8], STAR-CD [9] и др.), пользователи крайне редко пользуются ей, предпочитая оставлять те методы решения линейных систем, которые в программном пакете выбраны «по умолчанию». Однако при работе с открытыми свободно распространяемыми пакетами, такими как OpenFOAM [10], пользователям приходится явно указывать методы решения СЛАУ, поскольку установки «по умолчанию» отсутствуют. Пакет Kratos [11] также позволяет выбирать решатели из поддерживаемых внешних библиотек, таких как Intel® MKL [4] и AMGCL [12]. У пакета OpenFOAM существует расширенная версия [13], в которой пользователю доступно больше вариантов решателей СЛАУ. Помимо выбора решателя из числа предлагаемых и настроек его параметров в указанных пакетах существует возможность добавления своего решателя. Решатель при этом может быть как реализован "с нуля", так и базироваться на наработках из таких открытых библиотек, как AMGCL [12], PETSc [14] и др. При этом необходимо не только выбрать эффективный итерационный метод, но и построить на основе его эффективный вычислительный алгоритм.

Целью данной работы является сравнение эффективности решателей разреженных систем линейных алгебраических уравнений, построенных на основе одних из наиболее быстрых итерационных методов, метода BiCGStab (метода бисопряженных градиентов со стабилизацией [15]) и метода FGMRES (гибкого метода обобщенных минимальных невязок [16]), относящихся к проекционным методам крыловского типа [1].

2. Постановка задачи

Рассмотрим систему линейных алгебраических уравнений

$$Ax = b, \quad (1)$$

возникающую при дискретизации уравнения $Lu = f$. Здесь $A \in M(R)_{N_d \times N_d}$;

$x, b \in R^{N_d}$; L – дифференциальный либо интегро-дифференциальный оператор; f – известная функция; u – искомая функция. Будем считать, что матрица A является разреженной (имеет M_d ненулевых элементов, причем M_d много меньше N_d^2), невырожденной ($\det A \neq 0$) и не обладает специальными свойствами (симметрией, положительной определенностью). Для решения систем вида (1) с точностью ε далее будем использовать методы BiCGStab и FGMRES, в том числе в сочетании с ILU-предобуславливателем [17] и многосеточным [18].

В качестве тестовой задачи рассмотрим задачу о моделировании обтекания кругового профиля, совершающего в невозмущенном потоке вынужденные

поперечные колебания по заданному закону [19]. Разностная схема для решения задачи строится на прямоугольной структурированной сетке интегро-интерполяционным методом LS-STAG [20] – методом погруженных границ [21] с функциями уровня [22]. На каждом шаге расчета по времени решение задачи сводится к решению уравнения Гельмгольца

$$(\Delta + k^2)u_1 = f_1 \quad (2)$$

для прогноза скорости u_1 и уравнения Пуассона

$$\Delta u_2 = f_2 \quad (3)$$

для поправки давления u_2 (здесь Δ – оператор Лапласа). Разностные аналоги уравнений (2) и (3) представляют собой системы линейных алгебраических уравнений вида (1), для которых $N_d = 71040$, $M_d = 354128$. Разностный аналог уравнения Пуассона (3) в отличие от разностного аналога уравнения Гельмгольца (2) является плохо обусловленной системой [23], поэтому требует особой тщательности при выборе решателя.

3. Метод BiCGStab

Алгоритм метода BiCGStab с предобуславливанием [1] показан ниже (4).

$$r^0 = b - Ax^0, p^0 = r^0, \forall r_*^0 : (r_*^0, r^0) \neq 0$$

$$\text{for } n = 0, 1, K, \text{ while } (\|r^{n+1}\|_2 \geq \varepsilon), \text{ do}$$

$$y^n = M^{-1}p^n$$

$$\alpha_n = \frac{(r_*^0, r^n)}{(Ay^n, r_*^0)}$$

$$s^n = r^n - \alpha_n Ay^n$$

$$z^n = M^{-1}s^n$$

$$\omega_n = \frac{(Az^n, s^n)}{(Az^n, Az^n)}$$

$$x^{n+1} = x^n + \alpha_n y^n + \omega_n z^n$$

$$r^{n+1} = s^n - \omega_n Az^n$$

$$\beta_n = \frac{\alpha_n (r_*^{n+1}, r_*^0)}{\omega_n (r^n, r_*^0)}$$

$$p^{n+1} = r^{n+1} + \beta_n (p^n - \omega_n Ay^n).$$

Здесь x^n – n -е итерационное приближение к искомому решению x ; $r^n = b - Ax^n$ – вектор невязки на n -й итерации; $p^n = x^{n+1} - x^n$ – вектор коррекции на n -й итерации; M – матрица предобуславливателя [23]. Если матрицу M положить равной единичной матрице соответствующей размерности, приведенный выше алгоритм становится алгоритмом метода BiCGStab без предобуславливания. Явного обращения матрицы M при проведении вычислений не производится: вместо этого решаются системы $My^n = p^n$ и $Mz^n = s^n$. При использовании ILU-предобуславливания для решения этих систем используется неполная LU-факторизация (ILU-факторизация) матрицы A , а при использовании многосеточного предобуславливателя – многосеточный метод. В рамках данной статьи на алгоритмах построения предобуславливателей подробно останавливаться не будем.

Трудоёмкость выполнения одной итерации алгоритма будем определять по числу умножений. При этом затраты на выполнение шагов, связанных с предобуславливанием, учитывать не будем. Таким образом, трудоёмкость одной итерации алгоритма (4) без предобуславливания составляет

$$\lambda_{BiCGStab} = 2M_d + 11N_d. \quad (5)$$

4. Метод FGMRES

Метод FGMRES, как и метод BiCGStab, относится к методам крыловского типа [1]. Основное различие между этими методами заключается в способе построения базиса в подпространстве Крылова: в методе BiCGStab для этого используется биортогонализация Ланцоша, а в FGMRES – ортогонализация Арнольди [1]. Алгоритм метода FGMRES с рестартами через каждые m итераций и гибким предобуславливанием [1] показан ниже (6).

Здесь $\bar{H}_m \in M(R)_{(m+1) \times m}$ – матрица коэффициентов ортогонализации $H_m \in M(R)_{m \times m}$ (верхняя матрица Хессенберга), дополненная строкой $(0 \ K \ 0 \ h_{m+1,m})$; $e^1 = (1 \ 0 \ K \ 0)^T \in R^{m+1}$; M_j – матрица предобуславливателя на j -й итерации. Если матрицу M_j на каждой итерации положить равной единичной матрице соответствующей размерности, приведенный выше алгоритм становится алгоритмом метода FGMRES без предобуславливания (GMRES). Таким образом, отличие алгоритмов с предобуславливанием и без состоит только в одном шаге ($z^j = M_j^{-1}v^j$); при этом явного обращения матрицы M_j при проведении вычислений не производится: вместо этого решается система $M_j z^j = v^j$.

Отметим, что векторы v^1, K, v^m образуют ортогональный базис в подпространстве Крылова размерности m , порожденным вектором v^1 и матрицей A , $K_m = K_m(A, v^1) = \text{span}\{v^1, Av^1, A^2v^1, K, A^{m-1}v^1\}$ [1].

$$\begin{aligned} \text{Start : } & r^0 = b - Ax^0, \beta = \|r^0\|_2, v^1 = r^0 / \beta \\ \text{for } & j = 1, K, m \\ & z^j = M_j^{-1}v^j \\ & w^j = Az^j \\ \text{for } & i = 1, K, j \\ & h_{i,j} = (w^j, v^i) \\ & w^j = w^j - h_{i,j}v^i \\ & h_{j+1,j} = \|w^j\|_2 \\ & v^{j+1} = w^j / h_{j+1,j} \\ & Z_m = \begin{bmatrix} z^1 & K & z^m \end{bmatrix}, \bar{H}_m = \{h_{i,j}\} \\ & y^m = \arg \min_y \|\beta e^1 - \bar{H}_m y\|_2 \\ & x^m = x^0 + Z_m y^m \\ \text{if } & (\|b - Ax^m\|_2 \geq \varepsilon) \\ & x^0 = x^m \\ & \text{goto Start.} \end{aligned} \quad (6)$$

Из-за того, что в алгоритме (6) критерий останова проверяется через каждые m итераций, количество итераций при использовании данного алгоритма всегда будет кратно m . Средняя трудоёмкость одной итерации алгоритма (6) без предобуславливания составляет

$$\begin{aligned} \lambda_{FGMRES} &= \frac{\sum_{j=1}^m \left(M_d + 2N_d + \sum_{i=1}^j 2N_d \right)}{m} = \frac{(M_d + 2N_d)m + 2N_d \sum_{j=1}^m j}{m} = \\ &= \frac{(M_d + 2N_d)m + N_d m(m+1)}{m} = M_d + (3+m)N_d. \end{aligned} \quad (7)$$

Из формул (5) и (7) следует, что трудоемкость одной итерации метода FGMRES оказывается меньше трудоемкости одной итерации метода BiCGStab при выполнении следующего условия:

$$m < \frac{M_d}{N_d} + 8. \quad (8)$$

Поскольку при построении разностного аналога в методе LS-STAG используется пятиточечный шаблон дискретизации, можно считать, что $M_d \approx 5N_d$. Тогда из условия (8) получается, что число m в алгоритме (6) должно быть меньше 13.

5. Модификация алгоритма метода FGMRES

Недостаток алгоритма (6) заключается в том, что критерий останова проверяется не на каждой итерации алгоритма. Из-за этого получается, что могут выполняться "лишние" итерации. Построим модификацию алгоритма (6), позволяющую устранить данный недостаток.

Алгоритм (6) требует решения линейной задачи наименьших квадратов $y^m = \arg \min_y \|\beta e^1 - \bar{H}_m y\|_2$, т. е. решения системы

$$\bar{H}_m y = \beta e^1. \quad (9)$$

Для этого используем QR -разложение, построенное при помощи метода вращений [1]: умножим (9) слева на матрицу $Q_m = \Omega_m \cdot K \cdot \Omega_1$, $Q_m, \Omega_i \in M(R)_{(m+1) \times (m+1)}$, $i = \overline{1, m}$, где Ω_i – матрица вращений Гивенса: i -й и $(i+1)$ -й диагональные элементы этой матрицы равны $c_i = h_{i+1,i}/\Delta_i$, $\Delta_i = \sqrt{h_{i,i}^2 + h_{i+1,i}^2}$, остальные диагональные элементы – единицы; $(i+1)$ -й элемент i -й строки равен $s_i = h_{i,i}/\Delta_i$, i -й элемент $(i+1)$ -й строки равен $(-s_i)$, остальные элементы – нули. В итоге получаем систему

$$\bar{R}_m y = \bar{g}^m. \quad (10)$$

Здесь $\bar{R}_m = Q_m \bar{H}_m \in M(R)_{(m+1) \times m}$, $g^m = Q_m(\beta e^1) = (\gamma_1 \ K \ \gamma_{m+1})^T \in R^{m+1}$. После удаления из (10) последнего уравнения получаем

$$R_m y_m = g_m, \quad (11)$$

где $R_m \in M(R)_{m \times m}$ – верхнетреугольная матрица, что позволяет решить эту систему при помощи обратного хода метода Гаусса. Полученное решение y^m системы (11) также является решением задачи (9) и позволяет вычислить

итерационное приближение x^m из (6). При этом для нормы вектора невязки из (6) справедливо равенство $\|b - Ax^m\|_2 = |\gamma_{m+1}|$ [1]. Благодаря этому свойству критерий останова можно проверять на каждой итерации без решения системы (11), и, если он выполняется, вычислять решение, не дожидаясь m -й итерации. При этом удобно хранить матрицы Z_m и H_m по столбцам (packed storage). Для того, чтобы пересчитывать на j -й итерации элементы j -го столбца матрицы H_m и элементы правой части системы (11) g_j и $g_{j+1} = \gamma$, также необходимо хранить векторы косинусов и синусов $c = (c_1 \ K \ c_m)^T, s = (s_1 \ K \ s_m)^T \in R^m$.

Тогда алгоритм (6) можно переписать следующим образом так, как показано ниже (13).

Средняя трудоемкость одной итерации алгоритма (13) без предобуславливания составляет совпадает с оцениваемой по формуле (7) для итераций, предшествующих рестарту, и равна

$$\lambda_{FGMRES-mod} = M_d + (3+k)N_d \quad (12)$$

для итераций, после которых не было рестарта (т.е. сработал критерий останова). Поскольку $k \leq m$, получаем, что $\lambda_{FGMRES-mod} \leq \lambda_{FGMRES}$. Соответственно, если размерность m подпространства Крылова удовлетворяет условию (8), то трудоемкость одной итерации модифицированного алгоритма метода FGMRES (13) оказывается меньше трудоемкости одной итерации метода BiCGStab, также как и трудоемкость одной итерации основного алгоритма метода FGMRES (6). При этом благодаря проверке критерия останова на каждой итерации алгоритма (13) вычислительные затраты на решение СЛАУ уменьшаются по сравнению с использованием исходного алгоритма (6).

6. Вычислительные эксперименты

Для описанной выше тестовой задачи о моделировании обтекания кругового профиля, совершающего в невозмущенном потоке вынужденные поперечные колебания по заданному закону [19], были проведены расчеты, состоящие из 301 шага по времени. Таким образом, в рамках каждого расчета решалась серия СЛАУ, состоящая из 602 разностных аналогов уравнений Гельмгольца (2) и 301 разностного аналога уравнений Пуассона (3). Системы решались с точностью $\varepsilon = 10^{-6}$.

Start : $r^0 = b - Ax^0$, $\beta = \|r^0\|_2$, $w^0 = r^0$, $\gamma = \beta$, $k = m$

for $j = 1, K, m$

$$v^j = w^{j-1} / \beta$$

$$z^j = M_j^{-1} v^j$$

$$w^j = Az^j$$

for $i = 1, K, j$

$$h_{i,j} = (w^j, v^i)$$

$$w^j = w^j - h_{i,j} v^i$$

for $i = 1, K, j-1$

$$\tilde{h} = h_{i,j}, h = h_{i+1,j}$$

$$h_{i,j} = c_i \tilde{h} + s_i h$$

$$h_{i+1,j} = -s_i \tilde{h} + c_i h$$

$$\tilde{h} = h_{j,j}, \beta = \|w^j\|_2$$

$$h_{j,j} = \sqrt{\tilde{h}^2 + \beta^2}$$

$$c_j = \beta / h_{j,j}$$

$$s_j = \tilde{h} / h_{j,j}$$

$$g_j = \mathcal{R} c_j$$

$$\gamma = -\mathcal{R} s_j$$

$$\text{if } (|\gamma| < \varepsilon)$$

$$k = j$$

$$j = m + 1$$

$$Z_k = \begin{bmatrix} z^1 & K & z^k \end{bmatrix}, H_k = \{h_{i,j}\}, g^k = (g_1 \quad K \quad g_k)^T$$

$$x^m = x^0 + Z_k H_k^{-1} g^k$$

$$\text{if } (|\gamma| \geq \varepsilon)$$

$$x^0 = x^m$$

goto Start.

(13)

Табл. 1. Минимальное, среднее и максимальное число итераций, а также среднее квадратичное отклонение (СКО) числа итераций при решении серии разностных аналогов уравнений Гельмгольца разработанными решателями
Table 1. The minimum, average and maximum iterations number, as well as the standard deviation (SD) of iterations number when solving a series of the Helmholtz equations difference analogues by developed solvers

Метод	Минимальное	Среднее	Максимальное	СКО
BiCGStab	1	24	58	5
FGMRES	0	33	52	5
BiCGStab+ILU	1	2	2	0
FGMRES +ILU	0	2	2	0
BiCGStab+MG	1	2	4	0
FGMRES +MG	0	2	4	0

Табл. 2. Минимальное, среднее и максимальное число итераций, а также среднее квадратичное отклонение (СКО) числа итераций при решении серии разностных аналогов уравнений Пуассона разработанными решателями
Table 2. The minimum, average and maximum iterations number, as well as the standard deviation (SD) of iterations number when solving a series of the Poisson equations difference analogues by developed solvers

Метод	Минимальное	Среднее	Максимальное	СКО
BiCGStab	88	588	1671	300
FGMRES	98	105	107	1
BiCGStab+ILU	16	105	254	49
FGMRES +ILU	89	92	93	0
BiCGStab+MG	6	24	177	13
FGMRES +MG	5	7	13	1

Напомним, что размерность систем равнялась $N_d = 71040$, а матрицы систем содержали $M_d = 354128$ ненулевых элементов. В решателях реализованы алгоритмы (6) и (13). Размерность подпространства Крылова в решателях, основанных на методе FGMRES, была выбрана с учетом условия (8) и равнялась $m = 12$. Методы использовались как без предобуславливания, так и с ILU- и многосеточным предобуславливанием.

В табл. 1 и 2 собраны статистические сведения о сходимости разработанных решателей при решении серий разностных аналогов уравнений Гельмгольца и Пуассона соответственно: указаны минимальное, среднее и максимальное число итераций, а также среднее квадратичное отклонение (СКО) числа

итераций. Гистограммы распределения числа итераций для нескольких наиболее интересных случаев (решение серий разностных аналогов уравнений Гельмгольца и Пуассона рассматриваемыми методами без предобуславливания, а также уравнения Пуассона решателями с ILU-предобуславливанием) представлены на рис. 1–3.

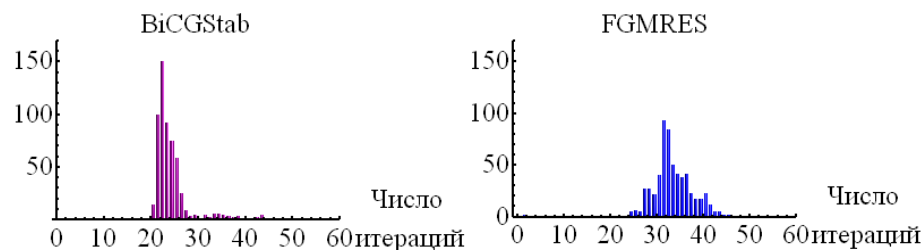


Рис. 1. Гистограмма распределения числа итераций при решении серии разностных аналогов уравнений Гельмгольца методами BiCGStab (слева) и FGMRES (справа) без предобуславливания

Fig. 1. A histogram of the iterations number distribution when solving a series of the Helmholtz equations difference analogues by the methods BiCGStab (left) and FGMRES (right) without preconditioning

Из табл. 1 и рис. 1 видно, что распределение числа итераций при решении серии разностных аналогов уравнений Гельмгольца (1) имеет примерно одинаковые статистические характеристики как в случае использования решателей, основанных на методе BiCGStab, так и в случае использования решателей, основанных на методе FGMRES. При этом необходимо отметить, что итерации модифицированного алгоритма метода FGMRES (12) менее дорогостоящие с вычислительной точки зрения чем итерации метод BiCGStab, поскольку размерность подпространства Крылова m в используемой реализации алгоритма метода FGMRES удовлетворяет полученному выше условию (8).

Иная картина наблюдается при решении серии плохо обусловленных систем, являющихся разностными аналогами уравнений Пуассона (3): видно, что при решении систем решателями, основанными на методе BiCGStab, число итераций изменяется в очень широком диапазоне и имеет значительное среднее квадратичное отклонение (табл. 2). При решении этих же систем решателями, основанными на методе FGMRES, напротив, число итераций мало зависит от шага по времени и имеет СКО, близкое к нулю (рис. 2 и 3). При этом среднее число итераций при использовании метода FGMRES оказывается в 5.60 раз меньше при сравнении методов без предобуславливания, в 1.14 раз меньше при сравнении методов с использованием ILU-предобуславливания и в 3.43 раза – с использованием многосеточного предобуславливателя. Таким образом, за наименьшее число итераций

рассматриваемые разностные аналоги уравнения Пуассона позволяет решать метод FGMRES с многосеточным предобуславливанием.

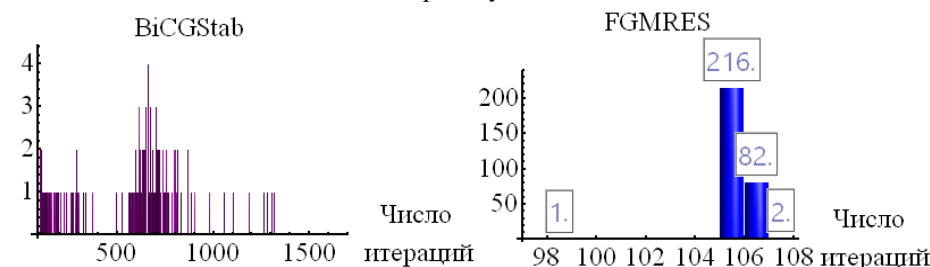


Рис. 2. Гистограмма распределения числа итераций при решении серии разностных аналогов уравнений Пуассона методами BiCGStab (слева) и FGMRES (справа) без предобуславливания

Fig. 2. A histogram of the iterations number distribution when solving a series of the Poisson equations difference analogues by the methods BiCGStab (left) and FGMRES (right) without preconditioning

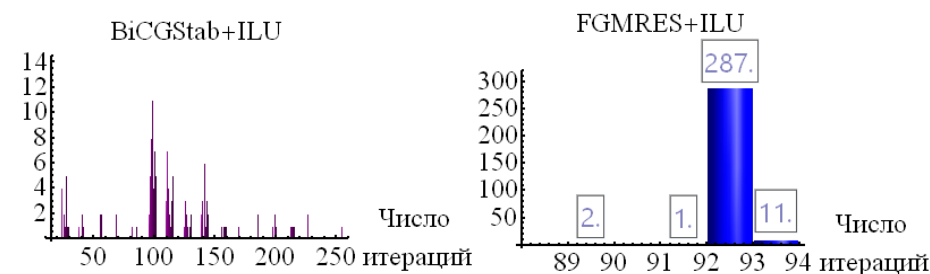


Рис. 3. Гистограмма распределения числа итераций при решении серии разностных аналогов уравнений Пуассона методами BiCGStab (слева) и FGMRES (справа) с ILU-предобуславливанием

Fig. 3. A histogram of the iterations number distribution when solving a series of the Poisson equations difference analogues by the methods BiCGStab (left) and FGMRES (right) with ILU preconditioning

Для разностного аналога уравнения Пуассона, который решается в рассматриваемой тестовой задаче на первом шаге по времени на рис. 4 в логарифмическом масштабе показано убывание нормы невязки при использовании разработанных решателей. Видно, что при использовании решателей, основанных на методе FGMRES, норма невязки убывает строго монотонно, причем после очередного рестарта скорость убывания резко увеличится. При использовании решателей, основанных на методе BiCGStab, напротив, норма невязки убывает не монотонно: на итерациях с большими номерами (для рассматриваемого примера – больше 50)

наблюдаются резкие увеличения нормы невязки, характер расположения точек на графике в некоторые моменты расчета становится хаотичным. Эти эффекты наблюдаются для методов FGMRES и BiCGStab как при решении системы без предобуславливания, так и при решении рассматриваемой СЛАУ с реализованными предобуславливателями. Из всего этого можно сделать вывод, что сам метод FGMRES действует не только на высокочастотные компоненты ошибки, убираемые на первых итерациях, но и на низкочастотные, в то время как метод BiCGStab обладает преимущественно сглаживающими свойствами [23].

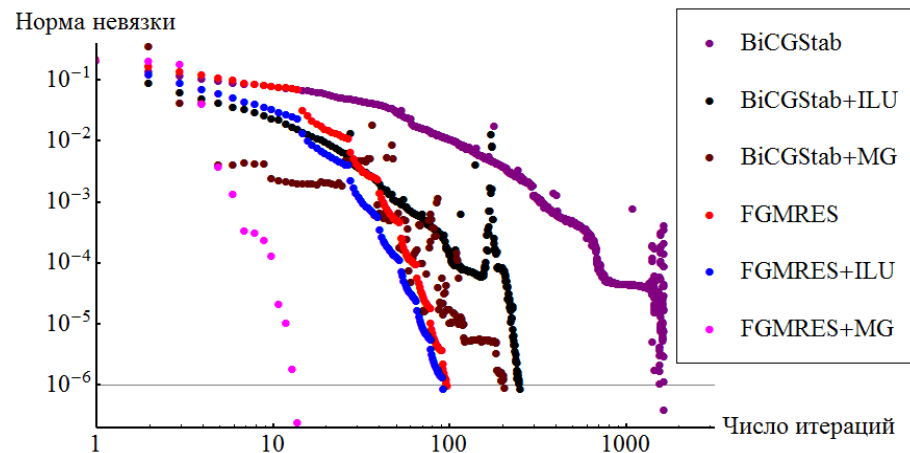


Рис. 4. Убывание нормы невязки при решении разностного аналога уравнения Пуассона разработанными решателями

Fig. 4. Decrease in the residual norm when solving the Poisson equation difference analogue by the developed solvers

Таким образом, метод FGMRES на задачах рассматриваемого класса демонстрирует более высокую скорость сходимости по сравнению с методом BiCGStab, причем особенно сильно это наблюдается при решении плохо обусловленных систем линейных алгебраических уравнений.

Определим, какое влияние оказывает использование того или иного решателя на общую продолжительность решения рассматриваемой тестовой задачи. На диаграмме на рис. 5 указано время проведения всех расчетов в мс. Вычислительные эксперименты проводились на рабочей станции, построенной на платформе Intel H81 с использованием двухядерного процессора Intel Core i3-4350T (Haswell) с поддержкой HyperThreading (4 логических ядра), работающем на частоте 3100 МГц. Станция оснащена 8 Гбайт оперативной памяти DDR3-1333, SSD-накопителем Crucial объемом 128 Гбайт и жестким диском Seagate объемом 1 Тбайт.

Из рис. 5 видно, что время проведения расчета при использовании метода FGMRES сократилось более чем в 6.5 раз без предобуславливания и примерно в 3 раза с предобуславливанием по сравнению с использованием аналогичных решателей, основанных на методе BiCGStab. При этом несмотря на то, что использование многосеточного предобуславливания позволяет решать рассматриваемые разностные аналоги уравнений Пуассона за наименьшее число итераций, использование данного предобуславливателя и для разностных аналогов уравнений Гельмгольца приводит к небольшому увеличению времени счета по сравнению с использованием ILU-предобуславливания, поскольку сокращения числа итераций при этом не происходит, а вычислительные затраты на предобуславливание увеличиваются.

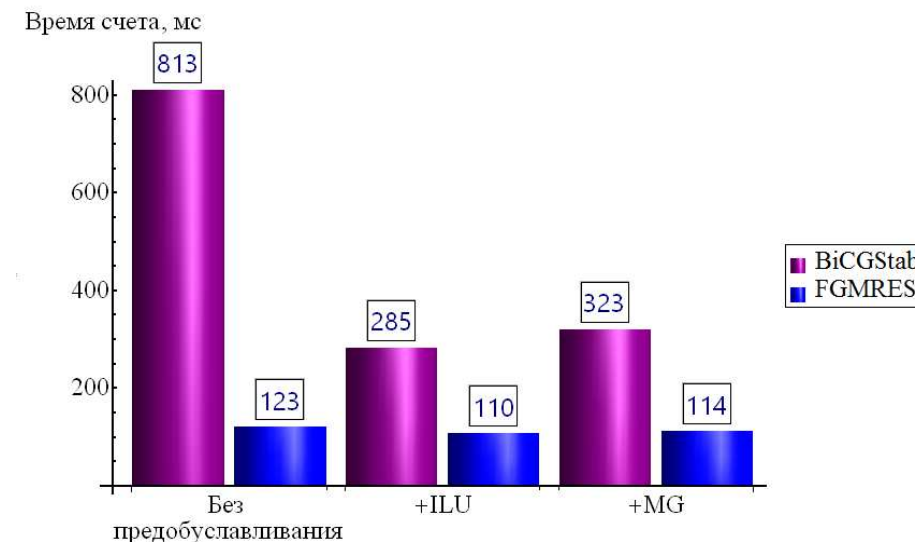


Рис. 5. Время счета при использовании различных решателей

Fig. 5. Computational time at different solvers usage

Проверим также, насколько правильным был выбор размерности подпространства Крылова m в используемой реализации алгоритма метода FGMRES исходя из полученного выше условия (8). Для этого сравним время счета при использовании решателей на основе метода FGMRES при $m = 12$ и $m = 20$. Из рис. 6 видно, что увеличение m действительно приводит к росту вычислительных затрат на проведение одной итерации и, соответственно, увеличению времени счета. Наиболее сильно это наблюдается при использовании методов без предобуславливания, поскольку при этом для

решения СЛАУ выполняется больше итераций. В этом случае время счета различается почти в 2 раза.

Также сравним разработанную реализацию модифицированного алгоритма метода FGMRES (12) с аналогом из библиотеки высокооптимизированных математических алгоритмов Intel® Math Kernel Library (MKL). В данной библиотеке не реализован многосеточный предобуславливатель, поэтому будем сравнивать только работу решателей без предобуславливания и с ILU-предобуславливанием. Из рис. 6. видно, что разработанная реализация модифицированного алгоритма метода FGMRES при $m = 12$ позволила получить ускорение по сравнению с использованием Intel® MKL в 3.4 раза без предобуславливания и в 1.4 раза при использовании ILU-предобуславливания.

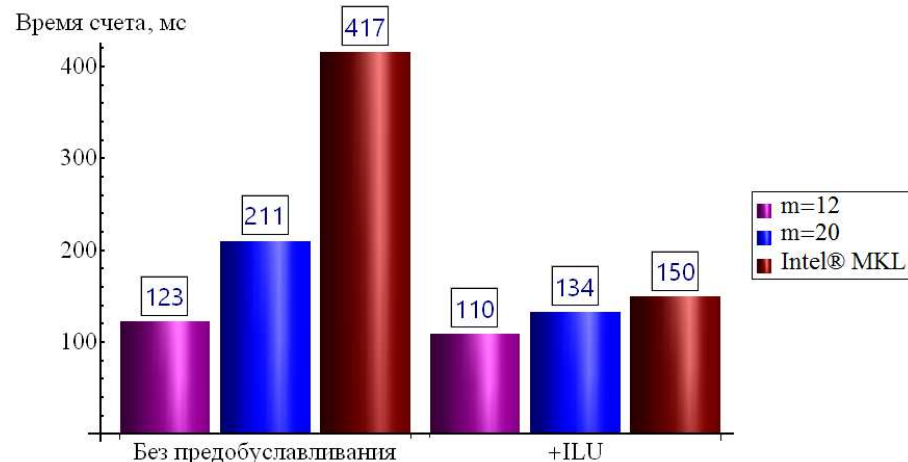


Рис. 6. Время счета при использовании различных реализаций метода FGMRES
Fig. 6. Computational time at different FGMRES method implementations usage

Выше было продемонстрировано, что использование многосеточного предобуславливания позволяет значительно сократить вычислительные затраты при решении разностных аналогов уравнений Пуассона, но при этом менее дорогостоящее с вычислительной точки зрения ILU-предобуславливание позволяет решать разностные аналоги уравнений Гельмгольца за то же число итераций, что и многосеточное предобуславливание (табл. 1). Поэтому логично при решении рассматриваемой тестовой задачи для решения разностных аналогов уравнений Гельмгольца использовать решатели с ILU-предобуславливанием, а для решения разностных аналогов уравнений Пуассона – решатели с многосеточным предобуславливанием. Результаты таких вычислительных экспериментов представлены на рис. 7.

Из рис. 7 видно, что применение различных методов предобуславливания позволило сократить время счета при использовании метода FGMRES

примерно в 2 раза по сравнению с применением одного и того же предобуславливателя для решения всех СЛАУ. При этом выбор значения размерности подпространства Крылова m , удовлетворяющего условию (8) позволяет получить ускорение примерно в 1.4 раза по сравнению с использованием больших значений m . При этом ускорение по сравнению с использованием аналогичных решателей, основанных на методе BiCGStab составляе примерно 4.2 раза.

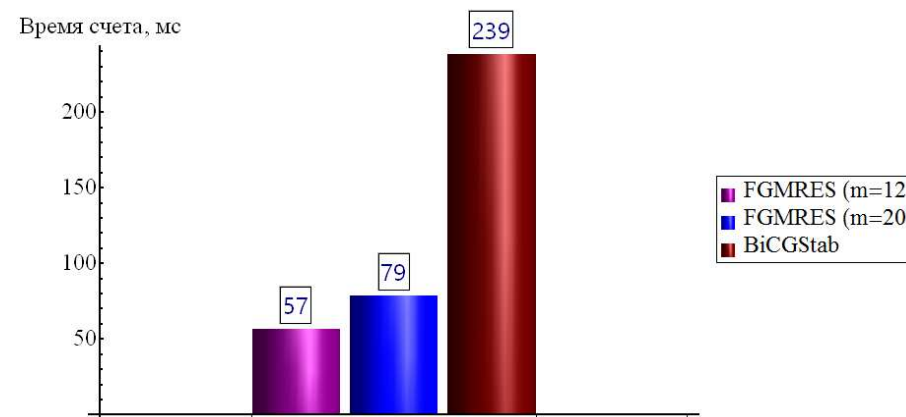


Рис. 7. Время счета при использовании решателей с ILU-предобуславливанием для разностных аналогов уравнений Гельмгольца и решателей с многосеточным предобуславливанием для разностных аналогов уравнений Пуассона
Fig. 7. Computational time when solving the Helmholtz equations difference analogues by the methods with ILU preconditioning and the Poisson equations difference analogues by the methods with multigrid preconditioning

7. Заключение

Исследованы решатели разреженных систем линейных алгебраических уравнений, построенные на основе методов BiCGStab и FGMRES. Получены оценки трудоемкости выполнения одной итерации этих методов. Кроме того, получено условие, которому должна удовлетворять размерность подпространства Крылова в методе FGMRES для того, чтобы трудоемкость одной итерации данного метода была меньше трудоемкости одной итерации метода BiCGStab. Представлена модификация метода FGMRES, позволяющая останавливать алгоритм до очередного рестарта в случае достижения заданной точности. На основе представленных алгоритмов разработаны решатели разреженных систем линейных алгебраических уравнений. Разработанные решатели позволяют решать системы как без

предобуславливания, так и использовать ILU- и многосеточное предобуславливание.

Сравнение эффективности разработанных решателей проводилось на разностных аналогах уравнений Гельмгольца и Пуассона из тестовой задачи о моделировании обтекания кругового профиля, совершающего вынужденные поперечные колебания. Вычислительные эксперименты показали, что метод FGMRES на задачах такого класса демонстрирует более высокую скорость сходимости по сравнению с методом BiCGStab. Время проведения расчета при использовании метода FGMRES сократилось более чем в 6.5 раз без предобуславливания и примерно в 3 раза с предобуславливанием. Для сравнения эффективности также использовались аналогичные алгоритмы, реализованные в библиотеке высокооптимизированных математических алгоритмов Intel® Math Kernel Library. Расчеты показали, что разработанная реализация модифицированного алгоритма метода FGMRES позволила получить ускорение по сравнению с использованием Intel® MKL в 3.4 раза без предобуславливания и в 1.4 раза при использовании ILU-предобуславливания.

Работа выполнена за счет гранта Российского научного фонда (проект №17-79-20445).

Список литературы

- [1]. Saad Y. Iterative Methods for Sparse Linear Systems. N.Y.: PWS Publ., 1996. 547 p.
- [2]. Schenk O., Gartner K. Solving unsymmetric sparse systems of linear equations with PARDISO // J. of Future Generation Computer Systems. 2004. № 20. P. 475–487.
- [3]. PARDISO. URL: <http://www.pardiso-project.org/> (accessed: 15.10.2017).
- [4]. Intel® Math Kernel Library – Documentation. URL: <https://software.intel.com/en-us/articles/intel-math-kernel-library-documentation> (accessed: 15.10.2017).
- [5]. Марчевский И.К., Пузикова В.В. Исследование эффективности распараллеливания вычислений при моделировании течений вязкой несжимаемой среды методом LS-STAG на системах с общей памятью. Вычислительные методы и программирование. 2015. Т. 16. С. 595–606.
- [6]. ANSYS All Products. URL: <http://www.ansys.com/products/all-products> (accessed: 15.10.2017).
- [7]. MSC Nastran. URL: <http://www.mssoftware.ru/products/msc-nastran> (accessed: 15.10.2017).
- [8]. ANSYS Fluent. URL: <http://www.ansys.com/Products/Fluids/ANSYS-Fluent> (accessed: 15.10.2017).
- [9]. STAR-CD. URL: <https://mdx.plm.automation.siemens.com/star-cd> (accessed: 15.10.2017).
- [10]. OpenFOAM® Documentation. URL: <https://www.openfoam.com/documentation/> (accessed: 15.10.2017).
- [11]. What is KRATOS. URL: http://www.cimne.com/kratos/about_whatiskratos.asp (accessed: 15.10.2017).
- [12]. AMGCL documentation. URL: <http://amgcl.readthedocs.io/en/latest/> (accessed: 15.10.2017).

- [13]. Foam-extend. URL: <https://sourceforge.net/projects/foam-extend/> (accessed: 15.10.2017).
- [14]. Portable, Extensible Toolkit for Scientific Computation. URL: <https://www.mcs.anl.gov/petsc/> (accessed: 15.10.2017).
- [15]. Van der Vorst H.A. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for solution of non-symmetric linear systems. *SIAM J. Sci. Stat. Comp.* 1992. № 2. P. 631–644.
- [16]. Saad Y. A flexible inner-outer preconditioned GMRES algorithm. *SIAM J. Sci. Comput.* 1993. № 14. P. 461–469.
- [17]. Баландин М.Ю., Шурина Э.П. Методы решения СЛАУ большой размерности. Новосибирск: Изд-во НГТУ, 2000. 70 с.
- [18]. Wesseling P. An introduction to multigrid methods. Chichester: John Wiley & Sons Ltd., 1991. 284 p.
- [19]. Guilmineau E., Queutey P. A numerical simulation of vortex shedding from an oscillating circular. *J. Fluid Struct.* 2002. № 16. P. 773–794.
- [20]. Cheny Y., Botella O. The LS-STAG method: A new immersed boundary/level-set method for the computation of incompressible viscous flows in complex moving geometries with good conservation properties. *J.Comp. Phys.* 2010. №229. P.1043-1076.
- [21]. Mittal R., Iaccarino G. Immersed boundary methods. *Annu. Rev. Fluid Mech.* 2005. № 37. P. 239–261.
- [22]. Osher S., Fedkiw R.P. Level set methods and dynamic implicit surfaces. N. Y.: Springer, 2003. 273 p.
- [23]. Марчевский И.К., Пузикова В.В. Анализ эффективности итерационных методов решения систем линейных алгебраических уравнений. Математическое моделирование и численные методы. 2014. № 4. С. 37–52.

The efficiency comparison of solvers for sparse linear algebraic equations systems based on the BiCGStab and FGMRES methods

I.K. Marchevsky <iliamarchevsky@mail.ru>

V.V. Puzikova <vvp@dms-at.ru>

BMSTU, 5 2nd Baumanskaya st., Moscow, Russian Federation, 105005

Abstract. The efficiency comparison of solvers for sparse linear algebraic equations systems based on one of the fastest iterative methods, the BiCGStab method (bi-conjugate gradient method with stabilization), and the FGMRES method (flexible method of generalized minimal residuals) is presented in this study. Estimates of computational cost per one iteration are presented for the considered methods. The condition imposed on the Krylov subspace dimensionality in the FGMRES is obtained. When this condition is fulfilled, the computational cost per one iteration of the FGMRES method is less than the computational cost per one iteration of the BiCGStab. In addition, the FGMRES modification, which allows to stop the algorithm before the next restart in case of achieving the specified accuracy, is presented. Solvers on the basis of presented the BiCGStab and FGMRES methods algorithms including ILU and multigrid preconditioning are developed on the C++ language for sparse linear algebraic equations systems. The efficiency comparison of developed solvers was carried out on the difference analogs of the Helmholtz and Poisson equations. The systems

were taken from the test problem about simulation of the flow around a circular profile, which makes forced transverse oscillations. The difference scheme for the problem solution is constructed by the LS-STAG method (immersed boundaries method with level-set functions). Computational experiments showed that the FGMRES demonstrates a higher convergence rate on problems of this class in comparison with the BiCGStab. The FGMRES usage allowed to reduce the computation time by more than 6.5 times without preconditioning and more than 3 times with preconditioning. The implementation of the modified FGMRES algorithm was also compared with a similar solver from the Intel® Math Kernel Library. Computational experiments showed that the developed FGMRES implementation allowed to obtain acceleration in comparison with Intel® MKL by 3.4 times without preconditioning and by 1.4 times with ILU-preconditioning.

Keywords: sparse linear algebraic equations systems; Krylov-space methods; the BiCGStab method; the FGMRES method; preconditioner; multigrid method; Helmholtz equation; Poisson equation; the LS-STAG immersed boundary method; Intel® Math Kernel Library

DOI: 10.15514/ISPRAS-2018-30(1)-13

For citation: Marchevsky I.K., Puzikova V.V. The efficiency comparison of solvers for sparse linear algebraic equations systems based on the BiCGStab and FGMRES methods. Trudy ISP RAN/Proc. ISP RAS, vol. 30, issue 1, 2018, pp. 195-214 (in Russian). DOI: 10.15514/ISPRAS-2018-30(1)-13

References

- [1]. Saad Y. Iterative Methods for Sparse Linear Systems. N.Y.: PWS Publ., 1996. 547 p.
- [2]. Schenk O., Gartner K. Solving unsymmetric sparse systems of linear equations with PARDISO // J. of Future Generation Computer Systems. 2004. № 20. P. 475–487.
- [3]. PARDISO. URL: <http://www.pardiso-project.org/> (accessed: 15.10.2017).
- [4]. Intel® Math Kernel Library – Documentation. URL: <https://software.intel.com/en-us/articles/intel-math-kernel-library-documentation> (accessed: 15.10.2017).
- [5]. Marchevsky I.K., Puzikova V.V. Efficiency investigation of computation parallelization for viscous incompressible flow simulation on systems with shared memory. *Vychislitel'nye metody i programmirovaniye* [Computational methods and programming], 2015. Vol. 16. P. 595–606 (in Russian)
- [6]. ANSYS All Products. URL: <http://www.ansys.com/products/all-products> (accessed: 15.10.2017).
- [7]. MSC Nastran. URL: <http://www.mssoftware.ru/products/msc-nastran> (accessed: 15.10.2017).
- [8]. ANSYS Fluent. URL: <http://www.ansys.com/Products/Fluids/ANSYS-Fluent> (accessed: 15.10.2017).
- [9]. STAR-CD. URL: <https://mdx.plm.automation.siemens.com/star-cd> (accessed: 15.10.2017).
- [10]. OpenFOAM® Documentation. URL: <https://www.openfoam.com/documentation/> (accessed: 15.10.2017).
- [11]. What is KRATOS. URL: http://www.cimne.com/kratos/about_whatiskratos.asp (accessed: 15.10.2017).

- [12]. AMGCL documentation. URL: <http://amgcl.readthedocs.io/en/latest/> (accessed: 15.10.2017).
- [13]. foam-extend. URL: <https://sourceforge.net/projects/foam-extend/> (accessed: 15.10.2017).
- [14]. Portable, Extensible Toolkit for Scientific Computation. URL: <https://www.mcs.anl.gov/petsc/> (accessed: 15.10.2017).
- [15]. Van der Vorst H.A. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for solution of non-symmetric linear systems. SIAM J. Sci. Stat. Comp. 1992. № 2. P. 631–644.
- [16]. Saad Y. A flexible inner-outer preconditioned GMRES algorithm. SIAM J. Sci. Comput. 1993. № 14. P. 461–469.
- [17]. Balandin M.Yu., Shurina E.P. Metody resheniya SLAU bol'shoj razmernosti [Methods for large SLAE solving] Novosibirsk: NGTU publ., 2000. 70 p. (in Russian)
- [18]. Wesseling P. An introduction to multigrid methods. Chichester: John Wiley & Sons Ltd., 1991. 284 p.
- [19]. Guilmineau E., Queutey P. A numerical simulation of vortex shedding from an oscillating circular. J. Fluid Struct. 2002. № 16. P. 773–794.
- [20]. Cheny Y., Botella O. The LS-STAG method: A new immersed boundary/level-set method for the computation of incompressible viscous flows in complex moving geometries with good conservation properties. J.Comp. Phys.2010. №229. P.1043-1076.
- [21]. Mittal R., Iaccarino G. Immersed boundary methods. Annu. Rev. Fluid Mech. 2005. № 37. P. 239–261.
- [22]. Osher S., Fedkiw R.P. Level set methods and dynamic implicit surfaces. N. Y.: Springer, 2003. 273 p.
- [23]. Marchevsky I.K., Puzikova V.V. The efficiency analysis of iterative methods for solving linear algebraic equations systems. *Matematicheskoe modelirovaniye i chislennyye metody* [Mathematical modeling and numerical methods], 2014. № 4. P. 37-52 (in Russian)