

Использование полинейного рекуррентного метода с переменным параметром компенсации для решения разностных эллиптических уравнений

Л. Н. ФОМИНА

Кемеровский государственный университет, Россия
e-mail: lubafomina@mail.ru

Приводится алгоритм итерационного полинейного рекуррентного метода решения пятидиагональной системы линейных алгебраических уравнений (СЛАУ), возникающей при разностной аппроксимации двумерных уравнений динамики вязкой жидкости и тепломассопереноса. Тестовая задача решается модифицированным полинейным и полинейным рекуррентным методами как с постоянным, так и с переменным параметрами компенсации. Анализируется эффективность использования переменного параметра компенсации в сравнении с постоянным.

Ключевые слова: система линейных алгебраических уравнений, итерационный метод решения, переменный параметр компенсации.

Введение

Современные методы решения краевых задач, связанных с исследованием явлений гидродинамики и тепломассопереноса, сводятся, как правило, к разностной аппроксимации многомерных дифференциальных уравнений, что в свою очередь приводит к построению систем линейных алгебраических уравнений, матрицы которых имеют большую размерность и разреженно-упорядоченную структуру [1, 2]. Для одномерных по пространству задач матрицы СЛАУ имеют трехдиагональную структуру, решение таких задач легко находится прямым экономичным методом скалярной прогонки, в то время как для двумерных задач число диагоналей возрастает, как правило, до пяти, что существенно осложняет процедуру решения системы. В первую очередь эти осложнения связаны с необходимостью использования итерационных подходов при построении методов решения подобных СЛАУ [1–4].

Повышение эффективности итерационных методов решения СЛАУ, согласно идее, высказанной еще в монографии [3], определяется скоростью передачи информации от границ расчетной области ко всем ее внутренним узлам. В свою очередь скорость передачи информации и, как следствие, скорость сходимости напрямую зависят от степени неявности алгоритма метода [1–5]. По этой причине поточечные методы, такие как, например, метод последовательной верхней релаксации, хотя и позволяют достаточно надежно строить приближенное решение СЛАУ, но делают это медленно, поскольку за одну итерацию информация передается всего на один узел сеточного пространства.

В полинейных методах скорость сходимости удается повысить путем комбинации неявных прогонок вдоль сеточных линий выбранного направления и поточечного — по

остальным направлениям [3]. Однако из-за того, что лишь по одной пространственной координате информация передается за одну итерацию, а по остальным поперечным направлениям действует все тот же механизм поточечной передачи, скорость сходимости этих методов продолжает оставаться относительно невысокой.

Другой подход, также позволяющий заметно повысить эффективность итерационных методов, представляет собой так называемый принцип компенсации Булеева [2]. Согласно этому принципу компоненты вектора решения, входящие в правую часть итерационной записи уравнений и, следовательно, берущиеся с предыдущей итерации, приближенно выражаются через компоненты, определяемые на текущей итерации. Тем самым увеличивается степень неявности разрешающего оператора и соответственно повышается скорость сходимости итераций.

В работе [5] предложена модификация полинейного метода, суть которой заключается в том, что при построении решения вдоль выбранной сеточной линии с поперечного направления используются не сами значения искомой функции с предыдущей итерации, а коэффициенты двухточечной связи, характеризующие взаимоотношения значений функции. Поэтому, хотя и приближенно, информация о решении со всего поперечного направления используется при построении решения вдоль всего продольного направления в пределах одной итерации. Такой подход в сочетании с использованием принципа компенсации позволяет на порядок сократить необходимое число итераций по сравнению с классическим вариантом полинейного метода.

В работе [6] также решена проблема одномоментной передачи информации от всех границ ко всем внутренним узлам области за одну итерацию путем построения практически полностью неявного алгоритма решения СЛАУ. Несмотря на свою относительную сложность, при решении тестовой задачи предлагаемый метод позволяет на различных сетках понижать начальную норму невязки на три-четыре порядка за одну-две итерации.

В работе [7] в целях повышения эффективности модифицированного полинейного метода [5] используется технология подбора компенсационного значения итерационного параметра индивидуально для каждого разностного уравнения на каждом итерационном шаге. Суть идеи заключается в минимизации влияния очередного приближения решения с предыдущего итерационного шага на построение следующего приближения решения. Для этого в правой части каждого разностного уравнения выделяется сумма слагаемых, которая возникает вследствие итерационного характера записи уравнения и соответственно связана с приближением решения на предыдущей итерации. Эта сумма приравнивается к нулю, а из полученного уравнения вычисляется значение параметра компенсации. Результаты расчетов тестовой задачи показывают, что в зависимости от сеточного разбиения число итераций, необходимых для уменьшения начальной нормы невязки на четыре порядка, снижается в 3–10 раз по отношению к соответствующим расчетам с постоянным параметром.

В настоящей работе по аналогии с [7] рассматривается технология использования переменного параметра компенсации применительно к методу [6] и анализируются результаты ее применения.

1. Постановка задачи

Рассматривается единичная область $\Omega = \{(x, y) : 0 \leq x \leq 1, 0 \leq y \leq 1\}$, внутри которой стационарное поведение искомой функции Φ описывается обобщенным диф-

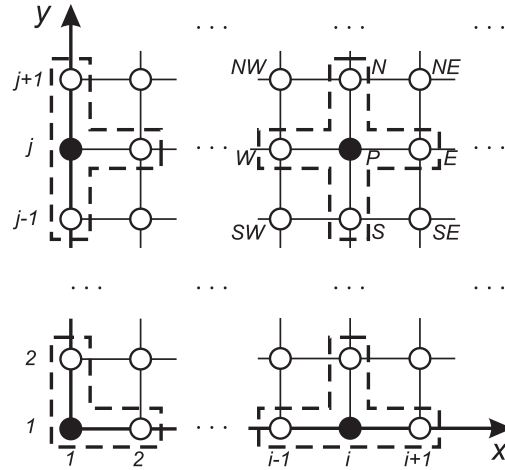


Рис. 1. Три типа шаблонов, возникающих при разностной аппроксимации уравнения и граничных условий

дифференциальным уравнением:

$$\frac{\partial}{\partial x} \left(\nu_x \frac{\partial \Phi}{\partial x} \right) + \frac{\partial}{\partial y} \left(\nu_y \frac{\partial \Phi}{\partial y} \right) = -S, \quad (1)$$

а на границе области Γ соответствует граничным условиям третьего рода:

$$a_\Gamma \frac{\partial \Phi}{\partial n} + b_\Gamma \Phi = c_\Gamma. \quad (2)$$

Здесь a_Γ , b_Γ , c_Γ — известные величины, а n — нормаль к границе.

Разностная аппроксимация (1) производится, в общем случае, по пятиточечному шаблону (рис. 1) методом контрольного объема [3] и имеет следующий вид:

$$a_{P_{ij}} \Phi_{ij} = a_{E_{ij}} \Phi_{i+1j} + a_{W_{ij}} \Phi_{i-1j} + a_{N_{ij}} \Phi_{ij+1} + a_{S_{ij}} \Phi_{ij-1} + b_{ij}. \quad (3)$$

Здесь $1 < i < n$, $1 < j < m$, где n, m — количество узлов сеточного разбиения Ω по координатам x, y соответственно. Разностное уравнение на границе области получается путем аппроксимации уравнения (1) с учетом того, что поток Φ по нормали к границе выражается через (2).

В общем случае это будет уравнение, расписанное по четырехточечному шаблону (рис. 1). Например, на граничной линии $x = 0$ или ($i = 1$) оно имеет вид

$$a_{P_{1j}} \Phi_{1j} = a_{E_{1j}} \Phi_{2j} + a_{N_{1j}} \Phi_{1j+1} + a_{S_{1j}} \Phi_{1j-1} + b_{1j}. \quad (4)$$

Аналогичным образом разностное уравнение в угловых узлах области Ω расписывается, в общем случае, по трехточечному шаблону (рис. 1). Например, в углу $(0,0)$ или ($i = 1, j = 1$) разностное уравнение имеет вид

$$a_{P_{11}} \Phi_{11} = a_{E_{11}} \Phi_{21} + a_{N_{11}} \Phi_{12} + b_{11}.$$

В итоге получается система линейных алгебраических уравнений с разреженной пятидиагональной матрицей.

2. Общая идея метода решения

Рассматриваемый метод решения СЛАУ сводится к реализации следующих пяти шагов.

1. Для простоты изложения изначально предполагается, что существует процедура устойчивых преобразований уравнений системы, расписанных по шаблонам с центральными узлами на линиях $i = 1$ и $i = 2$ (при этом $1 \leq j \leq m$), такая, что позволяет получить уравнения, соответствующие шаблонам с центральными узлами на линии $i = 2$. Причем структура этих новых шаблонов идентична структуре граничных шаблонов исходных уравнений на линии $i = 1$ (рис. 2). В этом случае преобразованная система может быть разделена на две подсистемы: первая подсистема — это уравнения на линии $i = 1$, $1 \leq j \leq m$, а вторая подсистема состоит из уравнений, соответствующих шаблонам с центральными узлами (i, j) : $2 \leq i \leq n$, $1 \leq j \leq m$, — и эта вторая подсистема будет замкнутой, поскольку в ней присутствуют вновь выведенные “граничные” трех- и четырехточечные уравнения на линии $i = 2$. Следовательно, она может быть решена независимо от уравнений на граничной линии $i = 1$.

2. Последовательное применение п.1 алгоритма вдоль оси x (увеличение индекса i , прямой ход) путем комбинации “граничных” уравнений на линии i с “приграничными” на линии $i+1$ позволяет постепенно уменьшать число уравнений в последующих замкнутых подсистемах, состоящих из уравнений, соответствующих линиям $i \dots n$.

3. В итоге при $i = n - 1$ в подсистеме останется $2m$ уравнений, соответствующих шаблонам, представленным на рис. 3. Последнее применение предполагаемой процедуры преобразований к уравнениям с шаблонами на линиях $n - 1$ и n позволяет получить замкнутую систему трехточечных по j уравнений вдоль линии $i = n$. Трехточечная структура этой системы обусловлена естественным усечением исходных разностных уравнений на границе области $x = 1$ ($i = n$). Такая подсистема может быть решена трехточечной скалярной прогонкой независимо от остальных уравнений преобразованной СЛАУ.

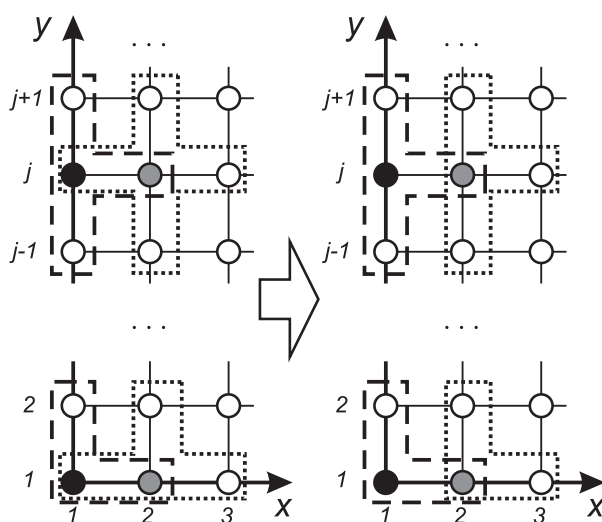


Рис. 2. Схема полинейного рекуррентного преобразования на разностные шаблоны, подобные граничным

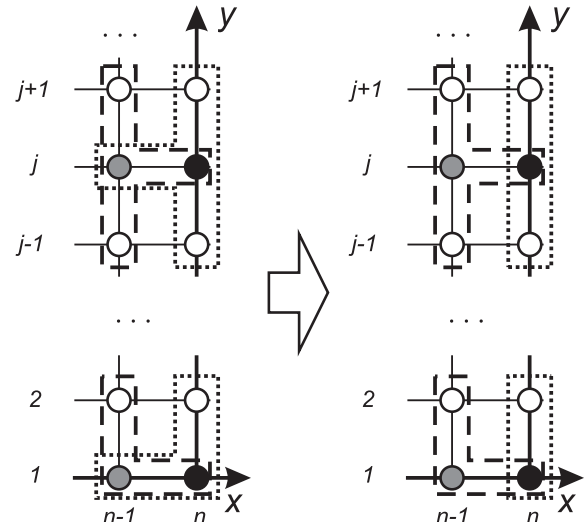


Рис. 3. Схема замыкания полинейного рекуррентного преобразования на правой границе расчетной области

4. Найденные компоненты подвектора решения $\{\Phi_{nj}\}$, $j = \overline{1, m}$ подставляются в уравнения, соответствующие, в общем случае, четырехточечным шаблонам на линии $i = n - 1$ преобразованной СЛАУ, что заново приводит к замкнутой системе трехточечных по j уравнений, которая снова решается прогонкой.

5. Последовательное применение п. 4 алгоритма вдоль оси x (уменьшение индекса i , обратный ход) позволяет, в итоге, определить все компоненты вектора решения $\{\Phi_{ij}\}$.

Нетрудно видеть, что изложенная пятишаговая идея метода кратко может быть сформулирована следующим образом: предполагается существование устойчивого преобразования исходной СЛАУ, в результате воздействия которого структура матрицы системы примет верхний блочнотреугольный вид. Размерность блоков в данном случае будет равна m . При этом диагональные блоки представляют собой подматрицы с ненулевыми элементами на главной и прилегающих к ней диагоналях. Блоки, прилегающие сверху к диагональным блокам, — это диагональные подматрицы с ненулевыми элементами на главных диагоналях. Все остальные блоки матрицы преобразованной СЛАУ — нулевые.

3. Процедура преобразования

Для вывода предполагаемого в п. 1 разд. 2 преобразования сначала рассматриваются первые три уравнения исходной СЛАУ:

$$a_{P_{11}} \Phi_{11} = a_{E_{11}} \Phi_{21} + a_{N_{11}} \Phi_{12} + b_{11}; \quad (5)$$

$$a_{P_{12}} \Phi_{12} = a_{E_{12}} \Phi_{22} + a_{N_{12}} \Phi_{13} + a_{S_{12}} \Phi_{11} + b_{12}; \quad (6)$$

$$a_{P_{13}} \Phi_{13} = a_{E_{13}} \Phi_{23} + a_{N_{13}} \Phi_{14} + a_{S_{13}} \Phi_{12} + b_{13}. \quad (7)$$

Комбинация выражений (5) и (6) с целью исключения Φ_{11} позволяет записать уравнение

$$\alpha_{P_{12}} \Phi_{12} = a_{N_{12}} \Phi_{13} + a_{E_{12}} \Phi_{22} + \alpha_{SE_{12}} \Phi_{21} + \beta_{12}, \quad (8)$$

где $\alpha_{P_{12}} = a_{P_{12}} - a_{N_{11}} a_{S_{12}} / a_{P_{11}}$, $\alpha_{SE_{12}} = a_{E_{11}} a_{S_{12}} / a_{P_{11}}$, $\beta_{12} = (a_{S_{12}} / a_{P_{11}}) b_{11} + b_{12}$.

Такое преобразование устойчиво, поскольку деление производится на элемент главной диагонали матрицы СЛАУ, который всегда: 1) отличен от нуля; 2) по модулю больше суммы модулей остальных элементов строки матрицы или равен этой сумме [1–3]. Комбинация выражений (7) и (8) с целью исключения Φ_{12} позволяет записать уравнение

$$\alpha_{P_{13}} \Phi_{13} = a_{N_{13}} \Phi_{14} + a_{E_{13}} \Phi_{23} + \tilde{\alpha}_{SE_{13}} \Phi_{22} + \alpha_{SSE_{13}} \Phi_{21} + \tilde{\beta}_{13}, \quad (9)$$

где $\alpha_{P_{13}} = a_{P_{13}} - a_{N_{12}} a_{S_{13}} / \alpha_{P_{12}}$, $\tilde{\alpha}_{SE_{13}} = a_{E_{12}} a_{S_{13}} / \alpha_{P_{12}}$, $\alpha_{SSE_{13}} = \alpha_{SE_{12}} a_{S_{13}} / \alpha_{P_{12}}$,

$$\tilde{\beta}_{13} = (a_{S_{13}} / \alpha_{P_{12}}) \beta_{12} + b_{13}.$$

Поскольку затруднительно предложить прямое устойчивое преобразование, позволяющее исключить из (9) слагаемое, содержащее Φ_{21} , не привлекая новых неизвестных, следует применить компенсационный подход. Пусть $\Delta \Phi_{ij}^{k+1} = \Phi_{ij}^{k+1} - \Phi_{ij}^k$ — поправка компоненты вектора решения в узле (i, j) между k и $k+1$ итерациями. Линейная экстраполяция $\Delta \Phi_{21}^{k+1}$ через $\Delta \Phi_{22}^{k+1}$ и $\Delta \Phi_{23}^{k+1}$ для равномерной сетки имеет вид $\Delta \Phi_{21}^{k+1} = \theta \left(2\Delta \Phi_{22}^{k+1} - \Delta \Phi_{23}^{k+1} \right)$ или

$$\Phi_{21}^{k+1} = \Phi_{21}^k + \theta \left[2 \Phi_{22}^{k+1} + \Phi_{23}^{k+1} - 2 \Phi_{22}^k - \Phi_{23}^k \right]. \quad (10)$$

Здесь θ — параметр компенсации. В силу того, что компенсация не может быть более чем полной [2], параметр θ имеет естественное ограничение: $0 \leq \theta \leq 1$.

Подстановка (10) в (9) позволяет записать уравнение

$$\alpha_{P_{13}} \Phi_{13}^{k+1} = a_{N_{13}} \Phi_{14}^{k+1} + \alpha_{E_{13}} \Phi_{23}^{k+1} + \alpha_{SE_{13}} \Phi_{22}^{k+1} + \beta_{13}, \quad (11)$$

где

$$\begin{aligned} \alpha_{P_{13}} &= a_{P_{13}} - \alpha_{N_{12}} a_{S_{13}} / \alpha_{P_{12}}, & \alpha_{E_{13}} &= a_{E_{13}} - \theta \alpha_{SE_{12}} a_{S_{13}} / \alpha_{P_{12}}, \\ \alpha_{SE_{13}} &= (a_{E_{12}} + 2\theta \alpha_{SE_{12}}) (a_{S_{13}} / \alpha_{P_{12}}), \\ \beta_{13} &= b_{13} + \left[\beta_{12} + \alpha_{SE_{12}} (\Phi_{21}^k - 2\theta \Phi_{22}^k + \theta \Phi_{23}^k) \right] (a_{S_{13}} / \alpha_{P_{12}}). \end{aligned}$$

Как и при выводе (8), данное преобразование также является устойчивым. Вместо линейной формулы для поправки решения можно воспользоваться квадратичной, которая, в общем случае, будет точнее и соответственно может повысить скорость сходимости:

$$\Delta \Phi_{21}^{k+1} = \theta \left[3 \left(\Delta \Phi_{22}^{k+1} - \Delta \Phi_{23}^{k+1} \right) + \Delta \Phi_{24}^{k+1} \right]$$

или

$$\Phi_{21}^{k+1} = \Phi_{21}^k + \theta \left[3 \left(\Phi_{22}^{k+1} - \Phi_{23}^{k+1} \right) + \Phi_{24}^{k+1} - 3 \left(\Phi_{22}^k - \Phi_{23}^k \right) - \Phi_{24}^k \right]. \quad (12)$$

Подстановка (12) в (9) позволяет вместо (11) записать уравнение

$$\alpha_{P_{13}} \Phi_{13}^{k+1} = a_{N_{13}} \Phi_{14}^{k+1} + \alpha_{NE_{13}} \Phi_{24}^{k+1} + \alpha_{E_{13}} \Phi_{23}^{k+1} + \alpha_{SE_{13}} \Phi_{22}^{k+1} + \beta_{13}, \quad (13)$$

где

$$\begin{aligned} \alpha_{P_{13}} &= a_{P_{13}} - a_{N_{12}} a_{S_{13}} / \alpha_{P_{12}}, & \alpha_{NE_{13}} &= \theta \alpha_{SE_{12}} a_{S_{13}} / \alpha_{P_{12}}, \\ \alpha_{E_{13}} &= a_{E_{13}} - 3\theta \alpha_{SE_{12}} a_{S_{13}} / \alpha_{P_{12}}, & \alpha_{SE_{13}} &= (a_{E_{12}} + 3\theta \alpha_{SE_{12}}) (a_{S_{13}} / \alpha_{P_{12}}), \\ \beta_{13} &= b_{13} + \left[\beta_{12} + \alpha_{SE_{12}} \left(\Phi_{21}^k + 3\theta \Phi_{23}^k - 3\theta \Phi_{22}^k - \theta \Phi_{24}^k \right) \right] (a_{S_{13}} / \alpha_{P_{12}}). \end{aligned}$$

Для определенности дальнейшие преобразования проводятся для варианта метода с линейной экстраполяцией поправки решения. Преобразования варианта с квадратичной экстраполяцией хотя и несколько более громоздкие, но принципиального отличия от линейного случая не имеют.

Комбинация исходного уравнения с центральной точкой в узле (1, 4)

$$a_{P_{14}} \Phi_{14}^{k+1} = a_{E_{14}} \Phi_{24}^{k+1} + a_{N_{14}} \Phi_{15}^{k+1} + a_{S_{14}} \Phi_{13}^{k+1} + b_{14}$$

с уравнением (11) с целью исключения Φ_{14}^{k+1} и последующим применением формулы линейной экстраполяции (10) вновь приводит к уравнению типа (11). В общем случае для произвольного j будет иметь место уравнение

$$\alpha_{P_{1j}} \Phi_{1j}^{k+1} = a_{N_{1j}} \Phi_{1j+1}^{k+1} + \alpha_{E_{1j}} \Phi_{2j}^{k+1} + \alpha_{SE_{1j}} \Phi_{2j-1}^{k+1} + \beta_{1j}, \quad (14)$$

где коэффициенты вычисляются аналогично (11).

Повторение подобной цепочки преобразований, но уже не по возрастанию индекса j от нижней границы $j = 1$, а по его убыванию от верхней границы $j = m$ позволяет выписать подобное (14) уравнение вида

$$\gamma_{P_{1j}} \Phi_{1j}^{k+1} = a_{S_{1j}} \Phi_{1j-1}^{k+1} + \gamma_{E_{1j}} \Phi_{2j}^{k+1} + \gamma_{NE_{1j}} \Phi_{1j+1}^{k+1} + \delta_{1j}. \quad (15)$$

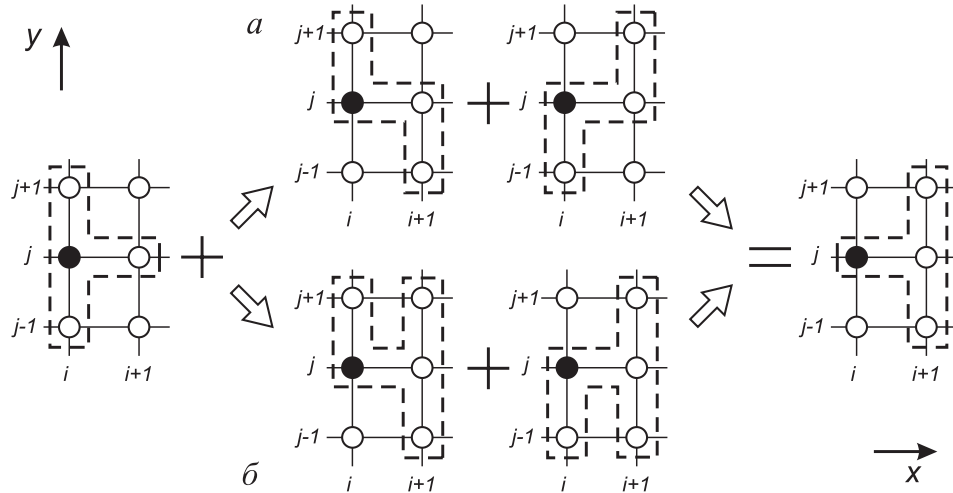


Рис. 4. Схемы комбинаций исходного уравнения с преобразованиями в целях получения уравнения, соответствующего “зеркальному” шаблону: *a* — линейная экстраполяция, *б* — квадратичная экстраполяция

В итоге, для одного и того же центрального узла $(1, j)$ имеют место три уравнения: (4), (14), (15). Причем в этих уравнениях заложена информация, которая, во-первых, содержится в уравнениях с центральными узлами только на линии $i = 1$, во-вторых, передана с границ $y = 0$ (уравнение (14)) и $y = 1$ (уравнение (15)).

Вычитание из (4) уравнений (14) и (15) приводит к уравнению вида

$$p_{P_{1j}} \Phi_{1j}^{k+1} = p_{E_{1j}} \Phi_{2j}^{k+1} + p_{SE_{1j}} \Phi_{1j-1}^{k+1} + p_{NE_{1j}} \Phi_{1j+1}^{k+1} + q_{1j}, \quad (16)$$

разностный шаблон которого “зеркален” шаблону (4) за исключением того, что его центральный узел $(1, j)$, в отличие от (4), расположен “на краю” шаблона. На рис. 4 представлены шаблоны соответствующих преобразований для вариантов как с линейной экстраполяцией поправки решения (рис. 4, *a*), так и с квадратичной (рис. 4, *б*).

Комбинация (16) с уравнением (3), записанным для $i = 2$, с целью исключения Φ_{1j}^{k+1} приводит к уравнению вида

$$\bar{a}_{P_{2j}} \Phi_{2j}^{k+1} = \bar{a}_{N_{2j}} \Phi_{2j+1}^{k+1} + \bar{a}_{E_{2j}} \Phi_{3j}^{k+1} + \bar{a}_{S_{2j}} \Phi_{2j-1}^{k+1} + \bar{b}_{2j}, \quad (17)$$

структура которого идентична структуре уравнения (4), что и требовалось выполнить согласно предположению п. 1 разд. 2 алгоритма преобразований.

Необходимо заметить, что общая идея метода решения изложена для случая, когда глобальное уменьшение числа уравнений подсистемы (прямой ход) и вычисления подвекторов решения (обратный ход) производятся вдоль оси x , а локальные преобразования уравнений — вдоль оси y . Поскольку алгоритм можно симметрично развернуть, сделав координату y глобальной, а x — локальной, под одной итерацией в дальнейшем будет пониматься последовательность двух глобальных проходов по обеим координатам. Причем приближение решения, полученное при прохождении вдоль первой координаты, используется в качестве известного для определения поправки решения при прохождении вдоль второй координаты.

Подробные расчетные формулы вычислительного алгоритма и обоснование корректности метода изложены в [6].

4. Оценка эффективности метода

Эффективность предлагаемого метода оценивается путем сравнения результатов решения задачи Дирихле разными численными методами. Задача формулируется в единичной области Ω уравнением (1) при условии

$$\nu_x = 1 + 2 \left[(x - 0.5)^2 + (y - 0.5)^2 \right], \quad \nu_y = 1 + 2 \left[0.5 - (x - 0.5)^2 - (y - 0.5)^2 \right].$$

В качестве решения выбирается пробная функция

$$u(x, y) = 256 [xy(1-x)(1-y)]^2, \quad (18)$$

которая отличается от пробной функции из работ [1, 5] $w(x, y) = \text{const } xy(1-x)(1-y)$ тем, что имеет четвертую степень зависимости от координат, а не вторую. Дело в том, что для случая квадратичной зависимости решения от координат экстраполяционная формула (12) является точным выражением. В этом случае предлагаемый метод в варианте квадратичной экстраполяции поправки решения будет прямым, что и показали проверочные расчеты, когда независимо от начального приближения, сетки и требуемой точности решение “сходилось” за одну итерацию. По той же причине вариант метода с линейной экстраполяцией поправки решения (формула (10)), как и модифицированный полинейный метод [5], является прямым для случая линейного решения.

Как следствие из (18) краевое условие первого рода задачи имеет вид $\Phi_\Gamma = 0$.

Система линейных уравнений получается путем аппроксимации исходной задачи схемой вида (3) на равномерной сетке с одинаковым числом узлов по каждой координате. При этом вектор правой части системы рассчитывается как произведение матрицы коэффициентов системы на вектор точного решения $\{u_{ij}\}$, компоненты которого вычислены в узлах сеточного разбиения по формуле (18). Динамика сходимости решения отслеживается по значению отношения норм векторов невязок $\|R^k\| / \|R^0\|$, где

$$\|R^k\| = \left\{ \sum_{ij} \left[a_{P_{ij}} \Phi_{ij}^k - \left(a_{E_{ij}} \Phi_{i+1j}^k + a_{W_{ij}} \Phi_{i-1j}^k + a_{N_{ij}} \Phi_{ij+1}^k + a_{S_{ij}} \Phi_{ij-1}^k + b_{ij} \right) \right]^2 \right\}^{1/2}. \quad (19)$$

Суммирование в (19) производится по всем узлам сеточного разбиения расчетной области. В качестве начального приближения берется единичный вектор $\Phi_{ij}^0 = 1$. Решение считается найденным, если выполняется условие $\|R^k\| / \|R^0\| < \varepsilon$, где ε — заданная точность.

Для сравнения были выбраны следующие методы: последовательной верхней релаксации (*SOR*) [1], полинейный (*LL*) [3], модифицированный полинейный метод (*mLL*) [5]. Поскольку эффективность используемых методов, включая предлагаемый, существенно зависит от величины параметра релаксации θ , каждый расчет каждым методом проводился при оптимальном для данного метода значении θ_{opt} . Делалось это для того, чтобы сравнительные характеристики сходимости были получены при наилучших для каждого метода условиях (т. е. проводилась своеобразная “оценка сверху” возможностей каждого метода). Подбор θ_{opt} производился серией пробных расчетов с целью минимизации числа итераций для достижения заданной точности.

По аналогии с аббревиатурой тестовых методов для предлагаемого метода введены сокращенные наименования: полинейный рекуррентный метод с линейной экстраполяцией — *LR1*, с квадратичной экстраполяцией — *LR2*.

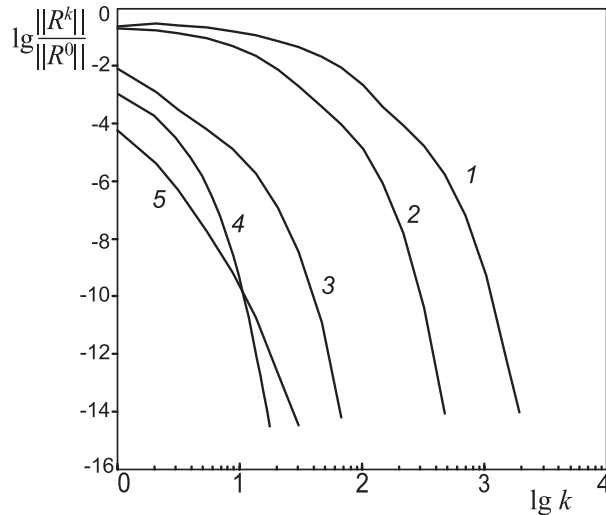


Рис. 5. Поведение отношений норм невязок в зависимости от номера итерации: 1 — *SOR*, 2 — *LL*, 3 — *mLL*, 4 — *LR1*, 5 — *LR2* lg

Расчеты проводились на сетке 101×101 при $\varepsilon = 5 \cdot 10^{-14}$. Поведение отношений норм невязок в зависимости от номера итерации k представлено на рис. 5. Видно, что за первый итерационный шаг метод *LR2* понижает начальную норму невязки более чем на четыре порядка, *LR1* — примерно на 3 порядка, *mLL* — примерно на два порядка. Классические методы *SOR* и *LL* дают за первую итерацию понижение отношения нормы невязки около одного порядка. Общее минимальное число итераций, необходимых для достижения заданной точности, также различается на порядки: для *SOR* и *LL* точность тяготеет к величине 10^3 , для *mLL* — к 10^2 , а для *LR1* и *LR2* — к первому порядку. Такой результат однозначно демонстрирует преимущество методов, обеспечивающих передачу информации от одного расчетного узла к другому практически за одну итерацию, над методами с механизмом поточно-полинейной передачи.

5. Анализ применения переменного итерационного параметра

В работе [7] формула вычисления переменного значения параметра компенсации при проходе вдоль x -координаты с учетом требования $0 \leq \theta_{ij} \leq 1$ имеет вид

$$\theta_{ij} = \max \left\{ 0, \min \left\{ 1, \left[a_{E_{ij}} \Phi_{i+1j}^k + a_{W_{ij}} \Phi_{i-1j}^k \right] / \left[(a_{E_{ij}} + a_{W_{ij}}) \Phi_{ij}^k \right] \right\} \right\}.$$

Несколько расширяя данную технологию вычисления итерационного параметра, можно принять $\theta_{ij} = \theta_0 \theta_{ij}^v$, где θ_0 — const, параметр задачи, а θ_{ij}^v рассчитывается по технологии [7]. При этом здесь и далее $\theta_0 \in [0, 1]$.

Аналогичным образом для полинейного рекуррентного метода можно построить алгоритм вычисления значения переменного параметра для каждого итерационного уравнения СЛАУ. Для этого необходимо в уравнениях типа (11) или (13), но выписанных в общем случае для узла (i, j) , приравнять нулю многочлены в круглых скобках, используемых в выражениях для расчета величин β_{ij} и δ_{ij} . Например, для уравнения

$$\alpha_{P_{ij}} \Phi_{ij}^{k+1} = a_{N_{ij}} \Phi_{ij+1}^{k+1} + \alpha_{E_{ij}} \Phi_{i+1j}^{k+1} + \alpha_{SE_{ij}} \Phi_{i+1j-1}^{k+1} + \beta_{ij},$$

в котором $\beta_{ij} = b_{ij} + \left[\beta_{ij-1} + \alpha_{SE_{ij-1}} \left(\Phi_{i+1j-2}^k - 2\theta \Phi_{i+1j-1}^k + \theta \Phi_{i+1j}^k \right) \right] (a_{S_{ij}} / \alpha_{P_{ij-1}})$, необходимо положить $\Phi_{i+1j-2}^k - 2\theta \Phi_{i+1j-1}^k + \theta \Phi_{i+1j}^k = 0$, откуда следует выражение для вычисления θ_{ij}^v .

Тогда, с учетом ограничения $0 \leq \theta_{ij} \leq 1$, при использовании метода с линейной экстраполяцией поправки решения (*LR1*) в случае глобального направления вдоль оси x (индекс i) нетрудно получить выражение для вычисления переменного параметра компенсации:

— при возрастании j

$$\theta_{ij} = \theta_0 \cdot \min \left\{ 1, \max \left\{ 0, \Phi_{i+1j-2}^k / \left(2\Phi_{i+1j-1}^k - \Phi_{i+1j}^k \right) \right\} \right\}, \quad (20)$$

— при убывании j

$$\theta_{ij} = \theta_0 \cdot \min \left\{ 1, \max \left\{ 0, \Phi_{i+1j+2}^k / \left(2\Phi_{i+1j+1}^k - \Phi_{i+1j}^k \right) \right\} \right\}. \quad (21)$$

Подобным же образом для варианта метода с использованием квадратичной экстраполяции поправки решения (*LR2*) в общем случае получаются следующие расчетные выражения:

— при возрастании j

$$\theta_{ij} = \theta_0 \cdot \min \left\{ 1, \max \left\{ 0, \Phi_{i+1j-2}^k / \left(3 \left(\Phi_{i+1j-1}^k - \Phi_{i+1j}^k \right) + \Phi_{i+1j+1}^k \right) \right\} \right\}, \quad (22)$$

— при убывании j

$$\theta_{ij} = \theta_0 \cdot \min \left\{ 1, \max \left\{ 0, \Phi_{i+1j+2}^k / \left(3 \left(\Phi_{i+1j+1}^k - \Phi_{i+1j}^k \right) + \Phi_{i+1j-1}^k \right) \right\} \right\}. \quad (23)$$

Поскольку в приграничных узлах в методе *LR2* применяется линейная экстраполяция поправки решения, для определения параметра компенсации в этом случае необходимо использовать формулы типа (20) и (21).

Аналогичные формулы выписываются и для глобального направления вдоль оси y (индекс j) с естественным учетом того, что в качестве известных значений приближения решения берутся величины, полученные при прохождении вдоль глобального направления оси x .

Для выяснения эффективности применения переменного параметра компенсации были проведены сравнительные расчеты задачи разд. 4 с точностью $\varepsilon = 5 \cdot 10^{-14}$ на сетках 101×101 , 201×201 и 401×401 методами *mLL*, *LR1* и *LR2*. При этом в половине расчетов параметр θ_0 полагался равным 1 (не оптимизировался), а в другой половине — подбирался исходя из требования минимизации числа итераций, необходимых для достижения требуемой точности.

В таблице представлены результаты расчетов по числу затраченных итераций. Для метода *LR2* в силу того, что в некоторых случаях оптимальные результаты достигались при $\theta_0 = 1.0$, эти результаты отнесены к колонкам с оптимизированным θ_0 , а в соответствующих колонках с не оптимизированным θ_0 проставлены прочерки.

По результатам видно, что применение переменного параметра хорошо проявляет себя для методов *mLL* и *LR1* при отсутствии оптимизации, когда θ либо θ_0 принимался равным 1 на всех расчетных сетках. Исключение составил расчет методом *mLL* с переменным параметром на сетке 401×401 , в котором отношение $\|R^k\| / \|R^0\|$ в лучшем случае достигало примерно 10^{-3} . В то же время для метода *LR2* применение переменного параметра не дает сколько-нибудь заметного эффекта как для не оптимизированного θ

Результаты расчетов

Метод	Сетка	Итерационный параметр							
		не оптимизированный				оптимизированный			
		постоянный		переменный		постоянный		переменный	
		θ	Число итераций	θ_0	Число итераций	θ	Число итераций	θ_0	Число итераций
Модифицированный линейный (mLL)	101×101	1	229	1	83	0.997	36	0.9975	35
	201×201	1	449	1	188	0.9992	58	0.9992	53
	401×401	1	877	1	Нет сходимости	0.9998	104	0.9998	118
Полинейный рекуррентный с линейной экстраполяцией (LR1)	101×101	1	52	1	26	0.997	16	0.998	17
	201×201	1	109	1	63	0.9989	24	0.9991	25
	401×401	1	3363	1	387	0.9996	35	0.9996	36
Полинейный рекуррентный с квадратичной экстраполяцией (LR2)	101×101	1	21	1	18	0.99992	16	0.99999	17
	201×201	1	26	–	–	0.99999	20	1.0	23
	401×401	–	–	–	–	1.0	31	1.0	28

(либо θ_0), так и для оптимизированного. В целом для методов *mLL* и *LR1* нетрудно заметить, что для оптимизированных θ (либо θ_0) число итераций сокращается в 2–3 раза и более и практически не зависит от того, использован переменный параметр или нет. Примечательно, что в этих случаях оптимизированные значения θ и θ_0 практически совпадают.

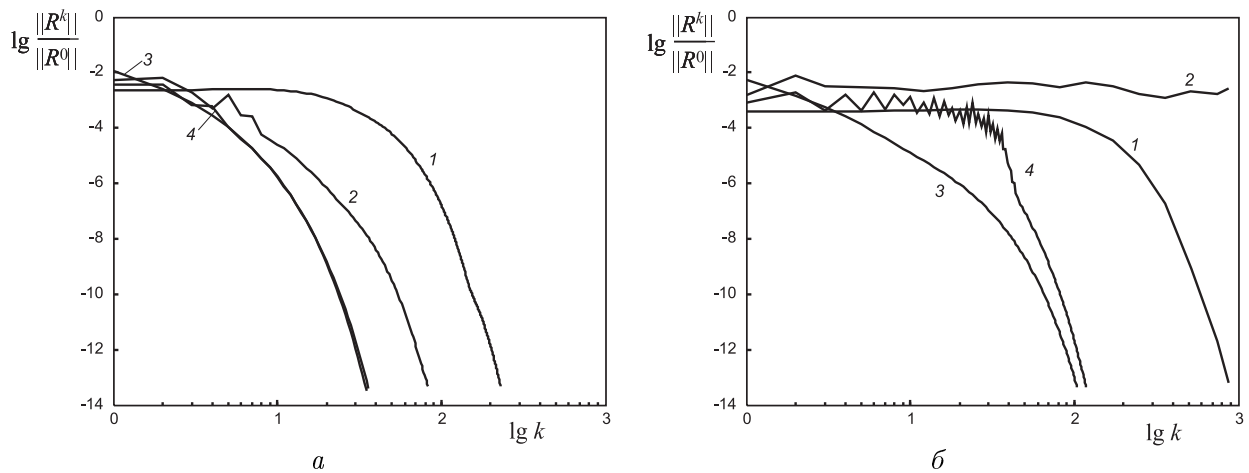


Рис. 6. Динамика отношений норм невязок при расчетах методом *mLL*: *a* — сетка 101×101 , *b* — сетка 401×401 ; параметр компенсации постоянный: 1 — $\theta = 1.0$, 3 — $\theta = \theta_{\text{opt}}$; переменный: 2 — $\theta_0 = 1.0$, 4 — $\theta_0 = \theta_{\text{opt}}$

На рис. 6 представлена динамика отношения $\|R^k\| / \|R^0\|$ в зависимости от номера итерации на различных сетках при использовании метода mLL . Из поведения графиков хорошо видно, что во всем диапазоне требования по точности применение переменного параметра компенсации не исчерпывает всего потенциала ускорения расчетов, а в случае оптимизации θ иным способом — практически не влияет на скорость сходимости. Об этом говорит заметное различие между кривыми 1 и 2, в то время как кривые 3 и 4 практически сливаются. К тому же здесь выявляется еще одна любопытная деталь применения переменного параметра: оно может приводить к понижению устойчивости вычислительного процесса на начальных итерациях либо вообще не позволяет достигнуть наперед заданной точности (рис. 6, б).

Подобные результаты на всем диапазоне точности наблюдаются и при использовании метода $LR1$ (рис. 7) за исключением того, что в данных расчетах не выявлено негативного влияния применения переменного параметра на устойчивость вычислений.

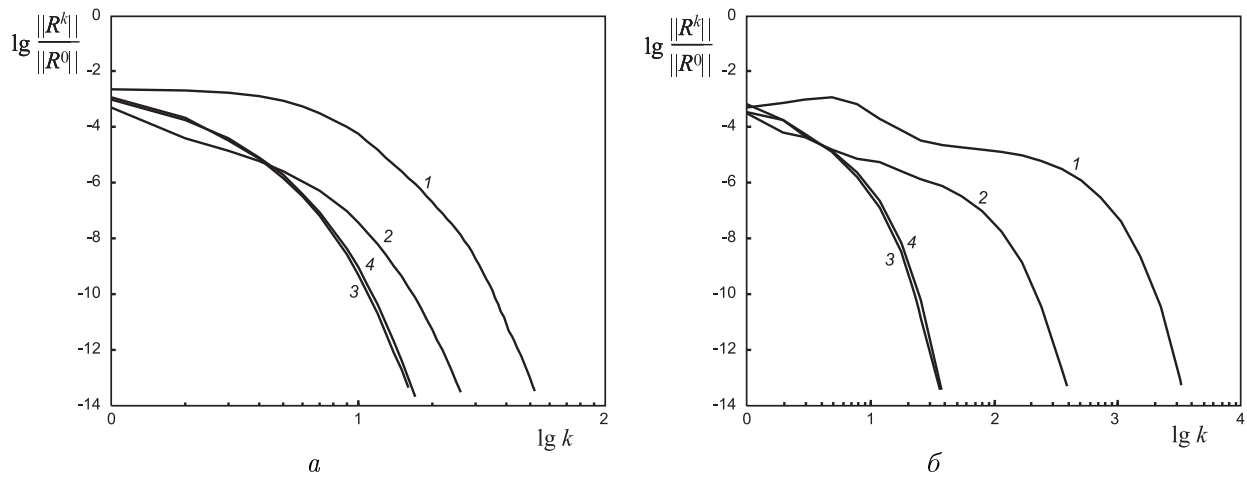


Рис. 7. Динамика отношений норм невязок при расчетах методом $LR1$: a — сетка 101×101 , b — сетка 401×401 ; параметр компенсации постоянный: 1 — $\theta = 1.0$, 3 — $\theta = \theta_{opt}$; переменный: 2 — $\theta_0 = 1.0$, 4 — $\theta_0 = \theta_{opt}$

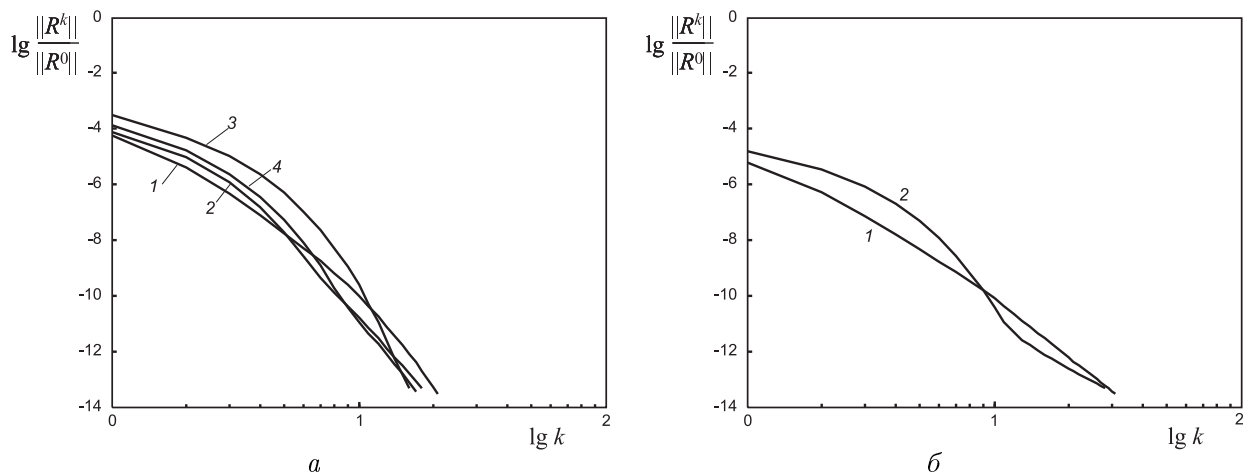


Рис. 8. Динамика отношений норм невязок при расчетах методом $LR2$: a — сетка 101×101 , b — сетка 401×401 ; параметр компенсации постоянный: 1 — $\theta = 1.0$, 3 — $\theta = \theta_{opt}$; переменный: 2 — $\theta_0 = 1.0$, 4 — $\theta_0 = \theta_{opt}$

И, наконец, на рис. 8 приведены результаты расчетов методом $LR2$. Плотнo примыкающие друг к другу кривые свидетельствуют о том, что значение параметра компенсации близко к 1, а использование переменного параметра практически не сказывается на повышении скорости сходимости.

Заключение

По результатам изложенного материала можно сделать следующие выводы.

1. Применение переменного параметра компенсации не исчерпывает всего потенциала оптимизации.
2. Не для каждого метода использование переменного параметра эффективно: метод $LR2$ практически не чувствителен к нему.
3. Оптимизация параметра компенсации, как постоянного, так и переменного, дает фактически одинаковый результат по динамике сходимости методов.
4. Применение переменного параметра может заметно понижать устойчивость итерационного алгоритма, а при определенных условиях не позволяет достигнуть сходимости решения.

Список литературы

- [1] САМАРСКИЙ А.А., НИКОЛАЕВ Е.С. Методы решения сеточных уравнений. М.: Наука, 1978.
- [2] ИЛЬИН В.П. Методы неполной факторизации для решения алгебраических систем. М.: Физматлит, 1995.
- [3] ПАТАНКАР С. Численные методы решения задач теплообмена и динамики жидкости. М.: Энергоатомиздат, 1984.
- [4] АНДЕРСОН Д., ТАННЕХИЛЛ ДЖ., ПЛЕТЧЕР Р. Вычислительная гидродинамика и теплообмен. Т. 1. М.: Мир, 1990.
- [5] ЗВЕРЕВ В.Г. Модифицированный полинейный метод решения разностных эллиптических уравнений // ЖВМ и МФ. 1998. Т. 38, № 9. С. 1553–1562.
- [6] ФОМИН А.А., ФОМИНА Л.Н. Полинейный рекуррентный метод решения разностных эллиптических уравнений. Деп. в ВИНТИ. 2007. № 385-В2007.
- [7] ZVEREV V.G. About the iteration method for solving difference equations // Lecture notes in computer science. Berlin; Heidelberg: Springer-Verlag, 2005. Vol. 3401. P. 621–628.

Поступила в редакцию 11 ноября 2008 г.