

МАТЕМАТИКА

УДК 519.612

MSC 65F10

Улучшение одной из оценок скорости сходимости метода Зейделя

А. Н. Борзых

Санкт-Петербургский государственный университет,
Российская Федерация, 199034, Санкт-Петербург, Университетская наб., 7–9

Для цитирования: Борзых А. Н. Улучшение одной из оценок скорости сходимости метода Зейделя // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. 2019. Т. 6 (64). Вып. 2. С. 185–195. <https://doi.org/10.21638/11701/spbu01.2019.201>

В статье рассматривается метод Зейделя для решения системы линейных алгебраических уравнений и оценка скорости сходимости метода Зейделя. Предлагается построение эквивалентной системы, для которой метод Зейделя сходится также, но оценка скорости сходимости лучше. Построение эквивалентной системы производится отдельным итерационным процессом, один шаг которого требует $O(n)$ операций. Доказывается сходимость этого процесса. Представляются результаты численных экспериментов, показывающие улучшение оценки скорости сходимости.

Ключевые слова: метод Зейделя, одношаговый циклический процесс, СЛАУ, система линейных алгебраических уравнений, итерационные методы решения, сходимость метода Зейделя, оценка скорости сходимости метода Зейделя.

1. Введение. Различные физические задачи часто приводят к необходимости решения систем линейных алгебраических уравнений. Если размерность задачи велика, то применение точных методов оказывается затратным, пользуются теми или иными итерационными методами, получающими приближенное решение с определяемой пользователем точностью. В основе большинства итерационных методов лежит идея метода простой итерации, заключающаяся в представлении системы уравнений в виде

$$x = Ax + f, \tag{1}$$

выборе произвольного x_0 и последующем вычислении $x_{k+1} = Ax_k + f$. Известно (см., например, [1]), что метод простой итерации сходится, если спектральный радиус матрицы A меньше единицы: $\rho(A) < 1$.

Одним из вариантов метода простой итерации является метод Зейделя (одношаговый циклический процесс). Он отличен от метода простой итерации тем, что переход от вектора x_k к вектору x_{k+1} происходит не сразу, а за n «простых» шагов. Каждый «простой» шаг заключается в вычислении i -го элемента вектора x_{k+1} , где i меняется от 1 до n . Для определения i -го элемента вектора x_{k+1} используются уже вычисленные элементы с 1 по $i - 1$ вектора x_{k+1} и элементы с i по n вектора x_k . Формула перехода такова:

$$x_{k+1}^i = \sum_{j=1}^{i-1} a_{ij} x_{k+1}^j + \sum_{j=i}^n a_{ij} x_k^j + f^i.$$

Известно (см., например, [1]), что метод Зейделя эквивалентен методу простой итерации для равносильной системы $x = Bx + g$, где $B = (E - \hat{L})^{-1}(\hat{D} + \hat{R})$, $g = (E - \hat{L})^{-1}f$, E — единичная матрица, $\hat{L}$, $\hat{D}$, $\hat{R}$ — нижнетреугольная (Left), диагональная (Diag) и верхнетреугольная (Right) матрицы, образующие матрицу A : $A = \hat{L} + \hat{D} + \hat{R}$.

Следовательно, метод Зейделя сходится, если спектральный радиус B меньше единицы: $\rho(B) < 1$. Величина $\rho(B)$, кроме того, является коэффициентом скорости сходимости.

На практике вычисление $\rho(B)$ является столь же сложной задачей, как и решение СЛАУ: требуется матричное обращение, матричное умножение и вычисление наибольшего по модулю собственного числа. В силу этого для определения факта сходимости (возможности применения метода Зейделя) и примерной оценки скорости сходимости (количества шагов для достижения заданной точности) пользуются какими-либо достаточными условиями сходимости. Для метода Зейделя удобно вычисляемым достаточным условием является условие

$$\mu < 1, \quad \text{где} \quad \mu = \max_i \mu_i, \quad \mu_i = \frac{\gamma_i}{1 - \beta_i}, \quad \beta_i = \sum_{j=1}^{i-1} |a_{ij}|, \quad \gamma_i = \sum_{j=i}^n |a_{ij}|.$$

Недостаток этой оценки — слишком «грубый» характер величины μ . Ставится вопрос о способах «улучшения» μ , которые не требовали бы существенных вычислительных затрат и давали бы меньшее значение μ , более близкое к $\rho(B)$.

Один из способов был предложен в работе [2]. Идея заключалась в домножении (1) слева на некоторую матрицу D :

$$Dx = DA(D^{-1}D)x + Df,$$

и получении эквивалентной системы $\tilde{x} = \tilde{A}\tilde{x} + \tilde{f}$, где $\tilde{x} = Dx$, $\tilde{A} = DAD^{-1}$, $\tilde{f} = Df$.

Матрица D выбиралась из класса матриц перестановок (в каждой строке и каждом столбце все нули и одна единица) и «подбиралась» таким образом, чтобы значение $\mu(\tilde{A})$ было наименьшим. Несмотря на факториальное количество различных вариантов матриц D , был предложен и доказан алгоритм, определяющий оптимальную матрицу D за время $O(n^2)$.

В настоящей работе мы продолжим исследование «оптимизации» μ , используем тот же прием, но матрицу D выбираем из класса диагональных матриц, т. е. ставим

задачу поиска диагональной матрицы D , обеспечивающей минимум величины $\mu(\tilde{A})$, где $\tilde{A} = DAD^{-1}$.

Отметим, что преобразование $\tilde{A} = DAD^{-1}$ изменяет μ , но никоим образом не влияет на $\rho(B)$, т. е. на «действительную» скорость сходимости метода Зейделя:

$$\begin{aligned}\tilde{A} &= DAD^{-1} = D(\hat{L} + \hat{D} + \hat{R})D^{-1} = D\hat{L}D^{-1} + D\hat{D}D^{-1} + D\hat{R}D^{-1}, \\ \tilde{B} &= (E - D\hat{L}D^{-1})^{-1}(D\hat{D}D^{-1} + D\hat{R}D^{-1}) = D(E - \hat{L})^{-1}(\hat{D} + \hat{R})D^{-1} = DBD^{-1}, \\ \tilde{B} &\sim B \quad \rightarrow \quad \rho(\tilde{B}) = \rho(B).\end{aligned}$$

В этом смысле целью работы является получение более качественной оценки, а не более быстрого вычислительного алгоритма.

2. Предлагаемый алгоритм. Матрицу $\tilde{A}$ строим как предел итерационного процесса:

$$A_{k+1} = D_k A_k D_k^{-1}, \quad \text{где } A_0 = A, \quad D_k(i, \alpha) = \text{diag}(1, \dots, \alpha, \dots, 1).$$

Тогда $D = \dots D_k \dots D_2 D_1 D_0$. Параметры i и α на каждом шаге выбираем следующим образом: $i = \underset{i}{\operatorname{argmin}} \mu_i(A_k)$, то есть i определяет минимальное значение μ_i матрицы A_k ; α — решение задачи на минимум $\mu(A_{k+1})$:

$$\mu(A_{k+1}) = \max_j \mu_j(A_{k+1}) \xrightarrow{\alpha} \min.$$

В этом смысле описанный алгоритм эквивалентен методу поиска экстремума

$$f(d_1, \dots, d_n) = \mu(DAD^{-1}) \xrightarrow{d_1, \dots, d_n} \min, \quad D = \text{diag}(d_1, \dots, d_n),$$

использующему покоординатный спуск.

3. Практическая реализация предлагаемого алгоритма. Здесь и далее считаем, что в начальной матрице A нет нулевых элементов. Если это не так, то построение эквивалентной СЛАУ $x = Ax + f$, не содержащей нулевых элементов в A , является отдельной задачей, которую в данной работе не рассматриваем. Отсутствие нулевых элементов в A гарантирует отсутствие нулевых элементов во всех матрицах A_k .

Утверждение 1. Если $\alpha > 1$, то $\mu_i(A_{k+1}) > \mu_i(A_k)$ и $\forall j \neq i \mu_j(A_{k+1}) < \mu_j(A_k)$, то есть осуществляемое преобразование при $\alpha > 1$ увеличивает значение μ_i (в матрице A_k оно было минимальным) и уменьшает все остальные μ_j .

ДОКАЗАТЕЛЬСТВО. Обозначим $\hat{\gamma}_i = \gamma_i - |a_{ii}| = \sum_{j=i+1}^n |a_{ij}|$. В таких обозначениях имеем $\mu_i(A_k) = \frac{|a_{ii}| + \hat{\gamma}_i}{1 - \beta_i}$. Преобразование $A_{k+1} = D_k A_k D_k^{-1}$ умножает элементы строки i на α , делит элементы столбца i на α , оставляет без изменения a_{ii} и все остальные элементы. Тогда получаем

$$\mu_i(A_{k+1}) = \frac{|a_{ii}| + \alpha \hat{\gamma}_i}{1 - \alpha \beta_i}. \quad (2)$$

Рассмотрим три случая:

- 1) если $i = 1$, то $\beta_i = 0$, $\hat{\gamma}_i > 0$, $|a_{ii}| + \alpha\hat{\gamma}_i > |a_{ii}| + \hat{\gamma}_i$, $1 - \alpha\beta_i = 1 - \beta_i$,
- 2) если $1 < i < n$, то $\beta_i > 0$, $\hat{\gamma}_i > 0$, $|a_{ii}| + \alpha\hat{\gamma}_i > |a_{ii}| + \hat{\gamma}_i$, $1 - \alpha\beta_i < 1 - \beta_i$,
- 3) если $i = n$, то $\beta_i > 0$, $\hat{\gamma}_i = 0$, $|a_{ii}| + \alpha\hat{\gamma}_i = |a_{ii}| + \hat{\gamma}_i$, $1 - \alpha\beta_i < 1 - \beta_i$.

Во всех трех случаях получаем

$$\mu_i(A_{k+1}) = \frac{|a_{ii}| + \alpha\hat{\gamma}_i}{1 - \alpha\beta_i} > \frac{|a_{ii}| + \hat{\gamma}_i}{1 - \beta_i} = \mu_i(A_k).$$

Первая часть утверждения 1 доказана.

Поскольку преобразование $A_{k+1} = D_k A_k D_k^{-1}$ изменяет не только строку, но и столбец, то меняются все остальные μ_j . Покажем, что они уменьшаются.

Возможны два случая.

1. Случай $j < i$. В строке j изменяется элемент a_{ji} , лежащий *правее* диагонали и влияющий на величину параметра $\hat{\gamma}_j$, который уменьшается:

$$\hat{\gamma}_j(A_{k+1}) = (\hat{\gamma}_j(A_k) - |a_{ji}|) + \frac{1}{\alpha}|a_{ji}| < \hat{\gamma}_j(A_k),$$

а параметр β_j остается без изменения. Числитель дроби $\frac{|a_{jj}| + \hat{\gamma}_j}{1 - \beta_j}$ уменьшается, следовательно дробь уменьшается: $\mu_j(A_{k+1}) < \mu_j(A_k)$.

2. Случай $j > i$. В строке j изменяется элемент a_{ji} , лежащий *левее* диагонали и влияющий на величину параметра β_j , который уменьшается:

$$\beta_j(A_{k+1}) = (\beta_j(A_k) - |a_{ji}|) + \frac{1}{\alpha}|a_{ji}| < \beta_j(A_k),$$

а параметр $\hat{\gamma}_j$ остается без изменения. Знаменатель дроби $\frac{|a_{jj}| + \hat{\gamma}_j}{1 - \beta_j}$ увеличивается, следовательно дробь уменьшается: $\mu_j(A_{k+1}) < \mu_j(A_k)$.

Утверждение 1 доказано. □

Замечание 1. Для обеспечения строгих неравенств $\mu_i(A_{k+1}) > \mu_i(A_k)$, $\mu_j(A_{k+1}) < \mu_j(A_k)$ было использовано оговоренное ранее отсутствие нулевых элементов в матрице A_0 и во всех последующих матрицах A_k .

Замечание 2. Доказательство утверждения можно построить иначе. Возьмем от выражения (3) производную по α :

$$\mu'_i(\alpha) = \frac{\hat{\gamma}_i(1 - \alpha\beta_i) - (|a_{ii}| + \alpha\hat{\gamma}_i)(-\beta_i)}{1 - \alpha\beta_i}^2 = \frac{\hat{\gamma}_i + \beta_i|a_{ii}|}{1 - \alpha\beta_i}^2.$$

Только одна из величин $\hat{\gamma}_i$ и β_i может быть нулем. Получаем неравенство $\mu'_i(\alpha) > 0$, и $\mu_i(\alpha)$ есть строго монотонно возрастающая функция. Тогда при $\alpha > 1$ имеем $\mu_i(A_{k+1}) = \mu_i(\alpha) > \mu_i(1) = \mu_i(A_k)$. Аналогично показывается, что $\forall j \neq i$ $\mu_j(\alpha)$ есть строго монотонно убывающие функции (требуется отдельное рассмотрение случаев $j < i$ и $j > i$). Данные вычисления также необходимы для утверждения 8 (см. ниже).

Утверждение 2. Решение задачи $\mu(A_{k+1}) \xrightarrow{\alpha} \min$ эквивалентно решению задачи $\alpha = \max_j \alpha_j$, где α_j — решения уравнений $\mu_j(A_{k+1}) = \mu_i(A_{k+1})$.

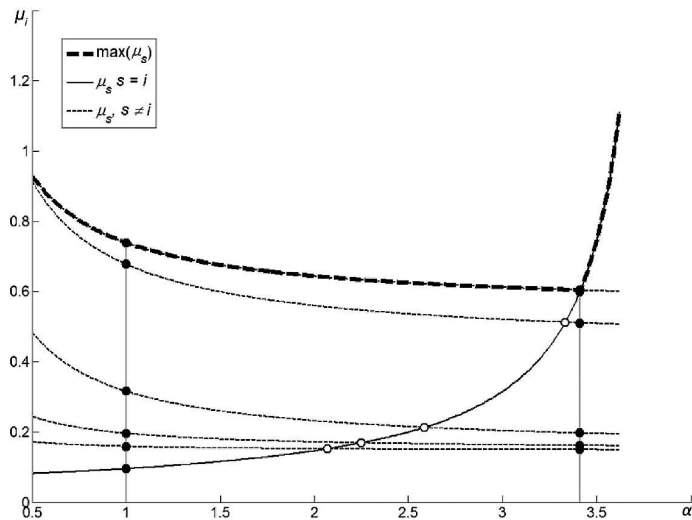


Рис. 1. Зависимости μ_i от α .

ДОКАЗАТЕЛЬСТВО. Рассмотрим μ_j как функции от α . По замечанию 2 имеем, что $\mu_i(\alpha)$ есть строго монотонно возрастающая функция, $\mu_j(\alpha)$ при $j \neq i$ — строго монотонно убывающие функции. На рис. 1 представлен пример для матрицы размерности 6.

Очевидно, что необходимое условие минимума $\mu(\alpha) = \max_j \mu_j(\alpha)$ есть условие $\mu_j(\alpha) = \mu_i(\alpha)$ для одного из j , а именно того, при котором значение $\mu_j(\alpha)$ будет наибольшим, что соответствует наибольшему α_j . Доказано. $\square$

Замечание 3. Может возникнуть иллюзия, что достаточно решить лишь одно уравнение $\mu_p(\alpha) = \mu_i(\alpha)$, где $p = \underset{j}{\operatorname{argmax}} \mu_j(A_k)$, т.е. искать пересечение самого верхнего убывающего графика $\mu_p(\alpha)$ с возрастающим графиком $\mu_i(\alpha)$. Это не так, из неравенства $\mu_p(1) > \mu_j(1)$, $j \neq i$, не следует $\mu_p(\alpha) > \mu_j(\alpha)$, т.е. графики могут «меняться местами».

Замечание 4. Тем не менее, существенную часть уравнений можно не рассматривать. Пусть найден корень α_p уравнения $\mu_p(\alpha) = \mu_i(\alpha)$, где $p = \underset{j}{\operatorname{argmax}} \mu_j(A_k)$. Тогда уравнения $\mu_j(\alpha) = \mu_i(\alpha)$, для которых $\mu_j(1) < \mu_i(\alpha_p)$, можно не рассматривать, так как их корни заведомо меньше α_p . Численный эксперимент показывает, что на матрицах размерности 100 средний выигрыш из-за устранения лишних уравнений составляет 92%, а на матрицах размерности 1000 — 98%.

4. Итоговая вычислительная схема предлагаемого алгоритма.

1. Положим $k = 0$. Для всех i вычислим параметры $\beta_i, \hat{\gamma}_i, \mu_i$ начальной матрицы A_0 . Время вычисления — $O(n^2)$.
2. Положим $i = \underset{j}{\operatorname{argmin}} \mu_j(A_k)$, $s = \underset{j}{\operatorname{argmax}} \mu_j(A_k)$.
3. Найдем корень α_s уравнения $\mu_s(A_{k+1}) = \mu_i(A_{k+1})$. Рассмотрим два случая.

А. Если $s < i$, то в строке s меняется элемент a_{si} , лежащий правее диагонали. Следовательно, меняется $\hat{\gamma}_s : \hat{\gamma}_s(A_{k+1}) = \hat{\gamma}_s(A_k) - |a_{si}| + \frac{1}{\alpha}|a_{si}|$. Уравнение $\mu_s(A_{k+1}) = \mu_i(A_{k+1})$ имеет вид

$$\frac{|a_{ss}| + \hat{\gamma}_s - |a_{si}| + \frac{1}{\alpha}|a_{si}|}{1 - \beta_s} = \frac{|a_{ii}| + \alpha\hat{\gamma}_i}{1 - \alpha\beta_i},$$

что дает квадратное уравнение относительно α : $c_2\alpha^2 + c_1\alpha + c_0 = 0$, где $c_2 = \hat{\gamma}_i(1 - \beta_s) + t\beta_i$, $c_1 = |a_{ii}| - \beta_s|a_{ii}| - t + \beta_i|a_{si}|$, $c_0 = -|a_{si}|$, $t = |a_{ss}| + \hat{\gamma}_s - |a_{si}|$. Из двух вещественных корней берем наибольший.

Б. Если $s > i$, то в строке s меняется элемент a_{si} , лежащий левее диагонали. Следовательно, меняется β_s : $\beta_s(A_{k+1}) = \beta_s(A_k) - |a_{si}| + \frac{1}{\alpha}|a_{si}|$. Уравнение $\mu_s(A_{k+1}) = \mu_i(A_{k+1})$ принимает вид

$$\frac{|a_{ss}| + \hat{\gamma}_s}{1 - \beta_s + |a_{si}| - \frac{1}{\alpha}|a_{si}|} = \frac{|a_{ii}| + \alpha\hat{\gamma}_i}{1 - \alpha\beta_i}.$$

Соответствующее квадратное уравнение таково: $c_2\alpha^2 + c_1\alpha + c_0 = 0$, где $c_2 = t\hat{\gamma}_i + \beta_i(|a_{ss}| + \hat{\gamma}_s)$, $c_1 = t|a_{ii}| - \hat{\gamma}_i|a_{si}| - |a_{ss}| - \hat{\gamma}_s$, $c_0 = -|a_{si}||a_{ii}|$, $t = 1 - \beta_s + |a_{si}|$. Из двух вещественных корней также берем наибольший. Сложность данных вычислений — $O(1)$.

4. Для всех $j \neq i$, s выполняем следующее:

а) если $\mu_j < \mu_i(\alpha_s) = \frac{|a_{ii}| + \alpha_s\hat{\gamma}_i}{1 - \alpha_s\beta_i}$, полагаем $\alpha_j = 1$ и ничего не вычисляем;

б) в противном случае α_j вычисляем как корень $\mu_j(A_{k+1}) = \mu_i(A_{k+1})$. Схема вычисления описана в пункте 3. Сложность вычисления всех векторов α_j в «худшем» случае — $O(n)$.

5. Положим $\alpha = \max_j \alpha_j$, где α_j — найденные корни уравнений $\mu_j(A_{k+1}) = \mu_i(A_{k+1})$.

6. Выполняем преобразование $A_{k+1} = D_k A_k D_k^{-1}$, $D_k(i, \alpha) = \text{diag}(1, \dots, \alpha, \dots, 1)$. Преобразование изменяет одну строку и столбец, сложность вычислений — $O(n)$.

7. Произведем перерасчет $\beta_j, \hat{\gamma}_j$:

а) для $j = i$: $\beta_j(A_{k+1}) = \alpha\beta_j(A_k)$, $\hat{\gamma}_j(A_{k+1}) = \alpha\hat{\gamma}_j(A_k)$;

б) для $j < i$: $\beta_j(A_{k+1}) = \beta_j(A_k)$, $\hat{\gamma}_j(A_{k+1}) = \hat{\gamma}_j(A_k) - |a_{ji}| + \frac{1}{\alpha}|a_{ji}|$;

в) для $j > i$: $\beta_j(A_{k+1}) = \beta_j(A_k) - |a_{ji}| + \frac{1}{\alpha}|a_{ji}|$, $\hat{\gamma}_j(A_{k+1}) = \hat{\gamma}_j(A_k)$.

А после проведем перерасчет всех μ_j : $\mu_j(A_{k+1}) = \frac{|a_{ii}| + \hat{\gamma}_j(A_{k+1})}{1 - \beta_j(A_{k+1})}$.

Сложность данных вычислений — $O(n)$.

8. Обновим D и k : $D = D_k D$, $k = k + 1$.

9. Проверяем критерий остановки: количество выполненных шагов или иной. Переходим к пункту 2.

На рис. 2 показан ход работы алгоритма для матрицы размерности 10. По оси абсцисс отложен номер шага алгоритма, по оси ординат — значения μ_j на соответствующих шагах алгоритма (10 графиков). Данный пример показывает уменьшение

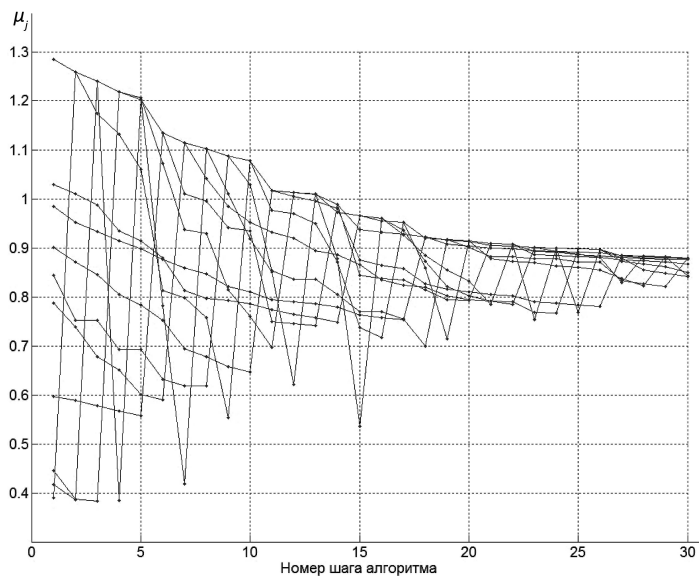


Рис. 2. Значения μ_j на различных шагах алгоритма.

$\mu(A_k)$ с величины примерно 1.3 (сходимость метода Зейделя не ясна) до величины примерно 0.9 (метод Зейделя гарантированно сходится).

5. Доказательство сходимости. Доказательство сходимости сформулируем в виде лемм 1–2 и утверждений 3–9. Подробные доказательства каждого из утверждений оставим предметом для последующей публикации.

Лемма 1 (о собственных векторах матрицы с неотрицательными элементами). Пусть A — матрица с неотрицательными элементами (используем обозначение $A \triangleright 0$) и она неразложима (для этого достаточно отсутствия нулевых элементов). Если λ — наибольшее по модулю собственное число, то оно вещественное, имеет кратность 1 и ему соответствует собственный вектор, элементы которого имеют одинаковые знаки; если λ — иное собственное значение, то ему соответствует собственный вектор, элементы которого имеют различные знаки.

Лемма 2 (об увеличении наибольшего собственного числа матрицы с неотрицательными элементами). Если $A \triangleright 0$, $B \triangleright 0$, то $\lambda_{\max}(A + B) > \lambda_{\max}(A)$.

Утверждение 3 (необходимое условие оптимальной матрицы $\tilde{A}$). Если значение $\mu(\tilde{A})$ минимально (т. е. матрица $\tilde{A}$ оптимальная), то $\forall i, j \mu_i(\tilde{A}) = \mu_j(\tilde{A})$.

Утверждение 4 (существование и единственность оптимальной матрицы $\tilde{A}$). Матрица $\tilde{A}$ существует и является единственной с точностью до знаков элементов.

Утверждение 5 (необходимое и достаточное условие оптимальной матрицы $\tilde{A}$). Матрица $\tilde{A}$ оптимальная тогда и только тогда, когда $\forall i, j \mu_i(\tilde{A}) = \mu_j(\tilde{A})$.

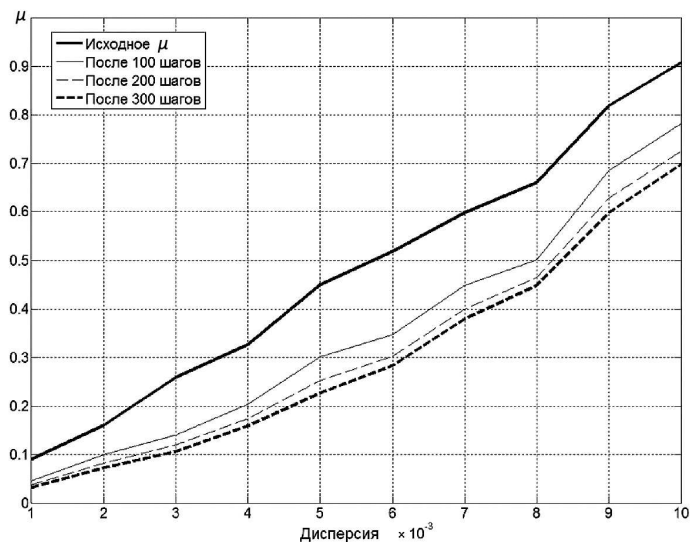


Рис. 3. Зависимость μ от дисперсии элементов матрицы.

Утверждение 6 (о двухсторонней оценке оптимального μ). Пусть $\mu(\tilde{A}) = \mu^*$, $\tilde{A}$ — оптимальная матрица. Тогда справедлива оценка: $\min_i \mu_i(A) \leq \mu^* \leq \max_i \mu_i(A)$, где A — начальная или любая другая матрица, получаемая алгоритмом.

Утверждение 7 (о двухсторонней оценке элементов матрицы A_k). Пусть A_k получена на шаге k предложенным алгоритмом: $A_k = DAD^{-1}$, $D = \text{diag}(d_1, \dots, d_n)$, A — начальная матрица, a_{ps} — ее элементы, a_{ij}^k — элементы A_k . Справедлива оценка:

$$\forall i, j \quad m^2 \leq |a_{ij}^k| \leq 1, \quad \text{где} \quad m = \min_{p,s} |a_{ps}|.$$

Утверждение 8 (о двусторонней оценке α). Рассмотрим один шаг алгоритма: $A_{k+1} = D_k A_k D_k^{-1}$, $D_k(i, \alpha) = \text{diag}(1, \dots, \alpha, \dots, 1)$, $i = \underset{s}{\text{argmin}} \mu_s(A_k)$, $\alpha = \max_j \alpha_j$, α_j — решения уравнений $\mu_j(A_{k+1}) = \mu_i(A_{k+1})$. Справедлива оценка для α :

$$1 + \frac{m^2}{3} \left(\max_s \mu_s(A_k) - \min_s \mu_s(A_k) \right) \leq \alpha \leq \frac{1}{m^2}.$$

Утверждение 9 (о линейной скорости сходимости алгоритма). $|\mu(A_{k+1}) - \mu^*| < C \cdot |\mu(A_k) - \mu^*|$, $C = 1 - \frac{m^5}{12}$, $\mu^* = \mu(\tilde{A})$, $\tilde{A}$ — оптимальная матрица.

6. Численные эксперименты. Численные эксперименты проведем на случайных матрицах размерности 100, элементами которых являются случайные величины с нормальным законом распределения с математическим ожиданием 0 и дисперсией, которую варьировем от 0.001 до 0.01. Для каждой матрицы отмечаем ее исходное

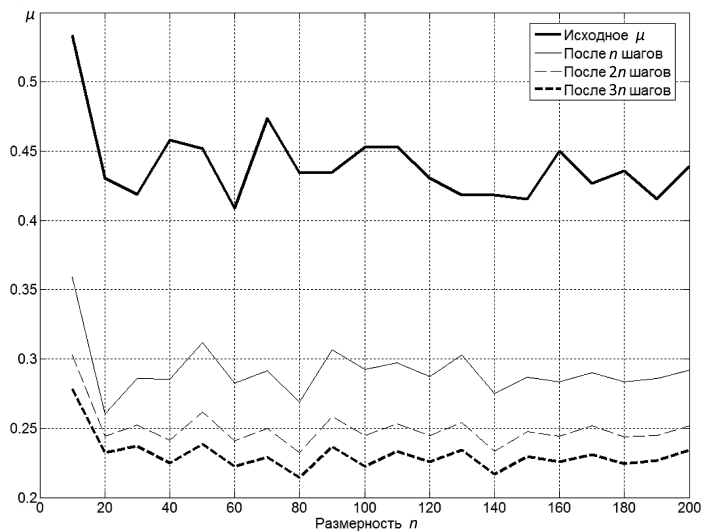


Рис. 4. Зависимость μ от размерности матрицы.

значение μ , а также значения μ , которые получены после 100, 200 и 300 шагов предложенного алгоритма. Результаты представлены на рис. 3.

Следующую серию экспериментов проведем на матрицах размерностей от 10 до 200. Элементами матриц являются случайные величины с нормальным законом распределения с математическим ожиданием 0 и дисперсией $\frac{1}{2n}$. Выполнение предложенного алгоритма осуществляем за n , $2n$ и $3n$ шагов. Результаты представлены на рис. 4.

Из проведенных экспериментов видим, что предложенный алгоритм дает уменьшение скорости сходимости μ в среднем на 40%, требуя на вычисления порядка $O(n^2)$ операций.

Последнюю серию экспериментов проведем на матрице фиксированной размерности 100 с фиксированной дисперсией элементов 0.005, демонстрируя скорость сходимости предложенного алгоритма. Результаты представлены на рис. 5.

По оси абсцисс отложен номер шага алгоритма. По оси ординат, представленной в логарифмическом масштабе, отображаются:

- 1) фактическая ошибка: $\mu(A_k) - \mu^*$, где значение μ^* получено этим же алгоритмом за 2000 шагов;
- 2) апостериорная теоретическая ошибка: $\max_i \mu_i(A_k) - \min_i \mu_i(A_k)$;
- 3) априорная теоретическая ошибка: $\left(1 - \frac{m^5}{12}\right)^k \left(\max_i \mu_i(A_0) - \min_i \mu_i(A_0)\right)$.

Практически горизонтальный вид априорной ошибки обусловлен тем, что величина m^5 весьма мала, величина $1 - \frac{m^5}{12}$ близка к единице. В этом смысле представленное доказательство линейной скорости сходимости имеет преимущественно теоретическую ценность доказательства самой сходимости, а не практический способ оценки ее скорости. Скачкообразный вид апостериорной теоретической ошибки обусловлен тем, что в ходе работы алгоритма обеспечивается уменьшение $\max_i \mu_i(A_k)$

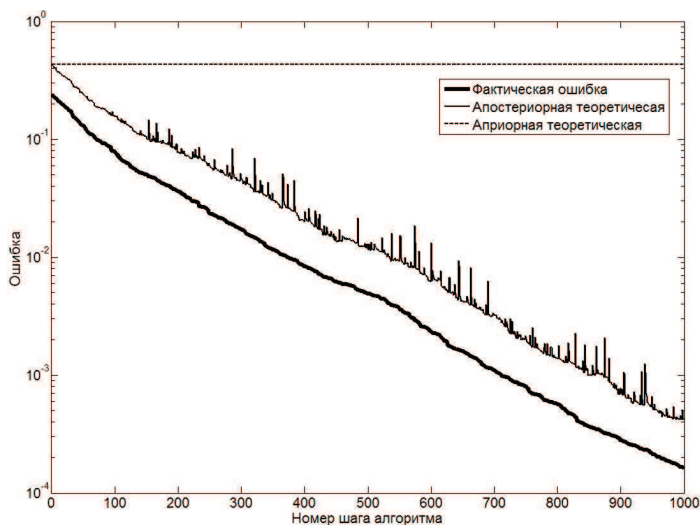


Рис. 5. Фактическая, априорная и апостериорная ошибки.

на каждом шаге алгоритма, однако увеличение $\min_i \mu_i(A_k)$ на каждом шаге не гарантировано.

7. Открытые вопросы. 1. В данной работе переход от системы $x = Ax + f$ к системе $\tilde{x} = \tilde{A}\tilde{x} + \tilde{f}$ осуществлялся посредством преобразования $\tilde{A} = DAD^{-1}$, где D — диагональная матрица. В работе [2] использовался подход, в котором матрицей D являлась матрица перестановок. Каждый из подходов, как видно из результатов численных экспериментов, дает существенное уменьшение μ . Интересен вопрос об объединении этих подходов, а также использовании каких-либо иных видов матрицы D .

2. Продemonстрированные подходы дают уменьшение μ на 20–50%. Интересен вопрос о том, насколько данное уменьшение можно считать значимым, насколько получаемое значение μ близко к истинной скорости сходимости, определяемой спектральным радиусом матрицы $B = (E - \hat{L})^{-1}(\hat{D} + \hat{R})$. Численные эксперименты показывают, что в зависимости от вида матрицы A возможны как почти точные совпадения «улучшенного» значения μ и величины $\rho(B)$, так и их существенные различия, что оставляет вопрос о достоверной оценке скорости сходимости метода Зейделя актуальным.

3. Предложенный алгоритм является итерационным, и, следовательно, интересен вопрос об оценке его вычислительной сложности, например, в сравнении со сложностью метода Зейделя, для которого он получает оценку. Очевидно, что алгоритм, получающий оценку скорости сходимости, должен иметь существенно меньшую сложность, чем алгоритм, для которого эта оценка получается (иначе в нем смысла). Для матриц размерности 100, описанных выше, эксперименты показывают, что вычислительная сложность составляет примерно 10% от сложности метода Зейделя.

8. Заключение. Предложенный алгоритм позволяет получить более хорошую оценку скорости сходимости метода Зейделя, чем оценка, описанная в работе [1].

Литература

1. Фаддеев Д. К., Фаддеева В. Н. Вычислительные методы линейной алгебры. СПб.: Изд-во Лань, 2002.

2. Борзых А. Н. Улучшение одной из оценок скорости сходимости метода Зейделя путем выбора оптимального порядка уравнений системы линейных алгебраических уравнений // Ж. вычисл. матем. и матем. физ. 2017. Т. 57, № 1. С. 3–8.

Статья поступила в редакцию 8 сентября 2018 г.;
после доработки 10 декабря 2018 г.;
рекомендована в печать 20 декабря 2018 г.

Контактная информация:

Борзых Алексей Николаевич — канд. физ.-мат. наук, доц.; alex@borz.ru

Enhancing an estimate of the rate of the Seidel method convergence

A. N. Borzykh

St. Petersburg State University, Universitetskaya nab., 7–9, St. Petersburg, 199034, Russian Federation

For citation: Borzykh A. N. Enhancing an estimate of the rate of the Seidel method convergence. *Vestnik of Saint Petersburg University. Mathematics. Mechanics. Astronomy*, 2019, vol. 6 (64), issue 2, pp. 185–195. <https://doi.org/10.21638/11701/spbu01.2019.201> (In Russian)

The article discusses the Seidel method for solving a system of linear algebraic equations and an estimate of the rate of the Seidel method convergence. It is proposed to construct an equivalent system for which the Seidel method also converges, but the rate of convergence is better. An equivalent system is constructed by a separate iterative process, where each single step requires $O(n)$ operations. Stability of this iterative process is proved. Results of numerical experiments are presented showing an improvement of the estimate of the rate of convergence.

Keywords: Seidel method, one-step cyclic process, SLAE, system of linear algebraic equations, iterative solution methods, Seidel method convergence, estimate of the rate of the Seidel method convergence.

References

1. Faddeev D. K., Faddeeva V. N., *Computational methods of linear algebra* (Lan' Publ., St. Petersburg, 2002). (In Russian)

2. Borzykh A. N., "Improving an estimate of the convergence rate of the seidel method by selecting the optimal order of equations in the system of linear algebraic equations", *Comput. Math. Math. Phys.* (57) (1), 1–6 (2017). <https://doi.org/10.1134/S0965542517010055>

Received: September 8, 2018

Revised: December 10, 2018

Accepted: December 20, 2018

Author's information:

Alexey N. Borzykh — alex@borz.ru