



УДК 519.612

Неявная итерационная схема на основе алгоритма псевдообращения и ее применения

А. И. Жданов^{1,2}, Ю. В. Сидоров¹

¹ Самарский государственный технический университет,
Россия, 443100, Самара, ул. Молодогвардейская, 244.

² Филиал ФГБОУ ВО «СамГТУ» в г. Новокуйбышевске,
Россия, 446200, Новокуйбышевск, ул. Миронова, 5.

Аннотация

Предложена новая версия неявной итерационной схемы, для реализации которой требуются лишь матрично-векторные вычислительные процедуры. Это делает предлагаемую вычислительную схему потенциально высокоэффективной для решения широкого класса задач большой размерности на современных высокопроизводительных вычислительных платформах, например Nvidia Cuda. Показано, что предлагаемые алгоритмы могут быть использованы для решения плохо обусловленных линейных систем и задач наименьших квадратов, а также для построения итерационных алгоритмов регуляризации. Приводятся результаты вычислительных экспериментов, подтверждающие эффективность предлагаемых вычислительных алгоритмов.

Ключевые слова: неявная итерационная схема, метод простых итераций, плохо обусловленные задачи, итерационное псевдообращение Бен–Израэля, итерационная регуляризация, матрично-векторные операции.

Получение: 29 мая 2023 г. / Исправление: 12 февраля 2024 г. /
Принятие: 4 марта 2024 г. / Публикация онлайн: 22 мая 2024 г.

Математическое моделирование, численные методы и комплексы программ Научная статья

© Коллектив авторов, 2024

© СамГТУ, 2024 (составление, дизайн, макет)

Контент публикуется на условиях лицензии [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/deed.ru) (<https://creativecommons.org/licenses/by/4.0/deed.ru>)

Образец для цитирования

Жданов А. И., Сидоров Ю. В. Неявная итерационная схема на основе алгоритма псевдообращения и ее применения // Вестн. Сам. гос. техн. ун-та. Сер. Физ.-мат. науки, 2024. Т. 28, № 1. С. 117–129. EDN: **HIGWRZ**. DOI: [10.14498/vsgtu2026](https://doi.org/10.14498/vsgtu2026).

Сведения об авторах

Александр Иванович Жданов <https://orcid.org/0000-0001-6082-9097>

доктор физико-математических наук, профессор; профессор; каф. прикладной математики и информатики¹; профессор; каф. электроэнергетики, электротехники и автоматизации технологии процессов²; e-mail: zhdanov.ai@samgtu.ru

Юрий Вячеславович Сидоров <https://orcid.org/0000-0002-8138-9200>

кандидат физико-математических наук; доцент; каф. прикладной математики и информатики; e-mail: sidorov.yv@samgtu.ru

Введение. Неявные итерационные схемы широко применяются для решения различных задач вычислительной математики, таких как плохо обусловленные и сингулярные системы линейных алгебраических уравнений (СЛАУ) большой размерности [1], задач наименьших квадратов, а также в итерационных методах регуляризации решений приближенных СЛАУ [2]. Известно, что одним из существенных преимуществ неявных итерационных схем перед явными является их абсолютная сходимость и достаточно высокая вычислительная устойчивость, но эти схемы не лишены недостатков.

В [2] предложена вычислительная схема неявного итерационного метода на базе сингулярного разложения (SVD-разложения). Однако применение этой вычислительной схемы приводит к существенному увеличению вычислительной работы и усложнению алгоритмической реализации итерационного алгоритма. Кроме того, использование в алгоритме SVD-разложения приводит к тому, что этот алгоритм не может быть адаптирован для эффективной обработки разреженных матриц. В конечном итоге все это серьезно осложняет практическое применение данного алгоритма для решения задач большой размерности на современных высокопроизводительных вычислительных платформах.

В [3] предложена еще одна неявная итерационная схема на основе расширенных линейных систем. Эта схема представляет большой интерес для решения плохо обусловленных больших разреженных линейных систем и задач наименьших квадратов, а также итерационных алгоритмов регуляризации. Однако для ее применения требуется использовать известные прямые методы решения («решатели») СЛАУ.

В настоящей работе предлагается новый вариант неявного метода простых итераций с использованием итерационного алгоритма псевдообращения Бен—Израэля [4]. Этот вариант алгоритма имеет полностью итерационный характер и обладает алгоритмической простотой. Для реализации этого метода требуются лишь матрично-векторные вычислительные процедуры, при его применении нет необходимости использования дополнительных стандартных «решателей» СЛАУ. Этот факт делает его потенциально более эффективным, чем указанные методы [2, 3], для решения задач большой размерности на современных высокопроизводительных вычислительных платформах, например Nvidia Cuda.

С вычислительной точки зрения наиболее затратной составляющей предлагаемого метода является итерационный метод Бен—Израэля. Однако учитывая специальную структуру матрицы с использованием «регуляризирующего» параметра, который одновременно выполняет роль предобуславливателя в задаче, данный метод, как показывают многочисленные вычислительные эксперименты, обладает высокой скоростью сходимости и вычислительной устойчивостью.

1. Постановка задачи. Рассмотрим задачу решения СЛАУ вида

$$Au = f, \quad (1)$$

где $A \in \mathbb{R}^{m \times n}$, $m \geq n$, $\text{rank } A = n$.

Псевдорешение исходной СЛАУ (1) определяется как $u_* = A^+ f$, где A^+ — псевдообратная матрица Мура—Пенроуза [4].

Отметим, что $A^+ = (A^\top A)^{-1}A^\top$, поскольку предполагается, что матрица A имеет полный столбцовый ранг, т.е. $\text{rank } A = n$, и, следовательно, является левой обратной матрицей.

В [2] показано, что в общем виде неявный метод простых итераций можно представить выражением

$$u_{k+1} = \underset{u \in \mathbb{R}^n}{\operatorname{argmin}} \left\| \begin{pmatrix} A \\ \sqrt{\alpha} E_n \end{pmatrix} u - \begin{pmatrix} f \\ \sqrt{\alpha} u_k \end{pmatrix} \right\|_2^2, \quad (2)$$

где

$$\begin{pmatrix} A \\ \sqrt{\alpha} E_n \end{pmatrix} \in \mathbb{R}^{(m+n) \times n}, \quad \begin{pmatrix} f \\ \sqrt{\alpha} u_k \end{pmatrix} \in \mathbb{R}^{m+n},$$

E_n — единичная матрица порядка n , $\|\cdot\|_2$ — евклидова векторная норма, $\alpha > 0$, $k = 0, 1, \dots$.

Из (2) видно, что различные варианты реализации неявного метода простых итераций отличаются лишь методом решения на каждой итерации линейной задачи наименьших квадратов:

$$\min_{u \in \mathbb{R}^n} \left\| \begin{pmatrix} A \\ \sqrt{\alpha} E_n \end{pmatrix} u - \begin{pmatrix} f \\ \sqrt{\alpha} u_k \end{pmatrix} \right\|_2^2. \quad (3)$$

В зависимости от методов решения задачи (3) получаются различные варианты неявных итерационных схем.

Первый подход основан на применении метода нормальных уравнений. В этом случае получаем классическую форму неявного метода простых итераций, которая была предложена J.D. Riley в [5]:

$$(A^\top A + \alpha E_n) u_{k+1} = A^\top f + \alpha u_k. \quad (4)$$

Второй подход для решения задачи наименьших квадратов (3) предполагает использование метода, основанного на QR-разложении.

Третий подход основан на SVD-разложении. Этот подход рассмотрен в работе [2].

В [3] показано, что неявная итерационная схема (4) решения задачи (3) эквивалентна решению последовательности регуляризованных расширенных систем

$$\begin{pmatrix} \omega E_m & A \\ A^* & -\omega E_n \end{pmatrix} \begin{pmatrix} y_{k+1} \\ u_{k+1} \end{pmatrix} = \begin{pmatrix} f \\ -\omega u_k \end{pmatrix}, \quad (5)$$

где $y_k = \omega^{-1}(f - Au_k)$, $\omega = \sqrt{\alpha}$.

Итерационная схема (5) имеет существенно более высокую скорость сходимости по сравнению с неявной итерационной схемой (4) и значительно меньшее число обусловленности вычислительной задачи. Важнейшим достоинством итерационной схемы (5) является возможность решать большие разреженные СЛАУ. Это обусловлено тем, что для решения расширенных систем (5) можно использовать хорошо известные «решатели», адаптированные для решения разреженных СЛАУ.

2. Новый вариант неявного итерационного метода. В настоящей работе предлагается новый вариант неявного метода простых итераций с использованием итерационного алгоритма псевдообращения Бен—Израэля.

Пусть

$$\left\| \begin{pmatrix} A \\ \sqrt{\alpha} E_n \end{pmatrix} u - \begin{pmatrix} f \\ \sqrt{\alpha} u_k \end{pmatrix} \right\|_2^2 = \|A_\alpha u - f_k^\alpha\|_2^2,$$

где

$$A_\alpha = \begin{pmatrix} A \\ \sqrt{\alpha} E_n \end{pmatrix} \in \mathbb{R}^{(m+n) \times n}, \quad f_k^\alpha = \begin{pmatrix} f \\ \sqrt{\alpha} u_k \end{pmatrix} \in \mathbb{R}^{m+n}.$$

Как показано в [2], итерационную схему (2) можно представить в виде

$$u_{k+1} = A_\alpha^+ f_k^\alpha. \quad (6)$$

Пусть $A_\alpha^+ = (U_\alpha : V_\alpha) \in \mathbb{R}^{n \times (m+n)}$, где $U_\alpha \in \mathbb{R}^{n \times m}$, $V_\alpha \in \mathbb{R}^{n \times n}$. Тогда неявный метод простых итераций (6) можно записать в виде

$$u_{k+1} = U_\alpha f + \sqrt{\alpha} V_\alpha u_k. \quad (7)$$

Для последовательности $(u_k)_{k=0}^\infty$, определяемой итерационной формулой (7) при произвольном начальном u_0 , справедливо $\lim_{k \rightarrow \infty} \|u_k - u_*\|_2 = 0$, где u_* — псевдорешение СЛАУ (1).

Так как матрица A_α всегда имеет полный столбцовый ранг, т.е. $\text{rank } A_\alpha = n$, для вычисления A_α^+ можно воспользоваться итерационным методом Бен—Израэля [4]:

$$X_{i+1} = (2E_n - X_i A_\alpha) X_i, \quad (8)$$

где $X_i \in \mathbb{R}^{n \times (m+n)}$, $i = 0, 1, 2, \dots$.

Итерационный метод Бен—Израэля (8) имеет квадратичную скорость сходимости [4] и, как отмечено в [6, 7], высокую численную устойчивость.

Пусть $X_i = (Y_i : Z_i)$, где $Y_i \in \mathbb{R}^{n \times m}$, $Z_i \in \mathbb{R}^{n \times n}$. Тогда (8) можно записать в виде

$$X_{i+1} = [2E_n - (Y_i A + \sqrt{\alpha} Z_i)] X_i \quad (9)$$

или

$$(Y_{i+1} : Z_{i+1}) = (2E_n - Y_i A - \sqrt{\alpha} Z_i)(Y_i : Z_i).$$

Если $X_0 = \beta A_\alpha^\top = \beta (A^\top : \sqrt{\alpha} E_n)$ или $Y_0 = \beta A^\top$, $Z_0 = \beta \sqrt{\alpha} E_n$ и

$$0 < \beta < \frac{2}{\sigma_1^2(A_\alpha)} = \frac{2}{\sigma_1^2(A) + \alpha},$$

где $\sigma_1(A_\alpha)$ и $\sigma_1(A)$ — максимальные сингулярные числа матриц A_α и A соответственно, то последовательность (8) при $i \rightarrow \infty$ сходится к A_α^+ [4].

Так как $\sigma_1^2 = \|A\|_2^2 \leq \|A\|_F^2$, можно принять, что

$$\beta = \frac{1.8}{\|A\|_F^2 + \alpha},$$

где $\|A\|_F$ — сферическая матричная норма (норма Фробениуса), $\|A\|_2$ — спектральная матричная норма [8, § 2.3].

Критерий останковки итерационного процесса (9): $i = 0, 1, \dots, i(\delta)$, где

$$i(\delta) = \min \left\{ i \in \mathbb{N} : \frac{\|X_{k+1} - X_k\|_\infty}{1 + \|X_k\|_\infty} \leq \delta \right\},$$

δ — априори заданное число для останковки итерационного процесса (8) по относительной скорости сходимости, $\|X\|_\infty$ — матричная норма, подчиненная кубической векторной норме (∞ -норме) [8, § 2.3].

Тогда (7) принимает вид

$$\tilde{u}_{k+1} = \sqrt{\alpha} V_\delta \tilde{u}_k + g_\delta, \quad (10)$$

где $X_\delta = X_{i(\delta)}$, а $U_\delta = Y_{i(\delta)}$, $V_\delta = Z_{i(\delta)}$ и $g_\delta = U_\delta f$.

3. Вычислительные эксперименты. Проведем сравнение предлагаемого в статье итерационного алгоритма (10) для решения задач вида (1) с методами, основанными на вычислении псевдообратной матрицы A^+ , т.е. когда решения находятся по формуле $u_* = A^+ f$.

В первом случае псевдообратная матрица A^+ вычисляется на основе хорошо известного алгоритма сингулярного разложения (SVD-разложения).

Во втором случае псевдообратная матрица A^+ вычисляется известным итерационным методом Бэн—Израэля. В случае, когда матрица A задачи (1) — невырожденная квадратная матрица, для вычисления обратной матрицы применяется итерационный метод Шульца [9], который является частным случаем итерационного алгоритма псевдообращения Бен—Израэля.

Все дальнейшие вычисления выполняются на Python 3.7.4.

3.1. Решение плохо обусловленных совместных СЛАУ. Рассмотрим решение плохо обусловленных совместных СЛАУ вида (1) с квадратными матрицами $A \in \mathbb{R}^{n \times n}$.

Для первого вычислительного эксперимента матрица A системы (1) генерировалась с помощью функции `deriv2` из пакета `RTools` [10], вектор точного решения брался в виде $u_{\text{ext}} = (1, 2, \dots, n)^\top$, а вектор правой части определялся по формуле $f = Au_{\text{ext}}$. Спектральное число обусловленности матрицы A определяется по формуле $\kappa_2(A) = \sigma_1/\sigma_n$, где σ_1 и σ_n — максимальное и минимальное сингулярные числа матрицы A соответственно.

Решение задачи (1) находилось с помощью следующих подходов:

- решение вычислялось как $u_{\text{svd}} = A^+ f$, где матрица A^+ вычислялась на основе SVD-алгоритма;
- нахождение обратной матрицы A^{-1} с помощью итерационного алгоритма Шульца и вычисление решения задачи (1);
- вычисление решения задачи (1) с помощью предлагаемого в статье алгоритма (10).

Критерий останковки итерационных алгоритмов Шульца и Бен—Израэля:

$$\frac{\|X_{k+1} - X_k\|_\infty}{1 + \|X_k\|_\infty} \leq 10^{-7}.$$

Критерий останковки итерационного алгоритма (10):

$$\text{Err} = \frac{\|u_{k+1} - u_k\|_\infty}{1 + \|u_k\|_\infty} \leq 10^{-16}.$$

При использовании итерационного алгоритма (10) для рассматриваемой задачи параметр $\omega = \sqrt{\alpha}$ выбирался равным $\sigma_n/2$, σ_n , $2\sigma_n$, $3\sigma_n$.

В табл. 1 и на рис. 1 представлены результаты вычислений для задачи (1), когда $n = 512$, $\kappa_2(A) = 3.19 \cdot 10^5$, $\sigma_n = 3.17 \cdot 10^{-7}$, полученные при использовании итерационного алгоритма (10). Здесь и далее $\kappa_2(A_\alpha)$ — спектральное число обусловленности матрицы A_α , it_1 — число итераций алгоритма (10), it_2 — число итераций алгоритма Бен—Израэля, $RErr = \frac{\|u_{ext} - u_k\|_2}{\|u_{ext}\|_2}$ — относительная ошибка вычислений.

Таблица 1

Результаты вычислений для тестовой задачи (решение плохо обусловленных совместных СЛАУ (1)) при использовании итерационного алгоритма (10)

[The results of computations for a test problem (solving the ill-conditioned consistent systems of linear equations (1)) by the iteration algorithm (10)]

ω	$\kappa_2(A_\alpha)$	it_1	it_2	RErr
$\sigma_n/2$	$2.85 \cdot 10^5$	23	41	$1.90 \cdot 10^{-11}$
σ_n	$2.25 \cdot 10^5$	53	40	$1.88 \cdot 10^{-11}$
$2\sigma_n$	$1.43 \cdot 10^5$	151	39	$1.52 \cdot 10^{-11}$
$3\sigma_n$	$1.01 \cdot 10^5$	309	38	$2.16 \cdot 10^{-11}$

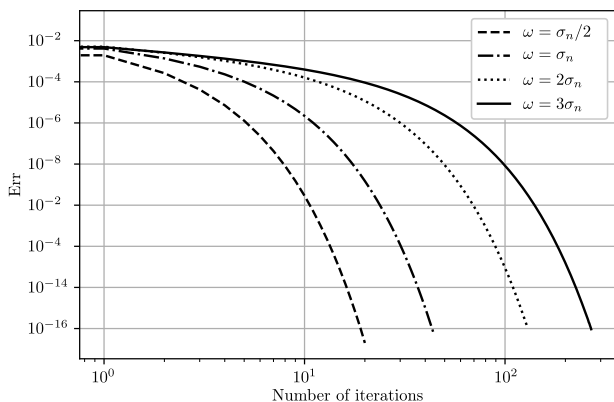


Рис. 1. Скорость сходимости итерационного алгоритма (10) при решении плохо обусловленных совместных СЛАУ (1)

[Figure 1. Convergence rate of the iterative algorithm (10) when solving ill-conditioned systems of linear equations (1)]

Отметим, что для этой же задачи для SVD-алгоритма получена относительная ошибка вычислений, равная $1.62 \cdot 10^{-10}$, а для алгоритма Шульца при числе итераций, равном 41, получена относительная ошибка вычислений, равная $5.75 \cdot 10^{-8}$.

Данный вычислительный эксперимент демонстрирует, что предлагаемый в работе итерационный алгоритм (10) по точности превосходит метод на основе итерационного алгоритма Шульца и не уступает методам, основанным на сингулярном разложении, применяемым для решения подобных плохо обусловленных совместных СЛАУ. Итерационный метод Бен—Израэля вычисления псевдообратной матрицы для предложенного способа выбора параметра ω , имеет достаточно высокую скорость сходимости и не сильно зависит от параметра ω .

Рассматриваемый в статье итерационный алгоритм (10) при выборе параметра ω , близким к минимальному сингулярному числу σ_n матрицы A , позволяет вычислять решения плохо обусловленных задач вида (1) с достаточно высокой скоростью и точностью. Увеличение параметра ω для задач данного класса приводит к значительному замедлению итерационного процесса, что, в свою очередь, может вызвать потерю точности получаемого решения.

Как отмечалось ранее, одними из преимуществ предлагаемого алгоритма (10) являются алгоритмическая простота и возможность реализации итерационного процесса только с помощью матрично-векторных операций.

3.2. Решение плохо обусловленных линейных задач наименьших квадратов. Рассмотрим плохо обусловленную линейную задачу наименьших квадратов

$$Au = f, \quad (11)$$

где

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 10^{-8} & 0 & 0 & 0 & 0 \\ 0 & 10^{-8} & 0 & 0 & 0 \\ 0 & 0 & 10^{-8} & 0 & 0 \\ 0 & 0 & 0 & 10^{-8} & 0 \\ 0 & 0 & 0 & 0 & 10^{-8} \end{pmatrix},$$

$$u = (1, 1, 1, 1, 1)^\top, f = Au + r_0, r_0 \in \text{null}(A^\top).$$

Учитывая, что вектор невязки $r_0 \in \text{null}(A^\top)$, псевдорешение $u_* = A^+ f$ системы (11) равно вектору u . Спектральное число обусловленности матрицы A задачи (11) равно $\kappa_2(A) = 2.24 \cdot 10^8$.

Решение задачи (11) с помощью псевдообратной матрицы, вычисленной с применением SVD-алгоритма, получено с относительной ошибкой вычислений, равной $9.42 \cdot 10^{-10}$. Решение этой задачи итерационным алгоритмом Бен—Израэля при числе итераций, равном 60, получено с относительной ошибкой вычислений, равной $1.46 \cdot 10^{-9}$. При этом в качестве критерия остановки итерационного алгоритма Бен—Израэля использовалось условие

$$\frac{\|X_{k+1} - X_k\|_\infty}{1 + \|X_k\|_\infty} \leq 10^{-7}.$$

В табл. 2 представлены результаты вычислений для задачи (11), полученные с помощью алгоритма (10). Здесь σ_5 — минимальное сингулярное значение матрицы A для задачи (11). При этом в качестве критерия остановки

Таблица 2

Результаты вычислений для тестовой задачи (решение плохо обусловленной линейной задачи наименьших квадратов (11)) при использовании итерационного алгоритма (10)

[The results of computations for a test problem (solving the ill-conditioned linear least squares problem (11)) by the iteration algorithm (10)]

ω	$\kappa_2(A_\omega)$	it ₁	it ₂	RErr
σ_1	≈ 1	64	7	$5.98 \cdot 10^{-15}$
$\sigma_1/100$	$\approx 1 \cdot 10^2$	7	18	$2.67 \cdot 10^{-16}$
σ_5	$\approx 1 \cdot 10^8$	30	59	$3.67 \cdot 10^{-8}$

итерационного процесса использовалось условие

$$\frac{\|u_{k+1} - u_k\|_\infty}{1 + \|u_k\|_\infty} \leq 10^{-16}.$$

Анализ результатов вычислительного эксперимента демонстрирует, что при решении плохо обусловленных линейных задач наименьших квадратов предлагаемый алгоритм (10) по точности не уступает известным методам решения этой задачи, а соответствующий выбор параметра ω позволяет получать более точные решения. Алгоритм Бен—Израэля для данного вычислительного эксперимента, как и для предыдущего, демонстрирует достаточно высокую скорость сходимости и близкую к линейной зависимость от параметра ω .

3.3. Решение задач с возмущенными исходными данными. В рассмотренных выше вычислительных экспериментах предложенный итерационный алгоритм (10) применялся для нахождения решения задач вида (1) с точными исходными данными (A, f) . Очень часто при решении различных прикладных задач вместо точных данных (A, f) имеются возмущенные исходные данные $(A, \tilde{f})$. Наиболее широко применяемыми на практике подходами для решения задач вида (1) с возмущенными исходными данными $(A, \tilde{f})$ являются методы регуляризации.

Рассматриваемый итерационный метод (10) при выборе критерия останова, в котором номер итерации играет роль параметра регуляризации (правило останова по невязке), можно отнести к итерационным алгоритмам регуляризации.

Проиллюстрируем регуляризирующие свойства итерационного алгоритма (10) на следующей задаче решения СЛАУ:

$$Au = f, \quad (12)$$

где

$$A = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 + 10^{-8} & 1 - 10^{-8} \end{pmatrix}, \quad f = (1, 1)^\top.$$

Максимальное и минимальное сингулярные числа матрицы A для задачи (12) соответственно равны $\sigma_1 = 1$ и $\sigma_2 = 5 \cdot 10^{-9}$, а спектральное число обусловленности задачи $\kappa_2(A) = 2 \cdot 10^8$. Рассматриваемая задача относится к плохо обусловленным вычислительным задачам регрессионного анализа.

Вектор $u = (1, 1)^\top$ является точным решением задачи (12).

Рассмотрим задачу (12) с небольшим возмущением в векторе правой части:

$$Au = \tilde{f}, \quad (13)$$

где $\tilde{f} = f + (0.01, 0)^\top$.

Решая систему (13) известными методами, например с помощью процедуры `solve` библиотеки `NumPy`, получим $u_* = (-10^6, 10^6)^\top$.

В работе [11] для решения возмущенной задачи (13) предлагалось использовать усеченное сингулярное разложение (TSVD). Для возмущенной задачи (13) в [11] параметр усечения был принят равным $\text{tol} = 10^{-7}$ и получено решение $u_{\text{tsvd}} = (1.0050, 1.0049)^\top$.

Найдем решение возмущенной задачи (13) с помощью итерационного алгоритма (10) со следующими параметрами: $\delta = 0.01$, а $\omega = \sqrt{\alpha}$ со значениями σ_1 , $\sigma_1/2$, $\sigma_1/5$.

Критерий остановки итерационного процесса примем в виде

$$k_\delta = \min\{k \in \mathbb{N} : \|Au_k - \tilde{f}\| \leq \tau\delta\}, \quad (14)$$

где $\tau \geq 1$ — параметр демпфирования (в вычислительном эксперименте использовалось значение $\tau = 1.01$). Выражение (14) представляет собой итерационный принцип невязки Тихонова.

Результаты вычислений представлены в табл. 3 и на рис. 2.

Таблица 3

Результаты вычислений для тестовой задачи (решение возмущенной задачи (13)) при использовании итерационного алгоритма (10)

[The results of computations for a test problem (solving the perturbed problem (13)) by the iteration algorithm (10)]

ω	Number of iterations, k	$\ u - u_k\ /\ u\ $	u_k
σ_1	8	$1.07 \cdot 10^{-3}$	$(1.0011, 1.0011)^\top$
$\sigma_1/2$	4	$3.39 \cdot 10^{-3}$	$(1.0033, 1.0033)^\top$
$\sigma_1/5$	2	$3.51 \cdot 10^{-3}$	$(1.0035, 1.0035)^\top$

Полученные результаты позволяют сделать вывод, что предложенный алгоритм (10) с критерием остановки по итерационному принципу невязки (14) по точности не уступают методу регуляризации, основанному на сингулярном разложении. При этом важно отметить, что для решения задачи итерационной регуляризации можно использовать оценку максимального сингулярного числа σ_1 , например $\omega_1 = \|A\|_F$, которая может быть легко вычислена в отличие от оценки минимального сингулярного числа σ_n .

4. Заключение. Одним из преимуществ предлагаемого в статье итерационного алгоритма является его простая алгоритмическая структура. Основными операциями алгоритма являются матрично-векторные произведения, для которых существуют эффективные программные реализации на современных высокопроизводительных платформах. Например, данный алгоритм может быть реализован с использованием высокопроизводительных библиотек Intel® Math Kernel Library [12] для многоядерных/многопроцессорных систем, CUDA Toolkit [13] для систем с графическими ускорителями общего назначения или адаптирован к конкретной вычислительной платформе.

Предлагаемый в статье алгоритм позволяет получать высокоэффективные алгоритмы итерационной регуляризации, основанные на итерационном принципе невязки (discrepancy principle [14]).

Конкурирующие интересы. Конкурирующих интересов не имеем.

Авторская ответственность. Мы несем полную ответственность за предоставление окончательной версии рукописи в печать. Окончательная версия рукописи нами одобрена.

Финансирование. Исследование выполнялось без финансирования.

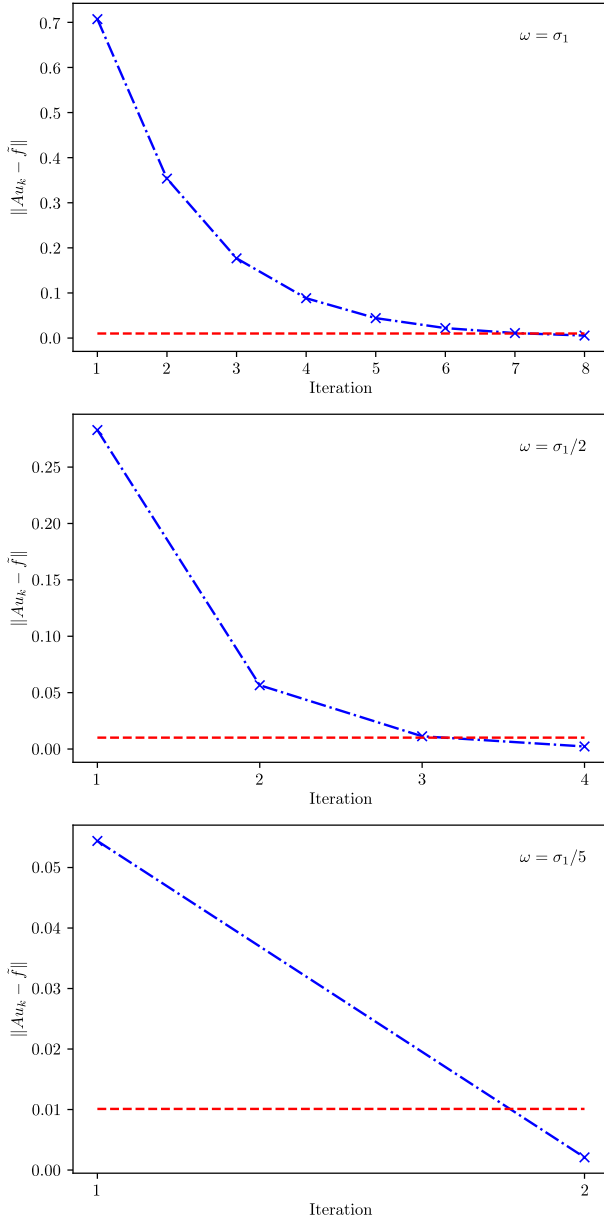


Рис. 2. Скорость сходимости итерационного процесса при решении возмущенной задачи (13); горизонтальная линия соответствует значению $\tau\delta$

[Figure 2. Convergence rate of the iterative process when solving the perturbed problem (13); the horizontal line corresponds to the value $\tau\delta$]

Библиографический список

1. Sun L., Wei Y., Zhou J. On an iterative method for solving the least squares problem of rank-deficient systems // *Int. J. Comp. Math.*, 2015. vol. 92, no. 3. pp. 532–541. DOI: <https://doi.org/10.1080/00207160.2014.900173>.
2. Zhdanov A. I. Implicit iterative schemes based on singular decomposition and regularizing algorithms // *Vestn. Samar. Gos. Tekhn. Univ., Ser. Fiz.-Mat. Nauki* [J. Samara State Tech. Univ., Ser. Phys. Math. Sci.], 2018. vol. 22, no. 3. pp. 549–556. EDN: PJITAX. DOI: <https://doi.org/10.14498/vsgtu1592>.
3. Жданов А. И. Неявная итерационная схема на основе расширенных линейных систем // *Докл. РАН. Матем., информ., проц. упр.*, 2022. Т. 503. С. 91–94. EDN: COMJBF. DOI: <https://doi.org/10.31857/S2686954322020205>.
4. Ben-Israel A., Greville T. N. E. *Generalized Inverses: Theory and Applications*. New York: Springer, 2003. xv+420 pp. DOI: <https://doi.org/10.1007/b97366>.
5. Riley J. D. Solving systems of linear equations with a positive definite, symmetric, but possibly ill-conditioned matrix // *Math. Comp.*, 1955. vol. 9. pp. 96–101. DOI: <https://doi.org/10.1090/S0025-5718-1955-0074915-1>.
6. Esmaeili H., Erfanifar R., Rashidi M. An efficient Schulz-type method to compute the Moore–Penrose inverse // *Int. J. Industr. Math.*, 2018. vol. 10, no. 2. pp. 221–228.
7. Toutounian F., Soleymani F. An iterative method for computing the approximate inverse of a square matrix and the Moore–Penrose inverse of a non-square matrix // *Appl. Math. Comp.*, 2013. vol. 224. pp. 671–680. DOI: <https://doi.org/10.1016/j.amc.2013.08.086>.
8. Golub G. H. van Loan C. F. *Matrix Computations*. Baltimore: Johns Hopkins Univ., 2013. xxiv+756 pp.
9. Schulz G. Iterative Berechnung der reziproken Matrix // *ZAMM*, 1933. vol. 13, no. 1. pp. 57–59. DOI: <https://doi.org/10.1002/zamm.19330130111>.
10. Hansen P. C. REGULARIZATION TOOLS: A Matlab package for analysis and solution of discrete ill-posed problems // *Numer. Algor.*, 1994. vol. 6. pp. 1–53. DOI: <https://doi.org/10.1007/BF02149761>.
11. Pillonetto G., Chen T., Chiuso A., et al. *Regularized System Identification: Learning Dynamic Models from Data*. Cham: Springer, 2022. xxiv+377 pp. DOI: <https://doi.org/10.1007/978-3-030-95860-2>.
12. Wang E., Zhang Q., Shen B. et al. Intel math kernel library / *High-Performance Computing on the Intel® Xeon Phi™*. Cham: Springer, 2014. pp. 167–188. DOI: https://doi.org/10.1007/978-3-319-06486-4_7.
13. Fatica M. CUDA toolkit and libraries / *2008 IEEE Hot Chips 20 Symposium* (24–26 August 2008). Stanford, CA, USA, 2008. pp. 1–22. DOI: <https://doi.org/10.1109/HOTCHIPS.2008.7476520>.
14. Zare H., Hajarian M. Determination of regularization parameter via solving a multi-objective optimization problem // *Appl. Num. Math.*, 2020. vol. 156. pp. 542–554. DOI: <https://doi.org/10.1016/j.apnum.2020.05.021>.

MSC: 65F10, 65F22

Implicit iterative scheme based on the pseudo-inversion algorithm and its application

A. I. Zhdanov^{1,2}, Yu. V. Sidorov¹

¹ Samara State Technical University,
244, Molodogvardeyskaya st., Samara, 443100, Russian Federation.

² Samara State Technical University, Novokuybyshevsk Branch,
5, Mironova st., Novokuybyshevsk, 446200, Russian Federation.

Abstract

A new version of the implicit iterative scheme is proposed for the implementation of which only matrix-vector computational procedures are required. This makes the proposed computational scheme potentially highly efficient for solving a wide class of high-dimensional problems on modern high-performance computing platforms, such as Nvidia Cuda. It is shown that the proposed algorithm can be used to solve ill-conditioned linear systems and least squares problems, as well as to construct iterative regularization algorithms. The results of computational experiments are presented, confirming the effectiveness of the proposed computational algorithms.

Keywords: implicit iterative scheme, simple iteration method, ill-conditioned problems, Ben-Israel iterative pseudo-inversion, iterative regularization, matrix-vector operations.

Received: 29th May, 2023 / Revised: 12th February, 2024 /

Accepted: 4th March, 2024 / First online: 22nd May, 2024

Competing interests. We have no competing interests.

Authors' contributions and responsibilities. We take full responsibility for submitting the final version of the manuscript for publication. The final version of the manuscript has been approved by us.

Funding. The research was conducted without funding.

**Mathematical Modeling, Numerical Methods and Software Complexes
Research Article**

© Authors, 2024

© Samara State Technical University, 2024 (Compilation, Design, and Layout)

 The content is published under the terms of the [Creative Commons Attribution 4.0 International License](http://creativecommons.org/licenses/by/4.0/) (<http://creativecommons.org/licenses/by/4.0/>)

Please cite this article in press as:

Zhdanov A. I., Sidorov Yu. V. Implicit iterative scheme based on the pseudo-inversion algorithm and its application, *Vestn. Samar. Gos. Tekhn. Univ., Ser. Fiz.-Mat. Nauki* [J. Samara State Tech. Univ., Ser. Phys. Math. Sci.], 2024, vol. 28, no. 1, pp. 117–129. EDN: [HIGWRZ](https://doi.org/10.14498/vsgtu2026). DOI: [10.14498/vsgtu2026](https://doi.org/10.14498/vsgtu2026) (In Russian).

Authors' Details:

[Aleksander I. Zhdanov](https://orcid.org/0000-0001-6082-9097)  <https://orcid.org/0000-0001-6082-9097>

Dr. Phys. & Math. Sci., Professor; Professor; Dept. of Applied Mathematics and Informatics¹; Professor; Dept. of Electrical Power Engineering, Electrical Engineering, and Automation Process Technology²; e-mail: zhdanov.ai@samgtu.ru

[Yuri V. Sidorov](https://orcid.org/0000-0002-8138-9200)  <https://orcid.org/0000-0002-8138-9200>

Cand. Phys. & Math. Sci.; Associate Professor; Dept. of Applied Mathematics and Informatics; e-mail: sidorov.yv@samgtu.ru

References

1. Sun L., Wei Y., Zhou J. On an iterative method for solving the least squares problem of rank-deficient systems, *Int. J. Comp. Math.*, 2015, vol. 92, no. 3, pp. 532–541. DOI: <https://doi.org/10.1080/00207160.2014.900173>.
2. Zhdanov A. I. Implicit iterative schemes based on singular decomposition and regularizing algorithms, *Vestn. Samar. Gos. Tekhn. Univ., Ser. Fiz.-Mat. Nauki* [J. Samara State Tech. Univ., Ser. Phys. Math. Sci.], 2018, vol. 22, no. 3, pp. 549–556. EDN: PJITAX. DOI: <https://doi.org/10.14498/vsgtu1592>.
3. Zhdanov A. I. Implicit iterative schemes based on augmented linear systems, *Dokl. Math.*, 2022, vol. 105, no. 2, pp. 131–134. EDN: SNWVMI. DOI: <https://doi.org/10.1134/S106456242202020X>.
4. Ben-Israel A., Greville T. N. E. *Generalized Inverses: Theory and Applications*. New York, Springer, 2003, xv+420 pp. DOI: <https://doi.org/10.1007/b97366>.
5. Riley J. D. Solving systems of linear equations with a positive definite, symmetric, but possibly ill-conditioned matrix, *Math. Comp.*, 1955, vol. 9, pp. 96–101. DOI: <https://doi.org/10.1090/S0025-5718-1955-0074915-1>.
6. Esmaeili H., Erfanifar R., Rashidi M. An efficient Schulz-type method to compute the Moore–Penrose inverse, *Int. J. Industr. Math.*, 2018, vol. 10, no. 2, pp. 221–228.
7. Toutounian F., Soleymani F. An iterative method for computing the approximate inverse of a square matrix and the Moore–Penrose inverse of a non-square matrix, *Appl. Math. Comp.*, 2013, vol. 224, pp. 671–680. DOI: <https://doi.org/10.1016/j.amc.2013.08.086>.
8. Golub G. H. van Loan C. F. *Matrix Computations*. Baltimore, Johns Hopkins Univ., 2013, xxiv+756 pp.
9. Schulz G. Iterative Berechnung der reziproken Matrix, *ZAMM*, 1933, vol. 13, no. 1, pp. 57–59. DOI: <https://doi.org/10.1002/zamm.19330130111>.
10. Hansen P. C. REGULARIZATION TOOLS: A Matlab package for analysis and solution of discrete ill-posed problems, *Numer. Algor.*, 1994, vol. 6, pp. 1–53. DOI: <https://doi.org/10.1007/BF02149761>.
11. Pillonetto G., Chen T., Chiuso A., et al. *Regularized System Identification: Learning Dynamic Models from Data*. Cham, Springer, 2022, xxiv+377 pp. DOI: <https://doi.org/10.1007/978-3-030-95860-2>.
12. Wang E., Zhang Q., Shen B. et al. Intel math kernel library, In: *High-Performance Computing on the Intel® Xeon Phi™*. Cham, Springer, 2014, pp. 167–188. DOI: https://doi.org/10.1007/978-3-319-06486-4_7.
13. Fatica M. CUDA toolkit and libraries, In: *2008 IEEE Hot Chips 20 Symposium* (24–26 August 2008). Stanford, CA, USA, 2008, pp. 1–22. DOI: <https://doi.org/10.1109/HOTCHIPS.2008.7476520>.
14. Zare H., Hajarian M. Determination of regularization parameter via solving a multi-objective optimization problem, *Appl. Num. Math.*, 2020, vol. 156, pp. 542–554. DOI: <https://doi.org/10.1016/j.apnum.2020.05.021>.