

ОБЩИЕ
ЧИСЛЕННЫЕ МЕТОДЫ

УДК 519.6

ИТЕРАЦИОННЫЕ ПРЕДОБУСЛОВЛЕННЫЕ МЕТОДЫ
В ПОДПРОСТРАНСТВАХ КРЫЛОВА: ТЕНДЕНЦИИ XXI ВЕКА¹⁾

© 2021 г. В. П. Ильин^{1,2}

¹ 630090 Новосибирск, пр-т Акад. Лаврентьева, 6, Институт вычислительной математики
и математической геофизики СО РАН, Россия

² 630073 Новосибирск, пр-т К. Маркса, 20, Новосибирский государственный технический университет, Россия
e-mail: ilin@sscc.ru

Поступила в редакцию 11.02.2020 г.
Переработанный вариант 16.03.2021 г.
Принята к публикации 07.07.2021 г.

Предлагается аналитический обзор основных проблем, а также новых математических и технологических находок в развитии методов решения СЛАУ. Данная стадия математического моделирования становится “узким горлышком”, поскольку здесь объемы вычислительных ресурсов растут нелинейно с увеличением числа степеней свободы задачи. Важно отметить, что эффективность и производительность вычислительных методов и технологий в значительной степени зависят от учета специфики класса решаемых прикладных проблем: задачи электромагнетизма, гидро-газодинамики, упруго-пластичности, многофазной фильтрации, тепломассопереноса и т.д. Развитие крыловских итерационных процессов ориентировано главным образом на построение двухуровневых алгоритмов с различными ортогональными, проекционными, вариационными и спектральными свойствами, включая аппарат не только полиномиальных, но и рациональных или гармонических приближений. Дополнительное ускорение таких алгоритмов осуществляется на основе подходов дефляции или агментации с использованием некоторых систем базисных векторов. Активные исследования направлены на конструирование экономичных предобуславливающих операторов, на основе многообразных принципов: новые многосеточные схемы и параллельные методы декомпозиции областей, мультипредобуславливание, вложенные и попеременно-треугольные факторизации, малоранговые и другие алгоритмы аппроксимации обратных матриц и т.д. Достижение высокой производительности и масштабируемого распараллеливания базируется на средствах гибридного программирования с использованием инструментов межузловых сообщений, многопоточковых вычислений, векторизации операции и графических ускорителей. Современные тенденции математического и программного обеспечения заключаются в создании интегрированного инструментального окружения, ориентированного на длительный жизненный цикл и массовые инновации в актуальных приложениях. Библ. 98.

Ключевые слова: разреженные СЛАУ, предобуславливание, итерационные методы, подпространства Крылова, симметричные и несимметричные матрицы, алгоритмы декомпозиции, многосеточные подходы, приближенная факторизация.

DOI: 10.31857/S0044466921110090

1. ВВЕДЕНИЕ

Предобусловленные итерационные методы в подпространствах Крылова для решения систем линейных алгебраических уравнений (СЛАУ) – одно из важнейших мировых достижений вычислительной математики в XX веке. Их значение особенно актуализируется с развитием суперкомпьютерного моделирования и решением междисциплинарных прямых и обратных, нелинейных и нестационарных многомерных проблем со сложными геометрическими и материальными свойствами. Современные требования к точности решения задач с реальными данными приводят к системам сверхбольших размерностей (10^{10} – 10^{11} и более) и огромными числами обусловленности (до 10^{14} и выше), когда вычисления со стандартной двойной точностью становятся

¹⁾ Работа выполнена при финансовой поддержке РФФИ (код проекта 19-11-50073) по конкурсу обзорных статей “Экспансия”, а также грантам МиНВО РФ № 2020-0012.

уже на грани доступного. При этом вычислительные затраты на решение СЛАУ составляют больше 80% для крупномасштабных машинных экспериментов.

Наибольший исследовательский интерес представляют алгебраические системы с разреженными матрицами, возникающими из аппроксимаций краевых задач методами конечных разностей, конечных объемов, конечных элементов и разрывными алгоритмами Галеркина различных порядков точности на неструктурированных сетках (см. обзор в [1]). В этих случаях портреты, или графы, матриц имеют нерегулярную структуру, т.е. распределение ненулевых элементов задается только перечислением, а их значения хранятся в сжатых форматах, что составляет важную особенность при реализации алгоритмов на современных супер-ЭВМ гетерогенной архитектуры с распределенной и иерархической общей памятью. Спектральные и структурные свойства СЛАУ сильно отличаются для различных приложений: задач электромагнетизма, гидрогазодинамики, упруго-пластичности, уравнений тепло-массопереноса и т.д., что порождает огромное разнообразие алгоритмов, развитие которых идет в направлении как исследования новых итерационных процессов с различными ортогональными, вариационными и проекционными свойствами, так и построения эффективных предобусловленных матриц. Симбиоз этих двух отдельных методологий и определяет успех в данной области. В значительной степени итоги исследований были отражены в книгах О. Аксельсона (см. [2]), Х. Эльмана, Д. Сильвестра и А. Ватена (см. [3]), Г.И. Марчука и Ю.А. Кузнецова (см. [4]), Й. Саада (см. [5]), Ван дер Ворста (см. [6]), М.А. Ольшанского и Е.Е. Тыртышниковой (см. [7]), Й. Лайзена и З. Стракоса (см. [8]), автора [9], [10], а также в огромном количестве журнальных статей и трудов конференций.

Если говорить об обобщении крыловских итерационных методов, то здесь следует отметить многообразные подходы к обогащению соответствующих подпространств, в которых из различных вариационных, ортогональных или проекционных принципов определяются векторы невязок или направлений поиска новых итерационных приближений, включая алгоритмы дефляции, агментации и комбинирования со спектральной оптимизацией или с принципами наименьших квадратов. Любопытно, что возникали также попытки построить некоторые “альтернативы” методам в подпространствах Крылова: ускорение Андерсона (которое изначально было предложено для решения нелинейных систем, см. [11]–[13]), подпространства Сонневельда (см. [14]), но на проверку они оказывались вариациями на общую тему.

Все возрастающий поток литературы по методам решения СЛАУ содержит богатую палитру идей по конструированию предобуславливающих матриц, которые должны легко обращаться, повышать скорость сходимости итерационных алгоритмов, а в итоге обеспечивать их производительность в целом. Здесь вторую молодость обрели многосеточные методы, которые асимптотически являются оптимальными по порядку (объем вычислений пропорционален числу степеней свободы дискретной задачи). Значительное развитие получили также методологии матричных разложений, в том числе малоранговые приближения матриц, вложенные и перемноженные факторизации. Разнообразные подходы сформировали единообразную технологию построения мультипредобусловленных процессов в подпространствах Крылова, в общем случае многоуровневых, а также аддитивных алгоритмов декомпозиции областей (МДО, или DDM – от Domain Decomposition Methods), представляющих собой главный инструмент распараллеливания для многомерных задач на многопроцессорных вычислительных системах (МВС). Методы масштабируемого распараллеливания и обеспечения высокой производительности алгоритмов DDM на суперкомпьютерах гетерогенной архитектуры с распределенной и иерархической общей памятью представляют ключевую проблему современной вычислительной алгебры (см. [5], [16]–[19]). Отметим, что свою специфику имеют методы решения СЛАУ, возникающих при неявных аппроксимациях начально-краевых задач (см. обзор в [20]). В частности, как показано в [21], за счет выбора на каждом временном шаге начальных итерационных приближений на основе обобщения алгоритма предиктор-корректор, можно значительно повысить эффективность расчетов.

Практическая востребованность алгебраических вычислений за полувековую историю привела к огромному объему программного обеспечения, как коммерческого, так и общедоступного, достаточно полный список которого приведен Дж. Донгарра в [22], а систематизация реализуемых алгоритмов приведена в [23]. Здесь следует отметить и общезначимые инструментарии типа SPARSE BLAS, и широкого распространения библиотеки PETSc, HYPRE, MKL INTEL и т.д. Международным сообществом вычислителей-алгебраистов разработаны стандартизованные форматы представления матриц с соответствующими конверторами, а также обширные коллекции матриц из реальных задач моделирования, играющих незаменимую роль для систематического тестирования и сравнительного анализа алгоритмов. Важной тенденцией послед-

них десятилетий является переход от конкретных библиотек и пакетов прикладных программ (ППП) к интегрированным вычислительным окружениям (ИВО), примерами чего являются системы DUNE, INMOST, OPEN FOAM, MATLAB, а также БСМ (Базовая Система Моделирования), концепция которой описана в [24] и включает принципы гибкого расширения состава моделей и алгоритмов, адаптацию к эволюции компьютерных архитектур, переиспользование внешних программных продуктов и согласованное участие различных групп разработчиков, что должно облегчить создание эффективной инструментальной экосистемы нового поколения с длительным жизненным циклом, объединяющей сообщества математиков-программистов и пользователей из различных прикладных областей. В целом проблема высокопроизводительного решения СЛАУ для широкого класса практических задач вовлекает огромный объем разнообразных задач и представляет собой перспективное поле деятельности по интеллектуализации как исследовательских, так и технологических направлений.

Данная работа построена следующим образом. В разд. 2 представлены общие современные принципы конструирования итерационных методов в подпространствах Крылова, в том числе двухуровневых. Раздел 3 содержит краткий обзор подходов к построению предобуславливающих матриц, в разд. 4 описываются технологии распараллеливания и повышения производительности методов решения СЛАУ, включая вопросы их программной реализации и эффективного использования в проблемах моделирования реальных процессов и явлений, а в Заключении обсуждаются перспективы развития исследуемых вопросов.

2. ОРТОГОНАЛЬНЫЕ И ВАРИАЦИОННЫЕ ПРИНЦИПЫ ПОСТРОЕНИЯ КРЫЛОВСКИХ ИТЕРАЦИОННЫХ МЕТОДОВ

Мы будем рассматривать вещественные алгебраические системы вида

$$Au = f, \quad A = \{a_{i,m}\} \in \mathcal{R}^{N,N}, \quad u = \{u_i\}, \quad f = \{f_i\} \in \mathcal{R}^N, \quad (1)$$

с положительно-полуопределенными матрицами

$$(Av, v) \geq \delta \|v\|^2, \quad \delta \geq 0, \quad (v, w) = \sum_{i=1}^N v_i w_i, \quad \|v\|^2 = (v, v),$$

среди которых важный класс представляют симметричные положительно-определенные (СПО) матрицы с $\delta > 0$. Решаемые СЛАУ могут быть представлены в блочной форме

$$A_{q,q}u_q + \sum_{r \in \Omega_q} A_{q,r}u_r = f_q, \quad q = 1, 2, \dots, P, \quad (2)$$

$$A = \{A_{q,r}\}, \quad A_{q,r} \in \mathcal{R}^{N_q, N_r}, \quad f_q \in \mathcal{R}^{N_q}, \quad N_1 + \dots + N_P = N,$$

где Ω_q есть множество номеров матричных строк, составляющих q -ю блочную строку матрицы A , а P – ее блочный порядок.

Одна из актуальных блочных структур порождает алгебраические системы седлового типа:

$$Au \equiv \begin{bmatrix} D & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} f_1 \\ 0 \end{bmatrix} = f, \quad D \in \mathcal{R}^{N_1, N_1}, \quad C \in \mathcal{R}^{N_2, N_1}, \quad u_1, f_1 \in \mathcal{R}^{N_1}, \quad (3)$$

которые возникают из аппроксимаций смешанных постановок краевых задач, а также в методах оптимизации. Матрица A из (3) обратима, если выполняются условия

$$\text{rank}(D) = N_1, \quad \text{rank}(C) = N_2, \quad \ker(D) \cap \ker(C) = \{0\}, \quad N_2 \leq N_1.$$

Наряду с исходной будем рассматривать предобуславливающие матрицы, которые обозначим символом B с какими-либо индексами. Выбор предобуславливателей осуществляется по условиям экономической обратимости и улучшения обусловленности предобусловленных матриц вида $B^{-1}A$ или AB^{-1} , которые в совокупности должны повысить производительность программной реализации алгоритма.

Достаточно общим представлением стационарного одношагового итерационного процесса для решения СЛАУ (1) является так называемая первая каноническая формула

$$u^{n+1} = u^n + B^{-1}r^n, \quad r^n = f - Au^n,$$

где r^n есть вектор невязки. Ей соответствует вторая каноническая форма

$$u^{n+1} = Tu^n + g, \quad T = (1 - B^{-1}A), \quad g = -B^{-1}f,$$

где T называется матрицей перехода. Критерием окончания итераций является выполнение следующего условия (см. подробнее [25], [26]):

$$\|r^n\| \leq \varepsilon_1 \|f\| + \varepsilon_2 \|A\| \cdot \|u^n\|, \quad 0 < \varepsilon_1, \varepsilon_2 \ll 1,$$

хотя зачастую для простоты полагается $\varepsilon_2 = 0$. Вопрос об оценке обеспечиваемой при этом ошибке итерационного приближения $\|u - u^n\|$ отнюдь не тривиальный, в особенности при учете используемой точности машинной арифметики, и ему посвящено достаточно много серьезных исследований (см. [8] и цитируемую там литературу).

2.1. Алгоритмы для симметричных СЛАУ

Зарождение и первый период развития, который можно отнести к 1950–1965 годам, итерационных методов в подпространствах Крылова, следует связать с именами авторов пионерских работ К. Ланцоша (1950, 1952), В.Е. Арнольди (1951), М.Р. Хестенеса и Е.Л. Штифеля (1951, 1952), Е.Х. Крейга (1954, 1955), Ю.В. Воробьева (1958), Д.К. Фаддеева и В.Н. Фаддеевой (1958, 1960), В.М. Фридмана (1962), Дж. Голуба (1965) и Р. Варги (1962). Список последующих исследований насчитывается сотнями.

Мы начнем рассмотрение алгоритмов решения симметричных СЛАУ, имея целью дать единообразный взгляд на эту активно развивающуюся и сейчас тематику, а также осветить актуальные вопросы вычислительной производительности методов.

Пусть итерационное приближение u^n и соответствующий вектор невязки $r^n = f - Au^n$ вычисляются по следующим формулам:

$$\begin{aligned} u^n &= u^{n-1} + \alpha_{n-1} p^{n-1} = u^0 + \alpha_0 p^0 + \dots + \alpha_{n-1} p^{n-1}, \\ r^n &= r^{n-1} - \alpha_{n-1} A p^{n-1} = r^0 - \alpha_0 A p^0 - \dots - \alpha_{n-1} A p^{n-1}, \end{aligned} \quad (4)$$

где α_k и p^k суть итерационные параметры и направляющие векторы. В данном представлении мы опишем семейство алгоритмов сопряженных направлений, характеризующихся различными вариационными и проекционными свойствами, определяемыми соответствующими скалярными произведениями и нормами векторов:

$$(u, v)_\gamma = (A^\gamma u, v), \quad \|u\|_\gamma^2 = (u, u)_\gamma, \quad (5)$$

где показатели степени будем брать равными $\gamma = 0, 1, 2$. Предположим, что направляющие векторы являются A^γ -ортогональными, т.е.

$$(A^\gamma p^n, p^k) = \rho_k^{(\gamma)} \delta_{n,k}, \quad \rho_k^{(\gamma)} = (A^\gamma p^k, p^k) = \|p^k\|_\gamma^2, \quad (6)$$

где $\delta_{n,k}$ есть символ Кронекера. Тогда невязки r^n удовлетворяют соотношениям

$$\Phi_\gamma(r^n) = (A^{\gamma-2} r^n, r^n) = (r^0, r^0)_{\gamma-2} - \sum_{k=0}^{n-1} [2\alpha_k (r^0, A^{\gamma-1} p^k) - \alpha_k^2 \rho_k]. \quad (7)$$

Отсюда следует, что при значении параметров

$$\alpha_k = \sigma_k / \rho_k, \quad \sigma_k = (r^0, p^k)_{\gamma-1} = (r^k, p^k)_{\gamma-1} \quad (8)$$

функционалы Φ_γ принимают минимальные значения

$$\Phi_\gamma(r^n) = \|r^0\|_{\gamma-2}^2 - \sum_{k=0}^{n-1} (r^0, p^k)_{\gamma-1}^2 / \rho_k. \quad (9)$$

Здесь и далее индекс γ у коэффициентов и векторов для краткости опускается, а матрица A пока предполагается невырожденной.

Для выполнения условия (6) есть два пути. Первый из них состоит в проведении A^γ -ортогонализации Ланцоша, которую можно записать в следующем виде:

$$\begin{aligned} p^0 &= 0, \quad \bar{\beta}_1 p^1 = f, \quad k = 1, 2, \dots, n: \\ \bar{\beta}_{n+1} p^{n+1} &= Ap^n - \bar{\alpha}_n p^n - \bar{\beta}_n p^{n-1}, \quad \bar{\alpha}_n = (p^n, p^n)_{\gamma+1}, \end{aligned} \quad (10)$$

где величины $\bar{\beta}_n$ выбираются по условию $\|p\|_\gamma = 1$. Другой путь (Хестенеса–Штифеля), имеющий наибольшее распространение в силу лучшей устойчивости, заключается в использовании двухчленной рекурсии

$$p^0 = r^0, \quad p^n = r^n + \beta_{n-1} p^{n-1}, \quad \beta_{n-1} = (A^\gamma r^n, p^{n-1}) / \rho_{n-1}, \quad (11)$$

который совместно с формулой для невязки в (4) приводит к другим соотношениям для направляющих векторов:

$$p^n = (1 + \beta_{n-1}) p^{n-1} - \alpha_{n-1} A p^{n-1} - \beta_{n-2} p^{n-2}. \quad (12)$$

С помощью соотношений (4), (11) можно установить свойство $A^{\gamma-1}$ -ортогональности векторов невязок, а также более удобные (для $\gamma = 1, 2$) формулы итерационных коэффициентов:

$$(A^{\gamma-1} r^n, r^k) = \sigma_k \delta_{n,k}, \quad \sigma_k = \|r^n\|_{\gamma-1}^2, \quad \beta_k = \sigma_{k+1} / \sigma_k. \quad (13)$$

Для случая $\gamma = 0$, однако, вычисление α_k необходимо делать другим образом. Поскольку точное решение СЛАУ может быть представлено в виде разложения по базису

$$u = u^0 + \alpha_0 p^0 + \dots + \alpha_{m-1} p^{m-1}, \quad m \leq N,$$

то векторы ошибки итерационного приближения и соответствующей невязки записываются в следующей форме:

$$\begin{aligned} v^n &= u - u^n = \alpha_n p^n + \dots + \alpha_m p^m, \\ r^n &= Av^n = \alpha_n A p^n + \dots + \alpha_m A p^m. \end{aligned} \quad (14)$$

Отсюда с помощью A^γ -ортогонализации векторов p^k имеем

$$\alpha_n = (v^n, p^n)_\gamma / \|p^n\|_\gamma^2 = -\alpha_{n-1} (v^n, A p^{n-1}) / \|p^n\|_\gamma^2 = -\alpha_{n-1} (r^n, p^{n-1})_\gamma / \|p^n\|_\gamma^2. \quad (15)$$

Здесь использована ортогональность p^n векторам p^{n-1} и p^{n-2} , а также его связь с вектором $A p^{n-1}$ из (12). Расчет коэффициента β_{n-1} при этом надо проводить по формуле (11).

Рассматриваемые итерационные процессы обладают оптимальными свойствами, обеспечивая минимизацию соответствующих функционалов $\Phi_\gamma(r^n)$ вида (7) в подпространствах Крылова

$$\mathcal{K}_n(r^0, A) = \text{Span}(r^0, Ar^0, \dots, A^{n-1} r^0). \quad (16)$$

Данные алгоритмы при $\gamma = 0, 1, 2$ имеют названия методов минимальных итераций, или ошибок (см. [8]), а также сопряженных градиентов и сопряженных невязок соответственно (см. [5], [9], [27]).

Описанные подходы допускают простое обобщение на СЛАУ, преобусловленные с помощью некоторых СПО матриц B . Для сохранения симметричности систем это сделать целесообразно путем двухстороннего преобусловливания с помощью формально вводимой матрицы $B^{1/2}$. В результате система (1) принимает вид

$$\bar{A} \bar{u} = \bar{f}, \quad \bar{A} = B^{-1/2} A B^{-1/2}, \quad \bar{u} = B^{1/2} u, \quad \bar{f} = B^{-1/2} f. \quad (17)$$

В результате применения формул сопряженных направлений к СЛАУ (17) после некоторых преобразований для $\gamma = 1, 2$ получаем следующий итерационный процесс:

$$\hat{p}^0 = \hat{r}^0 = B^{-1} r^0 = B^{-1} (f - Au^0), \quad n = 1, 2, \dots,$$

$$\begin{aligned}
 u^n &= u^{n-1} + \alpha_{n-1} \hat{p}^{n-1}, & \hat{r}^n &= \hat{r}^{n-1} - \alpha_{n-1} A \hat{p}^{n-1}, \\
 \hat{p}^n &= \hat{r}^n + \beta_{n-1} \hat{p}^{n-1}, & \alpha_{n-1} &= \sigma_{n-1} / \rho_{n-1}, & \beta_{n-1} &= \sigma_n / \sigma_{n-1}, \\
 \sigma_{n-1} &= (A^{\gamma-1} \hat{r}^{n-1}, \hat{r}^{n-1}), & \rho_{n-1} &= (B^{-1} A \hat{p}^{n-1}, A^{\gamma-1} \hat{p}^{n-1}),
 \end{aligned}
 \tag{18}$$

где новые векторы связаны со старыми соотношениями $\hat{p}^n = B^{-1} p^n$, $\hat{r}^n = B^{-1} r^n$. Для метода минимальных итераций с $\gamma = 0$ вычисление параметров $\alpha_{n-1}, \beta_{n-1}$ следует проводить по формулам (15), (11) с заменой величин r^k, p^k на $\bar{r}^k = B^{-1} r^k$, $\bar{p}^k = B^{-1} p^k$ соответственно.

Замечание 1. На практике зачастую актуальным является решение серии СЛАУ с одинаковой матрицей, но разными последовательно определяемыми правыми частями. В этом случае при реализации второй и последующих систем можно сэкономить значительный объем вычислений, если после первого расчета запомнить найденные векторы p^k и $A p^k$ и использовать их каким-то образом для расширения крыловского базиса при решении остальных СЛАУ. Этим вопросам посвящено много работ (см. [28] со списком литературы из 157 наименований), однако проблема существенного повышения “информативности” базиса итерационных процессов здесь пока требует дальнейших исследований.

Замечание 2. В книге [8] отмечался такой методический вопрос, что методы в подпространствах Крылова тесно связаны с проблемой моментов, играющей важную роль в теории операторов и многих прикладных задачах. Начала этой тематики заложены П.Л. Чебышевым, А.М. Марковым и Т. Стильтьесом.

Во многих актуальных приложениях большой интерес представляют алгебраические системы с симметричными матрицами, имеющими знакоопределенный спектр. Такие СЛАУ могут быть вырожденными, в том числе совместными или несовместными. Последние задачи имеют решение не в классическом смысле, а в обобщенном, т.е. в терминах проблемы наименьших квадратов. Для данных случаев определяется нормальное, или псевдообратное, решение, имеющее наименьшую норму и обеспечивающее минимум невязки:

$$\min \|u\|_2 : u = \operatorname{argmin} \|f - Au\|_2. \tag{19}$$

Такое решение всегда существует и является единственным, а его формальное представление получается с помощью левой трансформации Гаусса исходной системы ($A^T A u = A^T f$) и имеет вид

$$u^+ = (A^T A)^+ f = (A^2)^+ A f, \tag{20}$$

где A^+ означает псевдообратную, или обобщенно обратную, матрицы A .

Для решения таких СЛАУ в работах М.А. Саундерса и соавт. (см. обзор в [29]) разработаны методы SYMLQ, MINRES, MINRES–QLP, основанные на A -ортогонализации Ланцоша, которая в силу рекурсий (10) записывается в матричном виде как

$$A P_k = P_{k+1} T_k, \quad T_k = \begin{bmatrix} \bar{\alpha}_1 & \bar{\beta}_1 & & & \\ & \bar{\beta}_2 & \bar{\alpha}_2 & & \\ & & \ddots & \ddots & \bar{\beta}_k \\ & & & \bar{\beta}_k & \bar{\alpha}_k \\ & & & & \bar{\beta}_{k+1} \end{bmatrix}, \tag{21}$$

где $P_k = [p^0, p^1, \dots, p^{k-1}]$. При этом приближенное решение ищется в форме $u^k = P_k q^k$ и сводится к определению вектора $q^k \in \mathcal{R}^k$ по условию минимума невязки

$$r^k = f - A P_k q^k = \beta_1 p^1 - P_{k+1} T_k y^k. \tag{22}$$

Данная задача реализуется с помощью LQ- или QR-преобразований трехдиагональной матрицы T_k с ортогональной матрицей Q на основе устойчивых операций отражения Хаусхолдера. Приведенные авторами результаты представительных численных экспериментов демонстрируют хорошую эффективность алгоритмов, а их программные реализации доступны в Интернете.

Скорость сходимости итераций в методах крыловского типа для положительно полуопределенного СЛАУ характеризуется оценкой числа итераций $n(\varepsilon)$, необходимой для подавления начальной ошибки в ε^{-1} раз:

$$n(\varepsilon) \leq \frac{1}{2} \sqrt{\kappa} \log_e \frac{2}{\varepsilon}, \quad \kappa = \lambda_{\max} / \lambda_{\min},$$

где κ в общем случае есть эффективное число обусловленности, которое для вырожденных матриц выражается через λ_{\min} – минимальное ненулевое собственное число.

Для симметричных СЛАУ проблема устойчивого численного решения в основном была решена в XX веке, в том числе для знаконеопределенных, вырожденных и несовместных систем. Что касается классического метода сопряженных градиентов, то здесь особо следует отметить оригинальную теорию сходимости И.Е. Капорина (см. [30]), основанную на введенном им K -числе обусловленности, выражаемого через след и определитель матрицы.

Более конкретно рассмотрим предобусловленный алгоритм сопряженных градиентов в следующем виде:

$$\begin{aligned} r^0 &= f - Au^0, \quad p^0 = B^{-1}r^0, \quad n = 0, 1, \dots, \\ u^{n+1} &= u^n + \alpha_n p^n, \quad r^{n+1} = r^n - \alpha_n A p^n, \\ p^{n+1} &= B^{-1}r^{n+1} + \beta_n p^n, \quad \alpha_n = \sigma_n / \rho_n, \\ \beta_n &= \sigma_{n+1} / \sigma_n, \quad \sigma_n = (r^n, B^{-1}r^n), \quad \rho_n = (p^0, A p^n). \end{aligned}$$

При этом для выполнения условия

$$(r^n, B^{-1}r^n) \leq \varepsilon^2 (r^1, B^{-1}r^1)$$

в [31] доказывается достаточность проведения числа итераций

$$n(\varepsilon) = \log_2 K + \log_2 \varepsilon^{-1},$$

где K – число обусловленности предобусловленной матрицы $B^{-1}A$, определяемое как

$$K = K(B^{-1}A) = \left[\frac{1}{N} \text{trace}(B^{-1}A) \right]^N / \det(B^{-1}A).$$

В [32] для ряда предобусловливателей проведена оптимизация итерационных алгоритмов в плане минимизации K – числа обусловленности.

2.2. Крыловские алгоритмы для несимметричных СЛАУ

Изложение конкретных подходов мы продолжим с достаточно широкого класса мульти-предобусловленных алгоритмов полусопряженных направлений (SCD – Semi-Conjugate Direction, см. [31]). В общей блочной форме такие итерационные методы в подпространствах Крылова записываются следующим образом:

$$\begin{aligned} r^0 &= f - Au^0, \quad n = 0, \dots: \quad u^{n+1} = u^n + P_n \bar{\alpha}_n, \\ r^{n+1} &= r^n - AP_n \bar{\alpha}_n = r^q - AP_q \bar{\alpha}_q - \dots - AP_n \bar{\alpha}_n, \quad 0 \leq q \leq n, \\ P_n &= (p_1^n, \dots, p_{M_n}^n) \in \mathcal{R}^{N, M_n}, \quad \bar{\alpha}_n = (\alpha_{n,1}, \dots, \alpha_{n, M_n})^T \in \mathcal{R}^{M_n}. \end{aligned} \quad (23)$$

Здесь $p_1^n, \dots, p_{M_n}^n$ – направляющие векторы, составляющие на n -й итерации матрицу P_n , а $\bar{\alpha}_n$ – вектор итерационных параметров. Относительно векторов p_k^n в соотношениях (23) пока предполагается выполнение только условий ортогональности

$$\begin{aligned} (Ap_k^n, A^\gamma p_{k'}^{n'}) &= \rho_{n,k}^{(\gamma)} \delta_{n,n'}, \quad \rho_{n,k}^{(\gamma)} = (Ap_k^n, A^\gamma p_k^n), \\ \gamma &= 0, 1, \quad n' = 0, 1, \dots, n-1, \quad k, k' = 1, 2, \dots, M_n. \end{aligned} \quad (24)$$

Однако, если при этом коэффициенты $\bar{\alpha}_n = \{\alpha_{n,l}\}$ определить по формулам

$$\alpha_{n,l} = \sigma_{n,l} / \rho_{n,n}^{(\gamma)}, \quad \sigma_{n,l} = (r^0, A^\gamma \bar{p}_l^n), \quad (25)$$

то из (23) получаем следующие выражения для функционалов невязки:

$$\Phi_n^{(\gamma)}(r^{n+1}) \equiv (r^{n+1}, A^{\gamma-1} r^{n+1}) = (r^q, A^{\gamma-1} r^q) - \sum_{k=q}^n \sum_{l=1}^{M_n} (r^q, A^\gamma p_l^k)^2 / \rho_{k,l}^{(\gamma)}, \quad q = 0, 1, \dots, n, \quad (26)$$

которые достигают своих минимумов в блочных подпространствах Крылова

$$K_M = \text{Span}\{p_1^0, \dots, p_{M_0}^0, Ap_1^1, \dots, Ap_{M_1}^1, \dots, Ap_1^n, \dots, Ap_{M_n}^n\}, \quad M = M_0 + M_1 + \dots + M_n, \quad (27)$$

при $\gamma = 1$, а в случае симметричности матрицы A и для $\gamma = 0$. Здесь следует отметить, что для несимметричных матриц при $\gamma = 0$ применение алгоритмов фактически ограничивается СЛАУ с матрицами, имеющими положительно определенную симметричную часть $A_s = 0.5(A + A^T)$, так как для кососимметричной матрицы, например, $(Au, u) = 0$ при $u = \{1\}$.

Свойства ортогональности направляющих векторов (24) можно обеспечить, если их определить с помощью “мультипредобусловленных” рекуррентных соотношений, в которых каждому вектору p_l^{n+1} соответствует “своя” предобуславливающая матрица $B_{n+1,l}$:

$$p_l^0 = B_{0,l}^{-1} r^0, \quad p_l^{n+1} = B_{n+1,l}^{-1} r^{n+1} - \sum_{k=0}^n \sum_{l=1}^{M_k} \beta_{n,k,l}^{(\gamma)} p_l^k, \quad n = 0, 1, \dots, \\ B_{n,l} \in \mathcal{R}^{N,N}, \quad i = 1, \dots, M_n; \quad \gamma = 0, 1, 2, \\ \bar{\beta}_{n,k}^{(\gamma)} = \{\beta_{n,k,l}^{(\gamma)}\} = (\beta_{n,k,1}^{(\gamma)} \dots \beta_{n,k,M_n}^{(\gamma)})^T \in \mathcal{R}^{M_n}, \quad (28)$$

$$\beta_{n,k,l}^{(\gamma)} = -(A^\gamma p_l^k, AB_{n+1,l}^{-1} r^{n+1}) / \rho_{n,l}^{(\gamma)}, \quad n = 0, 1, \dots, \quad k = 0, 1, \dots, n, \quad l = 1, 2, \dots, M_n.$$

Если матрица A является симметричной, то данные рекурсии превращаются из длинных в двухчленные, и мы приходим к методам сопряженных градиентов или сопряженных невязок ($\gamma = 0, 1$ соответственно, в мультипредобусловленных или в классических вариантах). Для несимметричных СЛАУ эти алгоритмы называются методами полусопряженных градиентов и полусопряженных невязок (SCG или SCR – Semi-Conjugate Gradient или Semi-Graduate Residual). В блочных (мультипредобусловленных) методах сопряженных направлений для решения симметричных СЛАУ формулы (23)–(25) не меняются, а вычисление направляющих матриц P_n выполняется по следующим двухчленным рекуррентным соотношениям:

$$P_{n+1} = Q_{n+1} + P_n \bar{\beta}_n^{(\gamma)}, \quad Q_{n+1} = \{q_l^{n+1} = B_{n+1,l}^{-1} r^{n+1}\} \in \mathcal{R}^{N, M_{n+1}}, \\ \bar{\beta}_n^{(\gamma)} = \{\beta_{n,l}^{(\gamma)} = \beta_{n,n,l}^{(\gamma)}\} \in \mathcal{R}^{M_n}.$$

Данные формулы, как и (28), содержат “встроенные” в них предобуславливающие матрицы. Если положить $B_{n+1,l} = I$ и $M_n = 1$ для всех n , то получим крыловский процесс “в чистом виде”, без предобуславливания. Отметим, что формулы (28) реализуют алгоритм ортогонализации Грама–Шмидта, для повышения устойчивости которого целесообразно перейти к модифицированному методу ортогонализации (MGS, см. [5], [33]).

Особенностью рассматриваемых алгоритмов при решении плохо обусловленных несимметричных СЛАУ является высокая ресурсоемкость, в смысле объемов и вычислений, и требуемой памяти при проведении большого количества итераций. Средство борьбы с данным недостатком – это сокращение количества хранимых и используемых направляющих векторов, которое может осуществляться двумя способами. Первый из них заключается в сокращении рекурсии с учетом только ее последних m векторов. Второй путь состоит в периодическом проведении рестартов, когда через заданное число m итераций вектор невязки вычисляется не из рекуррентной формулы, а из исходного уравнения, как на нулевой итерации:

$$r^{n_t} = f - Au^{n_t}, \quad n_t = mt, \quad t = 0, 1, \dots, \quad (29)$$

где t – номер рестарта (далее вычисления до $n = n_{t+1}$ осуществляются по обычным рекурсиям вида (23)). Оба данных подхода приводят к существенному замедлению итерационного процесса.

Для устранения такого стагнирующего эффекта предлагается добавить второй уровень итераций с применением методов наименьших квадратов (МНК) (см. [34]). Пусть нам известны рестартовые приближения $u^{n_0}, u^{n_1}, \dots, u^{n_t}, n_0 = 0$. Тогда для коррекции итерационного вектора u^{n_t} , который является начальным для очередного рестартового периода метода (23)–(29), будем использовать следующую линейную комбинацию:

$$\begin{aligned} \hat{u}^{n_t} &= u^{n_t} + b_1 v_1 + \dots + b_t v_t = u^{n_t} + v^{n_t}, \quad v^{n_t} = V_t \bar{b}, \quad \bar{b} = (b_1, \dots, b_t)^T, \\ V_t &= \{v_k = u^{n_k} - u^{n_{k-1}}, k = 1, \dots, t\} \in \mathcal{R}^{N, t}, \end{aligned} \quad (30)$$

вектор коэффициентов \bar{b} которой определяется по условию минимума нормы $\|r^{n_t}\|$ невязки из обобщенного решения переопределенной алгебраической системы

$$W_t \bar{b} = r^{n_t} \equiv f - Au^{n_t}, \quad W_t = AV_t. \quad (31)$$

Решение этой СЛАУ можно получить, например, с помощью QR - или SVD -разложений матрицы W_t . Нормальное решение с минимальной нормой $\|\bar{b}\|$ определяется после применения к (31) левой трансформации Гаусса

$$W_t^T W_t \bar{b} = W_t^T r^{n_t}.$$

Более облегченный формат СЛАУ, в смысле уменьшения ее числа обусловленности, следует после умножения системы (31) слева на матрицу V_t^T :

$$C_t \bar{b} \equiv V_t^T AV_t \bar{b} = V_t^T r^{n_t}. \quad (32)$$

Если матрица V_t имеет полный ранг, то матрицы A и C_t будут невырожденными одновременно. При этом для вектора коррекции из (30) имеем $v^{n_t} = B_t r^{n_t} \equiv V_t (V_t^T AV_t)^{-1} V_t^T r^{n_t}$, где матрица $B_t = V_t \hat{A}^{-1} V_t^T$, $\hat{A} = V_t^T AV_t$, является малоранговой аппроксимацией матрицы A^{-1} . В рассматриваемом подходе все рестартовые векторы запоминаются в скорректированном виде, а соответствующие невязки вычисляются по формуле $r^{n_t} = f - Au^{n_t}$. Если для какой-то рассматриваемой матрицы обратной не существует, то используется обобщенная обратная матрица. Многочисленные эксперименты с использованием МНК для ускорения крыловских процессов с рестартами показывают его высокую эффективность.

Отметим еще следующую возможность повышения производительности методов SCD с рестартами: при проведении итераций первого рестартового периода запомнить все направляющие векторы p^n , а также векторы Ap^n , и при вычислениях последующих рестартовых периодов новые векторы p^n и Ap^k не рассматривать, а использовать старые.

Описанный класс методов полусопряженных направлений с динамическим мультипредобусловливанием по скорости сходимости итераций эквивалентен другим известным алгоритмам решения несимметричных СЛАУ в подпространствах Крылова, среди которых наиболее популярен обобщенный метод минимальных невязок GMRES, основанный на ортогонализации Арнольди и существующий в различных вариантах. Данный алгоритм, предложенный в 1986 г. Й. Саадом и М. Шульцем (см. [5]), получил широкую и заслуженную популярность. Среди многочисленных его исследований отметим результаты Л.А. Книжнермана (см. [35]) по оценкам погрешности ортогонализации Арнольди.

Другой принцип построения крыловских итерационных процессов для решения несимметричных СЛАУ основан на построении последовательностей биортогональных векторов. В этом случае направляющие векторы вычисляются из коротких (двучленных) рекурсий, но на каждой итерации требуется двукратное вычисление векторно-матричного произведения. На идеях биортогонализации были построены алгоритмы BiCG, CGS, BiCGStab в разных модификациях, затем появились их аналоги с A -биортогонализацией направляющих векторов (см. обзор в [36]). Примыкающее к ним семейство алгоритмов IDR (Induced Dimension Reduction) (см. [37] и цитируемую там литературу) базируется на подпространствах Сонневельда. Среди других подходов следует рекомендовать методы квазиминимальных невязок QMR и наименьших квадратов (LSQR, LSMR) (см. [38]). В последние годы возродились также алгоритмы, основанные на bidiagonal преобразованиях Крейга–Голуба–Кахана (изначально предложенных в 1955 и в

1965 г.), содержательное исследование, обобщение и сравнительный анализ которых даны в [25], [39]. И наконец, отметим еще “гибкие” методы сопряженных градиентов (FCG) (см. [40]), являющиеся обобщением классических методов CG (в том числе предобусловленных) на несимметричный случай.

3. МЕТОДЫ ПРЕДОБУСЛОВЛИВАНИЯ СЛАУ

В данном разделе при описании алгоритмов мы будем на качественном уровне затрагивать вопросы их потенциального распараллеливания. Более подробно технологические аспекты, определяющие производительность их реализации, будут обсуждаться позднее.

Помимо рассмотренного в (17), (18) конкретного способа использования предобусловливающей матрицы, это можно также сделать путем непосредственного предобусловливания (левого, правого или двустороннего) исходной СЛАУ:

$$\begin{aligned} \bar{A}u &= \bar{f}, \quad \bar{A} = B^{-1}A, \quad \bar{f} = B^{-1}f, \\ \hat{A}\hat{u} &= AB^{-1}Bu = f, \quad \hat{A} = AB^{-1}, \quad \hat{u} = Bu, \\ \check{A}\check{u} &= B_1^{-1}AB_2^{-1}B_2u = B_1^{-1}f = \check{f}, \quad \check{A} = B_1^{-1}AB_2^{-1}, \quad \check{u} = B_2u, \end{aligned} \quad (33)$$

где B, B_1, B_2 – некоторые невырожденные матрицы. Отметим, что если матрица A симметрична, то для сохранения этого свойства у матрицы \check{A} в последнем случае нужно выбрать $B_1 = B_2^T$. Поскольку матрицы \bar{A}, \hat{A} или \check{A} полученных предобусловленных алгебраических систем уже должны иметь улучшенную обусловленность, то к их решению следует применять крыловские итерационные формулы “в чистом виде”, без дополнительного предобусловливания (хотя формально это не возбраняется, и можно использовать мультипредобусловливание, используя модификацию как итерационного процесса типа (28), так и самого СЛАУ, в соответствии с (33)).

Среди простейших способов предобусловливания мы отметим такие, как масштабирование и бинормализацию матриц. Первый из них заключается в диагональном конгруэнтном преобразовании ($\tilde{A} = DAD$) для получения единичной главной диагонали, а второй основан на выравнивании норм у строк и столбцов матрицы (см. обзор в [41], [42]). Такие подходы могут применяться как предварительные в сочетании с другими предобусловливателями.

Обращаясь к блочной структуре СЛАУ (2), наиболее естественным представляется применить для ее решения блочный метод Якоби, который в несколько обобщенном виде записывается как

$$B_{q,q}u_q^{n+1} \equiv (A_{q,q} + \theta D_q)u_q^{n+1} = f_q + \theta D_q u_q^n - \sum_{r \in \Omega_q} A_{q,r}u_r^n, \quad q = 1, 2, \dots, P. \quad (34)$$

Здесь $\theta \in [0, 1]$ есть итерационный (компенсирующий) параметр, а D_q – диагональная матрица, определяемая из равенства

$$D_q e = \sum_{r \in \Omega_q} A_{q,r} e, \quad e = (1, \dots, 1)^T \in \mathcal{R}^{N_q}, \quad (35)$$

которое называется условием компенсации, или фильтрации (иногда введение матрицы D_q вида (35) называется также лампированием – от английского lumping). В некоторых случаях выбор θ обосновывается и описывается теоретически, позволяя существенно ускорить итерационный процесс. Имеются также обобщения принципа компенсации (35) на основе усложнения матриц D_q с ленточной структурой и использования нескольких компенсирующих (фильтрующих) векторов (см. [2], [9], [10]):

$$D_q y_s = \sum_{r \in \Omega_q} A_{q,r} y_s, \quad s = 1, 2, \dots, S, \quad (36)$$

однако в целом данная проблема требует дальнейших исследований.

Альтернативные аналоги блочного метода Якоби, относящегося к классу итерационных алгоритмов одновременных смещений, или аддитивных, является метод Зейделя, а также его развитие – алгоритмы последовательной верхней релаксации и симметричной последовательной верхней релаксации (SOR и SSOR). Последний метод с некоторыми модификациями опублико-

вался также под именами “попеременно-треугольные”, неполной факторизации и “явные схемы переменных направлений” (см. [2], [9]). Эти методы последовательных смещений, или мультипликативные (новое приближение u_q^{n+1} может быть выполнено только после нахождения предыдущих подвекторов $u_1^{n+1}, \dots, u_{q-1}^{n+1}$), обладают повышенной скоростью сходимости, в том числе к ним применим ускоряющий принцип компенсации вида (35), (36). Однако такие алгоритмы основаны на обращениях треугольных матриц, которые плохо распараллеливаются. Поэтому в эпоху многопроцессорных суперкомпьютеров они оказались менее актуальны, и мы на них здесь останавливаться не будем. С другой стороны, в [43] показано, что применение адаптивных методов последовательных смещений “по потоку” может значительно сокращать количество итераций.

Для многочисленных методов предобусловливания имеются свои теоретические обоснования и оценки скорости сходимости итераций (см. обзор в [44]–[46]). Особый случай представляет собой динамическое, или гибкое, предобусловливание с переменными матрицами (см. [17], [47]–[51]). В итерационных процессах при решении СЛАУ, возникающих из аппроксимации краевых задач на сетке с характерным шагом h и имеющих число обусловленности порядка $O(h^{-2})$, его величина обычно понижается до $O(h^{-1})$ за счет предобусловливателя, в результате чего количество итераций при использовании методов крыловского типа оценивается как $n(\epsilon) = O(h^{-1/2})$. Однако следует иметь в виду, что эти асимптотические выражения могут включать большие константы в “плохих” задачах, характеризующихся сильным разбросом значений элементов матрицы A .

3.1. Алгоритмы декомпозиции областей

При решении сеточных СЛАУ, особенно узлового типа, каждая компонента искомого вектора ассоциируется с одним узлом сетки, а структура и портрет матрицы (конфигурация ее ненулевых элементов) изоморфны сеточному графу не направленному для симметричных матриц и направленному – для несимметричных). В этом случае матричная блочная структура вида (2) геометрически наглядно представляется как декомпозиция сеточной расчетной области Ω на сеточные подобласти Ω_q , в каждой из которых формируется своя подзадача, алгебраически соответствующая диагональному блоку $A_{q,q}$. Исторический приоритет в методах декомпозиции принадлежит теории многомерных краевых задач еще с XIX века, когда решение дифференциальных уравнений в сложных расчетных областях с помощью альтернирующего метода Шварца сводилось к исследованию последовательности вспомогательных задач в пересекающихся подобластях с использованием условий Дирихле на вводимых внутренних границах. Позднее такие подходы были обобщены на непрерывном уровне в работах В.В. Смелова, П.Л. Лионса, А.М. Мацокина, О. Видлунда, Ф. Натафа и соавт., Ю.В. Василевского, М.А. Ольшанского и многих других авторов (см. обзоры литературы в [5], [7], [17]–[19], [52]–[57]), а также перенесены на дискретный, или сеточный, уровень. В эпоху многопроцессорных суперкомпьютеров методы декомпозиции стали главным инструментом для распараллеливания вычислений при решении многомерных начально-краевых задач со сложными реальными данными, требующими высокого разрешения и оперативности расчетов.

Многообразие МДО можно классифицировать по трем главным характеристикам. Первая – это размерность декомпозиции, определяемая тем, проводится ли разбиение областей одним, двумя или тремя семействами координатных плоскостей. Вторая характеристика заключается в величине пересечений подобластей, которые в простейшем случае определяются без налегания, а параметризованные пересечения на сеточном уровне строятся путем послынного последовательного расширения каждой подобласти элементарными ячейками сетки. Третье направление развития МДО основано на обобщении типов интерфейсных краевых условий при проведении итераций по подобластям.

В целом реализация МДО осуществляется как двухуровневый итерационный процесс в подпространствах Крылова: верхний уровень – это аддитивный блочный метод Шварца–Якоби вида (34), который для простоты в стационарном варианте можно записать как

$$u^{n+1} = u^n + B_1^{-1}(f - Au^n), \quad u^n = \{u_q^n\}, \quad B_1 = \text{block} - \text{diag}\{B_{q,q}\}, \quad (37)$$

где B_1 есть соответствующий предобусловливатель, который фактически будет использоваться в крыловском процессе. На каждой n -й итерации (37) осуществляется синхронное, или параллельное, решение вспомогательных подзадач в подобластях Ω_q , которое формально заключается в обращении блоков предобусловливателя $B_{q,q}$. Это может быть сделано или прямыми, или итерационными методами, среди которых для больших алгебраических систем естественно выбрать предобусловленные алгоритмы в соответствующих подпространствах Крылова. Отметим, что в последнем случае мы приходим к задаче многократного решения СЛАУ с одинаковыми матрицами, но разными правыми частями, что можно сделать экономично путем переиспользования базисов крыловских подпространств, как это указывалось выше в замечании 1.

С точки зрения распараллеливания вычислений на МВС, предпочтительнее разбивать сеточную расчетную область на большее число подобластей Ω_q , $q = 1, 2, \dots, P$, чтобы все соответствующие подсистемы решать синхронно на P процессорах. Однако при этом будет очевидным образом расти количество внешних итераций, которые можно оценить величиной $O(P^{1/d})$, где d есть геометрическая размерность области (в предположении, что декомпозиция формирует d -мерную “макросеть” из подобластей). Число внешних итераций можно сократить за счет увеличения размеров пересечения контактирующих подобластей (over-lapping), но тогда реализация каждого шага будет становиться более трудоемкой (на практике пересечение сеточных областей выбирается размером в несколько шагов h).

Для ускорения итераций крыловского типа в МДО существуют различные подходы, которые могут быть интерпретированы как применение некоторых дополнительных предобусловливателей, в том числе сглаживающего типа, в силу чего такие методы иногда называются гибридными (см. [58], [59]). Рассмотрим в качестве примера метод грубосеточной коррекции, основанный на принципе дефляции для выбора начального приближения u^0 .

Пусть у нас имеется какой-то информационный базис из векторов v_1, \dots, v_m , составляющих матрицу $V = (v_1 \dots v_m) \in \mathcal{R}^{N,m}$, $m < N$. Выбирая какой-то произвольный вектор u^{-1} , будем искать начальное приближение и соответствующий вектор невязки в виде

$$u^0 = u^{-1} + Vc, \quad r^0 = r^{-1} - AVc, \quad r^{-1} = f - Au^{-1},$$

где вектор неизвестных коэффициентов $c = (c_1, \dots, c_m)^T$ определяется по условию минимума невязки r^0 из переопределенного уравнения

$$AVc = r^{-1}. \quad (38)$$

Умножая обе части (38) слева на $V^T A^\gamma$ с параметром $\gamma = 0$ или $\gamma = 1$, получаем семейство обобщенных решений

$$c = (V^T A^{\gamma+1} V)^+ V^T A^\gamma r^{-1},$$

где случай $\gamma = 1$ соответствует нормальному решению, т.е. минимизирующему евклидовы нормы невязки $r^0 = r^{-1} - AVc$ и самого решения c . Отсюда для откорректированной невязки получаем выражения

$$V^T A^\gamma r^0 = 0, \quad r^0 = Tr^{-1}, \quad T = I - AV(V^T A^{\gamma+1} V)^{-1} V^T A^\gamma, \quad (39)$$

которые соответствуют минимуму функционала

$$\Phi_\gamma(r^0) = (A^{\gamma-1} r^0, r^0).$$

Нетрудно видеть, что при этом векторы v_s играют роль направляющих векторов в подпространствах Крылова, т.е. они обеспечивают вектор невязки r^0 ортогональными и вариационными свойствами, аналогичными тем, которые имеются в методах сопряженных градиентов и сопряженных невязок при $\gamma = 0$ и $\gamma = 1$ соответственно.

Следуя этим соображениям, далее для методов сопряженных направлений определим через r^0 начальный направляющий вектор из соответствующих ортогональных условий:

$$V^T A^{\gamma+1} p^0 = 0, \quad p^0 = B_2^{-1} r^0, \quad B_2^{-1} = I - A^{\gamma+1} V (V^T A^{2\gamma+2} V)^{-1} V^T A^{\gamma+1}. \quad (40)$$

Введенная здесь матрица B_2 играет роль второго предобусловителя, в дополнение к B_1 из (37), и может вместе с ним использоваться на последующих итерациях по формулам мульти-предобусловленных сопряженных направлений (см. [17]). Отметим, что в (40) степени матрицы A подобраны таким образом, чтобы обеспечить симметричность B_2 .

Вопрос выбора базисных векторов v_1, \dots, v_m , эффективно дополняющих подпространство Крылова, актуален во многих ускоряющих итерационных процедурах. Так, алгоритм грубосеточной коррекции, активно используемый в МДО, основан в простейшем случае на “геометрической” декомпозиции сеточной области Ω_q , а его компоненты равны 1 или 0 в зависимости от того, принадлежит ли соответствующий узел q -й подобласти или нет. Дальнейшее развитие этого подхода заключается в переходе от кусочно-постоянных базисных функций v_q к кусочно-полиномиальным более высоких порядков (см. [53]).

3.2. Многосеточные методы

Изначально в пионерских работах Р.П. Федоренко и Н.С. Бахвалова многосеточные подходы базировались на спектрально-аппроксимационных принципах, с раздельным подавлением компонент ошибки, соответствующих низким и высоким частотам. Дальнейшее их развитие осуществлялось в геометрическом плане, в алгебраическом (наиболее популярное название — AMG, от Algebraic Multi-Grid), а также в комбинаторном (см. [5], [56], [60]–[67]).

Данные подходы занимают исключительное место в вычислительной алгебре, поскольку являются асимптотически оптимальными по порядку, т.е. требуют объема компьютерных ресурсов, пропорционального размерности СЛАУ. Наиболее общая алгебраическая методология базируется на универсальном принципе предобусловливания матриц. А реализация технологий с различным количеством вложенных сеток трактуется как рекурсивное применение двухсеточного алгоритма. Для простоты рассмотрим последовательность вложенных сеток $\Omega_m \subset \dots \subset \Omega_2 \subset \Omega_1$, т.е. узлы более редкой сетки являются узлами “ближайшей” густой сетки Ω_k . Будем предполагать, что сетка Ω_k получается из Ω_{k+1} путем “сгущения вдвое”, так что соответствующие количества узлов соотносятся как $N_k \approx 2^d N_{k+1}$, где d есть размерность области. Для решения исходной СЛАУ $Au = f$ на Ω , которую можно переобозначить как $A_1 u_1 = f_1$, строится неявно предобусловливатель B_1 с помощью приближенного решения системы со специально конструируемой матрицей A_2 для сетки Ω_2 и т.д., процесс “зацикливается”.

В современной трактовке достаточно общего вида, многосеточный метод AMG может быть представлен следующими этапами итерационного решения алгебраической системы $A_k u_k = f_k$ на Ω_k .

1. На “густой” сетке Ω_k по заданному приближению для вектора u_k^n вычисляется невязка

$$r_k^n = f_k - A_k u_k^n, \quad A_1 = A. \quad (41)$$

2. Для вектора r_k^n производится препроцессинг (предварительное сглаживание), как правило, путем проведения нескольких итераций с помощью какого-либо простого алгоритма. Более конкретно, данная процедура реализуется за два шага. На первом вычисляется направляющий вектор

$$\hat{S}_k p_k^n = r_k^n, \quad (42)$$

где \hat{S}_k — оператор, или матрица, этого этапа (presmoothing). Фактически \hat{S}_k есть некоторая аппроксимация матрицы A_k (формальное представление результатов нескольких циклов применяемого итерационного метода сглаживания). На втором шаге определяется очередное (сглаженное) приближенное решение и соответствующая невязка:

$$\tilde{u}_k^n = u_k^n + \tilde{p}_k^n, \quad \tilde{r}_k^n = f - A_k \tilde{u}_k^n = r_k^n - A_1 \tilde{p}_k^n. \quad (43)$$

3. По определенному на густой сетке Ω_k вектору \tilde{r}_k^n формируется вектор невязки \tilde{r}_{k+1}^n на редкой сетке Ω_{k+1} :

$$\tilde{r}_{k+1}^n = R_k \tilde{r}_k^n, \quad R_k \in \mathcal{R}^{N_k, N_{k+1}}, \quad \tilde{r}_k^n \in \mathcal{R}^{N_k}, \quad (44)$$

где R_k есть некоторый оператор сужения, или ограничения (этап restriction). В простейшем случае данная операция есть проектирование вектора с сетки Ω_k на Ω_{k+1} .

4. На редкой сетке вычисляется направляющий вектор \hat{p}_{k+1}^n из решения СЛАУ

$$A_{k+1} \hat{p}_{k+1}^n = \tilde{r}_{k+1}^n, \quad A_{k+1} \in \mathcal{R}^{N_{k+1}, N_{k+1}}, \quad \hat{p}_{k+1}^n, \tilde{r}_{k+1}^n \in \mathcal{R}^{N_{k+1}}, \quad (45)$$

где A_{k+1} — матрица СЛАУ для сетки Ω_{k+1} , в некотором смысле наследующая аппроксимационные и алгебраические свойства оператора A_k на сетке Ω_k . Существуют различные приемы для построения матрицы A_k , которая иногда называется матрицей грубосеточной коррекции.

5. Найденный из решения системы (45) вектор \hat{p}_{k+1}^n продолжается с редкой сетки Ω на густую сетку Ω_k (этап prolongation):

$$\check{p}_k^n = P_k \hat{p}_{k+1}^n, \quad P_k \in \mathcal{R}^{N_k, N_{k+1}}, \quad \check{p}_k^n \in \mathcal{R}^{N_k}. \quad (46)$$

В некотором смысле согласованным определением вводимых операторов является такое, которое удовлетворяет соотношению $A_k^{-1} \approx P_k A_{k+1}^{-1} R_k$. Если, например, матрица A_k симметрична, то для наследования этого свойства на редкой сетке Ω_{k+1} целесообразно выбирать $P_k = R_k^T$. В частности, при этом определяется так называемая Галёркинская аппроксимация $A_{k+1} = P_k^T A_k P_k$.

6. Для вектора \check{p}_k^n определяются соответствующие векторы приближенного решения и невязки на густой сетке (residual update)

$$\check{u}_k^n = \check{u}_k^n + \check{p}_k^n, \quad \check{r}_k^n = f_k - A_k \check{u}_k^n = r_k^n - A_k \check{p}_k^n, \quad \check{r}_k^n \in \mathcal{R}^{N_k}. \quad (47)$$

7. Проводится “пост-процессинг”, т.е. повторное сглаживание, для вновь полученных направляющего вектора и итерационного приближения решения на густой сетке Ω_k , с одновременным вычислением нового направляющего вектора \hat{p}_k^n из решения вспомогательной СЛАУ с матрицей \check{S}_k (оператор повторного сглаживания, этап postsmoothing):

$$\check{S}_k \hat{p}_k^n = \check{r}_k^n, \quad (48)$$

аналогично (42).

8. Итоговый направляющий вектор определяется как результат первой итерации AMG:

$$p_k^n = \hat{p}_k^n + \check{p}_k^n + \tilde{p}_k^n = B_k^{-1} r_k^n,$$

где B_k есть предобуславливающая матрица данной двухсеточной стадии метода, явный вид которой можно выразить через операторы сглаживания, ограничения, грубосеточной коррекции и продолжения, однако ее конкретное представление не суть важно, так как общая схема вычислений и ее реализация описываются формулами (41)–(48). Частных вариантов двухсеточного метода может быть достаточно много, и все они (при организации стационарного итерационного процесса) имеют вид

$$u_k^{n+1} = u_k^n + p_k^n = u_k^n + B_k^{-1} r_k^n, \quad (49)$$

представляющий собой двухуровневый итерационный процесс, поскольку операция умножения на B_k^{-1} включает решение СЛАУ с матрицей A_{k+1} , что при больших значениях N_{k+1} прямым алгоритмом делать нецелесообразно. Понятно, что на основе алгоритма (49) можно построить предобусловленный метод в подпространствах Крылова.

Но идея многосеточного подхода состоит в использовании рекурсивного принципа: для приближенного решения уравнений на редкой сетке Ω_{k+1} применяются приведенные выше приемы с использованием еще более редкой сетки Ω_{k+2} и т.д. При этом на каждой сетке Ω_k весь цикл вы-

числений по формулам (41)–(48) можно повторять заданное число γ раз, что может быть записано как обобщение соотношения (49):

$$u_k^{n+1} = u_k^n + B_k^{-\gamma} r_k^n. \quad (50)$$

На практике число таких повторов выбирается равным $\gamma = 1$ или $\gamma = 2$. В первом случае получающаяся вычислительная схема называется V -циклом, а во втором – W -циклом. В [64] и [65] предложено развитие такого подхода: K -цикл, в котором при $\gamma = 2$ используется не стационарный итерационный процесс, а крыловского типа (более конкретно, “гибкий” метод сопряженных градиентов FCG из [40]). В последней формуле надо иметь в виду, что предобусловливатель B_k выражается рекурсивно для $k = 1, 2, \dots, m-1$, а на последней m -й сетке СЛАУ вида (45) с матрицей A_{m+1} решается непосредственно (прямым или итерационным методом).

Конкретные реализации многосеточных подходов, ставших уже классическими, отличаются способами выбора матричных операторов, определяющих последовательные этапы приведенной вычислительной схемы.

3.3. Алгоритмы неполной факторизации

Наиболее классическими приемами предобусловливания СЛАУ являются многочисленные явные и неявные методы приближенной факторизации матриц, обширный анализ которых без аспектов распараллеливания приведен в [10]. Современные подходы к этим технологиям, включая перспективные аппроксимации матриц, обратных к треугольным множителям из неполных матричных разложений, приведены в [68]–[70] и цитируемой там литературе.

Типичная форма легко обрабатываемой предобусловленной матрицы заключается в приближенной факторизации вида

$$B = (G + L)G^{-1}(G + U) = G + L + U + LG^{-1}U. \quad (51)$$

Здесь L и U – нижняя и верхняя треугольные части исходной матрицы $A = D + L + U$, а D – некоторая блочно-диагональная или диагональная в частном случае матрица. Из общего требования $B \approx A$ блочно-диагональная матрица G строится с помощью соотношения

$$G = D - \overline{LG^{-1}U}, \quad (52)$$

где черта над матрицей обозначает ее какую-то ленточную аппроксимацию.

На основе формул (51), (52) конструируются различные методы симметричной последовательной верхней релаксации (при этом $G = 2\omega^{-1}D$), где ω есть итерационный параметр, а также явной и неявной неполной факторизации, содержательный анализ которых представлен в книгах [2], [5], [10], [68], а также в специальных обзорных статьях [51], [69], [70]. Широкое семейство алгоритмов основано на неполном LU-разложении исходной матрицы, которое в симметричном случае сводится к приближенному разложению Холеского. Качество аппроксимации матрицы и скорость сходимости повышаются за счет “уплотнения” матричных множителей, но при этом “дорожает” каждая итерация.

Схожий подход, но ориентированный на распараллеливание, заключается в конструировании разреженной аппроксимации обратной матрицы, устраняющей необходимость решения треугольных систем на каждой итерации. Описанные методы исследованы как в поточечном, так и в блочных вариантах. Оригинальные результаты в этих направлениях получены А.Ю. Еремичевым, И.Е. Капоричевым, Л.Ю. Колотилюной, И.Н. Коньшиным и другими авторами в работах [31], [32], [71], [72]. Другое перспективное направление основано на факторизации, в том числе приближенной, иерархических матриц в HSS-форматах с использованием малоранговых блочных структур в LU-разложении (см. [73], [74]).

Наконец, отметим еще перспективное семейство алгоритмов построения предобусловливателей, базирующееся на сетевом программировании или на методах преобразования графов (см. [75]–[77]). Существенным является и такой момент, что если традиционно неполное треугольное разложение матриц реализуется для невырожденных матриц, то сейчас имеются его модификации и для положительно-полуопределенных случаев.

Один из общих подходов к ускорению итераций заключается в принципе компенсации, или согласования строчных сумм, который заключается в подборе матрицы G в (51), (52) по выполнению условий

$$By^{(l)} = Ay^{(l)}, \quad l = 1, 2, \dots, m, \quad (53)$$

на некотором заданном наборе из m пробных векторов (см. [49], [78] и цитируемую там литературу). В некоторых работах такой прием также называется фильтрацией.

Для удовлетворения условия (53) матрица G преобразуется к виду

$$G = D - \overline{LG^{-1}U} - \theta S, \quad (54)$$

где $\theta \in [0, \theta \in [0, 1]]$, – некоторый компенсирующий параметр, а S – блочно-диагональная матрица, формируемая по условию

$$Sy^{(l)} = (LG^{-1}U - \overline{LG^{-1}U})y^{(l)}, \quad l = 1, 2, \dots, m. \quad (55)$$

Среди методов неполной факторизации для решения сеточных СЛАУ наиболее быстрыми являются неявные, которые базируются на алгоритмах прогонки решения вспомогательных трехдиагональных систем. Такие алгоритмы эффективно распараллеливаются на МВС с помощью многопоточковых реализаций рекурсивных прогоночных процедур, основанных или на методах редукции исходных СЛАУ, или на методах окаймления. Отметим, что в [57] предложен двухпоточковый блочный вариант попеременно-треугольной факторизации исходной матрицы, когда каждый из множителей не является нижней или верхней треугольной матрицей, а состоит из блочных строк различной ориентации: одни являются нижними треугольными, а остальные – верхними треугольными (такое разложение названо авторами “twisted decomposition”, а в русскоязычной литературе данный подход традиционно определяется как алгоритм встречных прогонок, см. обзор в [27]).

Если матрица A симметричная, т.е. $D = D^T$ и $L = U^T$, то естественно выбирать матрицы G и B также симметричными, а для исходной СЛАУ – использовать двустороннее предобуславливание с сохранением симметрии итоговой системы:

$$\begin{aligned} \bar{A}\bar{u} &= \bar{f}, \quad \bar{A} = L_B^{-1}AU_B^{-1}, \quad \bar{u} = U_B\bar{u}, \quad \bar{f} = L_B^{-1}f, \\ B &= L_B U_B, \quad L_B = (G + L)U_G^{-1}, \quad G = L_G U_G, \quad U_G = L_G^T. \end{aligned} \quad (56)$$

Предобусловленная матрица \bar{A} после несложных преобразований может быть приведена к виду

$$\begin{aligned} \bar{A} &= (I + \bar{L})^{-1} + (I + \bar{U})^{-1} + (I + \bar{L})^{-1}(\bar{D} - 2I)(I + \bar{U})^{-1}, \\ \bar{D} &= L_G^{-1}DU_G^{-1}, \quad \bar{L} = L_G^{-1}LU_G^{-1}, \quad \bar{U} = L_G^{-1}UU_G^{-1}, \end{aligned} \quad (57)$$

которое допускает реализацию векторно-матричного умножения в виде

$$\bar{A}v = (I + \bar{L})^{-1}[v + (\bar{D} - 2I)w] + w, \quad w = (I + \bar{U})^{-1}v, \quad (58)$$

во многих случаях дающем существенную экономию вычислений. Отметим, что матричные представления (56)–(58) справедливы и без требований $L_B = U_B^T$, $L_G = U_G^T$, т.е. данные формулы приемлемы и для несимметричных СЛАУ.

Непосредственное обращение треугольных матриц, которое присутствует в описанных выше алгоритмах, плохо распараллеливается. Существуют специальные технологические приемы, позволяющие в определенной степени этот недостаток устранить (см., например, [79] и обзор литературы в [80]). Мы рассмотрим другой подход, использующий вместо L , U переменнотреугольные матрицы \hat{L} , \hat{U} , имеющие на части строк ненулевые элементы только слева от главной

диагонали, а на остальных – только справа. Построение таких преобусловливателей мы проиллюстрируем на примере блочно-трехдиагональной матрицы

$$Au = \begin{bmatrix} D_1 & U_1 & & 0 \\ L_2 & D_2 & U_2 & \\ & \ddots & \ddots & \ddots \\ & & & U_{N_x-1} \\ 0 & & L_{N_x} & D_{N_x} \end{bmatrix}. \tag{59}$$

Переменно-треугольные матрицы в данном случае определим следующим образом:

$$\hat{L} = \begin{bmatrix} 0 & 0 & & & & & \\ L_2 & 0 & 0 & & & & \\ & L_3 & 0 & 0 & & & \\ & & L_4 & 0 & U_4 & & \\ & & & 0 & 0 & U_5 & \\ & & & & 0 & 0 & U_6 \\ 0 & & & & & 0 & 0 \end{bmatrix}, \quad \hat{U} = \begin{bmatrix} 0 & U_1 & & & & & \\ 0 & 0 & U_2 & & & & 0 \\ & 0 & 0 & U_3 & & & \\ & & 0 & 0 & 0 & & \\ & & & L_5 & 0 & 0 & \\ & & & & 0 & L_6 & 0 & 0 \\ 0 & & & & & & L_7 & 0 \end{bmatrix}. \tag{60}$$

Поскольку в данном случае $A = D + \hat{L} + \hat{U}$, формулы неполной факторизации (51)–(58) для определения преобусловливателя B остаются в силе. Надо только заменить L и U на \hat{L} и \hat{U} соответственно. При этом трехдиагональные блоки матрицы $\bar{G} = \{\bar{G}_i\}$ находятся по формулам, аналогичным (28), (29), но только с помощью встречных рекурсий (блочных прогонок, см. [25]), которые мы выпишем для произвольного нечетного $N_x = 2m + 1$:

$$\begin{aligned} G_1 &= D, \quad G_i = D_i - L_i G_{i-1}^{-1} U_{i-1} - \theta S_i, \quad i = 2, \dots, m, \\ G_{N_x} &= D_{N_x}, \quad G_i = D_i - U_i G_{i+1}^{-1} r U_{i+1} - \theta S_i, \quad i = N_x - 1, \dots, m + 2, \\ S_i y^{(l)} &= L_i (G_{i-1}^{-1} - G_{i-1}^{-1}) U_{i-1} y^{(l)}, \quad l = 1, \dots, l_i, \\ G_{m+1} &= D_{m+1} - L_{m+1} G_m^{-1} U_m - U_{m+1} G_{m+2}^{-1} L_{m+2} - \theta S_{m+1}, \\ S_{m+1} y^{(l)} &= [L_{m+1} (G_m^{-1} - G_m^{-1}) U_m - U_{m+1} (G_{m+2}^{-1} - G_{m+2}^{-1})] y^{(l)}, \end{aligned} \tag{61}$$

где S_l суть диагональные матрицы для $l = 1$ и трехдиагональные при $l = 2$.

При использовании матрицы \bar{B} в качестве преобусловливателя для какого-либо итерационного процесса на каждом шаге требуется решать вспомогательную систему вида

$$Bp \equiv (G + \hat{L})G^{-1}(G + \hat{U})p = r.$$

Ее решение можно найти из последовательных соотношений

$$(G + \hat{L})v = r, \quad Gw = \hat{U}p, \quad p = v - w,$$

которые реализуются с помощью встречных матричных прогонок по следующим формулам:

$$\begin{aligned} v_1 &= G_1^{-1} r_1, \quad i = 2, \dots, m : v_i = G_i^{-1} (r_i - \hat{L}_i v_{i-1}), \\ v_{N_x} &= G_{N_x}^{-1} r_{N_x}, \quad i = N_x - 1, \dots, m + 2 : v_i = G_i^{-1} (r_i - \hat{U}_i v_{i+1}), \\ v_{m+1} &= G_{m+1}^{-1} (r_{m+1} - \hat{L}_{m+1} v_m - \hat{U}_{m+1} v_{m+2}) = p_{m+1}, \\ i = m, \dots, 1 : w_i &= G_i^{-1} \hat{U}_i p_{i+1}, \quad p_i = v_i - w_i, \\ i = m + 2, \dots, N_x : w_i &= G_i^{-1} \hat{L}_i p_{i-1}, \quad p_i = v_i - w_i. \end{aligned} \tag{62}$$

В рекурсиях (61) и (62) этапы вычислений с увеличением индекса i и с его уменьшением можно назвать прямыми и обратными прогонами соответственно. Очевидно, что их реализации могут выполняться параллельно на двух потоках. Данный ограниченный параллелизм основан на

односменной триангуляции вида (60) для матриц L, U . Данный подход можно обобщить на P – сменную триангуляцию, т.е. определить матрицы \hat{L}, \hat{U} с условием $\hat{L} + \hat{U} = L + U$ как состоящие из P блочных строк, каждая из которых последовательно меняет тип триангуляции. Формулы блочных прогонок вида (61), (62) принимают более сложный вид, но могут распараллеливаться на P процессорах.

Матрица в (59) соответствует, например, пятиточечным уравнениям Лапласа на прямоугольной сетке с числом узлов $N = N_x \cdot N_y$. В этом случае D_i – трехдиагональные, а L_i и U_i – диагональные блоки. При переходе к трехмерным задачам на параллелепипедаидальной сетке с количеством узлов $N = N_x \cdot N_y \cdot N_z$ матрица A может быть также представлена в блочно-трехдиагональном виде (59), но каждый диагональный блок D_k будет теперь иметь размерность $N = N_x \cdot N_y$ и представлять собой матрицу, структурно совпадающую с (59).

Развитие рассмотренных выше подходов для решения СЛАУ на трехмерных сетках получило название методов вложенных факторизаций (nested factorization), предложенных в 1983 г. Дж. Апплеярдом и И. Чешире, а впоследствии исследованных многими авторами (см. [57], [82], [83] и цитируемую там литературу). Пусть матрица A представлена в виде

$$A = D + L_1 + U_1 + L_2 + U_2 + L_3 + U_3, \quad (63)$$

где D – диагональная, а L_k и $U_k, k = 1, 2, 3$, – нижние и верхние треугольные матрицы. Определим предобуславливающую матрицу B путем рекурсивной факторизации вида (51):

$$\begin{aligned} B &= (P + L_3)P^{-1}(P + U_3) = P + L_3 + U_3 + L_3P^{-1}U_3, \\ P &= (T + L_2)T^{-1}(T + U_2) = T + L_2 + U_2 + L_2T^{-1}U_2, \\ T &= (M + L_1)M^{-1}(M + U_1) = M + L_1 + U_1 + L_1M^{-1}U_1, \end{aligned} \quad (64)$$

в результате чего получаем

$$B = M + A - D + L_1M^{-1}U_1 + L_2T^{-1}U_2 + L_3P^{-1}U_3. \quad (65)$$

В зависимости от способа определения матрицы M могут быть сформированы различные предобуславливатели B . Мы остановимся на простом случае, когда матрица (63) соответствует семиточечной аппроксимации задачи Дирихле для уравнения Пуассона в параллелепипеде на параллелепипедной сетке. Тогда при естественной упорядоченности узлов матрицы M, T и P можно сформировать диагональную, трехдиагональную и пятидиагональную соответственно, а предобуславливатель можно определить по формулам

$$\begin{aligned} M &= D - L_1M^{-1}U_1 - \theta_2S_2 - \theta_3S_3, \\ B &= A + L_2T^{-1}U_2 + L_3P^{-1}U_3 - \theta_2S_2 - \theta_3S_3. \end{aligned} \quad (66)$$

Здесь θ_2 и θ_3 – итерационные (релаксирующие) параметры, а S_2 и S_3 – диагональные матрицы, определяемые из условий согласования строчных сумм, аналогично рассмотренному выше:

$$S_2e = L_2T^{-1}U_2e, \quad S_3e = L_3P^{-1}U_3e. \quad (67)$$

Отметим, что вместо (67) можно использовать согласование не строчных, а столбцовых сумм (см. [83] и приведенную там литературу):

$$e^T S_2 = e^T L_2 T^{-1} U_2, \quad e^T S_3 = e^T L_3 P^{-1} U_3. \quad (68)$$

Рассмотренный трехуровневый метод вложенных факторизаций можно структурно упростить, сведя его к двухуровневому. Для этого перепишем исходную матрицу A из (63) в виде

$$A = D_3 + L_3 + U_3, \quad D_3 = D_2 + L_2 + U_2, \quad D_2 = D + L_1 + U_1. \quad (69)$$

В данном случае D_3 есть блочно-диагональная матрица с пятидиагональными блоками $D_{3,i}$ размерности $N_y N_z$, каждый из которых соответствует плоской задаче в сечении $x = \text{const}$ и имеет структуру матрицы A в (59). Тогда матрицу P в (64) определяем формулой

$$P = \{G_i = D_{3,i} - \theta S_i\}, \quad S_i e = L_{3,i} G_{i-1}^{-1} U_{i-1} e,$$

что соответствует определению $L_3 P^{-1} U_3 = 0$ в (64). Отметим, что L_3 и U_3 можно определить как переменнотреугольные, и тогда реализацию алгоритма можно распараллеливать с помощью встречных блочных прогонок.

3.4. Методы решения седловых СЛАУ

Рассмотрим СЛАУ с матрицей седлового типа

$$A \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}, \quad A = \begin{bmatrix} D & C^T \\ C & 0 \end{bmatrix}, \quad (70)$$

где $A \in \mathcal{R}^{N,N}$, $D \in \mathcal{R}^{N_1, N_1}$, $C \in \mathcal{R}^{N_2, N_1}$, $N = N_1 + N_2$ и $f \in \mathcal{R}^{N_1}$ и $p, g \in \mathcal{R}^{N_2}$. Матрица D предполагается симметричной и положительно-полуопределенной. При этом необходимым и достаточным условием невырожденности A является $\ker(D) \cap \ker(C) = \{0\}$, а достаточным условием является положительная определенность матрицы D на $\ker(C)$ и $\ker(C^T) = \{0\}$, что мы будем предполагать выполненным в дальнейшем.

Задачи с седловой точкой представляют собой широко распространенную форму математических постановок в проблемах моделирования, включая начально-краевые смешанные формулировки для дифференциальных классических или обобщенных уравнений, оптимизационные задачи и вычислительную алгебру (см. [84]–[87] и цитируемую там литературу). Мы делаем акцент на методах решения седловых СЛАУ с большими разреженными матрицами, возникающими при сеточных аппроксимациях многомерных краевых задач, которые появляются во многих актуальных приложениях: электромагнетизме, гидро-газодинамике, упруго-пластичности, многофазной фильтрации в пористых средах, в задачах оптимизации и т.д.

В силу особенностей блочной структуры седловых алгебраических систем, методам их решения посвящено значительное количество работ: обзоры Дж. Голуба и соавт. [84], Ф. Бреucci [85], П. Василевского [72] и И. Нотей [86], монография Ю.В. Быченкова и Е.В. Чижонкова [87], цикл статей Ч. Грейфа и соавт. (в том числе для несимметричных СЛАУ седлового типа, см. [68], [85]) и М. Ариоли с соавт. [25], [75] (см. также обзоры в [90]–[92]).

Отметим, что без потери общности мы можем рассматривать седловую СЛАУ в виде

$$\begin{bmatrix} D & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}. \quad (71)$$

Действительно, если мы возьмем какое-либо частное решение системы $C\hat{u} = g$, то вектор $u = \tilde{u} + \hat{u}$, являющийся решением СЛАУ (70), будет удовлетворять системе

$$\begin{bmatrix} D & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} \tilde{u} \\ p \end{bmatrix} = \begin{bmatrix} f - D\hat{u} \\ 0 \end{bmatrix}.$$

Заметим также, что любые решения СЛАУ (71) удовлетворяют одновременно системе

$$\tilde{A}v = \tilde{A} \begin{bmatrix} u \\ p \end{bmatrix} \equiv \begin{bmatrix} \tilde{D} & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix} \equiv \tilde{f}, \quad \tilde{D} = D + R, \quad R = C^T K_0^{-1} C, \quad (72)$$

где $v, \tilde{f} \in \mathcal{R}^N$, а $K_0 \in \mathcal{R}^{N_1, N_1}$ – произвольная невырожденная матрица. Поскольку последняя система формально является регуляризацией, или обобщением, СЛАУ (71), мы в дальнейшем останавливаемся на алгоритмах решения именно уравнения (72).

С использованием дополнения Шура

$$S = -C\tilde{D}^{-1}C^T$$

матрица системы (72) факторизуется в виде

$$\tilde{A} = \begin{bmatrix} \tilde{D} & 0 \\ C & S \end{bmatrix} \begin{bmatrix} I & \tilde{D}^{-1}C^T \\ 0 & I \end{bmatrix}.$$

Если матрицы \bar{D} , S заменить их приближениями (предобусловливателями) B_d и B_s , то получаем предобусловливатель матрицы B в виде неполной блочной факторизации

$$B_1 = \begin{bmatrix} B_d & 0 \\ C & -B_s \end{bmatrix} \begin{bmatrix} I & B_d^{-1}C^T \\ 0 & I \end{bmatrix}. \quad (73)$$

Несколько более грубая аппроксимация, при использовании в (73) только одного (левого или правого) множителя приводит к неполному блочно-треугольному предобусловливанию с матрицей

$$B_2 = \begin{bmatrix} B_d & C^T \\ 0 & -B_s \end{bmatrix} \quad (74)$$

или к неполному предобусловливателю Узавы

$$B_3 = \begin{bmatrix} B_d & 0 \\ C & -B_s \end{bmatrix}. \quad (75)$$

Реализация каждого шага соответствующих итерационных процессов может быть представлена несколькими стадиями, на которых пересчитывается только одна блочная компонента искомого решения (поэтому эти методы называются иногда сегрегационными). В несколько обобщенной форме такой стационарный алгоритм (без крыловского ускорения) представим следующими тремя стадиями (см. [83], [85], [86]):

$$\begin{aligned} \hat{u}_d^{n+1} &= u_d^n + Q_d^{(1)}(f_d - \bar{D}u_d^n - C^T u_s^n), \\ u_s^{n+1} &= u_s^n - B_s^{-1}(f_s - C\hat{u}_d^{n+1}), \\ u_d^{n+1} &= \hat{u}_d^{n+1} + Q_d^{(2)}(f_d - \bar{D}A\hat{u}_d^{n+1} - C^T u_s^{n+1}). \end{aligned} \quad (76)$$

Здесь $Q_d^{(1)}$, $Q_d^{(2)}$ – некоторые аппроксимации матрицы, обратной или обобщенно обратной к предобусловливателю B_d . В частности, если $Q_d^{(1)} = B_d^{-1}$, $Q_d^{(2)} = 0$ или $Q_d^{(1)} = 0$, $Q_d^{(2)} = B_d^{-1}$, то из (76) мы получаем или алгоритм Узавы с предобусловливателем B_3 из (75) (при этом третья стадия опускается, т.е. $u_d^{n+1} = \hat{u}_d^{n+1}$), или неполное блочно-треугольное предобусловливание с матрицей B_2 из (74) (при этом первая стадия в (76) опускается и $u_d^{n+1} = u_d^n$).

Если матрица $Q_d = Q_d^{(1)} + Q_d^{(2)} - Q_d^{(2)}\bar{D}Q_d^{(1)}$ является невырожденной, то итерационному процессу (76) соответствует предобусловливатель

$$B_4 = \begin{pmatrix} I & 0 \\ CQ_d^{(1)} & I \end{pmatrix} \begin{pmatrix} Q_d^{-1} & 0 \\ 0 & -B_s \end{pmatrix} \begin{pmatrix} I & Q_d^{(2)}C^T \\ 0 & I \end{pmatrix}. \quad (77)$$

В этом случае для $Q_d^{(1)} = Q_d^{(2)} = B_d^{-1}$ из (77) следует так называемый симметризованный метод Узавы с предобусловливающей матрицей

$$B_5 = \begin{pmatrix} I & 0 \\ CB_d^{-1} & I \end{pmatrix} \begin{pmatrix} B_d(2B_d - \bar{D})^{-1}B_d & 0 \\ 0 & B_s \end{pmatrix} \begin{pmatrix} I & B_d^{-1}C^T \\ 0 & I \end{pmatrix}.$$

Одним из перспективных блочно-диагональных предобусловливателей для решения СЛАУ (72) является следующий:

$$B_6 = \begin{bmatrix} \tilde{D} + C^T K_1^{-1} C & 0 \\ 0 & K_2 \end{bmatrix}, \quad (78)$$

где K_1 и K_2 – некоторые симметричные невырожденные матрицы.

Собственные числа и векторы предобусловленной матрицы $\bar{A} = B_6^{-1}\tilde{A}$ из “возмущенной” СЛАУ (72) определяются спектральной задачей

$$\lambda B_6 z = \tilde{A} z, \quad z = (z_1^T, z_2^T)^T, \quad z_k \in \mathcal{R}^{N_k}, \quad k = 1, 2,$$

имеющей в компонентном виде следующее представление:

$$\begin{aligned}(D + C^T K_0 C)z_1 + C^T z_2 &= \lambda(\tilde{D} + C^T K_1^{-1} C)z_1, \\ Cz_1 &= \lambda K_2 z_2.\end{aligned}\tag{79}$$

Анализ данной спектральной задачи позволяет получать интересные результаты для различных частных случаев. В [81], [82] показано, в частности, что уникальные спектральные свойства матрицы \tilde{A} получаются при сильной вырожденности блока \tilde{D} , актуальной для алгоритмов решения систем уравнений Максвелла. Например, справедливы следующие результаты.

Утверждение 1. Пусть B_6 есть СПО-матрица, а $\{z_i, i = 1, 2, \dots, N_1 - N_2\}$ – базис ядра матрицы C . Тогда предобусловленная матрица $B_6^{-1} \tilde{A}$ имеет $N_1 - N_2$ линейно независимых векторов $(z_i^T, 0) \in \mathbb{R}^N$, которым соответствуют $N_1 - N_2$ кратных собственных значений $\lambda = 1$.

Утверждение 2. Пусть $K_1 = K_2 = K$, а \tilde{D} – симметричная положительно-полуопределенная матрица с размерностью ядра, равной r . Тогда $\lambda = 1$ есть собственное значение матрицы $B_6^{-1} \tilde{A}$ с кратностью N_1 , а $\lambda = -1$ – собственное значение кратности r . Остальные $N_2 - r$ собственных значений принадлежат интервалу $\lambda \in (-1, 0)$ и удовлетворяют соотношению

$$\lambda = -\mu/(\mu + 1),$$

где μ суть $N_2 - r$ положительных обобщенных собственных значений

$$\mu \tilde{D} z = C^T K C z.$$

Пусть $\{z_i, i = 1, 2, \dots, N_1 - N_2\}$ есть базис ядра C , $\{x_i, i = 1, 2, \dots, N_2 - 2\}$ – базис ядра \tilde{D} , а $\{y_i, i = 1, 2, \dots, N_2 - r\}$ – совокупность линейно независимых векторов, дополняющих $\ker(\tilde{D}) \cup \ker(C)$ в базисе \mathbb{R}^{N_1} . Тогда $N_1 - N_2$ векторов $(z_i, 0)$, r векторов $(x_i, K^{-1} C x_i)$ и $N_2 - r$ векторов $(y_i, K^{-1} y_i)$ суть линейно независимые собственные векторы, соответствующие собственным значениям $\lambda = 1$, а r векторов $(x_i, K^{-1} x_i)$ – собственные векторы, соответствующие $\lambda = -1$.

В целом наличие в предобусловливателе B_6 различных матриц K_0, K_1, K_2 предоставляет широкие возможности для конструирования эффективных алгоритмов в конкретных случаях.

Мы рассмотрим далее семейство итерационных методов для решения седловых симметричных СЛАУ с матрицей \tilde{A} из (72), основанное на эффективном подходе $G - K$ -бидиагонализации Голуба–Кахана, которая изначально была предложена для сингулярного разложения прямоугольных матриц, но затем в работах М. Саундерса, М. Ариоли, Ч. Грейфа и других авторов успешно применялась для решения алгебраических систем, в том числе с учетом блочной седловой структуры.

Без ограничения общности исследуемую СЛАУ запишем в виде

$$\tilde{A} \begin{bmatrix} u \\ p \end{bmatrix} \equiv \begin{bmatrix} \tilde{D} & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} 0 \\ g \end{bmatrix}, \quad \tilde{D} = D + C^T K^{-1} C.\tag{80}$$

Легко проверить, что если в (72) функцию u заменить на $u + \tilde{D}^{-1} f$, то эта система примет форму (80) с правой частью $g = -C \tilde{D}^{-1} f$. Предполагается, что в (80) \tilde{D} и K являются СПО-матрицами, а также выполняется неравенство $N_1 \geq N_2$.

Метод $G - K$ -бидиагонализации базируется на построении \tilde{D} -ортогональных векторов v_k и K -ортогональных векторов q_k , которые удовлетворяют условиям

$$\begin{aligned}C^T Q &= \tilde{D} V [B^T 0]^T, & V^T \tilde{D} V &= I_{N_1}, \\ CV &= K Q [B^T 0], & Q^T K Q &= I_{N_2},\end{aligned}\tag{81}$$

где $V = [v_1, \dots, v_{N_1}] \in \mathcal{R}^{N_1, N_2}$, $Q = [q_1, \dots, q_{N_1}]$, а $B \in \mathcal{R}^{N_2, N_2}$ есть двухдиагональная матрица

$$B = \begin{bmatrix} \alpha & \beta_1 & 0 & \dots & 0 \\ 0 & \alpha_2 & \beta_3 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \alpha_{N_2-1} & \beta_{N_2-1} \\ 0 & \ddots & 0 & 0 & \alpha_{N_2} \end{bmatrix}.$$

Вводя новые неизвестные функции

$$u = Vz, \quad p = Qg \quad (82)$$

и умножая систему (80) на блочно-диагональную матрицу $\text{block-diag}(V^T, Q^T)$, получаем

$$\begin{bmatrix} I_{N_2} & 0 & B \\ 0 & I_{N_1-N_2} & 0 \\ B^T & 0 & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ Q^T g \end{bmatrix}, \quad z_1 \in \mathcal{R}^{N_2}, \quad z_2 \in \mathcal{R}^{N_1-N_2}. \quad (83)$$

Из (83) следует, что вектор u зависит от N_2 столбцов матрицы V , поскольку $z_2 = 0$. Таким образом, СЛАУ (83) редуцируется к виду

$$\begin{bmatrix} I_{N_2} & B \\ B^T & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ Q^T g \end{bmatrix}. \quad (84)$$

Отсюда, полагая

$$q_1 = K^{-1}g / \|g\|_{K^{-1}}, \quad \alpha_1 \tilde{D}^{-1}v_1 = C^T q_1, \quad \|g\|_{K^{-1}} = (g, K^{-1}g)^{1/2},$$

находим начальный вектор v_1 :

$$v_1 = w / \alpha_1, \quad \alpha_1 \sqrt{w^T C^T q_1}, \quad w = \tilde{D} C^T q_1. \quad (85)$$

Дальнейшие векторы v_n , q_i и элементы α_i , β_i матрицы B вычисляются из рекуррентных соотношений, $n = 1, 2, \dots$:

$$\begin{aligned} s &= K^{-1}(Cv_n - \alpha_i Kq_n), \quad \beta_{n-1} = \sqrt{s^T Ks}, \\ q_{n+1} &= s / \beta_{n+1}, \quad w = \tilde{D}^{-1}(C^T q_{n+1} - \beta_{n+1} \tilde{D}v_n), \\ \alpha_{n+1} &= (w^T \tilde{D}w)^{1/2}, \quad v_{n+1} = w / \alpha_{n+1}. \end{aligned} \quad (86)$$

Последовательные приближения u^n в соответствии с (82) определяются по первым n столбцам матрицы V , т.е.

$$u^n = V_n z^n = \sum_{j=1}^n z_1^j v_j, \quad z^n = (z_1^1, \dots, z_1^n), \quad (87)$$

где z_1^j — компоненты вектора z_1 из (84), вычисляемые по формулам

$$z_1^n = \|g\|_{N^{-1}} / \alpha_1, \quad z_{j+1}^n = -\beta_{j+1} z_j^n / \alpha_{j+1}, \quad j = 1, 2, \dots, N_2. \quad (88)$$

Опуская детали вывода формул (см. подробнее [87]), мы приведем итоговые рекуррентные соотношения для итерационного решения:

$$\begin{aligned} u^{n+1} &= u^n + z_1^{n+1} v_{n+1}, \quad n = 1, 2, \dots, N_2, \\ p^{n+1} &= p^n - z_1^{n+1} d_{n+1}, \quad d_{n+1} = (q_{n+1} - \beta_{n+1} \alpha_n) / \alpha_{n+1}, \end{aligned} \quad (89)$$

где d_n есть n -й столбец $D = QB^{-1}$. Отметим, что описанный алгоритм обладает следующими свойствами оптимизации: на каждом n -м шаге ошибка итерационного приближения достигает своего минимума:

$$\min_{\substack{u_n \in \mathcal{U}_n \\ Cu^n - g \perp \mathcal{Q}_n}} \|u - u^n\|_{\tilde{D}}, \quad (90)$$

$$\mathcal{U}_n = \text{Span}\{v_1, \dots, v_n\}, \quad \mathcal{Q}_n = \text{Span}\{q_1, \dots, q_n\}.$$

Как отмечается в [71], данный метод обладает высокой скоростью сходимости при решении СЛАУ, возникающих при конечно-элементных аппроксимациях непрерывных многомерных краевых задач седлового типа, т.е. в смешанных постановках. При этом существенным фактором является то, что на каждой итерации требуется решать возникающие алгебраические подсистемы с матрицами \tilde{D} и K , которые в некотором смысле играют роль предобусловливателей. Их приближенное обращение приводит фактически к двухуровневым итерационным процессам в определенных подпространствах.

4. ВОПРОСЫ РАСПАРАЛЛЕЛИВАНИЯ И ПРОИЗВОДИТЕЛЬНОСТИ ИТЕРАЦИОННЫХ МЕТОДОВ

Современное понимание качества алгоритма включает две главных характеристики: математическая эффективность и производительность его реализации на конкретной архитектуре суперкомпьютера. Первая сторона включает конструирование и оптимизацию итерационных методов с высокой скоростью сходимости, а также теоретические оценки вычислительной ресурсоемкости (необходимые объемы арифметических операций и памяти). Второй аспект является чисто практическим и характеризуется реальным временем выполнения алгоритма для определенного класса задач, что зависит от масштабируемости его распараллеливания, а также от технологий программирования на конкретной компьютерной платформе.

Наиболее интересующие нас СЛАУ имеют высокие порядки ($10^8 - 10^{11}$) и разреженные матрицы с большими числами обусловленности ($10^{12} - 10^{15}$) и нерегулярной структурой. Это не только приводит к большому числу итераций, но и вынуждает работать с системами распределенной и иерархической общей памяти, а также существенно замедляет доступ к данным.

Основная количественная характеристика распараллеливания — это ускорение вычислений

$$S_p = T_1/T_p, \quad T_p = T_p^a + T_p^c,$$

где T_p — время решения задачи на p процессорах, складывающееся из времени информационных обменов и выполнения арифметических операций, которые описываются приближенными формулами

$$T^a = \tau_a N_a, \quad T^c = N_t(\tau_0 + \tau_c N_c).$$

Здесь τ_a и N_a — среднее время выполнения одного арифметического действия и их общее количество, N_t — число обменов, τ_0 и τ_c — время ожидания и длительность передачи одного числа, а N_c — средний объем одной коммуникации.

Поскольку для машинных констант справедливы неравенства $\tau_0 \gg \tau_c \gg \tau_0$, то для конструируемых алгоритмов следуют очевидные рекомендации: надо стараться минимизировать объем коммуникаций, а сами обмены осуществлять не маленькими, а большими порциями, т.е. по возможности осуществлять предварительное накопление буферов данных. Эти выводы тем более справедливы, что межпроцессорные передачи информации не только замедляют вычислительный процесс, но и являются наиболее энергозатратными операциями, и это становится существенным фактором в расходах на эксплуатацию суперкомпьютера (см. [93]).

Стратегия распараллеливания больших сеточных СЛАУ, возникающих из аппроксимации многомерных краевых задач, базируется на средствах гибридного программирования и аддитивных методах декомпозиции областей с двухуровневыми итерациями в подпространствах Крылова. Итерации верхнего уровня (по подобластям) осуществляются с помощью системы передачи сообщений MPI между процессами, каждый из которых выполняет (одновременно) решение алгебраической подсистемы в соответствующей подобласти. На каждой такой итерации происходят обмены значениями приближенных решений на интерфейсных граничных поверхностях

контактирующих подобластей. Естественно, все матричные и векторные данные для подсистем предварительно формируются в распределенном по процессам виде. Решение СЛАУ в каждой из подобластей распараллеливается с помощью многопоточковых вычислений (системы типа OpenMP) на многоядерных процессорах с общей памятью. Дополнительное ускорение здесь может быть достигнуто путем векторизации операций (системы команд типа AVX, основанная на технологиях SIMD – Single Instruction Multi Data) (см. обзоры в [18]). К сожалению, здесь можно констатировать пока отсутствие штатных систем программирования с автоматическим распараллеливанием алгоритмов, так что успехи в масштабируемости ускорения расчетов в значительной степени зависят от квалификации и искусства математика-вычислителя.

Отметим, что традиционные блочные методы Якоби–Шварца, лежащие в основе параллельных алгоритмов декомпозиции областей, базируются на спектральной оптимизации итерационных процессов. В определенном смысле альтернативой им могут стать проекционные блочные методы Чиммино (см. [9], [94]), пока что недостаточно исследованные и мало используемые на практике.

С точки зрения распараллеливания перспективными являются неявные методы переменных направлений, или продольно-поперечных прогонок (см. [90] и обзор в [10]), активно развивавшиеся в 1960–1970-е годы и имеющие рекордную скорость сходимости итераций для модельных сеточных краевых задач. В последние десятилетия это направление обрело второе дыхание в связи с развитием рациональных аппроксимаций в подпространствах Крылова (см. [95]), в том числе при решении актуальных матричных уравнений Ляпунова и Сильвестра. Отметим, что в [93] предложен метод к повышению степени распараллеливания на основе разложения рациональных функций в простые дроби.

Существенное ускорение вычислений может быть получено за счет использования переменной точности машинной арифметики. Традиционным при решении больших СЛАУ является применение стандартной двойной точности с длиной представления вещественного числа с плавающей запятой в 64 бита, однако для экстремально плохо обусловленных матриц необходим переход к четверной точности (128 бит). Наоборот, на отдельных стадиях алгоритма допустимо применение простой и даже половинной точности (32 и 16 бит соответственно), выполняемых гораздо быстрее. Такой подход естественен на первых шагах итерационного процесса, когда ошибка приближенного решения еще относительно велика. Другая возможность использования пониженной точности существует в DDM при решении вспомогательных СЛАУ в подобластях. Необходимо иметь в виду, что такие приемы требуют тщательного анализа реализуемых численных погрешностей метода для обеспечения устойчивых расчетов в целом.

Дальнейшим резервом повышения производительности является оптимизация кода, которая может быть достигнута с помощью использования качественного вычислительного инструментария (например, SPARSE BLAS), применения различных опций компилятора и специальных свойств используемой суперкомпьютерной платформы. Отметим также, что за последние десятилетия в мире накоплено большое количество качественных программных библиотек по задачам вычислительной алгебры (PETSc, HYPRE, MKL INTEL и др., см. обзор в [22], [23]). Еще можно назвать серьезные разработки во ВНИЭФ РФЯЦ (г. Саров, см. [96]), в проектах по интегрированным вычислительным окружениям DUNE (см. [97]), INMOST (см. [98]) и BSM (см. [1], [24]).

Создание, сопровождение и развитие математического и программного обеспечения по эффективному решению СЛАУ – это непрерывный наукоемкий технологический процесс, включающий регулярные экспериментальные исследования, требующие своих интеллектуальных средств поддержки: системы автоматизации верификации и тестирования алгоритмов и их компьютерных реализаций, каталогизированные коллекции методических и практических задач, примером которых являются популярные Sparse Matrix Collections от различных разработчиков, а также многообразные демонстрационные и обучающие версии программно-информационных продуктов (помимо традиционных монографий, учебников и пользовательской документации с рекомендациями по применению тех или иных алгоритмов к решению конкретных задач).

5. ЗАКЛЮЧЕНИЕ. ПРОБЛЕМЫ И ПЕРСПЕКТИВЫ РАЗВИТИЯ ИТЕРАЦИОННЫХ ПРОЦЕССОВ

Проблема высокопроизводительного решения СЛАУ с системной точки зрения – это широко востребованная сфера интеллектуальной деятельности, которая далеко не ограничивается написанием и обоснованием формул алгоритма, а включает и постановки новых практических задач

или теоретических идей, и технологические аспекты программных реализаций, и вопросы их эффективного применения в суперкомпьютерном эксперименте.

Новые практические вызовы математического моделирования кардинально поднимут требованность решения междисциплинарных и обратных задач с высоким разрешением на реальных данных, что только актуализирует высокопроизводительные решения СЛАУ с числом степеней свободы в десятки и сотни миллиардов и с экстремально плохой обусловленностью. Последние десятилетия методы вычислительной алгебры бурно развиваются как в теоретическом плане, так и в экспериментальном, и здесь сложились перспективные направления: мало-ранговые приближения матриц, в том числе рациональные аппроксимации, тензорные методологии, теоретико-графовые и комбинаторные подходы, методы агрегации и сегрегации, а также появляются новые оригинальные идеи в традиционных матричных факторизациях, алгебраических декомпозициях, многосеточных технологиях и т.д.

Другой важный фактор — это происходящее накопление огромного количества разнотипных задач (междисциплинарных, методических и практических, типовых и уникальных), а также методов и технологий их решения на разных классах компьютеров. При этом происходит переход количества обрабатываемых данных в качество, требующий концепции разработки математического и программного обеспечения нового поколения. Одна из главных его задач — создание активной базы математических знаний, призванной обеспечить автоматизацию или оптимизацию построения алгоритмов и их отображения на архитектуру суперкомпьютера, прототипом которой в определенной степени можно считать проект ALGOWIKI, разрабатываемый по технологиям Википедии под руководством Дж. Донгарры и Вл.В. Воеводина. Такая проблема относится к прерогативе искусственного интеллекта с машинным обучением, который вместе с технологиями работы с большими данными составляет фундамент современного моделирования. Достигнутые масштабы триады “задачи—методы—компьютеры” приводят к тому, что интегрированное вычислительное окружение для решения СЛАУ должно стать формой и содержанием иерархической инфраструктуры, составляющей инструментальную среду индустриального типа и реализующей дальнейшие этапы развития наукоемких вычислительно-информационных технологий, контуры которых очерчены в [1], [24], [93]. Чтобы углубленная специализация не привела к разобщению теоретиков-вычислителей, программистов-технологов и прикладных пользователей, их общие усилия должны быть направлены на создание интеллектуализированной экосистемы, поддерживающей межпрофесситехнологоональные и человеко-машинные интерфейсы как для интенсификации фундаментальных исследований, так и для быстроты реализации их инноваций.

Автор выражает искреннюю благодарность двум анонимным рецензентам, замечания которых значительно способствовали улучшению текста статьи.

СПИСОК ЛИТЕРАТУРЫ

1. *I'in V.P.* Mathematical modeling. Part I. Continuous and discrete models. Novosibirsk, SBRAS Publ., 2017.
2. *Axelsson O.* Iterative solution methods. Cambridge: Cambridge Univ. Press, 1994.
3. *Elman H.C., Silvester D.J., Wathen A.J.* Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics. Numerical mathematics and scientific computation. Oxford Univ. Press, 2nd ed., 2014.
4. *Marchuk G.I., Kuznetsov Yu.A.* Iterative methods and quadratic functionals (in Russian). CC SBRAS Publ., Novosibirsk, 1972.
5. *Saad Y.* Iterative methods for sparse linear systems, 2nd ed. SIAM, 2003.
6. *Van der Vorst H.A.* Iterative Krylov methods for large linear systems. Cambridge Univ. Press, 2003.
7. *Olshanskii M.A., Tyrtysnikov E.E.* Iterative methods for linear systems theory and applications. SIAM Philadelphia, 2014.
8. *Liesen J., Strakos Z.* Krylov subspace methods, principles and analysis. Oxford Univ. Press, 2013.
9. *I'in V.P.* Finite element methods and technologies (in Russian). ICM&MG SBRAS, Publ., Novosibirsk, 2007.
10. *I'in V.P.* Iterative incomplete factorization methods. Singapore, World Sci. Publ., 1992.
11. *Fang H., Saad Y.* Two classes of multisection methods for nonlinear acceleration // Numer. Linear. Alg. Appl. 2009. V. 16. № 3. P. 197–221.
12. *Prataru P.P., Suryanarayana J.E.* Anderson acceleration of the Jacobi iterative method. An efficient alternative to Krylov methods for large, sparse linear systems // J. Comput. Phys. 2016. V. 306. P. 43–54.
13. *Walkert H.F., Ni P.* Anderson acceleration for fixed-point iterations // SIAM J. Assoc. Comput. Math. 2011. V. 49. P. 1715–1735.

14. *Il'in V.P.* Iterative processes in the Krylov–Sonneveld subspaces // *J. Math. Sci.* 2017. V. 24. № 6. P. 890–899.
15. *Butyugin D.S., Gurieva Y.L., Il'in V.P., Perevozkin D.V.* Some geometric and algebraic aspects of domain decomposition methods // *LNCSE.* 2016. V. 104. P. 117–126.
16. Domain Decomposition Methods. URL: <http://ddm.org>
17. *Il'in V.P.* Multi-preconditioned domain decomposition methods in the Krylov subspaces // *LNCS*, 10187, Springer, 2017. P. 95–106.
18. *Il'in V.P.* Problems of parallel solution of large systems of linear algebraic equations // *J. Math. Sci.* 2016. V. 216. № 6. P. 795–804.
19. *Toselli A., Widlund O.* Domain decomposition methods: algorithms and theory. Berlin: Springer, 2005.
20. *Vabishchevic P.N.* Incomplete iterative implicit schemes // *Comput. Meth. Appl. Math.* 2019.
21. *Il'in V.P.* High-performance computation of initial boundary value problems. Springer Nature Switzerland AG, L. Sokolinsky and M. Zymbler (Eds.): PCT 2018, CCIS 910, 2018, P. 186–199.
22. *Dongarra J.* List of freely available software for linear algebra on the web (2006). <http://netlib.org/utk/people/JackDongarra/la-sw.html>
23. *Barret R., Bery M., Chan T.F., Demmel J., Donato J., Dongarra J., Eijkhout V., Pozo R., Romine C., van der Vorst H.A.* Templates for the solution of linear systems. Building blocks for iterative methods. SIAM Philadelphia, PA, 1994.
24. *Il'in V.P.* On an integrated computational environment for numerical algebra. Springer Nature Switzerland AG 2019 L. Sokolinsky and M. Zymbler (Eds.): PCT 2019, CCIS 1063, 2019, P. 91–106.
25. *Arioli M.* Generalized Golub–Kahan diagonalization and stopping criteria // *SIAM J. Matrix Anal. Appl.* 2013. V. 34. № 2. P. 57–592.
26. *Higham N.J.* Accuracy and stability of numerical algorithms. SIAM, Philadelphia, 2002.
27. *Il'in V.P.* Finite difference and finite volume methods for elliptic equations (in Russian). Novosibirsk ICM&MG SBRAN Publ., 2001.
28. *Soodhalter K.M., Sturler E., Kilmer M.* A survey of subspace recycling iterative methods. <https://arxiv.org/abs/2001.10347>
29. *Choi S.-C.T., Paige C.C., Saunders M.A.* INRES-QLP: A Krylov subspace method for indefinite or singular symmetric systems. SIAM, 2015, arXiv:1003.4042 [math.NA] 2
30. *Kaporin I.E.* New convergence results and preconditioning strategies for the conjugate gradient method // *J. Numer. Linear. Algebra Applic.* 1994. V. 1. № 2. P. 179–210.
31. *Kaporin I.E.* High quality preconditioning of a general symmetric positive definite matrix based on its UTU+UTR+RTU-?Decomposition // *J. Numer. Linear. Algebra Applic.* 1998. V. 5. № 6. P. 483–509.
32. *Kaporin I.E., Konshin I.N.* A parallel block overlap preconditioning with inexact submatrix inversion for linear elasticity problems // *J. Numer. Linear. Algebra Applic.* 2002. V. 9. P. 141–162.
33. *Il'in V.P.* Methods of semiconjugate directions // *Russian J. Numer. Anal. Math. Model.* 2008. V. 23. № 4. P. 369–387.
34. *Il'in V.P.* Two-level least squares methods in Krylov subspaces // *J. Math. Sci.* 2019. V. 232. № 6. P. 892–901.
35. *Knizhnerman L.A.* Error bounds for the arnoldi method: a set of extreme eigenpairs // *J. Numer. Linear. Algebra Applic.* 1999. V. 296. P. 191–211.
36. *Il'in V.P.* Biconjugate direction methods in Krylov subspaces // *J. Appl. Indust. Math.* 2010. V. 4. № 1. P. 65–178.
37. *Neuenhofen M.P., Greif C.* MSTAB: stabilized inducted dimension reduction for KRYLOV subspace recycling // *SIAM J. Sci. Comput.* 2018. V. 40. № 2. P. 554–571.
38. *Saunders M.A., Simon H.D., Yip E.L.* Two conjugate-gradient-type methods for unsymmetric linear equations // *SIAM J. Numer. Anal.* 1988. V. 25. № 4. P. 927–940.
39. *Estrin R., Greif C.* SPMR: a family of saddle-point minimum residual solvers // *SIAM J. Sci. Comput.* 2018. V. 40. № 3. P. 1884–1914.
40. *Notay Y.* Flexible conjugate gradients // *SIAM J. Sci. Comput.* 2000. V. 22. № 4. P. 1444–1460.
41. *Bradley A.M.* Algorithms of the equilibration of matrices and their application to limited-memory quasi-newton methods // PhD thesis, ICME, Stanford Univer., 2010.
42. *Livne O.E., Golub G.H.* Scaling by binormalization // *J. Numer. Algorithms.* 2004. V. 35. P. 97–120.
43. *Il'in V.P., Kellog R.* Analysis of flow directed iterations // *J. Comp. Math.* 1992. V. 10. № 1. P. 1–18.
44. *Arnold D.N., Faik R.S., Winther R.* Preconditioning in $H(\text{div})$ and applications // *J. Math. Comput.* 1997. V. 66. P. 937–984.
45. *Benzi M.* Preconditioning techniques for large linear systems: a survey // *J. Comput. Phys.* 2002. V. 82. № 2. P. 418–477.
46. *Higham N.J., Mary T.* A new preconditioner that exploits low-rank approximations for factorization error // *SIAM J. Sci. Comput.* 2019. V. 41. № 1. P. 59–82.

47. *Hiptmair R.* Operator preconditioning // J. Comput. Math. Appl. 2006. V. 52. № 5. P. 699–706.
48. *Mardal K., Winther R.* Preconditioning discretizations of systems of partial differential equations // J. Numer. Linear. Algebra Appl. 2011. V. 18. P. 1–40.
49. *Vassilevski P.S.* Multi-level block factorization preconditioners. New York: Springer, 2008.
50. *Wathen A.* Preconditioning // Acta Numer. 2015. № 24. P. 329–376.
51. *Bridson R., Greif C.* A multipreconditioned conjugate gradient algorithm // SIAM J. Matrix Anal. Appl. 2006. V. 27. № 4. P. 1056–1068.
52. *Dolean V., Jolivet P., Nataf F.* An introduction to domain decomposition methods: algorithms, theory and parallel implementation. SIAM, Philadelphia, 2015.
53. *Gurieva Y.L., Il'in V.P.* On coarse grid correction methods in the Krylov subspaces // J. Math. Sci. 2018. V. 232. № 6. P. 774–182.
54. *Laevsky Y.M., Matsokin A.M.* Decomposition methods for solving elliptic and parabolic boundary value problems (in Russian). 1999. V. 2. P. 361–372.
55. *Vassilevski Y.* A hybrid domain decomposition method based on aggregation // Numer. Linear. Alg. Appl. 2004. V. 11. P. 327–341.
56. *Vassilevski Y.V., Olshanskii M.A.* Short course on multi-grid and domain decomposition methods. Moscow, MAKS Press Publ., 2007.
57. *Xiang H., Nataf F.* Two-level algebraic domain decomposition preconditioners using Jacobi-Schwarz smoother and adaptive coarse grid corrections // J. Comput. Appl. Math. 2014. V. 261. P. 1–13.
58. *Gaul A., Gutknecht M.H., Hesenta J., Nabben R.* A framework for deflated and augmented Krylov subspace methods // SIAM J. Matrix Anal. Appl. 2013. V. 34. № 2. P. 495–518.
59. *Nabben R., Vuik C.* A comparison of deflation and coarse grid correction applied to porous media flow // SIAM J. Numer. Anal. 2004. V. 42. P. 631–647.
60. *Bank R., Falgout R., Jones T., Manteuffel T., McCormick S., Ruge J.* Algebraic multigrid domain and range decomposition (AMG-DD/AMG-RD) // SIAM J. Sci. Comput. 2015. V. 37. P. 113–136.
61. *Brandt A.* Algebraic multigrid theory: the symmetric case // J. Appl. Math. Comput. 1986. № 19. P. 23–56.
62. *Shaidurov V.V.* Some estimates of the rate of convergence for the cascading conjugate-gradient method // J. Comput. Math. Applic. 1996. V. 31. № 4/5. P. 161–171.
63. *Gurieva Y.L., Il'in V.P., Petukhov A.V.* On multigrid methods for solving two-dimensional boundary-value problems // J. of Math. Sci. 2020. V. 249. № 2. P. 118–127.
64. *Notay Y.* Algebraic multigrid for Stokes equations // SIAM J. Sci. Comput. 2017. V. 39. P. 88–111.
65. *Notay Y., Napov A.A.* An efficient multigrid method for graph Laplacian systems II: robust aggregation // SIAM J. Sci. Comput. 2017. V. 39. № 5. P. 379–403.
66. *Olshanskii M.A.* Lecture notes on multigrid methods. Moscow: INM RAS PUB, 2012.
67. *Tang J.M., Nabben R., Vuik C., Erlangga Y.A.* Comparison of two-level preconditioners derived from deflation, domain decomposition and multigrid methods // J. Sci. Comput. 2009. № 39. P. 340–370.
68. *Varga R.S.* Iterative analysis, Springer, 1962.
69. *Kolotilina L.Yu.* The convergence of certain incomplete block factorization splittings // J. Math. Sci. (New York). 1997. V. 86. № 4. P. 2828–2834.
70. *Yeremin A.Yu., Kolotilina L.Yu., Nikishin A.A.* Factorized sparse approximate inverse preconditioning III. Iterative construction of preconditioners // J. Math. Sci. (New York). 2000. V. 101. № 4. P. 3237–3254.
71. *Kumar P., Grigori L., Nataf F., Nui Q.* On relaxed nested factorization and combination preconditioning // Inter. J. Comput. Math. 2016. V. 93. № 1. P. 178–199.
72. *Vassilevski P.S.* Preconditioning mixed finite element saddle-point elliptic problems // J. Numer. Linear Algebra Applic. 1996. V. 3. Iss. 1. P. 1–20.
73. *Hackbusch W.* Hierarchische Matrizen: Algorithmen und Analysis. Berlin: Springer, 2009.
74. *Solov'yev S.A.* Multifrontal hierarchically solver for 3D discretized elliptic equations, In: Dimov I., Faragi I., Vulkov L. (eds.) FDM 2014. LNCS, Springer, Cham, 2015, V. 9045, P. 371–378.
75. *Arioli M., Manzini M.* A network programming approach in solving Darcy's equations by mixed finite elements methods // Electron. Transact. on Numer. Anal. 2006. V. 22. P. 41–70.
76. *Bern M., Gilbert J., Hendrickson B., Nguyen N., Toledo S.* Support-graph preconditioners // SIAM J. Matrix Anal. Appl. 2006. V. 27. № 4. P. 930–951.
77. *Ponce C., Vassilevski P.S.* Solving graph Laplacian systems through recursive partitioning and two-grid preconditioning // SIAM J. Matrix Anal. Appl. 2017. V. 38. № 2. P. 621–648.
78. *Il'in V.P., Laevsky K.Yu.* Generalized compensation principle in incomplete factorization methods // Russian J. Numer. Anal. and Mathem. Model. 1997. V. 12. № 5. P. 399–412.
79. *Heath M., Romine C.* Parallel solution of triangular systems on distributed-memory multiprocessors // SIAM J. Sci. Stat. Comp. 1988. V. 9. № 3. P. 558–588.

80. *Hutter E., Solomonik E.* Communication-avoiding Cholesky QR-2 for rectangular matrices. arXiv: 1710.08471v6
81. *Ruhe A.* Rational Krylov sequence methods for eigenvalue computation // *J. Linear Algebra Appl.* 1984. V. 58. P. 39–405.
82. *Kuznetsova N.N., Diyankov O.V., Kotegov S.V., Krasnogorov I.V., Pravilnikov V.Y., Maliassov S.Y.* The family of nested factorizations // *Russian J. Numer. Anal. Math. Model.* 2007. V. 22. P. 393–412.
83. *Wang R., Nu Q, Lu L.* A twisted block tangential filtering decomposition preconditioner. *Nath. Prob. In Engineer.*, 2009, Article ID 282307.
84. *Benzi M., Golub G.H., Liesen J.* Numerical solution of saddle point problems // *Acta Numerica.* 2005. V. 14. P. 1–137.
85. *Brezzi F.* Stability of saddle points in finite dimensions, frontiers in numerical analysis. Berlin, Heidelberg: Springer, 2013, P. 17–61.
86. *Notay Y., Napov A.* Further comparison of additive and multiplicative coarse grid correction // *J. Appl. Numer. Math.* 2013. V. 65. P. 53–62.
87. *Bychenkov Y.V., Chizhonkov E.V.* Iterative methods for solving saddle point problems (in Russian). Moscow: Binom Publ., 2010.
88. *Greif C., Schotzau D.* Preconditioners for saddle point linear systems with highly singular (1,1) Blocks, in: Special Volume on Saddle Point Problems // *Electron. Trans. J. Numer. Anal.* 2006. V. 22. P. 114–121.
89. *Greif C., Schotzau D.* Preconditioners for the discretized harmonic Maxwell equations in mixed form // *Numer. Linear Algebra Applic.* 2007. V. 14. P. 281–287.
90. *Notay Y.* Convergence of some iterative methods for symmetric saddle point linear systems // *SIAM J. Matrix Anal. Appl.* 2019. V. 40. № 1. P. 122–146.
91. *Il'in V.P., Kazantsev G.Y.* Iterative solution of saddle-point systems of linear equations // *J. Math. Sci.* 2020. V. 249. № 2. P. 199–2008.
92. *Notay Y.* Algebraic two-level convergence theory for singular systems // *SIAM J. Matrix Anal. Appl.* 2016. V. 37. P. 1419–1439.
93. *Dongarra J., Grigori L., Higham N.J.* Numerical algorithms for high performance computational science // *Phil. Trans. R. Soc.* 2020, A 378.
94. *Il'in V.P.* Projection methods in Krylov subspaces // *J. Math. Sci.* 2019. V. 240. № 6. P. 772–782.
95. *Gorbenko N.I., Il'in V.P.* The additive Peaceman–Rachford method // *J. Math. Sci.* 2016. V. 216. № 6. P. 753–760.
96. *Aleynikov A.Y., Barabanov R.A., Bartenev Y.G., et al.* An application of parallel solvers for SLAEs in the applied packages for engineering (in Russian), in: *Proceed. Inter. Conf. “Supercomputing and Mathematical Modeling”*, Unicef. Publ., 2015, P. 102–110.
97. *Bastian P., Blatt M.* Iterative solver template library (DUNE). <https://www.dune-project.org/>
98. *Vassilevski Y.V., Konshin I.N., Kopytov G.V., Terekhov K.M.* INMOST-program platform and graphic media for development of the parallel numerical methods on general type grids (in Russian). Moscow, MSU Publ., 2018.