ОБ ИНТЕГРИРОВАННОМ ВЫЧИСЛИТЕЛЬНОМ ОКРУЖЕНИИ ДЛЯ ВЫСОКОПРОИЗВОДИТЕЛЬНОГО РЕШЕНИЯ СЛАУ

В. П. Ильин

Институт вычислительной математики и математической геофизики СО РАН, Новосибирск
Новосибирский государственный университет, Новосибирск

Рассматривается концепция и общая структура интегрированного вычислительного окружения (ИВО), предназначенного для высокопроизводительного решения широкого класса систем линейных алгебраических уравнений (СЛАУ), возникающих из конечно-объемных или конечно- элементных аппроксимаций многомерных краевых задач на неструктурированных сетках. Распараллеливание алгоритмов осуществляется на основе двухуровневых методов декомпозиции подобластей в подпространствах Крылова с параметризованными пересечениями подобластей и разными типами интерфейсных условий, с применением различных предобуславливателей и ускорительных алгоритмов грубосеточной коррекции, дефляции и малоранговой аппроксимации исходных матриц, представляемых с помощью сжато-разреженных алгебраических структур данных (АСД). Для больших СЛАУ (10^9 и более уравнений) АСД формируется в распределенном по вычислительным процессорам (MPI-процессам) виде для последующей параллельной реализации итерационных методов. Синхронное решение вспомогательных СЛАУ в подобластях выполняется с помощью многопотоковых вычислений, векторизации операций и графических ускорителей GPGPU. Разрабатываемые алгоритмы реализуются в составе библиотеки программ KRYLOV, которая представляет собой прототип ИВО и предусматривает длительный жизненный цикл с гибким расширением типов решаемых задач и состава алгоритмов, с адаптацией к эволюции суперкомпьютерных платформ, с многоязыковостью модулей, с переиспользованием внешних программных продуктов и с возможностями согласованного участия в проекте различных групп разработчиков.

Ключевые слова: алгебраические системы, разреженные матрицы, структуры данных, предобусловленные итерационные методы, подпростраества Крылова, параллельные алгоритмы, интегрированное вычислительное окружение.

1. Введение

Решение систем линейных алгебраических уравнений (СЛАУ) – это безусловно основная задача вычислительной алгебры, которая, в свою очередь, является ключевым этапом в проблемах математического моделирования процессов и явлений. При всем разнообразии классических и обобщенных, дифференциальных и/или интегральных, стационарных и эволюционных, линейных и нелинейных исходных постановок, после дискретизации, аппроксимации и линеаризации мы приходим к необходимости решения СЛАУ. Более того, эта стадия является «узким горлышком» крупномасштабного вычислительного эксперимента, поскольку здесь ресурсоёмкость реализации, как правило, растет нелинейно с ростом числа степеней свободы и составляет главную часть в оценках вычислительной сложности расчетов.

Естественно, что главным объектом нашего внимания является решение «больших» и «трудных» систем на современных гетерогенных многопроцессорных вычислительных системах (МВС) с распределенной и иерархической общей памятью. Большими будем считать СЛАУ с числом неизвестных N порядка $10^8 – 10^{11}$, а трудными – у которых матрица $A = \{a_{l,m}\}$ алгебраической системы имеет число обусловленности condA около $10^{12} – 10^{15}$. Мы здесь и далее предполагаем, что арифметические действия выполняются со стандартной двойной точностью при 64-битовом представлении числа с плавающей запятой, обеспечивающей относительную погрешность примерно 10^{-15}. Важно подчеркнуть, что с увеличением быстродействия суперкомпьютера до постпетафлопсной и выше актуальность проблемы решения СЛАУ не уменьшается, так как «аппетит приходит во время еды» и одновременно растут как порядки текущих реализаций алгебраических систем, так и числа их обусловленности. Можно уверенно прогнозировать, что в недалеком будущем двойной точности окажется недостаточно для устойчивых вычислений и тогда станет насущной проблема использования «умных» арифметических действий с переменной разрядностью чисел, включающей, например, представления с количеством бит как больше 100, так и меньше 50.

Значительный интерес представляют СЛАУ, получаемые после аппроксимации исходных многомерных краевых задач для дифференциальных уравнений или их обобщённых вариационных постановок с помощью методов конечных разностей, конечных объемов, конечных элементов или разрывных методов Галеркина (МКР, МКО, МКЭ или РМГ соответственно) на неструктурированных сетках. Порождаемые при этом матрицы имеют две, как правило, важные особенности. Во-первых, они являются разреженными и ленточными, т. е. общее количество ненулевых элементов относительно мало ($NNZ \ll N^2$), причем все они расположены в полосе шириной $M \ll N$ около главной диагонали. Во-вторых, их портреты имеют нерегулярную структуру, когда номера внедиагональных ненулевых элементов в каждой строке могут быть заданы только перечислением. Это приводит к необходимости представления матриц в разреженных сжатых форматах (например, CSR – Compressed Sparse Row), основанных на хранении в памяти только ненулевых элементов и их номеров, что обуславливает значительную специфику программной реализации алгоритмов, а также замедляет процесс доступа к значениям матричных элементов.

Важной характеристикой СЛАУ, в значительной степени влияющей на свойства алгоритмов ее решения, является блочная структура, которая определяется особенностями и исходной краевой задачи, и используемой сетки, и применяемыми методами аппроксимации, и выбираемой нумерации векторных компонент. Одним из наиболее сложных в структурном смысле матриц являются те, которые возникают из сеточных аппроксимаций многомерных междисциплинарных задач, описываемых системой функциональных уравнений со многими векторными неизвестными функциями (например, распределения плотностей каких-то субстанций, давления, температуры, скорости, электромагнитных полей и т. д.).

Главным средством решения больших разреженных систем являются итерационные методы, вследствие их меньших требований к объёму используемой памяти и количеству затрачиваемых вычислительных операций. Наиболее эффективными и распространёнными являются предобусловленные алгоритмы в подпространствах Крылова, являющиеся одним из главных достижений вычислительной математики в XX веке, по которым последние десятилетия непрерывно ведутся активные исследования и публикуется огромное количество литературы, из которой мы ограничимся ссылками на монографии [1, 2]. Особое место в данных вопросах занимает проблема масштабируемого распараллеливания алгоритмов на МВС. Здесь главным средством является метод декомпозиции областей (МДО), ставший одним из актуальных разделов вычислительной алгебры, которому посвящены специальные монографии, конференции и интернетовские сайты, см. [3, 4]. Главный принцип МДО заключается в разбиении исходной задачи для сложной расчетной области на взаимосвязанные вспомогательные задачи в подобластях, каждая из которых может решаться синхронно на соответствующем процессоре супер-ЭВМ. Что касается технологий распараллеливания, то здесь масштабирование достигается средствами гибридного программирования с использованием MPI – процессов, многопотоковых вычислений и векторизации операций. В определенном смысле альтернативу МДО представляют многосеточные подходы, использующие аппроксимации исходной краевой задачи на последовательности сгущающихся сеток, см. монографию [5] и цитируемую там литературу, которые в теоретическом плане асимптотически на гладких решениях дают оптимальные оценки объема вычислений, но являются достаточно трудно распараллеливаемыми на МВС.

Исследование итерационных методов и их практические реализации осуществляются за последние десятилетия в двух главных направлениях. Первое – это разработка экономичных способов предобуславливания исходных СЛАУ, с целью улучшения их обусловленности, для чего активно конструировались многочисленные алгоритмы приближенной факторизации матриц, см. обзоры в [1, 2]. Второй путь состоит в развитии самих итерационных процессов в подпространствах Крылова, где различные подходы связаны с методами дефляции, агрегации, грубосеточеной коррекции, малоранговых аппроксимаций матриц, многошаговые методы наименьших квадратов и т. д., см. обзоры в [6, 7]. Однако здесь следует заметить, что такие приемы ускорения можно интерпретировать как построение предобуславливателей специального вида.

Важным также является тот факт, что очень большое число алгебраических методов уже реализовано и является общедоступным в компьютерных сетях. Достаточно полный список таких разработок, овеществляющих огромный интеллектуальный потенциал, приведен в [8]. Здесь в первую очередь следует отметить библиотеки векторно-матричных операций BLAS и SPARSE BLAS, содержащие, в частности, высокопроизводительные реализации для различных типов суперкомпьютеров. Использование таких типовых функций в прикладных разработках, как правило, значительно повышает эффективность получаемых программ.

Целью представленной работы является разработка концепции, архитектуры, структуры данных и основных компонент интегрированного вычислительного окружения (ИВО) для высокопроизводительного решения широкого класса СЛАУ, ориентированного на его эффективное использование в прикладных программных комплексах (ППК) и на активную востребованность у конечных пользователей с различной профессиональной подготовкой. Рассматриваемое математическое и программное обеспечение реализуется в составе библиотеки KRYLOV [9], представляющей собой подсистему базовой системы моделирования БСМ [10], предназначенной для поддержки всех основных технологических стадий математического моделирования на суперЭВМ.

Данная работа построена следующим образом. В разделе 2 мы приводим классификацию рассматриваемых типов СЛАУ. Разделы 3, 4 посвящены краткому изложению методов решения алгебраических систем и существующего программного обеспечения по их реализации. В следующем разделе описываются принципы построения и технические требования к разработке ИВО, а в заключении обсуждаются планы дальнейших работ по развитию интегрированной инструментальной среды для решения задач вычислительной алгебры. Отметим, что мы зачастую для краткости употребляем без ссылок сложившиеся в вычислительной алгебре термины и понятия, которые легко находятся в интернете.

2. Классификация рассматриваемых задач

С формальной точки зрения, перед нами стоит тривиальная математическая проблема о решении СЛАУ

$$Au = f, \quad A = \{a_{l,m}\} \in R^{N,N}, \quad u = \{u_l\}, f=\{f_l\} \in R^N, \tag{1}$$

где матрицу A порядка N для простоты мы считаем вещественной, квадратной и положительно определенной в смысле выполнения условия

$$(A\mathrm{v}, \mathrm{v}) \geq \delta \, \|\mathrm{v}\|^2, \quad \delta > 0 \quad (\mathrm{v},\mathrm{w}) = \sum_{i=1}^{N} \mathrm{v}_i \mathrm{w}_i, \qquad \|\mathrm{v}\|^2 = (\mathrm{v}, \mathrm{v}), \tag{2}$$

но симметричность при этом не требуется. Большинство (точнее говоря, некоторые) из рассматриваемых ниже алгоритмов применимы и для более общих СЛАУ: комплексных (эрмитовых или неэрмитовых), вырожденных или знаконеопределенных. Особый интерес представляют разреженные системы, возникающие из аппроксимации многомерных краевых задач методами конечных разностей, конечных объектов, конечных элементов или разрывными алгоритмами Галеркина на неструктурированных сетках, см. [11] и цитируемую там литературу. Такого типа сеточные уравнения удобно записывать в виде

$$a_{i,i}u_i + \sum_{j \in \omega i} a_{i,j} u_j = f_i, \quad i \in \Omega^h, \tag{3}$$

где ω_i обозначает множество внедиагональных ненулевых элементов, лежащих на i-й строке матрицы. С другой стороны, иногда матрицы представляются в блочной форме

$$\begin{gathered} A_{q,q}u_q + \sum_{r \in \Omega q}^{n} A_{q,r}u_r = f_q, \quad q = 1,\ldots,P, \\ A_{q,r} \in R^{N_q,N_r}, \quad f_q \in R^{N_q}, \quad N_1 + \ldots + N_p = N, \end{gathered} \tag{4}$$

где Ω_q означает множество номеров ненулевых матричных вычислительных блоков, расположенных в q-й блочной строке.

В случае сеточных алгебраических уравнений блочное представление СЛАУ можно наглядно ассоциировать с геометрической декомпозицией сеточной расчётной области Ω на непересекающиеся подобласти Ω_q:

$$\Omega = \Omega_1 + \ldots + \Omega_P, \quad \text{где} \quad \Omega_q \cap \Omega_r = 0 \quad \text{при } q \neq r. \tag{5}$$

Соотношения (5) можно интерпретировать также как алгебраическую декомпозицию, если Ω_q рассматривать просто как множество из N_q индексов, соответствующих подвектору u_q или f_q.

Приведение матриц к блочному виду связано с перенумерацией векторных компонент и матричных строк, что может порождать большое количество вычислительных методов и технологий. Например, на основе первоначальной алгебраической декомпозиции сеточной расчетной области вида (5) без пересечений можно сформировать декомпозиции с параметризованными пересечениями, при различных числах общих сеточных слоев. При этом для больших матриц естественно рассматривать их представление в памяти распределенным образом, как в смысле различных физических процессоров, так и на логическом уровне MPI-процессов. Аналогичные проблемы перенумерации и формирования различных блочных структур возникают и в эффективных многосеточных подходах решения СЛАУ, см. [5].

Структурные свойства рассматриваемых матриц сильно зависят, естественно, от особенностей исходных краевых задач и от способов их аппроксимации. Типовые блочные характеристики и матричные портреты можно классифицировать по типу систем уравнений для соответствующих прикладных проблем: тепломассопереноса, гидро-газодинамики, напряженно-деформированного состояния, многофазной фильтрации, электромагнетизма и т. д. Важно также подчеркнуть, что с теоретической и практической точек зрения нас интересует решение не конкретной СЛАУ на фиксированной сетке с характерным шагом h, а серии однотипных алгебраических систем на последовательности сгущающихся (возможно, вложенных) сеток.

В последние десятилетия актуализируются аппроксимации повышенной точности, имеющие погрешность численного решения порядка $O(h^\gamma)$ где $\gamma > 1$. С ростом порядка (до значений $\gamma = 2, 4, 6$ и выше) увеличивается количество *NNZ* ненулевых матричных элементов и вычислительная сложность алгоритмов, но зато значительно уменьшается объем памяти, необходимой для обеспечения заданной точности результата. Последний фактор приводит к сокращению коммуникаций, которые не только замедляют общий вычислительный процесс, но и являются наиболее энергозатратными операциями.

С технологической точки зрения важно также классифицировать матрицы по способам их хранения. Следует отметить, что помимо универсального формата CSR, существуют еще другие общераспространенные матричные представления, а также программные конверторы из одних форматов в другие, см., например, библиотеку INTEL MKL [11]. Отметим, в частности, «мелкоблочные» форматы, построенные по аналогии с CSR, но в которых вместо числовых элементов фигурируют матрицы фиксированного небольшого размера. Естественно также, что для некоторых матриц простой структуры можно использовать свои экономичные форматы. Например, для симметричной матрицы достаточно хранить только главную диагональ и одну из ее треугольных частей. Если же матрица является тёплицевой или квазитёплицевой (а может – еще и ленточной), то тогда проблема хранения её ненулевых элементов вообще отсутствует. Наконец, отметим еще специальный тип матриц, которые будем называть блочно-структурированными, характеризующимися различными способами хранения для разных блоков. Такие представления могут иметь высокую экономичность при использовании квазиструктурированных сеток [12], в которых сеточная расчётная область разбивается на сеточные подобласти с разными типами сеток, в том числе регулярными и равномерными.

Отметим еще такой актуальный круг векторов, как решение СЛАУ на каждом временном шаге при использовании неявных аппроксимаций для численного интегрирования многомерных начальных краевых задач. В работе [7], например, показано, что специальный выбор начальных приближений с помощью методов наименьших квадратов позволяет значительно сократить трудоёмкость расчетов.

3. Предобусловленные методы в подпространствах Крылова

Нашей целью в данном разделе является краткий обзор современных итерационных алгоритмов решения СЛАУ, а также технологий их распараллеливания и программных реализаций в общедоступных библиотеках.

3.1. Мультипредобусловленные методы полусопряжённых направлений

Главные современные подходы к решению больших разреженных СЛАУ – это предобусловленные итерационные методы в подпространствах Крылова, которые по своей вычислительной сложности и ресурсоёмкости, а также по способам реализации делятся на две группы – алгоритмы решения симметричных и несимметричных алгебраических систем. Первая группа – это главным образом экономичные методы сопряженных градиентов и сопряженных невязок (CG и CR соответственно), основанные на коротких (двучленных) рекурсиях. Для решения несимметричных систем существуют алгоритмы, основанные на биортогонализации вычисляемых векторов аналогично с помощью коротких рекурсий. Здесь примерами могут служить методы BiCG и BiCR, также их стабилизированные версии BiCGStab и BiCRStab, однако такие алгоритмы менее проработаны в теоретическом и практическом отношениях. Наиболее популярными и надежными являются методы обобщенных минимальных невязок (GMRES или FGMRES), имеющие теоретически оптимальную скорость сходимости итераций и высокую практическую надежность (робастность), но основанные на вычислениях длинных векторных рекурсий. Мы рассмотрим эквивалентные по скорости сходимости итераций методы полусопряженных направлений, являющиеся непосредственным обобщением алгоритмов CG и CR на несимметричные СЛАУ. Мы дадим формальное изложение этих итерационных процессов в «мультипредобусловленном» варианте с возможностью использования на каждом шаге по нескольку предобуславливающих матриц, количество и форма которых при этом может изменяться. Такие методы относятся к классу блочных, поскольку они используют по нескольку направляющих векторов, а порождаемые ими крыловские подпространства естественно рассматривать как блочные, см. [13] и цитируемые там работы:

$$\begin{gathered} r^0 = f - Au^0, \quad n = 0,\ldots, \quad u^{n+1} = u^n + P_n\bar{\alpha}_n, \\ r^{n+1} = r^n - AP_n\bar{\alpha}_n = r^q - AP_q\bar{\alpha}_q - \ldots - AP_n\bar{\alpha}_n, \quad 0 \le q \le n, \\ P_n = \left(p_q^n,\ldots,p_{M_n}^n\right) \in R^{N,M_n}, \quad \bar{\alpha}_n\left(\alpha_{n,1,\ldots,}\alpha_{n,1}\right)^T \in R^{M_n}. \end{gathered} \tag{6}$$

Здесь $\bar{\alpha}_n$ суть векторы коэффициентов, а P_n – матрицакаждый из M_n столбцов которой есть направляющий вектор p_l^n, связанный с соответствующей предобуславливающей матрицей $B_{n,l}$, $l = 1,\ldots, M_n$, вид которой пока не конкретизируем. Соотношение (6) обладает тем замечательным свойством, что если направляющие векторы удовлетворяют условиям ортогональности

$$\begin{gathered} \left(Ap_k^n, A^\gamma p_{k'}^{n'}\right) = \rho_{n,k}^{(\gamma)}\delta_{n,n'}^{k,k'}, \quad \rho_{n,k}^{(\gamma)} = \left(Ap_k^n, A^\gamma p_k^{\dot{n}}\right) \\ \gamma = 0,1 \quad n' = 0,1,\ldots,n-1, \quad k,k' = 1,2,\ldots,M_{n,} \end{gathered} \tag{7}$$

то имеют место следующие соотношения для функционалов невязки:

$$\Phi_n^{(\gamma)}\left(r^{n+1}\right) \equiv \left(r^{n+1}, A^{\gamma-1}r^{n+1}\right) = \left(r^q, A^{\gamma-1}r^q\right) - \frac{\sum_{k=q}^{n}\sum_{l=1}^{M_n}\left(r^q, A^\gamma P_l^k\right)^2}{\rho_{k,l}^{(\gamma)}}, \tag{8}$$

если только коэффициенты определяются формулами

$$\alpha_{n,1} = \sigma_{n,l} \big/ \rho_{n,n}^{(\gamma)}, \quad \sigma_{n,l} = \left(r^0, A^\gamma p_l^n\right). \tag{9}$$

При этом функционал (8) достигает минимума в блочном подпространстве

$$K_n = \mathrm{Span}\left\{r^0, A\,p_1^0,\ldots,A\,p_{M_0}^0,\ldots,A\,p_1^n,\ldots,A\,p_{M_n}^n, \quad M = 1 + M_0 + \ldots M_n,\right. \tag{10}$$

в случае $\gamma = 1$. Для симметричной матрицы A минимум $\Phi_n^{(\gamma)}$ достигается при любом значении γ.

Условия ортогональности будут выполнены, в частности, если направляющие векторы определить с помощью некоторых предобуславливающих невырожденных матриц $B_{n,l}$ из следующих рекуррентных соотношений:

$$p_l^0 = B_{0,l}^{-1} r^0, \quad p_l^{n+1} = B_{n+1,l}^{-1} r^n - \sum_{k=0}^{n} \sum_{l=1}^{M_k} \beta_{n,k,l}^{(\gamma)} p_l^k, \quad n = 0,1,...,$$
$$B_{n,l} \in R^{N,N}, \quad \gamma = 0,1, \tag{11}$$
$$\beta_{n,k,l}^{\gamma} = -\frac{\left(A^{\gamma} p_l^k, AB_{n+1,l}^{-1} r^{n+1}\right)}{\rho_{n,l}^{(\gamma)}}, \quad n = 0,1,...; \quad k = 0,...,\mathrm{n}; \quad 1 = 1,...,\mathrm{M}_n .$$

При этом формула (10) определяет мультипредобусловленные подпространства Крылова.

В случае симметричности матриц A и $B_{n,l}$ из формул (6) – (11) получаем предобусловленные методы сопряженных градиентов и невязок (MPCG и MPCR для $\gamma = 0,1$, соответственно), при этом $\beta_{n,k,l}^{(\gamma)} = 0$ для $k < n$ и инаправляющие векторы вычисляются из двучленных рекурсий.

Если же $A \neq A^T$, то наиболее целесообразным, по-видимому, является применение мультипредобусловленного метода полусопряженных невязок MPSCR (Multi-Preconditioned Semi-Conjugate Residual). В этом случае при решении плохо обусловленных несимметричных СЛАУ, связанных с большим количеством итераций из-за ресурсоёмкости длинных рекурсий (главным образом вследствие увеличения объема используемой памяти) их приходится принудительно укорачивать, что делается путем или введения так называемой процедуры рестарта, или ограничения количества ортогонализируемых направляющих векторов (или матриц – при мультипредобуславливании), или использования обоих подходов одновременно. Во всех этих случаях скорость сходимости итераций падает, иногда очень сильно, что является неизбежной платой за экономию памяти.

Для борьбы с данным деградирующим эффектом мы рассмотрим применение метода наименьших квадратов (МНК, [6]) ограничиваясь при этом использованием рестартов «в чистом виде». Предположим для простоты, что рестарты повторяются периодически через одинаковое количество итераций m. Это означает, что на каждой итерации с номером $n_t = mt$, $t = 0,1,\ldots$, вектор невязки вычисляется не из рекуррентных соотношений (6), а из исходного уравнения, т. е.

$$r^{n_t} = f - Au^{n_t}, \tag{12}$$

после чего рекурсии опять используются обычным образом. Более точно такой итерационный процесс удобно описать в двухиндексных обозначениях соответствующих номеров последовательных приближенных векторов

$$u^{n_t} = u^{t,0}, \quad u^n = u^{t,k} k, \quad k = n - n_t, \quad \text{для} \quad \mathrm{n} \in \left[n_t, n_{t+1}\right].$$

Пусть нам уже известны значения «рестартовых» приближений $u^{n_0}, u^{n_1}, ..., \; n_0 = 0$. Для коррекции последнего итерационного приближения сформируем линейную комбинацию векторов

$$\hat{u}^{n_t} = u^{n_t} + b_1 v_1 + ... + b_t v_t = u^{n_t} + V_t \overline{b}, \quad \overline{b} = \left(b_1, ..., b_t\right)^T,$$
$$V_t = \left\{v_k = u^{n_k} - u^{n_{k-1}}, \quad k = 1,...,t\right\} \in R^{N_n}, \tag{13}$$

коэффициенты b_n которой будем находить из обобщенного решения переопределенной системы, получаемой после умножения уравнений (13) на исходную матрицу A:

$$W_t \overline{b} = r^{n_t} = f - Au^{n_t}, \quad W_t = AV_t. \tag{14}$$

Здесь есть несколько путей для решения СЛАУ (14): использование разложений SVD или QR для матрицы W_t, нахождение обобщенной обратной матрицы W_t^+ по формулам Гревилля, вычисление нормального решения методом наименьших квадратов (с помощью левой трансформации Гаусса, т. е. умножения общих частей уравнения (14) слева на W_t^T), или применения «облегченного» (по числу обусловленности) преобразования путем сведения к квадратной системе

$$V_t^T A V_t \overline{b} = V_t^T r^{n_t}. \quad (15)$$

Во всех этих случаях после нахождения вектора $\overline{b}$ и проведения коррекции итерационного приближения по формуле (15) следующий «рестарт» начинается с вычисления невязки

$$r^{n_{t+1}} = f - Au^{n_{t+1}}, \quad u^{n_{t+1}} = \hat{u}^{n_t}. \quad (16)$$

3.2. Методы декомпозиции областей в подпространствах Крылова

Исторически МДО возник в XIX веке как альтернирующий метод Шварца для теоретического исследования геометрически сложных краевых задач путем сведения их к простым. Если перейти от дифференциальных уравнений к сеточным, то на алгебраическом языке исходная декомпозиция области на подобласти соответствует применению блочного метода Зейделя для решения соответствующей СЛАУ. Со временем этот метод трансформировался в аддитивный блочный метод Якоби-Шварца, представляющий собой естественный путь к распараллеливанию вычислительного процесса на МВС. В настоящее время геометрическая и алгебраическая интерпретации МДО существуют равноправно, взаимно обобщая каждый из подходов.

Обращаясь к блочному представлению СЛАУ (4), мы можем записать итерационный метод Якоби-Шварца в следующем виде:

$$B_{q,q} u_q^{n+1} \equiv \left(A_{q,q} + \theta D_q \right) u_q^{n+1} = f_r + \theta D_q u_q^n - \sum\nolimits_{r \in \Omega_q} A_{q,r} u_r^n, \quad = 1, ..., P. \quad (17)$$

Здесь $\theta \in \left[0,1 \right]$ – некоторый итерационный параметр, а D_q – диагональная матрица, определяемая соотношением

$$D_q e = \sum\nolimits_{r \in \Omega_q} A_{q,r} e, \quad e = \left(1, ..., \right)^T \in R^{Nq}.$$

Переходя к геометрической интерпретации данного алгоритма, подчеркнем, что решение каждого q-го уравнения (17) означает независимое решение вспомогательной краевой задачи в подобласти Ω_q, при этом выделенные формально величины θ и D_q соответствуют использованию интерфейсных условий между контактирующими подобластями, см. подробнее [13, 14].

Итерационный процесс (17) можно представить в форме

$$u^{n+1} = u^n + B_1^{-1} \left(f - Au^n \right),$$

$$u^n = \left\{ u_q^n \right\}, \quad B_1 = \text{blok-diag} \left\{ B_{q,q} \right\}, \quad (18)$$

где B_1 является предобуславливающей матрицей данного варианта МДО, которая может использоваться в формулах (11), порождая таким образом предобусловленные методы в подпространствах Крылова.

Рассмотрим теперь возможности ускорения описанных блочных итерационных методов крыловского типа на основе дефляционного подхода, см. обзор в [14]. В данном случае, помимо традиционных вариационных и/или ортогональных свойств вычислительных последовательных приближений, на них накладываются дополнительные условия ортогональности специально вводимому m-мерному фиксированному дефляционному подпространству, ассоциированному с прямоугольной матрицей

$$V = \left(v_1 ... v_m \right) \in R^{N,m}.$$

Мы обсудим для краткости использование метода дефляции в применении к алгоритму сопряженных невязок для решения СЛАУ с симметричной матрицей A. Во-первых, здесь существует подход к оптимизации, в определенном смысле, вектора начального итерационного приближения u^0. Пусть задан произвольный вектор u^{-1}, тогда определим векторы

$$u^0 = u^{-1} + Vc, \quad r^0 = r^{-1} - AVc. \quad (19)$$

Вектор неизвестных коэффициентов $c = (c_1, \ldots, c_m)^T$ в (19) будем определять (полагая формально $r^0 = 0$) из решения переопределенной системы

$$Wc = AVc = r^{-1}. \quad (20)$$

Применяя в (20) метод наименьших квадратов (МНК), находим нормальное решение

$$c = \left(W^T W\right)^+ W^T r^{-1},$$

обеспечивающее минимальную норму вектора невязки:

$$r^0 = T_0 r^{-1}, \quad T_0 = I - W\left(W^T W\right)^{-1} W^T,$$
$$\left(r^0, r^0\right) = \left(r^{-1}, r^{-1}\right) - \left(W\left(W^T W\right)^{-1} W^T r^{-1}, r^{-1}\right) = \left(W^T W z, z\right) - \left(\left(W^T W\right)^{-1} z, z\right), z = W^T r^{-1}, \quad 21)$$

где матрица T_0 является симметричным (ортогональным) проектором, обладающим следующими свойствами:

$$T_0 = T_0^T = T_0^2, \quad W^T T_0 = T_0 W = 0,$$

т. е. пространство Span(W) принадлежит его ядру $N(T_0)$. Определяя далее начальный вектор направлений в форме

$$p^0 = r^0 - V\left(W^T W\right)^{-1} W^T A r^0 = B r^0, \quad B = I - V\left(W^T W\right)^{-1} W^T A, \quad (22)$$

для векторов r^0 и p^0 получаем дефляционные условия ортогональности

$$W^T r^0 = 0, \quad A p^0 = 0. \quad (23)$$

При этом введенная матрица B обладает следующими легко проверяемыми ортогональными свойствами:

$$W^T A B = 0, \quad B V = 0. \quad (24)$$

Дальнейшие итерации получаемого дефляционного алгоритма DCR (Deflated Conjugate Residual) осуществляются по «стандартным» формулам метода сопряженных невязок с предобуславливающей матрицей, роль которой в данном случае играет B из (22):

$$u^{n+1} = u^n + \alpha_n p^n, \quad \alpha_n = \frac{\sigma_n}{\rho_n}, \quad \rho_n = \left(A p^n, A p^n\right),$$
$$\mathrm{r}^{n+1} = r^n - \alpha_n A p^n, \quad \sigma_n = \left(A B r^n, r^n\right), \quad (25)$$
$$p^{n+1} = B r^{n+1} + \beta_n p^n, \quad \beta_n = \frac{\sigma_{n+1}}{\sigma_n}.$$

При этом на каждой итерации обеспечивается минимум нормы невязки $\left\| r^n \right\|$ в предобусловленном подпространстве Крылова

$$K_n\left(A, r^0, B\right) = Span\left(r^0, A B r^0, \ldots, \left(A B\right)^{n-1} r^0\right),$$

а вычисляемые векторы удовлетворяют условиям ортогональности

$$\left(A B r^k, r^n\right) = \sigma_n \delta_{k,n}, \quad \left(A p^k, A p^n\right) = \rho_n \delta_{k,n}, \quad k = 0,1,\ldots,n-1,$$
$$W^T r^{\,n} = 0, \qquad W^T A p^n = 0, n = 0,1,\ldots. \quad (26)$$

Отметим, что введенная предобуславливающая матрица B является вырожденной, поскольку $BW = 0$, однако соотношения (24) и (26) обеспечивают ортогональность всех векторов невязок ядру матрицы $\overline{A} = AB$, $N\left(\overline{A}\right) = Span\left(W\right)$, что обеспечивает сходимость итерационного процесса (21) – (25).

Рассмотренный дефляционный подход является достаточно универсальным и в разных вариантах исследовался под наименованиями методов агрегации, грубосеточной коррекции и малоранговых аппроксимаций матриц, см. [13]. Дефляционные предобуславливающие матрицы вида B из (22) – (26), в частности, могут использоваться эффективно совместно с B_1в мультипредобусловленном методе полусопряженных невязок (MP-SCR). В этом случае фактически мы будем иметь применение грубосеточной коррекции для ускорения МДО, а каждый из столбцов матрицы V из (19) – (22) при этом соответствует какой-то из подобластей (например, вектор v_q, где $1 \le q \le m$, может иметь единичные компоненты в сеточной подобласти Ω_q и нулевые – в остальных подобластях).

В целом можно сказать, что вариантов МДО и других предобусловленных алгоритмов в подпространствах Крылова существует очень много. Например, по методологии квазиструктурированных сеток в отдельных подобластях вспомогательные СЛАУ специального вида могут решаться одновременно либо сверхбыстрыми алгоритмами преобразования Фурье [15], либо оптимальными неявными методами переменных направлений [16], либо современными адаптивными приёмами неполной факторизации, либо малоранговыми приближениями HSS-матриц (Hierarchecial Semi-Separable, см. [17, 18]), либо перспективными «сетевыми» алгоритмами (Graph Supported), основанными на построении предобуславливающих матриц путем специальных графовых преобразований сеточных уравнений см. обзор в [19, 20]. Данные методы были инициированы результатами П. М. Вайдя (P. M. Vaidya), которые в 1991 г. он доложил на конференции IMA по теории графов и вычислениям с редкими матрицами, но нигде их не публиковал. Отметим, что такие алгоритмы перекликаются с «электротехническим» методом, идея которого восходит к эквивалентным преобразованиям «звезда-треугольник» и «треугольник-звезда», хорошо известным в электротехнике, см. [21, 22].

Наиболее классическими приемами предобуславливания СЛАУ являются многочисленные явные и неявные методы приближенной факторизации матриц, обширный анализ которых, без аспектов распараллеливания, приведен в книге [23]. Современные подходы к этим технологиям, включая перспективные аппроксимации матриц, обратных к треугольным множителям из неполных матричных разложений, приведены в работах [24] – [26] и цитируемой там литературе.

Особое место среди алгебраических решателей занимают многосеточные методы, изначально возникшие из аппроксимационных принципов и асимптотически при характерных шагах сетки $h \to 0$, обеспечивающие на достаточно гладких решениях оптимальную производительность, когда общий объем вычислений пропорционален количеству узлов сетки $O(h^d)$, где d – геометрическая размерность задачи. По различным многосеточным алгоритмам имеется огромное количество специальной литературы, среди которых выделяется алгебраическое направление (AMG – Algebraic Multi-Grid), где такая методология интерпретируется на принципах предобуславливания СЛАУ. Практическим недостатком алгоритмов AMG является большие коммуникационные потери при решении алгебраических систем высоких порядков. Для устранения этого эффекта в последние годы предложены подходы, сочетающие многосеточные методы с технологиями декомпозиции областей грубосеточной коррекции (AMG – DD), см. [27] и цитируемые там работы.

4. О технологии программирования параллельных алгоритмов

Итоговая производительность компьютерных реализаций методов решения СЛАУ очевидным образом зависит от двух основных факторов: математическая эффективность применения алгоритмов (которые мы уже обсудили в предыдущем пункте) и технологичность их отображения на архитектуру суперЭВМ, определяющая качество распараллеливания вычислительного процесса на реально существующих кластерных системах гетерогенного типа с разнородными арифметическими устройствами (универсальные CPU или специализированные ускорителями вида GPGPU, или Intel Phi), функционирующими на распределенной или на иерархической общей памяти.

В рамках двухуровневых методов декомпозиции областей в подпространствах Крылова масштабируемое распараллеливание естественным образом достигается средствами гибридного программирования. На верхнем уровне блочные методы Якоби-Шварца по подобластям с межпроцессорными обменами реализуются распределенным образом путем организации MPI-процессов, а на нижнем – синхронное решение вспомогательных СЛАУ в подобластях выполняется путем применения «внутреннего» параллелизма на многоядерных процессорах с общей памятью по технологиям многопотоковых вычислений с помощью системы Open MP. Дополнительное ускорение можно осуществить с помощью векторизации операций в системе команд AVX.

Для обеспечения высокой производительности необходимо учесть несколько технологических аспектов. Во-первых, для избегания простоя процессоров, реализующих решение различных вспомогательных СЛАУ, требуется выполнить сбалансированное разбиение области на подобласти, что в общем случае является непростой задачей. Строго говоря, все СЛАУ в подобластях должны решаться за одинаковое время, чего добиться на разнородных устройствах весьма проблематично. Даже для одинаковых вычислителей, например, равенство размерностей алгебраических систем не гарантирует синхронности их решения. Второй момент связан с сокращением коммуникационных потерь. Одна из возможностей решения задачи связана с совмещением по времени передачи данных и арифметических действий, а другая – с формированием информационных буферов и специальным управлением коммуникационными операциями. В целом следует отметить, что исследование и конструирование оптимальных вычислительных схем, в том числе параллельных, – это главным образом экспериментальная работа, успех которой в значительной степени зависит от качества планирования и оснащенности технологическими инструментариями.

При разработке математического и программного обеспечения нового поколения нельзя не иметь в виду, что в настоящее время уже имеется большое количество общедоступных библиотек и специализированных вычислительных инструментов с реализацией методов вычислительной алгебры, в том числе высокого качества и адаптированных под современные платформы суперЭВМ, которые овеществляют огромный интеллектуальный потенциал и могут быть эффективно использованы в прикладных программных комплексах.

Мировое сообщество по численным методам линейной алгебры исторически оказалось хорошо организованным и имеет свои регулярные много лет существующие журналы, традиционные конференции и активно контактирующие группы. Что особенно важно, работы здесь ведутся не только по разработке и теоретическому анализу новых методов, но и по созданию программного обеспечения, а также по экспериментальному исследованию эффективности и производительности компьютерных реализаций на современных платформах.

Наибольшие успехи достигнуты в задачах вычислительной алгебры с относительно небольшими матрицами, как плотными, так и разреженными, соответствующие библиотекам BLAS и SPARSE BLAS стали фактически стандартными и широко используются (например, в библиотеке Intel MKL, где имеется один из самых эффективных «прямых решателей» PARDISO). В данном направлении, можно сказать, принципиальные направления проблемы программного обеспечения в основаны закрыты, хотя НИРы НИОКРы непрерывно ведутся по усовершенствованию как самих алгоритмов, так и их реализаций на вновь появляющихся компьютерных архитектурах.

Что касается разработки и апробации методов для больших задач с разреженными матрицами, то здесь исследования идут в более академическом плане, поскольку они связаны с общими наукоемкими проблемами математического моделирования, включая вопросы построения сеток, аппроксимации исходных уравнений и другие, связанные с неразрывными стадиями единого вычислительного процесса. В частности, формирование подобластей для больших задач необходимо начинать на этапе генерации сеток в распределенном по различным процессорам режиме, поскольку глобальная матрица может не уместиться в памяти одного компьютерного узла.

Тем не менее, библиотеки для решения больших разреженных СЛАУ имеются в достаточно большом количестве. Из наиболее известных можно назвать SPARSE KIT, PETSc, HYPRE, см. содержательный обзор в [8], а также библиотеку LParSol [28]. Однако тенденцией последних десятилетий в разработках программного обеспечения для математического моделирования является создание интегрированных вычислительных окружений, ориентированных на согласованное участие различных групп в формировании продуктов с длительным жизненным циклом, с непрерывным развитием функциональности и с адаптацией к эволюции компьютерных архитектур. В определенной степени примерами таких разработок могут служить DUNE (Distributed Unified Numerical Environment [29]), INMOST (разработка ИВМ РАН им. Г. И. Марчука [30]) и базовая система моделирования БСМ [10], которые включают библиотеки алгебраических решателей.

5. Технические требования к созданию ИВО

Обзор классов задач и методов их решения свидетельствует, что создание математического и программного обеспечения, претендующего на роль интегрированного вычислительного окружения для широкой предметной области – это большая комплексная проблема, требующая вовлечения значительных инвестиций и специалистов высокой квалификации с различной профессиональной подготовкой. В силу этого такой наукоёмкий и ресурсоёмкий проект должен быть спланирован на серьёзной методической основе, которую можно сформулировать следующим перечнем технических требований к содержанию научно-исследовательских и опытно-конструкторских работ (НИОКР).

Широкая функциональность с непрерывной поддержкой современного научно-технологического уровня. В нашем конкретном случае это означает гибкое расширение состава решаемых СЛАУ с различными спектральными, вариационными, структурными и другими свойствами, а также применяемых методов, адаптируемых к особенностям алгебраических систем, информационным способам их представления и архитектурным спецификациям используемых МВС. В частности, избыточная функциональность должна в целом обеспечить возможность выбора наилучшего алгоритма из состава библиотеки для решения конкретной задачи и достичь трудно совместимых качеств эффективности и универсальности.

Высокая производительность кода с масштабируемым распараллеливанием вычислений средствами гибридного программирования, без формальных ограничений на число степеней свободы реализуемых задач и алгоритмов, а также на количество используемых вычислительных узлов, ядер и других устройств гетерогенных суперЭВМ с распределенной и иерархической общей памятью.

Адаптируемость к эволюции компьютерных платформ и многоверсионность функционального наполнения библиотеки, обеспечиваемые современными компонентными технологиями и согласованием внешних и внутренних рабочих интерфейсов программно-инструментального ядра.

Универсальные и конвертируемые структуры данных, согласованные с имеющимися распространенными форматами и поддерживающие возможности переиспользования внешних программных продуктов, представляющих большой интеллектуальный потенциал.

Многоязыковость и кросс-платформенность программного функционального наполнения, возможность использования различных стилей коллективной деятельности и открытость к согласованному участию в проекте различных групп разработчиков.

Наличие интеллектуальных пользовательских и внутренних дружественных интерфейсов с ориентацией на широкое применение и активную востребованность в многообразных производственных сферах специалистами с различной профессиональной подготовкой. В частности, здесь следует различать поддержку эффективной деятельности математиков – разработчиков новых алгоритмов, информационных технологов (включая экспертов по распараллеливанию), а также конечных пользователей, для которых идеальным является минимум обязательных сведений о внутренней «кухне» ИВО.

Рассматриваемые архитектурные принципы ориентированы на длительный жизненный цикл разрабатываемого продукта, а также на высокую производительность программистского труда участников данного наукоёмкого проекта. При этом нацеленность на суперкомпьютерные вычисления означает не только быстродействие алгоритмов, но и интеллектуальность инструментальной среды, и квалифицированную работу с большими данными во избежание коммуникационных потерь.

Естественно, что системная инфраструктура ИВО должна включать следующие библиотечные инструментарии, необходимые для активного развития, сопровождения и эффективной эксплуатации производственного программного продукта:

– средства автоматизированного тестирования и сравнительного анализа эффективности алгоритмов на характерных СЛАУ из методических и практических задач, включая международные матричные коллекции MATRIX MARKET, FLORIDA, BOEING и т. д., а также генераторы матриц для характерных приложений;

– пользовательская документация с описаниями исходных данных и особенностей применения алгоритмов библиотеки, включая примеры (EXAMPLEs) запуска решателей и рекомендации по их выбору для конкретных типов СЛАУ, а также архивы вычислительных сценариев с результатами расчетов;

– инструменты конфигурационного управления и поддержки многоверсионности функционального ядра библиотеки, обеспечивающие гибкое расширение состава вычислительных модулей и их адаптации к компьютерным платформам;

– средства формирования интерфейсов с разработчиками, конечными пользователями и внешними программными продуктами, а также комплексирования формируемых библиотечных модулей с прикладными программными комплексами;

– постоянно обновляемая база знаний по методам и технологиям вычислительной алгебры, содержащая актуальную информацию с когнитивным анализом и целевым поиском по научным публикациям, программным реализациям и по комплектам тестовых примеров; некоторыми прототипами таких разработок, в основном информационного типа, являются интеграционный проект СО РАН «Дерево математики» (www.mathtree.ru) и проект МГУ ALGO WIKI (www.parallel.ru).

Необходимо иметь в виду, что на этапах разработки, верификации, валидации и тестирования вычислительных библиотечных модулей, а также в периоды опытной или производственной эксплуатации программные коды находятся в разных ипостасях и требуют своих способов поддержки. В зависимости от различных путей организации технологического процесса могут быть определены разные стадии готовности продукта, от пробного или опытного варианта (α -версии) до конечного производственного изделия с качественными эксплуатационными показателями. Обширный спектр особенностей рассматриваемых разработок – от требований современного мирового научного уровня до высокой технологичности – обуславливает “вертикаль” квалификации участников проекта – от экспертов академического уровня до специалистов по суперомпьютерным информационным технологиям.

Особо следует выделить важность создания учебных версий разрабатываемого ИВО, сопровождаемых когнитивными методиками и соответствующими материалами, ориентированными как на студентов с университетскими преподавателями, так и на слушателей курсов повышения компьютерной квалификации.

6. Заключение

Главный результат данной работы заключается в конкретизации концептуальных положений в создании математического и программного обеспечения нового поколения для высокопроизводительного решения широкого класса задач вычислительной алгебры в рамках интегрированного вычислительного окружения, ориентированного на длительный жизненный цикл с непрерывным развитием функционального ядра и с адаптацией к эволюции компьютерных архитектур, а также рассчитанные на согласованное участие различных групп разработчиков, на активное использование внешних программных продуктов и на массовую востребованность конечными пользователями из различных производственных сфер. Системное наполнение такого проекта призвано значительно повысить производительность труда программистов за счет интеллектуальных средств автоматизации построения параллельных алгоритмов и их отображения на архитектуру суперЭВМ. Инструментальная среда рассматриваемого ИВО должна, на промышленном языке, обеспечить переход к производству средств производства прикладных программных комплексов как основы предсказательного математического моделирования.

Работа *поддержана грантами РФФИ № 16-29-15122 офи-м и № 18-01-00295*

Литература

1. Saad Y. Iterative methods for sparse linear systems // SIAM. – 2003.

2. Ильин В. П. Методы и технологии конечных элементов. Новосибирск: ИВМиМГ СО РАН, 2007.

3. Dolean V., Jolivet P., Nataf F. An introduction to domain decomposition methods: algorithms, theory and parallel implementation // SIAM. – 2015.

4. Domain Decomposition Methods [Electronic resource]. Mode of access: http://ddm.org.

5. Shapira Y. Matrix-based Multigrid: Theory and Application. – NY: Springer Science & Business Media, 2008.

6. Ильин В. П. Двухуровневые методы наименьших квадратов в подпространствах Крылова // Записки научных семинаров ПОМИ РАН. – 2017. – Т. 463. – С. 224-239.

7. Il'in V. P. High-Performance Computation of Initial Boundary Value Problems Springer Nature Switzerland AG 2018 / L. Sokolinsky and M. Zymbler (Eds.): PCT 2018, CCIS 910, 2018, P. 186 – 199.

8. Dongarra J. List of freely available software for linear algebra on the web. 2006. [Electronic resource]. Mode of access: http://netlib.org/utk/people/JackDongarra/la-sw.html.

9. Бутюгин Д. С., Гурьева Я. Л., Ильин В. П., Перевозкин Д. В., Петухов А. В., Скопин И. Н. Библиотека параллельных алгебраических решателей Krylov // Труды конференции ПАВТ-2013, Челябинск: Изд-во ЮУрГУ, 2013. – С. 76 – 86.

10. Ильин В. П., Гладких В. С. Базовая система моделирования (БСМ): концепция, архитектура и методология // Современные проблемы математического моделирования, обработки изображений и параллельных вычислений 2017 (СПММОИиПВ-2017). Ростов-на-Дону: Изд-во ДГТУ, 2017. – С. 151 – 158.

11. Intel Math Kernel Library. Reference Manual. [Electronic resource]. Mode of access: https://software.intel.com/en-us/articles/mkl-reference-manual.

12. Ильин В.П. Математическое моделирование. Часть 1. Непрерывные и дискретные модели, Новосибирск: изд-во СО РАН, 2017.

13. Il'in V. P. Multi-preconditioned domain decomposition methods in the Krylov subspaces // LNCS. – 2017. – 10187. – P. 95 – 106.

14. Gurieva Y., Il`in V. P. On parallel computational technologies of augmented domain decomposition methods // 13th International Conference PaCT Parallel Computing Technologies. Petrozavodsk, Russia, 2015. – P. 33 – 46.

15. Гасенко В. Г. Дифференциальный метод Фурье // Сиб. ж. инд. мат. т. – 2017. – 20, № 1. – С. 21 – 30.

16. Горбенко Н. И., Ильин В. П. Аддитивный метод Писмана-Речфорда // Записки научных семинаров ПОМИ РАН. – 2015. – Т. 439. – С. 47 – 58.

17. Solovyev S. Multifrontal hierarchically solver for 3D discretized elliptic equation in finite difference methods // Theory and Applications / edited by I.Dimov, Farago and L.Vukov. Springer International Publishing, 2015. – P. 371-378.

18. Chaillat S., Desiderio L., Ciarlet P. Theory and implementation of – matrix based iterative and direct solvers for Helmholtz and electrodynamic oscillatory kernels // Journal of Computational Physics. – 2017. – 351. – P. 165 – 186.

19. Bern M., Gilbert J. R., Hendrickson B., Nguyen N., Toledo S. Support-Graph Preconditioners // SIAM J. Matrix Anal. Appl. – 2006. – V. 27, no 4. – P. 930 – 951.

20. Spielman D. A., Teng S.-H. Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems // SIAM J. Matrix Anal. Appl. – 2014. – V. 35, no 3. – P. 835 – 885.

21. Ильин В. П., Сандер С. А. Электротехнический метод решения систем сеточных уравнений. Новосибирск: Автометрия. – 1987. – № 4. – С. 52 – 59.

22. Ильин В. П., Нефедов В. Л. Метод диагональных переносов (TS-алгоритм) для решения пятиточечных уравнений // ЖВМиМФ. – 2001. – Т. 41, № 2, – С. 186 – 199.

23. Ильин В. П. Методы неполной факторизации для решения алгебраических систем. – М.: Наука, 1995.

24. Капорин И. Е. , Милюкова О.Ю. MPI+OPENMP параллельная реализация метода сопряженных градиентов с некоторыми явными предобуславливателями: препринт N 8. – М. ИПМ РАН, 2018.

25. Anzt H., Huckle T. K., Brackle J., Dongarra J. Incomplete sparse approximate inverses for parallel preconditioning // Parallel Computing. – 2018. – V. 71, – P. 1 – 22.

26. Konshin I. E., Olshanski M. A., Vassilevski Y. V. LU factorizations and ILU preconditioning for stabilized disctretizations of incompressible Navier-Stokes equations: preprint N 49. Houston, UH, 2016.

27. Bank R., Falgout R. D., Jones T., Manteuffel T. A., McCormick S. F., Ruge J. W. Algebraic multigrid domain and range decomposition (AMG-DD/AMG-RG) // SIAM J.Sci Comput. – 2015. – 37. – P. 113 – 136.

28. Бартенев Ю. Г., Ерзунов В. А., Карпов А. П., Петров Д. А., Пищулин И. А., Стаканов А. П., Щаникова Е. Б., Капорин И. Е., Милюкова О. Ю., Харченко С. А., Коньшин И. Н., Мееров И. Б., Сысоев А. В. Комплекс библиотек параллельных решателей СЛАУ LPARSOL версии 3 // XV Международная конференция «Супервычисления и математическое моделирование»: сб. науч. тр. / под ред. Р.М. Шагалиева. – Саров: ФГУП «РФЯЦ-ВНИИЭФ», 2014. – С. 102 – 110.

29. DUNE.URL. [Electronic resource]. Mode of access: http://www.dune-project.org. Cited 15 Jan 2016.

30. INMOST. http://www.inmost.org. [Electronic resource]. Mode of access: Cited 10. Feb. 2017.

ON THE INTEGRATED COMPUTIONAL ENVIRONMENT FOR HIGH-PERFORMANCE SLAE SOLUTIONS

V. P. Ilyin

Institute of Computational Mathematics and Mathematical Geophysics,
Siberian Branch of the Russian Academy of Sciences, Novosibirsk
Novosibirsk State University, Novosibirsk

The concept and general structure of the integrated computional environment (IVO) intended for high-performance solutions of a wide class of systems of linear algebraic equations (SLAE) arising from finite volume or finite-element approximations of multidimensional boundary value roblems on unstructured grids is considered . Parallelization of algorithms is carried out based on two-level methods of decomposition of subdomains in Krylov subspaces with parametrized overlapping of subdomains and different types of interface conditions, using different preconditioners and accelerating algorithms of coarse-grid correction, deflation and low-rank approximation of the original matrices represented by compressed-sparse algebraic data structures (ADS). For large SLAE (10^9) and more equations), the ADS is formed in the form distributed over computational processors (MPI-processes) for the subsequent parallel implementation of iterative methods. The synchronous solution of auxiliary linear equations in subdomains is performed using a multi-thread computation, operations vectorization and graphics accelerators GPGPU. The developed algorithms are implemented as part of the KRYLOV software library, which is an IVO prototype and provides a long life cycle with a flexible extension of the types of tasks to be solved and the composition of algorithms adapted to the evolution of supercomputer platforms, with multilingual modules, with the reusing of external software products and with the possibility of coordinated participation in the project of various groups of developers.

Keywords: algebraic systems, sparse matrices, data structures, preconditioned iterative methods, Krylov subspaces, parallel algorithms, integrated computational environment.