

РЕШЕНИЕ ПЛОХООБУСЛОВЛЕННЫХ СИММЕТРИЧНЫХ СЛАУ ДЛЯ ЗАДАЧ СТРОИТЕЛЬНОЙ МЕХАНИКИ ПАРАЛЛЕЛЬНЫМИ ИТЕРАЦИОННЫМИ МЕТОДАМИ

В.Л. Якушев, В.Н. Симбиркин, А.В. Филимонов, П.А. Новиков, И.Н. Коньшин, Г.Б. Сушко, С.А. Харченко

Введение

В России развернута работа по реализации крупномасштабных строительных проектов. К их числу относится, например, строительство комплекса «Москва-Сити», высотных зданий в Москве, Санкт-Петербурге, Сочи, Новороссийске и других крупных городах, а также уникальных спортивных сооружений (Олимпийские объекты в Сочи, новый футбольный стадион на Крестовском острове в Санкт-Петербурге и др.). Кроме того, за последние годы произошел ряд аварий и катастроф с разрушением строительных объектов, например, Аквапарка и Басманного рынка в Москве, машинного зала Саяно-Шушенской ГЭС и др. Это способствовало ужесточению требований к расчетным моделям сооружений и точности их реализации численными методами.

В ходе научно-исследовательских и проектных работ возникла необходимость существенного развития вычислительных средств и информационных технологий для обеспечения надежности и экономичности строительства и эксплуатации указанных объектов. Сейчас предъявляются всё большие требования к подробности конечно-элементных моделей сооружений, и, следовательно, растет размерность моделей. Актуальной проблемой также является достижение высокой скорости решения задач.

Следует заметить, что проектирование многих крупных строительных объектов в России уже сейчас выполняется проектно-строительными фирмами Европы, США и других стран. Эти фирмы имеют соответствующий опыт, в том числе в применении программных продуктов и вычислительных систем большой мощности. К сожалению, в России разработки CAE- и CAD-систем для автоматизации инженерных расчетов и конструирования в строительстве никем не финансируются, разработка программного обеспечения ведется небольшими фирмами за счет средств, получаемых, в основном, от другой коммерческой деятельности. В дальнейшем отсутствие отечественных исследований и разработок в данном направлении приведет к тотальному господству в инженерной практике западных программных продуктов и технологий, как это уже произошло в ряде отраслей.

Разработка отечественного CAE-продукта, в том числе для мощных ЭВМ с параллельной архитектурой, соответствует современным тенденциям развития автоматизации исследований и проектирования в строительстве. При этом могут быть более полно реализованы перспективные отечественные подходы к расчету строительных конструкций с использованием единых трехмерных моделей всех несущих конструкций сооружения и основания, учитывающих пространственный характер воздействий, физическую и геометрическую нелинейность, технологию и последовательность возведения конструкций и другие важные факторы.

Данная статья является результатом совместной работы нескольких коллективов по созданию эффективных итерационных методов решения плохообусловленных симметричных систем линейных алгебраических уравнений (СЛАУ). Для генерации входных данных был использован программный комплекс STARK ES, разработанный в ООО «Еврософт» (Москва, Россия) и предназначенный для расчета пространственных строительных конструкций на прочность, устойчивость и колебания [1,2]. При проведении численных экспериментов и анализе эффективности использовались алгоритмы и программы, разработанные ООО «ТЕСИС» (Москва, Россия).

Моделирование задач строительной механики.

Для моделирования плосконапряженного/плоскодеформированного и изгибного напряженного состояния используются плоские четырехугольные и треугольные конечные элементы. При моделировании плосконапряженного/плоскодеформированного состояния в каждом узле используются три степени свободы в плоскости элемента: две поступательные степени свободы и одна вращательная. Используются три типа конечных элементов: две разновидности гибридных конечных элементов и один вид элементов метода перемещений. При построении гибридных элементов использован функционал Рейсснера, а элементов метода перемещений — функционал Лагранжа. Первый тип гибридного элемента получен с использованием квадратичной аппроксимации напряжений по площади элемента (для четырехугольных элементов) или линейной аппроксимации напряжений по площади элемента (для треугольных элементов и квадратичной аппроксимации перемещений по границам элемента (схема Олмана для перемещений)). Второй тип гибридного элемента получен с использованием квадратичной аппроксимации напряжений по площади элемента (для четырехугольных элементов) или линейной аппроксимации напряжений по площади элемента (для треугольных элементов) и кубической аппроксимации перемещений по границам элемента (схема Пиана). Элементы метода перемещений получены по схеме Олмана. Для моделирования изгибного напряженного состояния также используются три типа конечных элементов. При построении гибридных элементов использован функционал

Рейсснера, а для элементов метода перемещений — функционал Лагранжа. Гибридные элементы первого типа строятся на основе теории толстых плит Миндлина-Рейсснера и имеют в узле три степени свободы: поперечное перемещение и два угла поворота нормали к срединной плоскости. В изгибном элементе поперечное перемещение на границе элемента аппроксимируется полиномом второго порядка, а углы поворота нормали - полиномом первого порядка. Использована квадратичная аппроксимация напряжений по площади элемента (для четырехугольных элементов и линейная аппроксимация напряжений по площади элемента (для треугольных элементов). Эти элементы свободны от эффекта «сдвигового заперения» и могут использоваться для расчета толстых и тонких плит. Гибридные элементы второго типа получены с использованием квадратичной аппроксимации напряжений по площади элемента (для четырехугольных элементов) или линейной аппроксимации напряжений по площади элемента (для треугольных элементов) и кубической аппроксимации поперечных перемещений по границам элемента (схема Пиана). Элементы метода перемещений получены по схеме Батоша на основе теории толстых плит Миндлина-Рейсснера [3].

Для моделирования напряженного состояния массивных тел используются современные объемные конечные элементы. Используются три типа объемных конечных элементов: два типа гибридных конечных элементов и один тип элементов метода перемещений. При построении гибридных элементов используется функционал Рейсснера, а элементов метода перемещений — функционал Лагранжа. Гибридные элементы первого типа являются равновесными, а второго типа неравновесными. Элементы метода перемещений являются изопараметрическими. Все типы элементов имеют в узле три степени свободы. Равновесные гибридные элементы свободны от эффекта «заперения по толщине», обладают высокой точностью по напряжениям и перемещениям и могут использоваться для расчета как толстых, так и тонких тел [3].

Для дальнейшего анализа были выбраны гибридные элементы первого типа.

Описание расчетных моделей.

Для анализа и проведения экспериментов отобран ряд конечно-элементных моделей проектируемых строительных объектов из практики института ЦНИИСК им. В.А. Кучеренко. Основные характеристики расчетных моделей сооружений представлены в таблице 1, а на рисунке 1 приведен общий вид некоторых из них.

Таблица 1. Описание моделей строительных конструкций.

№ п/п	Наименование объекта	Кол-во узлов	Кол-во конечных элементов	Кол-во степеней свободы	Кол-во ненулевых элементов в матрице жесткости
1	Оболочка покрытия 2-й сцены Мариинского театра, С.-Петербург	6670	22773	40020	871093
2	Спортивно-досуговый центр	14797	15801	88782	1189954
3	Нефтедобывающая платформа	17312	28623	98650	13842114
4	Горнолыжный спуск «Воробьевы Горы», Москва	18760	34822	112560	2826091
5	Высотное здание комплекса «Новосити», Новороссийск	32626	33612	195024	2736424
6	22-эт. здание по ул. Параллельная, Сочи	41257	48267	247542	3494364
7	Жилой комплекс «Королевский парк», Сочи	42762	62408	256572	5470229
8	Модельная задача Куб 50x50x50	132 651	125 000	397950	13842114
9	Высотное здание по ул. Мосфильмовской (жилой комплекс Воробьевы горы)	71169	88377	427014	4327831
10	Высотное здание «Москва-Сити», Н=380 м	125 932	178 059	755592	11228013

11	Стадион «Газпром-Арена», С.-Петербург, модель I	159 686	230 386	798230	15566477
12	Большая ледовая арена, Сочи	148 315	172 285	889890	16035641
13	Высотное здание 1	270 695	284 000	1 623 600	22203726
14	Стадион «Газпром-Арена», С.-Петербург, модель II (см. рис. 1)	426 071	577 500	2 534 446	49500433
15	Высотное здание 2	541 295	568 000	3 247 200	44404241
16	Высотное здание 3	811 895	852 000	4 870 800	66764339
17	Высотное здание 4	1 339 015	1 403 570	8 033 520	110 021 813

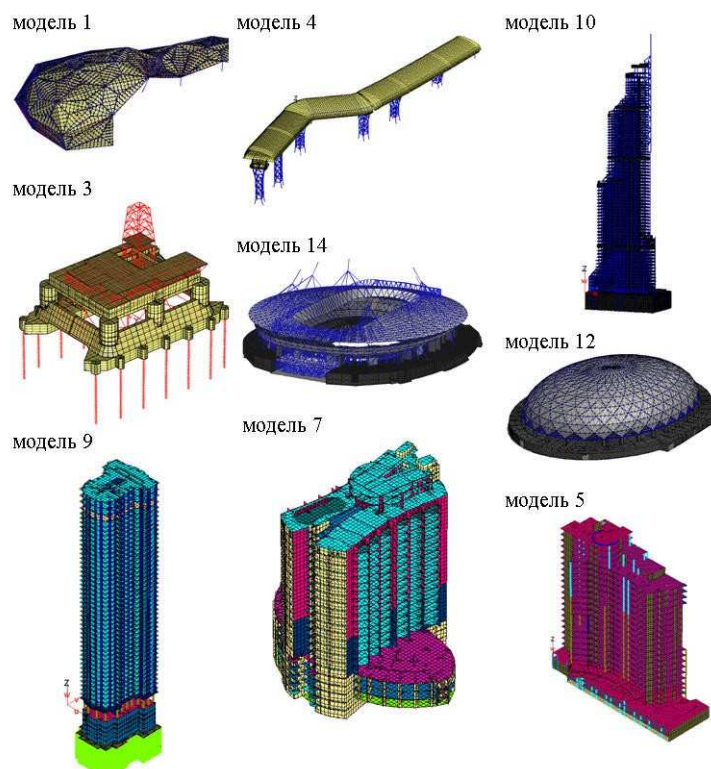


Рис. 1. Общий вид конечно-элементных моделей сооружений

Особенности СЛАУ задач строительной механики.

Эффективное решение систем линейных уравнений, возникающих в задачах строительной механики, предполагает использование самой современной вычислительной техники, поскольку число неизвестных в задачах быстро растет и уже сейчас доходит до нескольких десятков миллионов. Среди других различных типов систем линейных уравнений эти задачи выделяются гигантской обусловленностью ($10^9 \div 10^{13}$) даже при относительно небольшой размерности, что привело к тому, что при их решении почти всегда используются прямые методы. При попытках использования итерационных методов, как правило, возникают существенные проблемы со сходимостью алгоритмов. С другой стороны, основная общемировая тенденция развития алгоритмов решения СЛАУ состоит в использовании итерационных алгоритмов, в том числе по причине их высокой эффективности на самой современной вычислительной технике для широкого круга приложений. В данной работе сделана попытка применить существующие в ООО «ТЕСИС» разработки в области итерационных алгоритмов при решении СЛАУ задач строительной механики.

Шаблонный параллельный итерационный алгоритм решения СЛАУ.

Комбинированный режим параллельных вычислений.

Современные суперкомпьютерные вычислительные системы как правило имеют неоднородную архитектуру. С одной стороны, имеется набор вычислительных узлов с распределенной памятью, обмен данными между которыми может быть осуществлен по быстрой обменной сети. С другой стороны, каждый узел представляет собой многопроцессорный/многоядерный компьютер с общим доступом к оперативной памяти. Программная реализация вычислительных алгоритмов на компьютерах подобной архитектуры предполагает [4]

использование стандарта MPI при распараллеливании по распределенной памяти (по узлам вычислений), а по общей памяти узла распараллеливание по процессорам/ядрам естественно осуществлять на основе стандартов работы с потоками с встроенными механизмами динамической балансировки нагрузки, таких как OpenMP или Intel® Threading Building Blocks (ТВВ). При этом предполагается (Рисунок 2), что на каждом узле имеется только один MPI процесс, который порождает на этом узле нужное количество одновременно работающих потоков вычислений.

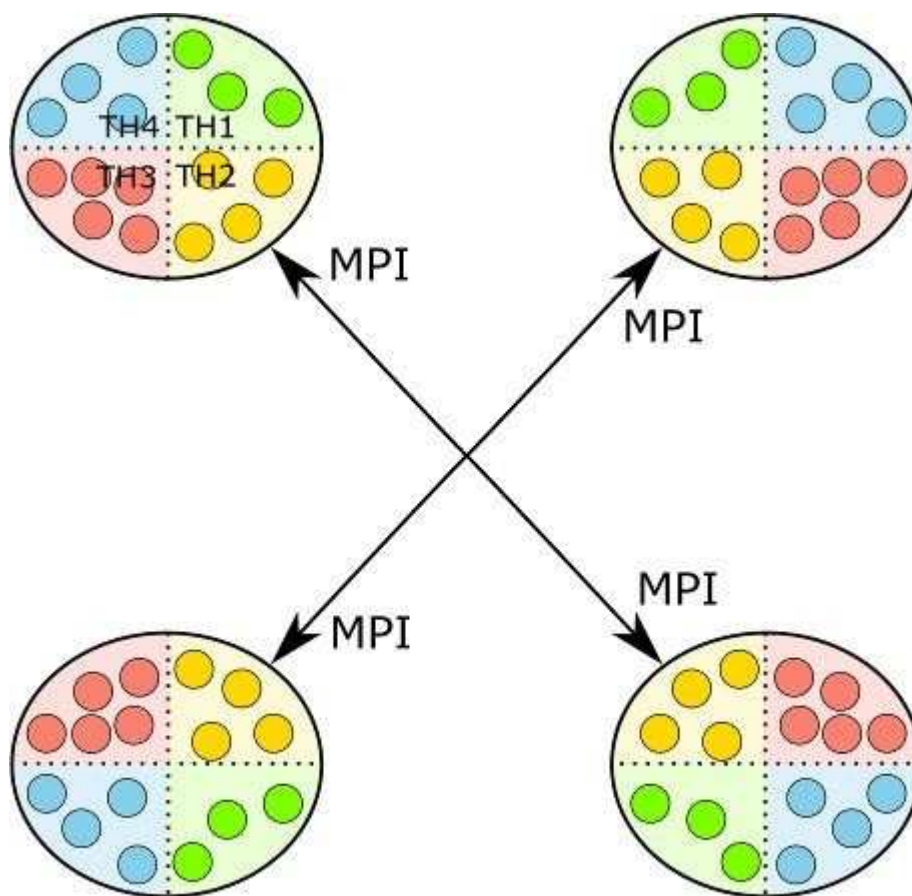


Рис. 2. Комбинированная MPI+threads организация параллельных вычислений

Шаблонная реализация, типы данных, мелко-блочные вычисления.

Многообразие приложений алгоритмов решения СЛАУ диктует различные форматы представления данных. В том числе, необходимо решать задачи с симметричными и несимметричными матрицами, вещественными и комплексными, с постоянным размером блока мелко-блочного биения (1, 3, 5, и др.), решать плохообусловленные задачи со сверхвысокой точностью, и т.д. Единая реализация алгоритмов решения СЛАУ для достаточно широкого многообразия типов данных обеспечивается шаблонной реализацией всех алгоритмов в терминах шаблонов C++. В частности, поддерживаются:

- а) три типа данных (short, int, long long) для хранения целочисленных массивов;
- б) два основных типа данных для хранения вещественных чисел (float, double) при хранении данных матрицы и предобусловливания;
- в) вещественные данные матрицы и предобусловливания могут храниться в блочном формате $k \times k$, где k – постоянный размер блока;
- г) для поддержки несимметричных неполных треугольных разложений используется двоякий формат хранения блоков мелко-блочного биения;
- д) поддерживается 4 типа вещественных векторных данных (double, double long, quad и octal), два последних типа данных — эмуляция четверной и восьмерной точности с помощью библиотеки QD [5].

Предобусловливание на основе неполных разложений второго порядка точности.

В работе [6] предложен алгоритм построения предобусловливания второго порядка точности, для которого имеют место соотношения:

$$A + E = L^{(I)} U^{(I)} + L^{(I)} U^{(II)} + L^{(II)} U^{(I)} \quad (1)$$

где A – матрица коэффициентов системы уравнений, $L^{(I)}$ и $U^{(I)}$ соответственно нижняя и верхняя треугольная части предобуславливания (элементы первого порядка точности), $L^{(II)}$ и $U^{(II)}$, соответственно, нижняя и верхняя треугольная части элементов второго порядка точности предобуславливания, и E – матрица ошибок. Предобусловленные итерации производятся с матрицей $A(L^{(I)}U^{(I)})^{-1}$. В процессе неполной факторизации в памяти одновременно хранятся только все элементы матриц A , $L^{(I)}$ и $U^{(I)}$ и некоторая небольшая часть элементов матриц $L^{(II)}$ и $U^{(II)}$. Элементы матрицы ошибок E отбрасываются с корректировкой соответствующих диагональных значений. В работе [6] приводятся некоторые теоретические оценки качества предобуславливания в симметричном случае. Этот алгоритм показал высокую эффективность и надёжность при последовательном решении широкого класса задач с плохо обусловленными матрицами коэффициентов. В случае мелко-блочных вычислений операции с элементами заменяются на соответствующие операции с блоками.

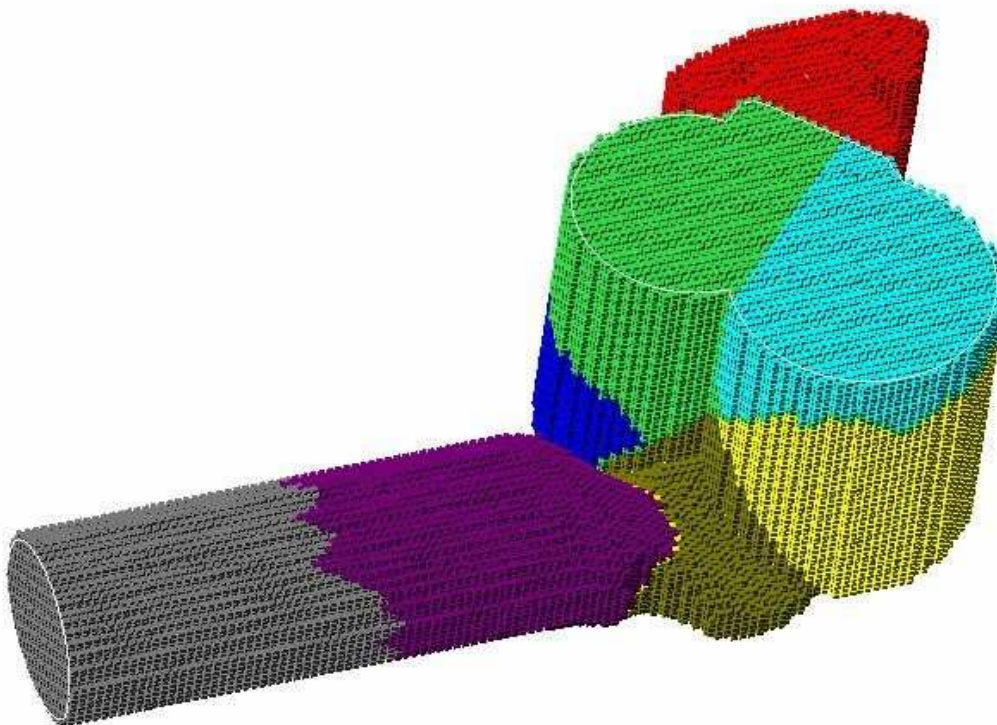


Рис. 3. Геометрическая объемная декомпозиция области

Параллельный алгоритм построения неполных треугольных разложений на основе геометрической редукции.

Основой одного из возможных способов распараллеливания алгоритма построения предобуславливания типа (1) является так называемая геометрическая редукция задачи [7]. Её смысл состоит в следующем:

1. Строится геометрическая объёмная декомпозиция области на p частей (Рис. 3);
2. В объемах выделяются строки, в столбцах которых нет элементов из других объемов, т.е. строится расщепление вида

$$A = \begin{bmatrix} A_1 & & 0 & C_1 \\ & \ddots & & \vdots \\ 0 & & A_p & C_p \\ B_1 & \cdots & B_p & D \end{bmatrix}; \quad (2)$$

3. Для матрицы дополнения по Шуру $S = D - \sum_{i=1}^p B_i A_i^{-1} C_i$ строится ее структурное приближение

вида $Sp(\tilde{D}) = Sp(D) + \sum_{i=1}^p Sp(B_i) * Sp(Z_i)$, где $Sp(Z_i) = Sp[(A_i)^{NCYCLE} C_i]$, символ Sp обозначает

равенство по структуре, $NCYCLE \geq 1$;

4. В строках матрицы D выделим наборы строк $M_{i,j}$, $1 \leq i \leq p$, $1 \leq j \leq p$, $i < j$ такие, что в каждой строке из набора $M_{i,j}$ в столбце есть элементы как из i -й, так и из j -й подобласти исходной геометрической декомпозиции области, а элементы из других исходных объемов отсутствуют. Множества ячеек из наборов $M_{i,j}$ имеют в 3D случае квази размерность 2;

5. По подматрице $Sp(\tilde{D}_{i,j})$ матрицы $Sp(\tilde{D})$ для строк и столбцов из множества $M_{i,j}$ проводится геометрическая декомпозиция на две части, с выделением границы между ними (Рисунок 4). Граница раздела имеет квази размерность 1 – это некоторая квази-кривая в 3D случае;

6. Аналогично в строках матрицы D выделяются тройные наборы строк $M_{i,j,k}$, квази-кривые в 3D случае, для которых проводится декомпозиция на 3 части;

7. Каждое множество из больше чем 3 соседних объемных блоков в строке рассматривается как целое — квази точка в 3D случае.

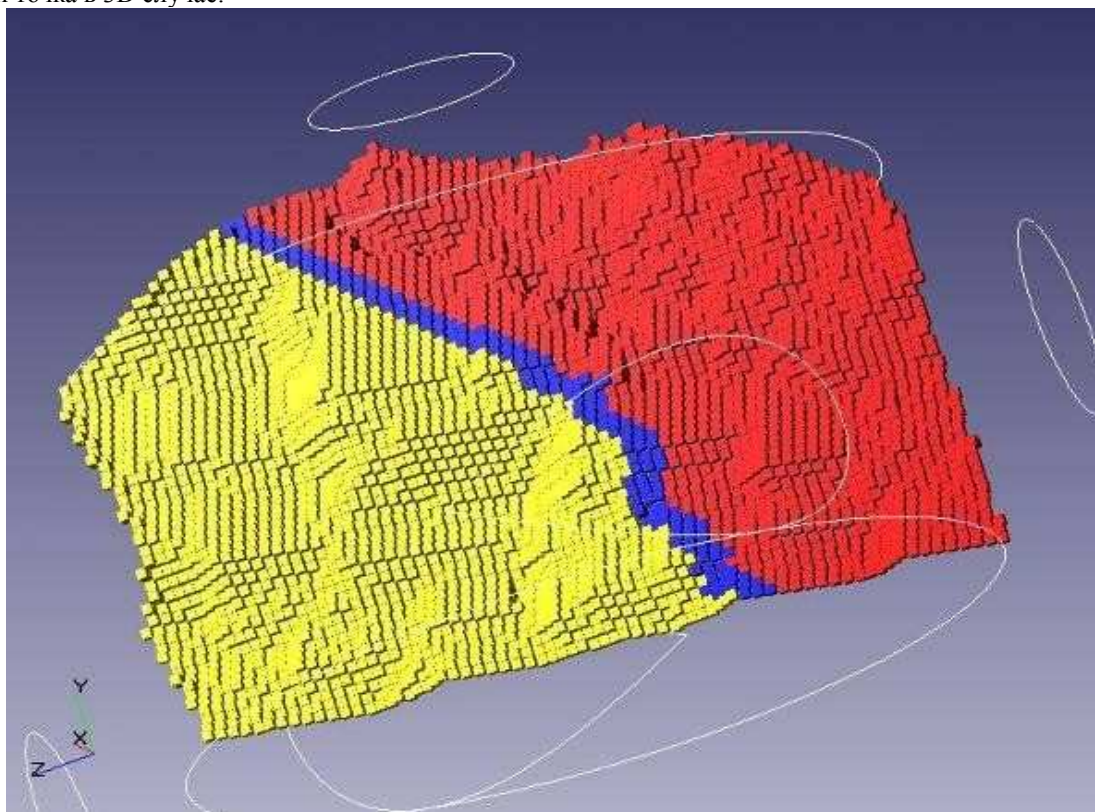


Рис. 4. Геометрическая декомпозиция квази поверхности

Таким образом строится некоторое блочное биение задачи. Упорядочивание блоков осуществляется следующим образом: сначала все объемы, затем все квази поверхности, затем все квази кривые, затем все квази точки. Распределение по процессорам осуществляется с учетом геометрической локальности: объемы — каждый своему процессору, квази-поверхности — каждая из частей одному из процессоров в примыкающих объемах, квази-кривые — каждая из частей одному из трех примыкающих процессоров, и т.д. Параллельный алгоритм построения предобуславливателя строится на основе геометрической редукции как «блочный по позициям» с неполным разложением «по значениям» второго порядка точности внутри каждого блока.

Итерационный алгоритм SOFGMRES(m). Рассмотрим систему линейных уравнений вида

$$Ax = b, \quad (3)$$

где $A \in \mathbb{R}^{N \times N}$ – заданная невырожденная, возможно незнакоопределенная несимметричная матрица, $b \in \mathbb{R}^N$ – заданный вектор правой части, $x \in \mathbb{R}^N$ – неизвестный вектор. В дальнейшем будем предполагать, что предобуславливатель и ненулевое начальное приближение, если таковые имеются, уже учтены соответствующим образом в A и b .

Основой алгоритма SOFGMRES(m) являются матричные соотношения [8] вида:

$$\begin{cases} AY_k = W_k R_k, \\ Y_k^T Y_k = I_k, \\ W_k^T W_k = I_k, \end{cases} \quad (4)$$

где $Y_k \in \mathbb{R}^{N \times k}$, $W_k \in \mathbb{R}^{N \times k}$, $R_k \in \mathbb{R}^{k \times k}$, матрица R_k – верхняя треугольная. Соотношения (4) будем называть матричными соотношениями в QR форме. В процессе итераций матричные соотношения (4) достраиваются, строится новое приближение к решению, затем построенные расширенные матричные соотношения фильтруются, новые направления пополняют начальное подпространство, и т.д. Оказывается, что если начальное подпространство и фильтрации отсутствуют, то при некоторых необременительных предположениях алгоритм SOFGMRES(m) математически эквивалентен алгоритму GMRES без рестартов. Оказывается также, что сходимость алгоритма SOFGMRES(m) определяется монотонной характеристикой подпространства – *обобщенным обратным числом обусловленности предобусловленной матрицы на подпространстве*. В симметричном положительно определенном случае если подпространство пусто, эта характеристика совпадает с обратным числом обусловленности предобусловленной матрицы в обычном смысле. Кроме того, эта характеристика в общем случае принимает значения в интервале $[0,1]$, монотонно не убывает при расширении подпространства, и чем она ближе к 1, тем быстрее сходимость алгоритма SOFGMRES(m) к решению системы уравнений (3).

Существует множество способов построения начального подпространства: а) итерационный, алгоритм SOFGMRES(m); б) на основе априорной информации; в) последовательным уточнением подпространства при решении систем уравнений с последовательно возникающими правыми частями; г) через предысторию вычислений – финальное подпространство для предыдущей системы уравнений можно свести к набору полей моделируемых физических величин, а затем на этой основе построить начальное подпространство для следующей системы линейных уравнений, в том числе и при небольших изменениях расчетной области.

Высокоточное решение плохообусловленных СЛАУ. Идея высокоточного решения плохообусловленных СЛАУ состоит в использовании повышенной точности для небольшой части вычислений. Высокоточные вычисления при решении плохообусловленных СЛАУ проводятся на основе программной эмуляции в пакете QD [5]. Основная идея этого пакета состоит в том, чтобы хранить и проводить вычисления с «четверной» double-double и «восьмерной» quad-double точностью как с не вычисленной явным образом суммой двух или четырех вещественных чисел двойной точности. К сожалению, явное использование программной эмуляции повышенной точности во всех вычислениях является слишком дорогой операцией (~10-кратное замедление для double-double и ~100-кратное замедление для quad-double), поэтому для эффективности высокоточных алгоритмов необходимо минимизировать вычисления с кратной точностью.

Подобная методика подробно рассматривается в работе [9]. В этой работе описывается двухуровневый алгоритм, в котором повышенная точность используется только на верхнем уровне итерационной схемы, на котором вычисляется текущая невязка в высокой точности. Далее, невязка преобразуется в двойную точность, и с ней, как с обычной правой частью, с сохранением подпространства на входе/выходе, проводятся вычисления в double для нахождения в double нового пробного вектора решения. Этот пробный вектор преобразуется назад в высокую точность, строится новое приближение к решению в высокой точности и новый вектор невязки, и т.д. При такой организации вычислений основные затраты — решение системы уравнений в double для последовательности правых частей — текущих невязок системы уравнений.

Результаты численных экспериментов.

В таблице 2 для набора тестовых задач представлены оптимальные параметры предобусловливания при решении СЛАУ с первой правой частью в режиме 1 MPI x 4 threads на одном узле суперкомпьютера «Ломоносов». Параметр «Размер блока» в таблице означает размер блока мелко-блочного биения матрицы и предобусловливания при решении СЛАУ, параметр «DShift» означает диагональный сдвиг после мелко-блочного масштабирования, параметры τ_1 и τ_2 обозначают мелко-блочные пороги фильтрации первого и второго порядка точности, соответственно. Критерий остановки итерационной схемы: падение относительной нормы невязки в 10^9 раз. Результат для наибольшей задачи №17 приведен для расчета на 4-х узлах, т. к. памяти одного узла не хватает для построения предобусловливателя.

Таблица 2 - Основные параметры решения тестовых задач.

№ модели	Кол-во неизвестных	Размер блока	DShift	τ_1	τ_2	Время, с.
1	40 020	3	1e-6	1e-3	1e-6	0.42
2	88 782	3	1e-5	3e-3	1e-5	3.65
3	98 650	2	1e-6	1e-3	1e-6	8.59

4	112 560	3	1e-6	1e-3	1e-6	4.10
5	195 024	3	1e-6	1e-3	1e-6	10.4
6	247 542	3	1e-5	3e-3	1e-5	17.0
7	256 572	3	1e-5	3e-3	1e-5	16.8
8	397 950	3	1e-4	1e-2	1e-4	53.6
9	427 014	3	1e-6	1e-3	1e-6	41.6
10	755 592	3	1e-7	3e-4	1e-7	101.
11	798 230	2	3e-4	3e-3	1e-5	242.
12	889 890	3	1e-5	3e-3	1e-5	72.
13	1 623 600	3	1e-10	1e-5	1e-10	113.
14	2 534 446	2	1e-6	1e-3	1e-6	431.
15	3 247 200	3	1e-10	1e-5	1e-10	351.
16	4 870 800	3	1e-12	1e-6	1e-12	823.
17	8 033 520	3	1e-8	1e-6	1e-10	766. (*)

В таблице 3 для задачи №13 приведены результаты решения СЛАУ с оптимальным набором параметров предобусловливания для различных вариантов выбора размера блока мелко-блочного биения матрицы. В таблице в том числе приведены значения относительного заполнения матрицы $(nzA(b)/nzA(1))$ по отношению к хранению данных в точечном формате и предобусловливания по отношению к хранению матрицы в блочном формате $(nzU(b)/nzAU(b))$. Из таблицы следует, что для задачи №13 оптимальным размером блока мелко-блочного биения является значение $b=3$.

Таблица 3 - Влияние выбора параметра размера блока на сходимость итераций для задачи №13.

Размер блока, b	$nzA(b)/nzA(1)$	$nzU(b)/nzAU(b)$	Кол-во итераций	Время(Прес), с.	Время(Iter), с.	Время, с.
1	1.00	7.40	58	393.3	146.0	539.3
2	1.60	7.47	55	83.8	81.9	165.7
3	2.05	7.28	60	46.0	77.0	123.0
4	2.62	7.24	61	51.3	84.7	136.0
5	3.20	7.05	60	58.6	79.6	138.2
6	2.16	7.46	65	60.1	96.1	156.2
8	3.87	6.37	54	119.3	96.2	215.5
10	4.42	6.25	55	129.7	107.0	236.7

В таблице 4 для задачи №13 приведено время итерационной схемы при решении задачи с несколькими правыми частями. Из таблицы видно, что алгоритм SOFGMRES, при меньшем числе итераций чем в CG для первой правой части, требует большего времени счета из-за дополнительных затрат на ортогонализации. С другой стороны, для второй правой части, при условии сохранения подпространства, алгоритм SOFGMRES находит решение за значительно меньшее время. При большом числе правых частей, что часто имеет место в задачах строительной механики, использование алгоритма SOFGMRES может быть более предпочтительным.

Таблица 4 - Время решения задачи №13 для одной и двух правых частей различными итерационными методами.

Итерационный метод	№ правой части	Кол-во итераций	Время, с.
CG	1	400	199.3
	2	402	199.4

SOFGMRES	1	337	348.6
	2 (с сохранением подпространства от 1-й)	45	48.5

Для некоторых задач были оценены числа обусловленности отмасштабированных к единичной диагонали матриц жесткости: для задачи №9 получена оценка $\sim 3 \cdot 10^9$, для задачи №13 $\sim 2 \cdot 10^{10}$, для задачи №15 $\sim 8 \cdot 10^{11}$, для задачи №16 $\sim 4 \cdot 10^{12}$, для задачи №17 $\sim 3 \cdot 10^{13}$. Алгоритм вычисления оценок числа обусловленности для симметричных матриц см. в [9]. При таких больших значениях чисел обусловленности матриц, достаточно близких к обратной точности представления числа в формате double, трудно ожидать высокой точности вектора решения СЛАУ даже при вычислении этого решения прямыми методами в формате double. По видимому, для получения высокоточного решения подобных СЛАУ необходимо использовать повышенную точность представления чисел, в том числе при построении исходной матрицы коэффициентов.

В таблице 5 для всего набора тестовых задач представлены результаты некоторых параллельных расчетов на 1, 2, 4 и 8 узлах суперкомпьютера «Ломоносов» с распараллеливанием по нитям вычислений (от 1-й до 4-х нитей). Приводятся времена построения предобусловливателя плюс время итерационной схемы (в скобках указано количество итераций).

Таблица 5 - Время расчета в многопроцессорном режиме.

№ модели	1 (1x1)	2 (1x2)	4 (1x4)	8 (2x4)	16 (4x4)	32 (8x4)
1	0.45+0.73(38)	0.23+0.83(66)	0.13+0.51(66)	0.08+0.97(187)	-	-
2	2.5+4.0(62)	1.2+5.1(126)	0.7+3.3(126)	0.3+2.9(197)	-	-
3	7.9+11.5(107)	3.8+15.3(248)	2.0+10.0(248)	-	-	-
4	3.1+7.5(90)	1.6+6.3(123)	0.8+4.3(124)	0.5+4.0(209)	-	-
5	11+8(48)	6+15(150)	3+10(150)	2+12(243)	1+9(363)	0.4+7.8(473)
6	13+17(88)	8+13(117)	5+9(117)	2+9(200)	-	-
7	20+16(82)	10+11(96)	6+7(96)	3+6(139)	1+7(221)	-
8	35+74(198)	18+55(198)	10+34(198)	5+18(204)	-	-
9	30+55(160)	19+53(218)	12+31(218)	-	-	-
10	135+168(180)	75+140(240)	49+100(240)	23+92(400)	11+91(630)	6+78(840)
11	174+191(173)	88+135(177)	116+126(197)	-	-	-
12	39+75(98)	19+52(99)	12+42(101)	8+20(99)	-	-
13	165+60(30)	86+63(46)	44+56(46)	21+37(46)	12+23(45)	-
14	443+476(141)	256+536(181)	132+349(206)	-	-	-
15	332+754(160)	169+580(160)	90+508(164)	48+287(176)	36+157(208)	23+151(367)
16	-	-	137+686(139)	-	-	-
17	-	-	-	-	96+670(350)	-

Выводы. В работе приведены данные по решению итерационными методами плохообусловленных симметричных СЛАУ для набора задач строительной механики в различных параллельных режимах на суперкомпьютере «Ломоносов». Из полученных оценок числа обусловленности для некоторых матриц следует, что эти задачи являются чрезвычайно трудными для решения итерационными методами. Результаты экспериментов показывают, что тем не менее все задачи были успешно решены. Кроме того, для некоторых задач рассмотренные итерационные методы демонстрируют неплохую вычислительную эффективность и параллельную масштабируемость. С другой стороны, для ряда самых трудных задач не удалось получить желаемую эффективность параллельных итерационных алгоритмов. По мнению авторов, это связано со значительным ухудшением качества предобусловливания при априорном блочном ограничении заполнения предобусловливания при распараллеливании по процессорам/нитям. Кроме того, не использованы в полной мере возможности улучшения эффективности алгоритмов за счет применения блочных (многовекторных) итерационных алгоритмов. Использование мелко-блочного формата хранения матрицы и предобусловливания, а также блочных итерационных алгоритмов позволяет рассчитывать, что предложенные итерационные алгоритмы могут быть эффективно реализованы на гибридных вычислительных системах с ускорителями, например, на основе GPU. В последующих работах авторы планируют развивать итерационные методы в этих направлениях.

ЛИТЕРАТУРА:

1. В.Н. Симбиркин, А.В. Фросин, Развитие и взаимодействие компонентов автоматизированной системы инженерного анализа СТАРКОН для массового применения в строительстве// Строительная механика и расчет сооружений. – 2010. – №5. – С. 67-71.
2. Ю.Н. Жук, В.Н. Симбиркин, Программный комплекс STARK ES// Современное высотное строительство: монография. – М.: ИТЦ Москомархитектуры, 2007. – 464 с.
3. Программный комплекс для расчета строительных конструкций на прочность устойчивость и колебания STARK ES. Версия 4x4 (2007). Руководство пользователя. – М.: Еврософт, 2008. – 399 с.
4. Г.Б. Сушко, С.А. Харченко, Экспериментальное исследование на СКИФ МГУ “Чебышев” комбинированной MPI+threads реализации алгоритма решения систем линейных уравнений, возникающих во FlowVision при моделировании задач вычислительной гидродинамики. Материалы международной научной конференции "Параллельные вычислительные технологии" (ПАВТ'2009), Нижний Новгород, 30 марта – 3 апреля 2009 г., Челябинск: Изд. ЮУрГУ, 316-324, 2009.
5. <http://crd.lbl.gov/~dhbailey/mpdist/>
6. I.E. Kaporin, High Quality Preconditioning of a General Symmetric Positive Definite Matrix Based on its $U^T U + U^T R + R^T U$ decomposition // Numer. Linear Algebra Appl. – 1998. – Vol. 5. – P. 483-509.
7. С.А. Харченко, "Влияние распараллеливания вычислений с поверхностными межпроцессорными границами на масштабируемость параллельного итерационного алгоритма решения систем линейных уравнений на примере уравнений вычислительной гидродинамики". Материалы международной научной конференции "Параллельные вычислительные технологии" (ПАВТ'2008), Санкт-Петербург, 28 января – 1 февраля 2008 г. Челябинск, Изд. ЮУрГУ, 2008, с.494-499.
8. С.А. Харченко, "Параллельная реализация итерационного алгоритма решения несимметричных систем линейных уравнений с частичным сохранением спектральной/сингулярной информации при явных рестартах" // "Вычислительные методы и программирование", 2010, т. 11, 373-381.
9. С.А. Харченко, "Высокоточный двухуровневый параллельный итерационный алгоритм решения плохообусловленных СЛАУ" (статья представлена на международную суперкомпьютерную конференцию «Научный сервис в сети Интернет 2011: эксафлопсное будущее»).