

**О НОВЫХ ВЫЧИСЛИТЕЛЬНЫХ
ВОЗМОЖНОСТЯХ SCAD++ ПРИ
РЕШЕНИИ БОЛЬШЕРАЗМЕРНЫХ
ЗАДАЧ НА МНОГОЯДЕРНЫХ
КОМПЬЮТЕРАХ**

С. Ю. Фиалко

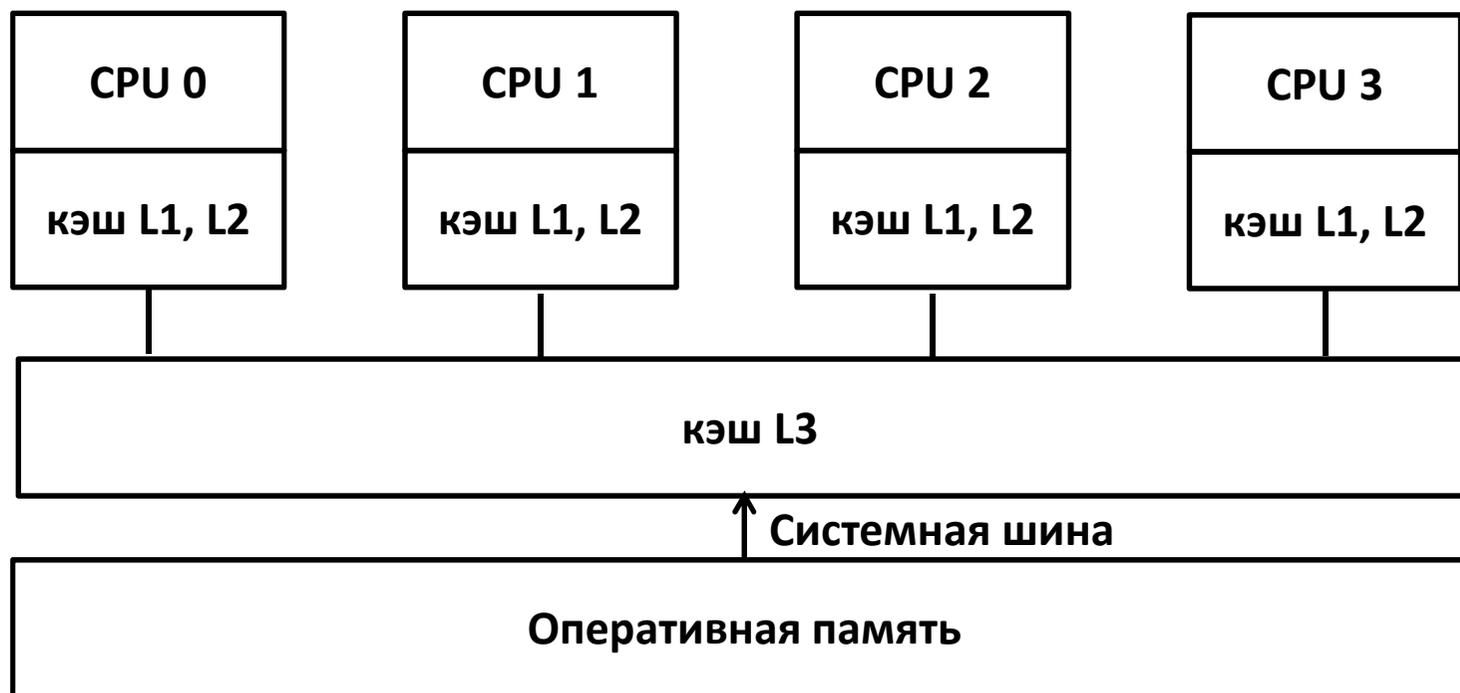
Технический университет
«Краковская политехника»

План

- Особенности вычислений на многоядерных настольных компьютерах и ноутбуках
- Прямые и итерационные методы решения большеразмерных задач
- PARFES
- PSICCG
- Численные результаты

Особенности вычислений на многоядерных настольных компьютерах и ноутбуках

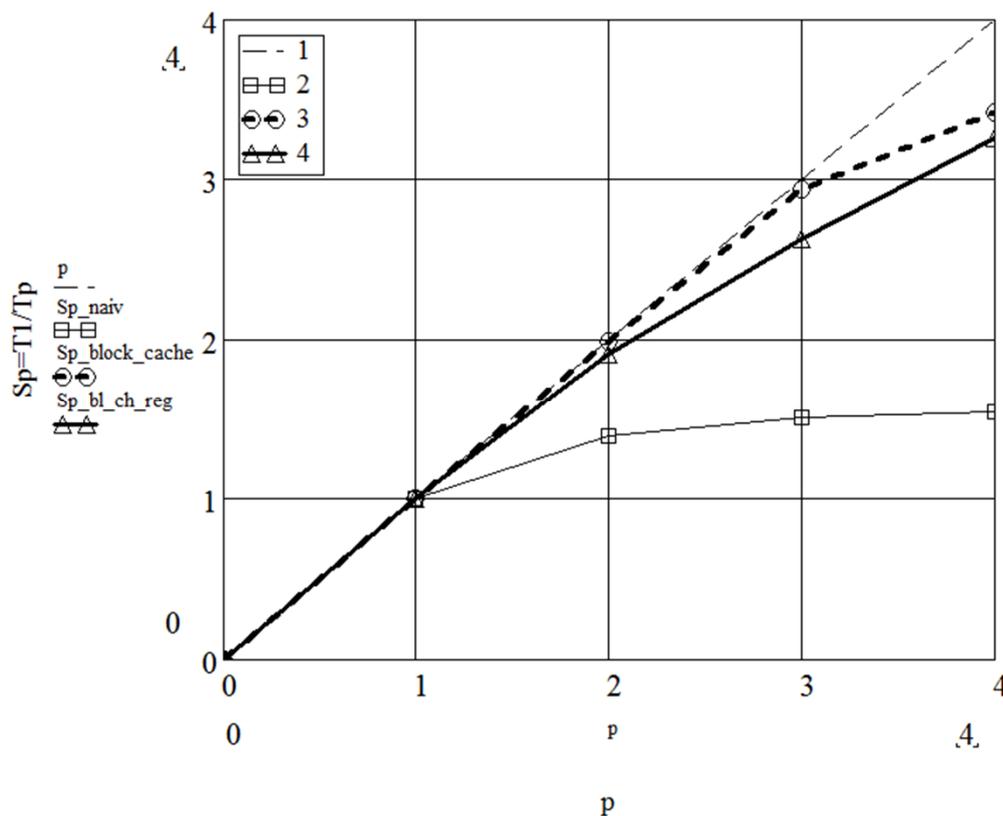
- Ограниченный объем оперативной памяти и низкая пропускная способность системной шины ПК приводят к тому, что методы, хорошо зарекомендовавшие себя на компьютерах с распределенной памятью, часто не эффективны.



Особенности вычислений на многоядерных настольных компьютерах и ноутбуках

Показатель производительности алгоритма: $q = f/m$,

f – к-во арифметических операций, m – к-во пересылок данных ОП – кэш - ОП



Алгоритмы низкой производительности ($q = 1, 2$) – процессор выполняет большое к-во пустых циклов в ожидании пересылки данных из ОП в кэш. Система памяти перегружена – низкая ускоряемость.

Алгоритмы высокой производительности ($q \sim \sqrt{M}$) – процессор работает с минимальным к-вом пустых циклов, т. к. интенсивно используется кэш. Система памяти не перегружена – высокая ускоряемость

Типичная ускоряемость алгоритмов низкой ($q = 1$, кривая 2) и высокой ($q \sim \sqrt{M}$, кривые 3, 4) производительности на многоядерных компьютерах

Достоинства и недостатки прямых и итерационных методов решения СЛАУ современных программных комплексов

- Мы ограничимся рассмотрением методов решения СЛАУ для разреженных симметричных матриц для пакета загрузжений

$$Kx_s = b_s, s = 1, 2, \dots, nrhs$$

где $K = K^T$, s – номер загрузки.

- **Прямые методы**
 - + Время решения не зависит от числа обусловленности матрицы K .
 - + Время решения слабо зависит от к-ва загрузжений.
 - + Возможность выявления геометрической изменяемости расчетной модели.
 - + Высокая производительность (матричное умножение) и ускоряемость на многоядерных компьютерах при разложении матрицы.

Достоинства и недостатки прямых и итерационных методов решения СЛАУ современных программных комплексов

- Прямые методы

- Квадратичная зависимость времени решения от размерности задачи.
- Если размерность задачи превышает возможности ОП, то для больших задач ($> 1\,000\,000$ уравнений) продолжительность разложения матрицы и прямых-обратных подстановок начинает значительно возрастать вследствие интенсивного использования диска.

$K = L \cdot S \cdot L^T$ - факторизация симметричной матрицы обобщенным методом Холецкого

$L \cdot Y = B \rightarrow Y$ – прямая подстановка, $B = \{b_s\}$, $Y = \{y_s\}$

$S \cdot Z = Y \rightarrow Z$ – диагональное масштабирование, $Z = \{z_s\}$

$L^T \cdot X = Z \rightarrow X$ – обратная подстановка, $X = \{x_s\}$.

Достоинства и недостатки прямых и итерационных методов решения СЛАУ современных программных комплексов

- Итерационные методы

- Количество итераций существенно зависит от обусловленности матрицы. Для плохо обусловленных задач сходимость замедляется или исчезает вовсе.
- Для каждой правой части итерационный процесс начинается с самого начала.
- Отсутствие достоверного выявления геометрической изменяемости.
- Низкая производительность ведущих алгоритмов ($y_k = Kx_k, Vz_k = r_k$) и, как следствие, низкая ускоряемость на многоядерных настольных компьютерах.
- + Квазилинейная зависимость времени решения от размерности задачи.
- + Задача решается в ОП – отсутствуют медленные операции обмена данных с диском.

Прямые методы решения большеразмерных задач в современных программных комплексах

- **Многофронтальный метод (BSMFМ)**
 - + Работает с произвольным методом упорядочения.
 - + Обеспечивает минимальный размер факторизованной матрицы.
 - + Использует высокопроизводительные алгоритмы и многопоточность.
 - + В случае дефицита ОП автоматически подключает диск.
 - Содержит избыточное к-во пересылок данных память – память и память – диск. Вследствие этого не позволяет достичь максимальной производительности и высокой ускоряемости на компьютерах архитектуры SMP.
- **PARDISO (Intel Math Kernel Library)**
 - + Если задача решается в ОП, демонстрирует отличную производительность и ускоряемость даже на обычных ПК и ноутбуках.
 - Если размерность задачи превосходит возможности ОП, метод становится беспомощным, поскольку режим ООС надлежащим образом не работает.

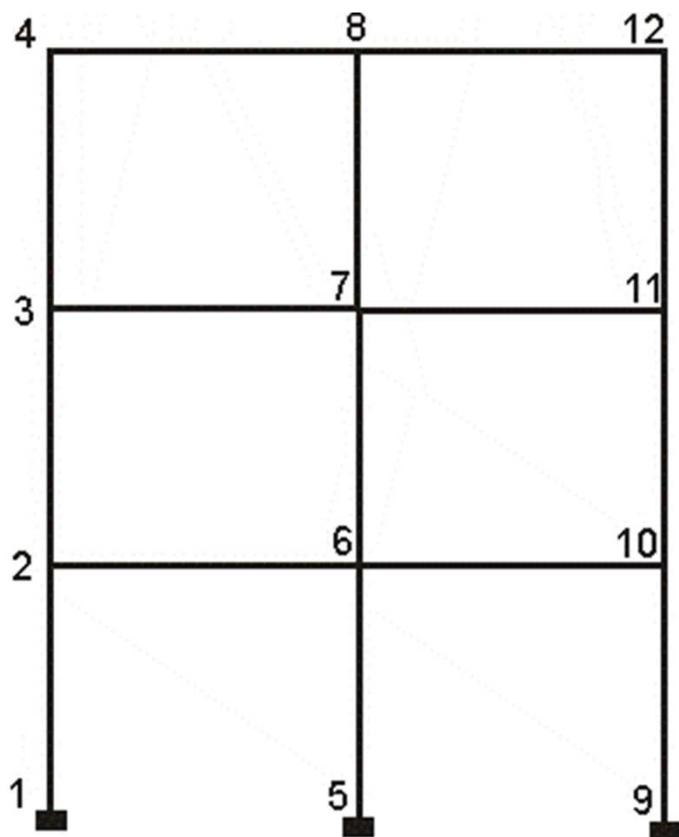
Прямые методы решения большеразмерных задач в современных программных комплексах

- **PARFES (Parallel Finite Element Solver)**
 - + В режиме ОП демонстрирует производительность и ускоряемость, близкую к PARDISO.
 - + При дефиците ОП автоматически переключается в режим ООС или ООС1 с использованием диска. В режиме ООС диск используется только для однократной записи разложенной матрицы. В режиме ООС1 количество обменов с диском значительно возрастает, однако этот режим позволяет решать большие задачи на компьютерах с малым объемом ОП.
 - Объем факторизованной матрицы несколько больший, чем для BSMFM. Поэтому в режимах ООС, ООС1 прямые-обратные подстановки идут дольше.

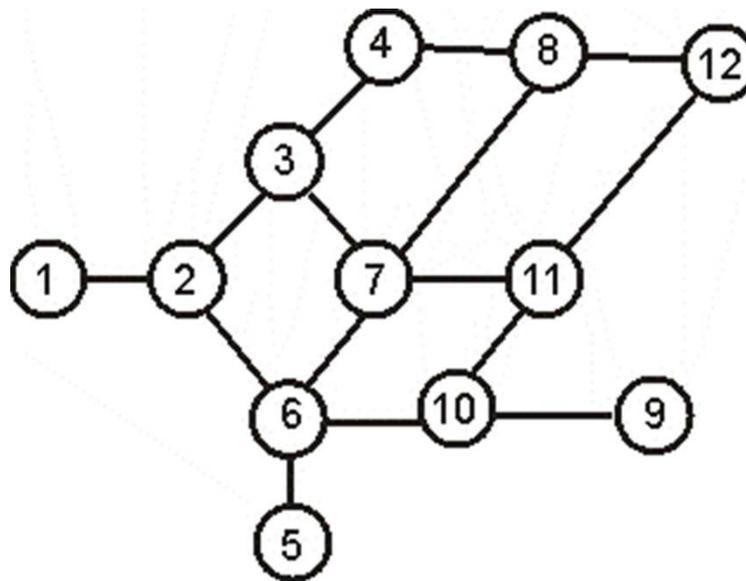
PARFES

- **Высокая производительность обеспечивается делением разреженной матрицы на плотные прямоугольные блоки и применением высокопроизводительных процедур (главным образом, матричного умножения) из библиотеки Intel Math Kernel Library.**
- **Хорошая ускоряемость при увеличении к-ва процессоров обеспечивается разработанным специфическим алгоритмом, обеспечивающим равномерную загрузку всех ядер (load balance).**

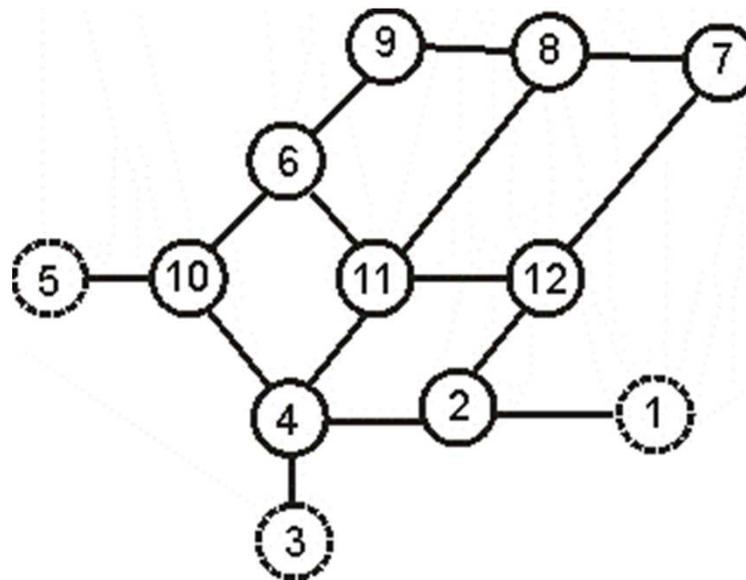
➤ **Пример (PARFES)**



Плоская рама



Граф смежности узлов до упорядочения



Граф смежности узлов после упорядочения

PSICCG

- Распараллеливание итерационных методов на многоядерных компьютерах – сложная задача.

Библиотека процедур высокой производительности	PCG solver	Умножение разреженной симметричной матрицы на вектор	Прямые-обратные подстановки для разреженной нижней (верхней) треугольной матрицы
Intel MKL 10.3	No	Yes	No
IMSL 7.0	No	Yes	No

No – параллельной версии не существует; **Yes** – существует.

Идея: каждая правая часть итерируется на отдельном потоке.

Предобуславливание

Дано: $\mathbf{K}\mathbf{x}^s = \mathbf{b}^s, \quad s = 1, 2, \dots, nrhs$ (1)

nrhs – к-во загрузжений.

Предобусловленная задача:

$$\mathbf{B}^{-1}\mathbf{K}\mathbf{x}^s = \mathbf{B}^{-1}\mathbf{b}^s, \quad s = 1, 2, \dots, nrhs \quad (2)$$

где $\mathbf{B} = \mathbf{B}^T, \quad \mathbf{B} > 0, \quad Cond(\mathbf{B}^{-1}\mathbf{K}) < Cond(\mathbf{K})$

В результате на каждой итерации необходимо решать:

$$\mathbf{B}\mathbf{z}_k^s = \mathbf{r}_k^s \quad (3)$$

Конструкция матрицы \mathbf{B} должна быть такой, что (3) решается гораздо быстрее, чем (1).

Метод сопряженных градиентов с предобуславливанием для пакета загрузений

1. Начальное приближение:

$$s = 1; \quad k = 0; \quad \mathbf{x}_0^s = 0; \quad \mathbf{r}_0^s = \mathbf{b}^s; \quad \mathbf{p}_0^s = \mathbf{z}_0^s = \mathbf{B}^{-1} \mathbf{r}_0^s$$

2. Линейный поиск, коррекция вектора решения и невязки:

$$\mathbf{w}_k^s = \mathbf{K} \mathbf{p}_k^s; \quad \alpha_k^s = \frac{(\mathbf{r}_k^s)^T \mathbf{z}_k^s}{(\mathbf{p}_k^s)^T \mathbf{w}_k^s};$$

$$\mathbf{x}_{k+1}^s = \mathbf{x}_k^s + \alpha_k^s \mathbf{p}_k^s; \quad \mathbf{r}_{k+1}^s = \mathbf{r}_k^s - \alpha_k^s \mathbf{w}_k^s;$$

3. Проверка сходимости:

$$\text{if} \left(\left\| \mathbf{r}_{k+1}^s \right\|_2 \leq \text{tol} \cdot \left\| \mathbf{b}^s \right\|_2 \wedge \left\| \mathbf{r}_{k+1}^s \right\|_{\text{inf}} \leq \text{tol} \cdot \left\| \mathbf{b}^s \right\|_{\text{inf}} \right)$$

$s++$; *if* ($s > nrhs$) *stop*;

else goto 1

Метод сопряженных градиентов с предобуславливанием для пакета загрузений

4. Определение нового сопряженного направления:

$$\mathbf{z}_{k+1}^s = \mathbf{B}^{-1} \mathbf{r}_{k+1}^s; \quad \beta_k^s = \frac{(\mathbf{r}_{k+1}^s)^T \mathbf{z}_{k+1}^s}{(\mathbf{r}_k^s)^T \mathbf{z}_k^s}; \quad \mathbf{p}_{k+1}^s = \mathbf{z}_{k+1}^s + \beta_k^s \mathbf{p}_k^s; \quad k++;$$

5. Go to 2

Здесь \mathbf{p}_k^s – вектор сопряженного направления.

Метод сопряженных градиентов с предобуславливанием для пакета загрузений

Создать очередь задач Q для каждого загрузения – $Q: \{1, 2, \dots, nrhs\}$

#pragma omp parallel

while(Q содержит какие-либо элементы)

lock 1

$s \leftarrow Q; Q \leftarrow Q/s;$

получить b_s

end lock 1

стартовать итерац. процесс для s -го загрузения

lock 2

записать x_s как окончательный результат

end lock 2

end while

end of parallel region

Построение предобуславливания

- Если $\mathbf{B} = \mathbf{K}$, то $\mathbf{B}^{-1}\mathbf{K} = \mathbf{I}$ и мы получаем точное решение при любом стартовом приближении за 1 итерацию.
- Идея: представим $\mathbf{B} = \mathbf{H}\cdot\mathbf{H}^T$, где \mathbf{H} – нижняя треугольная матрица, получаемая при факторизации матрицы \mathbf{K} отбрасыванием «малых» элементов. Тогда структура матрицы \mathbf{H} будет более разреженной, чем \mathbf{L} – полного фактора Холецкого.

if ($H_{ij}^2 < \psi H_{i,i} H_{j,j}$) $i \in L_j$, (*entry is small – reject it*)

$$H_{i,i}^+ = \sqrt{\frac{H_{i,i}}{H_{j,j}} |\mathbf{v}_{0,i}|}, \quad H_{j,j}^+ = \sqrt{\frac{H_{j,j}}{H_{i,i}} |\mathbf{v}_{0,i}|}, \quad H_{ij} = 0$$

else

$L_j \leftarrow \mathbf{v}_{0,i} / \sqrt{H_{j,j}}$, $List_i \leftarrow j$, (*entry is not small – retain it*
and put to nonzero structure of matrix H, add j to List_i)

Построение предобуславливания

- Параметр $0 \leq \psi \leq 1$ управляет этим процессом.
- Если $\psi = 0$ – полная факторизация (сходимость к точному решению обеспечивается за 1 итерацию, однако требования к оперативной памяти и время решения точно такие же, как для прямого метода ($H = L$)).
- Если $\psi = 1$, то H – диагональная матрица. Минимальные требования к оперативной памяти, отсутствие затратной процедуры $Bz_k^s = r_k^s$, однако слабые свойства предобуславливания – отсутствие сходимости для большинства интересующих нас задач.
- Стратегия: стараемся удерживать ψ как можно меньшим, пока объем ОП позволяет разместить все данные в ОП и время неполной факторизации приемлемо мало.

Построение предобуславливания

- Если объем ОП оказывается слишком малым для принятого значения ψ , алгоритм прерывает факторизацию, увеличивает ψ в 10 раз и начинает с начала. **Поэтому при повторном решении задачи следует просмотреть протокол предыдущего решения и принять из него то значение ψ , при котором удалось разложить матрицу. Тогда мы избежим лишних попыток разложения.**
- Если объема ОП достаточно, сходимость быстрая, но время факторизации большое, можно попробовать увеличить ψ (в 10 раз).
- **Решающее значение при решении больших задач имеет объем оперативной памяти компьютера, разрядность ОС (x64) и разрядность приложения SCAD (64-разрядное приложение).**

Численные результаты

Все примеры взяты из коллекции задач SCAD Soft (www.scadsoft.com) и решались на следующих компьютерах:

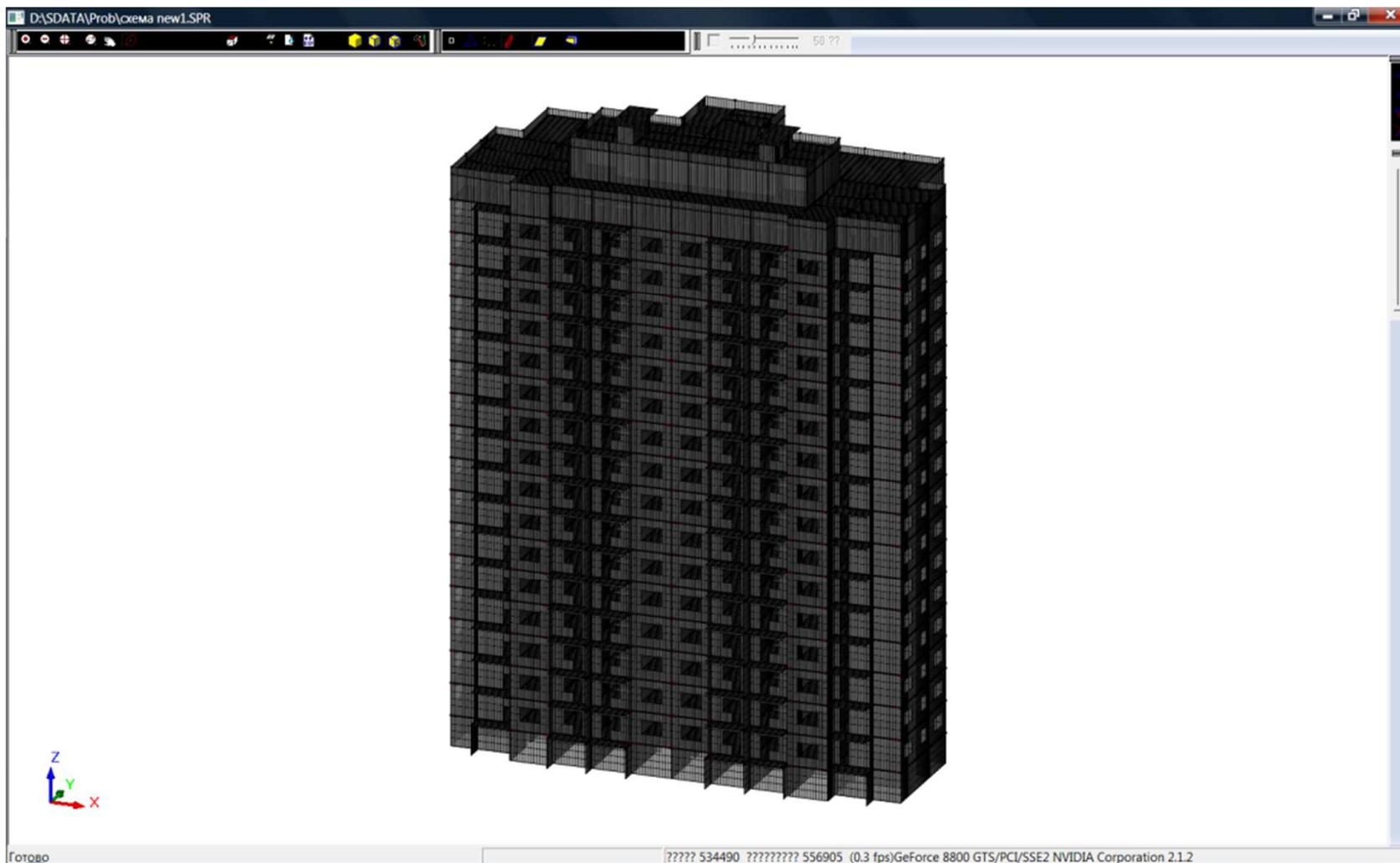
1. Laptop, DELL XPS L502X,
4-core Intel® Core™ i7-2760QM CPU 2.4/3.4 GHz processor,
RAM DDR3 8 GB, OS Windows 7 (64 bit) Professional, SP1.
2. AMD Phenom™ II x4 995 3.2 GHz, RAM DDR3, 16 GB,
OS Windows Vista™ Business (64-bit), SP 2.
3. Workstation, DELL,
two 6-core Intel Xeon X5660 @ 2.8/3.2 GHz processors,
RAM DDR3 24 GB, OS Windows 7 (64 bit) Professional, SP1.

Численные результаты

Мы сравниваем время решения задачи предложенным итерационным PSICCG методом со временем решения этой же задачи методом PARFES и классическим итерационным методом ICCG0 с коррекцией диагонали, обеспечивающей положительную определенность неполного фактора H .

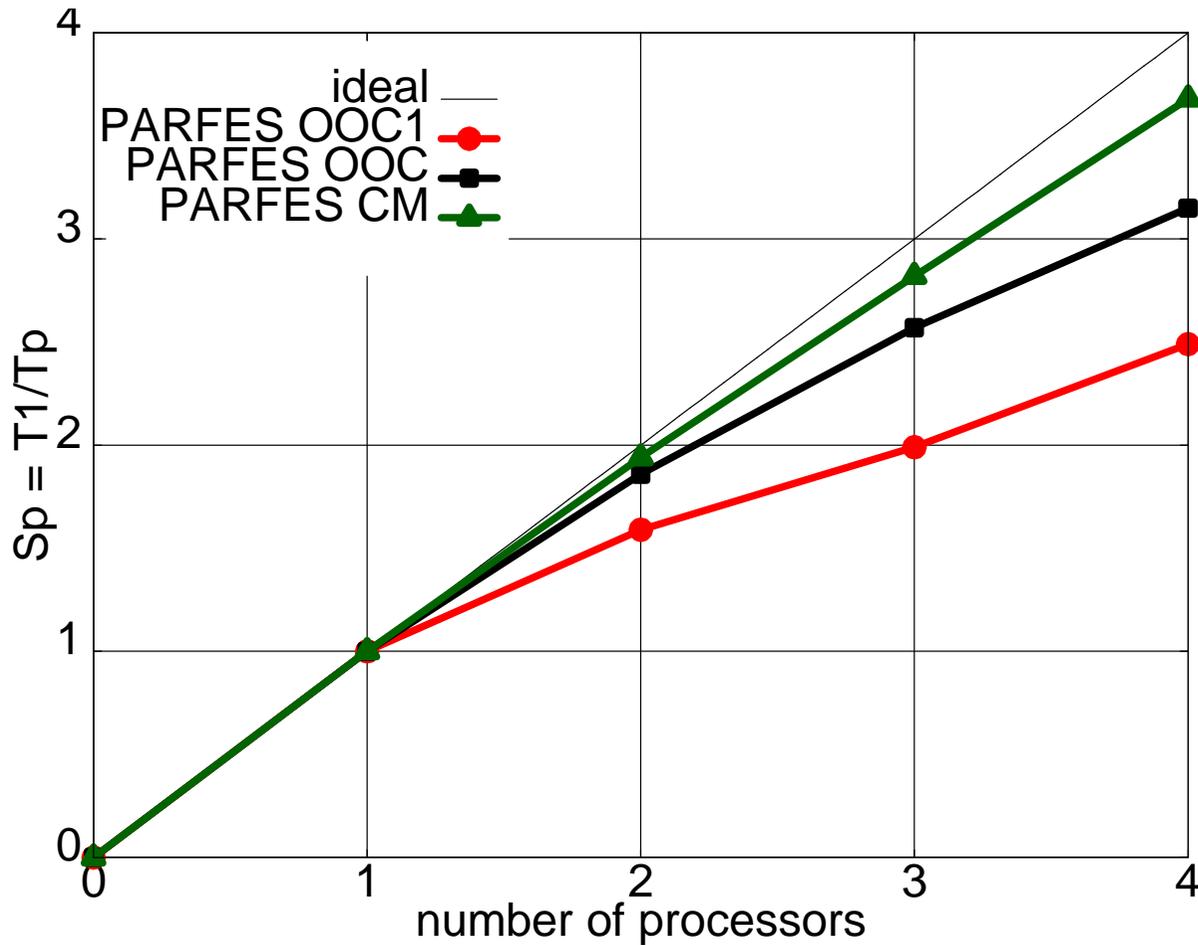
В методе ICCG0 в неполном факторе H сохраняются только те элементы, которые расположены на позициях ненулевых элементов исходной матрицы K .

Численные результаты



Задача schema_new_1, 3 198 609 уравнений

Численные результаты



32 453 MFLOPS
12 186 MB RAM

22 898 MFLOPS
2 995 MB RAM

8 610 MFLOPS
706 MB RAM

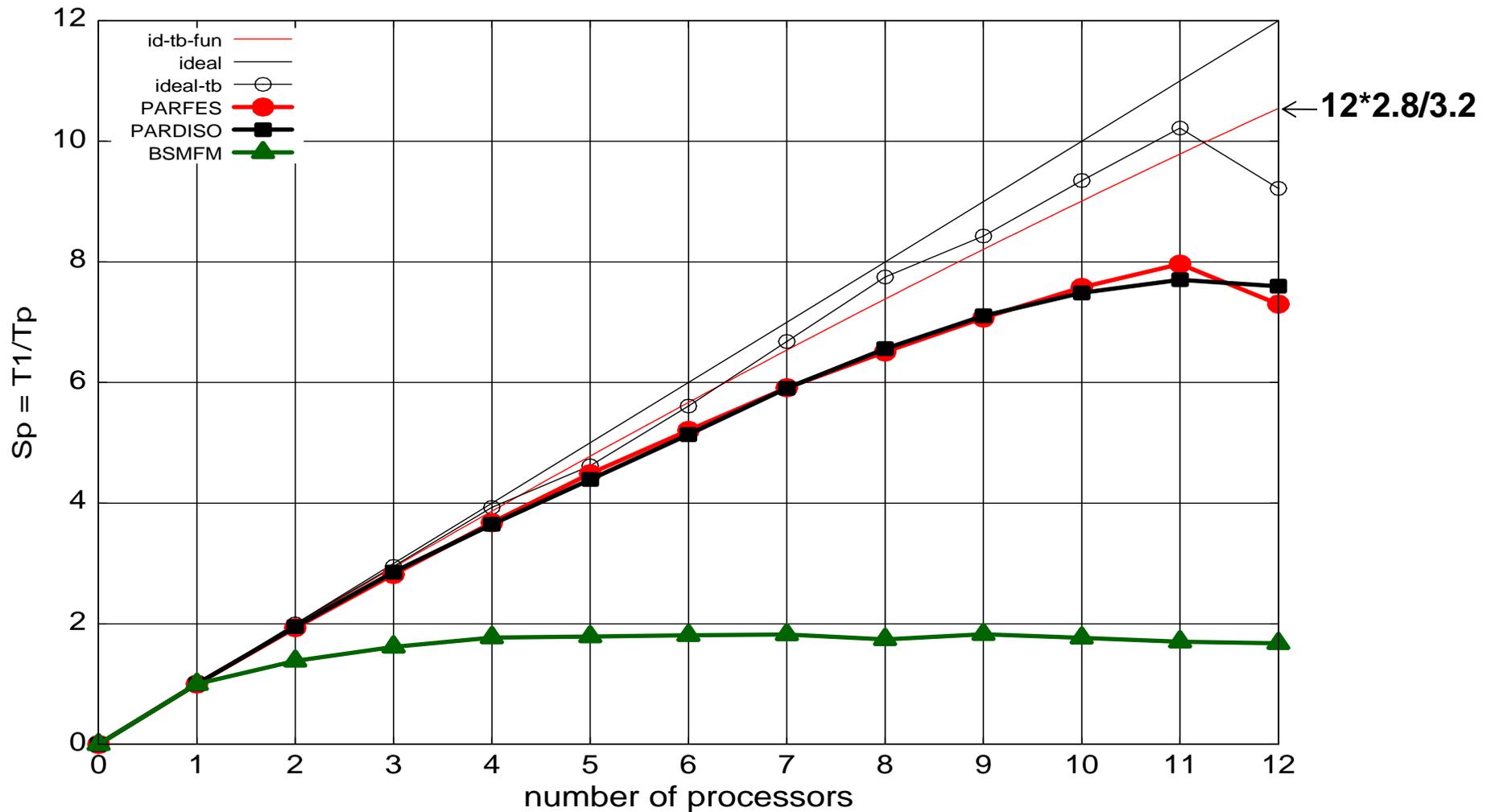
**Задача Schema_new_1, компьютер с процессором
AMD Phenom™ II x4 995 (этап численной факторизации)**

Численные результаты

Продолжительность этапов решения задачи **schema_new_1** (3,198,609 уравнений), рабочая станция с процессо-ром **Intel Xeon X5660 @ 2.8 GHz /3.2 GHz (12 ядер)**, RAM 24 GB, DDR3, режим CM, платформа x64

К-во проц.	PARFES		PARDISO		BSMFM	
	Анал., с	Чис. Факт., с	Анал., с	Чис. Факт., с	Анал., с	Чис. Факт., с
1	16.9	654	31.06	596	13	1406
2	16.9	337.8	23.59	305.3	13	1015
3	16.9	232.1	25.33	208.6	13	869
4	16.9	177.9	23.26	163.3	13	793
5	16.9	145.7	23.79	135.7	13	786
6	16.9	125.6	25.68	116	13	777
7	16.9	110.6	23.11	100.9	13	772
8	16.9	100.5	23.43	90.83	13	807
9	16.9	92.5	23.98	83.85	13	770
10	16.9	86.3	23.71	79.6	13	796
11	16.9	82.1	29.86	77.36	13	825
12	16.9	87.5	28.58	78.45	13	839

Численные результаты



Задача Schema_new_1, компьютер с процессором **Intel Xeon X5660 @ 3.2/2.8 GHz (12 ядер)** (этап численной факторизации)

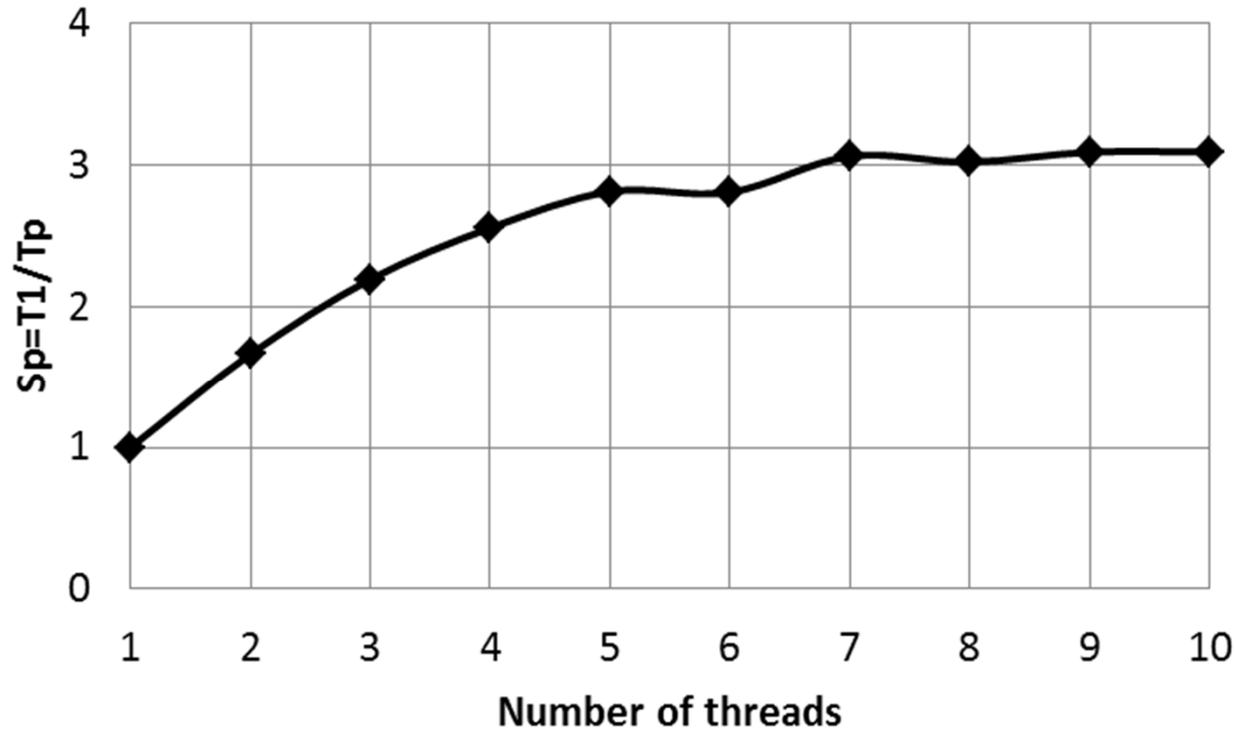
Численные результаты

Продолжительность решения задачи Schema 1 методом PSICCG на рабочей станции DELL с двумя процессорами Intel Xeon X5660 (12 ядер), RAM DDR3 24 GB, 13 загрузений ($\psi = 10^{-14}$, $\psi_1 = 10^{-7}$ и $tol = 10^{-4}$).

Number of threads	Incomplete factorization, s	Iterative process, s	Total task time, s
1	784	808	1,592
2	481	476	957
3	372	354	726
4	337	286	623
5	327	239	566
6	332	235	567
7	331	189	520
8	338	189	527
9	326	189	515
10	326	189	515

К-во итераций для каждого загрузкиения (метод PSICCG):
71/55/62/65/73/62/59/59/18/64/63/65

Численные результаты



Ускоряемость метода PSICCG для задачи «Schema 1» при увеличении к-ва потоков. Компьютер с процессором Intel Xeon X5660 .

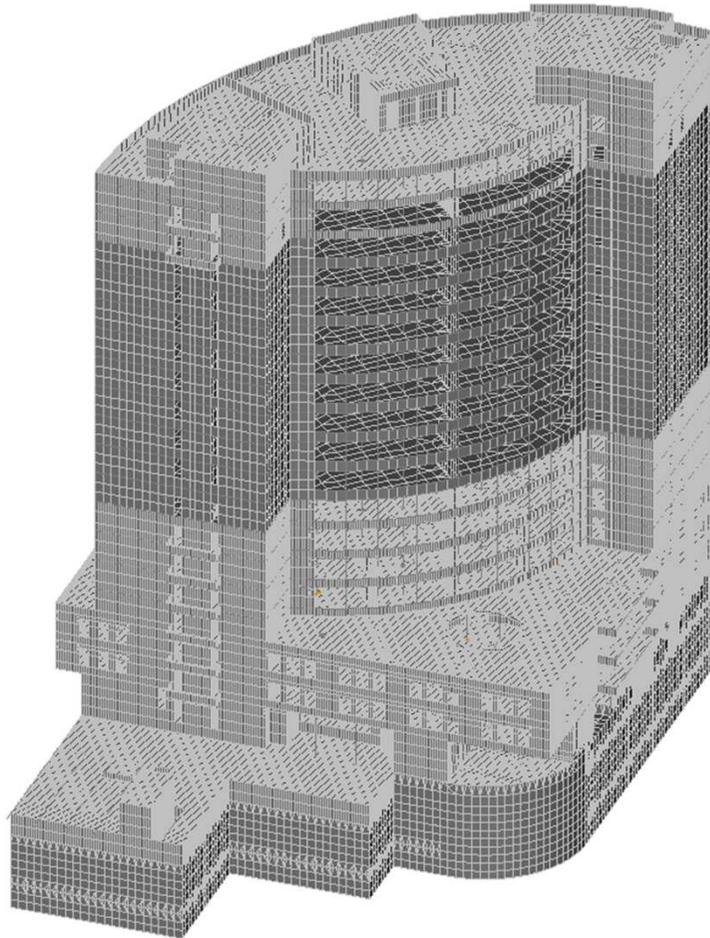
PARFES : 230 с на 10 потоках режим ОП

BSMFM: 1 030 с на 10 потоках режим ОП

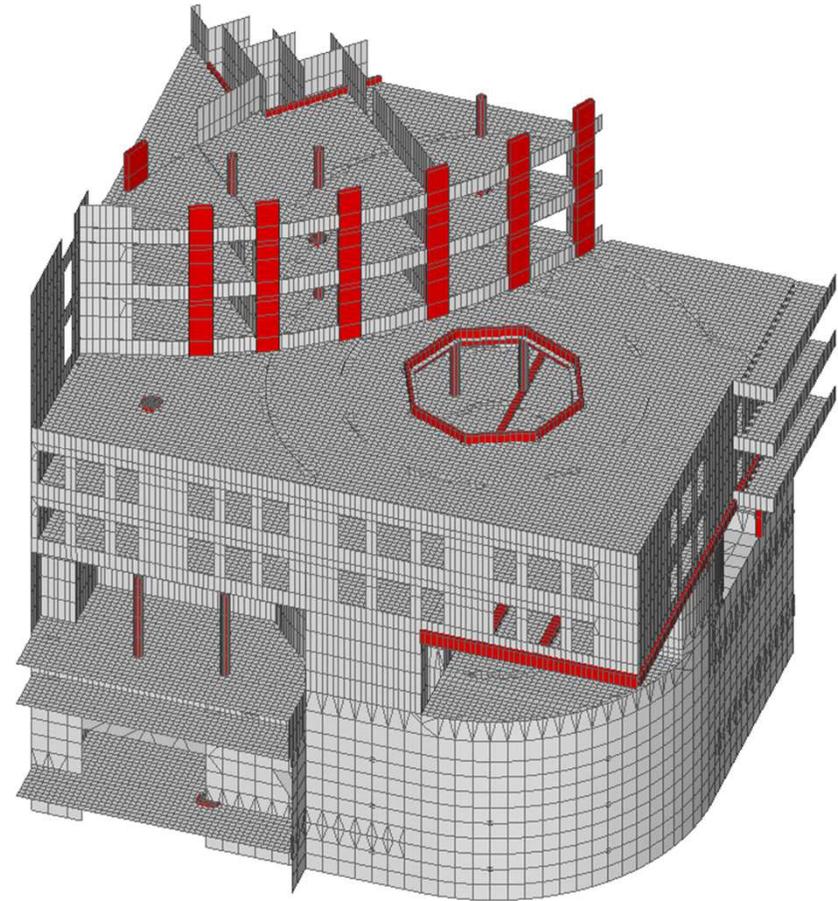
PSICCG : 515 с на 10 потоках ($\psi = 10^{-14}$, $\psi_1 = 10^{-7}$ и $tol = 10^{-4}$)

ICCG0 : 22 258 с на 10 потоках ($tol = 10^{-4}$)

Численные результаты



**Расчетная модель жилого здания
(проект «Ракетный бульвар»
2,002,848 уравнений, 7 загрузжений)**



Фрагмент расчетной схемы

Численные результаты

Продолжительность решения задачи «Ракетный бульвар» (2,002,848 уравнений) на ноутбуке DELL с процессором Intel® Core™ i7-2760QM CPU 2.4 GHz, RAM DDR3 8 GB. 7 загрузений.

PARFES : 492 с на 4 потоках, режим ООС

BSMFM: 695 с на 4 потоках, режим ООС

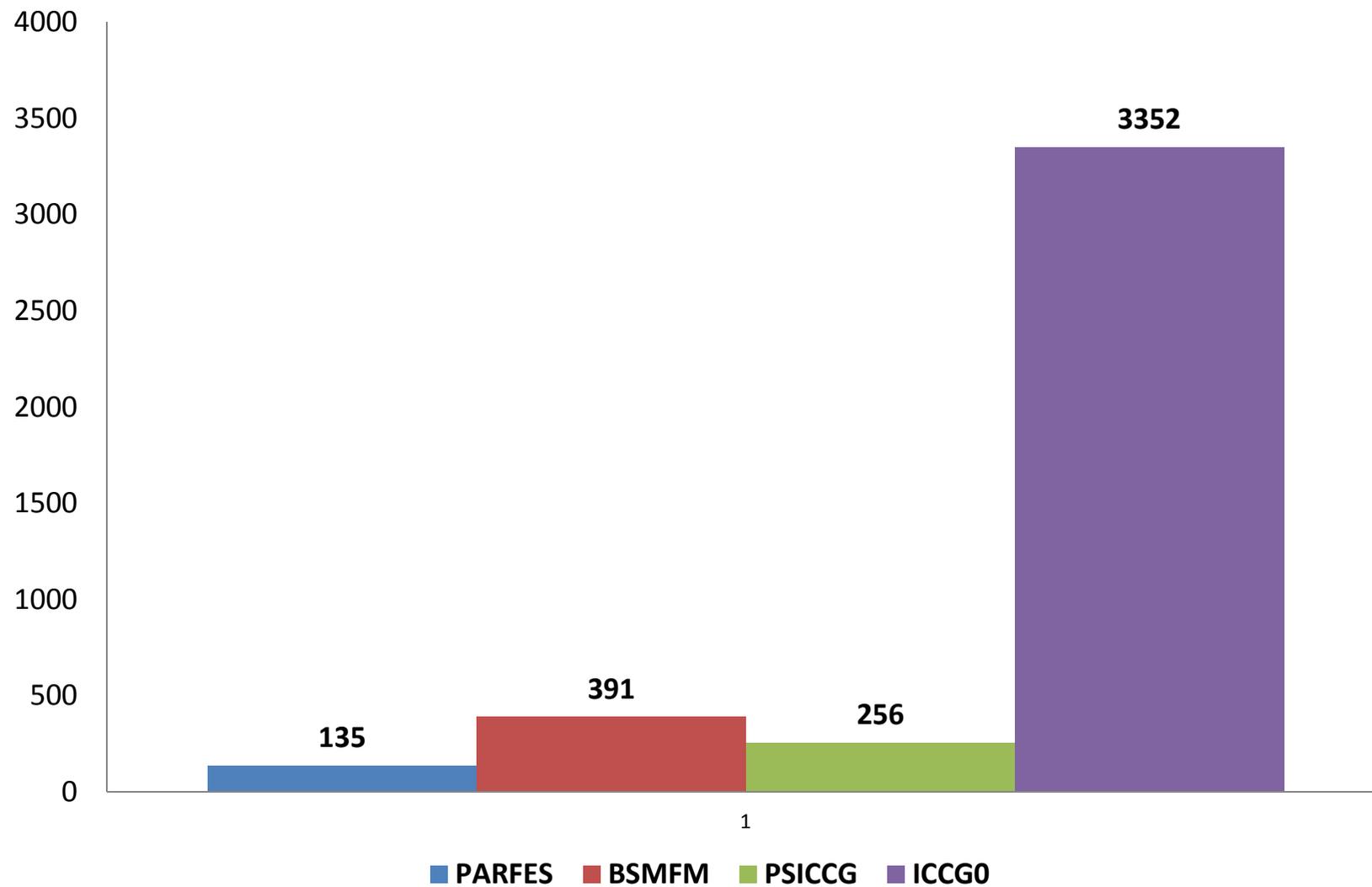
PSICCG : 357 с на 4 потоках, ОП ($\psi = 10^{-10}$ и $\text{tol} = 10^{-4}$)

ICCG0 : 4 332 с на 4 потоках, ОП ($\text{tol} = 10^{-4}$)

К-во итераций для каждого загрузкиения (метод PSICCG):
159/150/144/147/124/114/143

К-во итераций для каждого загрузкиения (метод ICCG0):
5033/5649/6754/6833/4062/6203/5804

Численные результаты



Численные результаты

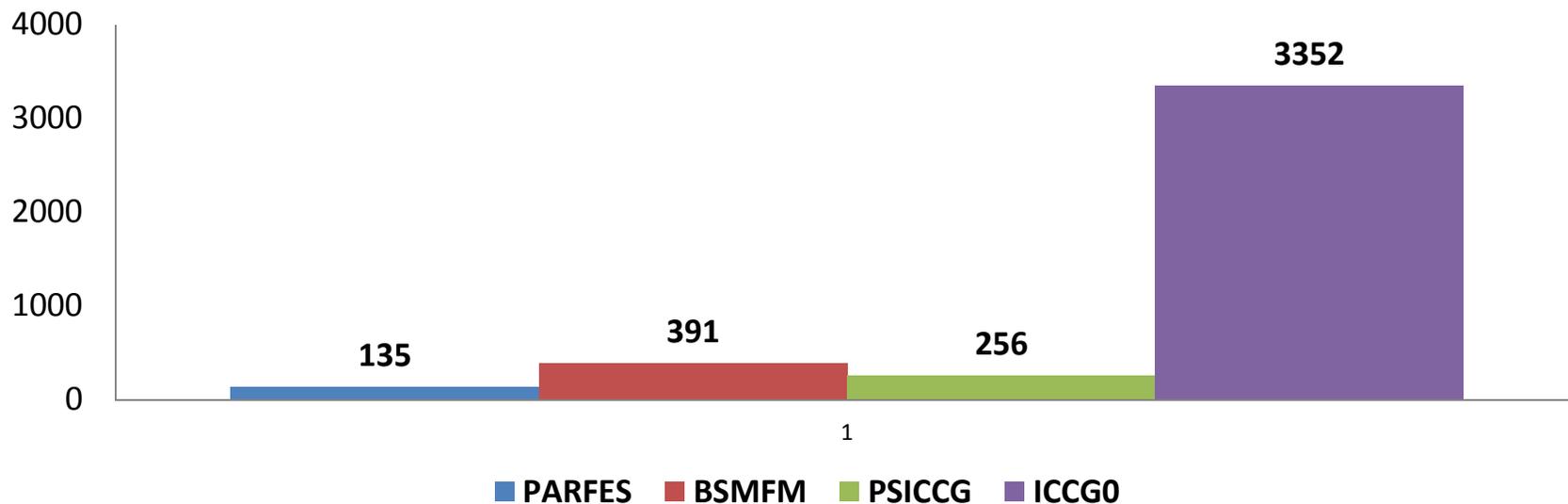
Продолжительность решения задачи «Ракетный бульвар» (2,002,848 уравнений) на рабочей станции DELL с процессором Intel Xeon X5660 @ 2.8 GHz /3.2 GHz, RAM DDR3 24 GB, 7 загрузений

PARFES : 135 с на 8 потоках, режим ОП (core mode)

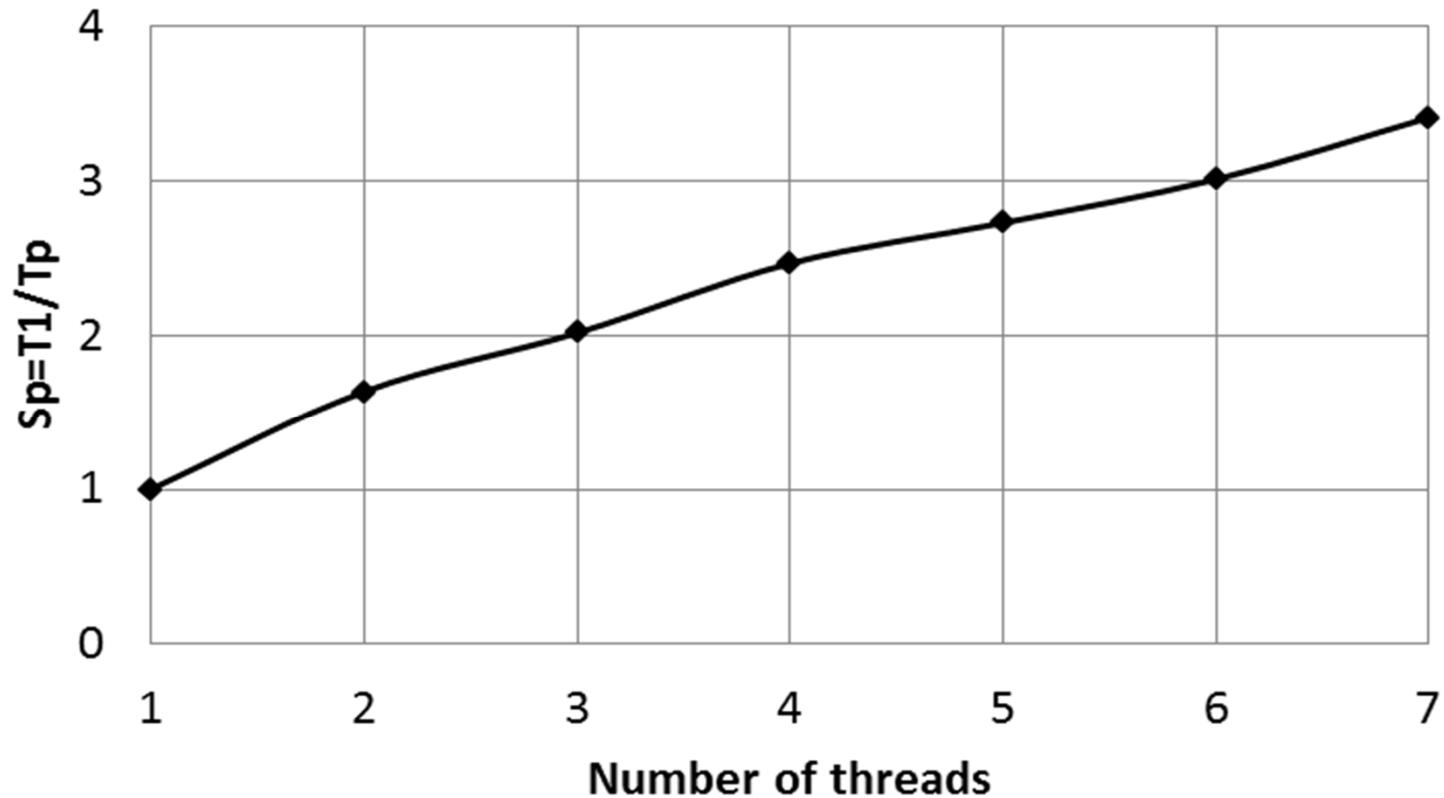
BSMFM: 391 с на 8 потоках, режим ОП (core mode)

PSICCG : 256 с на 7 потоках ($\psi = 10^{-10}$ и $\text{tol} = 10^{-4}$)

ICCG0 : 3 352 с на 7 потоках ($\text{tol} = 10^{-4}$)

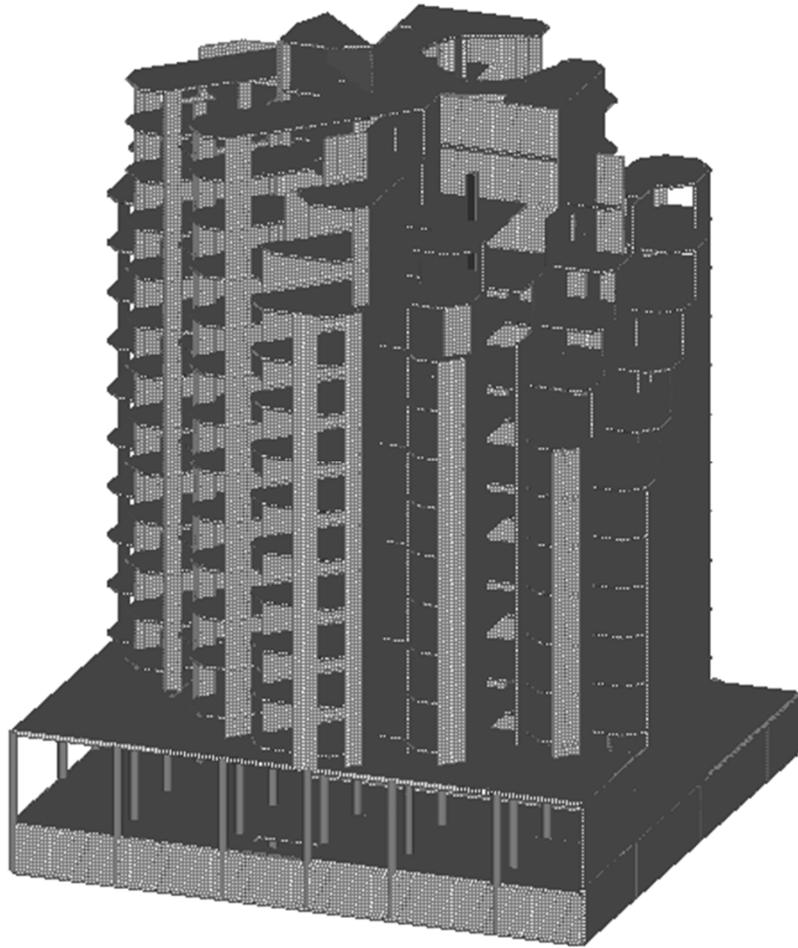


Численные результаты

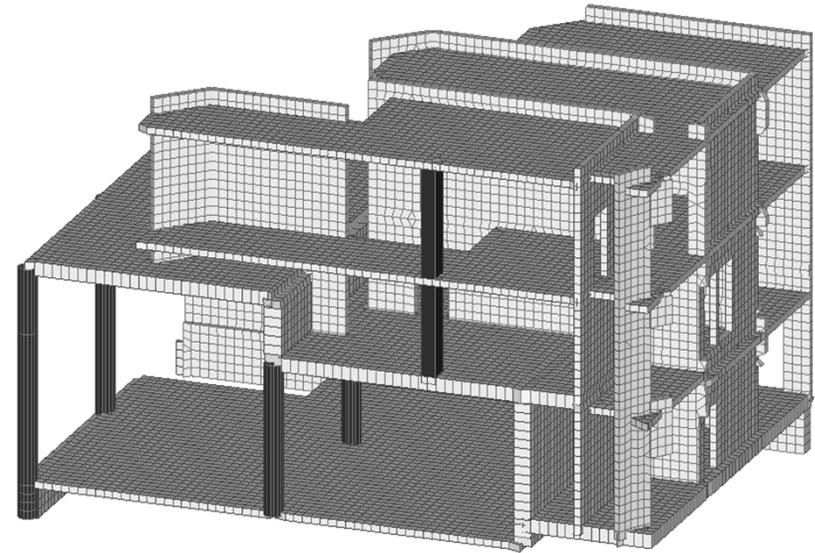


Ускоряемость метода PSICCG для задачи «Ракетный бульвар» при увеличении к-ва потоков. Компьютер с процессором Intel Xeon X5660 .

Численные результаты



**Atrium 4_1, расчетная модель
офисного здания (7,328,394 уравнений,
5 загрузжений)**



Фрагмент расчетной модели

Численные результаты

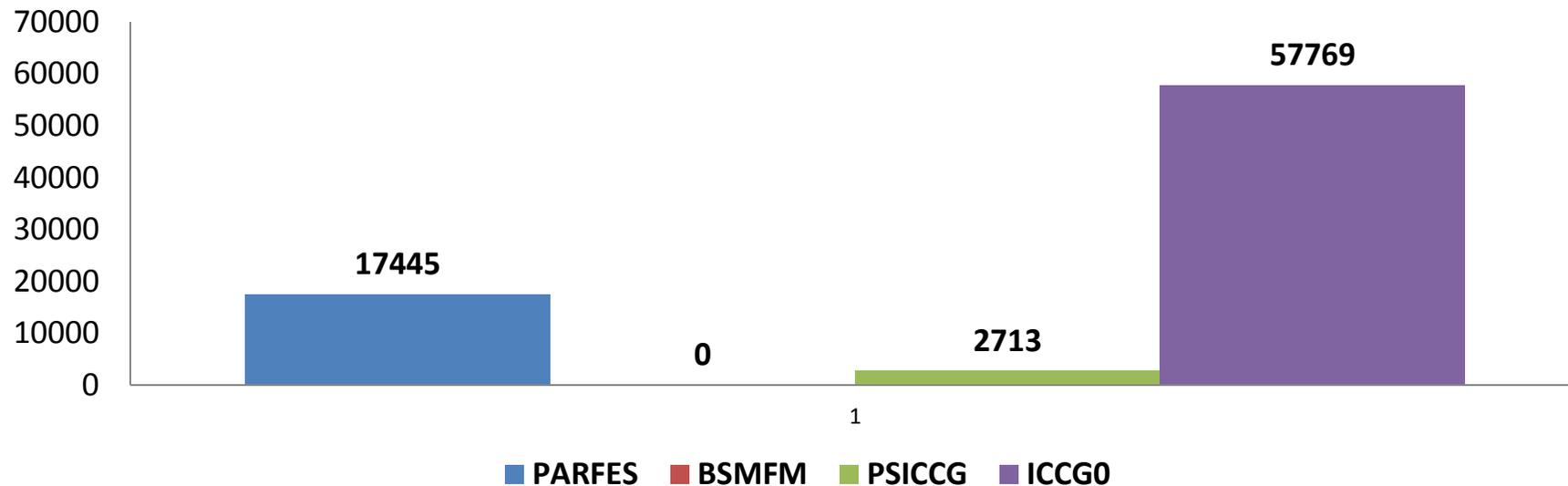
Продолжительность решения задачи Atrium 4_1 на компьютере с процессором AMD Phenom™ II x4 995, RAM DDR3 16 GB, 5 загрузений.

PARFES : 17 445 с на 4 потоках, режим OOC1

BSMFM : не решено – мало ОП

PSICCG : 2 713 с на 4 потоках ($\psi = 10^{-11}$, $\psi_1 = 10^{-7}$, $tol = 10^{-4}$)

ICCG0 : 57 769 с на 4 потоках ($tol = 10^{-4}$)



Численные результаты

Продолж. решения задачи Atrium 4_1 на рабочей станции с процессорами Intel Xeon X5660 (12 ядер), RAM DDR3 24 Gb, 5 загрузж.

PARFES : 2 699 с на 10 потоках, режим OOC

BSMFМ : 6 048 с на 10 потоках, режим OOC

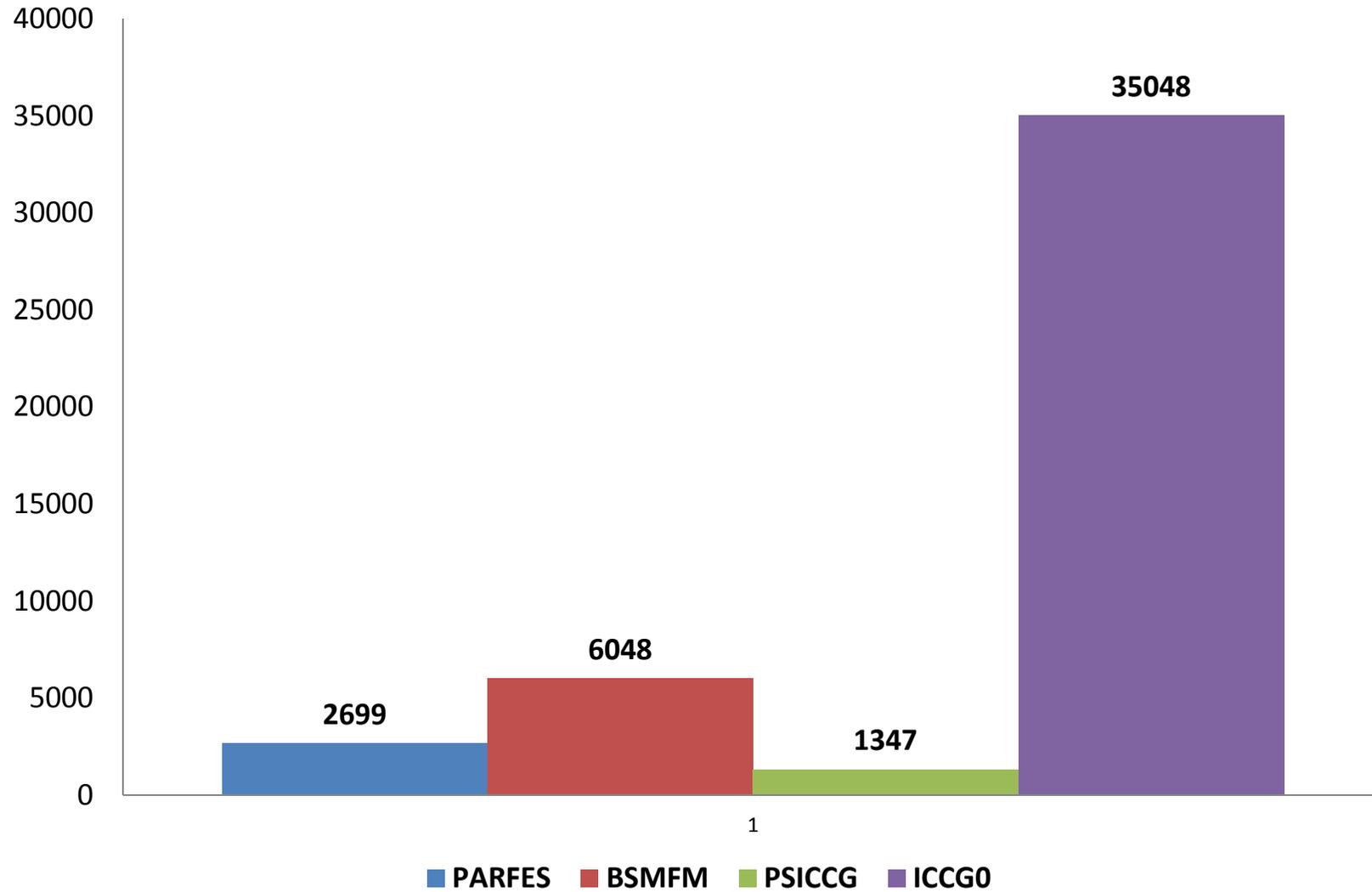
PSICCG : 1 347 с на 5 потоках ($\psi = 10^{-12}$, $\psi_1 = 10^{-7}$ и $tol = 10^{-4}$)

ICCG0 : 35 048 с на 5 потоках ($tol = 10^{-4}$)

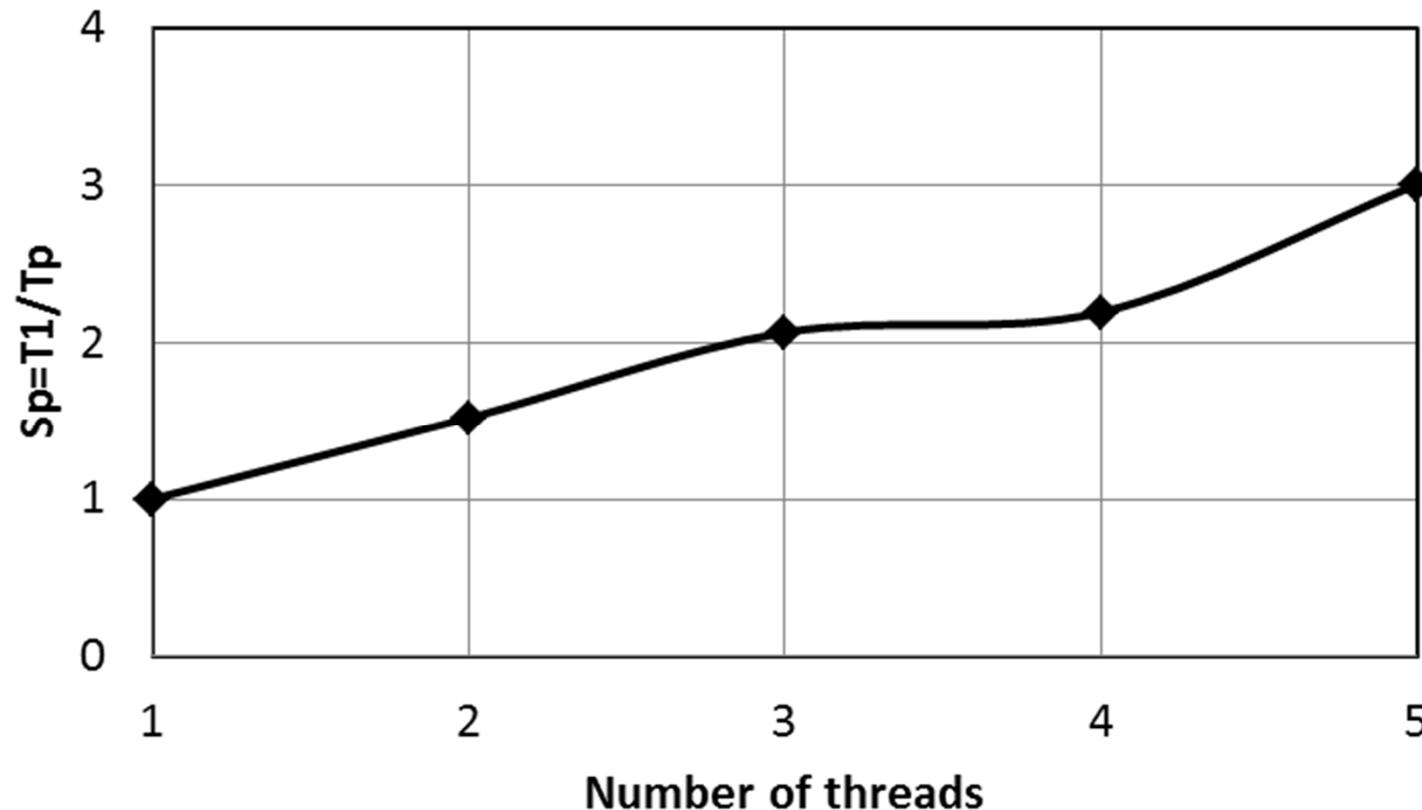
К-во итераций для каждого загрузжения (метод PSICCG) :
188/140/179/178/174

К-во итераций для каждого загрузжения (метод ICCG0) :
10 521/15 149/15 371/16 136/14 969

Численные результаты

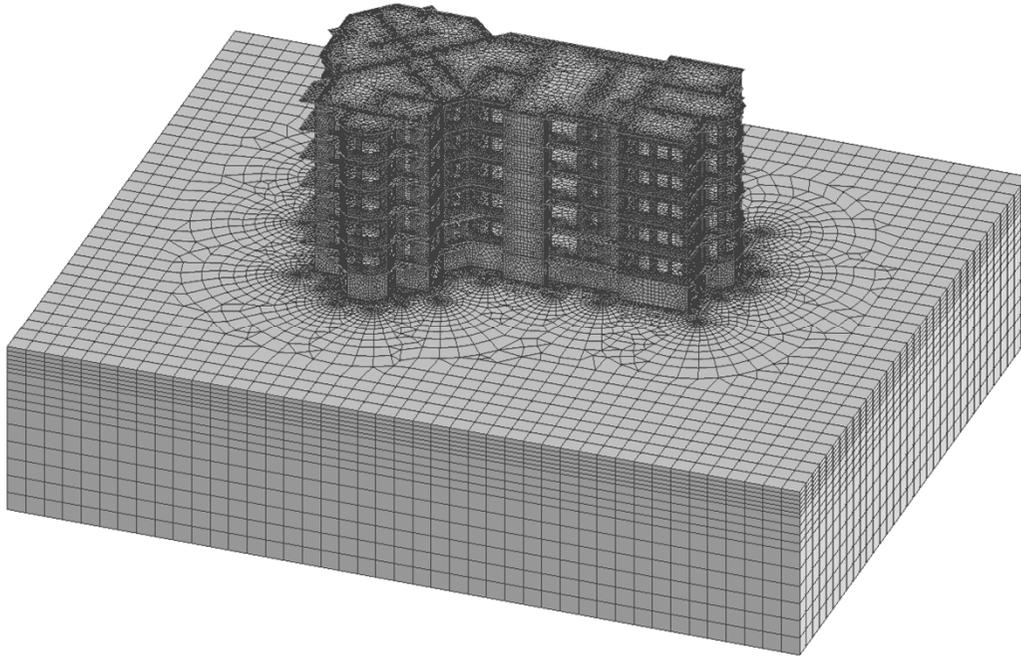


Численные результаты

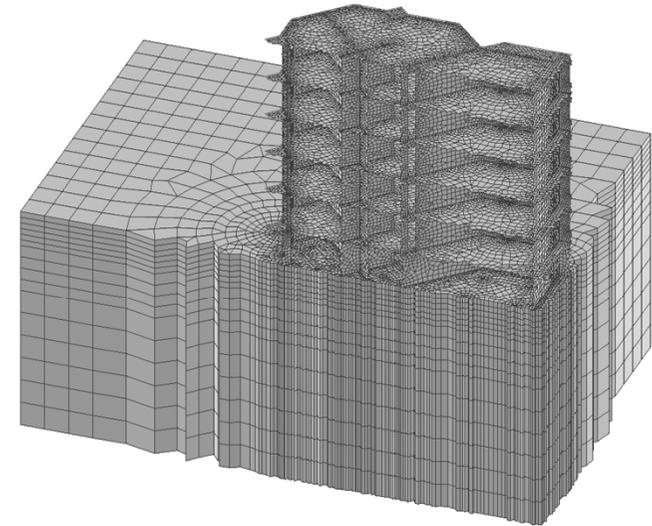


Ускоряемость метода PSICCG для задачи Atrium_4_1 при увеличении к-ва потоков. Компьютер с процессором Intel Xeon X5660.

Численные результаты



Расчетная модель Oster-RS-34-PS2-36KE. Взаимодействие здания с основанием. (1,929,544 уравнений, 4 загрузки)



Фрагмент расчетной модели

Численные результаты

Продолжительность решения задачи Oster-RS-34-PS2-36KE на компьютере с процессором Intel® Core™ i7-27600QM, RAM 8GB, 5 загрузений .

PARFES : 1 079 с на 4 потоках, режим OOC

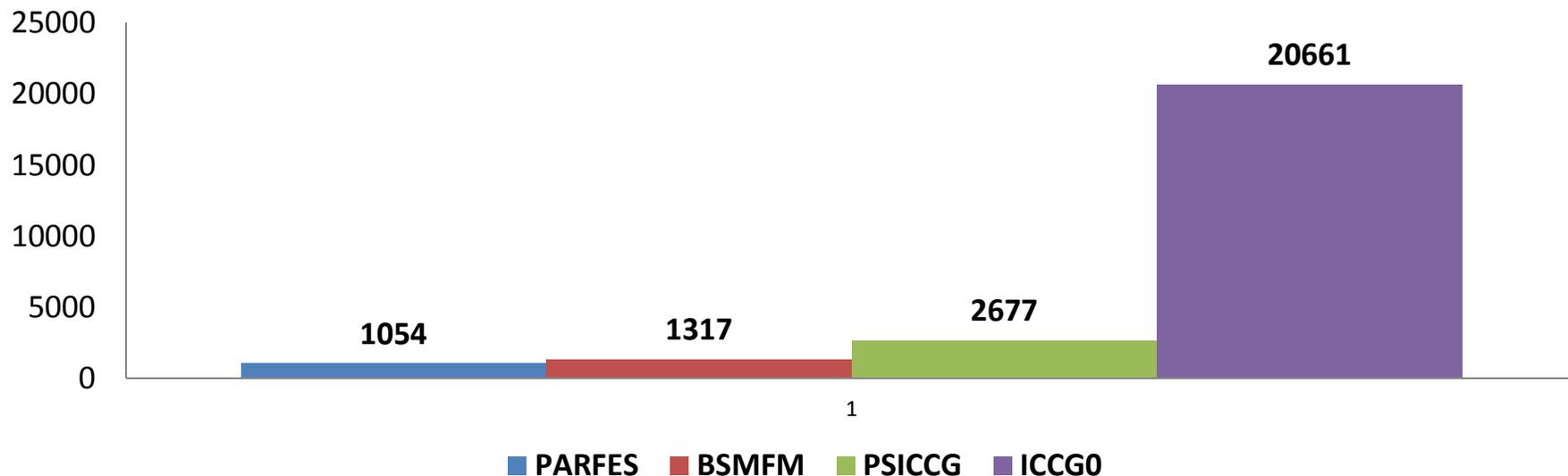
BSMFM: 1 649 с на 4 потоках, режим OOC

PSICCG : 2 293 с на 4 потоках, режим ОП ($\psi = 10^{-9}$, $\psi_1 = 10^{-7}$, $tol = 10^{-4}$)

ICCG0 : 12 842 с на 4 потоках, режим ОП ($tol = 10^{-4}$)

К-во итераций для метода PSICCG: 591/607/604/607/591 .

К-во итераций для метода ICCG0: 13 429/13 429/13 493/13 627/13 493



Численные результаты

Продолжительность решения задачи Oster-RS-34-PS2-36KE на компьютере с процессором AMD Phenom™ II x4 995 , RAM 16 GB. 5 загрузений.

PARFES : 1 054 с на 4 потоках, режим OOC

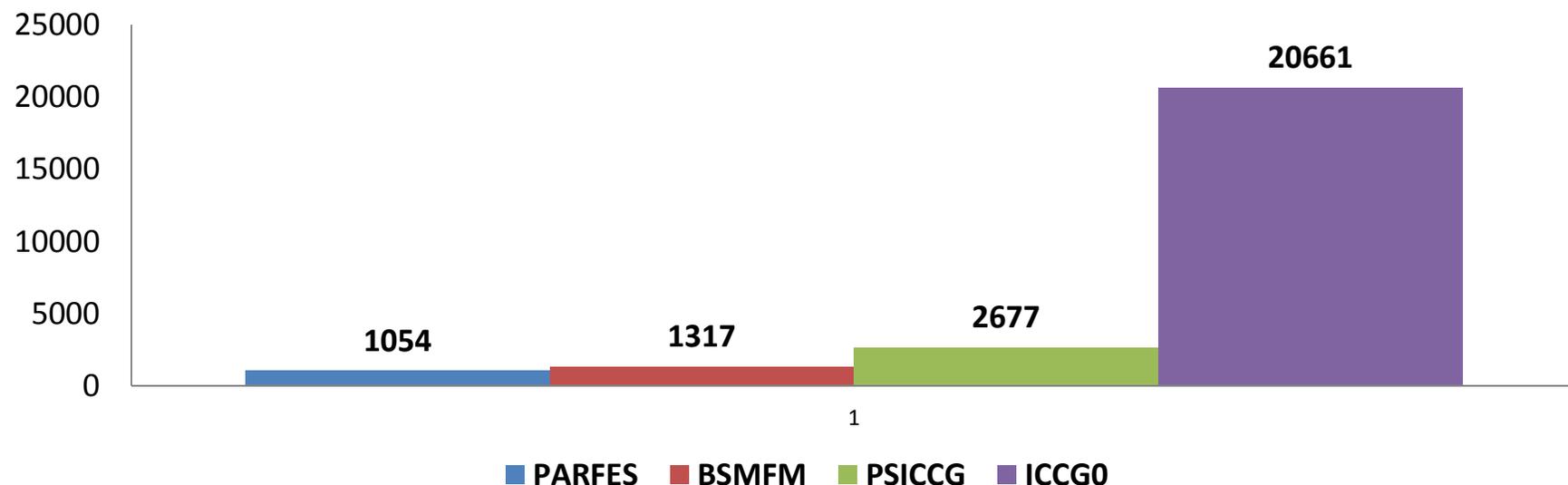
BSMFM : 1 317 с на 4 потоках, режим OOC

PSICCG : 2 677 с на 4 потоках ($\psi = 10^{-10}$, $\psi_1 = 10^{-7}$, $\text{tol} = 10^{-4}$)

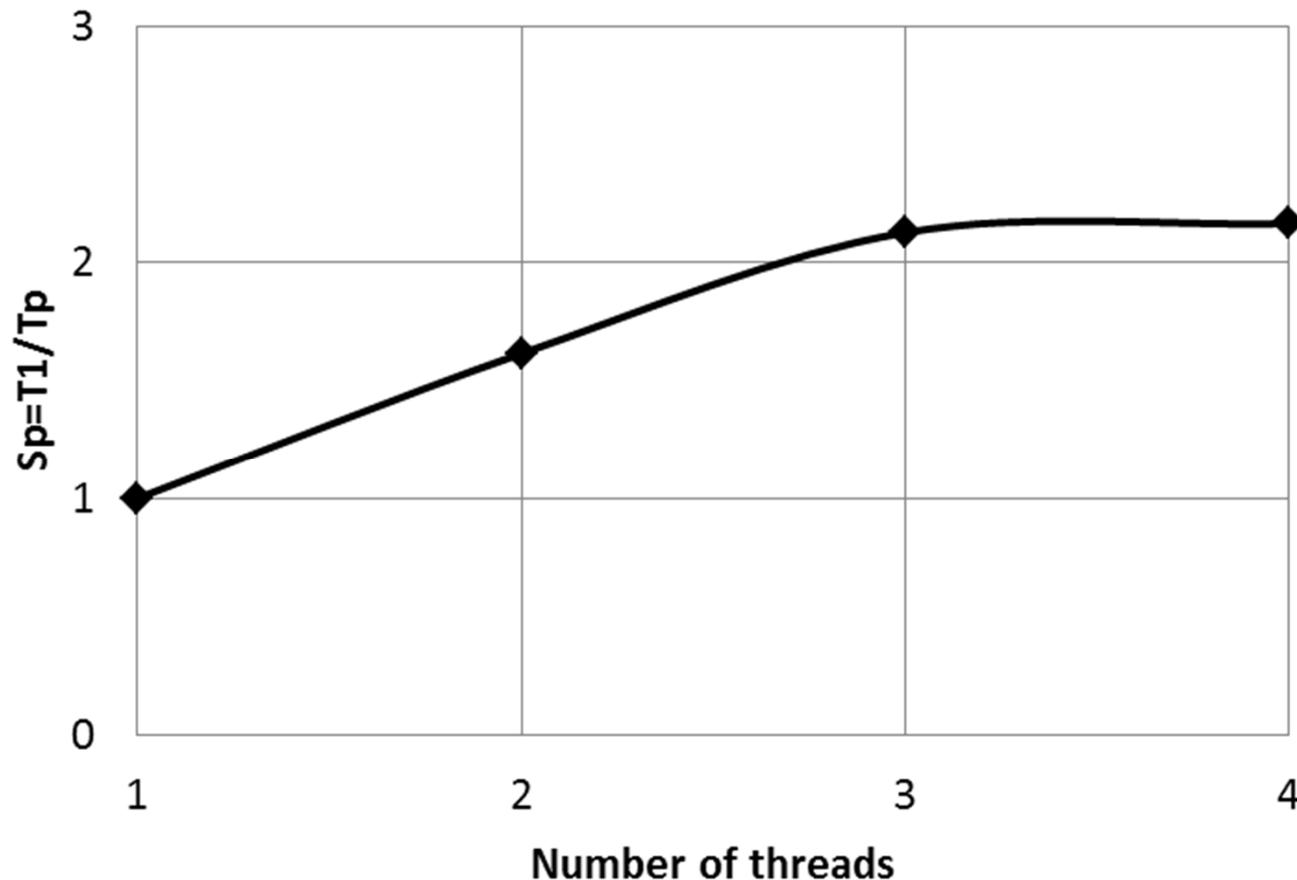
ICCG0 : 20 661 с на 4 потоках ($\text{tol} = 10^{-4}$)

К-во итераций для метода PSICCG: 396/393/392/393/392 .

К-во итераций для метода ICCG0: 13 429/13 429/13 493/13 627/13 493



Численные результаты



**Ускоряемость метода PSICCG для задачи Oster-RS-34-PS2-36KE.
при увеличении к-ва потоков. Компьютер с процессором AMD
Phenom™ II x4 995, RAM DDR3 16 GB**

ЗАКЛЮЧЕНИЕ

- 1. Для большеразмерных расчетных моделей многоэтажных зданий PSICCG демонстрирует быструю, устойчивую сходимость и приемлемую ускоряемость на настольных многоядерных компьютерах и ноутбуках.**
- 2. Данный круг задач является очень тяжелым для итерационных методов вследствие значительного разброса жесткостей, использования разнотипных элементов, сложности геометрии и наличия конечных элементов специального назначения. Перечисленные выше факторы приводят к плохой обусловленности задачи, поэтому классические итерационные методы оказываются мало эффективными.**

ЗАКЛЮЧЕНИЕ

- 3. Сравнение времени решения PSICCG и PARFES методов показывает, что если факторизованная матрица жесткости помещается в ОП, то обычно PARFES демонстрирует более быстрое решение. Однако если размерность задачи превышает объем ОП, то PARFES переключается в режим ООС или ООС1. При этом в большинстве случаев итерационный метод PSICCG становится более эффективным.**
- 4. На многоядерных настольных компьютерах PARFES демонстрирует более высокую производительность и ускоряемость, чем BSMFM.**

ЗАКЛЮЧЕНИЕ

- 5. При анализе большеразмерных задач решающее значение имеет объем оперативной памяти компьютера, доступное количество ядер (процессоров) и поддержка опций высокой производительности (AVX).**