

# Анализ эффективности ряда параллельных итерационных методов решения СЛАУ в упругопластической задаче на кластерной системе<sup>1</sup>

А.В. Толмачев, А.В. Коновалов, А.С. Партин

Институт машиноведения УрО РАН

В процессе решения упругопластической задачи возникает необходимость решить систему линейных уравнений большой размерности. Данная система является ленточной и разреженной внутри ленты. Её решение прямыми методами нецелесообразно, так как требуется большой объём памяти из-за образующегося заполнения. В данной работе описаны результаты вычислительного эксперимента и анализа производительности параллельных итерационных методов: релаксации (SOR), бисопряжённых градиентов (BiCGStab) и обобщённый метод минимальных невязок (GMRES).

## 1. Введение

Упругопластическая задача с большими пластическими деформациями физически и геометрически существенно нелинейная [1] и требует большого количества времени для её решения методом конечных элементов на персональном компьютере. Для сокращения времени расчётов необходимо использовать параллельные вычисления, в частности, на кластерных системах.

Решение методом конечных элементов упругопластической задачи осуществляется в условиях пошагового нагружения и на каждом таком шаге состоит из трёх основных этапов [1]:

1) расчёт локальных матриц жёсткости для конечных элементов и формирование глобальной матрицы жёсткости  $A$  и вектора  $b$  правой части системы линейных алгебраических уравнений (СЛАУ)

$$Ax = b \tag{1}$$

относительно искомого вектора  $x$  обобщённой скорости в узлах конечно-элементной сетки;

2) решение СЛАУ (1);

3) вычисление напряжённо-деформированного состояния конечных элементов в конце шага нагружения.

На каждом шаге нагружения этап 1 выполняется один раз, а этапы 2 и 3 — от десяти до пятнадцати раз для удовлетворения итерационно с приемлемой точностью условию пластичности в конце шага нагружения. При этом матрица жёсткости остается постоянной, а изменяется только правая часть системы уравнений. Матрица  $A$  является ленточной матрицей большой размерности и разреженной внутри ленты.

Производительность различных прямых методов решения СЛАУ в упругопластической задаче рассматривалась в работах [2-4]. Использование прямых методов для решения СЛАУ приводит к образованию заполнения внутри ленты и большим затратам по как памяти, так и по времени счёта.

Целью данной работы является исследование возможностей итерационных параллельных алгоритмов метода релаксации [5], стабилизированного метода бисопряжённых градиентов (BiCGStab) [6] и обобщённого метода минимальных невязок (GMRES) [5] для решения СЛАУ в упругопластических задачах на кластерной системе.

Ранее эффективность метода релаксации исследовалась в работе [7], однако там рассматривались матрицы системы из двухмерной упругопластической задачи с малой размерностью.

---

<sup>1</sup> Работа выполнена при поддержке программы фундаментальных исследований Президиума РАН №15, проект 12-П-1-1025

## 2. Используемые методы решения СЛАУ

*Формы хранения матрицы системы.* При формировании матрицы  $A$  использовали координатную форму хранения [5] с хранением троек из номеров строк, номеров столбцов и элементов матрицы в красно-коричневом дереве [8]. Для выполнения вычислений над матрицей, она переводится в сжатую по строкам форму [5]. Использование красно-коричневого дерева позволяет выполнить преобразование из координатной в сжатую форму без сортировок за единственный обход дерева.

*Метод релаксации.* Метод релаксации является одним из базовых итерационных методов линейной алгебры. Он является улучшением метода Гаусса-Зейделя [5] путём добавления коэффициента релаксации  $0 < \omega < 2$ . При значении коэффициента  $\omega = 1$  метод релаксации сводится к методу Гаусса-Зейделя. Структура метода релаксации приведена на рисунке 1.

Повторять до схождения

for  $i = 1, \dots, n$

$$\dot{x} = \frac{1}{a_{ii}} \left( b_i - \sum_{i \neq j} a_{ij} x_j \right)$$

$$\delta = \dot{x} - x_i$$

$$x_i = x_i + \omega \delta$$

**Рис. 1.** Алгоритм метода релаксации

*Метод BiCGStab.* Метод BiCGStab является вариантом метода сопряжённых градиентов для несимметричных матриц. Он относится к семейству методов, основанных на подпространствах Крылова. Рассматривается СЛАУ  $Ax = b$  с начальным приближением  $x_0$ . Здесь  $(a, b)$  означает вычисление скалярного произведения векторов  $a$  и  $b$ . Алгоритм непередобусловленного метода BiCGStab представлен на рисунке 2.

$$r_0 = b - Ax_0$$

$$\rho_0 = (r_0, r_0)$$

$$p_0 = r_0$$

for  $j = 0, 1, \dots$

$$v = Ap_j$$

$$\alpha_j = \rho_j / (v, r_0)$$

$$s_j = r_j - \alpha_j v$$

$$t = As_j$$

$$\omega_j = \frac{(t, s_j)}{(t, t)}$$

$$x_{j+1} = x_j + \alpha_j p_j + \omega_j s_j; \text{ проверка на сходимость}$$

$$r_{j+1} = s_j - \omega_j t$$

$$\rho_{j+1} = (r_{j+1}, r_0)$$

$$\beta_j = \frac{\rho_{j+1} \alpha_j}{\rho_j \omega_j}$$

$$p_{j+1} = r_{j+1} + \beta_j (p_j - \omega_j v)$$

end

**Рис. 2.** Алгоритм непередобусловленного метода BiCGStab

Для уменьшения количества итераций, необходимых для достижения сходимости метода до заданной точности, использовали вариант метода с предобуславливателем [5].

*Обобщённый метод минимальных невязок.* Обобщённый метод минимальных невязок так же относится к семейству методов, основанных на подпространствах Крылова. Его подробное описание приведено в [5].

*Предобуславливатель.* Для уменьшения количества итераций, необходимых для схождения итерационных методов BiCGStab и GMRES использовали предобуславливатель, основанный на процессе неполного LU-разложения без заполнения (ILU(0)) [5].

### 3. Характеристики матрицы системы

Для упрощения анализа эффективности рассматриваемых методов решения СЛАУ использовали регулярную сетку при разбиении тела на конечные элементы.

При решении трёхмерной задачи с регулярной сеткой имеет место ленточная матрица с пятью ненулевыми лентами. Параметры матрицы системы в зависимости от количества разбиений  $d$  по каждой координатной оси в конечноэлементной сетке приведены в таблице. Здесь  $n$  - размерность матрицы,  $\beta$  - полуширина ленты,  $z$  - количество элементов в матрице, удовлетворяющих условию  $|a_{ij}| \geq 10^{-6}$ , которое на два порядка меньше критерия останова в методе Ньютона-Равсона, использованного при достижении условия пластичности,  $\delta$  - число обусловленности матрицы. Число обусловленности для матрицы оценивалось как отношение наибольшего к наименьшему сингулярных значений матрицы. Сингулярные значения матрицы оценивались при помощи функции svds пакета MATLAB. Для задач с сеткой  $d = 30, 40$  и  $50$  сингулярные значения оценивались с использованием библиотеки SLEPc [9] с критерием останова  $10^{-3}$ .

**Таблица 1.** Параметры матрицы в трёхмерной упругопластической задаче

$d$	$n$	$\beta$	$z$	$\delta/10^3$
10	3993	401	234864	1,14
20	27783	1391	1910269	2,34
30	89373	2981	5856367	4,2
40	206763	5171	13630871	8,1
50	397953	7961	30080934	10,6

### 4. Результаты вычислительных экспериментов

Эффективность итерационных методов для решения упругопластической задачи определяли по результатам вычислительного эксперимента по сжатию параллелепипеда плоскими плитами.

Вычислительные эксперименты проводили на кластере «Уран» Института математики и механики УрО РАН. Он состоит из 208 вычислительных узлов, установленных в модулях с высокой плотностью упаковки. Каждый вычислительный узел оснащен 2-мя процессорами Intel Quad-Core Xeon, работающими на частоте 3.00 ГГц, и 16/32 гигабайтами оперативной памяти. В общей сложности пользователям доступно 1664 вычислительных ядра и 3584 Гбайт оперативной памяти. Для передачи данных между вычислительными узлами используется высокоскоростная сеть Infiniband с пропускной способностью 20 Гбит/сек.

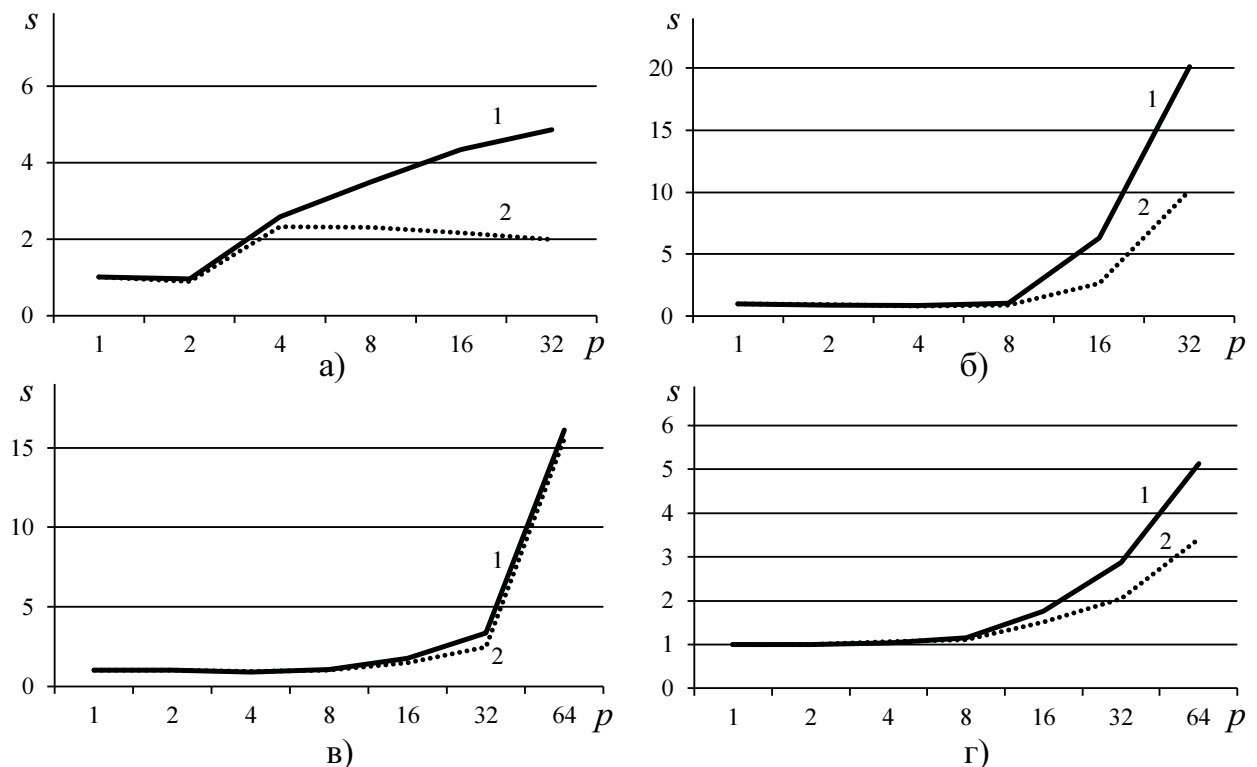
В качестве реализаций итерационных методов решения СЛАУ была использована библиотека lis [10]. Решение СЛАУ производилось до удовлетворения условия останова  $\frac{\|r\|}{\|b\|} < 10^{-9}$ ,

где  $r$  - вектор невязки,  $b$  - вектор правой части,  $\|\bullet\|$  - 2-норма вектора.

Анализ результатов, полученным методом релаксации показал, что при значениях коэффициента релаксации  $\omega > 1$  на сетках с параметром  $d > 15$  метод расходится. При значениях  $\omega < 1$

метод релаксации сходится, но решение СЛАУ производится значительно медленнее, чем при использовании методов BiCGStab и GMRES.

На рисунке 3 представлена зависимость ускорения  $s$  методов BiCGStab и GMRES от количества используемых MPI-процессов  $p$  для решения СЛАУ.

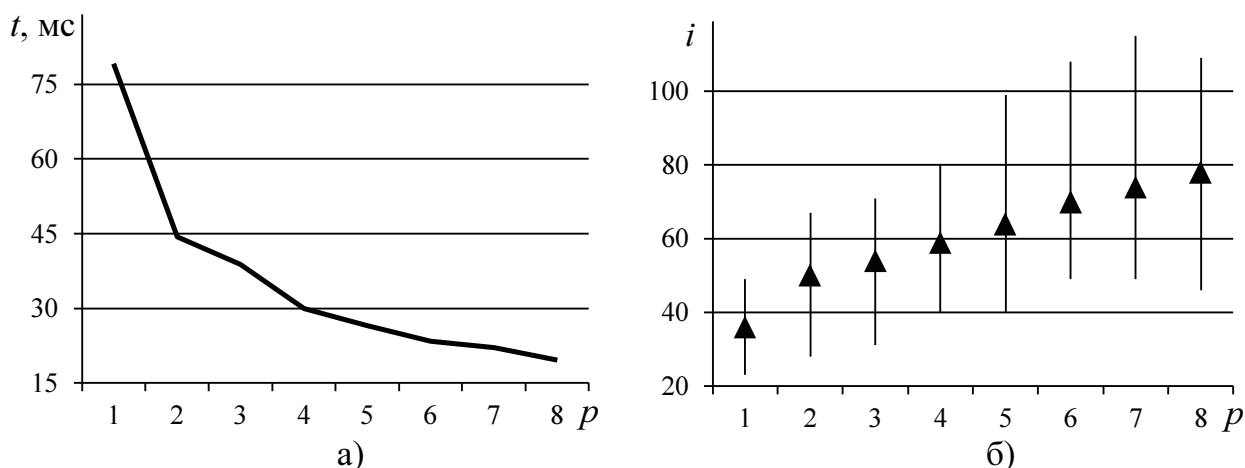


**Рис. 3.** Зависимость ускорения  $s$  решения СЛАУ от количества MPI-процессов  $p$  для сеток с параметром  $d$  равным 10 (а), 20 (б), 30 (в) и 50 (г): 1 - метод BiCGStab, 2 - метод GMRES

На всех рассмотренных сетках метод BiCGStab показал лучшее ускорение по сравнению с методом GMRES. Для сетки с  $d = 10$  (см. рис. 3,а) у метода GMRES после при использовании более 4 процессов имеет место замедление вычислений за счёт времени на передачу данных. А в методе BiCGStab данный эффект отсутствует. Для сеток наблюдается скачкообразный рост ускорения в трёх случаях, а именно: для сетки с  $d = 10$  при переходе с 2 на 4 процесса, для сетки с  $d = 20$  при переходе с 16 на 32 процесса и для сетки с  $d = 30$  при переходе с 32 на 64 процесса. Этот эффект обусловлен тем, что матрица системы уравнений начинает полностью помещаться в процессорный кэш. Характер графика для сетки  $d = 40$  аналогичен графику для сетки  $d = 50$ . При этом ускорение метода BiCGStab достигало 5,8 на 64 процессах.

На рисунке 4,а показаны график зависимости среднего времени, затрачиваемого на одну итерацию от количества процессоров на задаче с сеткой  $d = 30$ . На рисунке 4,б показана зависимость наибольшего, среднего и наименьшего количества итераций при решении СЛАУ.

Уменьшение времени на итерацию одновременно с увеличением количества итераций приводит к тому, что на отрезке от 1 до 8 процессов не происходит существенного ускорения (см. рис. 3). Отсутствие ускорения на рассматриваемом отрезке объясняется тем, что используемая реализация предобуславливателя ILU(0) в библиотеке `lis` не производит, по нашему мнению, межпроцессорный обмен. Поэтому при увеличении количества процессов требуется большее количество итераций для достижения требуемой точности. При использовании более 8 процессов время на итерацию уменьшается быстрее, чем увеличивается количество итераций, поэтому общее время на решение СЛАУ уменьшается. Применение других предобуславливателей может существенно повлиять на данный эффект.



**Рис. 4.** Зависимость для метода BiCGStab от количества MPI-процессов  $p$  времени  $t$  на одну итерацию (а), наибольшее, среднее (маркеры) и наименьшее количество итераций при решении СЛАУ (б)

Метод GMRES показывает лучшее абсолютное время решения СЛАУ на всех рассматриваемых сетках на небольшом количестве процессоров, однако при увеличении количества процессоров метод BiCGStab начинает быть быстрее, так как он требует меньше операций передачи данных. Например, для решения СЛАУ в задаче с сеткой  $d = 40$  1 процессом методу BiCGStab требуется затратить 17,5 секунд, а методу GMRES — 13,1 секунды. Однако при использовании 64 процессоров методу BiCGStab требуется затратить 2,8 секунды, а методу GMRES — 3,4 секунды.

Применение прямых методов решения СЛАУ в упругопластической задаче методами из работ [3,4] не позволяло решать на кластере "Уран" ИММ УрО РАН задачи с сетками с  $d > 35$  поскольку для хранения матрицы жёсткости требовалось много оперативной памяти. Применение разреженных схем хранения матрицы в совокупности с предобусловленными итерационными методами позволило сократить затраты на хранение матрицы жёсткости и дало возможность решать задачу с большим разбиением.

## Выводы

1. Итерационные методы позволили решать упругопластическую задачу методом конечных элементов с большим разбиением, чем прямые методы.
2. Метод релаксаций по сравнению с методами BiCGStab и GMRES затрачивает значительно большее время решение СЛАУ.
3. Метод GMRES показывает лучшее абсолютное время при решении СЛАУ на небольшом количестве процессоров, однако метод BiCGStab показывает лучшее ускорение и при использовании большого количества процессоров показывает лучшее абсолютное время.

## Литература

1. Поздеев А.А., Трусков П.В., Няшин Ю.И. Большие упруго-пластические деформации. М.: Наука, 1986. 232 с.
2. Толмачев А.В. Коновалов А.В. Партин А.С. Использование усечённого варианта алгоритма SPIKE из библиотеки Intel Adaptive SPIKE-Based Solver для решения упругопластических задач // Вычислительные методы и программирование: новые вычислительные технологии. 2011. – Т. 12. – № 1. – С. 154-159.
3. Коновалов А.В., Толмачев А.В., Партин А.С. Параллельное решение упругопластической задачи с применением трехдиагонального алгоритма LU-разложения из библиотеки SCA-LAPACK // Вычислительная механика сплошных сред. – 2011. – № 4. С. 34-41.

4. Толмачев А.В., Коновалов А.В., Партин А.С.. Применение распределённого по замкнутому тору алгоритма LU-разложения к решению упругопластической задачи (с. 745) // Параллельные вычислительные технологии (ПаВТ'2012): труды международной научной конференции (Новосибирск, 26 – 30 марта 2012 г.) [Электронный ресурс] – Челябинск: Издательский центр ЮУрГУ, 2012. – 774 с.
5. Saad Y. Iterative methods for sparse linear systems, SIAM 2003. С. 447.
6. H. A. van der Vorst, BI-CGSTAB: a fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 13 (2) (1992) 631–644. doi:10.1137/0913035
7. Демешко И.П., Акимова Е.Н., Коновалов А.В. Применение параллельных алгоритмов для решения системы линейных алгебраических уравнений с ленточной матрицей итерационными методами на кластерной системе // Труды международной конференции "Параллельные Вычислительные технологии ( ПаВТ'2009)", Нижний Новгород, 30 марта – 3 апреля 2009. – С. 444 – 449. – Челябинск: Изд. ЮУрГУ, 2009. (электронное издание).
8. Кормен Т.Х., Лейзерсон Ч.И., Ривест Р.Л., Штайн К. Алгоритмы: построение и анализ. – М.: Вильямс, 2005. – 1296 с.
9. Vicente Hernandez, Jose E. Roman, Vicente Vidal. SLEPc: A Scalable and Flexible Toolkit for the Solution of Eigenvalue Problems // ACM Transactions on Mathematical Software 31(3), 2005. pp. 351-362
10. Akira Nishida. Lis User Manual. Research Institute for Information Technology – Fukuoka. Электронный ресурс. URL: <http://www.ssisc.org/lis/lis-manual-en.pdf> (дата обращения 15.11.2012)