

Анализ эффективности применения перенумерации узлов конечно-элементной сетки применительно к методу конечных элементов для решения трёхмерных задач механики деформируемого твёрдого тела

А.Р. Филатов

1. Введение

На сегодняшний день имеет место быстрое развитие вычислительных комплексов высокой производительности. Этот рост позволяет брать во внимание задачи всё большего объёма. Основным методом решения задач теории упругости остаётся метод конечных элементов (МКЭ). Для высокой точности решения требуется строить довольно мелкую сетку по пространству. Но решение систем линейных алгебраических уравнений (СЛАУ) для таких задач занимает всё больше времени, а, следовательно, логично предложить метод по упрощению матрицы для исходного СЛАУ, который снизит вычислительную затратность этого процесса.

Одним из способов решения этой проблемы является перенумерация узлов конечно-элементной (КЭ) сетки. Основные преимущества этого подхода:

- 1) Независимость от сложности исходной геометрии модели;
- 2) Отсутствие потерь точности при использовании;
- 3) Лёгкая встраиваемость в уже существующие алгоритмы решения задач упругости;
- 4) Возможность распараллеливания на несколько потоков.

2. Общий подход

Рассмотрим упругое тело, занимающее область $\Omega \in \mathbb{R}^{d_s}$, где $d_s = 2,3$ — размерность пространства. Построим на нём КЭ сетку любым

подходящим методом, как известно, получим граф (V, E) . Любой граф можно представить в виде матрицы смежности $A_{n \times n}$, причём если рёбра E графа (V, E) не ориентированы, то, очевидно, матрица A будет симметричной относительно главной диагонали.

Задача состоит в следующем: дана разреженная матрица A , симметричная относительно главной диагонали, необходимо перенумеровать узлы исходного графа (V, E) так, чтобы как можно ближе к оси симметрии концентрировались ненулевые элементы (другими словами, надо уменьшить профиль матрицы A). После чего требуется составить перестановку из вершин графа (V, E) и применить её к исходной расчётной КЭ сетке. Стоит помнить, с сеткой могут быть связаны другие сущности, относящиеся к этой задаче, например, граничные и начальные условия принято задавать на узлах КЭ сетки, а значит надо позаботиться и об их перенумерации.

Далее следуют шаги предусмотренные алгоритмом решения задачи. Данный подход очень полезен при решении задач больших деформаций, в которых существенно изменяется форма модели (задачи описаны в [9], [10]). При решении механических задач с большим количеством включений ([8]) или задач о композитных материалах ([6], [7]) нумерации не будет отражать геометрической особенности тела, соответственно этот подход будет необходим для ускорения процесса решения.

3. Описания работы алгоритмов

3.1.Алгоритм RCM

Алгоритм CM был разработан в 1969 году Elizabeth Cuthill и J. McKee, затем Alan George разработал его «обратную» модификацию [2]. Будем считать, что наша модель Ω была единым целым, а значит и граф (V, E) имеет лишь одну компоненту связности. Алгоритм построит упорядоченный кортеж вершин R , представляющий новую нумерацию вершин. В алгоритме

нами будет использоваться вспомогательная структура данных — очередь, будем обозначать её Q . Особенностью очереди, является порядок добавления и извлечения элементов: первый, добавленный в неё элемент, будет извлечён первым; а последний — последним.

1. Возьмём узел наименьшей степени, который не был ранее добавлен в R , назовём его P . Добавим P на первую свободную позицию R ;
2. Добавим в очередь Q все узлы смежные с P узлы в порядке возрастания их степеней;
3. Возьмём первый узел из очереди Q , пусть это будет узел C . Если C ранее не был в R , то добавим его в первую свободную позицию R ;
4. Добавим в очередь Q всех соседей узла C , ранее не принадлежащих R , в порядке возрастания их степеней;
5. Если очередь Q не пустая, то возвращаемся к шагу №3;
6. Изменяем порядок элементов кортежа R на обратный.

В итоге получаем перестановку R , в которой на k -ой позиции стоит номер узла (в старой нумерации), который в новой нумерации будет иметь номер k . Если же исходное тело Ω обладало несколькими компонентами связности, то этот алгоритм можно последовательно применить к каждой компоненте связности (а именно: перед переходом на шаг №6 переходить к шагу №1 последующей компоненты связности).

На самом деле результат работы данного алгоритма можно много улучшить, изменив правило поиска первого узла P . Чтобы получить хороший результат, в качестве P нам понадобится периферийная вершина графа (V, E) . Но эта задача для общего случая является вычислительно ёмкой, поэтому вместо неё будем искать псевдопериферийную вершину (используя модификацию эвристического алгоритма Гиббса [3]). В этом алгоритме

используется понятие корневой структуры уровней: для заданной вершины v структурой уровней с корнем в вершине v будет разбиение \mathcal{L} множества вершин V :

$$\mathcal{L}(v) = \{ L_0(v), L_1(v), \dots, L_{l(v)}(v) \}$$

Где подмножества $L_i(v)$ определены следующим образом:

$$L_0(v) = \{v\}$$

$$L_1(v) = Adj(L_0(v))$$

$$L_i(v) = Adj(L_{i-1}(v) \setminus L_{i-2}(v))$$

Где $i = 2, 3, \dots, l(v)$; $Adj(M)$ — множество смежности для M .

Теперь опишем сам алгоритм:

1. Выбираем произвольную вершину p из множества всех вершин V ;
2. Построим структуру уровней для корня p :

$$\mathcal{L}(p) = \{ L_0(p), L_1(p), \dots, L_{l(p)}(p) \};$$
3. Выберем вершину v с минимальной степенью из множества $L_{l(p)}(p)$;
4. Построим структуру уровней для корня v :

$$\mathcal{L}(v) = \{ L_0(v), L_1(v), \dots, L_{l(v)}(v) \};$$
5. Если $l(v) > l(p)$, то присвоить вершине $p := v$ и перейти к шагу №3;
6. Вершина v — искомая.

Таким образом, был описан алгоритм RCM для уменьшения профиля симметричных матриц, который с успехом можно применять к расчётным КЭ сеткам.

3.2. Алгоритм NodeND

NodeND был разработан George Karypis в 1995 году. Данный алгоритм основан на парадигме многоуровневого вложения разбиения, которая в свою очередь основана на алгоритме вычисления узла-разделителя для графа, соответствующего матрице [5]. Узлы-разделители перемещаются в конец матрицы, затем аналогичный процесс производится с каждой из двух полученных частей. Многоуровневая вложенная парадигма разбиения является достаточно эффективной для переупорядочения узлов, которое вытекает в матрицу с меньшим профилем. Этот алгоритм очень полезен, если при решении системы разреженных уравнений используется прямой метод. Полученная перестановка позволяет, как сократить время решения уравнения, так и уменьшить потребление памяти. Кроме того, это даст существенный выигрыш при параллельном использовании, так как достигается высокая степень параллелизма на этапе факторизации.

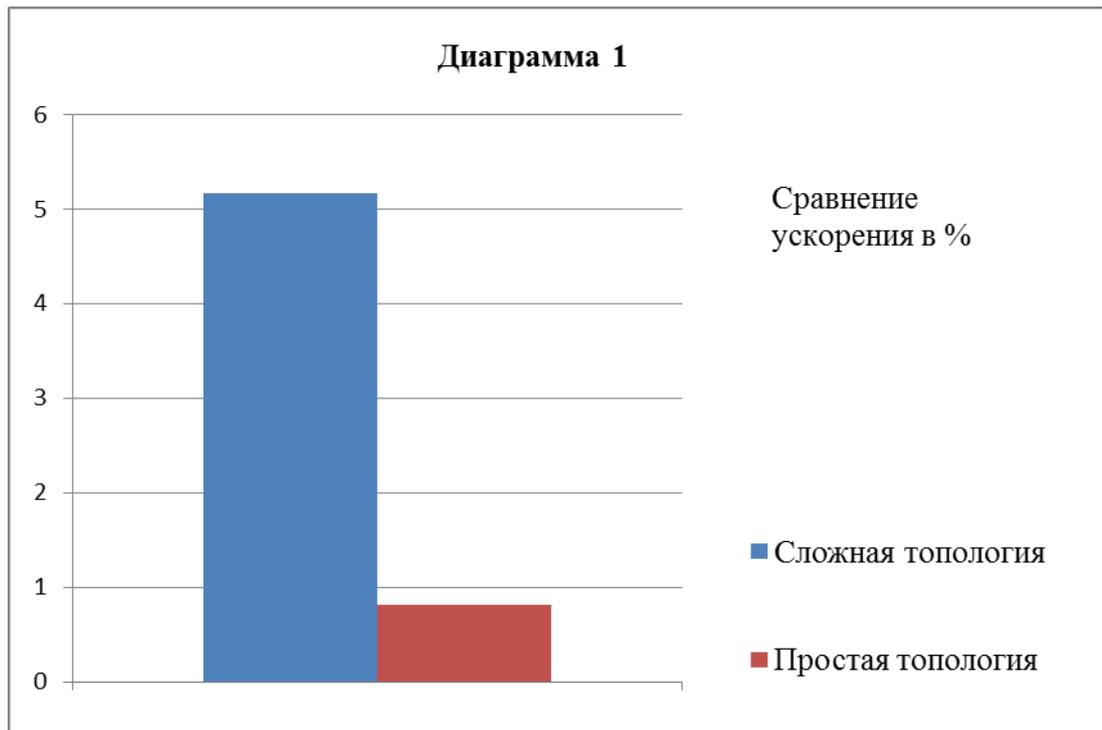
Но для использования NodeND, требуются низкоуровневые программные структуры созданные пользователем. Заполнение этих структур занимает дополнительное время, но не является вычислительно ёмким. Подробное описание этих структур можно найти в [4].

4. Результаты

RСМ тратит много памяти на хранение графа сетки, фактически сетка хранится в памяти дважды, что плохо с точки зрения ресурсо-экономии. Сложность задачи построения графа равна $O(e \cdot E)$; где E — количество элементов сетки, e — количество рёбер элемента. Другими словами, не экономный метод. NodeND же напротив, показывает очень хорошие результаты.

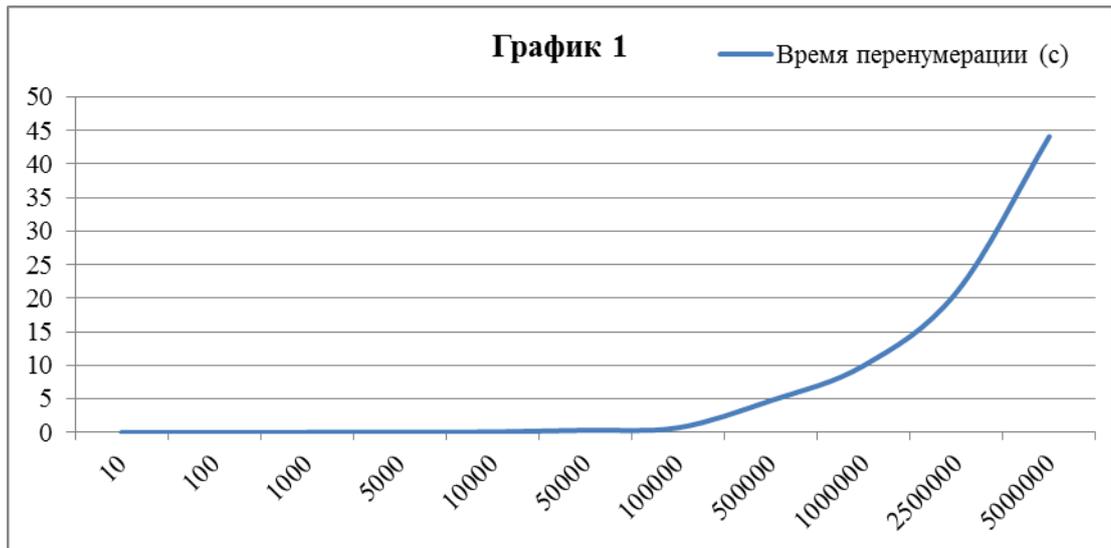
Тестирование проводилось на задачах статики, использовались тетраэдральные элементы. Рабочая станция обладала 16 GB оперативной

памяти (этим обусловлено ограничение в размере задачи — 5 млн. элементов) и 4х ядерным процессором (использовалось ускорение, путём распараллеливания OMP). Наиболее заметный эффект наблюдается на задачах со сложной топологией тела Ω или с большим соотношением сторон (то есть вдоль одной координаты тело очень вытянуто). Диаграмма 1 доказывает это утверждение.

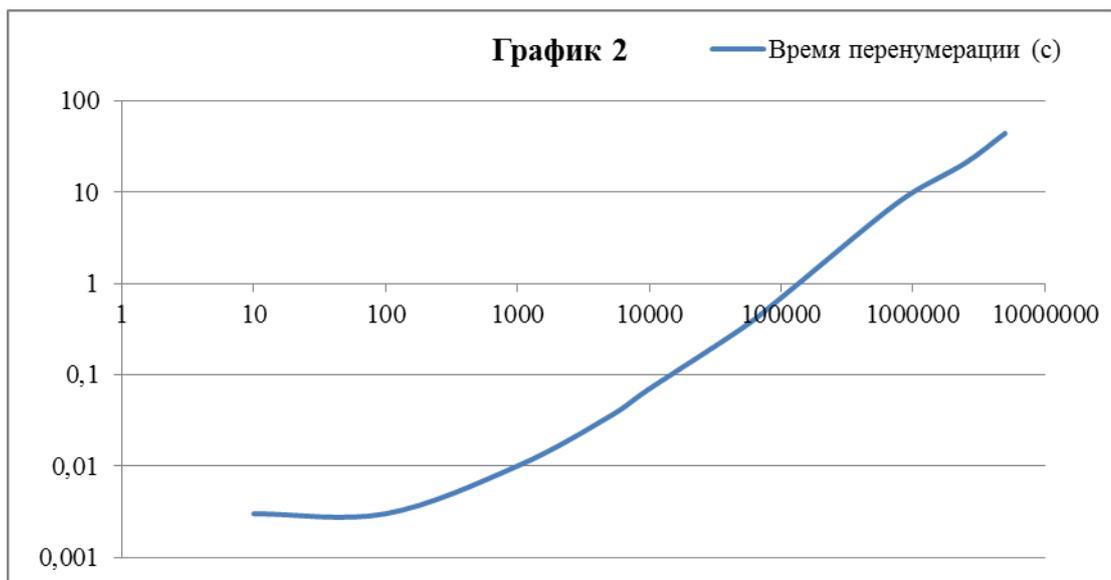


Стоит заметить, что перенумерация тратит значительно меньше памяти, нежели само решение, то есть если задача без перенумерации помещается в память, то и с перенумерацией поместится.

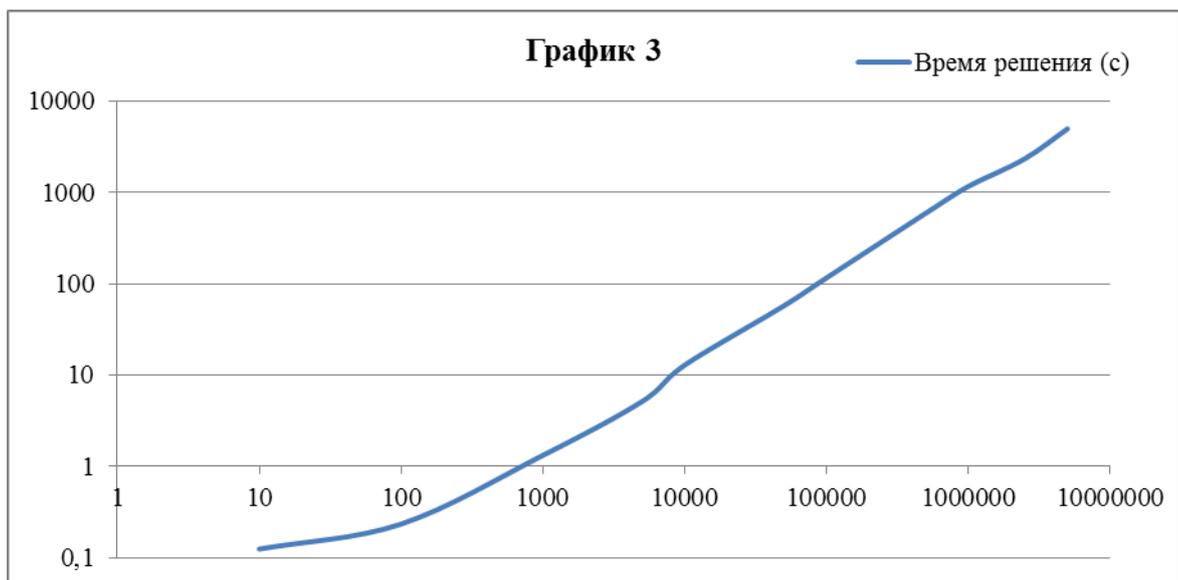
Далее приведён график 1, на котором отображено время, затрачиваемое на перенумерацию в зависимости от количества элементов.



Стоит обратить внимание, что по оси абсцисс масштаб не линейный, но в таком масштабе лучше виден результат. Для сравнения ниже приведён этот же график 2 в логарифмическом масштабе.



Если сравнить его с графиком 3, изображающим зависимость времени решения задачи (перенумерованной, без учёта времени перенумерации) от количества элементов, то видно, что перенумерация NodeND занимает не значительное время. Масштаб по всем осям логарифмический.



Но эти данные актуальны лишь при сравнении выигрыша от перенумерации и времени самой перенумерации ($t_{advant} - t_{renum}$), график 4 приведён ниже.

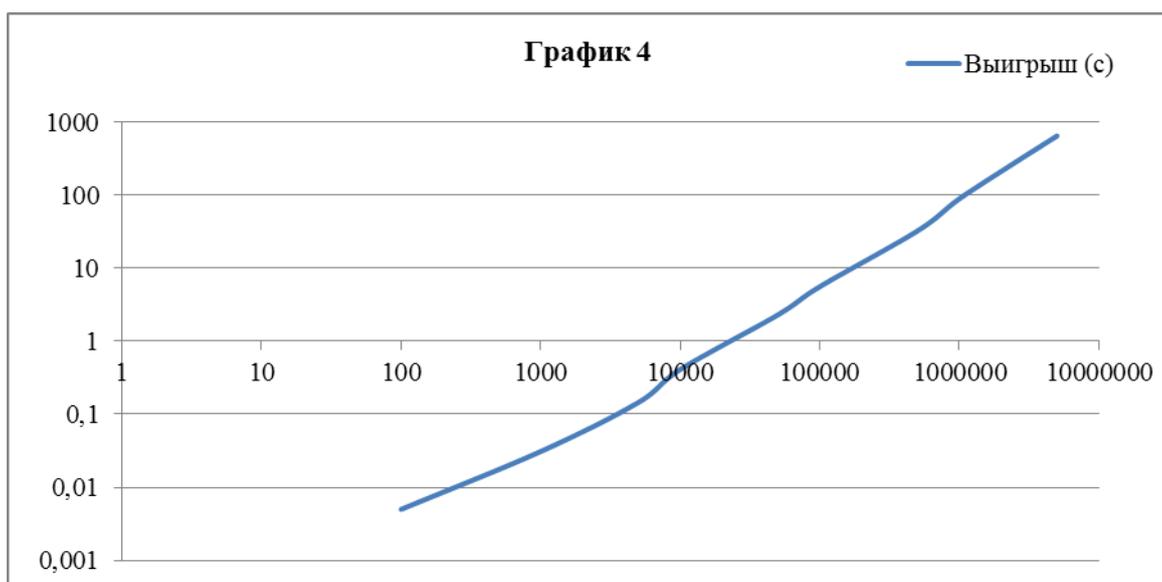
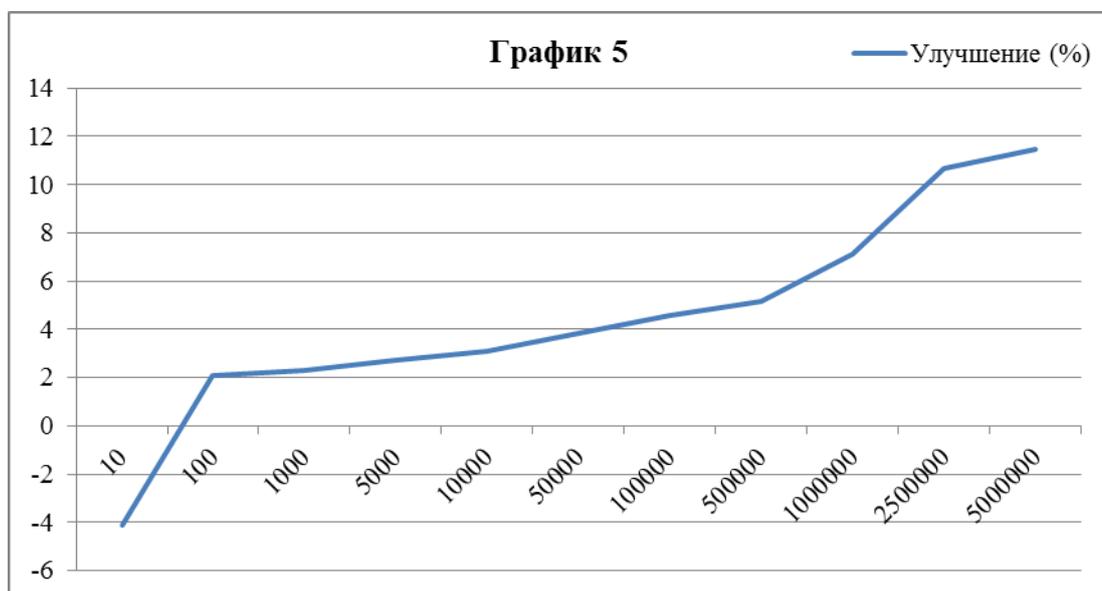


График 4 начинается не со значения 10 (элементов) по оси абсцисс, т.к. на таких малых задачах выигрыш отрицательный, соответственно, его не изобразить на логарифмической шкале.

5. Заключение

На основании проделанной работы можно сказать, что перенумерация как таковая имеет смысл лишь при размере задач более 1 миллиона

элементов. На графике 5 ниже приведено ускорение от перенумерации (с учётом времени на перенумерацию).



Выход из отрицательной зоны происходит уже на уровне 100 элементов, что значит, что перенумерация приносит пользу уже и на таких задачах (пусть незначительную).

Работа выполнена при финансовой поддержке Минобрнауки России по государственному контракту от 15.06.2012 г. №. 07.524.11.4019 в рамках ФЦП "Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007-2013 годы".

Список литературы:

1. Ю.В.Василевский, А.А.Данилов, К.Н.Липников, В.Н.Чугунов. Автоматизированные технологии построения неструктурированных расчетных сеток (частное сообщение), 290 с.
2. Ciprian Zavoianu Tutorial: Bandwidth reduction - The CutHill-McKee Algorithm [Электронный ресурс] // Ciprian Zavoianu – weblog, January 15, 2009 <http://ciprian-zavoianu.blogspot.ru/2009/01/project-bandwidth-reduction.html>

3. Джордж А., Лю Дж. Численное решение больших разреженных систем уравнений. — Москва.: Мир, 1984. — 333 с.
4. George Karypis METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices Version 5.1., August 4, 2011, Department of Computer Science & Engineering University of Minnesota Minneapolis, MN 55455, 34 p.
5. G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM Journal on Scientific Computing (pp. 359 — 392), 1999, 34 p.
6. Польской П.П., Маилян Д.Р. Композитные материалы – как основа эффективности в строительстве и реконструкции зданий и сооружений [Электронный ресурс] // «Инженерный вестник Дона», 2012, №4. – Режим доступа: <http://ivdon.ru/magazine/archive/n4p2y2009/1307> (доступ свободный) – Загл. с экрана. – Яз. рус.
7. Н.Е. Стёпин Реализация и сравнение различных вариантов алгоритма Узавы в задачах упругости для несжимаемых материалов [Электронный ресурс] // «Инженерный вестник Дона», Номер 2, 2013 г. – Режим доступа: <http://www.ivdon.ru/magazine/archive/n2y2013/1626> (доступ свободный) – Загл. с экрана. – Яз. рус.
8. Левин В.А., Левитас В.И., Лохин В.В., Зингерман К.М., Саяхова Л.Ф., Фрейман Е.И. Твердотельные фазовые переходы, вызванные действием механических напряжений в материале с наноразмерными неоднородностями: модель и вычислительный эксперимент // Доклады РАН. 2010. Т. 434, № 4. с. 481-485
9. V. A. Levin, K. M. Zingerman. A class of methods and algorithms for the analysis of successive origination of holes in a pre-stressed viscoelastic body. Finite strains// Communications in Numerical Methods in Engineering. 2008. V. 24. Issue 12. P. 2240-2251

10.Зингерман К.М., Левин В.А. Обобщение задачи Ламе-Гадолина для больших деформаций и ее аналитическое решение// Прикладная математика и механика. 2013. Т. 77, вып. 2, с. 322-336