

Конференция ПАВТ'11 28 марта – 1 апреля 2011

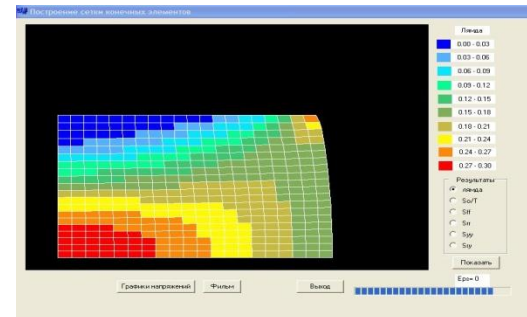
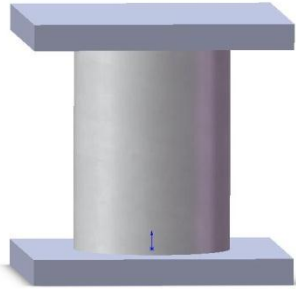
Использование усечённого варианта алгоритма SPIKE из библиотеки Intel Adaptive SPIKE-Based Solver для решения упругопластической задачи

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

Институт Машиноведения УрО РАН

Докладчик: Толмачев А.В.

АЛГОРИТМ РЕАЛИЗОВАН НА ПРИМЕРЕ УПРУГОПЛАСТИЧЕСКОЙ ЗАДАЧИ СЖАТИЯ ЦИЛИНДРА ПЛОСКИМИ ПЛИТАМИ



Решение задачи основывается на принципе виртуальной мощности в скоростной форме:

$$\int_V (\sigma + \Delta t \dot{\sigma}) \cdot \nabla h dV + \int_{\Sigma} (\sigma_{\tau} + \Delta t \dot{\sigma}_{\tau}) \cdot h_{\tau} d\Sigma = 0$$

На контакте с плитами приняты закон трения Кулона. Нагрузку в виде перемещения плиты прикладывали малыми шагами.

Решение задачи на шаге нагрузки методом КЭ состоит из трех основных этапов:

	Доля времени этапа
1. подготовка матрицы жесткости A	15%
2. решение СЛАУ $AX = F$	45%
3. вычисление напряженно-деформированного состояния	40%

Требуется 1000 шагов по нагрузке. На каждом шаге 10 -15 итераций этапов 2 и 3 для выполнения условия текучести Мизеса.

Intel Adaptive SPIKE-Based Solver

Для решения системы $AX=F$ использовали библиотеку Intel Adaptive SPIKE-Based Solver

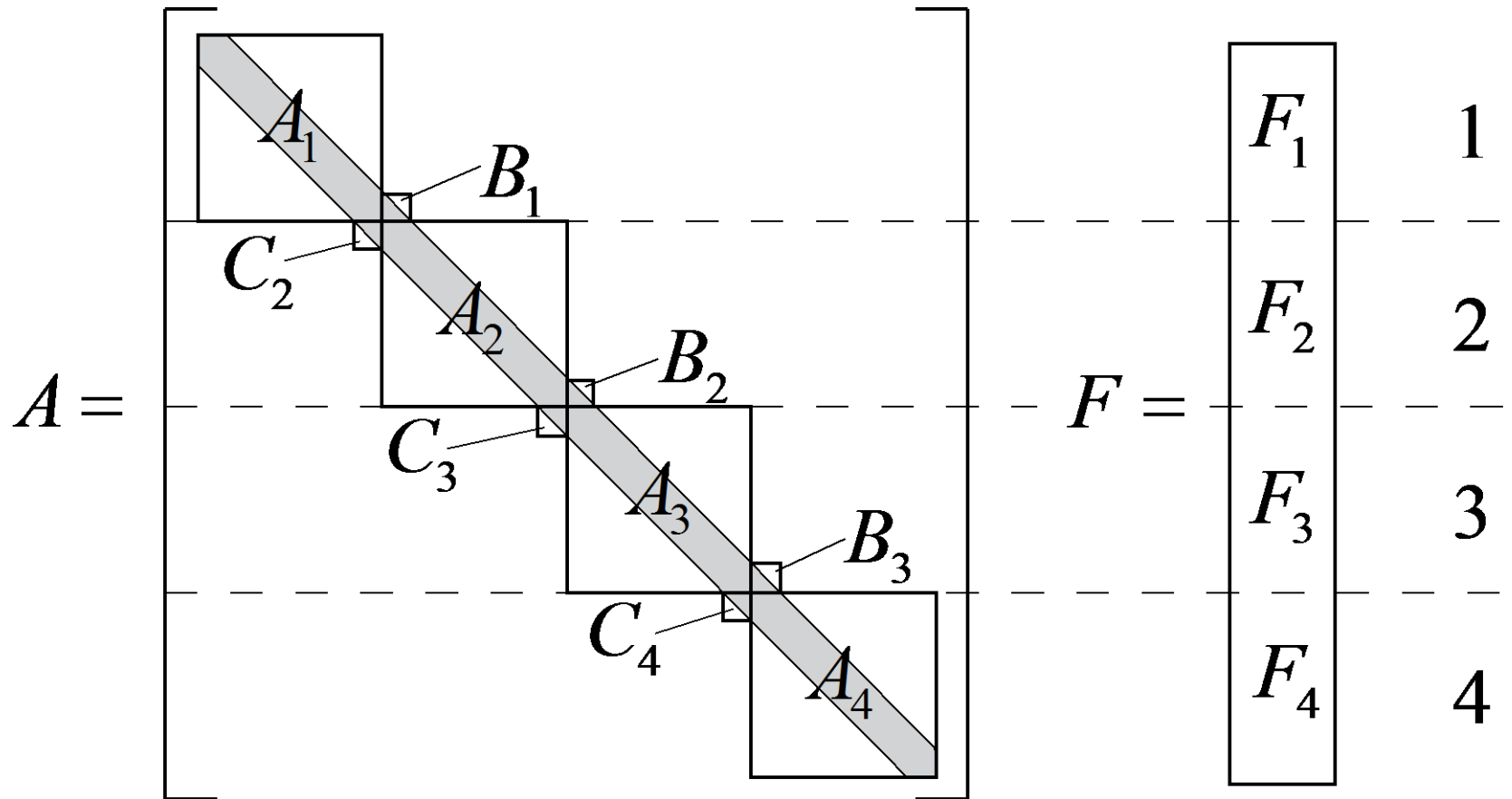
<http://software.intel.com/en-us/articles/intel-adaptive-spike-based-solver/>

Алгоритм SPIKE [1, 2]

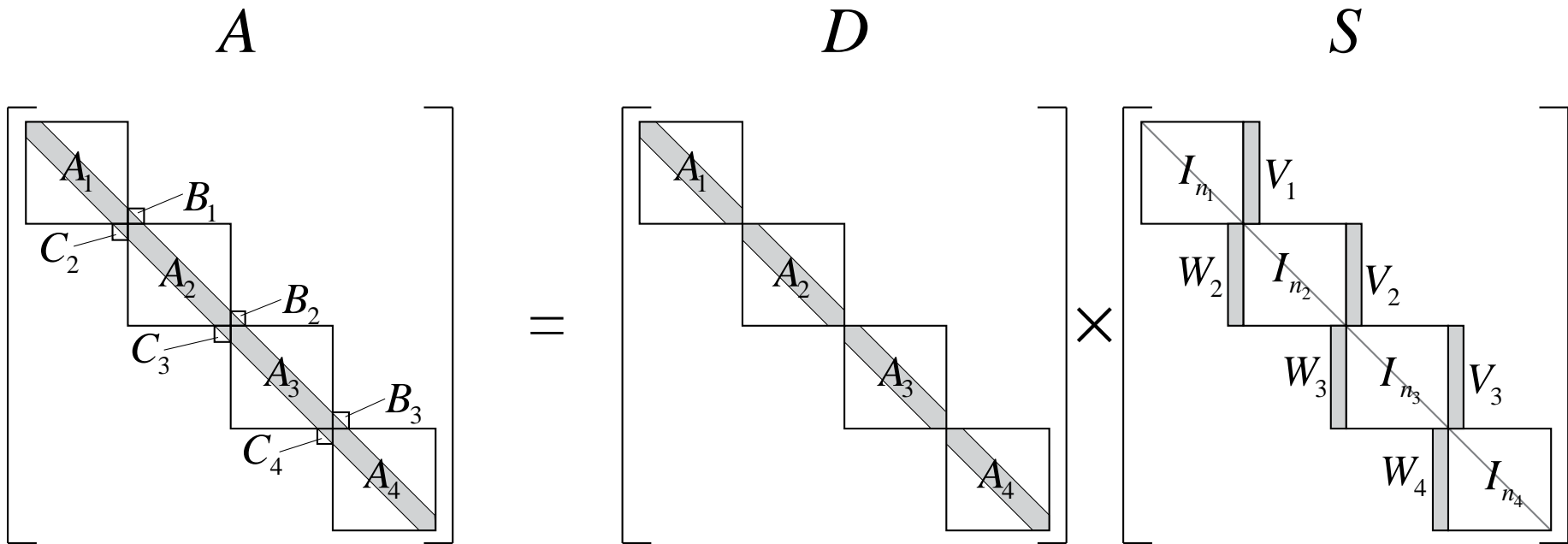
Этапы алгоритма SPIKE:

- Разбиение матрицы A на фрагменты
 - Разложение матрицы $A = DS$
 - Формирование сокращённой системы
 - Решение сокращённой системы
1. Polizzi E, Sameh A. SPIKE: A parallel environment for solving banded linear systems. *Computers & Fluids*. 2007 Jan 0;36:113--120.
 2. Polizzi E, Sameh AH. A parallel hybrid banded system solver: the SPIKE algorithm. *Parallel Comput*. 2006;32:177--194.

Разбиение матрицы A на фрагменты

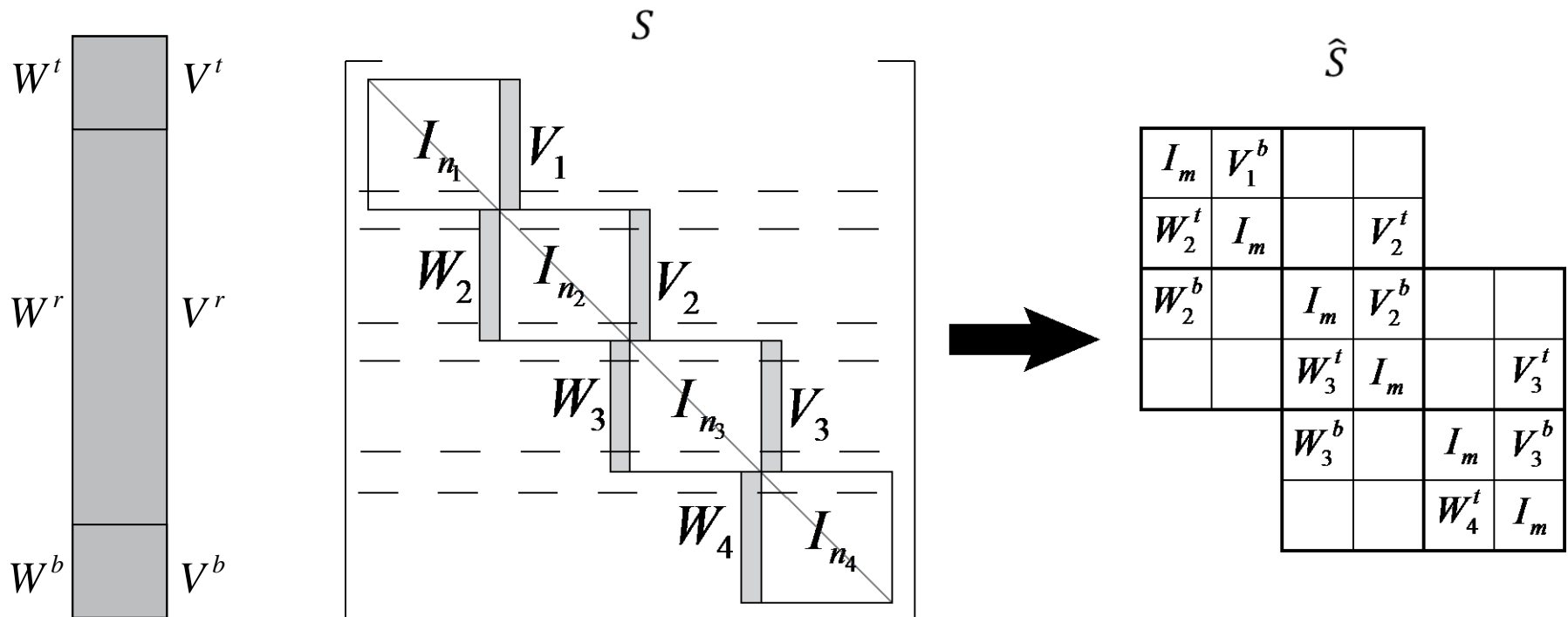


Разложение $A=DS$



W_i, V_i - узкие и длинные матрицы — шипы

Сокращённая система



$$\hat{S}\hat{X} = \hat{G}$$

Усечённый вариант алгоритма SPIKE

Особенности использования решателя в рассматриваемой задаче

- Матрица не имеет диагонального доминирования (степень 0.4), получаемое отклонение от метода простого LU-разложения порядка 10^{-12}
- Матрица формируется с порядком индексации $a[i, j]$, решатель требует $a[j, i]$, поэтому мы транспонируем матрицу

Вычислительный эксперимент

- Кластер um64 ИММ УрО РАН
- 32 двухпроцессорных двухядерных модуля с процессорами AMD Opteron (2.6 ГГц)
- 1024 ГБ оперативной памяти на процессор
- Intel Adaptive SPIKE-Based Solver от 23 февраля 2010 года
- Intel MKL версии 10.0.010
- OpenMPI версии 1.3.3 на сети InfiniBand

Параметры сеток

d	β	n	β/n
100	205	20402	0,0100
150	305	45602	0,0067
200	405	80802	0,0050
300	605	181202	0,0033

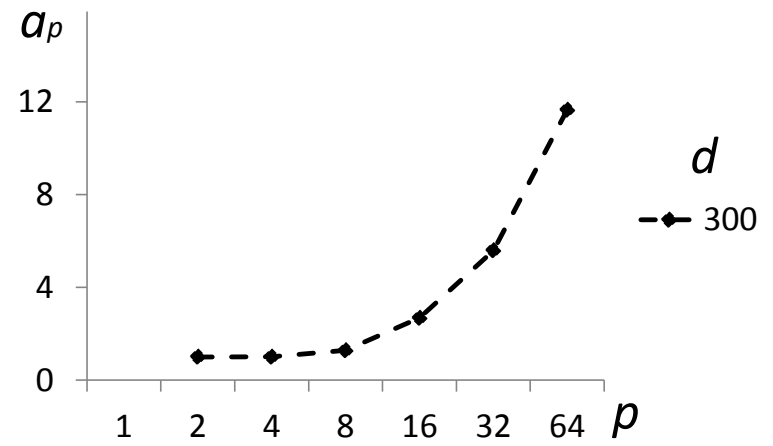
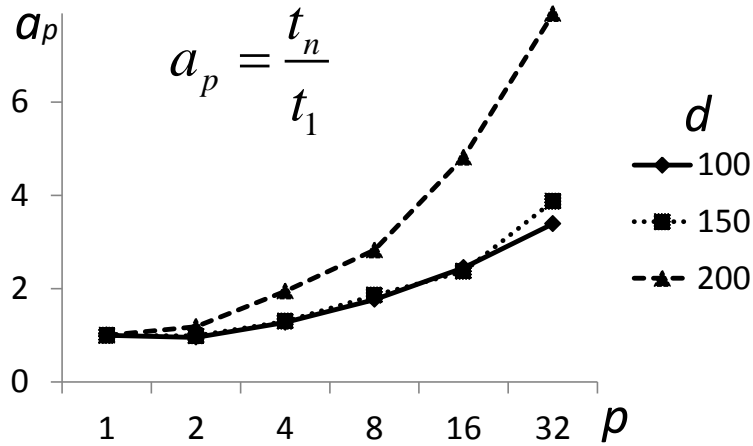
d – количество разбиений сетки

β – полуширина матрицы жёсткости

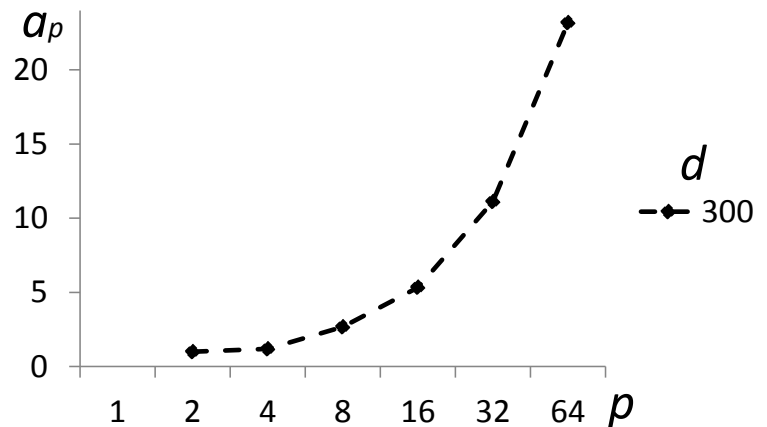
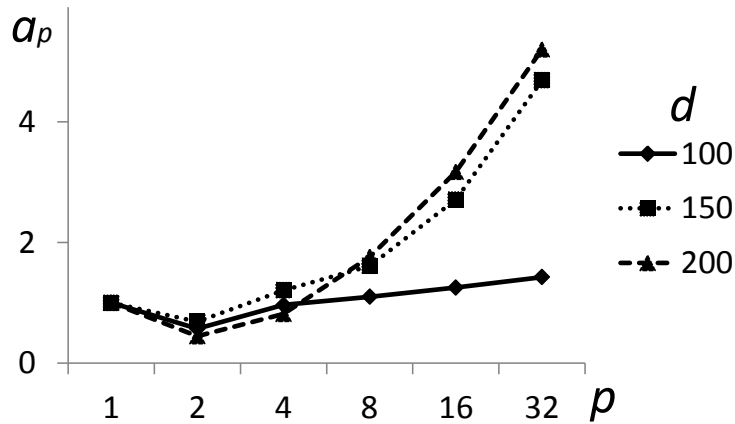
n – общее количество неизвестных

Результаты вычислительного эксперимента

Разложение $A=DS$

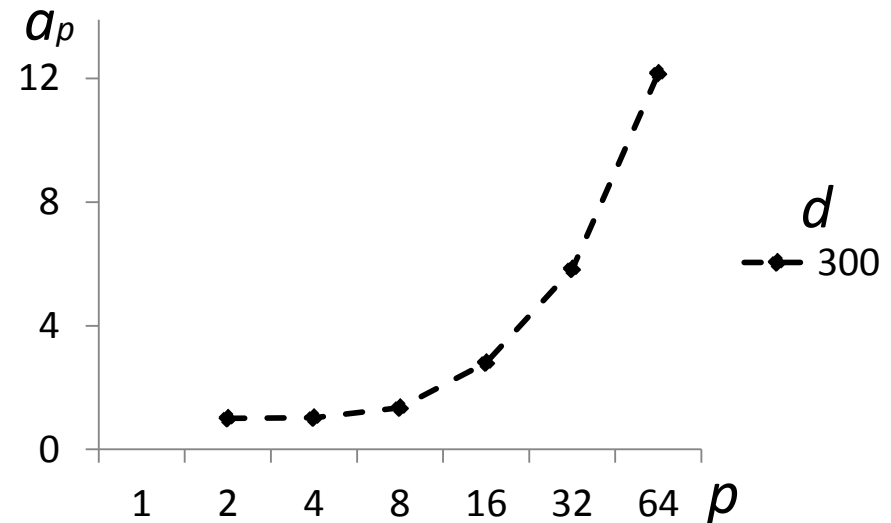
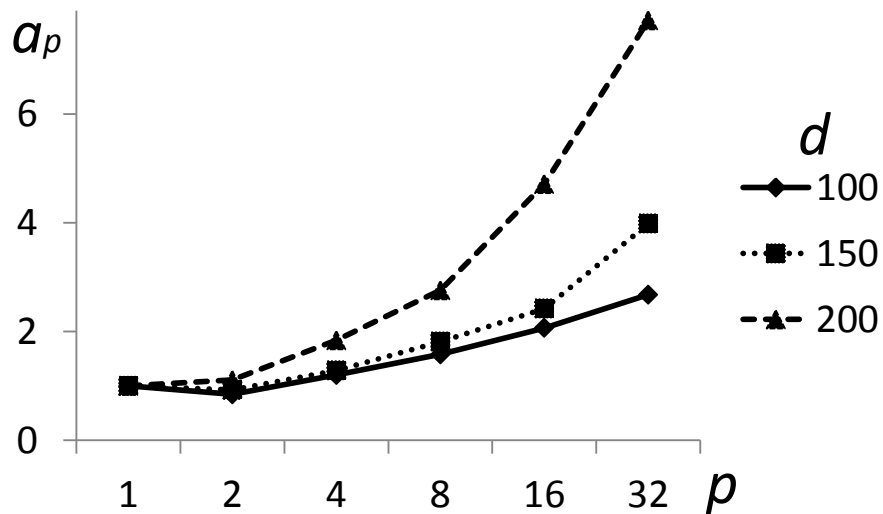


Решение $DSX=F$



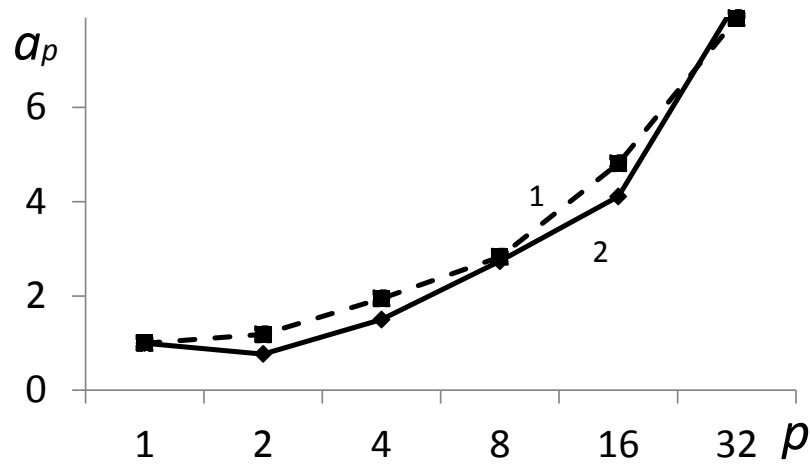
Результаты вычислительного эксперимента

Полное решение (1 разложение $A=DS$ и 15 решений $DSX=F$)

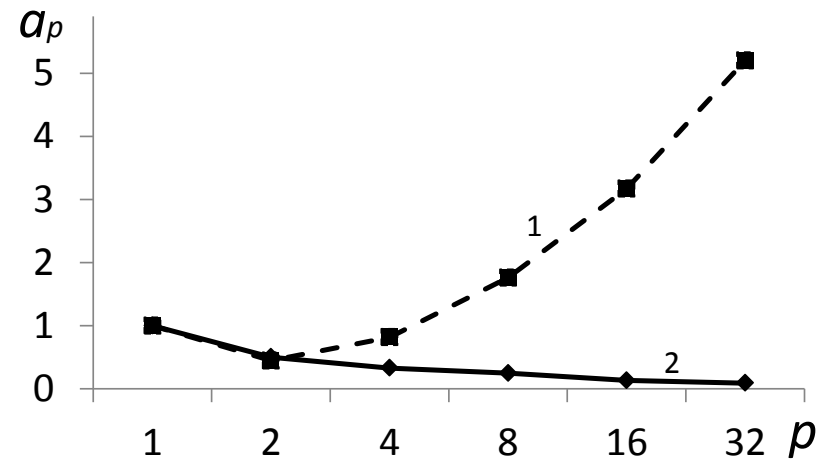


- Большое ускорение вычисления разложения $A=DS$
- Ускорение решения >1 на всех рассмотренных сетках
- Достигается значительное общее ускорение при решении СЛАУ

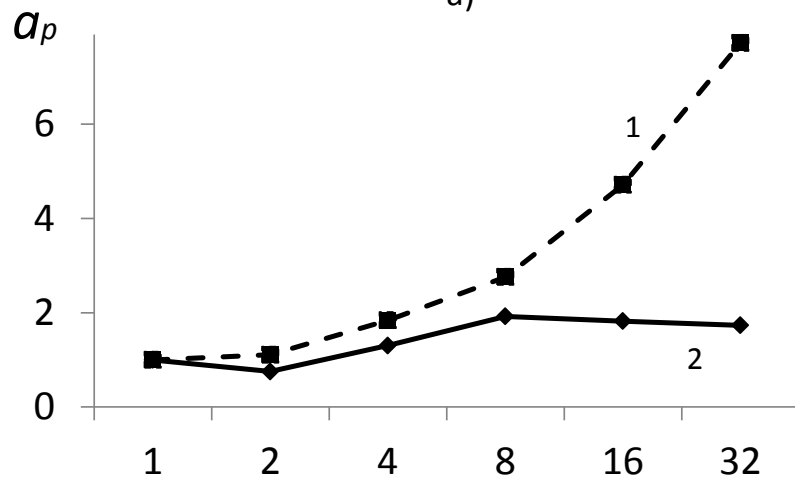
Сравнение производительности с решателем из ScaLAPACK ($d=200$)



а)



б)



в)

а) этап разложения

б) этап решения

в) полное решение $AX=B$ с 15 итерациями

1 – SPIKE

2 – ScaLAPACK

Спасибо за внимание!