# AlgoWiki: изучение разных вариантов одного численного метода и другие проблемы

А.В. Фролов[1], А.С. Антонов[2]

Федеральное государственное бюджетное учреждение науки
Институт вычислительной математики им. Г.И. Марчука Российской академии наук[1],

Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный университет имени М.В.Ломоносова»[2]

# Проблемы, возникающие при измерениях производительности

## Предыстория: схема «*чёрного ящика*».

Предполагалось, что для измерения производительности **не нужно непосредственно считать количество операций**, достаточно знать затраченное время и теоретическую оценку количества операций

# Проблемы, возникающие при измерениях производительности

2016г.: «зашкаливание» в QR-разложении методом Хаусхолдера производительности на разовом тесте.

Оказалось, что тестовая матрица слишком простой структуры.

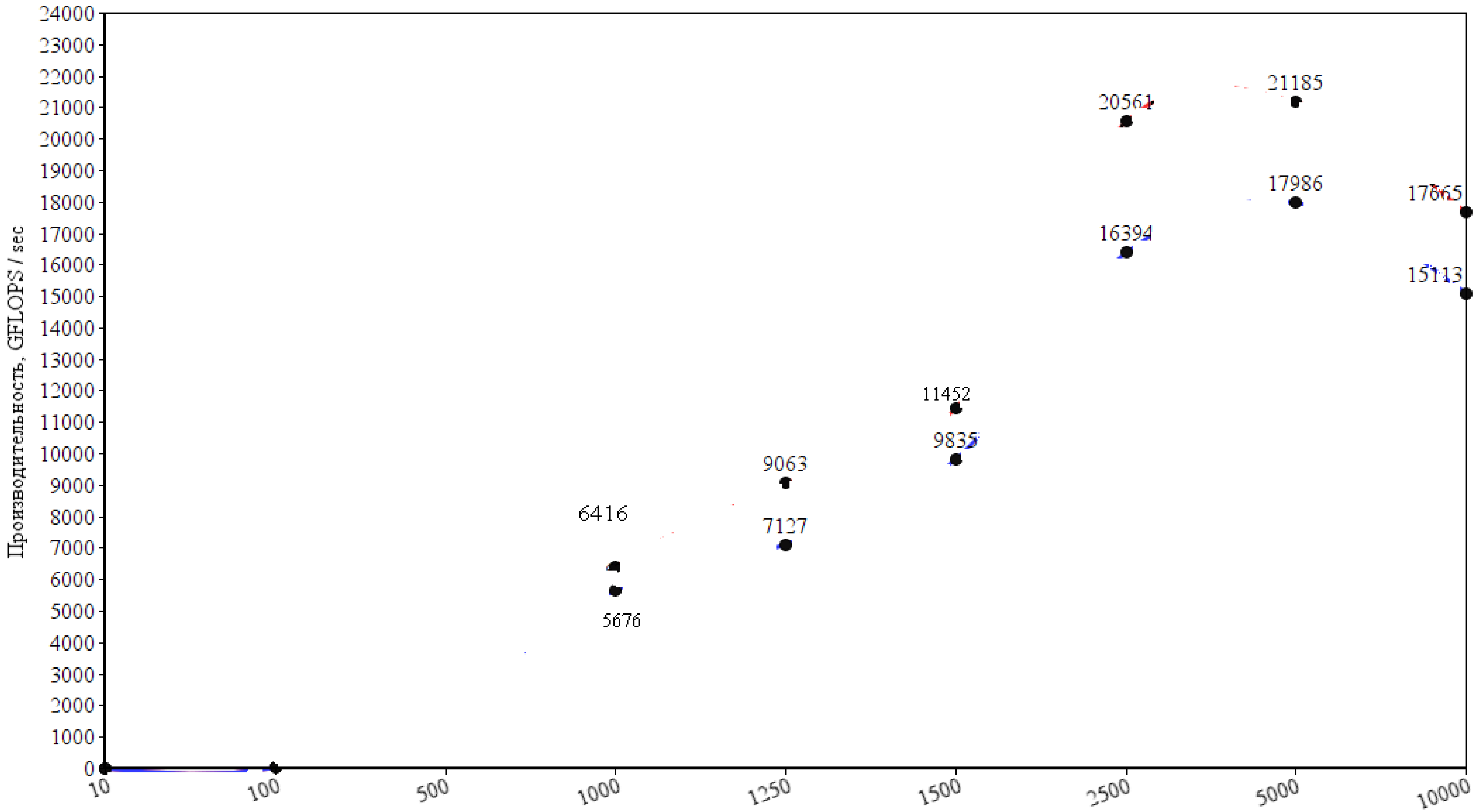
# Проблемы, возникающие при измерениях производительности

2018-2019гг.: тестирование по случайным матрицам. Хаусхолдер.

| Matrix | Cores | Time (sec) | SpeedUp | Perfomance (GFLOPS) |
|---|---|---|---|---|
| 4096 | 1 | 6.865000 | 1.000000 | 6.673414 |
| 4096 | 2 | 3.576000 | 1.919743 | 12.811237 |
| 4096 | 4 | 1.971000 | 3.483004 | 23.243523 |
| 4096 | 8 | 1.059000 | 6.482531 | 43.260609 |
| 4096 | 16 | 0.834000 | 8.231415 | 54.931636 |
| 4096 | 32 | 0.627000 | 10.948963 | 73.066961 |
| 4096 | 64 | 0.750000 | 9.153333 | 61.083979 |
| 8192 | 1 | 113.308000 | 1.000000 | 3.234581 |
| 8192 | 2 | 70.605000 | 1.324816 | 5.190905 |
| 8192 | 4 | 47.573000 | 2.381771 | 7.704031 |
| 8192 | 8 | 31.179000 | 3.623653 | 11.810998 |
| 8192 | 16 | 15.005000 | 7.551350 | 24.425450 |
| 8192 | 32 | 4.302000 | 26.338447 | 85.193834 |
| 8192 | 64 | 2.098000 | 54.007626 | 174.692029 |
| 16384 | 1 | 1445.902000 | 1.000000 | 2.027821 |
| 16384 | 2 | 662.908000 | 2.181150 | 4.422983 |
| 16384 | 4 | 453.624000 | 3.187446 | 6.463571 |
| 16384 | 8 | 303.797000 | 4.912895 | 9.4056909 |
| 16384 | 16 | 147.894000 | 9.777007 | 19.826024 |
| 16384 | 32 | 60.754000 | 23.799289 | 48.260707 |
| 16384 | 64 | 26.527000 | 54.506804 | 110.530064 |
| 32768 | 1 | 18757.072000 | 1.000000 | 1.250528 |
| 32768 | 2 | 9378.536000 | 2.000000 | 2.501056 |
| 32768 | 4 | 4689.268000 | 4.000000 | 5.002113 |
| 32768 | 8 | 2344.634000 | 8.000000 | 10.004226 |
| 32768 | 16 | 1172.317000 | 16.000000 | 20.010452 |
| 32768 | 32 | 642.553000 | 30.221478 | 36.504768 |
| 32768 | 64 | 275.257000 | 67.143851 | 85.215809 |

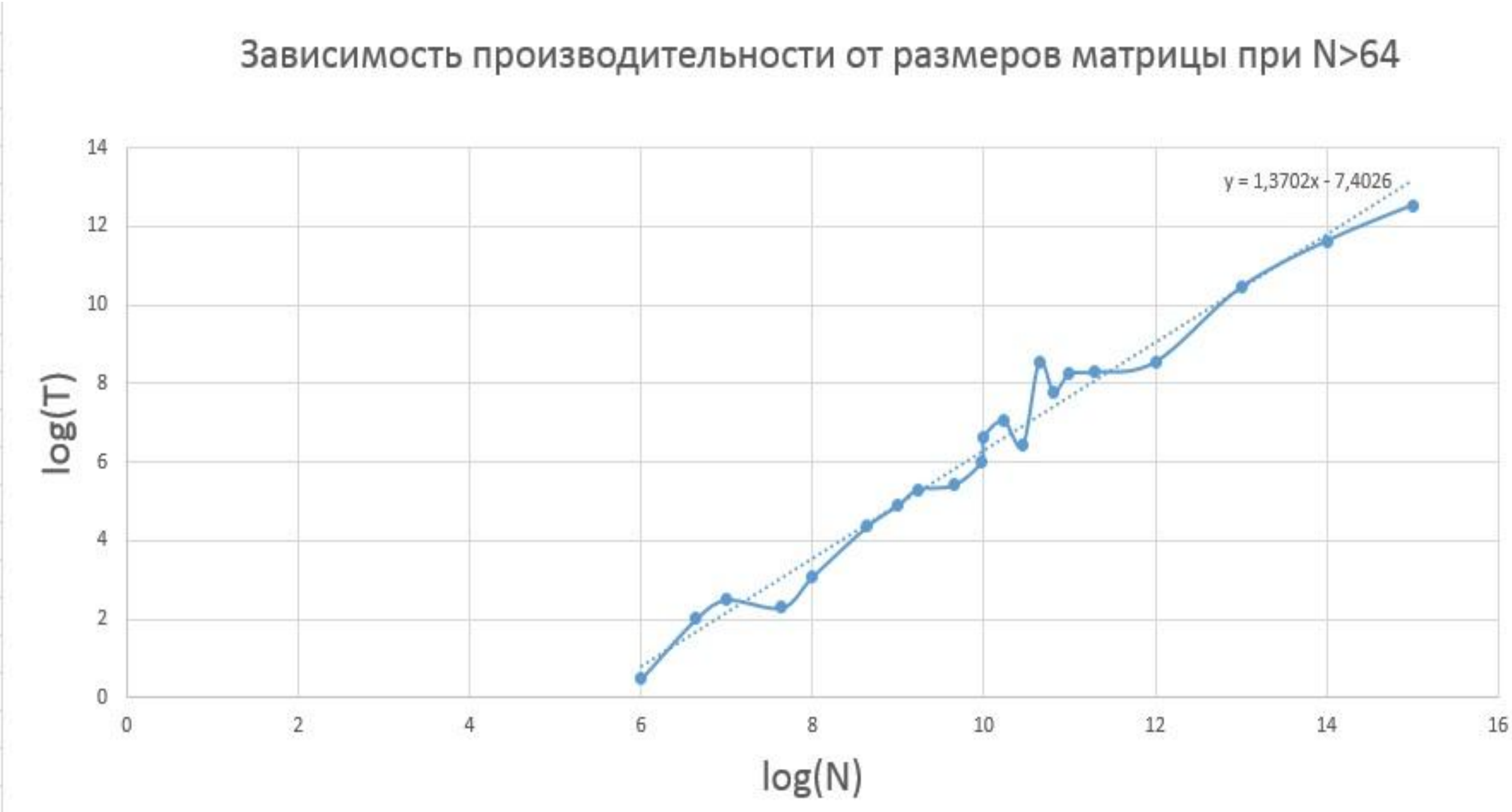
# Проблемы, возникающие при измерениях производительности

2018-2019гг.: тестирование по случайным матрицам. Спектральная задача на графических ускорителях. По абсциссе размеры задачи.



# Проблемы, возникающие при измерениях производительности

2018-2019гг.: тестирование по случайным матрицам. Спектральная задача, кластерная архитектура.



# Проблемы, возникающие при измерениях производительности

С чем связаны колебания в спектральном случае?

С тем, что на данных размерах сходимость отличается (коэффициентом при n) от среднестатистической сходимости ?

Или это реальные показатели «качества распараллеливания» на данной вычислительной системе?

# Проблемы, возникающие при измерениях производительности

Вывод: подход к программе как к «чёрному ящику» при замерах в данном случае не годится. Нужно брать не оценку количества итераций/операций, а вычислять её, забираясь в саму программу.

# Сравнение вариантов метода Якоби для спектральных задач

В сравнении с **остальными** методами:

- Важнейший недостаток – медленная сходимость
- Важнейшее преимущество – низкая относительная погрешность для малых собственных значений

Отсутствует в распространённых пакетах программ. Почему?

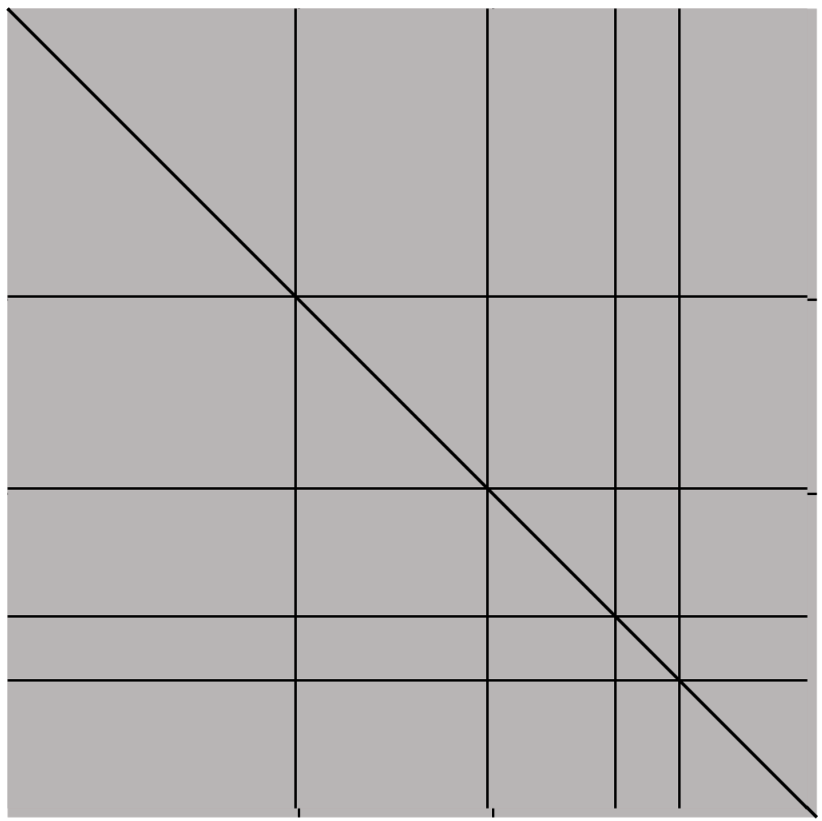
# Исследование и сопоставление разных вариантов метода Якоби

## Варианты решаемой задачи:

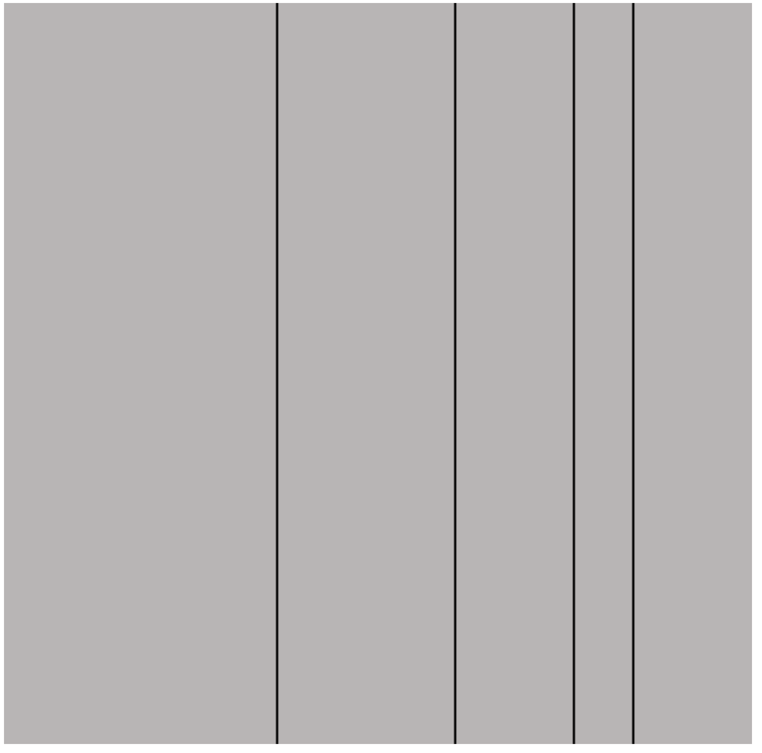
- Собственные значения для симметричных матриц
- Сингулярные значения для всех матриц

В учебниках (например, Дж.Деммеля): «легко распараллеливается» в варианте циклических вращений с барьерами

# Собственные значения – двусторонние вращения



# Сингулярные значения – односторонние вращения



В учебниках: «не требуется явного вычисления матрицы G*G».

В реальности: требуется явное вычисление элементов матрицы G*G.

# Метод Якоби с циклическим исключением и барьерами

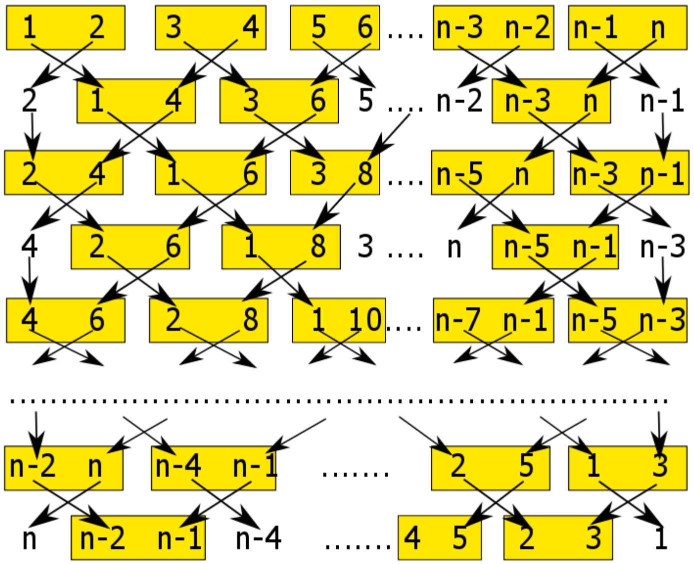
## Варианты решаемой задачи:

- Собственные значения для симметричных матриц
- Сингулярные значения для всех матриц

В реальности: между шагами итерации требуется передать сопоставимый с количеством арифметических операций объём данных

# Метод Якоби с циклическим исключением и барьерами Выбор последовательности пар – «задача о расписании турнира»

При постоянных обменах данными важными становятся постоянные **маршруты их передачи**



# Метод Якоби для вычисления сингулярных чисел

В исходном варианте: на каждом макрошаге вычисляется O(n) элементов матрицы G*G, каждый по O(n) операций (не менее логарифмического количества шагов на макрошаг)

Можно: отбросив «преимущество», в начале вычислять все $(n^2+n)/2$ элементов матрицы G*G, каждый по O(n) операций (не менее логарифмического количества шагов), а затем на каждом макрошаге они перевычисляются (каждый раз по $n^2/2$ элементов, поскольку n/2 элементов «обнуляются» принудительно).

# Метод Якоби для вычисления сингулярных чисел

Таким образом, для параллельных реализаций «преимущество» метода Якоби, состоящее в неявности его исполнения, - перестаёт быть преимуществом, ибо **замедляет возможное быстродействие**.

Де-факто метод Якоби для сингулярных чисел **более трудоёмок**, чем для собственных чисел симметричных матриц

**Вывод: неявность выполнения двусторонних вращений не нужна**.

# Спасибо за внимание!