

Использование технологий параллельного  
программирования для моделирования  
сейсмических волн с учетом топографии  
сеточно-характеристическим методом

Иванов А.М.

Хохлов Н.И.

Московский физико-технический институт

Суперкомпьютерные дни в России  
Москва, 25-26 сентября 2017 г.

# План

- Математическая модель
- Численный метод
- Численный эксперимент
- Программная реализация
- Последовательная реализация
- Распараллеливание
  - OpenMP и POSIX Threads
  - MPI
  - CUDA и OpenCL
- Заключение

# Математическая модель

Идеальный изотропный линейно-упругий материал

$$\begin{cases} \rho \dot{\vec{v}} = \nabla T & \text{Уравнение движения} \\ \dot{T} = \lambda(\nabla \cdot \vec{v})I + \mu(\nabla \otimes \vec{v} + \vec{v} \otimes \nabla) & \text{Закон Гука} \end{cases}$$

$\rho$  – плотность среды

$\lambda, \mu$  – параметры Ламе

$\vec{v}$  – скорость частиц среды

$\sigma^{ij}$  – компоненты симметричного тензора напряжений  $T$

$I$  – единичный тензор

# Численный метод

Метод расщепления:

$$\frac{\partial v}{\partial t} + A \frac{\partial v}{\partial x} + B \frac{\partial v}{\partial y} + C \frac{\partial v}{\partial z} = 0 \quad \Longrightarrow \quad \begin{aligned} \frac{\partial v}{\partial t} + A \frac{\partial v}{\partial x} &= 0 \\ \frac{\partial v}{\partial t} + B \frac{\partial v}{\partial y} &= 0 \\ \frac{\partial v}{\partial t} + C \frac{\partial v}{\partial z} &= 0 \end{aligned} \quad \Longrightarrow \quad \begin{aligned} v' &= f(A)v^n \\ v'' &= f(B)v' \\ v^{n+1} &= f(C)v'' \end{aligned}$$

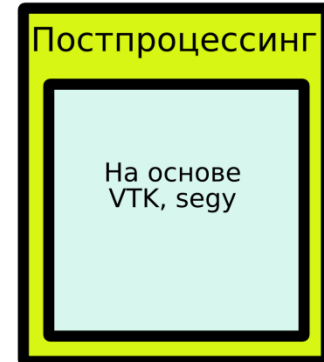
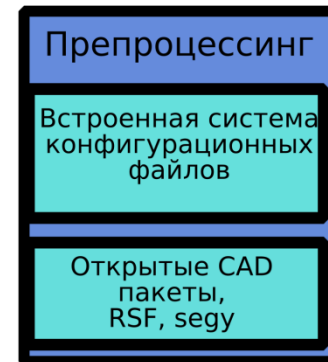
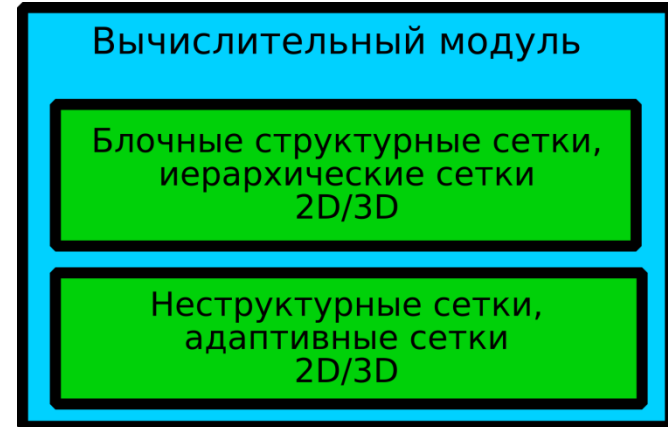
Замена переменных:

$$\frac{\partial v}{\partial t} + A \frac{\partial v}{\partial x} = 0 \quad \Longrightarrow \quad \begin{aligned} A &= \Omega_A^L \Lambda \Omega_A^R \\ \Omega_A^R \Omega_A^L &= I \\ \Lambda &= \text{diag}(s_1, s_2, s_3, s_4, s_5) \end{aligned} \quad \Longrightarrow \quad \begin{aligned} \omega &= \Omega_A^R v \\ \frac{\partial \omega}{\partial t} + \Lambda \frac{\partial \omega}{\partial x} &= 0 \end{aligned}$$

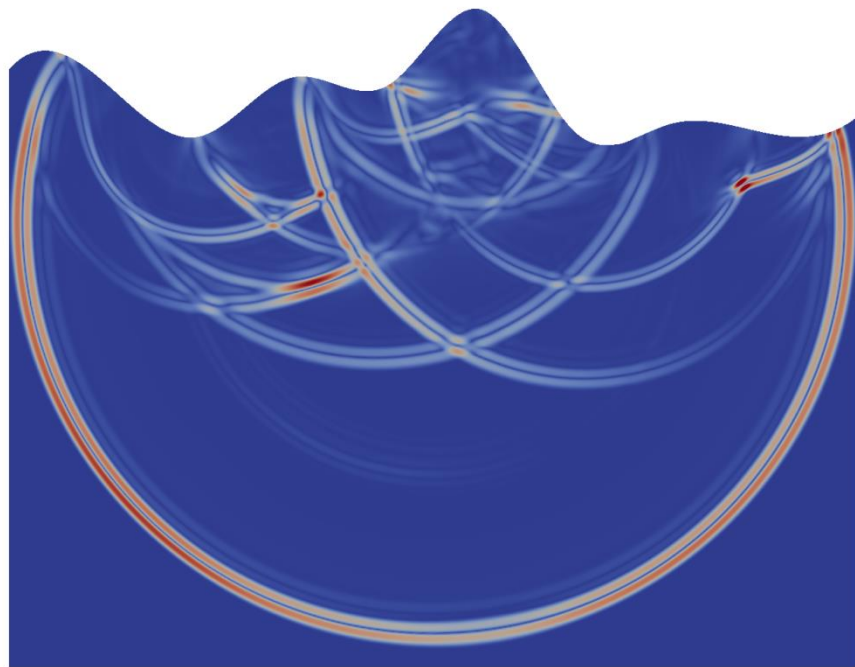
Полученные одномерные уравнения переноса решаются методом характеристик или с использованием конечных разностей

# Программный комплекс

- Возможность расчета в 2D, 3D
- Структурные сетки
  - Сеточно-характеристические, конечно-объемные, FDTD методы 1-4 порядка точности
  - Блочные структурные сетки, криволинейные сетки
  - Большие размеры сеток (более 1 млрд. узлов)
  - Распараллелен на MPI/OpenMP/CUDA/OpenCL
- Неструктурные сетки
- Этапы расчета:
  - Препроцессинг (собственный формат конфигурационных файлов)
  - Основной цикл интегрирования по времени
  - Постпроцессинг (визуализация расчетной области – ParaView, VisIt, Mayavi, сейсмограмм – segy)
- Расчет в неоднородных средах
- Распараллеливание – геометрический параллелизм
- Упругие/акустические среды, явное выделение контактов и неоднородностей



# Численный эксперимент в 2D



Волновая картина в момент  
 $t = 750$  мс

800 × 500 узлов

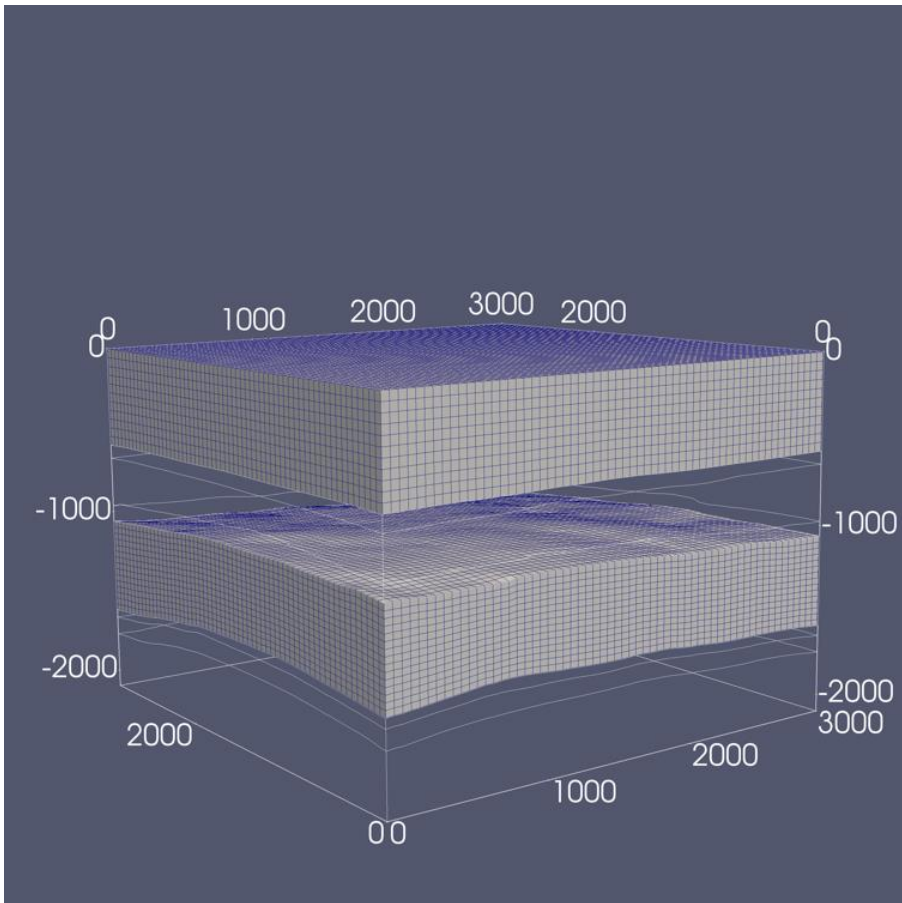
Начальное возмущение на глубине  
50 м от поверхности

$$C_p = 2500 \text{ м/с}$$

$$C_s = 1500 \text{ м/с}$$

$$\rho = 1500 \text{ г/см}^3$$

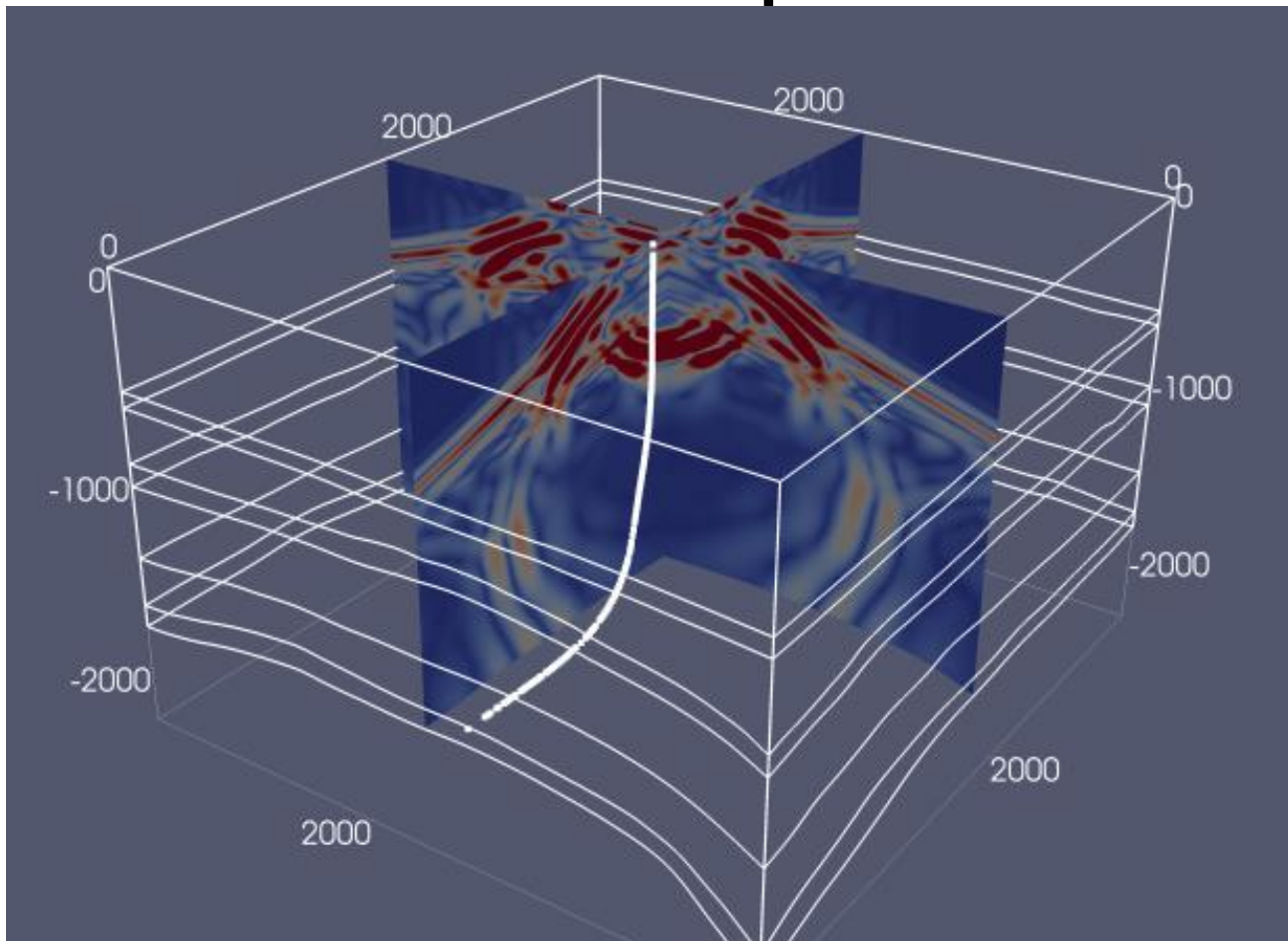
# Численный эксперимент в 3D



**Участок на берегу Баренцева моря**

Номер слоя	$V_p$ , м/с	$V_s$ , м/с	$\rho$ , кг/м <sup>3</sup>
1	2170	674	2000
2	2130	795	2300
3	2500	1090	2200
4	2680	1220	2300
5	3000	1385	2400
6	5550	3144	2700
7	6000	1250	2800
8	6000	1550	2850

# Численный эксперимент в 3D

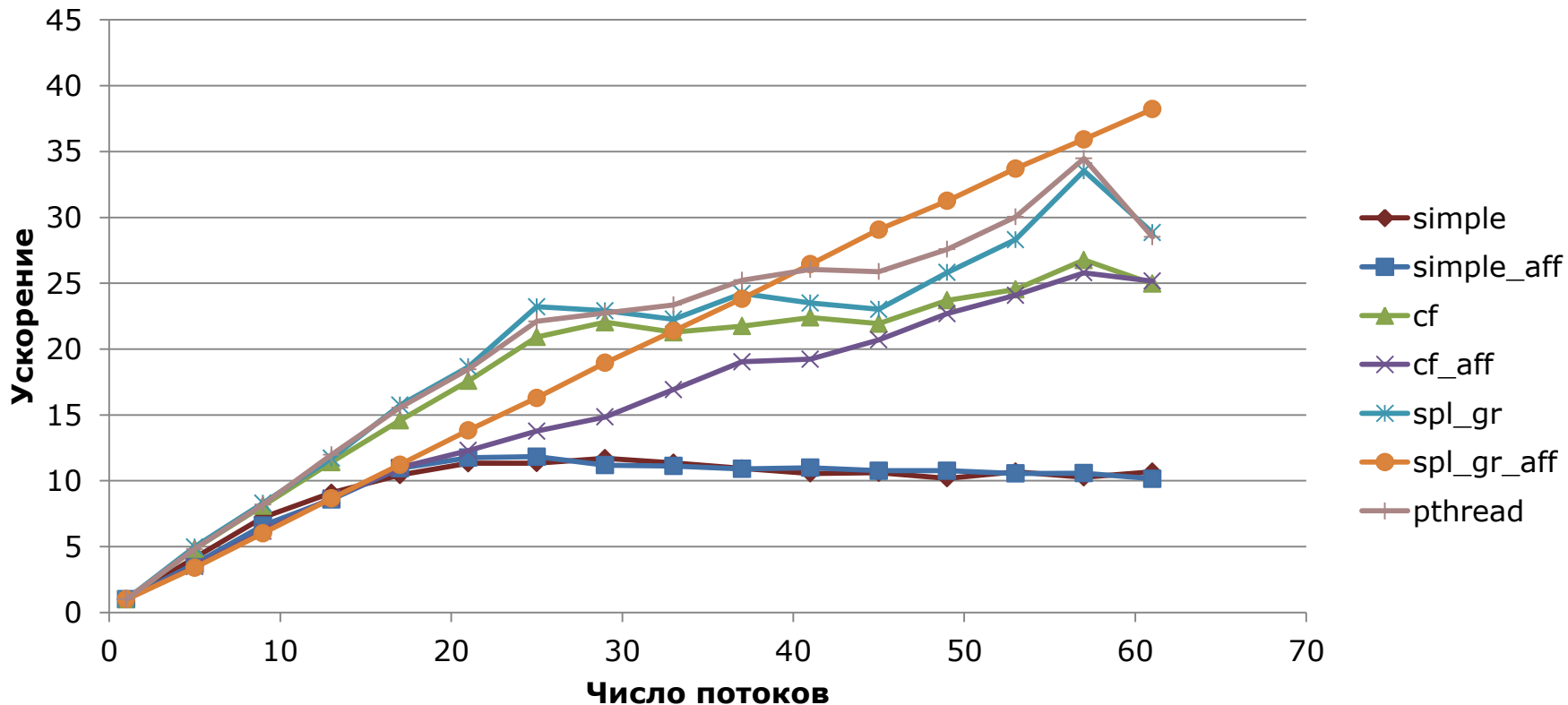




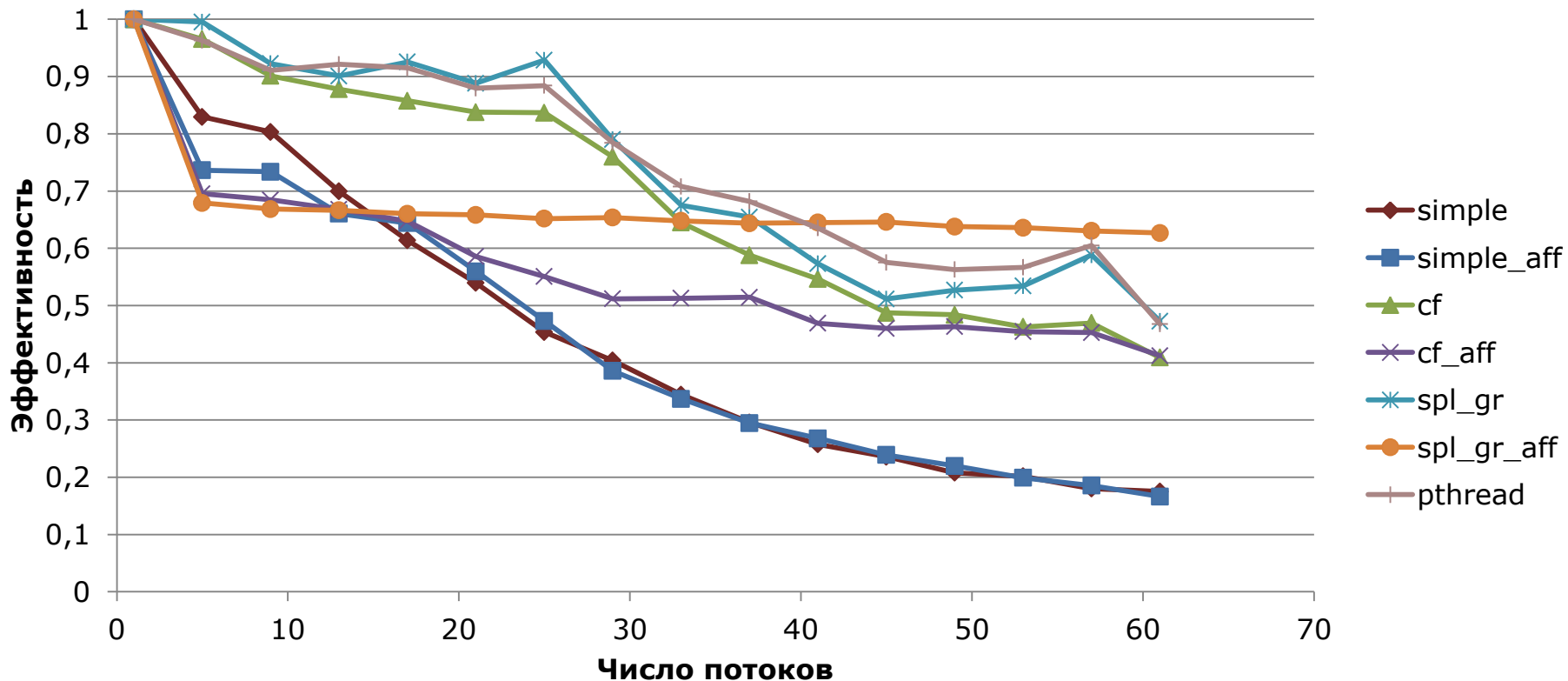
# OpenMP и POSIX Threads

- Условия тестирования:
  - 4000 × 4000 узлов
  - 100 шагов по времени
  - Время расчета ~7мин
- Машина: AMD Opteron 6272 (64 ядра)
- Оптимизации:
  - simple (#pragma omp parallel)
  - cf (последовательное обращение к памяти при обходе сетки)
  - spl\_gr (сетка разделена между потоками)
  - \*\_aff (привязка потоков к ядрам процессора)

# OpenMP и POSIX Threads



# OpenMP и POSIX Threads

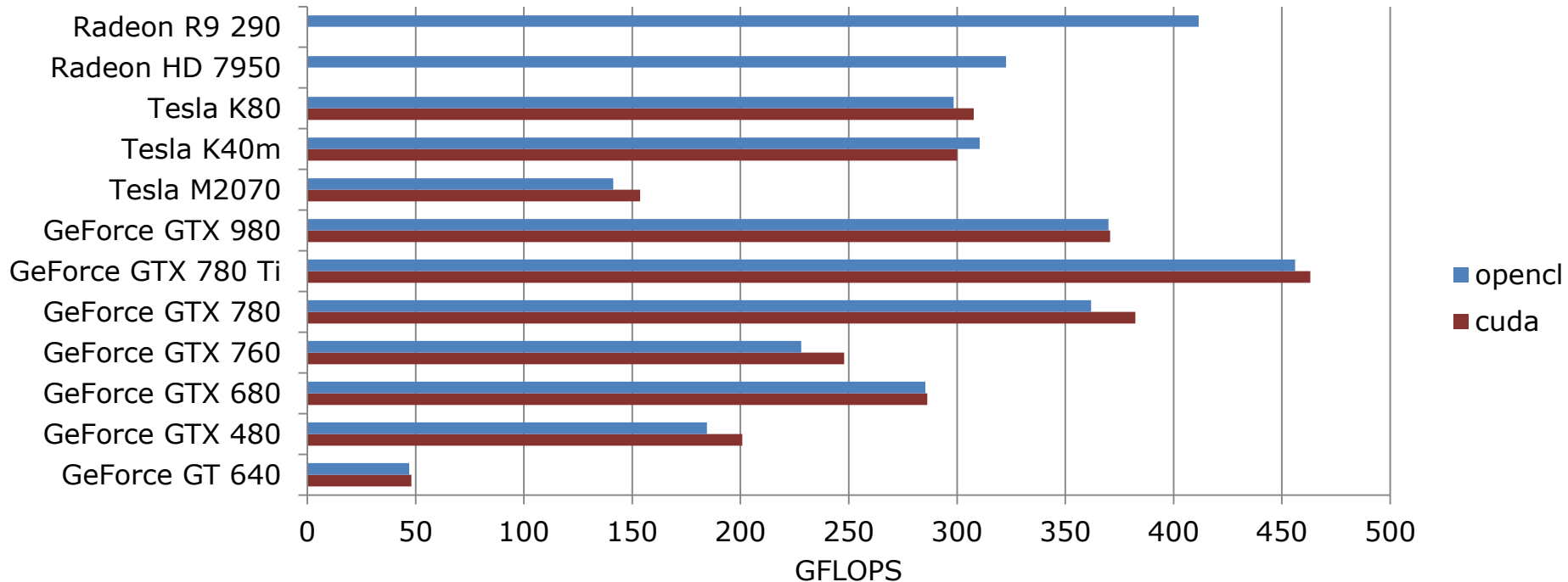


# CUDA и OpenCL

- Условия тестирования:
  - 4096 × 4096 узлов
  - 6500 шагов по времени
  - Время расчета ~10 мин
- Оптимизации:
  - массив структур → структура массивов
  - использование общей памяти графического процессора
  - последовательный (coalesced) доступ к памяти
  - подбор размеров блоков для минимизации дополнительных виртуальных узлов, требующихся для синхронизации
  - минимизация ветвлений в коде, выполняющемся на графическом процессоре
  - Обмен данными между минуя центральный процессор

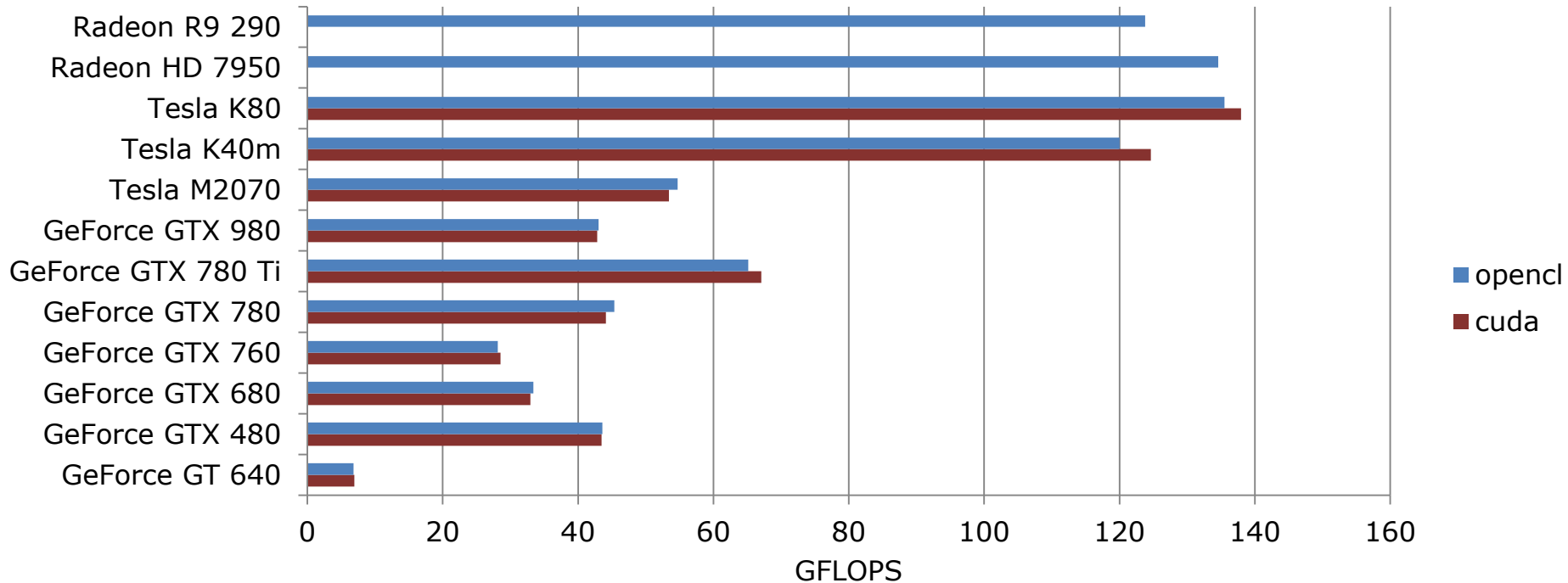
# CUDA и OpenCL

Performance - float + fast math

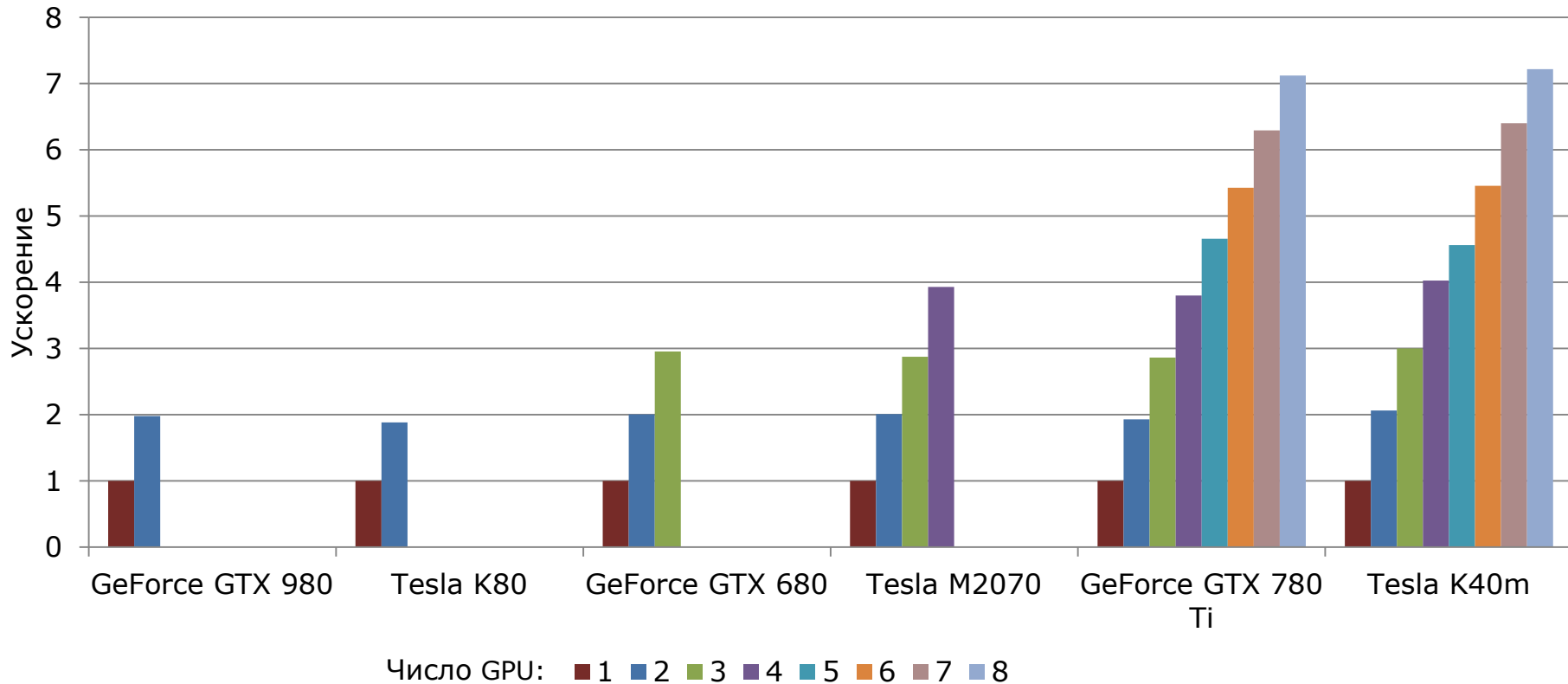


# CUDA и OpenCL

Performance - double



# CUDA и OpenCL

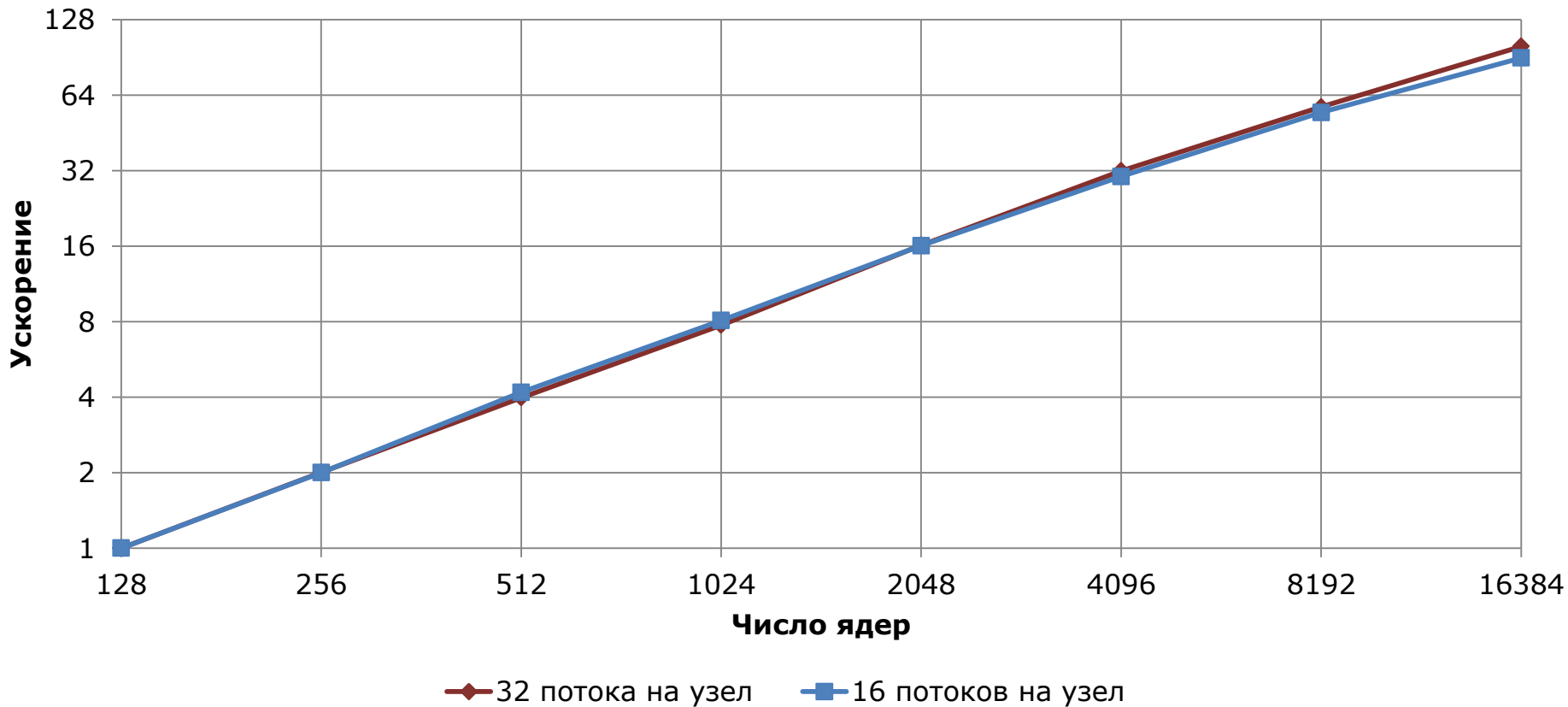


# MPI

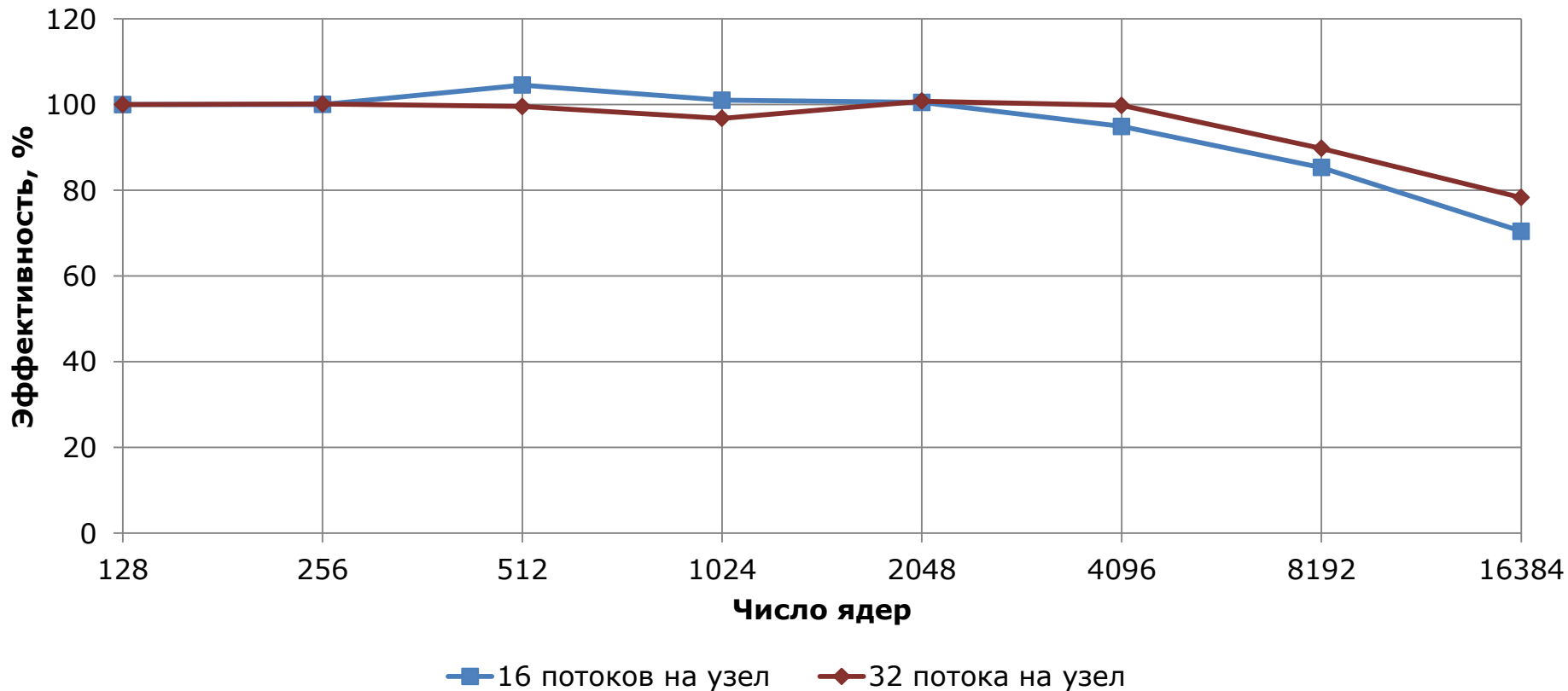
- Условия тестирования:
  - 1000 × 1000 × 1000 узлов
  - Кластер HECToR (2816 узлов по 2 процессора AMD Opteron 2.3GHz Interlagos (16 ядер))
  - Расчет на 128 ядрах ~4 часа.
- Обмен на основе функций MPI\_Isend, MPI\_Irecv



# MPI



# MPI



# Заключение

- Достигнута эффективность распараллеливания:
  - 60% с использованием технологии OpenMP на 64 ядрах
  - 70% с использованием MPI на 16384 ядрах
  - 85% с использованием OpenCL и CUDA на 8 графических процессорах
- Продемонстрирована возможность распараллеливания сеточно-характеристического метода на структурированных сетках, что позволяет проводить расчеты с более точным учетом топографии

Спасибо за внимание