



САПЕТИНА Анна Федоровна

**АРХИТЕКТУРНО-ОРИЕНТИРОВАННЫЙ ПОДХОД
К ПРОЕКТИРОВАНИЮ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ
НА ПРИМЕРЕ РЕШЕНИЯ
ТРЕХМЕРНОЙ ЗАДАЧИ ДИНАМИЧЕСКОЙ СЕЙСМИКИ**

Доклад по диссертации
на соискание ученой степени
кандидата физико-математических наук
по специальности 1.2.2 –
математическое моделирование,
численные методы и комплексы программ

Научный руководитель:
д.ф.-м.н. Г.В. Решетова

Актуальность темы исследования

Объект исследования – методы и средства архитектурно-ориентированной организации и оптимизации параллельных вычислений на предмет их развития и использования при решении задач численного моделирования физических процессов на современных вычислительных архитектурах.

- Решение современных задач математической физики требует масштабируемых высокопроизводительных вычислений для увеличения размерности и сокращения времени решения задачи.
- Проблема отображения (Г.И. Марчук): отображение задач вычислительной математики на архитектуру вычислительных систем.
- С развитием многопроцессорных, многоядерных и гибридных систем проблема отображения стала центральной в высокопроизводительных вычислениях.
- Архитектурно-ориентированные методы и средства организации параллельных вычислений обеспечивают согласование структуры вычислений, хранения данных и обменов с особенностями конкретных платформ.

Многоуровневый подход

[Глинский и др., 2015; Glinskiy et al., 2017]

- Учет особенностей целевой архитектуры на всех этапах решения задачи численного моделирования (со-дизайн);
- Оценка энергоэффективности и имитационное моделирование исполнения программы на большом числе ядер.

Актуальность темы исследования

- Решение трёхмерной задачи динамической сейсмологии для численного моделирования сейсмических полей требует разработки программных комплексов, ориентированных на эффективное использование современных высокопроизводительных вычислительных систем.
- Рост вычислительных мощностей способствовал широкому внедрению конечно-разностных методов для численного решения этой задачи, отличающихся простотой и удобством распараллеливания.
- Для ускорения 3D-моделирования на сетках с миллиардами узлов необходима высокоэффективная реализация конечно-разностных вычислений.

Подходы к ускорению конечно-разностных вычислений при решении задач сейсмологии

[Datta et al., 2009; Micikevicius, 2009; Michéa, Komatitsch, 2010; Nakata et al., 2011; Weiss, Shragge, 2013; Andreolli et al., 2015; Nasciutti et al., 2018; Louboutin et al., 2019; Serpa et al., 2019]

- Известные подходы недостаточно систематизированы;
- Оптимизации либо исследуются подробно, но на упрощённых моделях, либо практические задачи решаются без анализа эффективности на различных архитектурах;
- Необходима разработка комплексных алгоритмов для 3D моделирования сейсмических полей, ориентированных на использование разнородных вычислительных архитектур с анализом производительности, масштабируемости и энергоэффективности.

Цель и задача исследования

Цель исследования – развить методы эффективной организации параллельных вычислений для численного моделирования физических процессов за счёт использования многоуровневого подхода и архитектурно-ориентированных стратегий оптимизации и реализовать их в виде программно-алгоритмического комплекса для высокопроизводительных вычислений на современных кластерных системах.

Научные задачи:

1. Разработать многоуровневый архитектурно-ориентированный метод проектирования параллельных алгоритмов и программ для численного моделирования физических процессов, сочетающий:

комплексный анализ ресурсов параллелизма,

архитектурно-зависимые стратегии оптимизации и

имитационную оценку масштабируемости параллельных алгоритмов.

2. На основе предложенного метода разработать архитектурно-ориентированный программно-алгоритмический комплекс для решения трёхмерной динамической задачи сеймики:

реализовав полный цикл проектирования, начиная от выбора математической модели и численного метода,

с исследованием его эксплуатационных характеристик на современных высокопроизводительных вычислительных системах.

Многоуровневый архитектурно-ориентированный метод проектирования параллельных алгоритмов и программ

Этап 1: Со-дизайн методов решения вычислительно сложных задач численного моделирования

- **Физическая модель**
- **Математическая модель**
- **Численный метод**
- **Разработка параллельного алгоритма**
- **Учет особенности архитектуры суперЭВМ**
- **Выбор средств параллельной разработки**

Этап 2: Имитационное моделирование масштабируемости алгоритмов

Этап 3: Анализ энергоэффективности алгоритмов и реализации

Многоуровневый архитектурно-ориентированный метод проектирования параллельных алгоритмов и программ

Этап 1: Со-дизайн методов решения вычислительно сложных задач численного моделирования

- **Физическая модель**
- **Математическая модель**
- **Численный метод**

Сравнительный анализ ресурсов параллелизма:

- требуемый объем оперативной памяти ↓
- арифметическая интенсивность ↑
- коммуникационная нагрузка ↓

- **Разработка параллельного алгоритма**
- **Учет особенности архитектуры суперЭВМ**
- **Выбор средств параллельной разработки**

Этап 2: Имитационное моделирование масштабируемости алгоритмов

Этап 3: Анализ энергоэффективности алгоритмов и реализации

Многоуровневый архитектурно-ориентированный метод проектирования параллельных алгоритмов и программ

Этап 1: Со-дизайн методов решения вычислительно сложных задач численного моделирования

- **Физическая модель**
- **Математическая модель**
- **Численный метод**

Сравнительный анализ ресурсов параллелизма:

- требуемый объем оперативной памяти ↓
- арифметическая интенсивность ↑
- коммуникационная нагрузка ↓

- **Разработка параллельного алгоритма**
- **Учет особенности архитектуры суперЭВМ**
- **Выбор средств параллельной разработки**

Архитектурно-ориент. оптимизация:

- доступ к памяти в соответствии с иерархией
- загрузка вычислительных устройств
- обмены сообщениями

	Особенности архитектуры	Приёмы оптимизации
Многоядерные CPU	<ul style="list-style-type: none"> • Высокая тактовая частота • Векторные расширения • Поддержка SMT • Большой объем кэш-памяти 	<ul style="list-style-type: none"> • Векторизация (AVX) • Блочная декомпозиция, перестановка циклов • Балансировка нагрузки с учётом NUMA-топологии
GPU	<ul style="list-style-type: none"> • Массовый параллелизм на упрощенных ядрах • Развитая иерархия памяти 	<ul style="list-style-type: none"> • Настройка конфигурации блоков и потоков • Минимизация ветвлений • Соккрытие латентности за счёт массового параллелизма • Использование разделяемой памяти
Распределённые кластеры	<ul style="list-style-type: none"> • Межузловое взаимодействие • Высокая масштабируемость 	<ul style="list-style-type: none"> • Минимизация межузловых обменов (MPI) • Перекрытие коммуникаций с вычислениями • Иерархическая схема распараллеливания

Со-дизайн: математическая модель – динамическая теория упругости для скоростей и напряжений

$\vec{u} = (u, v, w)^T$ – вектор скоростей смещения,

$\vec{\sigma} = (\sigma_{xx}, \sigma_{yy}, \sigma_{zz}, \sigma_{xy}, \sigma_{xz}, \sigma_{yz})^T$ – тензор напряжений,

$$\rho \frac{\partial \vec{u}}{\partial t} = [A] \vec{\sigma} + \vec{F}(t, x, y, z), \quad \frac{\partial \vec{\sigma}}{\partial t} = [B] \vec{u},$$

$$A = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} & 0 \\ 0 & \frac{\partial}{\partial y} & 0 & \frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial z} \\ 0 & 0 & \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} & \frac{\partial}{\partial y} \end{bmatrix}, \quad B = \begin{bmatrix} (\lambda + 2\mu) \frac{\partial}{\partial x} & \lambda \frac{\partial}{\partial y} & \lambda \frac{\partial}{\partial z} \\ \lambda \frac{\partial}{\partial x} & (\lambda + 2\mu) \frac{\partial}{\partial y} & \lambda \frac{\partial}{\partial z} \\ \lambda \frac{\partial}{\partial x} & \lambda \frac{\partial}{\partial y} & (\lambda + 2\mu) \frac{\partial}{\partial z} \\ \mu \frac{\partial}{\partial y} & \mu \frac{\partial}{\partial x} & 0 \\ \mu \frac{\partial}{\partial z} & 0 & \mu \frac{\partial}{\partial x} \\ 0 & \mu \frac{\partial}{\partial z} & \mu \frac{\partial}{\partial y} \end{bmatrix},$$

$$\vec{\sigma} |_{t=0} = 0, \quad \vec{u} |_{t=0} = 0, \quad \sigma_{xz} |_{z=0} = 0, \quad \sigma_{yz} |_{z=0} = 0, \quad \sigma_{zz} |_{z=0} = 0$$

Со-дизайн: математическая модель – динамическая теория упругости для смещений

$\vec{U} = (U, V, W)^T$ – вектор смещений,

$$\rho \frac{\partial^2 \vec{U}}{\partial t^2} = [C] \vec{U} + \vec{F}(t, x, y, z)$$

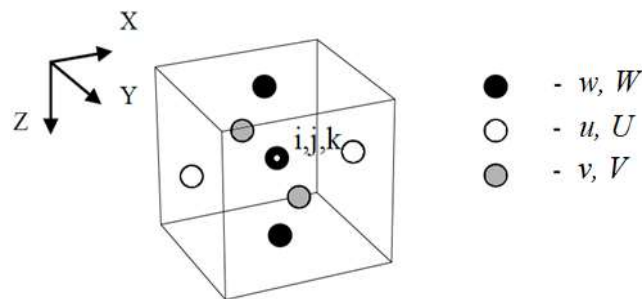
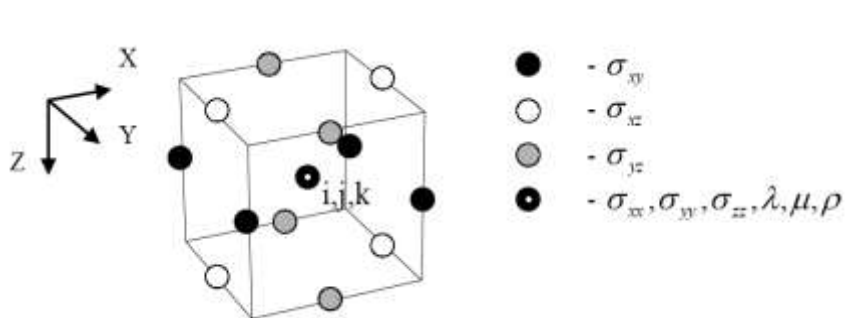
$$C = \begin{bmatrix} (\lambda + 2\mu) \frac{\partial^2}{\partial x^2} + \mu \left(\frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) & (\lambda + \mu) \frac{\partial^2}{\partial x \partial y} & (\lambda + \mu) \frac{\partial^2}{\partial x \partial z} \\ (\lambda + \mu) \frac{\partial^2}{\partial y \partial x} & (\lambda + 2\mu) \frac{\partial^2}{\partial y^2} + \mu \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2} \right) & (\lambda + \mu) \frac{\partial^2}{\partial y \partial z} \\ (\lambda + \mu) \frac{\partial^2}{\partial z \partial x} & (\lambda + \mu) \frac{\partial^2}{\partial z \partial y} & (\lambda + 2\mu) \frac{\partial^2}{\partial z^2} + \mu \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \end{bmatrix}$$

$$\vec{U} |_{t=0} = 0, \quad \sigma_{xz} |_{z=0} = 0, \quad \sigma_{yz} |_{z=0} = 0, \quad \sigma_{zz} |_{z=0} = 0$$

Со-дизайн: численный метод решения уравнений теории упругости

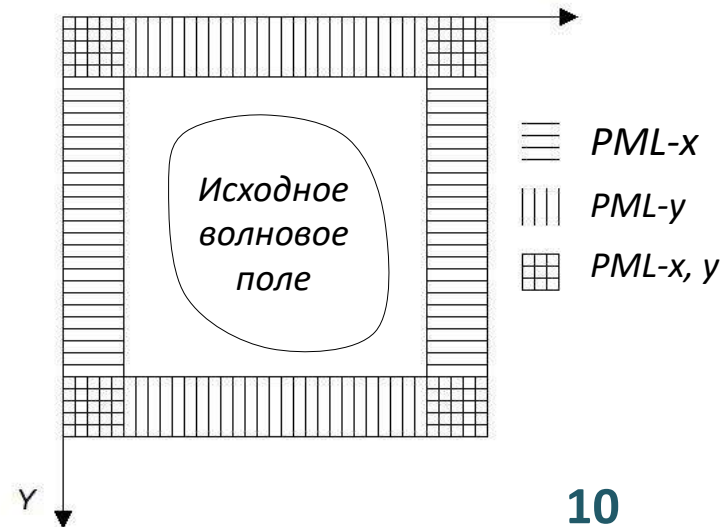
○ Однородные явные конечно-разностные схемы на сдвинутых сетках с модификацией сеточных коэффициентов в окрестности границ разрыва

- схема 2-го порядка аппроксимации для расчета \vec{u} и $\vec{\sigma}$ [Virieux, 1986];
- схема 4-го порядка аппроксимации для расчета \vec{u} и $\vec{\sigma}$ [Levander, 1988];
- схема 2-го порядок аппроксимации для расчета \vec{U} [Moczso et al., 2002];



○ Для устранения ложных отражений от границ расчетной области применяется метод поглощающих границ CFS-PML

- Эффективное поглощение различных типов волн без увеличения вычислительной нагрузки
- Экономичен с точки зрения памяти
- Величина поглощающего слоя составляет одну минимальную длину волны



Со-дизайн методов решения трехмерной динамической задачи сейсмоки

Этап 1: Со-дизайн методов решения вычислительно сложных задач

- **Физическая модель:** динамическая теория упругости для изотропных сред
- **Математическая модель:** две постановки, различные по количеству уравнений и неизвестных
- **Численный метод:** три явные конечно-разностные схемы на сдвинутых сетках 2-го и 4-го порядка аппроксимации

Неизвестные	\vec{u} и $\vec{\sigma}$		\vec{U}
	2ой	4ый	2ой
Порядок по пространству	2ой	4ый	2ой
Кол-во уравнений	9	9	3
Кол-во 3D массивов	17	17	14
Арифметическая интенсивность, Флоп/байт	0.47	0.93	1.43
Коммуникационная нагрузка	$O(3N^2)$	$O(6N^2)$	$O(3N^2)$

- **Разработка параллельного алгоритма**
- **Учет особенности архитектуры суперЭВМ**
- **Выбор средств параллельной разработки**

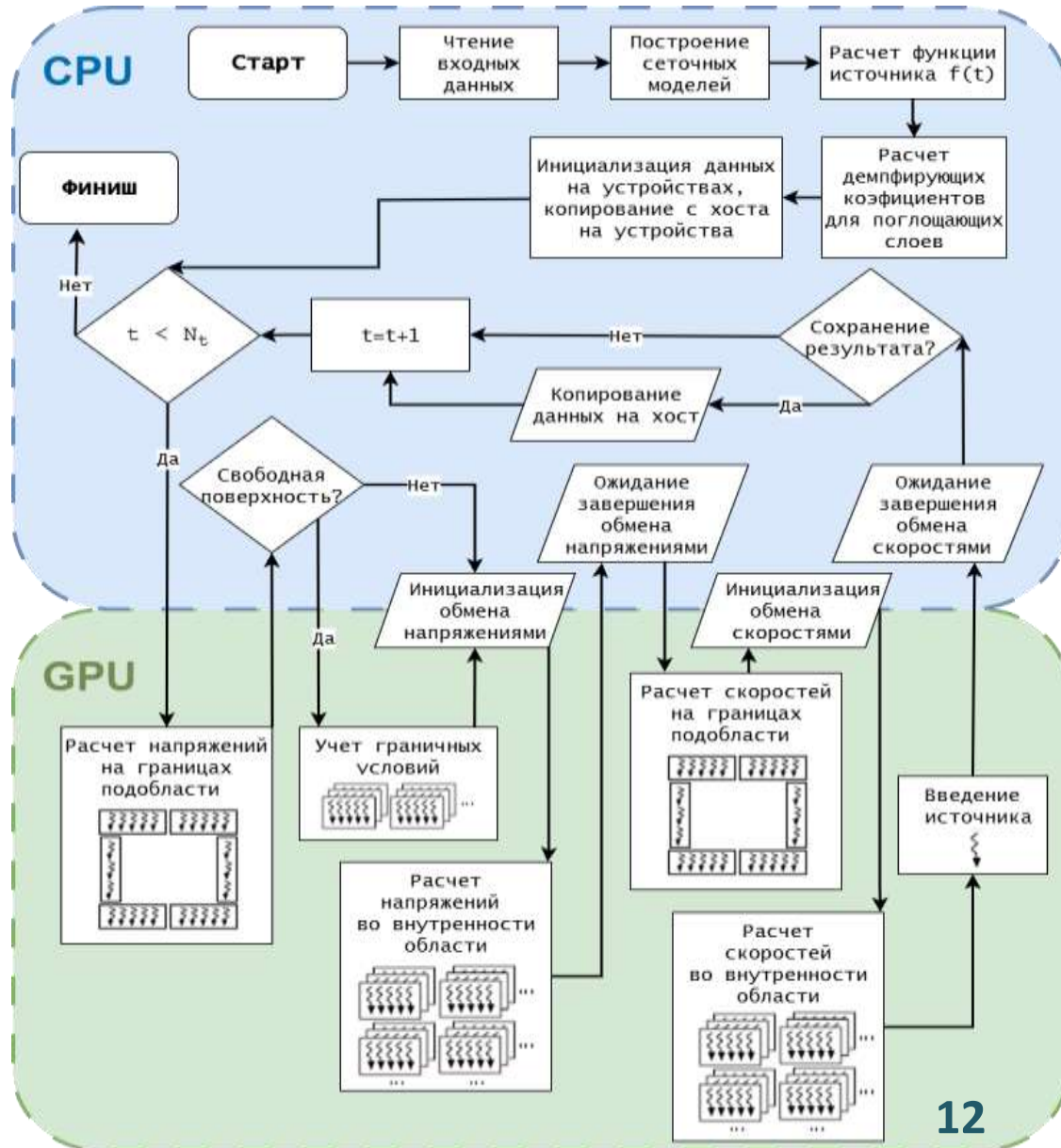
- Рассматриваются **две принципиально разные группы вычислителей:** многоядерные процессоры и графические ускорители
- Исследуется эффективность реализованных архитектурно-ориентированных приемов оптимизации

Этап 2: Имитационное моделирование масштабируемости алгоритмов

Этап 3: Анализ энергоэффективности алгоритмов и реализации

Разработка параллельного алгоритма: модуль для гибридного кластера с GPU

- На **CPU** выполняются вычислительно малонагруженные подготовительные этапы алгоритма.
- Вычислительно интенсивные расчеты по конечно-разностным формулам выполняются на **GPU**.
- На определенных временных слоях рассчитанное волновое поле в нескольких плоскостях копируется с устройства на хост для сохранения снимков поля и записи теоретических сеймотрасс.
- Иерархическая схема распараллеливания: **MPI + CUDA**.



Асинхронная схема обменов для гибридного кластера с GPU на основе 2D декомпозиции

Обмены между узлами:
неблокированная передача данных MPI

Обмены между ускорителями:
функции асинхронного копирования CUDA

1. На текущем временном шаге рассчитываются значения искомых величин на границах подслоев;

2. Рассчитанные на шаге 1 значения копируются с устройства на хост;

3. Инициализируются обмены между вычислительными узлами (обновление зоны гало слоев);

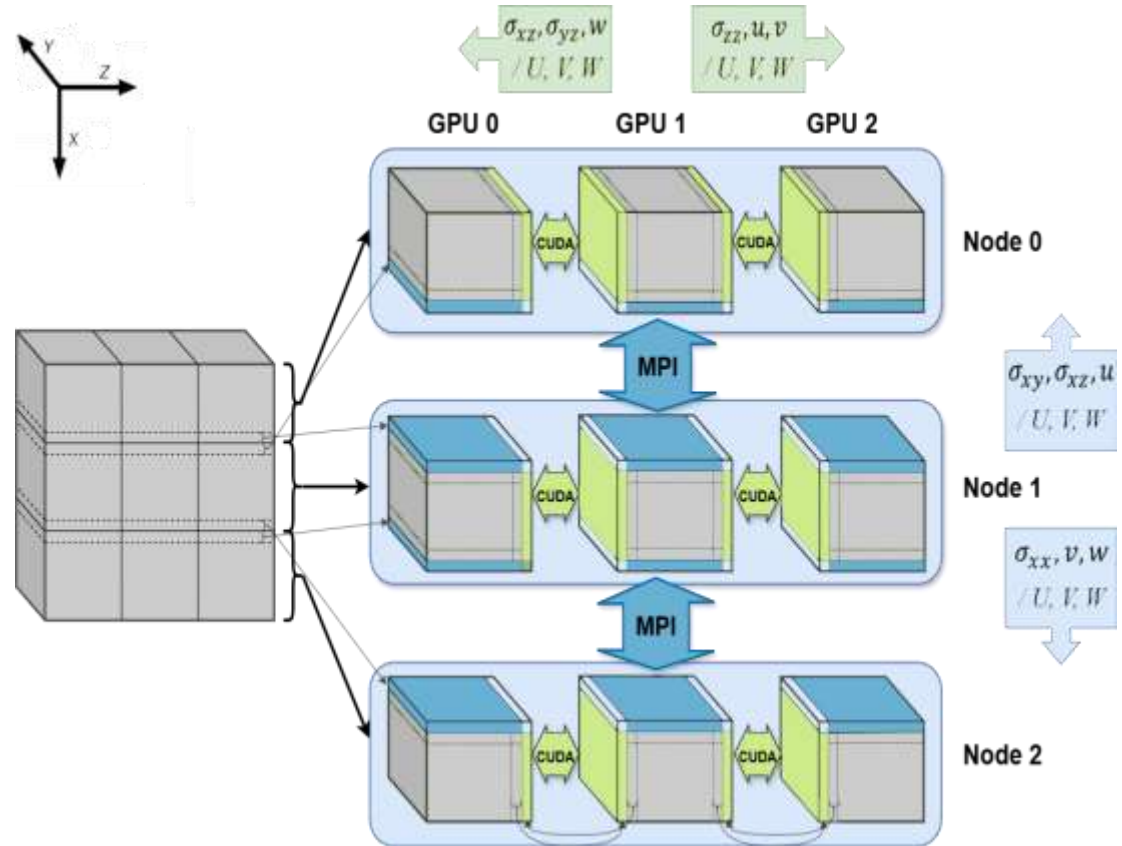
4. Инициализируются обмены между GPU на узле (обновление зоны гало подслоев);

5. Рассчитываются искомые значения волнового поля во внутренней части подслоев;

6. Проверяется завершение обменов, инициализированных на шаге 3 и 4;

7. Данные, полученные в ходе обмена между узлами, копируются на устройства;

8. Переход на следующий временной шаг.



Особенности распределения вычислений между ядрами GPU

Выполнены три серии вычислительных экспериментов с разным размером блока нитей для GPU с архитектурами **Fermi**, **Kepler** и **Pascal**:

- 3D блок, 3D сеть блоков
- Размер блока по компоненте x равный 32 или 64;
- Размер блока по компонентам y и z одинаковый и равный 4 или 2;
- Максимально возможный размер блока не обязательно выигрышный.

Размер блока 1024	Ускорение	Размер блока 512	Ускорение	Размер блока 256	Ускорение	Размер блока 128	Ускорение
16×8×8	1.0	16×8×4	1.02	16×4×4	1.02	16×4×2	0.98
32×8×4	1.25	32×4×4	1.29	32×4×2	1.23	32×2×2	1.23
4×64×4	0.49	32×8×2	1.23				
4×4×64	0.48	32×16×1	1.12				
64×4×4	1.24	64×4×2	1.26	64×2×2	1.25	64×2×1	1.08
128×4×2	1.25	128×2×2	1.27	128×2×1	1.12	128×1×1	1.08
256×2×2	1.22	256×2×1	1.14	256×1×1	1.13		
512×2×1	1.1	512×1×1	1.14				

Использование низколатентных видов памяти GPU

- Использование **константной памяти**: основные константы, используемые на каждом временном шаге (прирост производительности на 4 %)
- Использование **shared memory** позволяет уменьшить число обращений к глобальной памяти, выигрыш в производительности растёт с увеличением степени переиспользования данных

Размер блока нитей	Ускорение		
	Tesla M2090 Fermi	Tesla K40 Kepler	Tesla P100 Pascal
16×8×8	–	1.2	1.3
16×4×4	1.0	1.2	1.3
32×4×4	0.9	1.0	1.3
64×2×2	0.8	0.6	1.3
64×4×2	0.9	–	1.3
64×4×4	–	0.98	1.5
128×2×2	0.9	0.6	1.2

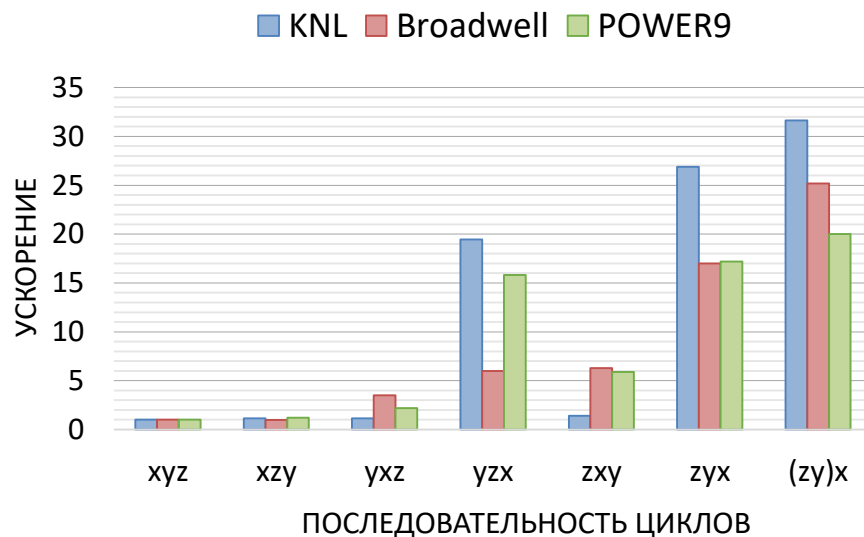
Разработка параллельного алгоритма: модуль для CPU-кластера

- Эффективность доступа к памяти: все основные массивы выравниваются
- Внешний цикл параллелизуется с использованием OpenMP
- Внутренний цикл векторизуется
 - для Intel с технологиями AVX2/AVX-512
 - Автовекторизация vs. встроенные функции (intrinsics)
 - Ускорение в несколько раз (x3 для Intel KNL)

```
for all time steps do
    #pragma omp paraller for...
    for all X points do
        for all Y points do
            #pragma simd
            for all Z points do
                U,V,W
                computations
            end for
        end for
    end for
    Snapshot check
end for
```

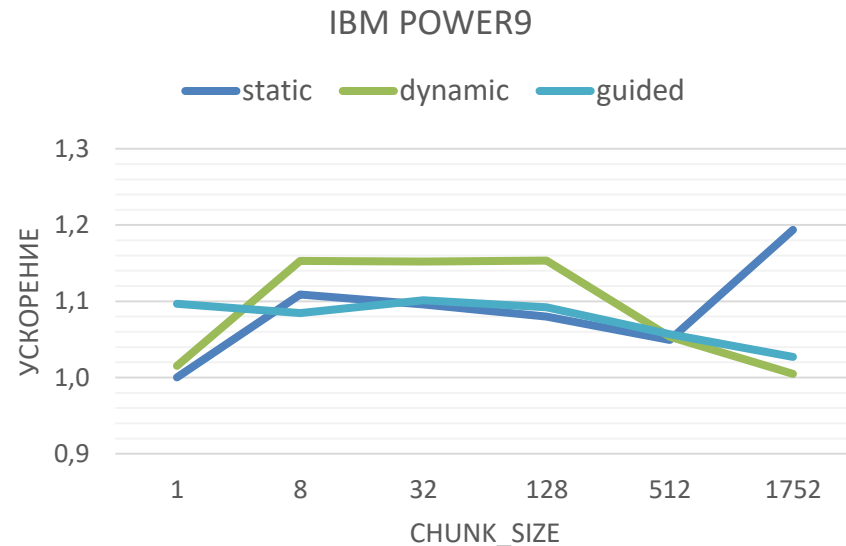
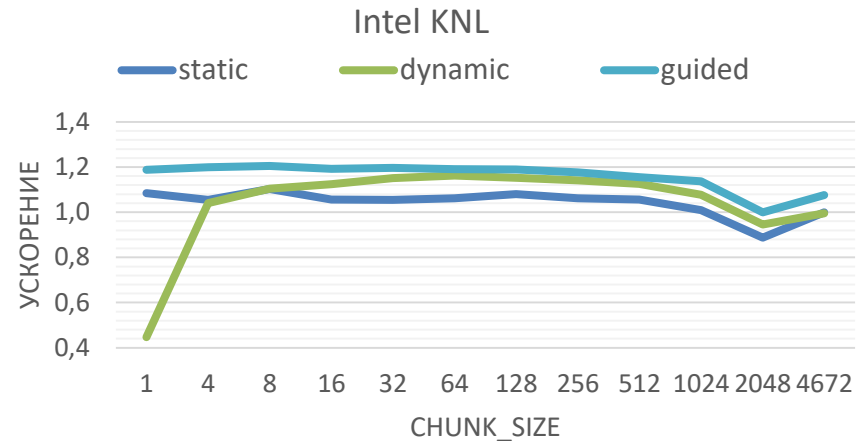
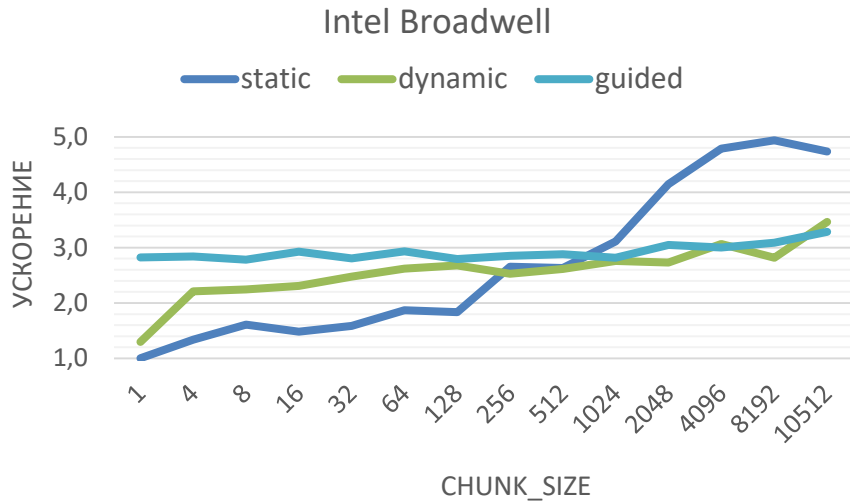
Оптимизации

- Кэширование: различная последовательность циклов – xyz / xzy / yxz / yzx / zxy / zyx
- POWER9**: система кэш-памяти больше и быстрее, по сравнению с Broadwell и KNL → yzx ≈ zyx .
- KNL**: отсутствует кэш L3 → максимальное ускорение для последовательности zyx (в 27 раз).
- Балансировка нагрузки: свёртка двух внешних циклов в один – (zy)x



Эмпирический анализ и оптимизация параметров планирования OpenMP

Для балансировки загрузки ядер CPU выполняется адаптивный выбор стратегии распределения итераций циклов на основе настройки типа расписания OpenMP и размера блока (`chunk_size`).



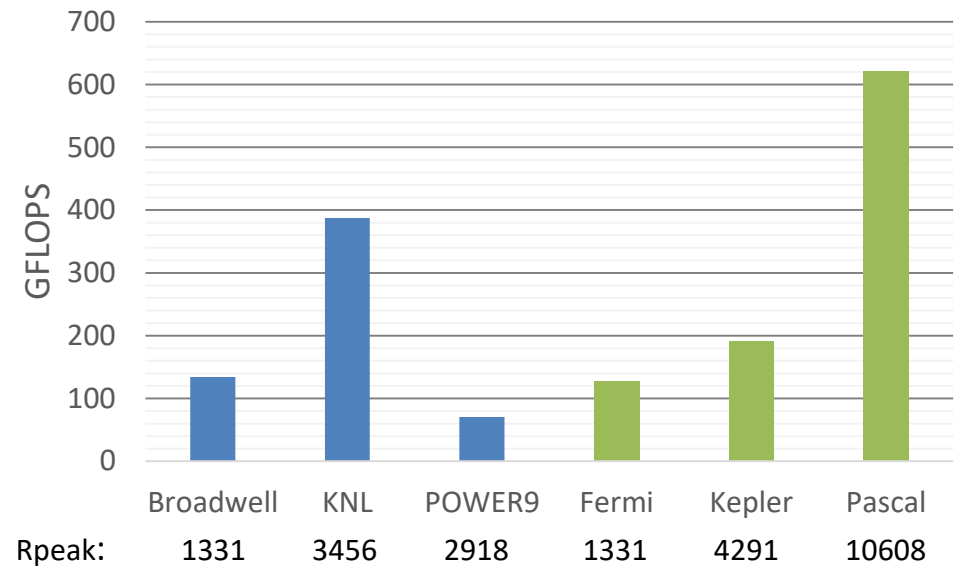
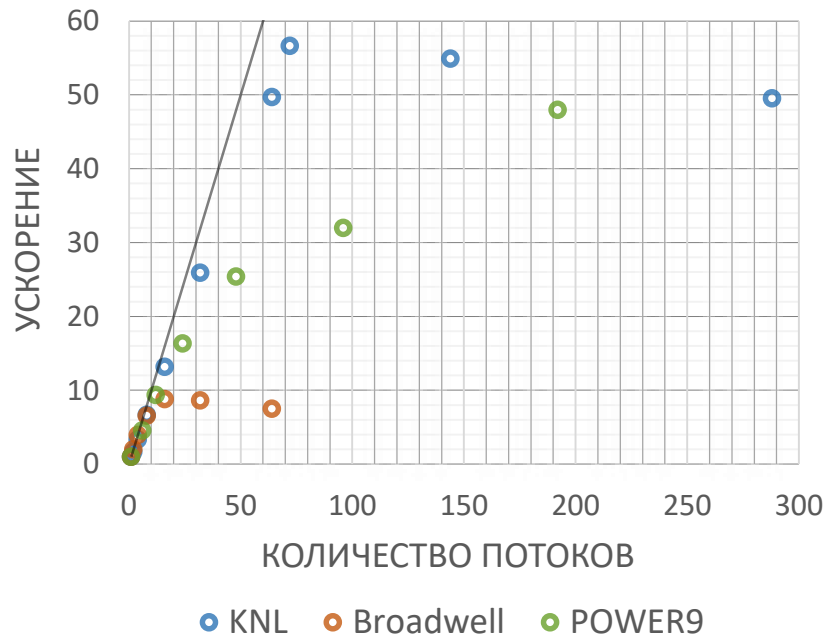
POWER9: влияние планировщика минимально, лучшее ускорение при статическом распределении.

Broadwell: оптимально статическое распределение, динамическое и управляемое с малым `chunk_size` снижают ускорение.

KNL: лучший результат даёт управляемое распределение с небольшим размером блока (ускорение $\times 1.2$).

Дополнительно: 16 Гб MCDRAM память в режиме ручного управления (flat mode): ускорение в 1.3 раза

Сравнительный анализ характеристик работы алгоритма по сильной масштабируемости и производительности для расчета смещений



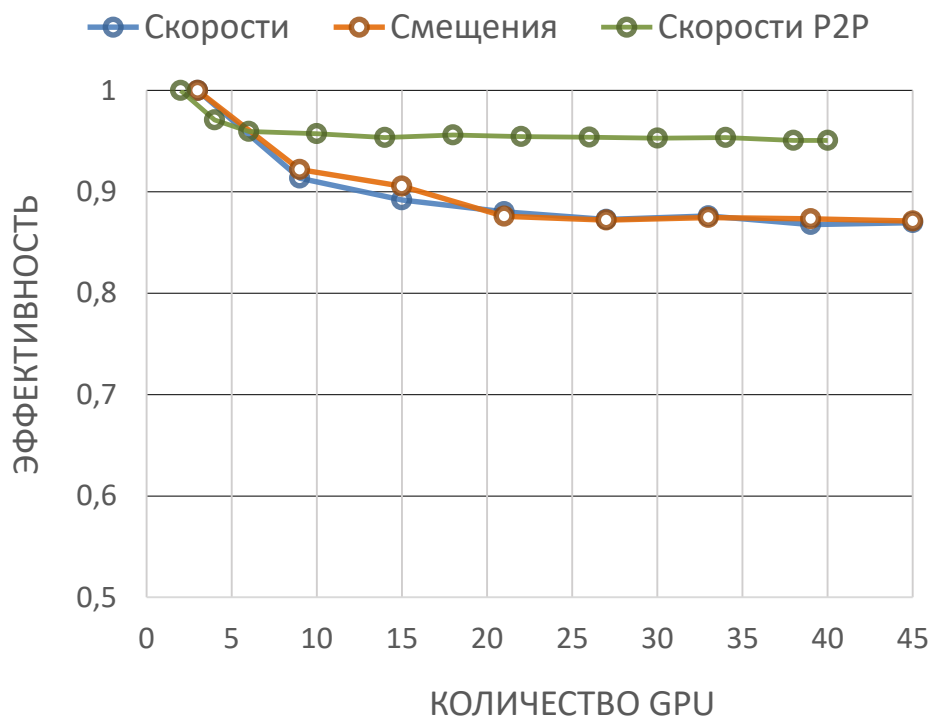
Одновременная многопоточность:

- для процессоров Intel лучше подходит – 1 поток на ядро
- для процессоров IBM – максимальное число потоков на ядро (SMT8)

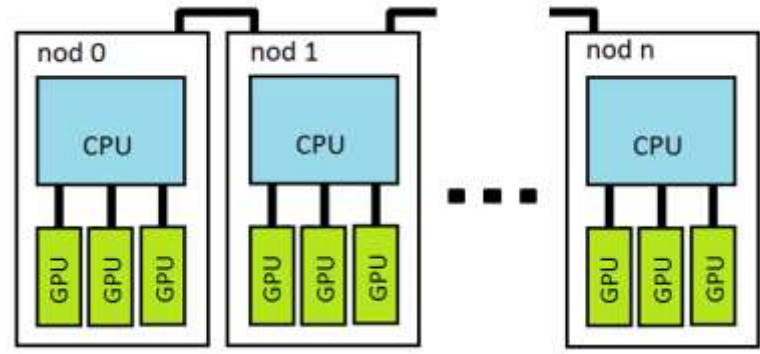
Процессор KNL – 388 Гфлопс, благодаря максимальному числу ядер с 512-битной векторизацией и высокой пропускной способности MCDRAM.

Графический ускоритель Tesla P100 (Pascal) – 621 Гфлопс, более чем в 3.2 раза выше других GPU и в 1.6 раза выше KNL при 3-кратном преимуществе в пиковой производительности

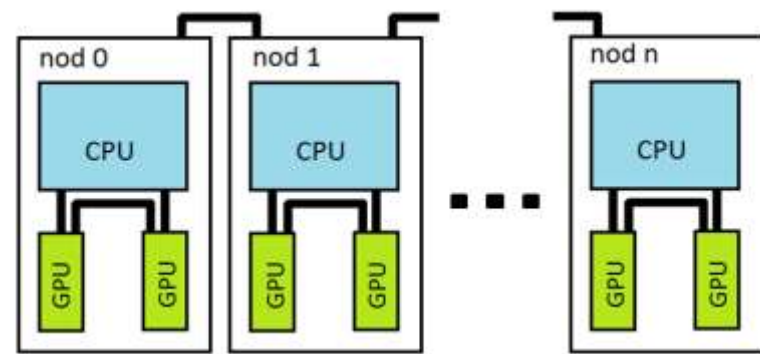
Сравнительный анализ характеристик работы алгоритма по слабой масштабируемости на гибридном кластере для схем 2-го порядка



ССКЦ СО РАН: кластер НКС-30Т+GPU



P2P конфигурация: GPU – GPU



Тип расчета*	Время, с на 45 GPU
Скорости смещения и напряжения	183.1
Смещения	174.8

*Расчетная сетка 1500x700x2100 узлов и 1000 шагов по времени

Построение имитационной модели взаимодействия параллельных процессов на основе передачи сообщений

Модель строится на основе

- схемы взаимодействия вычислительных процессов
- оценок времени вычислений и коммуникаций на основе линейных приближений:

$$T_U^{execut} = T^U \frac{LN_y N_z}{c},$$

$$T^{Comm} = T_{GPU}^{Comm} + T_{NOD}^{Comm},$$

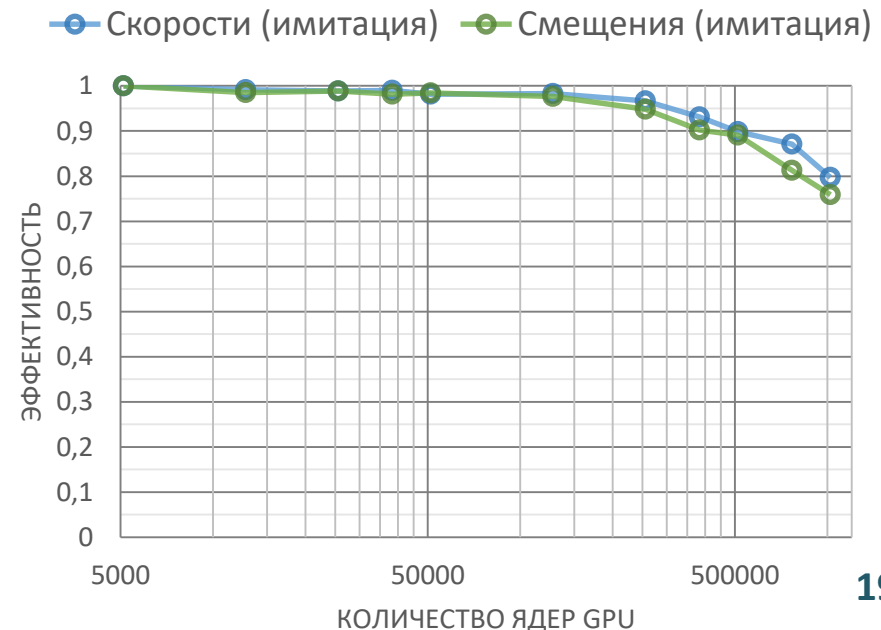
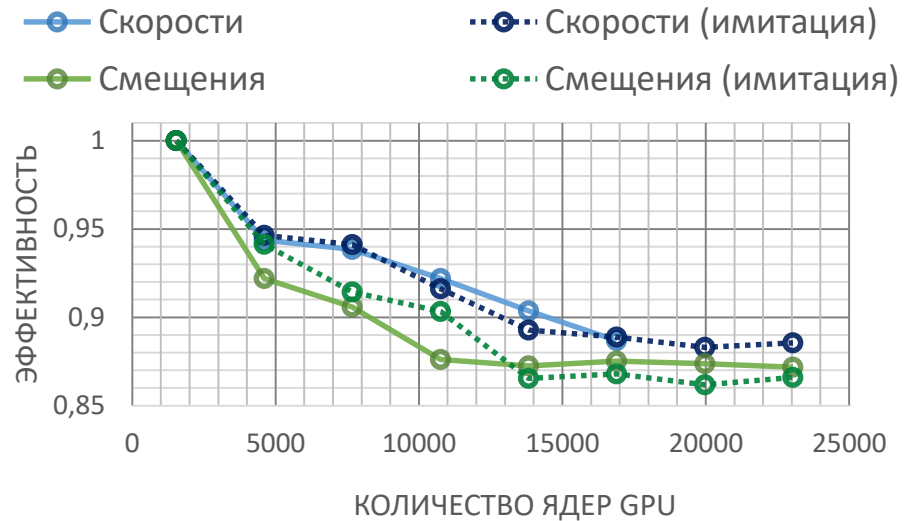
$$T_{NOD}^{Comm} = T_{to\ host}^{copy} + T_{to\ dev}^{copy} + \Lambda^{NOD},$$

$$T_{to\ host}^{copy} = T_{to\ dev}^{copy} = 2N_y N_z T_c^{GPU} + 18\Lambda^{GPU},$$

$$T_{GPU}^{Comm} = 24L\Lambda^{GPU} + 12LN_y T_c^{GPU}.$$

- характерных времен удельных вычислений и коммуникаций, полученных на основе исследования профиля исполнения:

	$T, \text{нс}$	$\Lambda^{NOD}, \text{мкс}$	$\Lambda^{GPU}, \text{мкс}$	$T_c^{GPU}, \text{нс}$
\bar{U}	810.8	5806.4	14.5	9.2
\bar{u}	1365.4	2871.8	13.8	3.9
$\bar{\sigma}$	2434.3	4458.4	4.5	3.6

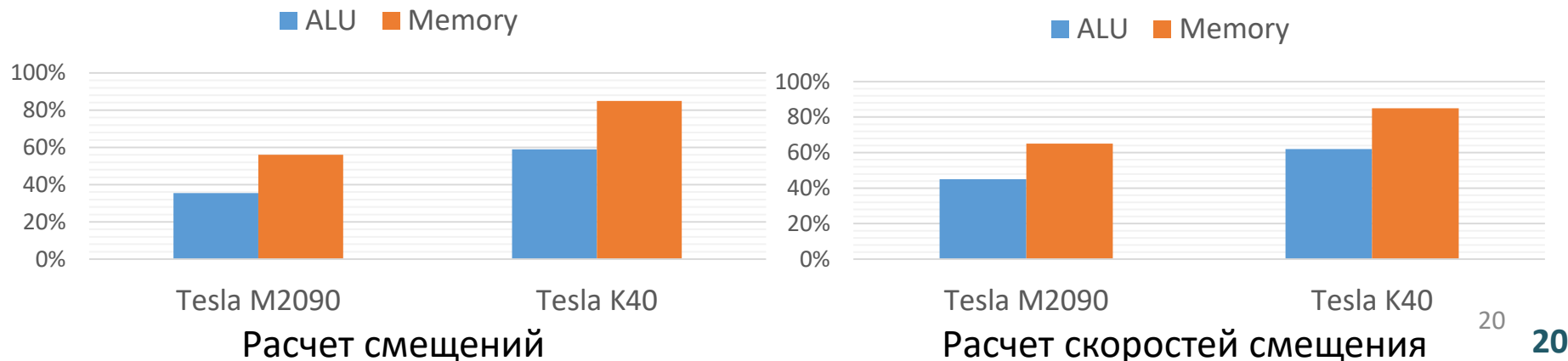


Сравнительный анализ энергоэффективности для GPU

Разработка энергоэффективных алгоритмов критически важна для экзафлопсных вычислений.

Ускоритель	Реализация	Энергоэффективность (Гфлопс/Вт)
NVIDIA Tesla K40	Расчет скоростей смещения	12.0
	Расчет смещений	9.0
NVIDIA Tesla M2090	Расчет скоростей смещения	4.5
	Расчет смещений	4.3

Энергоэффективность GPU Tesla K40 оказалась почти в 3 раза выше, чем у более старого поколения Tesla M2090, при выполнении одного и того же кода.



Сопоставление времен расчета скоростей смещения по разным схемам

Времена расчета скоростей на основе конечно-разностных схем 2-го и 4-го порядков

Размер моделируемой 3D области 4 км по каждому координатному направлению, модельное время 1.5 секунды, частота источника 8 Гц.

Кол-во точек сетки на длину волны, h, τ		NVIDIA M2090		Intel KNL	
		2-ой	4-ый	2-ой	4-ый
10 точек, $h = 0.0125, \tau = 0.003$	Порядок схемы	2-ой	4-ый	2-ой	4-ый
	Время, с	8.84	10.7	11.72	14.51
5 точек, $h = 0.025, \tau = 0.006$	Порядок схемы	4-ый		4-ый	
	Время, с	0.72		1.49	

Время решения практических задач

2D задача на сетке 6000×9000 узлов по пространству и 25000 шагов по времени решается за **12 минут** на **3 GPU NVIDIA M2090**

3D задача на сетке 1677×1059×971 узлов по пространству и 10313 шагов по времени решается за **2 ч 56 мин** на **30 GPU NVIDIA M2090** (15 вычислительных узлов).

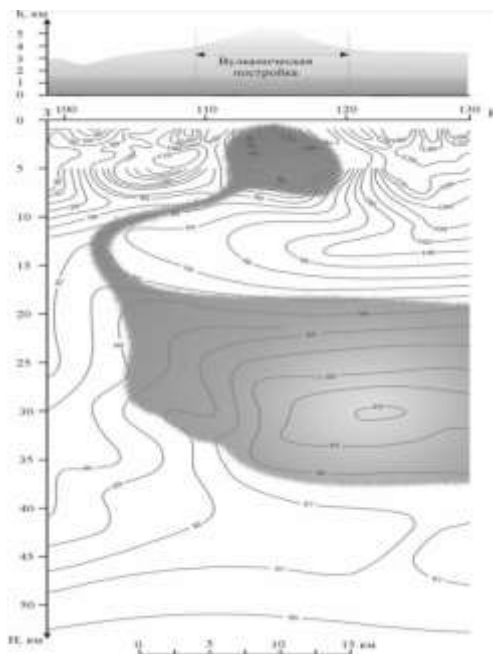
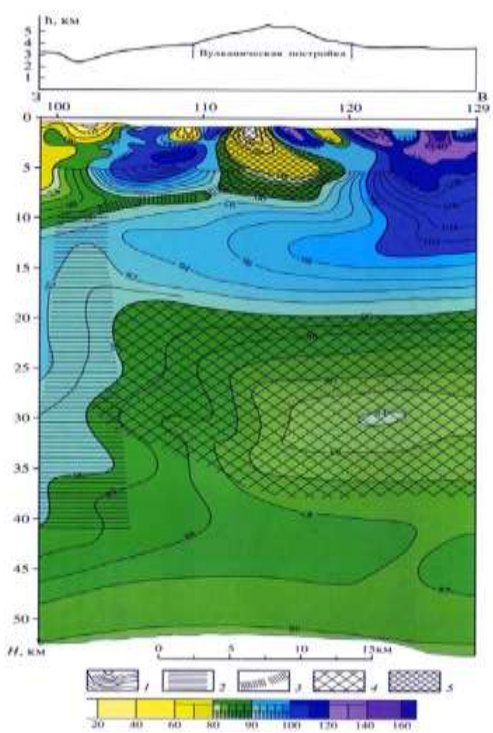
Моделирование сейсмических полей в районе вулкана Эльбрус

Собисевич А.Л. Избранные задачи математической геофизики, вулканологии и геоэкологии. Т. 1. М.: ИФЗ РАН. 2012.

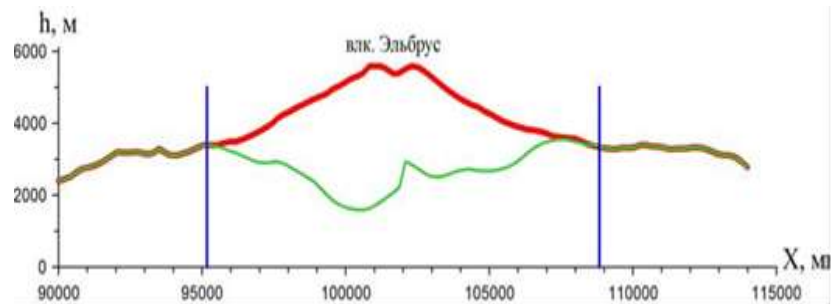
Породы, слагающие вулкан Эльбрус	ρ , кг/м ³
Граниты	2620
Кристаллические сланцы	2740
Эффузивные породы, слагающие вулканический конус	от 1800 до 2500
Породы, подстилающие вулканический конус Эльбруса	от 2650 до 2670
Плотность для более глубоких зон, лежащих ниже уровня моря	от 2750 до 2800

Значения скоростных параметров в земной коре района Кавказских Мин. Вод и Приэльбрусья

Интервал глубин, км	V_p , км/с	V_s , км/с
0 – 1	3.1 – 3.3	1.8 – 1.85
1 – 6	5.8 – 6.0	3.35 – 3.47
6 – 11	6.2 – 6.25	3.58 – 3.61
11 – 15	5.8 – 5.85	3.35 – 3.38
15 – 24	5.95 – 6.0	3.43 – 3.47
24 – 29	6.4 – 6.45	3.7 – 3.74
29 – 44	6.9 – 7.0	4.0 – 4.05
44	8.1	4.68



[Собисевич, Лиходеев, 2008]

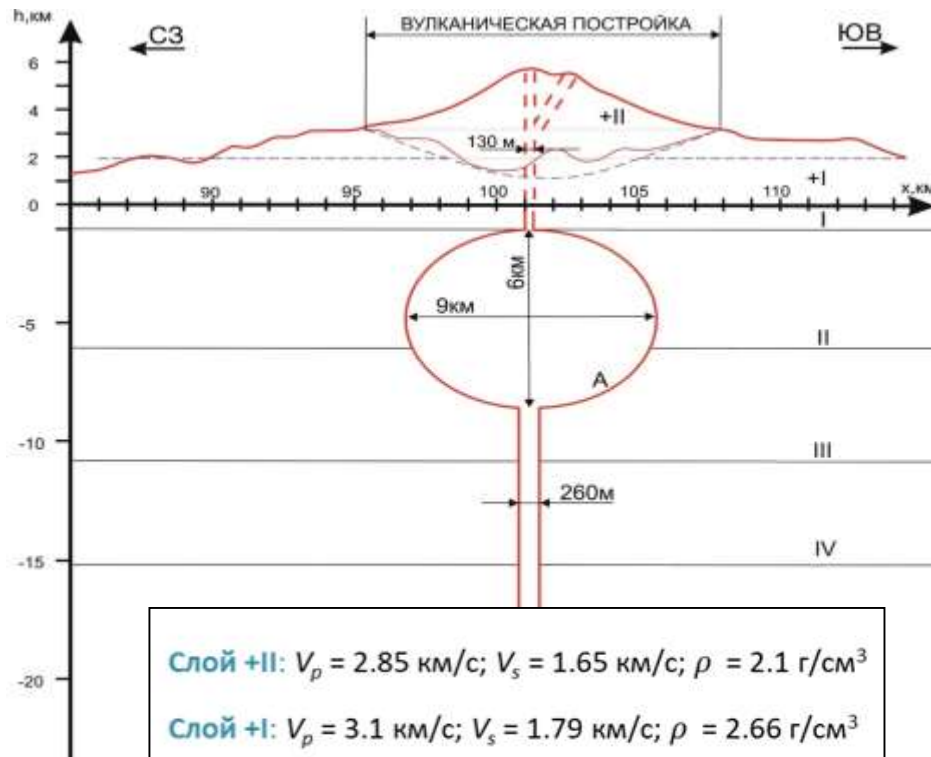


[Уткин и др., 2009]

Сейсмогеологические модели стратовулкана Эльбрус

Схематичная модель вулкана центрального типа

Уточненная модель



Слой +II: $V_p = 2.85$ км/с; $V_s = 1.65$ км/с; $\rho = 2.1$ г/см³

Слой +I: $V_p = 3.1$ км/с; $V_s = 1.79$ км/с; $\rho = 2.66$ г/см³

Слой I: $V_p = 3.2$ км/с; $V_s = 1.82$ км/с; $\rho = 2.7$ г/см³

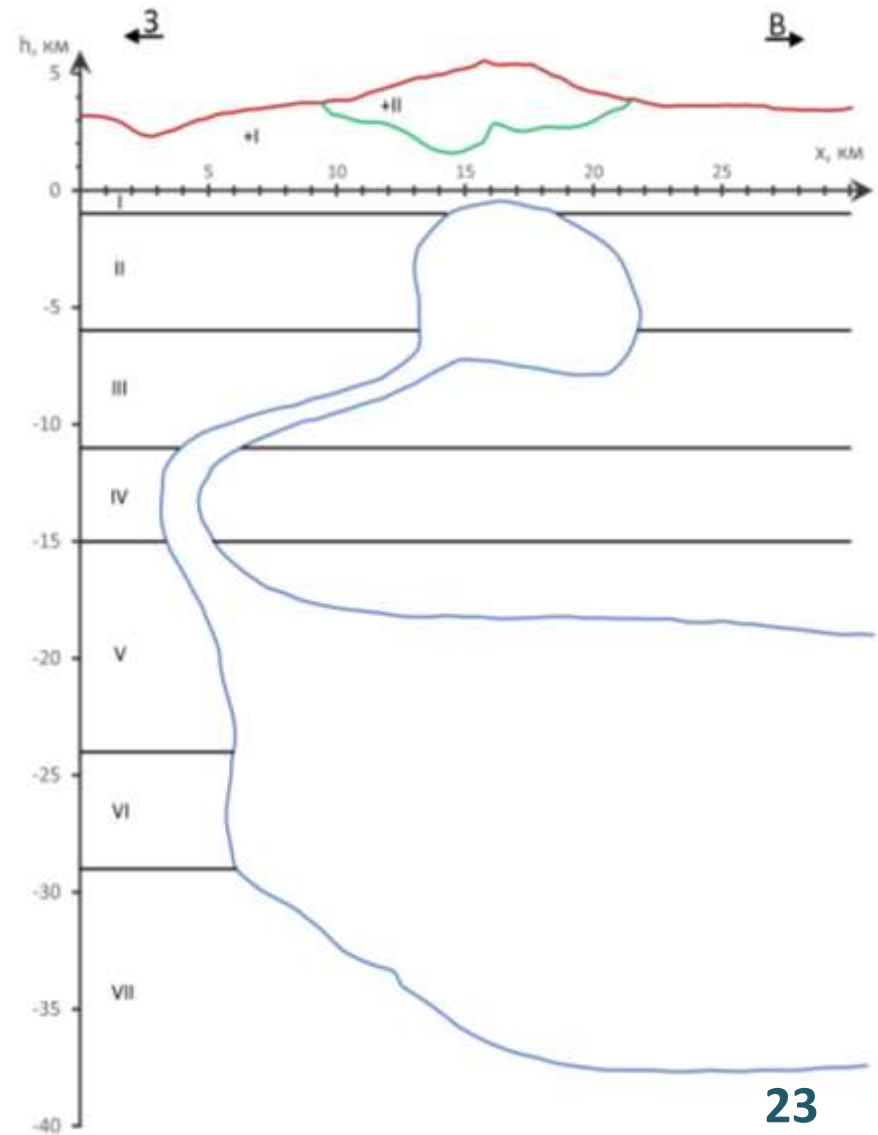
Слой II: $V_p = 5.9$ км/с; $V_s = 3.42$ км/с; $\rho = 2.85$ г/см³

Слой III: $V_p = 6.22$ км/с; $V_s = 3.59$ км/с; $\rho = 2.6$ г/см³

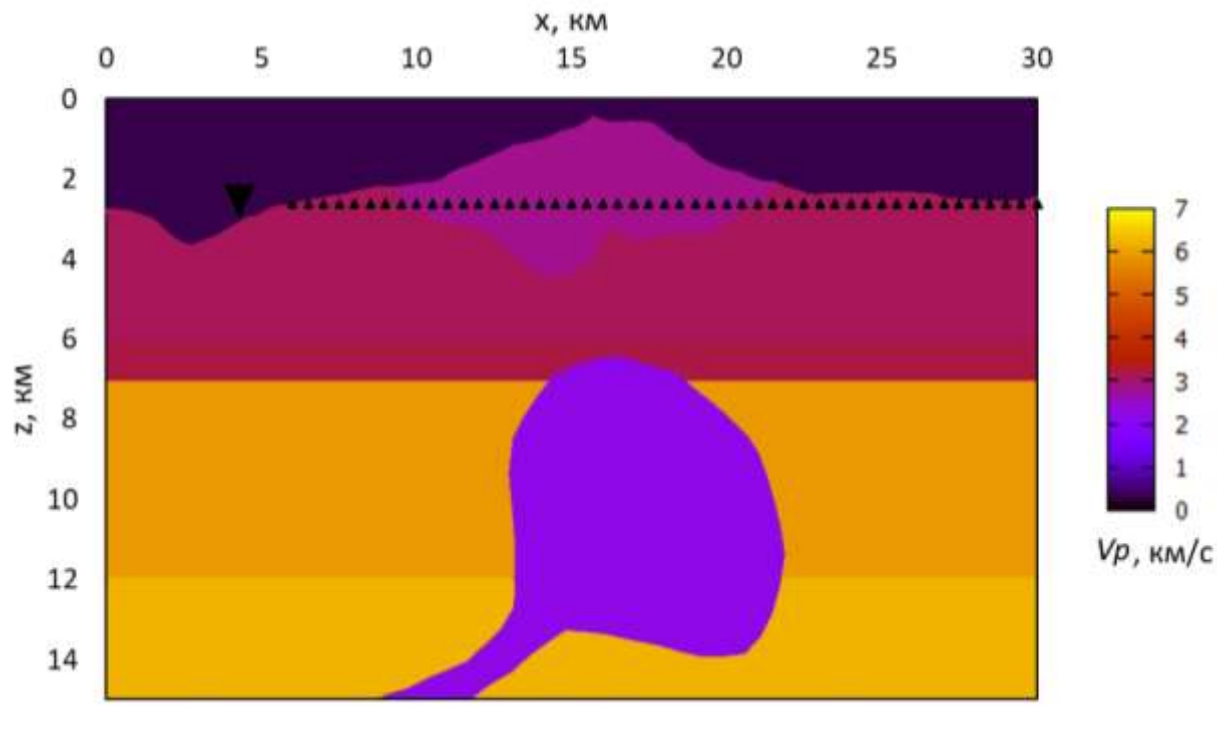
Слой IV: $V_p = 5.82$ км/с; $V_s = 3.37$ км/с; $\rho = 2.7$ г/см³

Слой V: $V_p = 5.97$ км/с; $V_s = 3.45$ км/с; $\rho = 2.75$ г/см³

Магматический очаг: $V_p = 2.2$ км/с; $\rho = 2.1$ г/см³



Моделирование вибросейсмического просвечивания схематичной и уточненной моделей вулкана Эльбрус



Размер расчетной области: 30 км по оси X и 15 км по оси Z.

Система наблюдения: линия приемников с шагом в 500 м на глубине 2.6 км

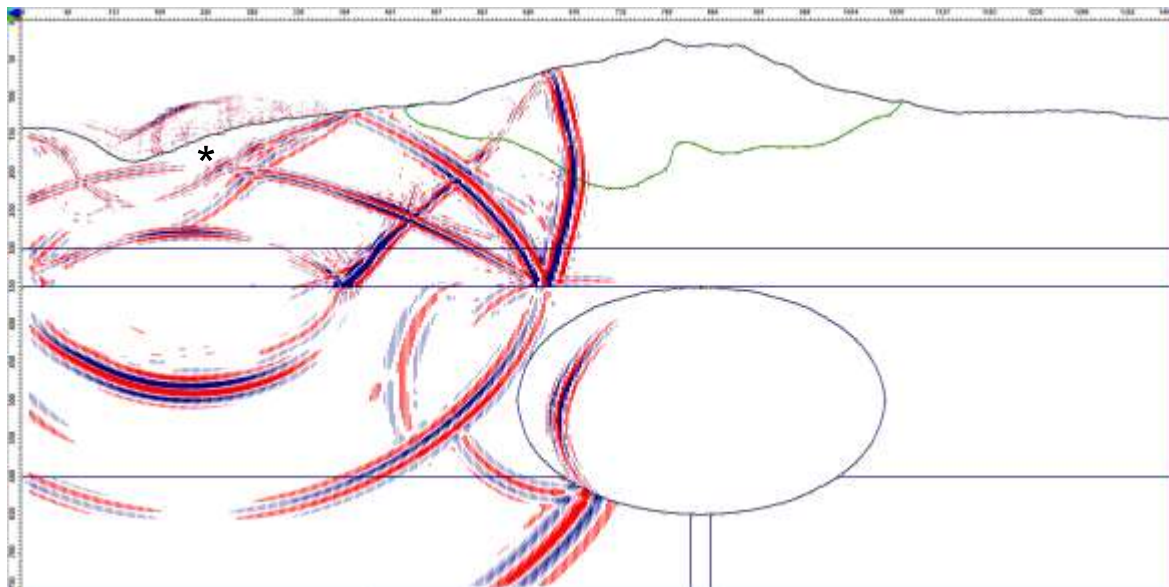
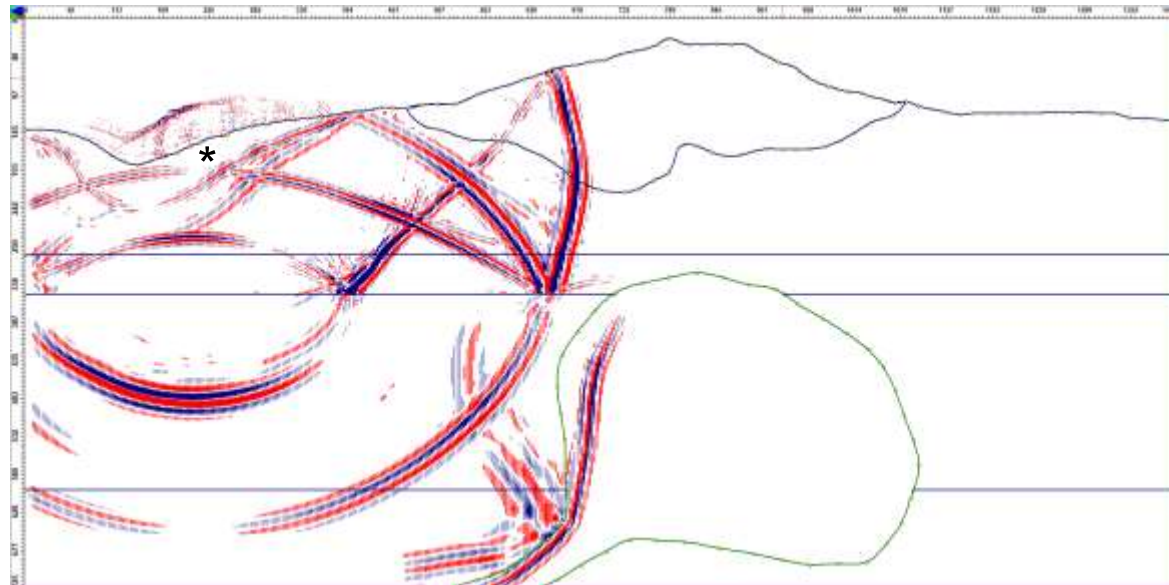
Система возбуждения: источник типа «центр давления» (импульс Пузырева с несущей частотой 8 Гц)

Расчетная сетка: 7500×3750 узлов

Один расчет на 30000 шагов на узле с NVIDIA GPU K40 (ССКЦ СО РАН) занимает около 13 мин.

Результаты 2D моделирования

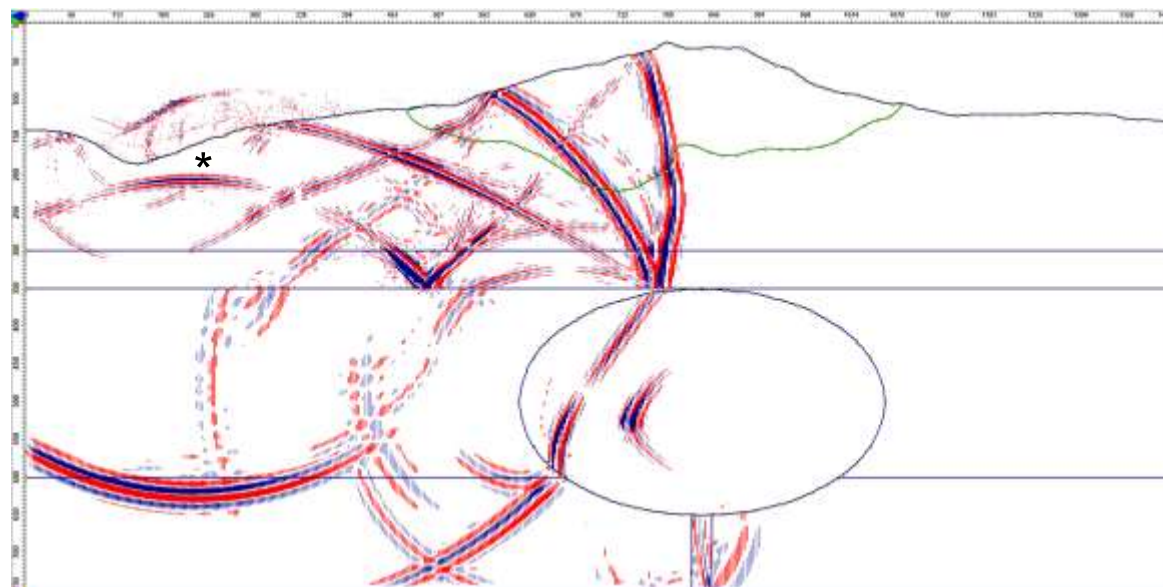
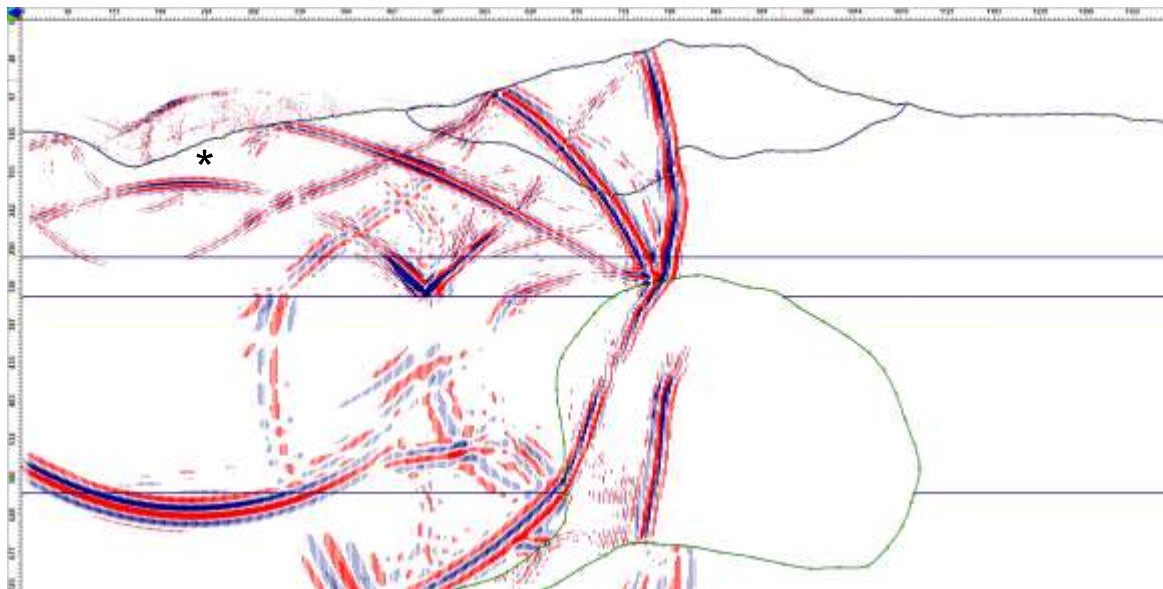
u, плоскость XZ
T = 2,4 с



Результаты 2D моделирования

u, плоскость XZ

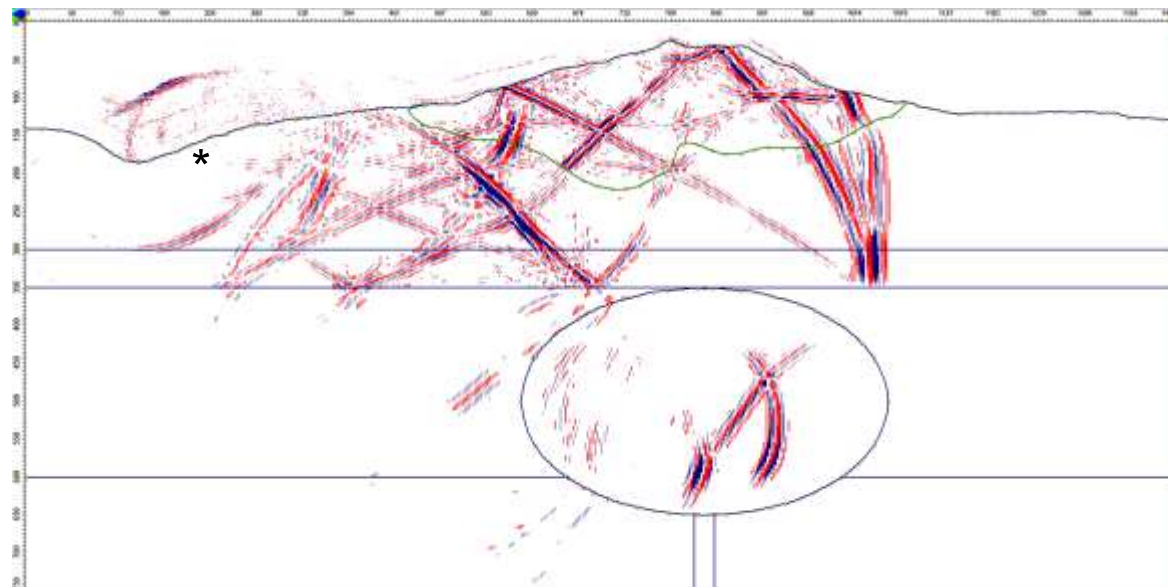
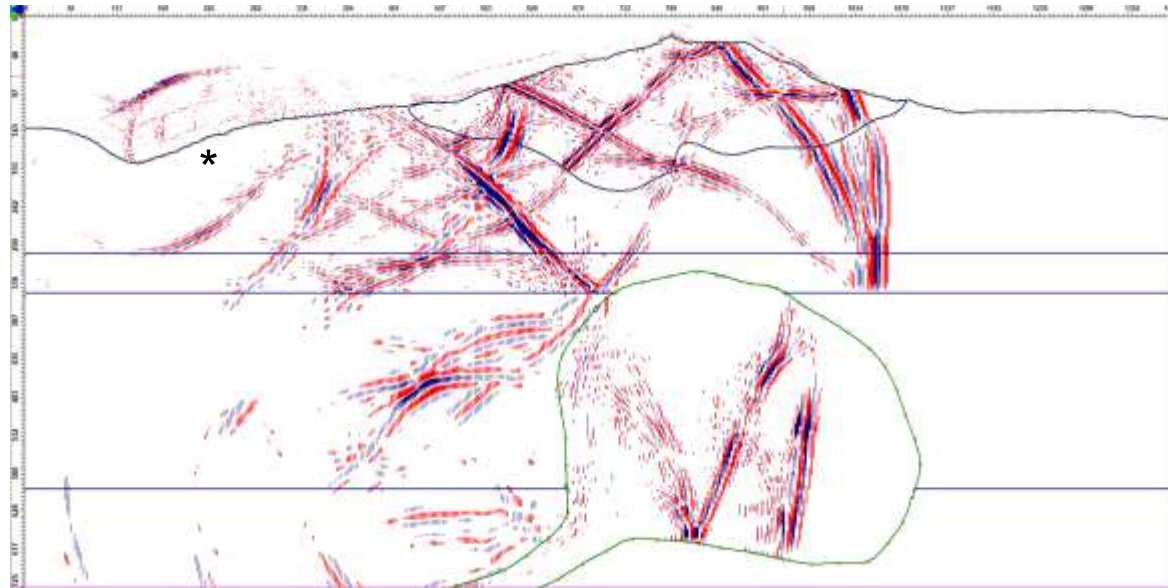
$T = 3,2$ с



Результаты 2D моделирования

u, плоскость XZ

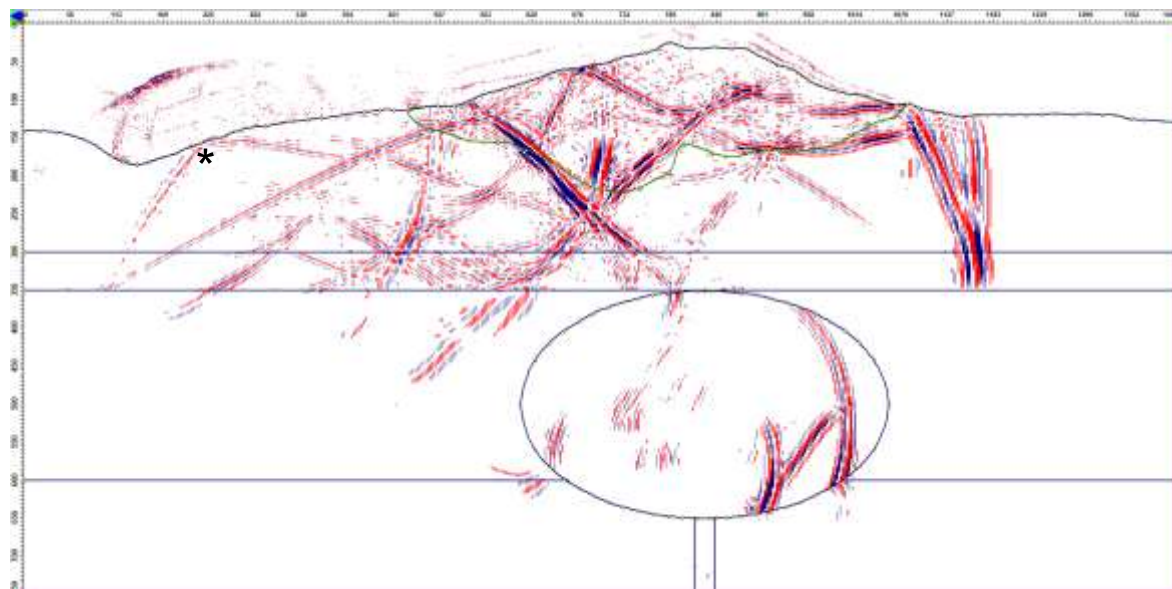
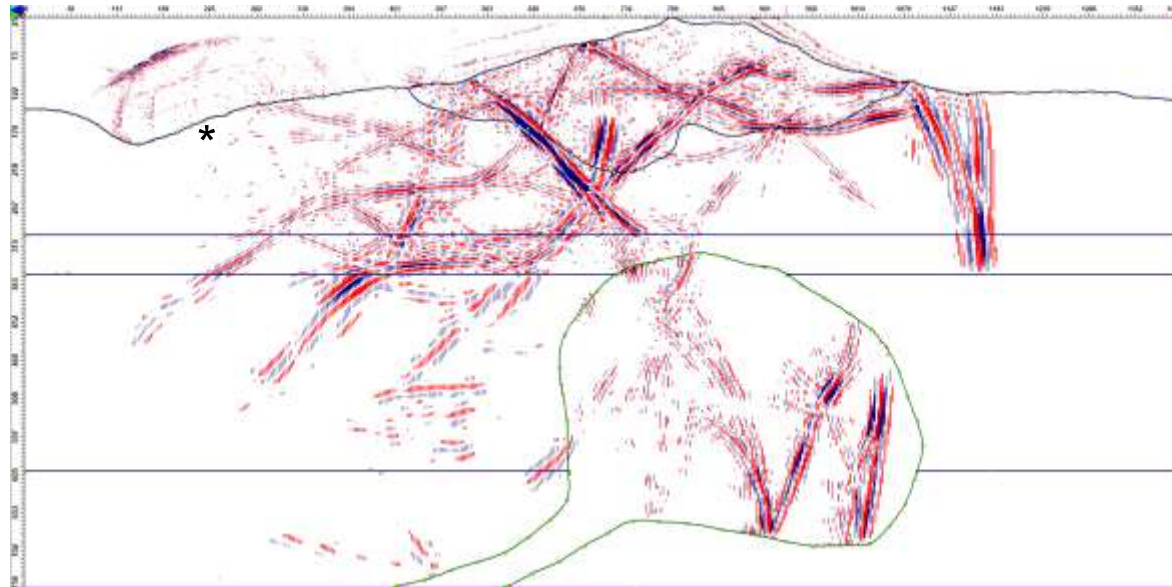
$T = 4,8$ с



Результаты 2D моделирования

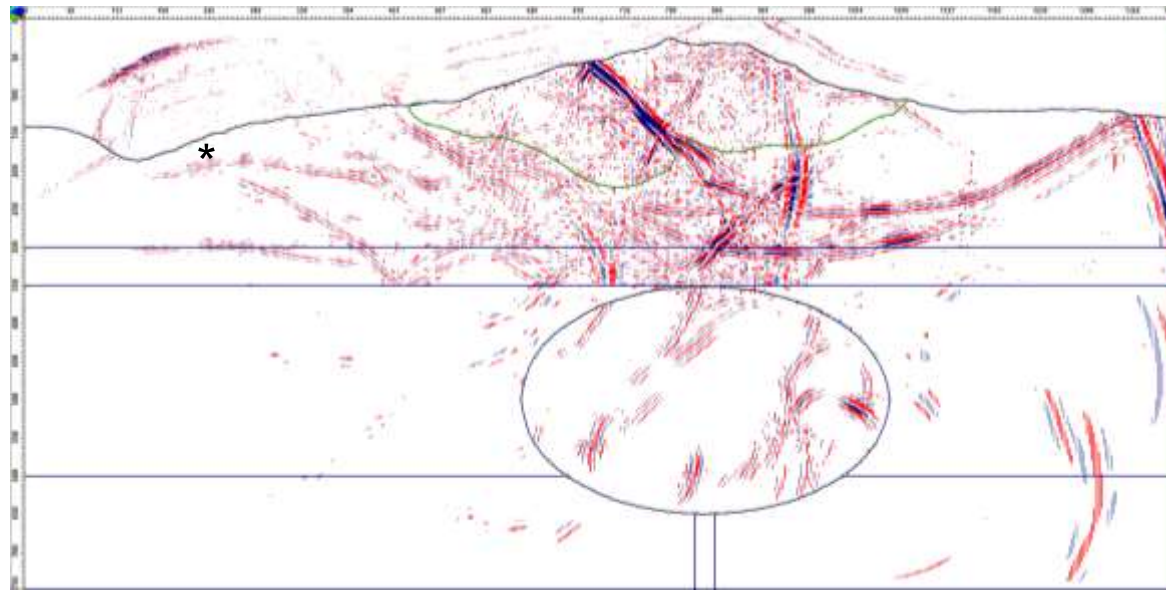
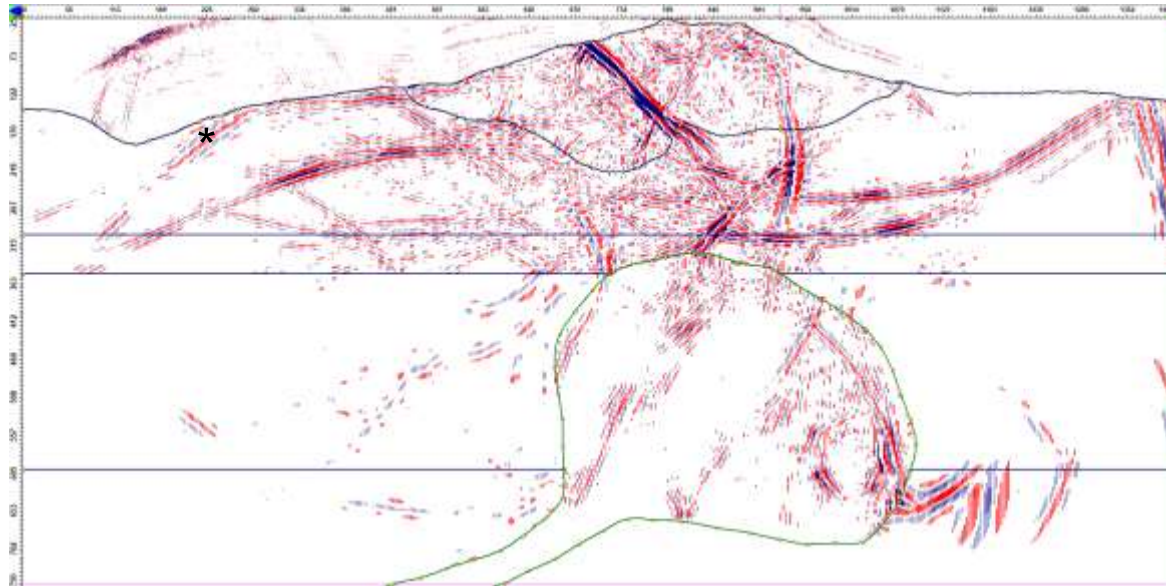
и, плоскость XZ

$T = 5,6$ с



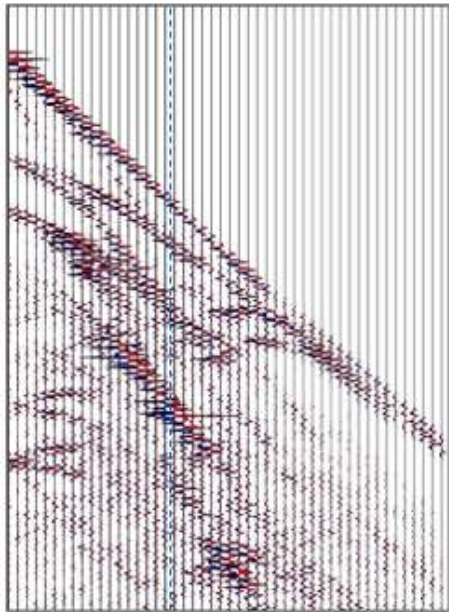
Результаты 2D моделирования

u, плоскость XZ
T = 7,2 с

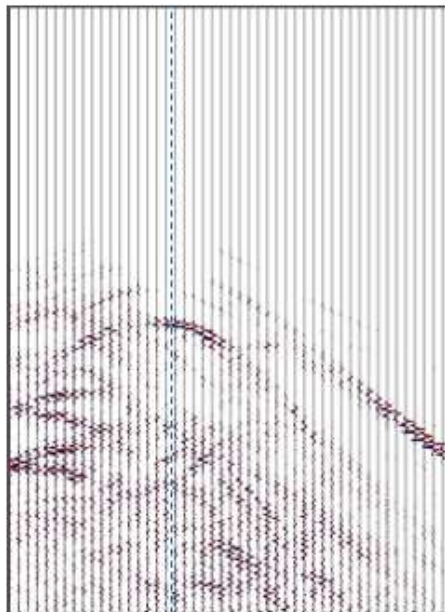


Результаты 2D моделирования: теоретические сейсмограммы

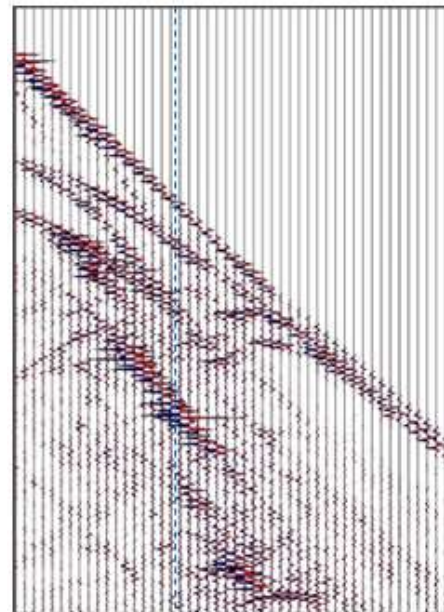
A1



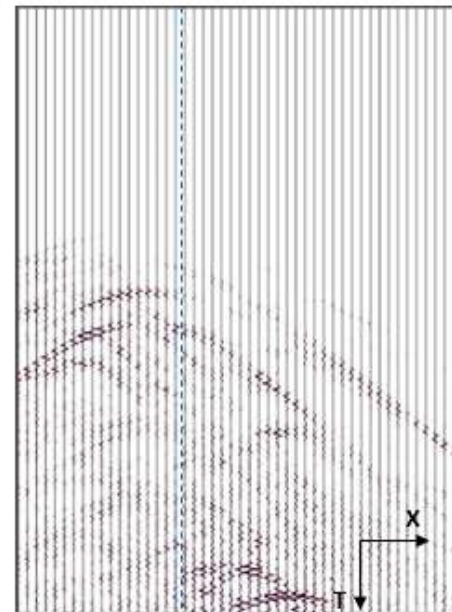
A2



B1



B2



Теоретические сейсмограммы компоненты u волнового поля по заглубленному горизонтальному профилю, начиная с 6 км с шагом 500 м. А – уточненная форма камеры, В – схематичная форма камеры. 1 – исходные трассы, 2 – разница с моделью без магматических включений. Пунктирной линией обозначена середина вулканической постройки.

Защищаемые научные результаты

1. Многоуровневый архитектурно-ориентированный метод проектирования параллельных алгоритмов и программ для численного моделирования физических процессов, сочетающий комплексный анализ ресурсов параллелизма, архитектурно-зависимые стратегии оптимизации и имитационную оценку масштабируемости параллельных алгоритмов.
2. Архитектурно-ориентированный программно-алгоритмический комплекс, реализующий предложенный метод на примере решения трёхмерной динамической задачи сеймики, и результаты сравнительного анализа эффективности его работы на широком наборе вычислительных платформ.

Научная новизна и личный вклад

1. Разработан и апробирован новый многоуровневый архитектурно-ориентированный метод проектирования параллельных алгоритмов и программ для численного моделирования физических процессов, включающий:
 - комплексный анализ ресурсов параллелизма;
 - архитектурно-зависимые стратегии оптимизации;
 - имитационную оценку масштабируемости.
2. Разработан новый архитектурно-ориентированный параллельный алгоритм конечно-разностного решения 3D динамической задачи теории упругости, реализующий:
 - иерархическую схему распараллеливания;
 - асинхронный порядок обмена данными на гибридном кластере;
 - оптимизацию размещения данных и балансировку загрузки ядер CPU/GPU.
3. Разработан и исследован программный комплекс для CPU- и GPU-кластеров:
 - выполнены численные эксперименты по оценке производительности, масштабируемости и (совместно с И.Г. Черных) энергоэффективности на широком наборе вычислительных платформ;
 - построена имитационная модель исполнения, с помощью реализации которой (совместно с Винсом Д.В.) показана высокая масштабируемость (до миллиона ядер).
4. Применение комплекса к моделированию сейсмических полей в районе Эльбрусского вулканического центра подтвердило возможность использования вибромониторинга для диагностики состояния магматических вулканов.

Заключение

- Разработанный многоуровневый архитектурно-ориентированный метод проектирования параллельных алгоритмов и программ обеспечивает системный подход к адаптации и оптимизации вычислений под целевые архитектуры.
- Разработанный на основе предложенного метода программно-алгоритмический комплекс обеспечивает ускорение и масштабирование численного моделирования сейсмических полей.
- Подтверждена применимость метода для моделирования сейсмических процессов в реальных геологических условиях (вулканические структуры).
- Ограничения разработанных метода и программно-алгоритмического комплекса связаны с
 - привязкой к архитектурам NVIDIA GPU и x86 CPU,
 - необходимостью ручной настройки параметров параллелизма (размеры блоков, границы выравнивания и пр.),
 - отсутствием возможности учёта вязкоупругих и анизотропных свойств среды.
- Дальнейшее развитие предполагает
 - адаптацию алгоритма под новые архитектуры (ARM),
 - автоматизацию настройки параметров распараллеливания,
 - расширение физической модели для повышения реалистичности расчётов с уточнением аппроксимации свободной поверхности.

Основные публикации по теме диссертации

Статьи в рецензируемых научных журналах из перечня ВАК

1. **Сапетина А.Ф.** Численное моделирование распространения сейсмических волн в сложно построенных средах на гибридном кластере / А.Ф. Сапетина // **Проблемы прочности и пластичности.** — 2014. — Т. 76, № 4. — С. 288–296.
2. The Hybrid-Cluster Multilevel Approach to Solving the Elastic Wave Propagation Problem / B.M. Glinskiy, **A.F. Sapetina** [et al.] // **Communications in Computer and Information Science.** — 2017. — Vol. 753. — P. 261–274.
3. The Integrated Approach to Solving Large-Size Physical Problems on Supercomputers / B.M. Glinskiy, ... **A.F. Sapetina** [et al.] // **Communications in Computer and Information Science.** — 2017. — Vol. 793. — P. 278–289.
4. **Sapetina A.F.** Numerical modeling results for vibroseismic monitoring of volcanic structures with different shape of the magma chamber / A.F. Sapetina, B.M. Glinskiy, V.N. Martynov // **Journal of Physics: Conference Series.** — 2021. — Vol. 1715. — Article Number 012057.
5. Glinskiy B.M. The Experimental Study and Simulation of Volcanic Structures Using Active Vibroseismic Methods / B.M. Glinskiy, ... **A.F. Sapetina** [et al.] // **Journal of Volcanology and Seismology.** — 2022. — Vol.16. — P. 280–298.
6. **Sapetina A.** The Efficiency Optimization Study of a Geophysical Code on Manycore Computing Architectures / A. Sapetina, B. Glinskiy // **Lecture Notes in Computer Science.** — 2023. — Vol. 14388. — P. 285–299.

Зарегистрированные программы

Свидетельство о государственной регистрации программы ЭВМ, зарегистрированной в Роспатенте № 2023610467. Программный комплекс для численного решения уравнений динамической теории упругости на суперЭВМ с возможностью использования ускорителей вычислений Intel Xeon Phi KNL или NVIDIA GPU / **Сапетина А.Ф.** — заявл. 29.12.2022; опубл. 11.01.2023.

Апробация работы

18 научных международных научных конференций в России и за рубежом

- «Advanced mathematics, computations and applications», Новосибирск, 2014;
- «Современные проблемы прикладной математики и информатики - 2014», Новосибирск, 2014;
- «Параллельные вычислительные технологии», Екатеринбург, 2015; Казань, 2017;
- «Дифференциальные уравнения и математическое моделирование», Улан-Удэ, 2015; Ханты-Мансийск, 2019;
- «Суперкомпьютерные дни в России», Москва, 2016, 2017, 2023 – I место за лучший научный доклад;
- «Теория и численные методы решения обратных и некорректных задач», Новосибирск, 2016;
- «Марчуковские научные чтения», Новосибирск, 2017, 2020, 2021 – диплом лауреата конкурса молодых ученых, 2024;
- «Интерэкспо ГЕО-Сибирь», Новосибирск, 2018;
- «Вычислительная математика и математическая геофизика», Новосибирск, 2018;
- «Numerical Computations: Theory and Algorithms», Калабрия, Италия, 2019;
- «Проблемы комплексного геофизического мониторинга сейсмоактивных регионов», Петропавловск-Камчатский, 2021;

Соответствие паспорту специальности

1.2.2 – математическое моделирование, численные методы и комплексы программ

Наименование отрасли науки, по которой присуждаются ученые степени:

Физико-математические / Технические

Направления исследований:

1. Разработка новых математических методов моделирования объектов и явлений

(физико-математические науки).

2. Разработка, обоснование и тестирование эффективных вычислительных методов с применением современных компьютерных технологий.

3. Реализация эффективных численных методов и алгоритмов в виде комплексов проблемно-ориентированных программ для проведения вычислительного эксперимента.

4. Разработка новых математических методов и алгоритмов интерпретации натурального эксперимента на основе его математической модели.

5. Разработка новых математических методов и алгоритмов валидации математических моделей объектов на основе данных натурального эксперимента или на основе анализа математических моделей.

6. Разработка систем компьютерного и имитационного моделирования, алгоритмов и методов имитационного моделирования на основе анализа математических моделей (технические науки).

7. Качественные или аналитические методы исследования математических моделей (технические науки).

8. Комплексные исследования научных и технических проблем с применением современной технологии математического моделирования и вычислительного эксперимента.

9. Постановка и проведение численных экспериментов, статистический анализ их результатов, в том числе с применением современных компьютерных технологий (технические науки).

*** Диссертационное исследование должно содержать все три составляющих названия специальности**

Спасибо за внимание!

Характеристики процессоров и ускорителей

Процессор	Архитектура	Кол-во ядер × SMT	L2 кэш (на ядро), КБ	L3 кэш, МБ	Длина векторного регистра, бит	Базовая тактовая частота, ГГц
Intel Xeon Phi 7290	KNL	72 × 4	512	0	512	1.5
2 × Intel Xeon E5-2697A v4	Broadwell	32 × 2	256	40	256	2.6
2 × IBM POWER9 Processor	POWER9	24 × 8	512	120	128	3.8

GPU NVIDIA	Архитектура	Кол-во ядер	Глобальная память, ГБ	L2 кэш, КБ	Разделяемая память, КБ	Пропускная способность памяти, Гбит/с
Tesla M2090	Fermi	512	6	768	16 / 48	177
Tesla K40	Kepler	2880	12	1536	16 / 48	288
Tesla P100	Pascal	3584	16	4096	64	733