

ТЕХНОЛОГИЯ МОДЕЛИРОВАНИЯ УПРУГИХ ВОЛН В
СРЕДАХ СО СЛОЖНОЙ 3D ГЕОМЕТРИЕЙ
ПОВЕРХНОСТИ НА ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ
КЛАСТЕРАХ.
КРАТКИЙ ОБЗОР DVM-СИСТЕМЫ ДЛЯ РАЗРАБОТКИ
ПАРАЛЛЕЛЬНЫХ ПРОГРАММ

Титов П.А., ИВМиМГ СО РАН

Новосибирск 2016

Введение

Моделирование 3D упругих волн в средах различного строения является важным аспектом создания геофизических 3D моделей, изучения особенностей волновых полей .

Зачастую решить обратную задачу геофизики (восстановление строения и параметров среды по экспериментально полученным записям сигналов) сложно, и одним из методов является решение набора прямых задач (моделирование сейсмополей в среде с заданными параметрами и строением) с варьированием значений параметров и геометрии среды при сравнении реальных данных с результатами моделирования.

Наиболее используемый метод для решения прямой задачи - метод конечных разностей (например, [1, R. Graves; 2, J. Virieaux; 3, D. Apello]).

Исследуемая область может обладать сложной 3D геометрией поверхности, поэтому важным отличительным моментом работы является построение криволинейной 3D сетки.

Учитывая масштабы области при решении реальных задач (десятки километров в каждом координатном направлении), задача численного моделирования становится невыполнима на персональной рабочей станции.

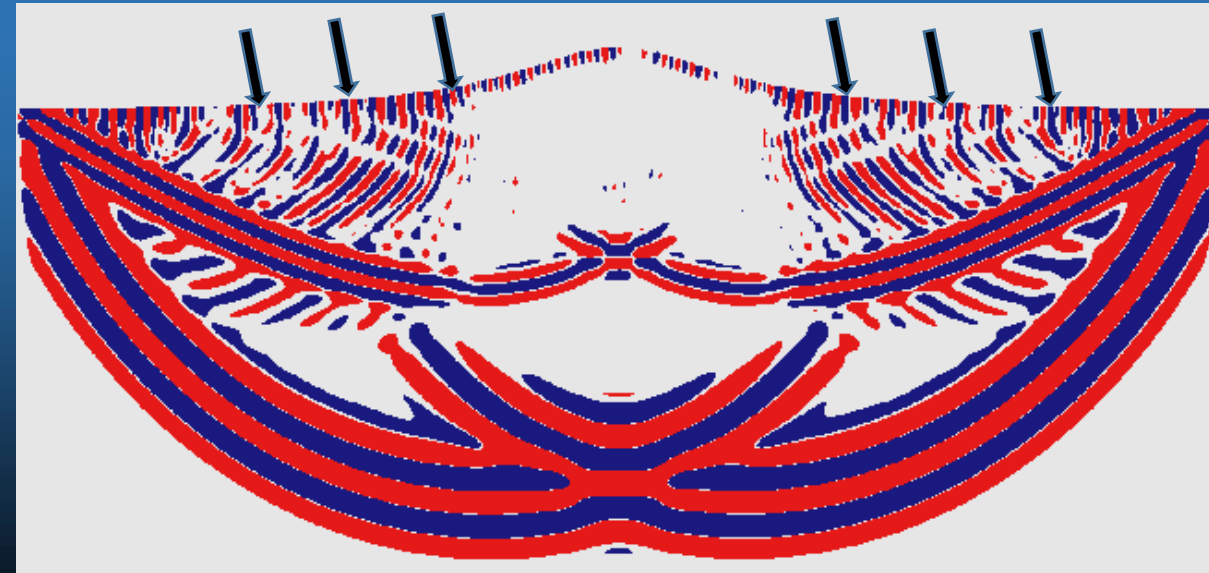
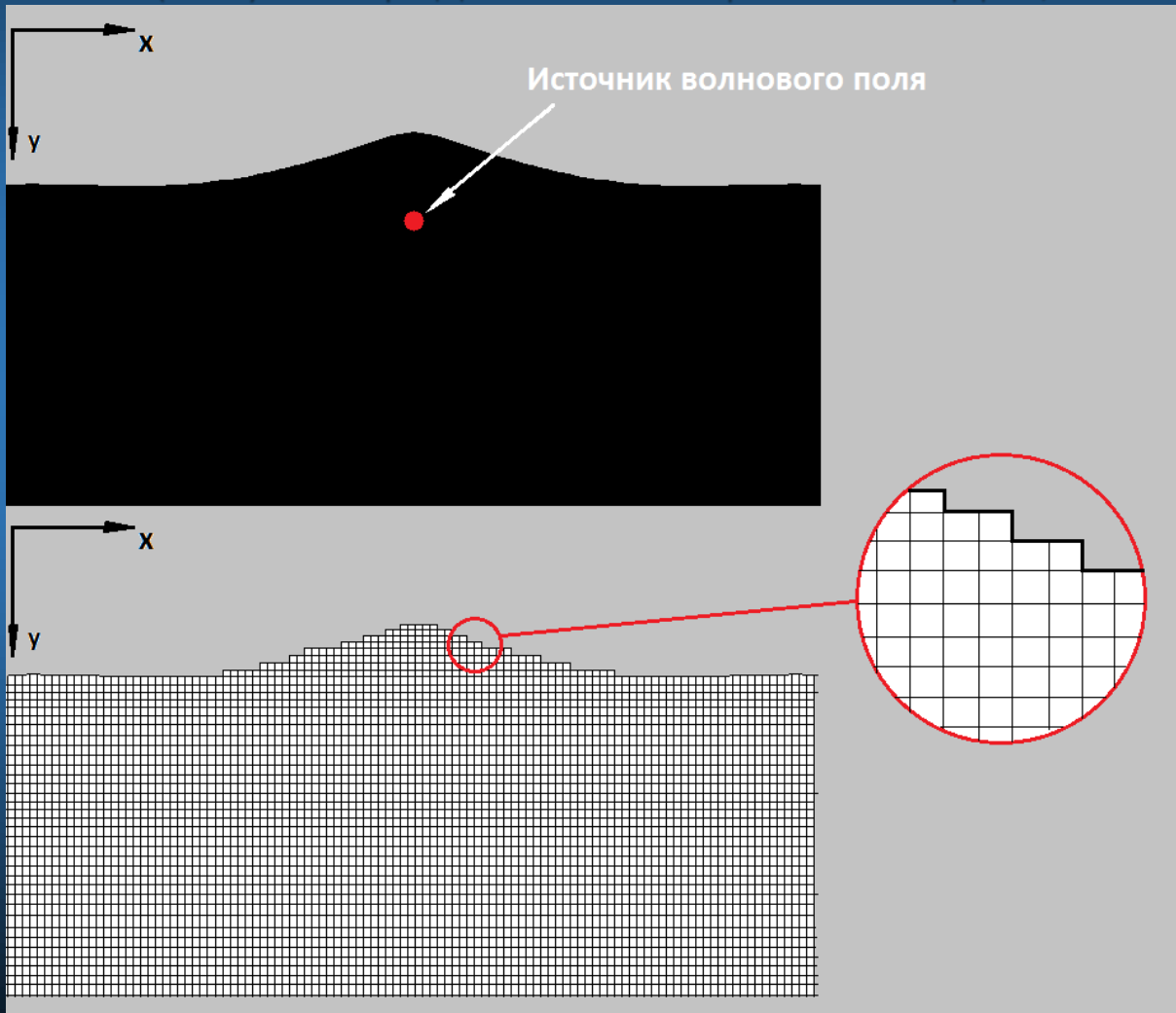
Поэтому был разработан 3D параллельный алгоритм и его программная реализация для проведения расчетов на высокопроизводительных кластерах.

Расчеты для данной работы проводились на кластере НКС-30Т+GPU (Сибирский Суперкомпьютерный Центр, www2.sccc.ru).

Также активно применяется и реализуется на практике идея со-дизайна.

При моделировании упругих волн часто используется разбиение области на прямоугольные ячейки([1-4]), что приводит к нефизичным эффектам вблизи свободной поверхности.

(Рисунок предоставлен Караваевым Д.А., ИВМиМГ СО РАН)

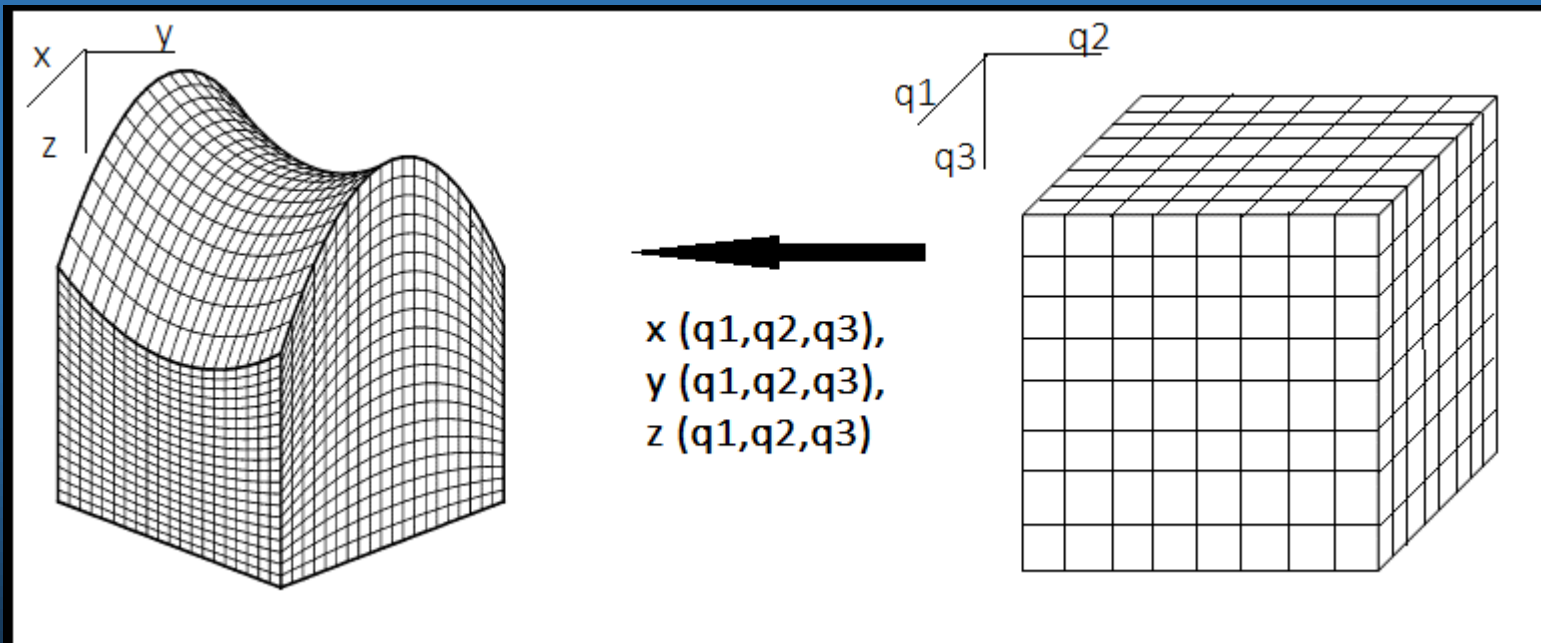


Построение криволинейной 3D сетки Метод отображений

Пример для 3D области:

Физическая область

Расчетная область



Криволинейная сетка
получается как
взаимно-однозначное
отображение равномерной
сетки прямоугольной
области
(метод отображений,
[4, стр. 12, В. Д. Лисейкин]).

Таким образом
задачу в исходной области
сложной формы можно
«перенести» на область
простой формы, сделав
замену переменных.

В данной работе применяется следующий подход к построению сетки: она строится методом трансфинитной интерполяции ([4, стр. 53, В. Д. Лисейкин]) и является локально-ортогональной возле свободной поверхности. Это позволяет увеличить точность реализации граничных условий и проводить моделирование для областей со значительной кривизной поверхности.

Постановка задачи и ее преобразование

В декартовой системе координат в переменных (x, y, z) уравнения имеют вид:

$$\begin{aligned}\rho \frac{\partial^2 u}{\partial t^2} &= \frac{\partial}{\partial x} \left((\lambda + 2\mu) \frac{\partial u}{\partial x} + \lambda \frac{\partial v}{\partial y} + \lambda \frac{\partial w}{\partial z} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial u}{\partial y} + \mu \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial u}{\partial z} + \mu \frac{\partial w}{\partial x} \right) + F_x \\ \rho \frac{\partial^2 v}{\partial t^2} &= \frac{\partial}{\partial x} \left(\mu \frac{\partial u}{\partial y} + \mu \frac{\partial v}{\partial x} \right) + \frac{\partial}{\partial y} \left(\lambda \frac{\partial u}{\partial x} + (\lambda + 2\mu) \frac{\partial v}{\partial y} + \lambda \frac{\partial w}{\partial z} \right) + \frac{\partial}{\partial z} \left(\mu \frac{\partial v}{\partial z} + \mu \frac{\partial w}{\partial y} \right) + F_y \\ \rho \frac{\partial^2 w}{\partial t^2} &= \frac{\partial}{\partial x} \left(\mu \frac{\partial u}{\partial z} + \mu \frac{\partial w}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mu \frac{\partial v}{\partial z} + \mu \frac{\partial w}{\partial y} \right) + \frac{\partial}{\partial z} \left(\lambda \frac{\partial u}{\partial x} + \lambda \frac{\partial v}{\partial y} + (\lambda + 2\mu) \frac{\partial w}{\partial z} \right) + F_z\end{aligned}$$

где F_x, F_y, F_z – компоненты массовой силы.

Граничные условия и начальные нулевые условия имеют вид :

$$u|_{\partial\Gamma} = v|_{\partial\Gamma} = w|_{\partial\Gamma} = 0, \quad \frac{\partial u}{\partial t} = \frac{\partial v}{\partial t} = \frac{\partial w}{\partial t} = 0$$

Условия на свободной поверхности ∂S :

$$\begin{aligned}n_x \left((\lambda + 2\mu) \frac{\partial u}{\partial x} + \lambda \frac{\partial v}{\partial y} + \lambda \frac{\partial w}{\partial z} \right) + n_y \left(\mu \frac{\partial u}{\partial y} + \mu \frac{\partial v}{\partial x} \right) + n_z \left(\mu \frac{\partial u}{\partial z} + \mu \frac{\partial w}{\partial x} \right) &= 0 \\ n_x \left(\mu \frac{\partial u}{\partial y} + \mu \frac{\partial v}{\partial x} \right) + n_y \left(\lambda \frac{\partial u}{\partial x} + (\lambda + 2\mu) \frac{\partial v}{\partial y} + \lambda \frac{\partial w}{\partial z} \right) + n_z \left(\mu \frac{\partial v}{\partial z} + \mu \frac{\partial w}{\partial y} \right) &= 0 \\ n_x \left(\mu \frac{\partial v}{\partial z} + \mu \frac{\partial w}{\partial y} \right) + n_y \left(\mu \frac{\partial v}{\partial z} + \mu \frac{\partial w}{\partial y} \right) + n_z \left(\lambda \frac{\partial u}{\partial x} + \lambda \frac{\partial v}{\partial y} + (\lambda + 2\mu) \frac{\partial w}{\partial z} \right) &= 0\end{aligned}$$

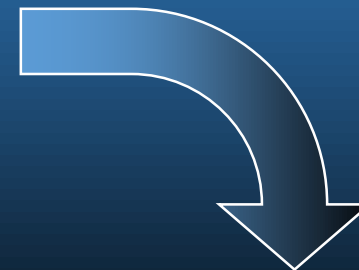
Задача является динамической. Ее решение связано с решением системы уравнений теории упругости в 3D случае, записанной в смещениях (U, V, W) .

В данной работе рассматриваются упругие волны для случая изотропной среды.

Свойства среды определяются тремя параметрами:

коэффициентами Ламэ λ, μ и плотностью ρ .

$$x = x(q_1, q_2, q_3), y = y(q_1, q_2, q_3), z = z(q_1, q_2, q_3)$$



$$\rho \frac{\partial^2 u}{\partial t^2} = \sum_{j=1}^3 \frac{\partial q^j}{\partial x} \frac{\partial}{\partial q^j} \left((\lambda + 2\mu) \sum_{i=1}^3 \frac{\partial q^i}{\partial x} \frac{\partial u}{\partial q^i} + \lambda \sum_{i=1}^3 \frac{\partial q^i}{\partial y} \frac{\partial v}{\partial q^i} + \lambda \sum_{i=1}^3 \frac{\partial q^i}{\partial z} \frac{\partial w}{\partial q^i} \right) +$$

$$\sum_{j=1}^3 \frac{\partial q^j}{\partial y} \frac{\partial}{\partial q^j} \left(\mu \sum_{i=1}^3 \frac{\partial q^i}{\partial y} \frac{\partial u}{\partial q^i} + \mu \sum_{i=1}^3 \frac{\partial q^i}{\partial x} \frac{\partial v}{\partial q^i} \right) +$$

$$\sum_{j=1}^3 \frac{\partial q^j}{\partial z} \frac{\partial}{\partial q^j} \left(\mu \sum_{i=1}^3 \frac{\partial q^i}{\partial z} \frac{\partial u}{\partial q^i} + \mu \sum_{i=1}^3 \frac{\partial q^i}{\partial x} \frac{\partial w}{\partial q^i} \right) + F_x$$

$$\rho \frac{\partial^2 v}{\partial t^2} = \sum_{j=1}^3 \frac{\partial q^j}{\partial x} \frac{\partial}{\partial q^j} \left(\mu \sum_{i=1}^3 \frac{\partial q^i}{\partial y} \frac{\partial u}{\partial q^i} + \mu \sum_{i=1}^3 \frac{\partial q^i}{\partial x} \frac{\partial v}{\partial q^i} \right) +$$

$$\sum_{j=1}^3 \frac{\partial q^j}{\partial y} \frac{\partial}{\partial q^j} \left(\lambda \sum_{i=1}^3 \frac{\partial q^i}{\partial x} \frac{\partial u}{\partial q^i} + (\lambda + 2\mu) \sum_{i=1}^3 \frac{\partial q^i}{\partial y} \frac{\partial v}{\partial q^i} + \lambda \sum_{i=1}^3 \frac{\partial q^i}{\partial z} \frac{\partial w}{\partial q^i} \right) +$$

$$\sum_{j=1}^3 \frac{\partial q^j}{\partial z} \frac{\partial}{\partial q^j} \left(\mu \sum_{i=1}^3 \frac{\partial q^i}{\partial z} \frac{\partial v}{\partial q^i} + \mu \sum_{i=1}^3 \frac{\partial q^i}{\partial y} \frac{\partial w}{\partial q^i} \right) + F_y$$

$$\rho \frac{\partial^2 w}{\partial t^2} = \sum_{j=1}^3 \frac{\partial q^j}{\partial x} \frac{\partial}{\partial q^j} \left(\mu \sum_{i=1}^3 \frac{\partial q^i}{\partial z} \frac{\partial u}{\partial q^i} + \mu \sum_{i=1}^3 \frac{\partial q^i}{\partial x} \frac{\partial w}{\partial q^i} \right) +$$

$$\sum_{j=1}^3 \frac{\partial q^j}{\partial y} \frac{\partial}{\partial q^j} \left(\mu \sum_{i=1}^3 \frac{\partial q^i}{\partial z} \frac{\partial v}{\partial q^i} + \mu \sum_{i=1}^3 \frac{\partial q^i}{\partial y} \frac{\partial w}{\partial q^i} \right) +$$

$$\sum_{j=1}^3 \frac{\partial q^j}{\partial y} \frac{\partial}{\partial q^j} \left(\lambda \sum_{i=1}^3 \frac{\partial q^i}{\partial x} \frac{\partial u}{\partial q^i} + \lambda \sum_{i=1}^3 \frac{\partial q^i}{\partial y} \frac{\partial v}{\partial q^i} + (\lambda + 2\mu) \sum_{i=1}^3 \frac{\partial q^i}{\partial z} \frac{\partial w}{\partial q^i} \right) + F_z$$



$x = x(q_1, q_2, q_3), y = y(q_1, q_2, q_3), z = z(q_1, q_2, q_3)$

Основные уравнения

Разностная схема для численного решения

Для уравнений строится разностный аналог. За основу берутся формулы из работы [3, D. Arpello], адаптированные для 3D-случая. Фактически, все дифференциальные операторы заменяются на разностные аналоги (центральные разности).

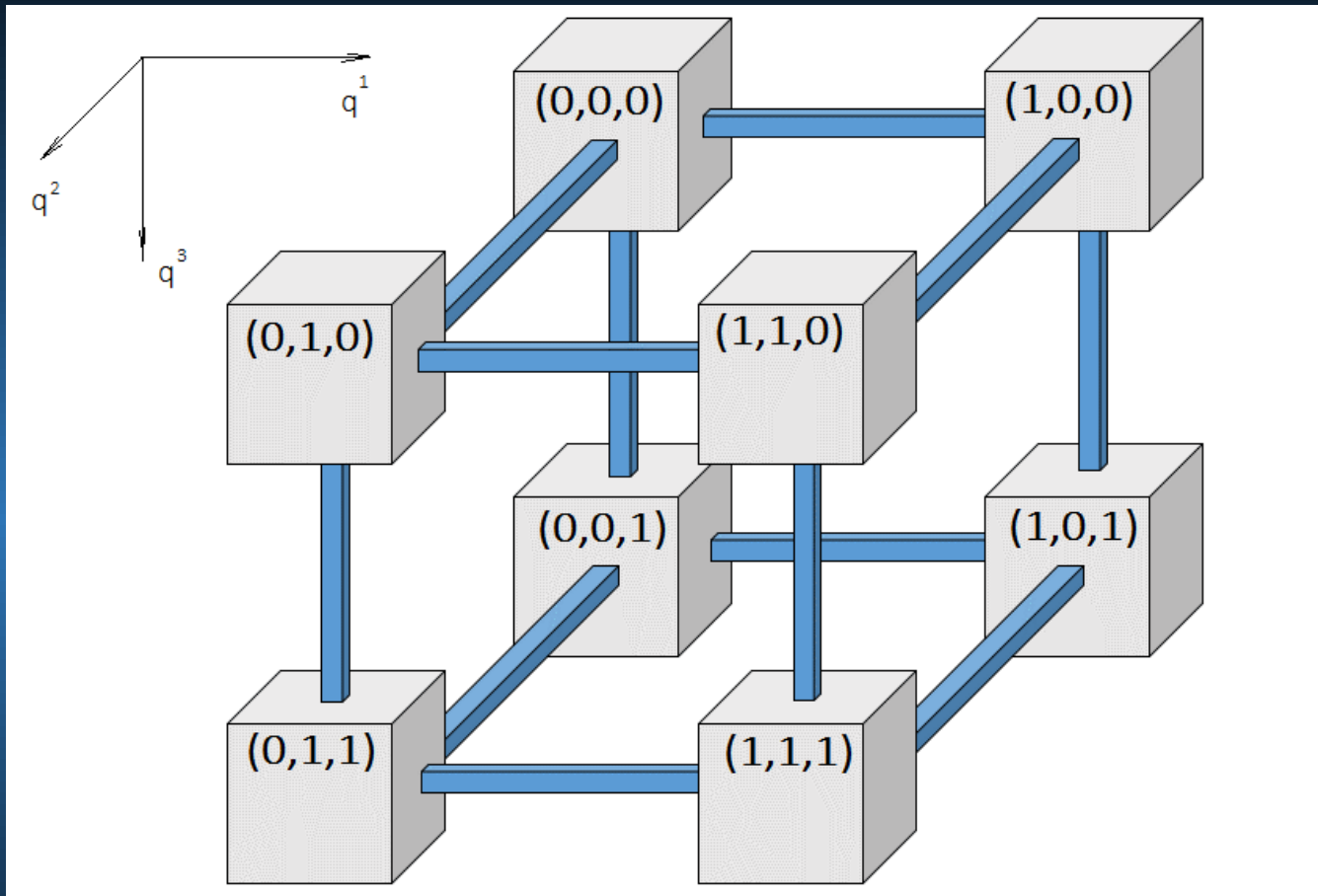
Условие устойчивости схемы получено автором данной работы, аналогично работе [2, J. Virieaux].

Схема обеспечивает второй порядок аппроксимации по времени и по пространству.

$$V_p \Delta t \leq \frac{\min |J| h}{\sqrt{3}}$$

- Условие устойчивости разностной схемы, где V_p – скорость продольных волн, J – значение якобиана преобразования расчетной области на физическую.

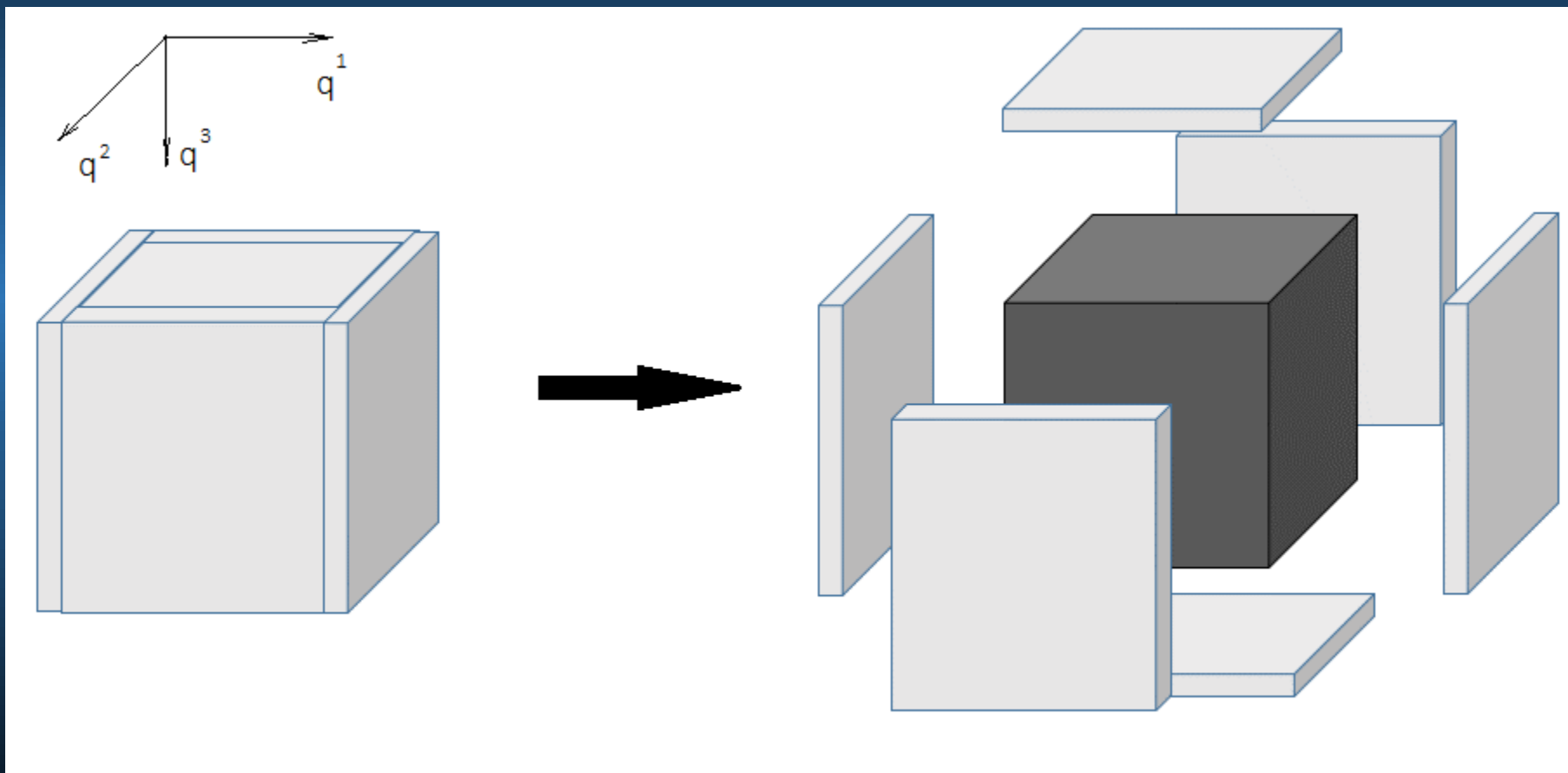
Параллельная реализация



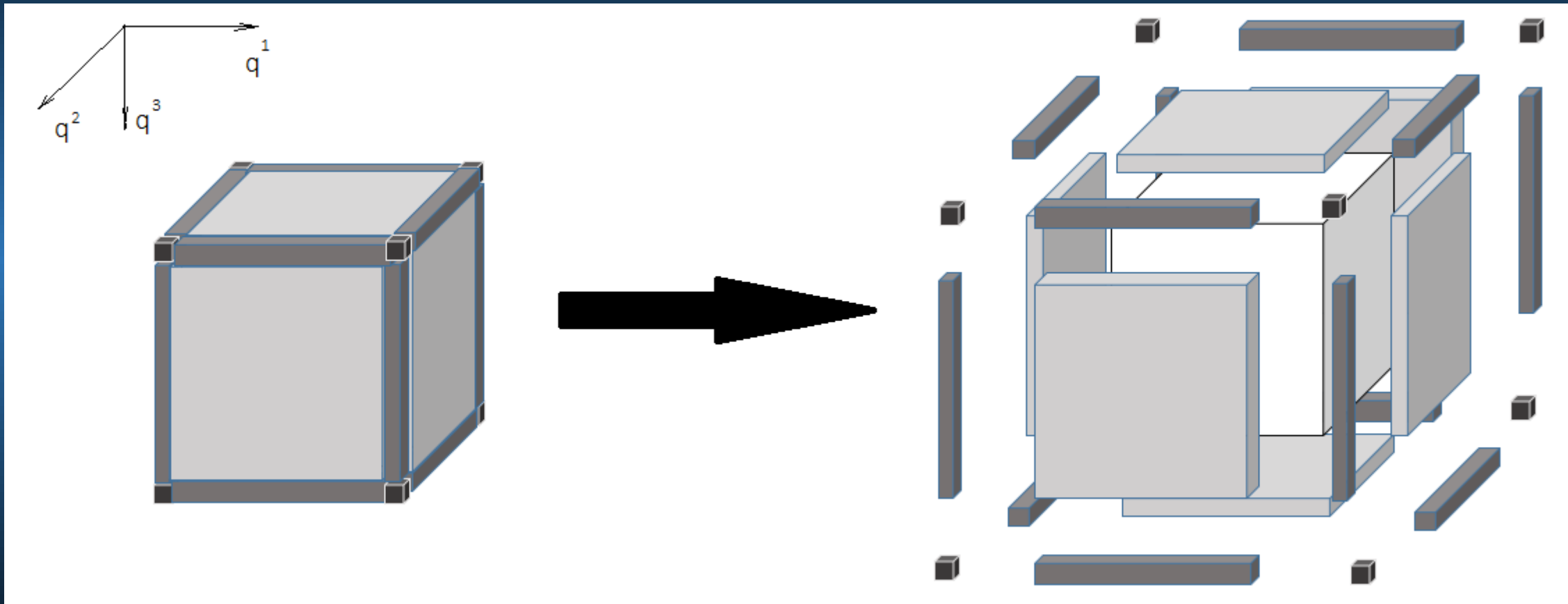
Для проведения расчетов была создана параллельная версия программы при помощи языка Fortran и пакета MPI.

Обмен данными между соседними процессами осуществляется через заданную 3D-топологию куб посредством неблокирующих пересылок `MPI_Isend`, `MPI_Irecv`.

Для ускорения работы программы вычисления внутри каждого отдельного куба разбиты на 2 части: граница куба и внутренняя часть куба (на рисунке – серым цветом).



Обмены между соседними процессами осуществляются через 24 разнонаправленных пересылки



Схематично работа программы для каждого блока выглядит следующим образом:

0) Построение криволинейной сетки и вычисление всех постоянных коэффициентов для задачи (обмены между блоками отсутствуют);



1) Вычисления на гранях каждого блока;

2) Запуск неблокирующих пересылок `MPI_Isend`, `MPI_Irecv`;

3) Вычисления внутри блока (пока идут пересылки);

4) Проверка завершения пересылок `MPI_Waitall`;

5) Переход к **1**).

Вычислительные ресурсы

НКС-30Т+GPU (сервер G6):

64 двойных блейд-сервера

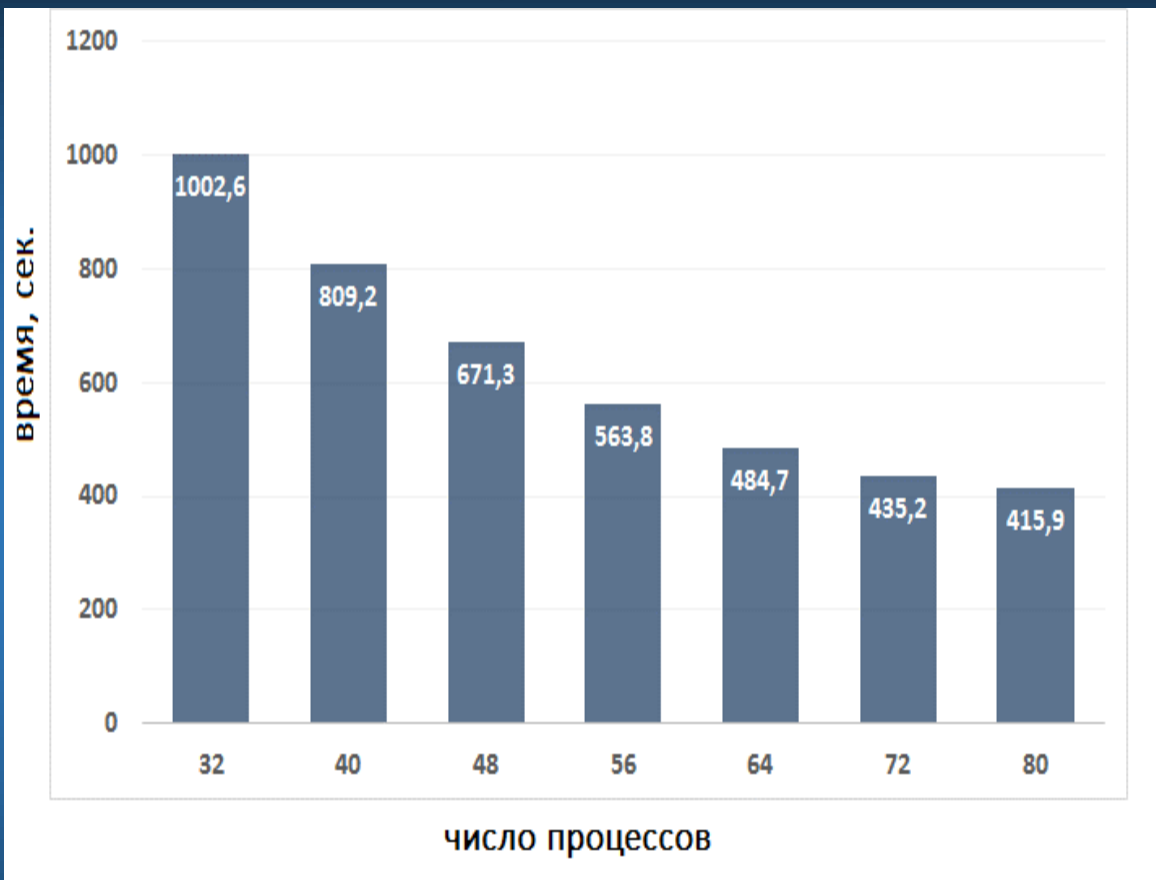
HP BL2x220 G6: 128 вычислительных
модулей, RAM модуля - 16 Гбайт,
256 (1024 ядра) процессоров Intel Xeon E5540

Производительность – 6,1 Тфлопс

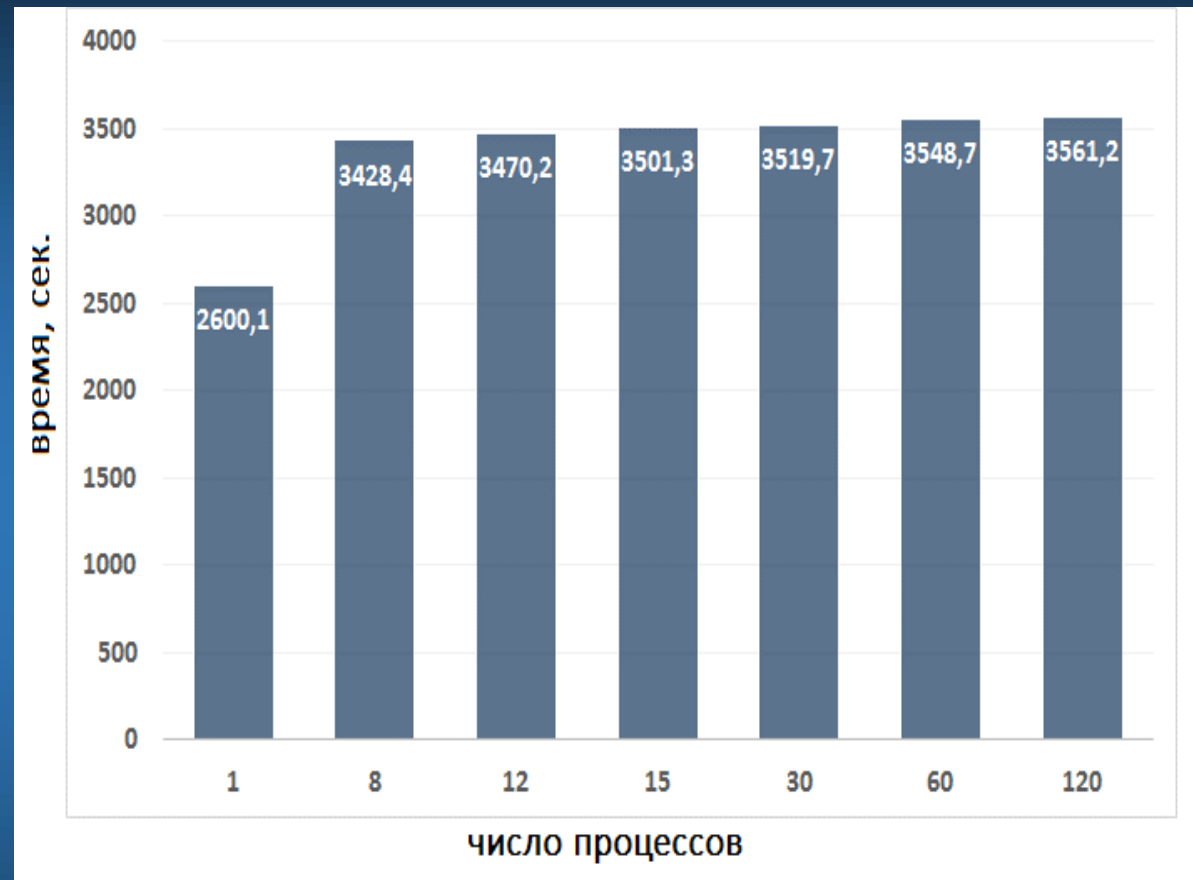
www2.sccc.ru



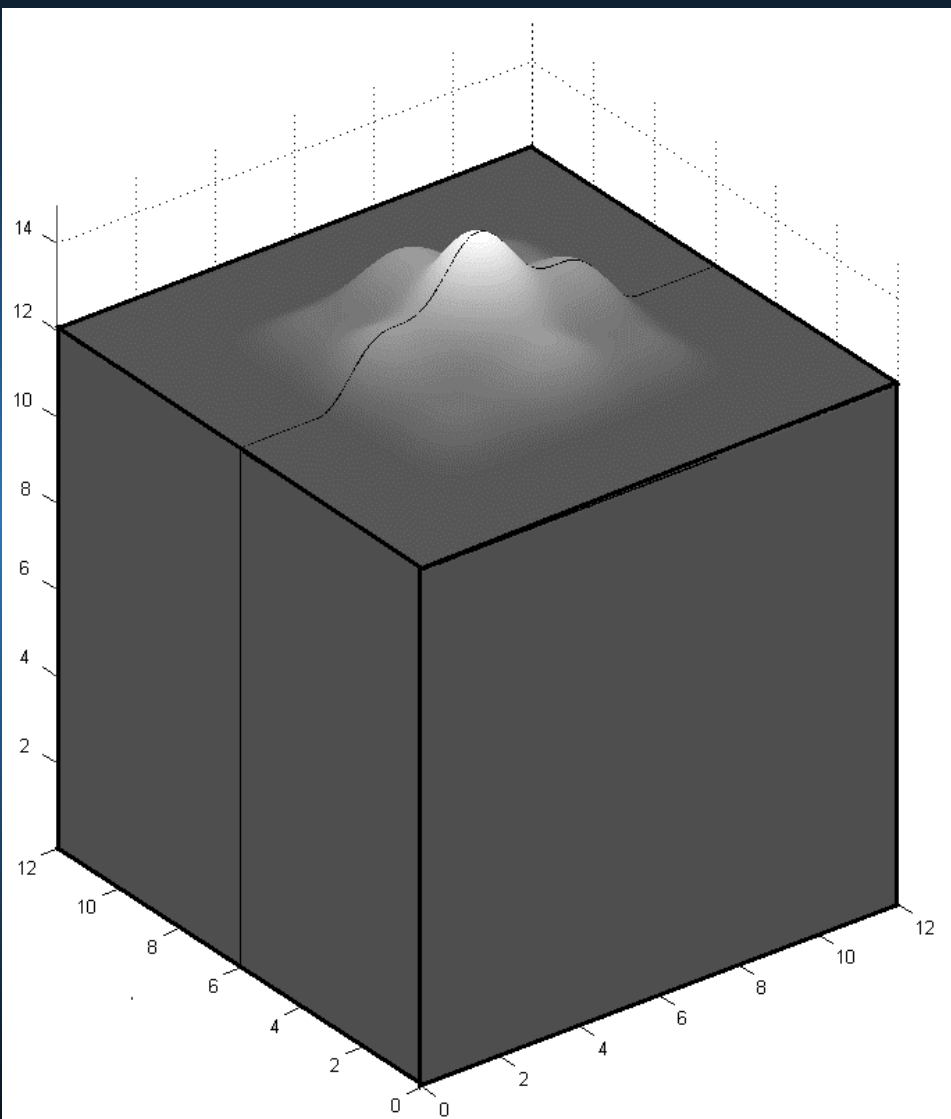
Тест на ускорение. Область 480x480x480 точек.
1000 итераций по времени



Тест на масштабируемость. Область 240x240x240 точек.
1000 итераций по времени



Тестовая модель



Модель:

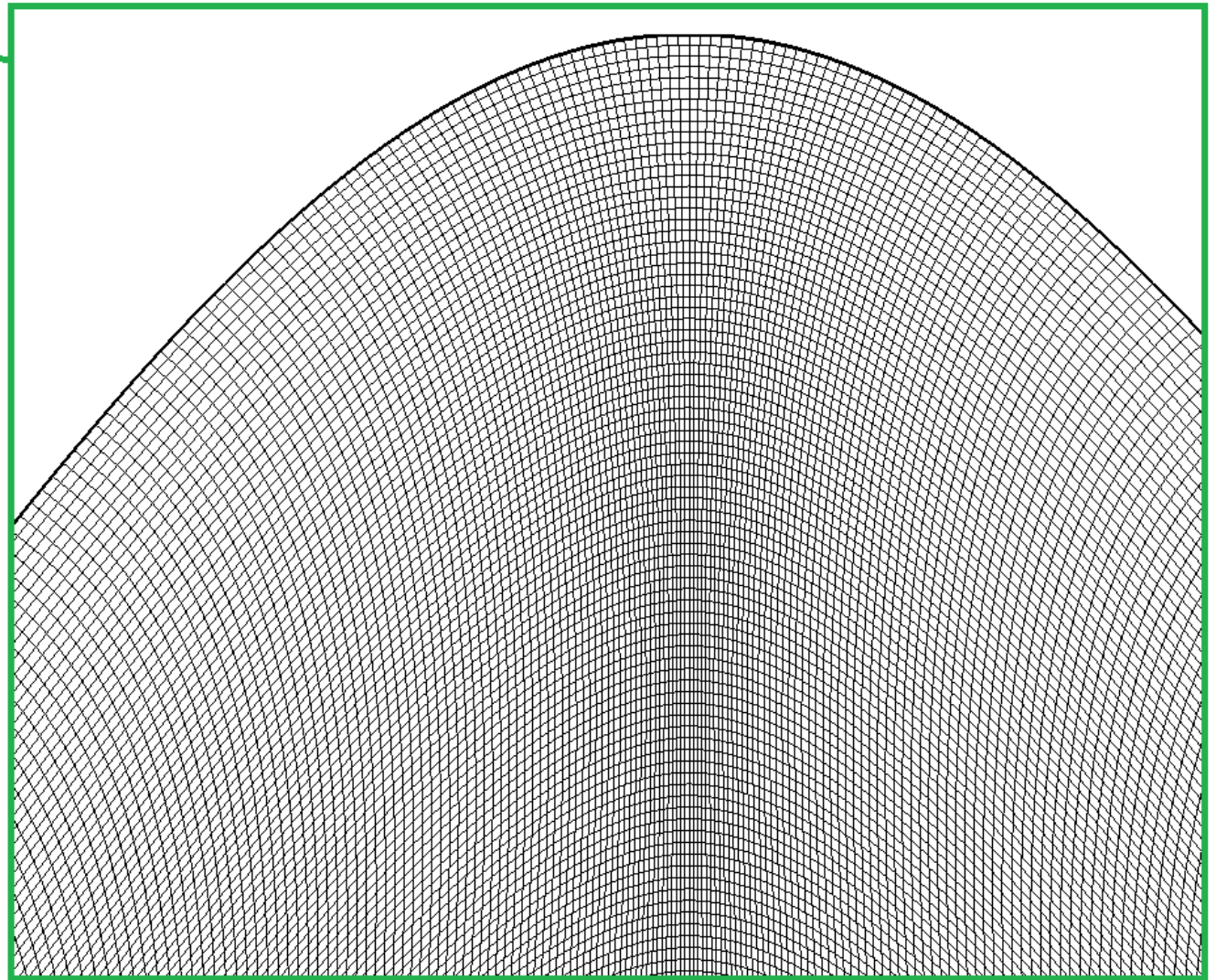
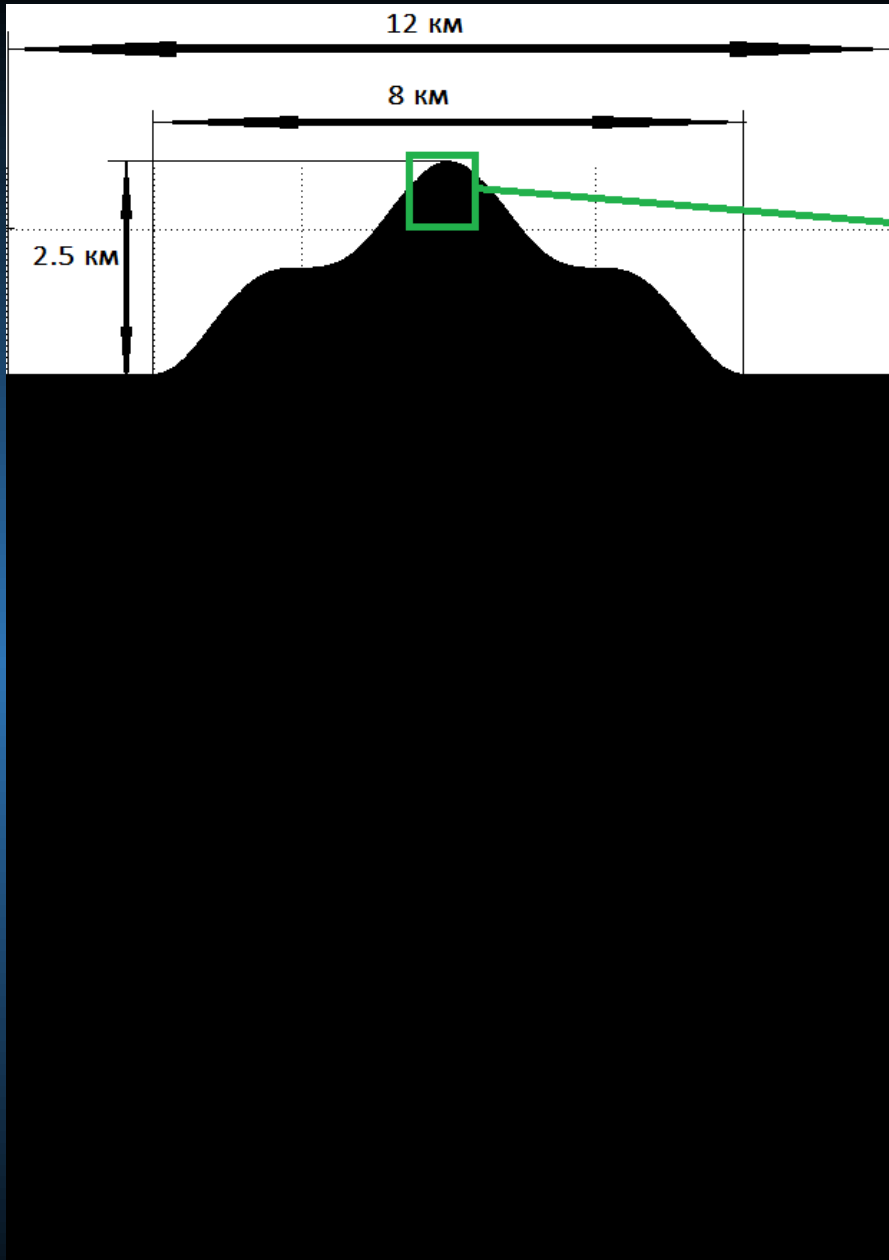
область 12x12x14,5 км.

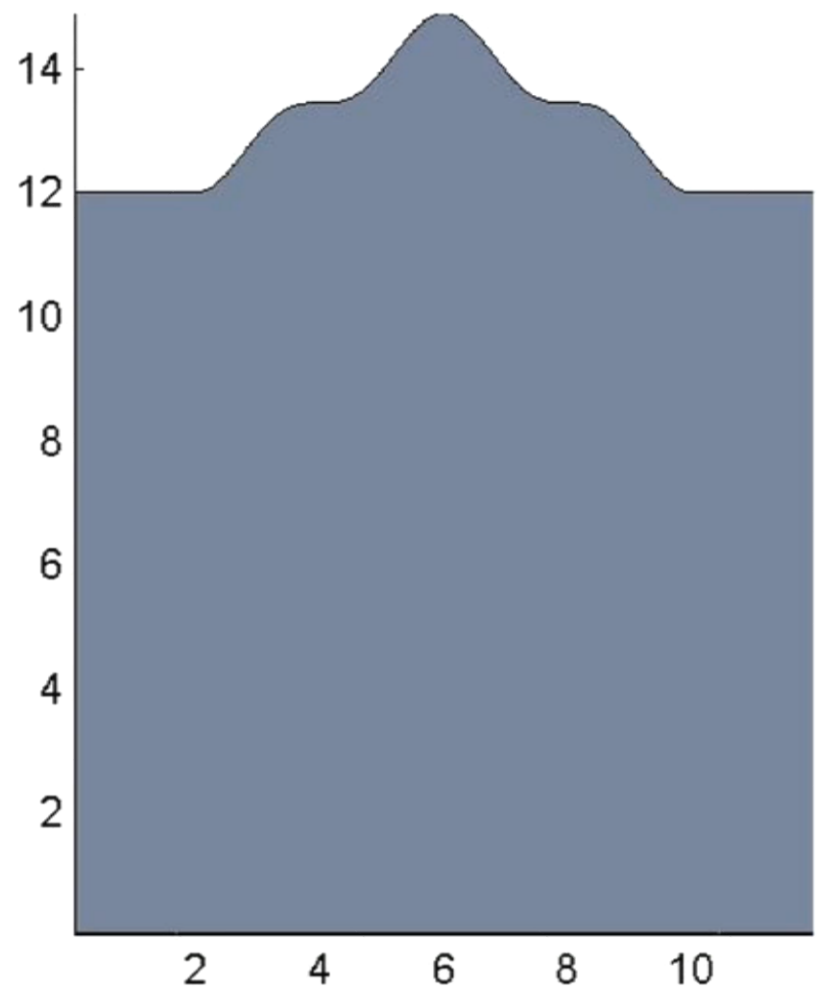
600x600x600 ячеек.

$\lambda = 0,5 \text{ кг}\cdot\text{м}/\text{с}^2$

$\mu = 0,25 \text{ кг}\cdot\text{м}/\text{с}^2$

Плотность среды – $1000 \text{ кг}/\text{м}^3$.





Результаты:

- Представлен параллельный алгоритм моделирования волновых полей в средах со сложной 3D геометрией поверхности.
- Получен критерий устойчивости разностной схемы для численного решения.
- Показан способ построения криволинейной 3D сетки.
- Проведены тестовые расчеты на многопроцессорной архитектуре, представлены результаты численного моделирования.

Литература

1. R.W. Graves Simulating seismic wave propagation in 3D elastic media using staggered grid finite differences. // Bull. Seism. soc. Am., vol.86, pp. 1091-1106, 1996
2. Virieux J. P-SV wave propagation in heterogeneous media: Velocity-stress finite-difference method. // Geophysics, Volume 51, Number 4, pp. 889–901 (1986)
3. Daniel Apello, N. Anders Petersson «A stable finite difference method for the elastic Wave equation on complex Geometries with free surfaces» // COMMUNICATIONS IN COMPUTATIONAL PHYSICS, January 2009, Vol. 5, No. 1, pp. 84-107.
4. В. Д. Лисейкин, «Разностные сетки. Теория и приложения» // Новосибирск, издательство СО РАН, 2014. – 254 с.

Работа выполнена при поддержке грантов РФФИ № 16-01-00455, 16-07-00434, а также программы фундаментальных исследований РАН №4 проект 4.9. «Модельные и экспериментальные исследования вулканических структур методами активной и пассивной сейсмологии».

DVM-СИСТЕМА

www.dvm-system.org

Система представляет собой расширения языков C и Fortran спецификациями параллелизма, оформленных в виде директив к компилятору. Это позволяет создавать эффективные параллельные программы (DVM и DVMH-программы) как для классических многоядерных, так и для гетерогенных архитектур (GPU, Phi). В систему также включены средства функциональной отладки и отладки эффективности DVM и DVMH-программ.

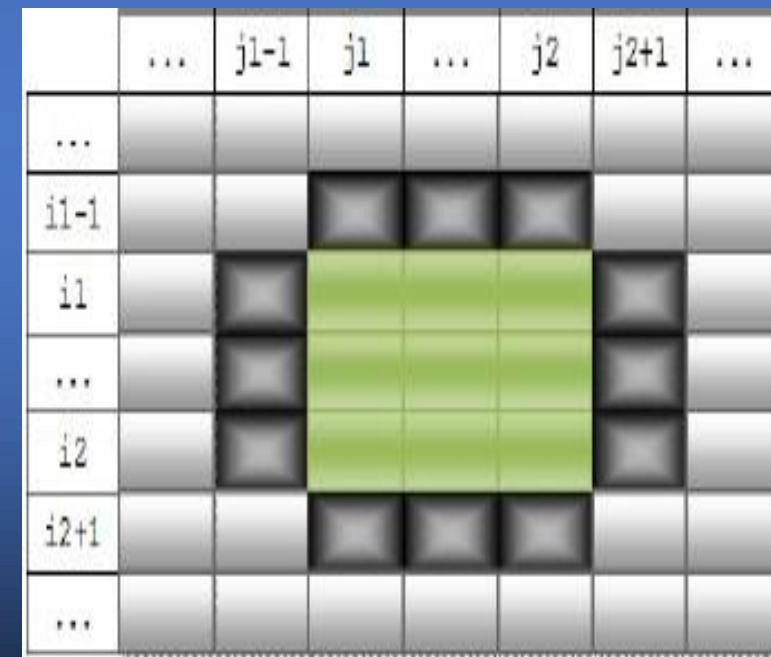
Для получения работающей DVM или DVMH-программы необходимо 2 вещи:
распределить данные и распределить вычисления над этими данными

Пример DVMH-программы

```
PROGRAM JACOBY_DVMH
PARAMETER (L=4096, ITMAX=100)
REAL A(L,L), B(L,L)
!DVM$ DISTRIBUTE (BLOCK, BLOCK) :: A
!DVM$ ALIGN B(I,J) WITH A(I,J)
PRINT *, ***** TEST_JACOBI *****
DO IT = 1, ITMAX
!DVM$ REGION INOUT(A,B)
!DVM$ PARALLEL (J,I) ON A(I, J)
DO J = 2, L-1
DO I = 2, L-1
A(I, J) = B(I, J)
ENDDO
ENDDO
!DVM$ PARALLEL (J,I) ON B(I, J), SHADOW_RENEW (A)
DO J = 2, L-1
DO I = 2, L-1
B(I, J) = (A(I-1, J) + A(I, J-1) + A(I+1, J) + A(I, J+1)) / 4
ENDDO
ENDDO
!DVM$ END REGION
!DVM$ ENDDO
!DVM$ IO_MODE(LOCAL, ASYNCHRONOUS)
OPEN(3, FILE='JAC.DAT', ACCESS='STREAM')
!DVM$ GET_ACTUAL(B)
WRITE(3, *) B
CLOSE(3)
END
```

Алгоритм Якоби в модели DVMH

Зависимость по данным для
соседних процессов



В директорию, где находится наша программа jac2d.fdv, следует скопировать dvm скрипт из директории с установленной DVM-системой.

```
$ ls  
dvm jac2d.fdv
```

Теперь можно скомпилировать нашу программу.

```
$ ./dvm f jac2d.fdv  
$ ls  
dvm jac2d jac2d.fdv
```

В текущей директории появился исполняемый файл jac2d, который можно запустить с помощью скрипта dvm, но предварительно надо настроить параметры запуска, отредактировав dvm скрипт.

```
#----- DVMH options:  
export DVMH_PPN='1' # Number of processes per node  
export DVMH_NUM_THREADS='0' # Number of CPU threads per process  
export DVMH_NUM_CUDAS='1' # Number of GPUs per process
```

Задав такие параметры, мы сообщаем, что хотим выполнять регионы только на одном графическом ускорителе. Всё, можем запускать:

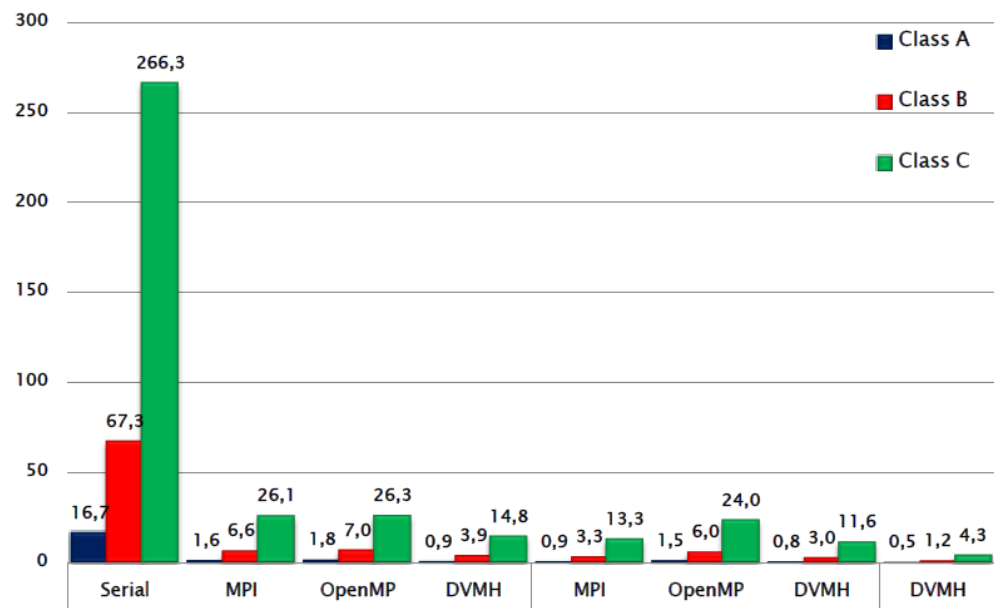
```
$ ./dvm run jac2d  
...  
IT = 197 EPS = 0.4086035E+02  
IT = 198 EPS = 0.4062500E+02  
IT = 199 EPS = 0.4044824E+02  
IT = 200 EPS = 0.4021777E+02  
time = 3.176400
```

Распараллеливание тестов NAS

- **EP** - генерация пар случайных чисел Гаусса
- **MG** - приближенное решение трехмерного уравнения Пуассона. Метод MultiGrid
- **BT** - 3D Навье-Стокс, блочная трехдиагональная схема. Метод переменных направлений
- **LU** - 3D Навье-Стокс. Метод последовательной верхней релаксации
- **SP** - 3D Навье-Стокс. Скалярная пятидиагональная схема. Beam-Warrior approximate factorization

Подробные результаты исследования эффективности доступны на сайте <http://www.dvm-system.org>

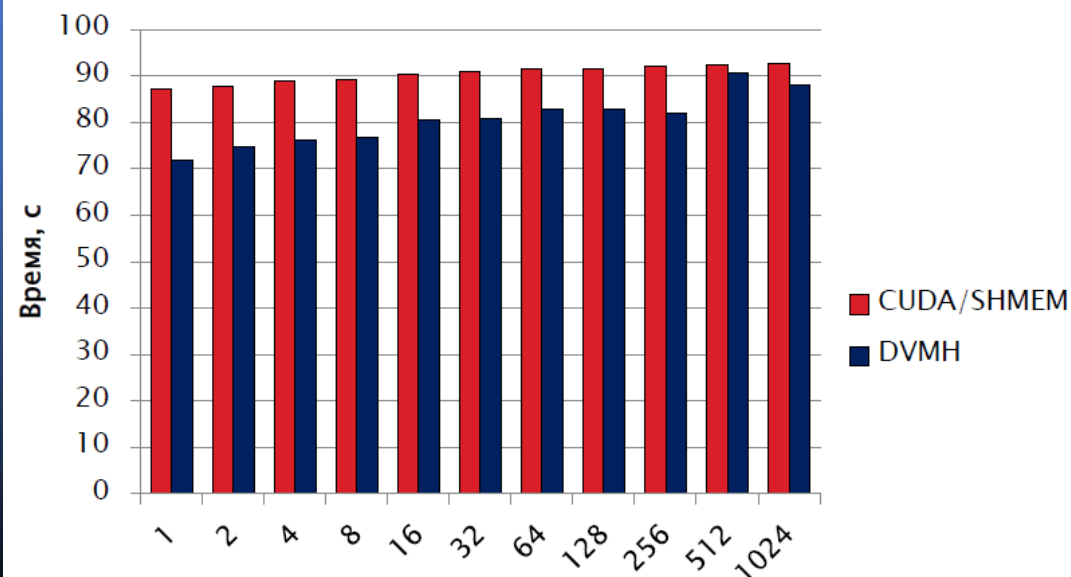
Времена выполнения теста EP, в секундах



Применение DVMH для приложений

- **Контейнер** - моделирование течения вязкой тяжелой жидкости под действием силы тяжести в прямоугольном контейнере с открытой верхней стенкой и отверстием в одной из боковых стенок в трехмерной постановке (Fortran-DVMH)
- **POLINOM** - расчет коэффициентов полиномиальной реконструкции (C-DVMH)
- **FLUXES** - расчет конвективных потоков через внутренние грани расчетных ячеек (C-DVMH)
- **Композит** – композиционная модель многокомпонентной фильтрации при разработке месторождений нефти и газа (Fortran-DVMH)

Программа Контейнер.
Сравнение DVMH vs C+SHMEM+CUDA



Состав DVM-системы

Система состоит из следующих компонент:

- Компилятор Fortran-DVMH
- Компилятор C-DVMH
- Библиотека поддержки LIB-DVMH
- DVMH-отладчик
- Анализатор производительности DVMH-программ

Возможности системы

- Распределение элементов массива между процессорами
- Распределение витков цикла между процессорами
- Спецификация параллельно выполняющихся секций программы (параллельных задач) и отображение их на процессоры
- Организация эффективного доступа к удаленным (расположенным на других процессорах/ускорителях) данным
- Организация эффективного выполнения редуцированных операций - глобальных операций с расположенными на различных процессорах/ускорителях данными (таких, как их суммирование или нахождение их максимального или минимального значения)
- Определение фрагментов программы (регионов) для возможного выполнения на ускорителях
- Управление перемещением данных между памятью ЦПУ и памятью ускорителей
- Управление параллельным вводом-выводом

Методы динамической настройки DVMH-программ

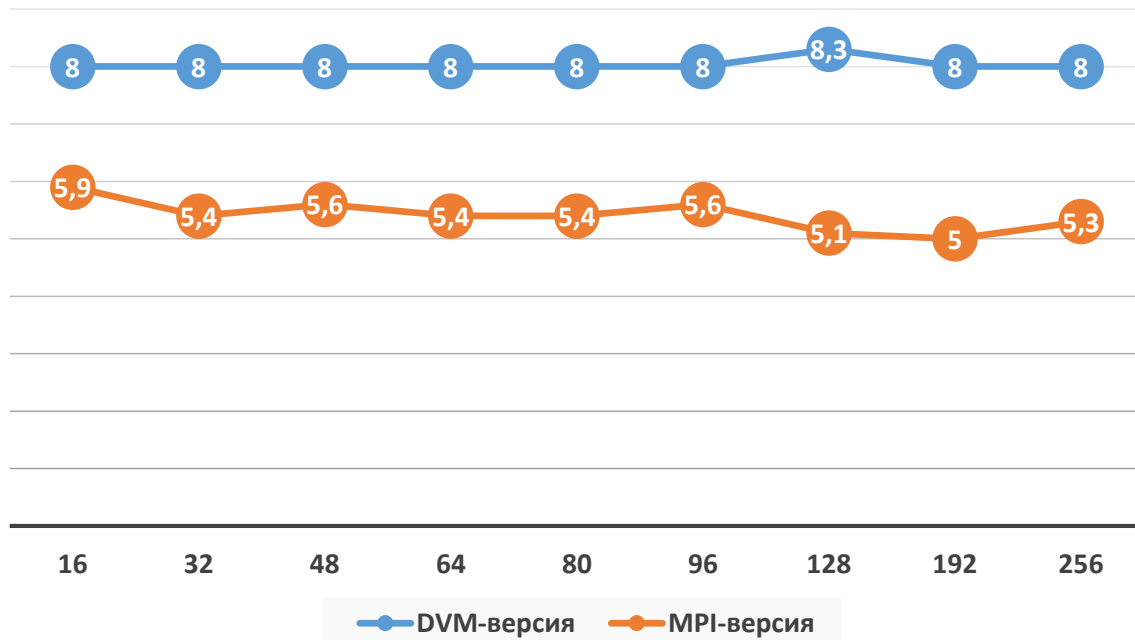
- Отображение массивов и циклов на узлы кластера с учетом их производительности
- Отображение массивов и циклов на устройства узла с учетом их производительности
- Трансформация массивов
- Использование динамической компиляции



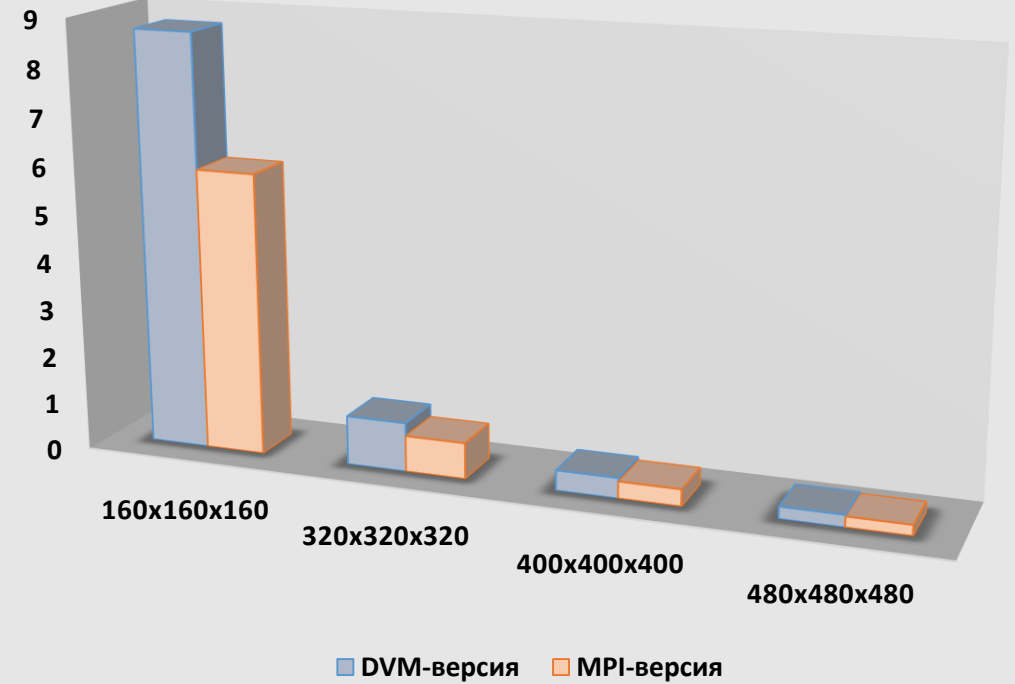
Позволяет эффективно работать на гетерогенных архитектурах, с использованием одновременно и CPU и ускорителя (GPU, Phi), обеспечивая балансировку нагрузки

Результаты тестов для программы автора

160x160x160 точек, масштабируемость, число итераций за секунду



Ускорение, число итераций в секунду



- DVM-система существенно упрощает процесс разработки параллельных программ для гибридных вычислительных кластеров.
- Получаемые DVMH-программы без каких-либо изменений могут эффективно выполняться на кластерах различной архитектуры, использующих многоядерные универсальные процессоры, графические ускорители и сопроцессоры Intel Xeon Phi. Это достигается как за счет различных оптимизаций, которые выполняются статически, при компиляции DVMH-программ, так и за счет динамических оптимизаций.
- Параллельные программы могут динамически настраиваться при запуске на выделенные для их выполнения ресурсы (количество узлов кластера, ядер, ускорителей и их производительность).

Более подробную информацию, включая саму систему, инструкцию пользователя, примеры программ, результаты тестов, публикации и выступления на конференциях, можно найти на сайте системы

dvm-system.org

СПАСИБО ЗА ВНИМАНИЕ