

Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего
профессионального образования Московский физико-технический институт
(государственный университет)

Факультет управления и прикладной математики

Кафедра информатики и вычислительной математики

РАЗРАБОТКА ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ДЛЯ ВЫЧИСЛЕНИЯ ИНТЕГРАЛОВ РЕЛЕЯ И КИРХГОФА В СЛУЧАЕ ЛИНЕЙНО-УПРУГИХ СРЕД

Выпускная квалификационная работа

Выполнил:

студент 273(в) группы

Мареичев Петр Алексеевич

Научный руководитель:

к.ф.-м.н.,

Фаворская Алена Владимировна

Москва 2016

Оглавление

Введение.....	3
Моделирования распространения волн в линейно-упругой среде	5
Метод.....	5
Алгоритм.....	7
Результаты работы программы	9
Обратная задача. Задача волновой миграции	11
Метод.....	11
Алгоритм.....	13
Результаты работы программы	14
Вывод.....	18
Список литературы.....	19
Публикации	20

Введение

Актуальность темы исследования

Нефтегазовая промышленность имеет огромное значение для экономики российской федерации. Одними из центральных задач, ставящихся в этой отрасли, являются задачи геологоразведки.

Проведение геологоразведочных мероприятий позволяет определять структуру и состав грунта, что, в свою очередь, позволяет обнаруживать новые месторождения полезных ископаемых, а также делать заключения о целесообразности разработки того или иного месторождения.

Одним из наиболее используемых, эффективных и надежных методов, применяемых в геологоразведке, является сейсморазведка, совмещающая в себе относительно невысокую стоимость и высокую точность получаемых данных.

Численное моделирование распространения волн позволяет существенно оптимизировать процесс геологоразведки и нефтедобычи, что дает возможность уменьшить себестоимость добычи и, как следствие, существенно увеличить прибыль.

Методы сейсмической миграции представляют огромный интерес для задач сейсморазведки. Данные методы позволяют на основе полученных в результате сейсморазведки сейсмограмм получить координаты отражающих поверхностей в среде, которые физически соответствуют линиям раздела сред. Полученные таким образом данные дают возможность эффективно и с высокой точностью оценить объем месторождения и его точные границы, которые являются очень важными показателями для задач геологоразведки.

Огромный интерес представляет использование новейших методов сейсмической миграции на основе интегралов Релея(Кирхгофа) для случая линейно-упругой среды, которые позволяют значительно повысить точность определения местонахождения полезных ископаемых, а также увеличить чувствительность метода.

Практическая значимость работы

Одной из особенностей программ, производящих расчеты для задач геофизики, является необходимость работы с колоссальным объемом данных, что, в свою очередь, требует большое количество времени. Так для задачи волновой миграции необходимо рассчитать точки трехмерного массива по координатам как функции точек трехмерного массива сейсмограммы по двум координатам и времени, потому сложность данных вычислений составляет $\sim O(n^6)$.

Использование параллельных вычислений и адаптивное алгоритма для вычисления на суперкомпьютере или кластере позволяет существенно сократить время, требуемое для расчета, что, в свою очередь, позволяет успешно применять данный алгоритм на практике.

Положения, выносимые на защиту

- Реализована параллельная программа, рассчитанная на работу в системах с распределенной и разделяемой памятью, моделирующая распространение упругих волн в линейно-упругой среде, основанная на использовании интегралов Кирхгофа.
- Реализована параллельная программа, рассчитанная на работу в системах с распределенной памятью, позволяющая численно решать задачу упругой миграции, основанная на использовании интегралов Кирхгофа. Проанализировано ускорение работы программы при работе на разном числе процессоров.

Моделирования распространения волн в линейно-упругой среде

Обобщенные интегралы Кирхгофа [1-4] имеют большое прикладное значение для многих задач физики и геофизики, позволяя получить решение краевой задачи теории упругости в замкнутой форме. Как показано в работе [5] решения, полученные при использовании данного подхода, совпадают качественно и количественно с решениями, полученными сеточно-характеристическим методом.

В программе реализован метод, описанный в работе [6], позволяющий найти приближенное решение системы уравнений, описывающей состояние линейно-упругой среды с применением интегралов Кирхгофа

Метод

Необходимо рассчитать смещение для каждой точки среды в каждый момент времени при заданных начальных условиях.

Формально задача ставится следующим образом:

Ищется вектор смещений \vec{U} в области $D : z > 0$

$$\mathbf{L}\vec{U} - \frac{\partial^2 \vec{U}}{\partial t^2} = 0 \quad D = [0, X] \times [0, Y] \times [0, Z]$$

$$\mathbf{L} = c_p^2 \nabla \nabla \cdot - c_s^2 \nabla \times \nabla \times$$

Где

c_p – скорость продольной волн

c_s – скорость поперечных волн

Обозначим импульс Рикера

$$f(t) = \left(1 - \frac{180}{\alpha^2} \left(t - \frac{\alpha}{2} \right)^2 \right) \exp \left(- \frac{90}{\alpha^2} \left(t - \frac{\alpha}{2} \right)^2 \right), t \in [0, \alpha]$$

В качестве начального условия рассматриваются импульсы двух видов:

Импульс Рикера в круге (С – круг в плоскости $z = 0$ с центром в точке (x_0, y_0))

$$U_k(x, y, 0, t) = \begin{cases} b_k f(t), & (x, y) \in C \\ 0, & (x, y) \notin C \end{cases}$$

Плоская волна:

$$U_1(x, y, 0, t) = b_1 f(t)$$

$$U_2(x, y, 0, t) = b_2 f(t)$$

$$U_3(x, y, 0, t) = b_3 f(t)$$

В работе [6] получены формулы для выражения компонент вектора смещений через интегралы Кирхгофа

$$(U_1)_{IJK}^L = -\left(c_s^2 b_1 \left(\mathbf{D}_{xxz} (I_p)_{IJK}^L + \mathbf{D}_{yyz} (I_s)_{IJK}^L + \mathbf{D}_{zzz} (I_s)_{IJK}^L\right) + c_s^2 b_2 \mathbf{D}_{xyz} (I_{ps})_{IJK}^L + c_p^2 b_3 \mathbf{D}_{xzz} (I_{ps})_{IJK}^L\right) \frac{1}{2\pi h^3}$$

$$(U_2)_{IJK}^L = -\left(c_s^2 b_2 \left(\mathbf{D}_{xxz} (I_s)_{IJK}^L + \mathbf{D}_{yyz} (I_p)_{IJK}^L + \mathbf{D}_{zzz} (I_s)_{IJK}^L\right) + c_s^2 b_1 \mathbf{D}_{xyz} (I_{ps})_{IJK}^L + c_p^2 b_3 \mathbf{D}_{yzz} (I_{ps})_{IJK}^L\right) \frac{1}{2\pi h^3}$$

$$(U_3)_{IJK}^L = -\left(c_p^2 b_3 \left(\mathbf{D}_{xxz} (I_s)_{IJK}^L + \mathbf{D}_{yyz} (I_s)_{IJK}^L + \mathbf{D}_{zzz} (I_p)_{IJK}^L\right) + c_s^2 b_1 \mathbf{D}_{xzz} (I_{ps})_{IJK}^L + c_s^2 b_2 \mathbf{D}_{yzz} (I_{ps})_{IJK}^L\right) \frac{1}{2\pi h^3}$$

Где за I_p , I_s , I_{ps} обозначены

1. Для импульса Рикера в круге С

$$r = \sqrt{(x' - x)^2 + (y' - y)^2 + (z')^2}$$

$$I_p = \int_0^x \int_0^y \frac{1}{r} \left(\frac{\alpha^2}{180} \exp\left(-\frac{90}{\alpha^2} \left(t' - \frac{r}{c_p} - \frac{\alpha}{2}\right)^2\right) - \frac{\alpha^2}{180} \exp\left(-\frac{90}{\alpha^2} \left(t' - \frac{\alpha}{2}\right)^2\right) \right) dx dy$$

$$I_s = \int_0^x \int_0^y \frac{1}{r} \left(\frac{\alpha^2}{180} \exp\left(-\frac{90}{\alpha^2} \left(t' - \frac{r}{c_s} - \frac{\alpha}{2}\right)^2\right) - \frac{\alpha^2}{180} \exp\left(-\frac{90}{\alpha^2} \left(t' - \frac{\alpha}{2}\right)^2\right) \right) dx dy$$

2. Для плоской волны

$$r = \sqrt{(x' - x)^2 + (y' - y)^2 + (z')^2}$$

$$I_p = \int_c \frac{1}{r} \left(\frac{\alpha^2}{180} \exp \left(-\frac{90}{\alpha^2} \left(t' - \frac{r}{c_p} - \frac{\alpha}{2} \right)^2 \right) - \frac{\alpha^2}{180} \exp \left(-\frac{90}{\alpha^2} \left(t' - \frac{\alpha}{2} \right)^2 \right) \right) dx dy$$

$$I_s = \int_c \frac{1}{r} \left(\frac{\alpha^2}{180} \exp \left(-\frac{90}{\alpha^2} \left(t' - \frac{r}{c_s} - \frac{\alpha}{2} \right)^2 \right) - \frac{\alpha^2}{180} \exp \left(-\frac{90}{\alpha^2} \left(t' - \frac{\alpha}{2} \right)^2 \right) \right) dx dy$$

Операторы дифференцирования D_{abc} имеют вид

$$D_{xxz} W_{IJK}^L = \left(W_{I+2,J,K+1}^L - 2W_{I+1,J,K+1}^L + W_{I,J,K+1}^L \right) - \left(W_{I+2,J,K}^L - 2W_{I+1,J,K}^L + W_{I,J,K}^L \right)$$

$$D_{yyz} W_{IJK}^L = \left(W_{I,J+2,K+1}^L - 2W_{I,J+1,K+1}^L + W_{I,J,K+1}^L \right) - \left(W_{I,J+2,K}^L - 2W_{I,J+1,K}^L + W_{I,J,K}^L \right)$$

$$D_{zzz} W_{IJK}^L = W_{I,J,K+3}^L - 3W_{I,J,K+2}^L + 3W_{I,J,K+1}^L - W_{I,J,K}^L$$

$$D_{xyz} W_{IJK}^L = W_{I+1,J+1,K+1}^L - W_{I+1,J+1,K}^L - W_{I,J+1,K+1}^L - W_{I+1,J,K+1}^L + W_{I+1,J,K}^L + W_{I,J+1,K}^L + W_{I,J,K+1}^L - W_{I,J,K}^L$$

$$D_{xzz} W_{IJK}^L = \left(W_{I+1,J,K+2}^L - 2W_{I+1,J,K+1}^L + W_{I+1,J,K}^L \right) - \left(W_{I,J,K+2}^L - 2W_{I,J,K+1}^L + W_{I,J,K}^L \right)$$

$$D_{yzz} W_{IJK}^L = \left(W_{I,J+1,K+2}^L - 2W_{I,J+1,K+1}^L + W_{I,J+1,K}^L \right) - \left(W_{I,J,K+2}^L - 2W_{I,J,K+1}^L + W_{I,J,K}^L \right)$$

Алгоритм

1 Интерфейс MPI

Вычислительный процесс делится между процессами по одной из координат.

Остаток распределяется между всеми процессами. Основной процесс не производит основных вычислений, а занимается только записью результатов и координированием остальных процессов. Размер блока данных $MyNx$ по оси X, содержащей Nx точек для процесса имеющего ранг $rank > 0$ и при общем числе процессов $size \geq rank$ рассчитывается по формуле

If($rank \leq (X \% (size - 1))$) $MyNx = (Nx / (size - 1)) + 1$;

Else $MyNx = ((Nx / (size - 1)))$;

Ход работы:

1. На первом этапе основной процесс считывает входные данные, и рассылает всем исполнителям переменные методом `MPI_Bcast(...)`. После чего каждый из процессов рассчитывает границы своей части массива и выделяет необходимое количество памяти.
2. Для каждого шага по времени, каждый из процессов вычисляет значения Массива интегралов для каждой из точек своего массива. После происходит обмен краевыми значениями между процессами, используя неблокирующую передачу и блокирующий прием.
3. Затем каждый процесс из полученных данных рассчитывает значение вектора сдвига. После массив смещений собирается и передается основному процессу, используя метод `MPI_Gatherv(...)` Затем основной массив высчитывает массив векторов сдвига, считает массив скаляров и осуществляет запись в файл, а остальные процессы переходят к расчёту следующего временного слоя.

2 Интерфейс OpenMp

массив делится между процессами по одной из координат.

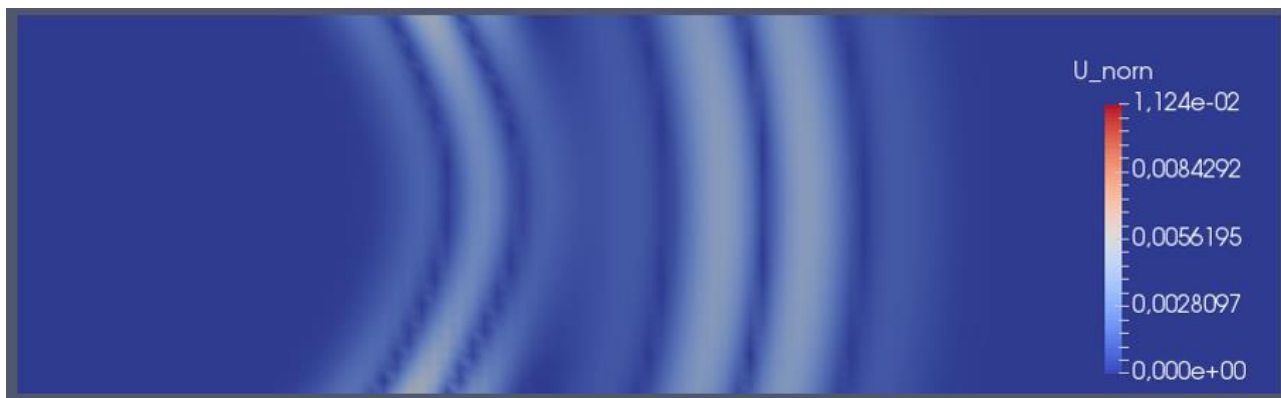
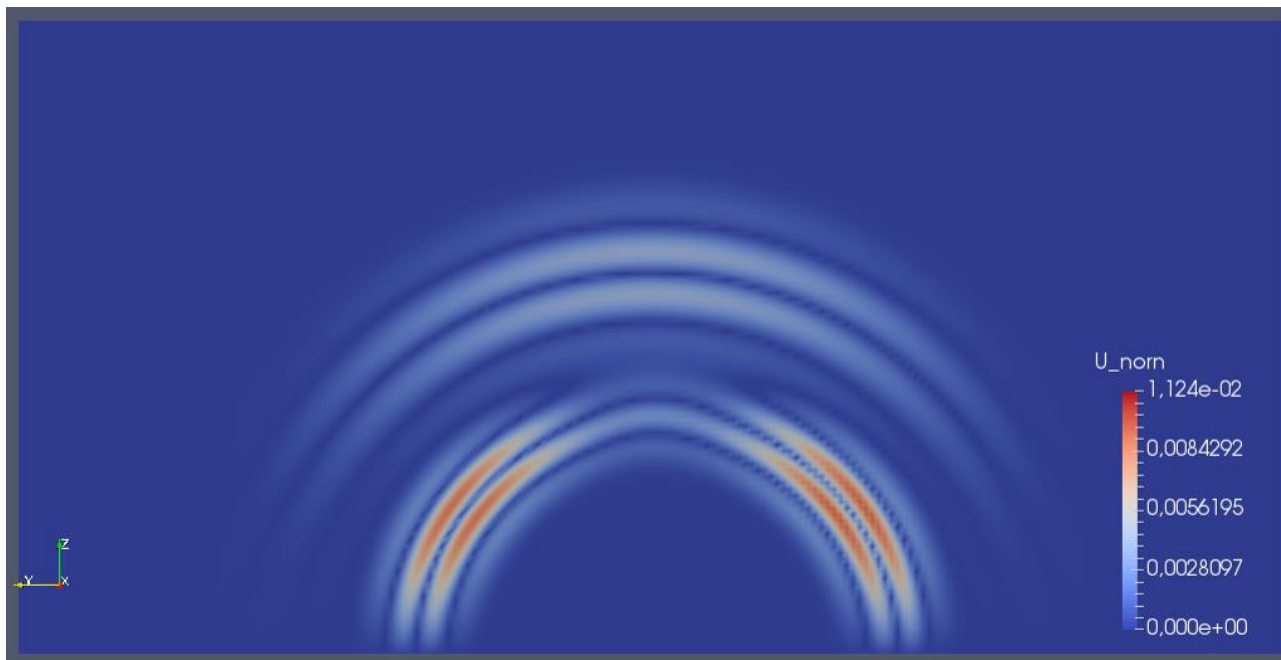
Ход работы:

1. Процесс считывает входные данные и вычисляет параметры.
2. Используя директиву `#pragma parallel for` порождается несколько нитей исполнения, высчитывающих массив интегралов. Все нити ждут пока завершится запись в массив. Таким образом гарантируется неизменность массива I при вычислении U
3. Считается массив U . Каждая нить рассчитывает свой участок массива, имея при этом доступ ко всему массиву I . После чего каждая нить ждет завершения расчетов всех остальных
4. Основной процесс производит запись полученного временного слоя в файл, после чего алгоритм делает шаг по времени и повторяет пункты 2-4

Результаты работы программы

Результаты визуализации массивов, полученных в результате работы программы. Для визуализации использовался пакет ParaView 5.0

Импульс Рикера в круге



Плоская волна



Обратная задача. Задача волновой миграции

Задача волновой миграции

Для имеющегося массива смещений, зависящего от координат на плоскости и от времени воссоздать, трехмерный массив, соответствующий исследуемой области, на котором будут видны отражающие поверхности в среде, физически означающие линии раздела областей, имеющих разную плотность [2][7].

Метод

Предложенный метод расчета волновой миграции, на основе интегралов Кирхгофа

Введем обозначения

x, y, z – координаты точки объема

x', y' - координаты точки поверхности

$$r = |\vec{r} - \vec{r}'|$$

Индекс n принимает значения [1; 2; 3]

Индекс A принимает значения P или S

$$\text{Индекс } B = \begin{cases} S & \text{при } n = 1, 2 \\ P & \text{при } n = 3 \end{cases}$$

Получено выражения для компонент вектора

$$U_n^{\text{MIGR}}(x, y, z) = \frac{1}{2\pi} \iint_S J_n ds$$

где:

$$J_1(x, y, z) = -R_{xxz}^{1p} - R_{yyz}^{1s} - R_{z3}^{1s} - R_{xyz}^{2p} + R_{xyz}^{2s} - R_{zzx}^{3p} + R_{zzx}^{3s}$$

$$J_2(x, y, z) = -R_{yyz}^{2p} - R_{xxz}^{2s} - R_{z3}^{2s} - R_{xyz}^{1p} + R_{xyz}^{1s} - R_{zzy}^{3p} + R_{zzy}^{3s}$$

$$J_3(x, y, z) = -R_{z3}^{3p} - R_{xxz}^{3s} - R_{yyz}^{3s} - R_{zzx}^{1p} + R_{zzx}^{1s} - R_{zzy}^{2p} + R_{zzy}^{2s}$$

$$R_{xyz}^{nA} = -3 \frac{1}{r^2} \frac{X}{r} \frac{Y}{r} \frac{z}{r} Q_{An}$$

$$R_{aab}^{nA} = \frac{1}{r^2} \frac{b}{r} \left\{ \left(3 - 7 \left(\frac{a}{r} \right)^2 \right) Q_{An} + \left(2 \left(\frac{a}{r} \right)^2 - 1 \right) S_{An} \right\}$$

$$R_{z3}^{nA} = 3 \frac{1}{r^2} \frac{z}{r} \left\{ \left(3 - 5 \left(\frac{z}{r} \right)^2 \right) Q_{An} + \left(2 \left(\frac{z}{r} \right)^2 - 1 \right) S_{An} \right\}$$

А функции Q_{An} и S_{An} имеют вид

$$Q_{An} = \left(\frac{c_B}{r} \right)^2 \int_0^{\frac{r}{c_A}} U_n(\bar{r}', \tau) \tau d\tau$$

$$S_{An} = \left(\frac{c_B}{c_A} \right)^2 U_n \left(\bar{r}', \frac{r}{c_A} \right)$$

Результирующий массив выражается через компоненты вектора смещений следующим образом:

$$U_{\text{MIGR}}(x, y, z) = \sqrt{\left(U_1^{\text{MIGR}}(x, y, z) \right)^2 + \left(U_2^{\text{MIGR}}(x, y, z) \right)^2 + \left(U_3^{\text{MIGR}}(x, y, z) \right)^2}$$

Легко видеть, что значение массива U_{port} в каждой точке есть функция координат и сейсмограммы и не зависит от значений соседних точек. В связи с этим целесообразным является параллельное решение задачи, и можно ожидать высокого коэффициента ускорения для параллельной реализации алгоритма.

Для реализации алгоритма используется интерфейс MPI (message passing interface).

Алгоритм

Распределение элементов между исполнителями происходит по общему числу элементов массива. Остаток от деления числа точек массива на число процессов считается малым по сравнению с общим объемом массива ($N \gg P^2$), потому принято решение добавить оставшиеся точки для расчета одному из процессов. Формулы для расчёта координаты (I, J, K) точки с номером m, для процесса ранга rank и общего число процессов size;

$$I \in [0, Nx]; J \in [0, Ny]; K \in [0, Nz]$$

If (m!=0) mTotal=m+((Nx*Ny*Nz)/size)*rank+((Nx*Ny*Nz)%size);

Else mTotal=m;

I=mTotal/(Ny*Nz);

J=(mTotal%(Ny*Nz))/Nz;

K=mTotal%Nz;

Ход работы:

1. основной процесс(root) считывает переменные из входного файла, и записывает в массив данные сейсмограммы.
2. Методом MPI_Bcast(...) значения переменных раздаются всем исполнителям. После чего исполнители вычисляют размер своей части массива (точки массива делятся поровну между процессами) и выделяют необходимый объем памяти.
3. Методом MPI_Bcast(...) каждому процессу рассылается копия массива сейсмограммы.
4. Каждый из исполнителей независимо производит вычисление своей части массива, который потом собирается и передается основному процессу методом MPI_Gather();
5. Основной процесс производит сохранение полученных данных и завершает работу.

Результаты работы программы

Система:

Тестирование проводилось на геосервере мфти

Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
CPU(s): 12
On-line CPU(s) list: 0-11
Thread(s) per core: 1
Core(s) per socket: 6
Socket(s): 2
CPU family: 6
Model: 62
Stepping: 4
CPU MHz: 2101.000

Входные данные:

$X = 10\text{км}$; $Y = 2\text{м}$; $Z = 2.5\text{км}$ – размер исследуемой области

$h = 10\text{м}$ – шаг сетки

$T = 4\text{с}$ – общее время записи

$\tau_{\text{ay}} = 0.002\text{с}$ – шаг по времени

$c_p = 2500\text{ м/с}$; $c_s = 1250\text{ м/с}$ – скорость продольных и поперечных волн

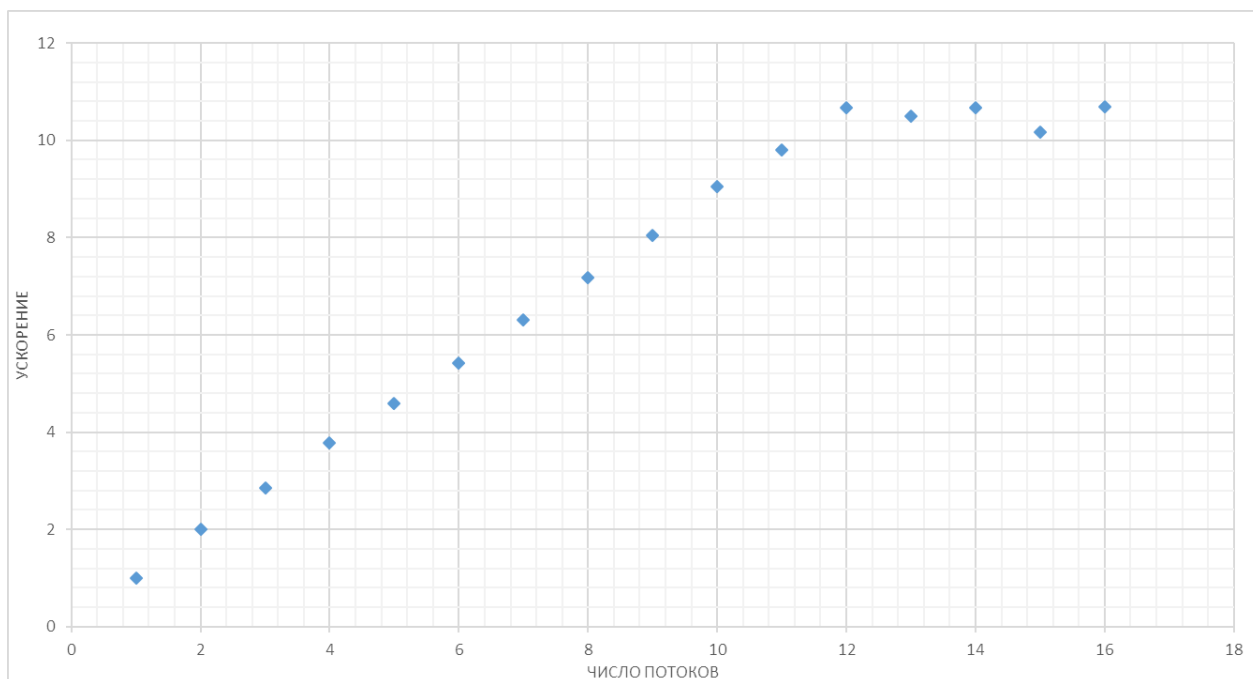
время работы:

на 12 процессорах(максимальное полученное ускорение) **9 минут 35 секунд**

на 1 процессоре **102 минуты 15 секунд**

График зависимости ускорения от числа процессов

ускорение(n) = $\frac{t(1)}{t(n)}$, где $t(n)$ – время работы программы для n процессов



Согласно графику, коэффициент ускорения $k \sim 0.9$

Результаты

Результаты визуализации массивов, воссозданных в результате работы программы. Для визуализации использовался пакет ParaView 5.0

Сверху приведены исходные данные модели, снизу – результат расшифровки сейсмограммы, полученной для данной модели

Модель 1

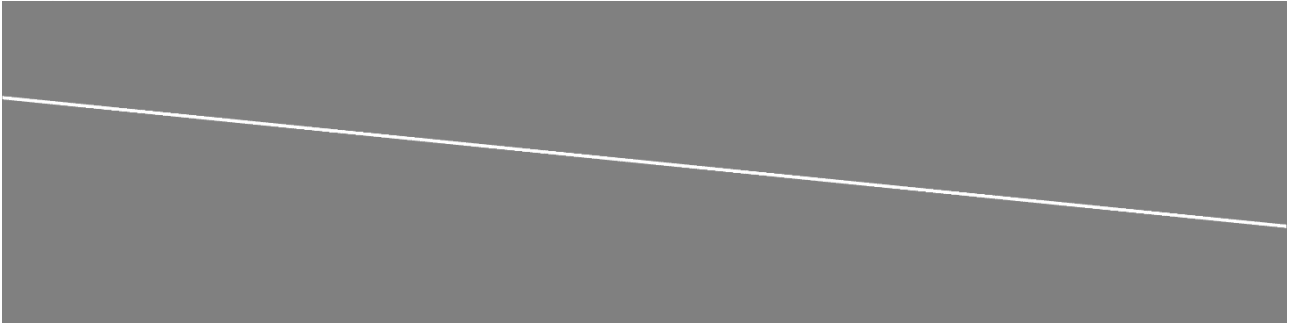


Рис 2.1.1

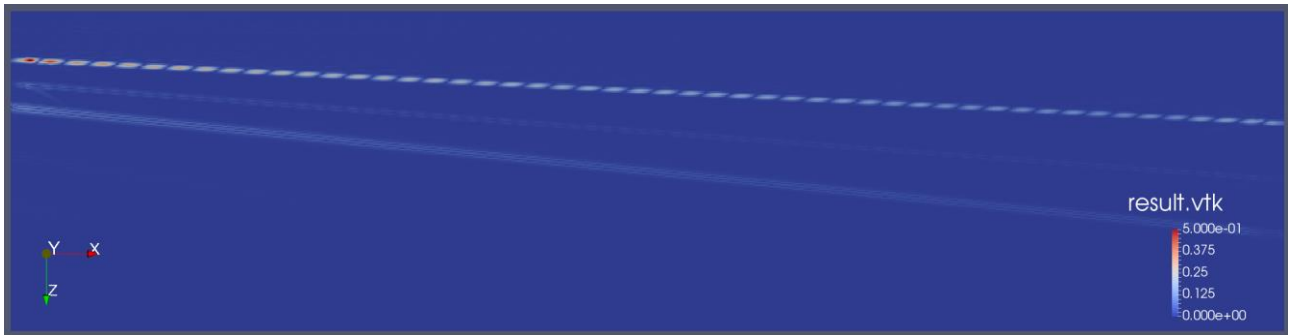


Рис 2.1.1

Модель 2

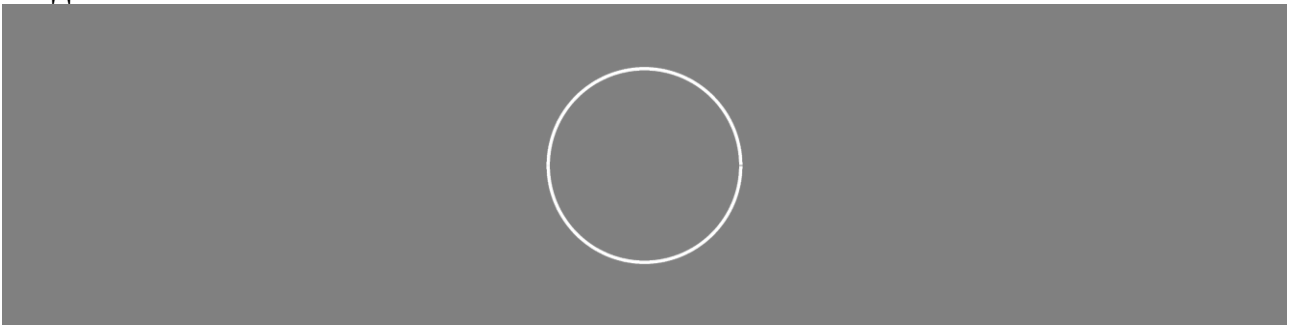


Рис 2.2.1

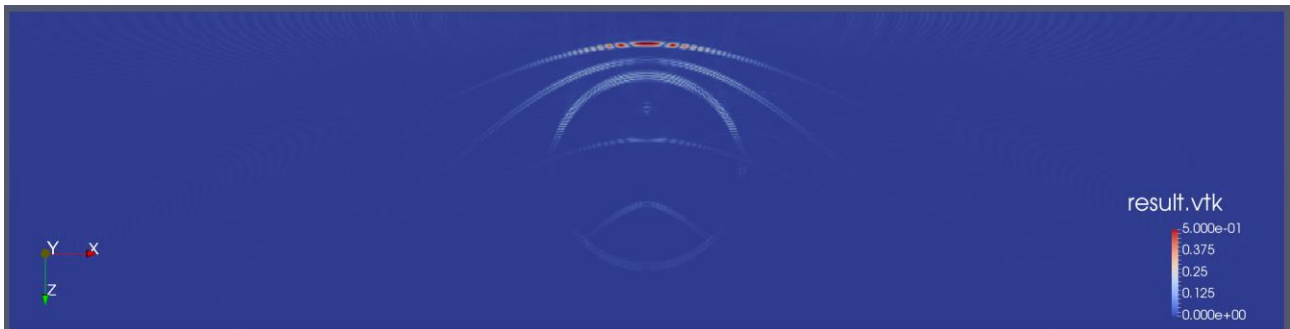


Рис 2.2.2

Модель 3

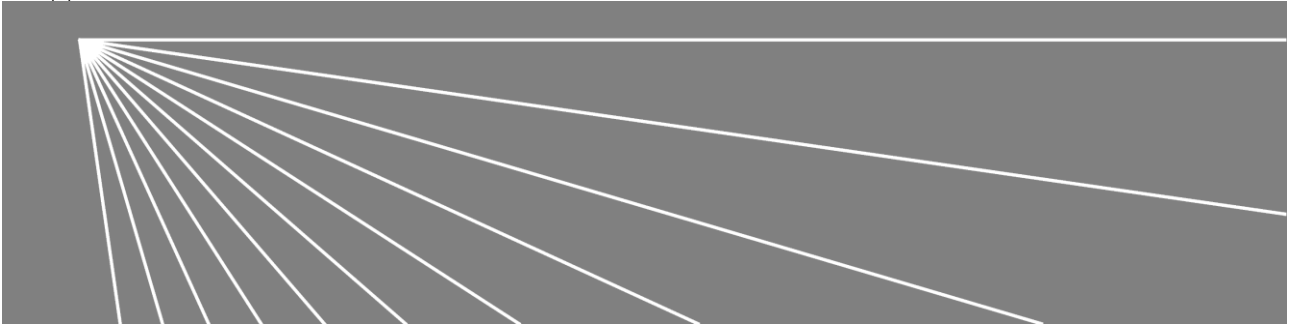


Рис 2.3.1

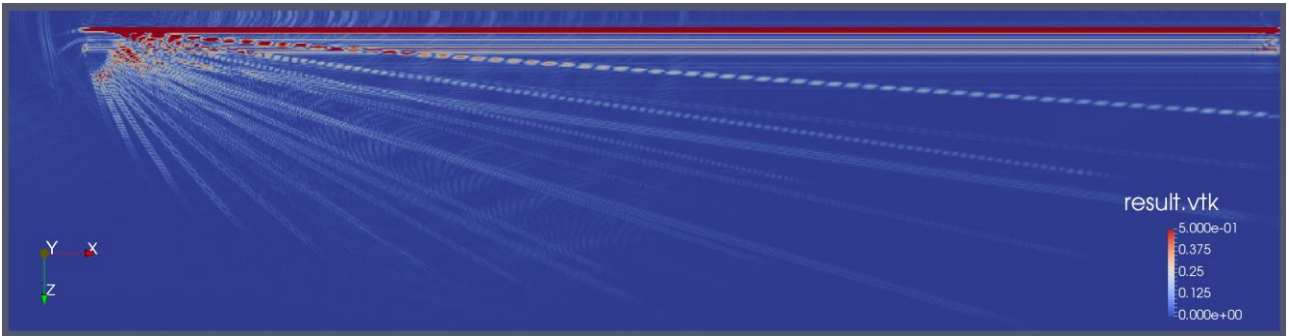


Рис 2.3.2

Модель 4



Рис 2.4.1

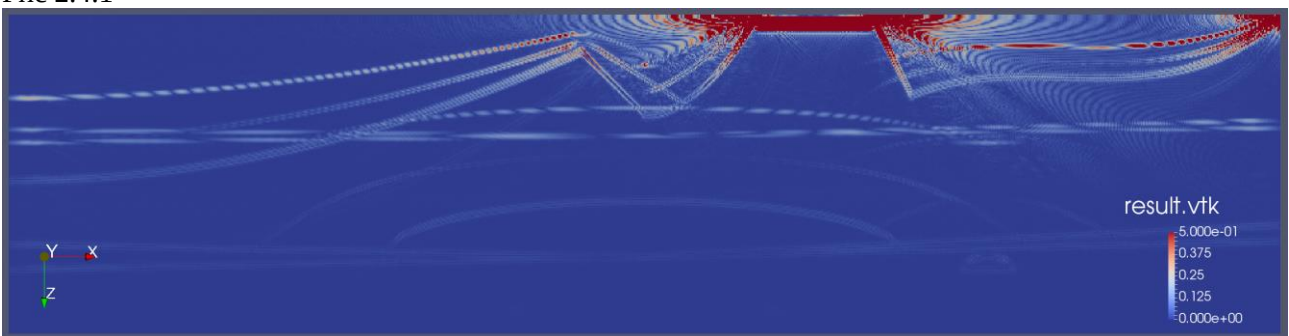


Рис 2.4.2

Вывод

При реализации предложенных методов добиться высокой производительности и хорошего коэффициента ускорения, что позволяет продолжать работать над алгоритмом и делает его пригодным для решения реальных задач сейсморазведки.

Список литературы

1. *Жданов, М.С., Матусевич В.Ю., Френкель М.А.* Сейсмическая и электромагнитная миграция. – М.: Наука, 1988. – 376 стр.
2. *Zhdanov M.S.* Integral Transforms in Geophysics. – New York, Berlin, London, Tokyo: Springer-Verlag, 1988. – 367 p.
3. *Zhdanov M.S.* Geophysical inverse theory and regularization problems. – Amsterdam: Elsevier, 2002. – 628 pp.
4. *Жданов М.С.* Теория решения обратных задач в геофизике. – М.: Научный Мир, 2007. – 710 стр.
5. *Фаворская А.В., Голубев В.И.* Решение прямых и обратных задач с помощью интегралов Кирхгофа // Труды 57-й научной конференции МФТИ. – 2014. – Т. 2. – С. 134 – 135.
6. *Фаворская А.В., Жданов М.С.* Обобщенные интегралы Кирхгофа для описания волновых процессов в линейно-упругих средах // Сборник научных трудов МФТИ «Математическое моделирование информационных систем». – 2015. – С. 4 – 8.
7. *Жданов М.С.* Геофизическая электромагнитная теория и методы // Пер. с англ. // Под ред. Е.П.Велихова. — М.: Научный мир, 2012. — 680 с:

Публикации

1. Тезисы научной конференции МФТИ (2015 год)
2. Сборник научных трудов МФТИ «Модели и методы обработки информации». – 2016.