

Инверсия дисперсии скоростей поверхностных волн

Проводит, в зависимости от фазы: предобработку сейсмических данных, спектральный анализ и извлечение кривых дисперсии скоростей поверхностных волн; инверсию полученных кривых; постобработку и интерполяцию результатов.

Входными данными могут быть 2D и 3D сейсмограммы ОПВ или 3D сейсмограммы ОГТ, размещенные в СБД или файлах SEG-Y.

1. Основные опции (колонки 9-10)

пусто - фаза инверсии задается параметром STEP.

Для любой фазы необходимо задавать все параметры

2. Параметры

2.1. Параметры, общие для всех опций

PATH="путь"

Путь к общему каталогу для обмена информацией между фазами инверсии, сохранения промежуточных и финальных результатов, данных для контроля качества и листингов.

По умолчанию: нет, обязательный параметр

STEP=фаза

Фаза инверсии: **PRE** – предобработка, спектральный анализ и извлечение кривых; **INV** – инверсия извлеченных кривых; **POST** – пост обработка и интерполяция результатов.

По умолчанию: нет, обязательный параметр

OVERWRITE

При указании этой опции разрешается перезапись результатов.

2.2. Параметры инверсии

Остальные параметры описаны в прилагаемом документе. При запуске любой фазы инверсии должны задаваться все параметры. Из них формируется файл параметров YAML, упоминаемый далее в описании модуля.

3. Рекомендации

Модуль использует окружение Python, обычно устанавливаемое в папку \$alt_root/Applis/python/OS, где OS – операционная система: RedOS7.3, CentOS7 и т.д.

Перед использованием модуля необходимо убедиться, что оно установлено и обновлено.

Модуль SWINV.

Руководство пользователя

Составил: Яблоков Александр Викторович,
к.ф.-м.н., с.н.с. ИНГГ СО РАН

2025 г.

Введение

Настоящее руководство предназначено для работы с программным модулем, разработанным для обработки сейсмических данных методом SWINV (Surface Waves Inversion) – инверсии кривых дисперсии скоростей поверхностных волн. Этот модуль представляет собой комплексный инструмент, который охватывает все этапы обработки данных, необходимые для построения 2D и 3D скоростных моделей S-волны верхней части геологического разреза.

Модуль реализует полный цикл обработки, начиная с входных сейсмических данных и заканчивая получением итоговых скоростных моделей, готовых для интерпретации и дальнейшего анализа. Обработка данных организована в виде последовательности взаимосвязанных подмодулей, каждый из которых выполняет определенную функцию:

Препроцессинг: выполняет предварительную обработку сейсмических данных, включая выборку трасс, разделение сейсмограмм, отбраковку по качеству (SNR), регуляризацию, и подготовку данных к дальнейшему анализу. Предобработанные данные сохраняются в формате SEG-Y для контроля качества.

Спектральный анализ: анализирует спектральные характеристики сейсмических данных с целью извлечения дисперсионных кривых. Реализованы два подхода к расчету V-f изображения, дополнение нулями для улучшения разрешения, автоматическое извлечение дисперсионных кривых различными методами. Полученные спектральные изображения и дисперсионные кривые сохраняются в форматах SEG-Y и PNG.

Инверсия: выполняет обращение извлеченных дисперсионных кривых для построения одномерных моделей скорости S-волн. Предусмотрено три алгоритма инверсии: Оккама (локальная оптимизация), GWO и SSA (глобальная оптимизация). Восстановленные скоростные модели и дисперсионные кривые сохраняются в формате PNG для контроля качества.

Постпроцессинг: выполняет постобработку полученных скоростных моделей, включая отбраковку моделей, интерполяцию, сглаживание, заполнение рельефа, обрезку площади по полигону, заполнение пропусков и сохранение результатов в форматах SEG-Y и FDM. Для контроля качества изображения интерполированных скоростных моделей сохраняются в формате HTML.

В данном руководстве подробно описаны функциональные возможности модуля, требования к входным данным, последовательность действий при обработке параметры, влияющие на результат обработки на каждом этапе. Изучение данного руководства позволит пользователям эффективно использовать модуль для построения 2D и 3D скоростных моделей S-волн, оптимизируя параметры обработки и достигая наилучших результатов.

Описание входных данных

Тип данных	Режим обработки	Требования к входным сейсмограммам	Дополнительные файлы
2D	—	Сейсмические трассы и заголовки: FieldRecord (FFID источника), SourceX, SourceY, GroupX, GroupY Предварительная сортировка не требуется.	1. Для обработки в сортировке ОТВ может быть задан текстовый (.txt) файл с FFID в формате 1 столбец: ffid. Принимает параметр «path4ffid_file». В отсутствии файла заголовок читается из данных. 2. Для обработки в сортировке ОГТ может быть задан текстовый (.txt) файл с CDP в формате 1 столбец: cdp. Принимает параметр «path4ffid_file». В отсутствии файла заголовок читается из данных
3D	в сортировке ОТВ (sort_3d_order = csp)	Сейсмические трассы и заголовки: FieldRecord (FFID источника), SourceX, SourceY, GroupX, GroupY Предварительная сортировка не требуется.	3. Для извлечения кривых используется табличный .csv файл в формате 3 столбца: freq, v_min, v_max (частота, минимальное и максимальное значение диапазона фазовой скорости). Принимает параметр «path4dc_limits». 4. Для инверсии может быть использована модель Vp в текстовом (.txt) формате 4 столбцов: x, y, z, vp. Либо в табличном .csv в формате 3 столбца: depth, vp, vp2vs (глубина, значение скорости Vp, значение отношения Vp / Vs). Либо в .segy файле в формате: шаг по глубине в задается в бинарном заголовке в dt в метрах*1000, и заголовки CDP_X, CDP_Y. Принимает параметр «path4vp_model».
3D	в сортировке ОГТ (sort_3d_order = cdp)	Сейсмические трассы и заголовки: CDP (номер трассы CDP), CDP_X, CDP_Y Предварительная сортировка не требуется. Координаты CDP_X, CDP_Y должны быть заранее пробинированы, а индекс (CDP)	5. Для инверсии методом глобальной оптимизации могут быть заданы ограничение пространства решений в .csv файле в табличном формате 5 столбцов: Vs_min, Vs_max, h_min, h_max (минимальное и максимальное значение Vs в слое, минимальная и максимальная мощность слоя). Мощности последнего слоя задаются нулями. Принимает параметр «path4vs_limits».

		иметь уникальное значение для каждой точки ОГТ. Размер бина выбирает обработчик.	<p>5. Файл рельефа в точках ПВ и ПП в текстовом (.txt) формате 3 столбца: x, y, z. В отсутствии файла задается нулевой рельеф. Принимает параметр «path2elevation».</p> <p>6. Файл полигона площади для 3D в текстовом (.txt) формате 2 столбца: x, y. В отсутствии файла будет применен выпуклый полигон. Принимает параметр «path2polygon».</p>
--	--	---	---

Функциональные возможности модуля

Обработка данных выполняется в несколько этапов (подмодулей):

1. **Препроцессинг.** Этот подмодуль выполняет предварительную обработку сейсмических данных, включая:
 - a. опциональную выборку сейсмических трасс по диапазону номеров источника или точек ОГТ с заданным шагом;
 - b. разделение сейсмограмм на правую и левую ветвь (для 2Д данных);
 - c. разделение сейсмограмм по секторам окружности с центром в источнике (для 3Д данных в режиме `sort_3d_order = csp`);
 - d. выбор сейсмотрасс в пределах заданных оффсетов;
 - e. отбраковку сейсмограмм по количеству трасс и уровню SNR (отношение сигнал / шум, вычисляется и логируется в процессе обработки);
 - f. регуляризацию сейсмограмм с уменьшением шага между приемниками;
 - g. расчет координат привязки результата инверсии;
 - h. запись лог-файла с информацией о результатах предварительной обработки.
 - i. Для контроля качества предобработанные сейсмограммы сохраняются в формате `segy`.
2. **Спектральный анализ.** Этот подмодуль анализирует спектральные характеристики сейсмических данных для извлечения дисперсионных кривых. В подмодуле реализовано:
 - a. Два подхода (на выбор пользователя) к расчету V-f изображения:
 - i. подход на основе двумерного преобразования Фурье;
 - ii. подход на основе преобразования Стоквелла и фильтрации сейсмограмм во области временного-частотного распределения амплитуд с усилением сигнала поверхностный волны;
 - b. процедура дополнения нулями (падинг) во временной и пространственной оси с целью снижения артефактов циклической свертки и улучшения частотного разрешения;
 - c. автоматическое извлечение дисперсионных кривых в трех вариантах (на выбор пользователя):
 - i. определение максимума амплитуды в заданном диапазоне фазовых скоростей.
 - ii. определение дисперсионной кривой на основе применения алгоритма DBSCAN для кластеризации спектрального изображения и выделения области преобладания амплитуд дисперсионной кривой из фонового шума;
 - iii. извлечение дисперсионной кривой с помощью предобученной сверточный нейронной сетью типа AutoEncoder (AE).
 - d. запись лог-файла с информацией о количестве извлеченных и отбракованных дисперсионных кривых.
 - e. Для контроля качества полученные спектральные изображения и извлеченные дисперсионные кривые сохраняются в формате `segy` и `png`.

3. **Инверсия.** Этот подмодуль выполняет обращение извлеченных дисперсионных кривых для построения одномерных моделей скорости S-волны. Он предлагает три алгоритма инверсии (на выбор пользователя):
- a. алгоритм Оккама (локальная оптимизация) - автоматически по наблюдаемой кривой строит начальную скоростную модель, которая Итеративно приближается к истинной модели в ходе решения линеаризованной задачи. Выход алгоритма - гладкая модель (только значения V_s);
 - b. алгоритм GWO (Grey Wolf Optimizer) (глобальная оптимизация) - определяет диапазоны поиска решения автоматически или на основе заданных пользователем параметров. В пределах диапазонов поиска решения стохастическим методом итеративно ищется глобальный минимум. Выход алгоритма - слоистая модель (значения V_s и мощности слоев).
 - c. алгоритм SSA (Salp Swarm Algorithm) (глобальная оптимизация) - определяет диапазоны поиска решения автоматически или на основе заданных пользователем параметров. В пределах диапазонов поиска решения стохастическим методом итеративно ищется глобальный минимум. Выход алгоритма - слоистая модель (значения V_s и мощности слоев).
 - d. запись лог-файла с информацией о количестве обращенных дисперсионных кривых.
 - e. Для контроля качества восстановленные скоростные модели и дисперсионные кривые сохраняются в изображение в формате png.
4. **Постпроцессинг.** Этот подмодуль выполняет постобработку полученных скоростных моделей:
- a. Отбраковку скоростных моделей по значению порога ошибки подбора дисперсионной кривой. Ошибка вычисляется как максимум MAPE (средняя модульная ошибка в процентах).
 - b. Интерполяцию восстановленных одномерных скоростных моделей в двумерные разрезы или трехмерные «кубы».
 - c. Сглаживание интерполированных скоростей и заполнение пропусков.
 - d. Обрезку данных по заданному полигону или выпуклой границе.
 - e. Заполнение рельефа из отдельного файла.
 - f. Сохранение скоростной модели в segy-файл и fdm-файл.
 - g. Для контроля качества интерполированные скоростные модели сохраняются в 2D и 3D изображения в html формате.

Представленная на рис. 1 блок-схема обеспечивает наглядное представление о структуре модуля SWINV, демонстрируя типы входных и выходных данных, используемые дополнительные файлы и этапы их применения, а также взаимосвязи между подмодулями. Это помогает пользователю понять логику работы модуля.

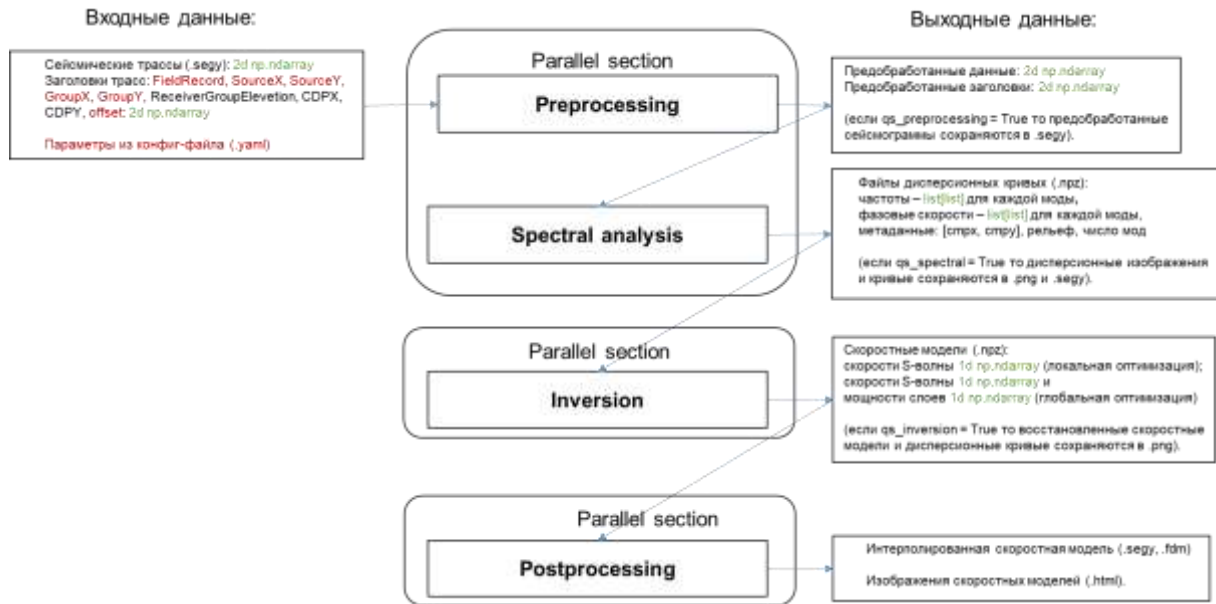


Рис. 1. Общая блок схема модуля SWINV.

Последовательность действий при подборе параметров и обработке данных

Далее в руководстве рассматривается рекомендуемый сценарий настройки параметров, с кратким обзором технических принципов, лежащих в основе метода SWI и влияния различных параметров на результат обработки. Первый запуск первого модуля создаст директорию `runs/` и поддиректории в соответствии с названиями подмодулей и выбранных режимов обработки. Рассмотрим подбор параметров и выполнение контроля качества в соответствии с вызываемыми подмодулями.

Подбор управляемых параметров осуществляется итеративно, путем анализа и контроля качества результатов после каждого этапа обработки. Это позволяет оптимизировать процесс обработки и получить наиболее точные скоростные модели. Управление параметрами осуществляется через конфигурационный файл (конфиг-файл), который представляет собой YAML-файл. Местоположение конфиг-файла относительно корневой директории проекта: `configs / <название текущего проекта> / main.yaml`. Пример структуры конфиг-файла представлен на рис. 2. Полный список параметров и их подробное описание приведены в [Приложении №1](#) к настоящему документу.

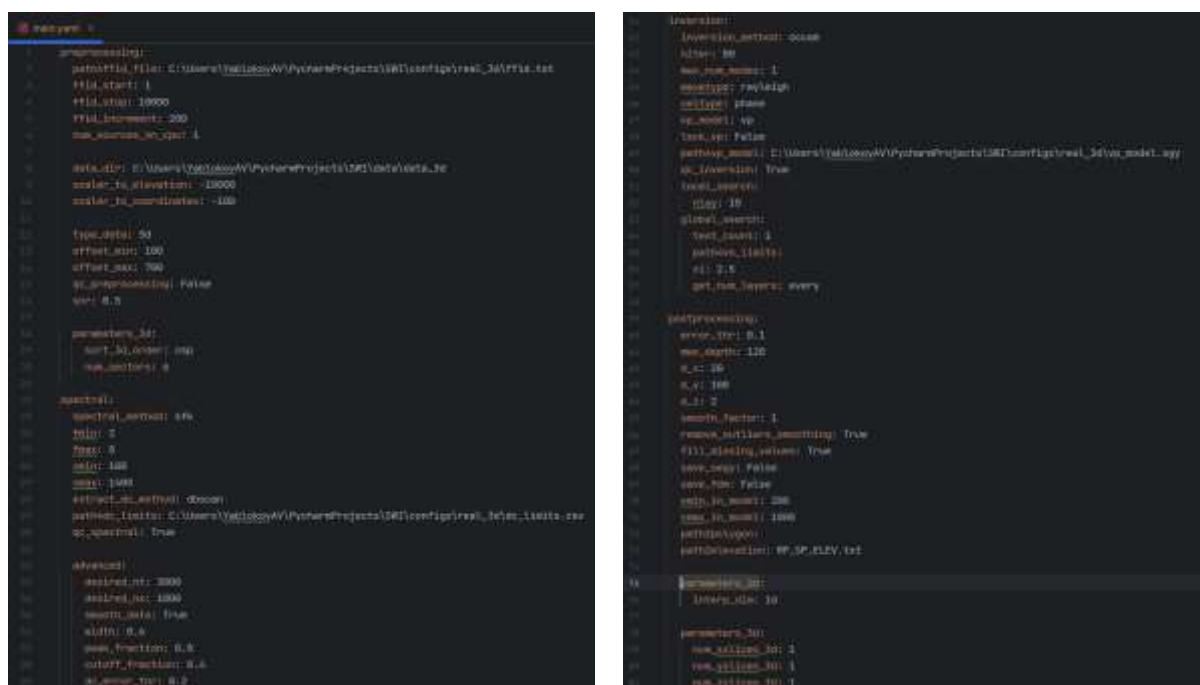


Рис. 2. Содержимое конфигурационного yaml-файла с управляемыми параметрами.

Параметры препроцессинга

Для повышения производительности и эффективного использования ресурсов процессора (CPU), подмодули препроцессинга и спектрального анализа объединены в единый процесс обработки. Это позволяет эффективно распределять данные по ядрам CPU, используя распараллеливание вычислений с использованием MPI.

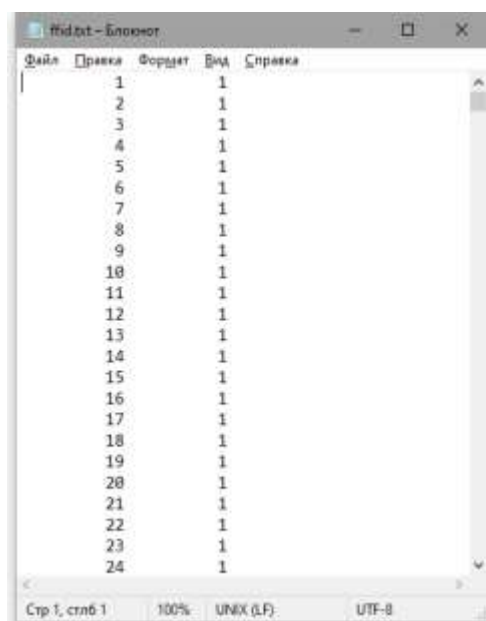
Определение диапазона источников / точек ОГТ

Параметры `ffid_start`, `ffid_stop`, и `ffid_increment`, описанные в [приложении №1](#), определяют диапазон номеров источников для обработки. В случае обработки данных в сортировке по методу `common shot point` (`sort_3d_order = csp`), эти параметры управляют обработкой определенного диапазона номеров источников (FFID). В случае сортировки по

common depth point (sort_3d_order = cdp), эти параметры определяют диапазон номеров точек ОГТ. Номер источника (FFID) или номер точки ОГТ (CDP) может быть получен двумя способами:

- Из отдельного файла (предпочтительный способ при загрузке данных из файлов SEG-Y): абсолютный путь к текстовому файлу, содержащему заголовки (FFID или CDP), указывается в параметре path4ffid_file. Формат файла: Две колонки: номер трассы и значение FFID (или CDP) (рис. 3).
- Из файла с данными (альтернативный способ): параметр path4ffid_file остается пустым. В этом случае FFID (или CDP) извлекается непосредственно из файла с сейсмическими данными (SEG-Y).

Если ffid_increment = 0, то обработка выполняется для всего доступного набора FFID (или CDP). Параметры ffid_start и ffid_stop в этом случае игнорируются. Параметр num_sources_on_cpu определяет количество источников (FFID) или точек ОГТ (CDP), которые обрабатываются на одном ядре CPU. Число задействованных ядер CPU = length([ffid_start:ffid_increment:ffid_stop]) // num_sources_on_cpu. Например, если обрабатывается 100 источников, ffid_increment=1, ffid_start=1, ffid_stop=100, и num_sources_on_cpu = 10, то будет использовано 10 ядер. Значение num_sources_on_cpu = 0 отключает режим параллельного вычисления, и обработка выполняется последовательно на одном ядре.



1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1
15	1
16	1
17	1
18	1
19	1
20	1
21	1
22	1
23	1
24	1

Рис. 3. Формат txt-файла с заголовком FFID, определенному по пути в параметре path4ffid_file. Первый столбец - номер трассы, второй столбец - значение FFID. Формат файла номеров точек ОГТ аналогичен.

Чтение данных

В текущей версии программного модуля SWINV входные данные считываются из набора данных сейсмической базы (СБД) данных Альтаир-М или внешних SEG-Y-файлов. Для указания местоположения данных в СБД нужно задать параметр data_dir в виде sds:PROJECT/IDENT/VERSION. Для задания файлов SEG-Y в data_dir нужно указать абсолютный путь к директории, содержащей данные или абсолютный путь файла данных.

Модуль последовательно обрабатывает наборы данных СБД или SEG-Y-файлы, находящиеся в указанной директории, используя текущие параметры конфигурации, заданные в конфиг-файле. Предполагается, что каждый SEG-Y-файл содержит данные для

одного 2D профиля или одной 3D площади. При этом необходимо указать тип данных (data_type): 2d или 3d.

При необходимости, можно масштабировать значения координат источников и приемников с помощью параметра scaler_to_coordinates. При этом, применяется следующая логика:

- Отрицательное значение коэффициента: значения координат делятся на абсолютное значение коэффициента.
- Положительное значение коэффициента: значения координат умножаются на коэффициент.
- Нулевое значение коэффициента: масштабирование не выполняется.

Методика профилирования 2Д и 3Д данных

Основная задача препроцессинга в 2D и 3D (при sort_3d_order = csp) заключается в приведении сейсмических данных к право-фланговой расстановке, обеспечивающей благоприятные условия для анализа поверхностных волн. В этой расстановке, источник возбуждения располагается слева, а линия приемников – справа. Такая конфигурация гарантирует, что направление распространения поверхностной волны всегда будет положительным, что необходимо для корректного анализа и извлечения фазовых скоростей. Схематическое представление принципа профилирования представлено на рис. 4.

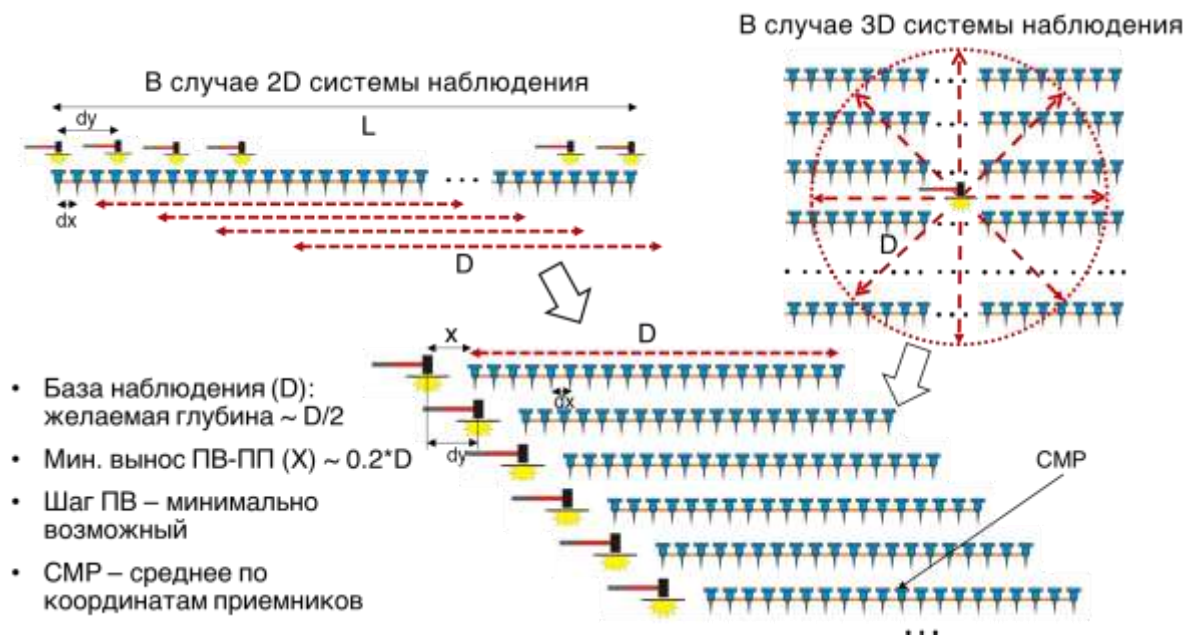


Рис. 4. Схема методики профилирования.

Таким образом, после препроцессинга на вход подмодуля спектрального анализа подаются правофланговые сейсмограммы. Ключевыми параметрами, характеризующими эти сейсмограммы, являются:

- База наблюдения: разница между максимальным и минимальным оффсетом (offset_max - offset_min).
- Минимальный оффсет: offset_min.
- Шаг между приемниками: соответствует шагу между приемниками в исходных данных.
- Шаг между выстрелами (ПВ): соответствует шагу между ПВ в исходных данных.

- Координаты привязки результата инверсии к земной поверхности. Определяется по общей средней точке (СМР – common mid point): координата СМР = координата ПВ + (максимальный оффсет - минимальный оффсет) / 2.

Параметры `offset_min` и `offset_max` позволяют ограничить диапазон оффсетов (расстояний между ПВ и ПП) для сейсмических трасс, используемых в процессе обработки. Выбор оптимальных значений этих параметров, как правило, осуществляется путем визуального анализа сейсмограмм. Необходимо выбрать такой диапазон оффсетов, в котором поверхностная волна четко прослеживается в данных. Пример выбора значений `offset_min` и `offset_max` показан на рис. 5.

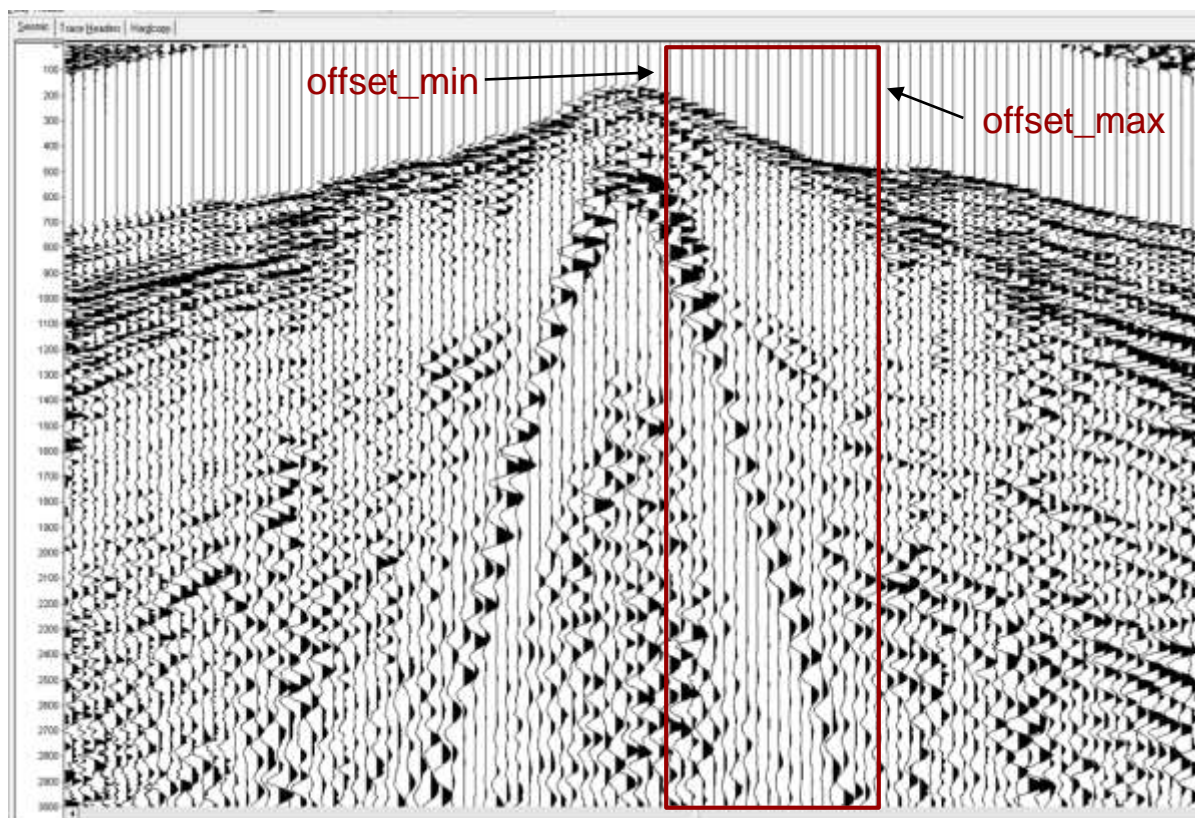


Рис. 5. Пример выбора значений `offset_min` и `offset_max`.

При обработке 3D сейсмограмм, отсортированных по общему пункту возбуждения (ОТВ - common shot point, `sort_3d_order = csp`), количество анализируемых сейсмограмм определяется количеством заданных секторов. В пределах каждого сектора выбираются приемники, данные с которых будут использоваться для обработки. Для разделения данных на сектора выполняется кластеризация по углу в полярной системе координат с использованием алгоритма К-средних. Алгоритм К-средних разделяет приемники на заданное количество (`num_sectors`) кластеров, минимизируя расстояние между приемниками внутри каждого кластера. Пример кластеризации приемников на 6 секторов для одной 3D сейсмограммы ОТВ представлен на рис. 6.

Для сейсмограмм, отсортированных по общей глубинной точке (ОГТ - common deep point, `sort_3d_order = cdr`), на этом этапе выполняется только фильтрация по оффсетам. Этот процесс включает в себя отбор сейсмических трасс, оффсет которых находится в заданном диапазоне (`offset_min` и `offset_max`).

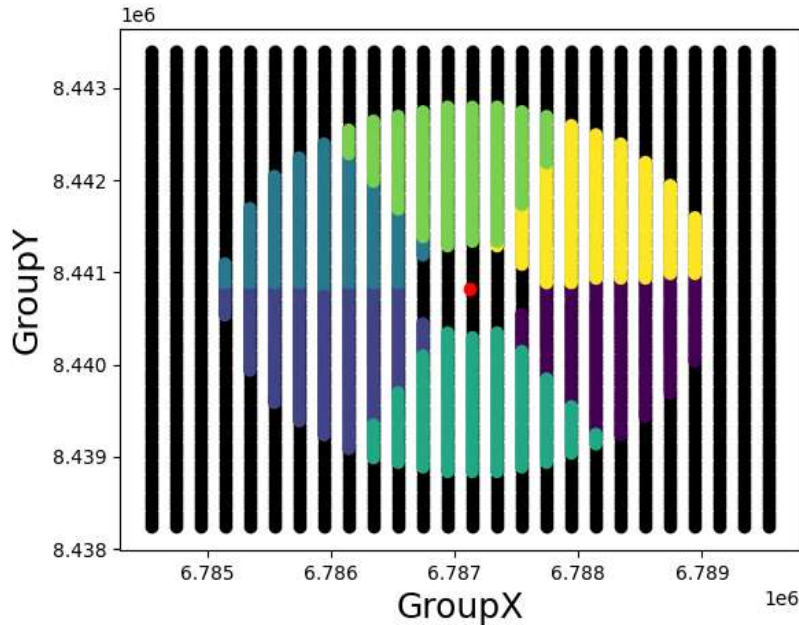


Рис. 6. Пример кластеризации координат сейсмоприемников на 6 секторов для одной 3D сейсмограммы ОТВ. Черными точками отмечены исходные координаты приемников. Красной точкой обозначен источник (пункт взрыва). Разными цветами обозначены кластеризованные сектора, ограниченные заданными `offset_min` и `offset_max`.

Регуляризация сейсмограмм.

После сегментации и обрезки сейсмограмм выполняется их регуляризация методом интерполяции Шпица. Метод интерполяции заключается в том, что линейные события, присутствующие на сейсмограмме и состоящие из равномерно расположенных трасс, могут быть интерполированы в f - x области на основе применения рекурсивной модели сигнала [Spitz, 1991].

Суть метода заключается в том, что N равномерно распределенных трасс, состоящих из L событий (каждое событие является инвариантным от трассы к трассе, за исключением постоянного временного сдвига), могут быть интерполированы, если известно L и если оно меньше N . Каждая входная трасса g_k может быть смоделирована в частотной области по формуле:

$$g_k(f) = \sum_{j=1}^L a_j(f) z_j^{k-1}(f), \quad k = 1, \dots, N$$

– модель сигнала в частотной области, где $a_j(f)$ – преобразование Фурье вейвлета, связанного с событием j , и где $z_j(f)$ – фазовый сдвиг события на частоте f , $z_j(f) = e^{2\pi i f p_j}$, p_j – временной сдвиг между соседними трассами. N – количество равноудаленных трасс из L событий ($L < N$).

Этот метод позволяет не только перенести данные на сетку с равным шагом, но и уменьшить шаг между приемниками и повысить прослеживаемость сигнала поверхностной волны, что существенно улучшает качество последующих спектральных изображений и повышает точность анализа. Этот шаг выполняется автоматически и не требует от пользователя настройки своих параметров. На рис. 7 показан пример сейсмограммы для одного из секторов 3D сейсмограммы до и после регуляризации.

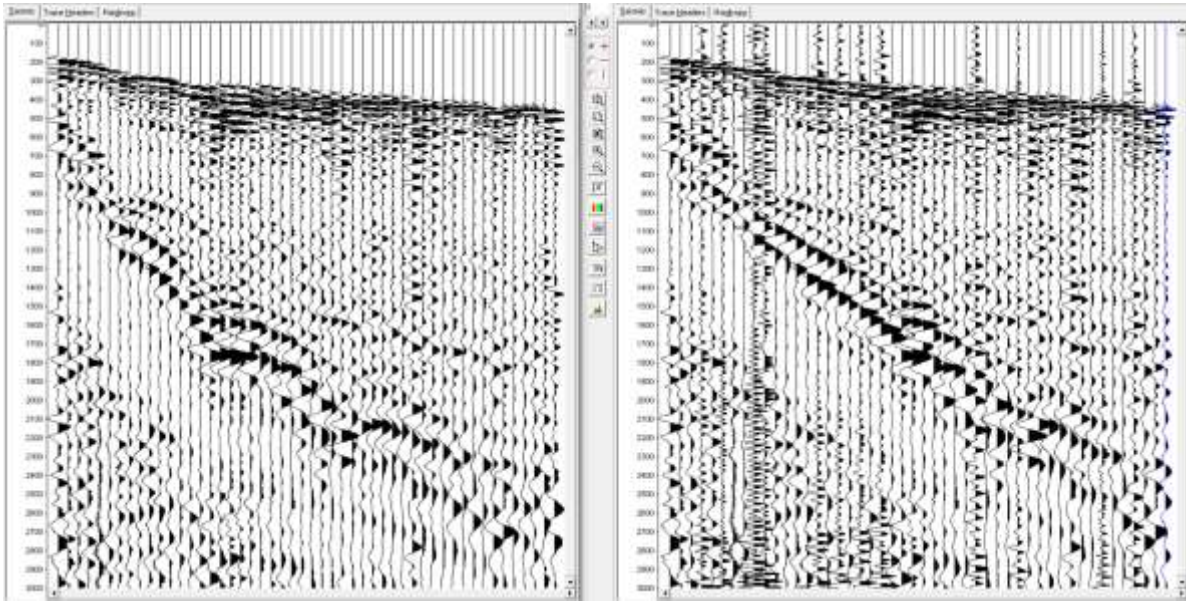


Рис. 7. Пример сегментированной сейсмограммы до (слева) и после (справа) регуляризации.

Отбраковка сейсмограмм по соотношению сигнал / шум.

Для отбраковки сейсмограмм со слабо коррелирующим сигналом поверхностной волны на этапе препроцессинга используется критерий отношения сигнал/шум (SNR). Сейсмограммы, для которых значение SNR ниже заданного порога, исключаются из дальнейшей обработки до этапа спектрального анализа. Это позволяет избежать внесения в результат анализа данных, содержащих преимущественно помехи. Значение сигнала определяется как максимум нормированной функции взаимной корреляции (ФВК) между соседними сейсмическими трассами. Для расчета ФВК используются участки трасс, ограниченные значениями v_{\min} и v_{\max} (из блока спектрального анализа). Значение шума оценивается как 1 - максимум (ФВК). Формула для нормированной ФВК имеет следующий вид [Белоусов, 2011]:

$$\widehat{B}_{1,2}(k) = \frac{\sum_{j=0}^{m-1-k} a_{1,j} \cdot a_{2,j+k}}{\sqrt{\sum_{j=0}^{m-1} a_{1,j}^2} \cdot \sqrt{\sum_{j=0}^{m-1} a_{2,j}^2}}$$

Такой способ расчета SNR выбран для эффективной отбраковки сейсмограмм, не содержащих регулярный и коррелируемый сигнал поверхностной волны. Это особенно важно для метода SWI, где качество выделения дисперсионных кривых напрямую зависит от наличия четкого сигнала поверхностной волны.

Значения SNR для всех сейсмограмм сохраняются в лог-файлах после этапа препроцессинга. На этапе подбора параметров рекомендуется проанализировать часть исходных данных и провести визуальную оценку сейсмограмм для определения характерного уровня шума и наличия полезного сигнала, изучить значения SNR, сохраненные в лог-файлах после препроцессинга. Это позволит установить статистическую связь между визуальной оценкой качества сейсмограмм и значениями

SNR, выбрать значение SNR, которое позволяет отбраковывать заведомо некачественные данные, сохраняя при этом сейсмограммы с полезным сигналом. В качестве примера на рис. 8 изображены две сейсмограммы из одного набора данных со слабо и сильно коррелирующим сигналом поверхностной волны.

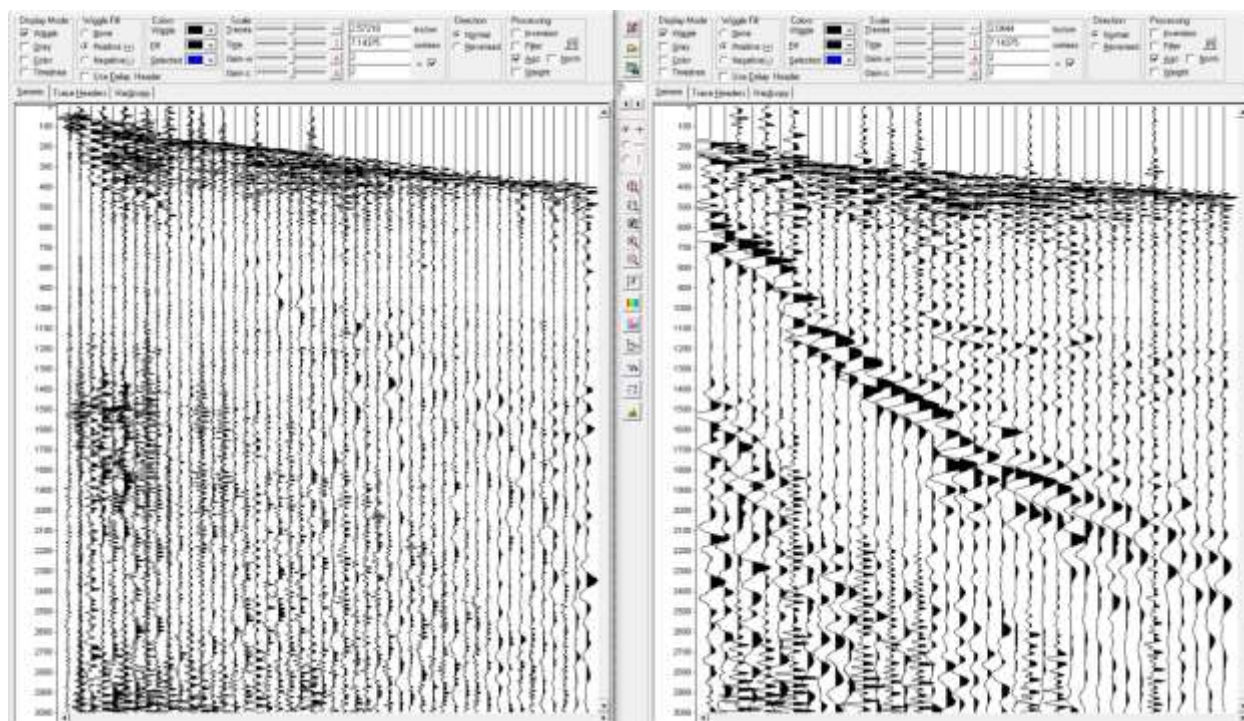


Рис. 8. Пример сейсмограммы со слабо (слева, $\text{snr} = 0.55$) и сильно (справа, $\text{snr} = 1.01$) коррелирующим сигналом поверхностной волны.

Для контроля качества предобработанные сейсмограммы сохраняются в формате segy по следующему пути:

- Для 3D: runs / 3d / preprocessing / <название директории с данными> / <num_sectors> или <cdp> /
- Для 2D: runs / 3d / preprocessing / <название директории с данными> / offsets_<min_offset_max_offset> /.

Параметры спектрального анализа

Этот раздел описывает параметры, управляющие процессом анализа дисперсионных изображений сейсмограмм для извлечения кривых зависимости фазовой скорости поверхностной волны от частоты (дисперсионных кривых). Корректная настройка этих параметров критически важна для получения точных и надежных результатов инверсии.

Расчет дисперсионных изображений

Параметром определяющим подход к расчету спектров является spectral_method. Доступны два варианта (на выбор пользователя):

1. "ФКТ" – расчет на основе двумерного преобразования Фурье. Процедура представляет из себя систему двумерных сглаженных прямоугольных оконных

преобразований. Переход к FK-спектрам выражается формулой [Яблоков и др., 2016]:

$$F(k, f) = \iint f(x, t) W(x - x_0, t - t_0) \exp(-i2\pi ft) \exp(i2\pi kx) dt dx.$$

где $f(x, t)$ - сейсмограмма, $W(x - x_0, t - t_0)$ - гладкая двумерная оконная функция. В результате рассчитывается FK-изображение в диапазоне заданных f_{\min} и f_{\max} .

2. "SFK" – расчет на основе преобразования Стоквелла. Выбор метода SFK (преобразование Стоквелла) часто предпочтительнее, поскольку он обеспечивает лучшее разрешение во времени и частоте, что особенно важно для анализа поверхностных волн, характеризующихся дисперсией. Кратко алгоритм SFK [Serdyukov et al., 2019] преобразования может быть описан следующими последовательными этапами:

- Трансформация сейсмических данных в пространство распределения энергии волн по времени, частоте (в диапазоне f_{\min} и f_{\max}) и расстоянию с использованием обобщенного вейвлет-преобразования Стоквелла.
- Расчет спектра амплитуд сейсмических волн вдоль множества прямых линий (в диапазоне v_{\min} и v_{\max}) с различными наклонами на фиксированных частотах с использованием одномерного преобразования Фурье по пространству. Примечание: значения v_{\min} и v_{\max} рекомендуется выбирать по визуальному анализу сейсмограмм, т.е. по области прослеживания поверхностных волн (рис. 9), а также учитывать что эти диапазоны применяются для всех данных сразу. При этом следует учитывать, что значения v_{\min} меньше 50 могут значительно увеличить время вычислений.
- Построение изображения распределения энергии сейсмических волн в пространстве частоты и волнового числа по максимумам спектра их амплитуд.

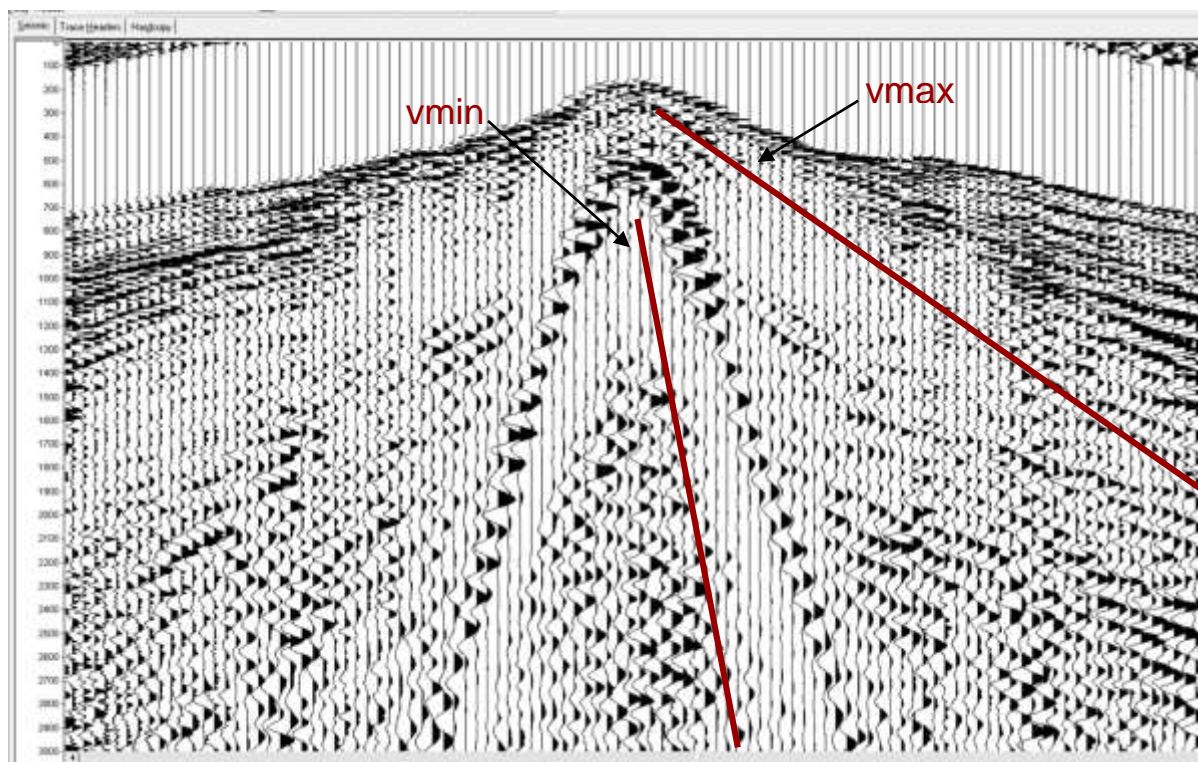


Рис. 9. Пример выбора значений vmin и vmax.

В результате рассчитывается FK-изображение в диапазоне заданных f_{min} и f_{max} . Более подробное описание алгоритма SFK приведено в [приложении №2](#). Сопоставление результатов работы алгоритмов SFK и FKT представлено на рис. 10.

Для дальнейшей оптимизации процесса спектрального анализа предусмотрены следующие дополнительные настройки:

- `desired_nt`: определяет желаемое количество отсчетов по времени (nt) перед расчетом спектрального изображения. Увеличение `desired_nt` путем дополнения данных нулями (zero-padding) позволяет улучшить частотное разрешение спектра.
- `desired_nx`: определяет желаемое количество трасс (nx) перед расчетом спектрального изображения. Аналогично `desired_nt`, увеличение `desired_nx` путем дополнения данных нулями улучшает разрешение в пространственной области (по волновому числу k). Использование `desired_nt` и `desired_nx` позволяет снизить артефакты циклической свертки.
- `smooth_data`: флаг, определяющий необходимость применения двумерного сглаживающего окна к сейсмограмме перед расчетом спектра. Включение этой опции (`smooth_data = True`) позволяет подавить резкие амплитудные всплески в спектре, что может улучшить стабильность и точность извлечения дисперсионных кривых.
- `width`: ширина окна Гаусса в преобразовании Стоквелла (используется только при `spectral_metod = "SFK"`). Этот параметр контролирует степень разделения поверхностной волны на плосковолновые компоненты. Чем меньше значение `width`, тем сильнее разделение. Правильный выбор `width` позволяет оптимизировать баланс между разрешением по фазовой скорости и стабильностью спектрального изображения.

Корректная настройка параметров `desired_nt`, `desired_nx`, `smooth_data` и `width` позволяет повысить качество спектрального изображения и, как следствие, улучшить

результаты извлечения дисперсионных кривых. В частности, параметры `desired_nt` и `desired_nx` могут быть использованы для уменьшения артефактов, связанных с циклической сверткой, а параметр `smooth_data` - для подавления шумов и повышения стабильности спектра. Оптимальное значение `width` зависит от характеристик данных и должно определяться эмпирически, по сопоставлению спектральных изображений.

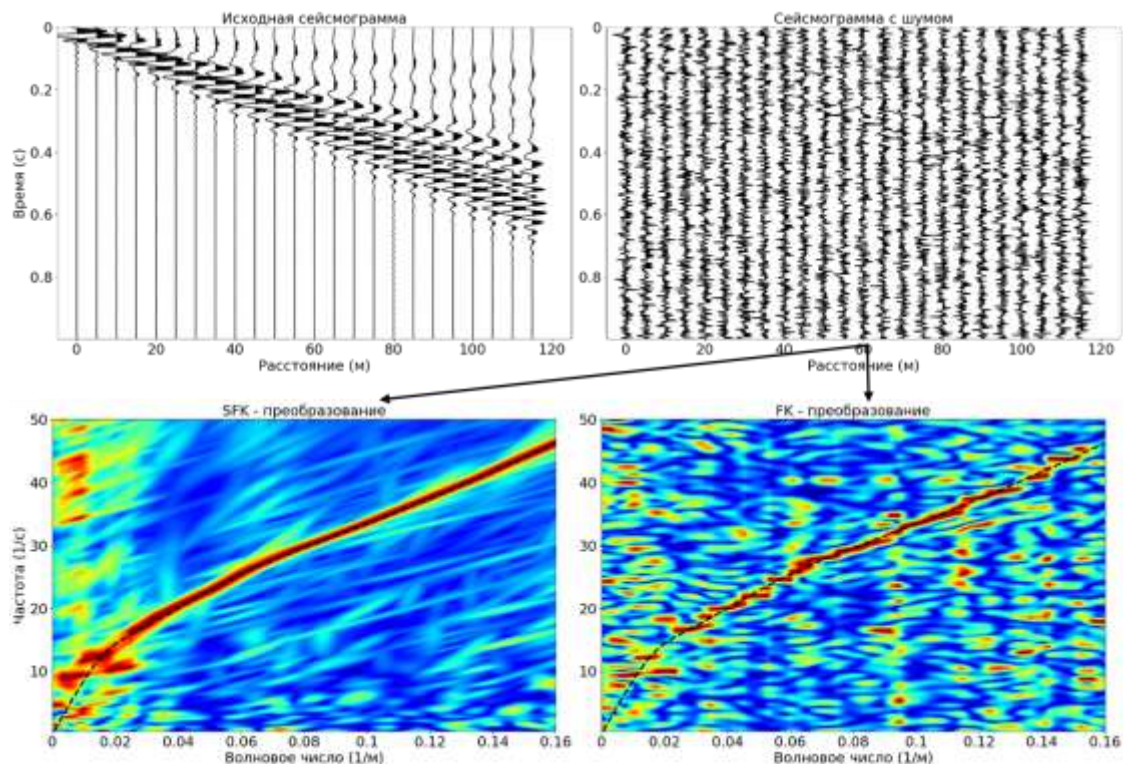


Рис. 10. Пример расчета FK-изображения зашумленной сейсмограммы (справа сверху) с помощью режимов SFK (слева внизу) и FKT (справа внизу).

Извлечение дисперсионных кривых

Извлечение дисперсионных кривых выполняется автоматически. Метод извлечения определяется параметром `extract_dc_method`:

- "max" — поиск максимума на каждой частоте в пределах заданных скоростей. Пределы скоростей задаются в файле, путь к которому указывается в параметре `path4dc_limits`.
- "dbscan" — кластеризация дисперсионного изображения, выбор наибольшего кластера и расчет средней кривой по точкам кластера. Дисперсионное изображение, как и при методе "max", ограничено пределами скоростей, заданными в файле, путь к которому указан в `path4dc_limits`. Кластеризация позволяет выделить области, соответствующие дисперсионным кривым, из фонового шума. Важно отметить, что кластеры разных мод сортируются и нумеруются последовательно по расстоянию от точки [0,0] в пространстве "фазовая скорость - частота". Эта информация может быть полезна при интерпретации результатов кластеризации и выборе интересующей дисперсионной моды.
- «АЕ» - извлечение дисперсионных кривых с помощью применения пред обученной сверточной нейронной сети типа Автоэнкодер. Дисперсионное изображение также ограничивается заданными пределами извлечения. Сверточная сеть сначала

извлекает маски, соответствующие сигналу фундаментальной моды, первой высшей моды и шуму. Затем в ходе регрессионного анализа строится кривая по индексам извлеченной маски.

Для точной настройки процесса кластеризации и последующего извлечения дисперсионных кривых используются следующие параметры:

- **peak_fraction**: Коэффициент, определяющий порог амплитуды, используемый при переходе от спектрального изображения к маске изображения. Этот параметр задает, какие амплитуды на каждой частоте из спектра будут включены в маску, используемую для кластеризации и пикировки. Значение задается в долях процентов от максимальной амплитуды на каждой частоте. Например, при $\text{peak_fraction} = 0.8$, в маску перейдут только те амплитуды, которые больше или равны 80% от максимальной амплитуды на данной частоте ($\geq \text{max_ampl} * 0.8$). Рекомендуется использовать $\text{peak_fraction} = 0.9$. Более высокие значения peak_fraction приводят к отбрасыванию слабых сигналов и фокусировке на наиболее выраженных пиках. Параметр также используется для оценки энергии вдоль извлеченной кривой, значения с меньшей энергией будут отброшены.
- **cutoff_fraction**: Коэффициент, использующийся при фильтрации незначительных кластеров после кластеризации. Определяет, кластеры какого размера по оси частот будут отброшены. Задается в долях процентов от размера максимального кластера по оси частот. Например, при $\text{cutoff_fraction} = 0.4$, останутся только те кластеры, у которых размер по оси частот (freq_size) больше или равен 40% от размера максимального кластера по оси частот ($\text{max_freq_size} * 0.4$). Рекомендуется использовать $\text{cutoff_fraction} = 0.4$. Этот параметр помогает отбросить небольшие, нерелевантные кластеры, которые могут быть результатом шума или артефактов, и сосредоточиться на основных дисперсионных модах.

Оптимальные значения peak_fraction и cutoff_fraction зависят от характеристик данных (уровень шума, выраженность дисперсии) и могут быть определены эмпирически путем итеративного анализа результатов спектрального анализа.

Параметр path4dc_limits задает абсолютный путь к табличному CSV-файлу с пределами извлечения фазовых скоростей для заданного набора частот. Пример такого файла представлен на рис. 11. Формат CSV-файла: три колонки, частота (freq), минимальное значение диапазона фазовой скорости (v_min), максимальное значение диапазона фазовой скорости (v_max). При определении кривых пределы частот автоматически ограничиваются значениями fmin и fmax . Пределы скоростей автоматически ограничиваются значениями vmin и vmax . Значения скоростей линейно интерполируются для всего набора частот в спектре.

После извлечения дисперсионные кривые автоматически сглаживаются, в них удаляются одиночные выбросы и не физические отклонения. Однако, даже после этих процедур не все извлеченные кривые являются достаточно гладкими и надежными. Для дополнительно отбраковки дисперсионных кривых используется параметр dc_error_thr – коэффициент, определяющий максимальный порог отличия между извлеченной дисперсионной кривой и аппроксимирующей ее кривой, полученной с помощью полинома четвертой степени. Этот параметр позволяет отбраковывать дисперсионные кривые, которые имеют значительные отклонения от гладкой формы. Метрика для оценки отклонения - MAPE (Mean Absolute Percentage Error) - средняя абсолютная процентная

ошибка. Для каждой извлеченной дисперсионной кривой вычисляется максимальное значение MAPE между кривой и ее аппроксимацией полиномом четвертой степени. Если это максимальное значение MAPE превышает заданный порог `dc_error_thr`, кривая отбраковывается. Например, при `dc_error_thr = 0.3`, будут отбракованы те кривые, у которых $\text{MAX}(\text{MAPE}(\text{кривая} - \text{полином четвертой степени})) > 0.3$. Рекомендуется использовать `dc_error_thr = 0.2`.

1	freq;v_min;v_max
2	1.5;250;1500
3	2;250;1300
4	3;225;1000
5	6;200;700
6	10;200;600
7	15;200;600

	freq	÷	v_min	÷	v_max	÷
1	1.5		250		1500	
2	2		250		1300	
3	3		225		1000	
4	6		200		700	
5	10		200		600	
6	15		200		600	

Рис. 11. Формат табличного csv-файла с пределами извлечения дисперсионных кривых, путь к которому задан в `path4dc_limits`. Слева, открытый в текстовом редакторе, справа, открытый библиотекой чтения csv-файлов.

На рис. 12 представлены примеры полученных дисперсионных изображений и извлеченных кривых для SFK и FKT подходов.

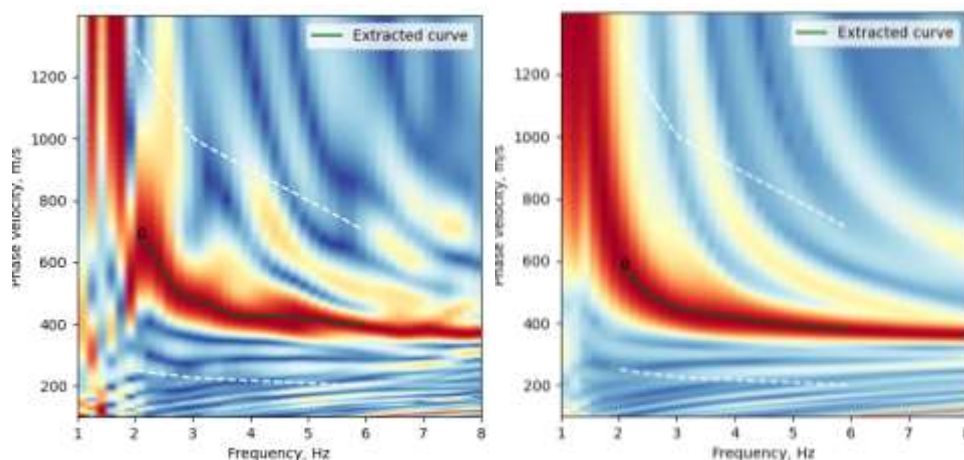


Рис. 12. Пример дисперсионного изображения для одной и той же сейсмограммы, полученного алгоритмами FKT (слева) и SFK (справа). Все управляющие параметры и диапазоны поиска заданы одинаковыми (`desired_nt: 3000`, `desired_nx: 1000`, `fmin: 1`, `fmax: 8`, `vmin = 100`, `vmax: 1400`, `extract_dc_method = dbscan`, `smooth_data: True`, `width: 0.6` (только SFK), `peak_fraction: 0.8`, `cutoff_fraction: 0.4`, `dc_error_thr: 0.2`)

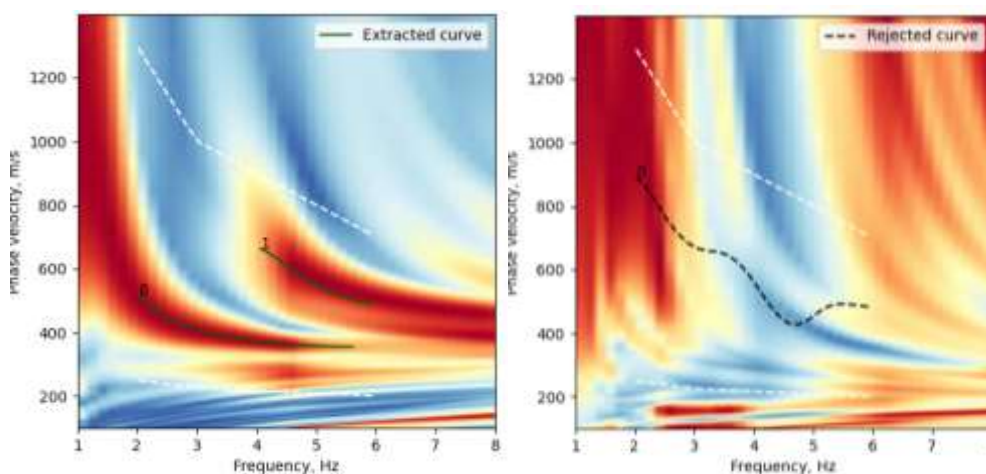


Рис. 13. Пример дисперсионного изображения с выделением двух мод поверхностной волны (слева). Пример дисперсионного изображения с отбракованной по гладкости кривой (справа).

Для контроля качества дисперсионные изображения сохраняются в формате segy и png по следующему пути:

- Для 3D: runs / 3d / spectral_analysis / <название директории с данными> / <num_sectors> или cdp / <spectral_method> / segy и image
- Для 2D: runs / 3d / spectral_analysis / <название директории с данными> / offsets_<min_offset_max_offset> / <spectral_method> / segy и image.

После завершения этапов препроцессинга и спектрального анализа все извлеченные дисперсионные кривые сохраняются в формате NPZ в директории по следующему пути:

- Для 3D: runs / 3d / spectral_analysis / <название директории с данными> / <num_sectors> или cdp / <spectral_method> / dc
- Для 2D: runs / 3d / spectral_analysis / <название директории с данными> / offsets_<min_offset_max_offset> / <spectral_method> / dc.

Параметры инверсии

На этом этапе параллелизация вычисления инверсии осуществляется для всего набора сохраненных дисперсионных кривых. Выбор оптимальных параметров инверсии критически важен для получения достоверных и детализированных скоростных моделей. Первоначальный выбор касается способа инверсии, который определяет алгоритм, используемый для поиска оптимальной скоростной модели. Определяется параметром `inversion_method`:

1. "оссам" – процедура локальной минимизации по алгоритму Оккама. В результате итеративного обновления начальной модели строится гладкая скоростная модель (только значения V_s). Начальная модель определяется автоматически по извлеченной дисперсионной кривой методом эмпирической инверсии. Более полное описание алгоритма Оккама и способа построения начальной модели см. в [приложении №3](#).

2. "GWO" – процедура глобальной минимизации по алгоритму GWO (Grey Wolf Optimizer). В результате определяется слоистая скоростная модель (значения V_s и мощности слоев). Диапазоны поиска могут определяться автоматически или задаваться вручную.
3. "SSA" – процедура глобальной минимизации по алгоритму SSA (Salp Swarm Algorithm). В результате стохастического подбора параметров скоростной модели в пределах диапазона поиска строится слоистая скоростная модель (значения V_s и мощности слоев). Диапазоны поиска могут определяться автоматически или задаваться вручную.

Алгоритмы глобальной оптимизации в значительной степени схожи между собой по сути описания и эффективности инверсии. В виду этого приводится подробное описание с результатами исследования только алгоритма SSA ([приложения №4](#)). Описание и исследование алгоритма GWO выполнено в следующей статье – [Song et al., 2015]. В [приложении №4](#) также дано описание способа автоматического определения диапазонов решения, который является общим для алгоритмов GWO и SSA.

В результате подбора значений мощностей алгоритмы глобального поиска предоставляют более “резкую”, блоковую по глубине модель, а алгоритм локального поиска предоставляет более гладкую по глубине модель.

Общие параметры инверсии

После выбора метода инверсии необходимо настроить общие параметры, не зависящие от выбранного алгоритма:

- `niter`: количество итераций для алгоритмов глобальной (SSA, GWO) и локальной (Оссам) инверсии. Рекомендуется использовать `niter = 80`. Увеличение количества итераций может повысить точность, но также увеличивает время вычислений.
- `max_num_modes`: максимальное число мод, которое будет использоваться при инверсии. Рекомендуется использовать `max_num_modes = 1`. Обычно достаточно подбора фундаментальной (основной) моды, но можно рассмотреть увеличение этого параметра при необходимости. Вовлечение высших мод в процесс инверсии должно уменьшать неоднозначность ее результатов.
- `wavetype`: тип поверхностной волны, используемой для инверсии:
 - "Rayleigh" – волна Релея.
 - "Love" – волна Лява. Примечание: Метод SWI применим к обработке поверхностных волн Лява, но требуется возбуждение и регистрация SH-волн. Рекомендуется использовать `wavetype = "Rayleigh"`.
- `veltype`: Тип скорости, используемой в дисперсионной кривой:
 - "Phase" – фазовая скорость.
 - "Group" – групповая скорость. Примечание: В текущей реализации SWI извлекаются только фазовые скорости, но запуск подбора кривых групповых скоростей также возможен. Рекомендуется использовать `veltype = "Phase"`.

Следующие параметры определяют использование дополнительной информации о скоростной модели Р-волн (V_p) или отношении V_p/V_s (необходимы при расчете дисперсионных кривых в процессе инверсии):

- `path4vp_model`: абсолютный путь к файлу со скоростной моделью V_p или отношением V_p/V_s . Возможны следующие опции:
 - SEG-Y-файл: скоростная модель V_p или V_p/V_s задается через SEG-Y-файл (с расширением “.seg” или “.sgy”). В этом случае необходимо указать шаг по глубине в `dt` (в метрах * 1000) и заголовки `CDP_X`, `CDP_Y`. Перед запуском инверсии выбирается ближайшая к дисперсионной кривой, полученной для точки [`cmp_x`, `cmp_y`], одномерная скоростная модель V_p или V_p/V_s из файла.
 - CSV-файл: задание одномерной модели V_p или V_p/V_s через табличный CSV-файл (константа для всей модели). Указывается путь к файлу с расширением “.csv”. Формат файла (рис. 14): три колонки - глубина (`depth`), значение скорости V_p (`vp`), значение отношения V_p/V_s (`vp2vs`). Примечание: В ходе инверсии заданная модель V_p или V_p/V_s интерполируется на глубины каждой восстанавливаемой скоростной модели V_s .
- `vp_model`: Тип задаваемой модели:
 - “vp” – скорость Р-волны.
 - “vp2vs” – отношение скорости Р-волны к скорости S-волны. Примечание: Определяет тип значения, читаемого из SEG-Y или CSV файла.
- `lock_vp`: Флаг, определяющий, фиксировать ли при инверсии скоростную модель V_p , задаваемую в SEG-Y или CSV файле, или нет. True: скоростная модель V_p , заданная в SEG-Y или CSV файле, будет применяться для ограничения подбираемых скоростных моделей V_s на каждой итерации. False: скоростная модель V_p , заданная в SEG-Y или CSV файле, будет использована для вычисления отношения `vp2vs` для начальной модели (локальная оптимизация) или диапазонов поиска (глобальная оптимизация), которое затем используется для вычисления V_p из восстановленной V_s на остальных итерациях. Рекомендация: Если нет уверенности в модели V_p , то использовать `lock_vp = False`.

1	depth;vp;vp2vs				
2	0;400;4				
3	20;800;5				
4	60;1200;2				
5	80;2000;2				

	depth	vp	vp2vs
1	0	400	4
2	20	800	5
3	60	1200	2
4	80	2000	2

Рис. 14. Формат табличного csv-файла с информацией о скоростной модели Р-волн или отношении V_p/V_s , путь к которому задан в `path4vp_model`. Слева, открытый в текстовом редакторе, справа, открытый библиотекой чтения csv-файлов.

Специальные параметры инверсии

В зависимости от выбранного метода инверсии, необходимо настроить специальные параметры для каждого алгоритма.

Параметры для алгоритма локальной оптимизации Оккама (`inversion method = “occam”`):

- `nlay`: количество слоев для гладкой скоростной модели при инверсии Оккама. Рекомендуется использовать `nlay = 10`. Увеличение числа слоев позволяет получить более детализированную, но и более сложную модель.

Для алгоритмов глобальной оптимизации GWO и SSA (`inversion_method = "GWO"` или `"SSA"`) используется группа параметров, объединенных в подгруппу `global_search`:

- `test_count`: число независимых запусков решения обратной задачи алгоритмами глобального поиска для одной дисперсионной кривой. Результирующая модель определяется как медианная средняя для каждого распределения восстановленного параметра (скорости и мощности в слое). Примечание: Повышение параметра `test_count` снижает неоднозначность инверсии, но многократно увеличивает время решения. Рекомендация: В зависимости от имеющихся ресурсов выбирать `test_count` в пределах 1 - 20.
- `path4vs_limits`: путь к CSV-файлу с заданными диапазонами поиска скоростей S -волны и мощностей слоев при инверсии. Формат таблицы в файле (рис. 15) – 5 колонок: `Vs_min`, `Vs_max`, `h_min`, `h_max`, `vp2vs` (минимальное значение V_s в слое, максимальное значение V_s в слое, минимальная мощность слоя, максимальная мощность слоя, значение отношения V_p/V_s). Мощности последнего слоя задаются нулями. Количество строк – число слоев. Примечание: параметр `path4vs_limits` может быть задан пустым, тогда диапазоны определяются автоматически с использованием параметра `xi`.
- `xi`: параметр слоистости, влияющий на автоматическое определение количества слоев в алгоритмах глобальной оптимизации. Чем он меньше, тем больше число слоев для глобальной инверсии. Увеличение числа слоев также повышает сложность расчета инверсии. Рекомендация: `xi = 2`.
- `get_num_layers`: способ автоматического определения числа слоев и диапазонов по дисперсионным кривым для инверсии методом глобального поиска (необходим если `path4vs_limits` задан пустым):
 - "every" - для каждой кривой своё число слоев и свои диапазоны поиска. Рекомендация: `get_num_layers = "every"`.
 - "mean" - рассчитывается средняя дисперсионная кривая для всего набора и задается только одна слоистая модель (число слоев и диапазоны поиска).
 - "classes" - набор кривых разбивается на некоторое число классов (согласно условию, что классы достаточно различаются друг от друга по заданной метрике), и диапазоны и число слоев рассчитываются в пределах одного класса.

1	<code>vs_min;vs_max;h_min;h_max</code>	<code>vs_min</code>	<code>vs_max</code>	<code>h_min</code>	<code>h_max</code>
2	<code>100;500;1;5</code>	100	500	1	5
3	<code>200;800;8;10</code>	200	800	8	10
4	<code>300;1000;0;0</code>	300	1000	0	0
5					

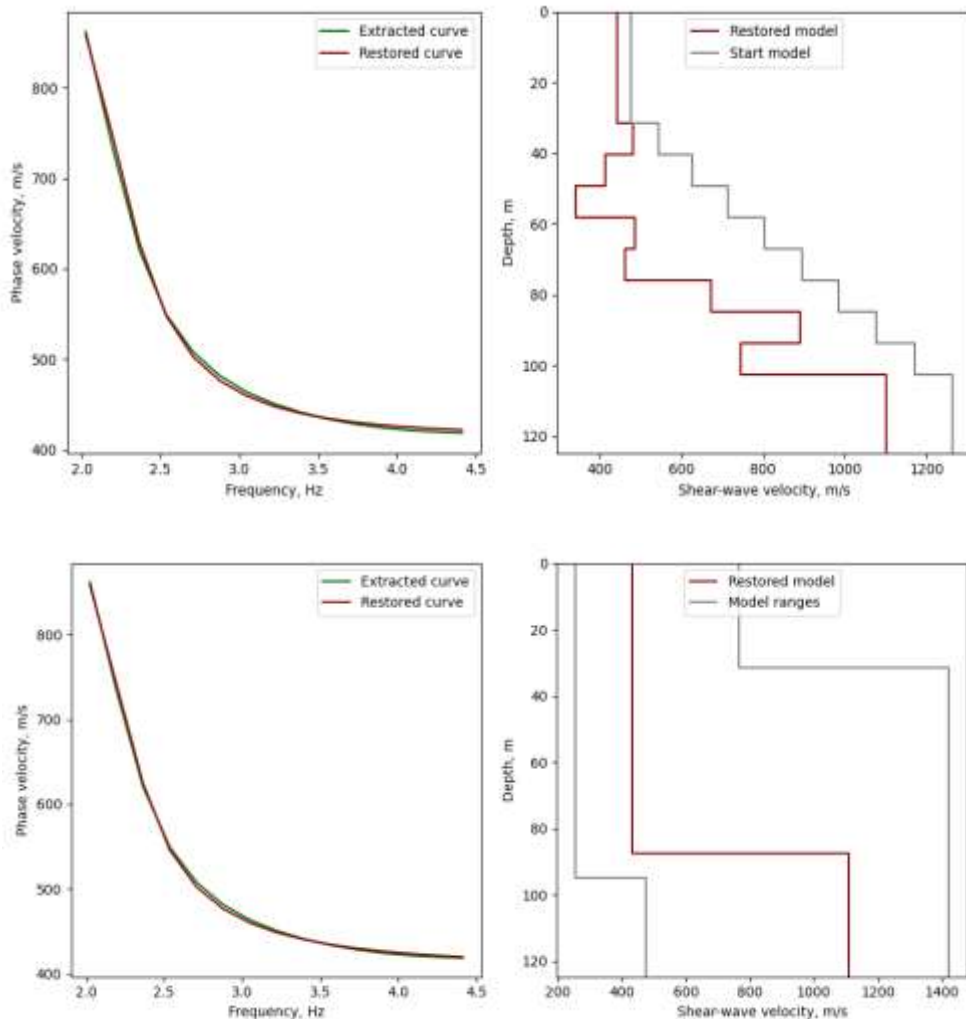
Рис. 15. Формат табличного csv-файла с заданными диапазонами поиска скоростей S -волны и мощностей слоев при инверсии. Слева, открытый в текстовом редакторе, справа, открытый библиотекой чтения csv-файлов.

Выбор оптимальных значений параметров для алгоритмов GWO и SSA требует баланса между точностью и вычислительными затратами. Использование файла `path4vs_limits` позволяет задать более точные ограничения на диапазоны поиска, что может улучшить результаты инверсии, но требует предварительной информации о геологическом строении. Автоматическое определение числа слоев и диапазонов с помощью параметра `xi` и `get_num_layers` удобно, если такой информации нет, но может потребовать большего времени вычислений.

Для контроля качества процесса инверсии, изображения восстановленных скоростных моделей и соответствующих дисперсионных кривых сохраняются в формате PNG. Расположение этих изображений зависит от типа данных (2D или 3D) и выбранных параметров обработки:

- 3D данные: `runs / 3d / inversion / <название директории с данными> / <num_sectors>` или `cdp / <spectral_method> / <inversion_method> / image`
- 2D данные: `runs / 3d / inversion / <название директории с данными> / offsets_<min_offset_max_offset> / <spectral_method> / inversion_method / image`

На рис. 16 представлены примеры восстановления одной дисперсионной кривой различными алгоритмами. Параметры для SSA и GWO заданы одинаковыми (`test_count`: 1, `path4vs_limits`: None, `xi`: 2 `get_num_layers`: every), для алгоритма Оккама `nlay`: 10.



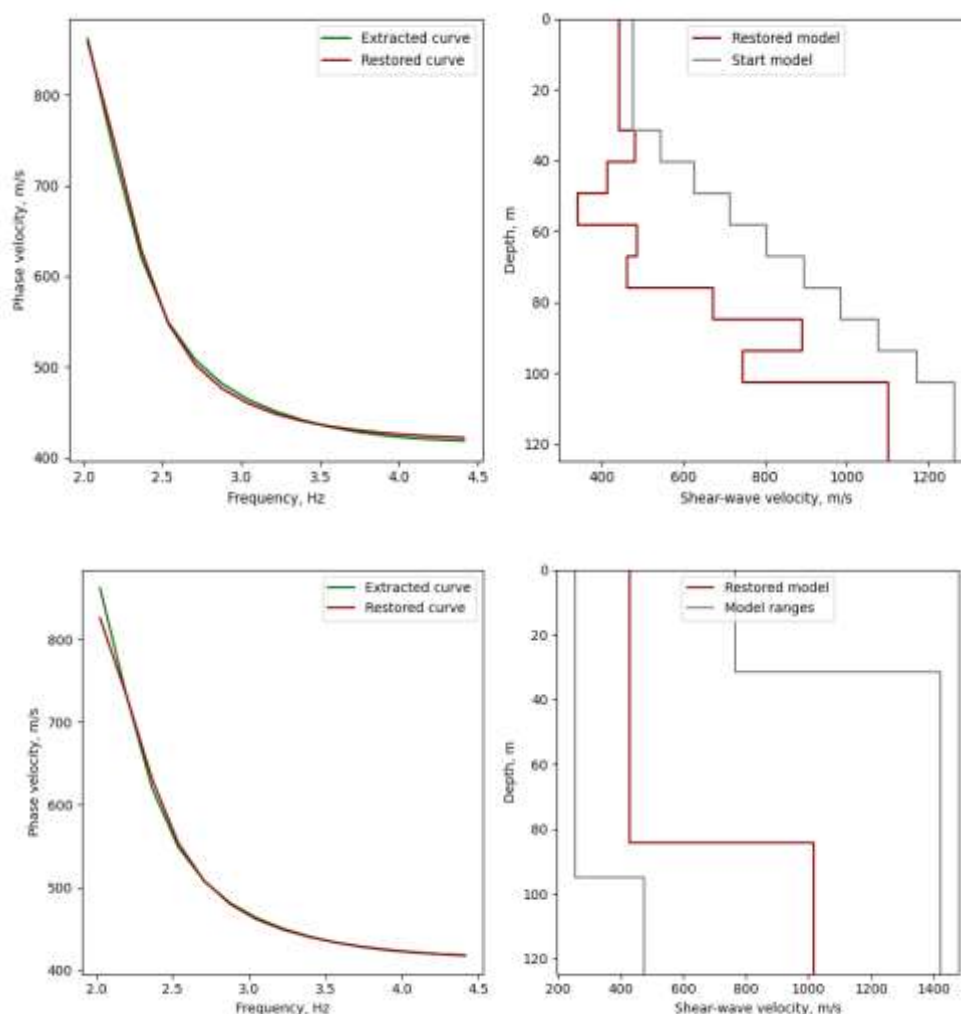


Рис. 16. Восстановленная дисперсионная кривая и скоростная модель S-волны различными методами инверсии: Оккама (сверху), GWO (в центре) и SSA (снизу). Серыми линиями показана стартовая модель для Оккама и диапазоны поиска для GWO и SSA.

Для дополнительного контроля качества и оценки результатов инверсии, модуль также сохраняет распределение (гистограмму) ошибок. Эта гистограмма отображает распределение ошибок между каждой парой: извлеченной дисперсионной кривой и восстановленной скоростной моделью.

Ошибка между извлеченной и восстановленной кривой вычисляется как $\text{MAX}[\text{MAPE}(\text{извлеченная кривая} - \text{восстановленная кривая})]$, где MAPE (Mean Absolute Percentage Error) - средняя абсолютная процентная ошибка, рассчитываемая на каждой частоте. Таким образом, для каждой дисперсионной кривой вычисляется максимальное значение MAPE, характеризующее степень соответствия между извлеченной и восстановленной кривыми (рис. 17). Анализ этой гистограммы позволяет оценить общую точность и надежность результатов инверсии, выявить возможные проблемы с подбором моделей, а также оценить качество инверсии для всего набора данных.

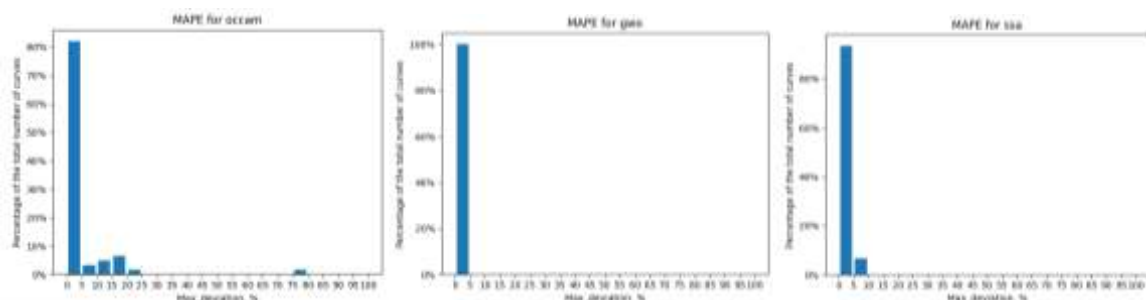


Рис. 17. Примеры распределения максимальной MAPE между извлеченными и восстановленными дисперсионными кривыми для различных алгоритмов инверсии.

Параметры постпроцессинга

Этап постпроцессинга предназначен для обработки результатов инверсии с целью получения интерполированной 2D/3D скоростной модели, пригодной для дальнейшего анализа и интерпретации. Он включает в себя отбраковку скоростных моделей, бинирование, интерполяцию, сглаживание, заполнение пропусков, сохранение результатов в форматах (SEG-Y и FDM) и визуализацию для контроля качества. Корректная настройка параметров постпроцессинга позволяет получить конечную скоростную модель, которая соответствует требованиям к точности, детализации и удобству представления данных. Визуализация результатов постпроцессинга (в 2D и 3D) позволяет контролировать качество интерполяции, сглаживания и заполнения пропусков, а также выявлять возможные артефакты и неточности.

Интерполяция по глубине и пространству

Первый шаг постпроцессинга - исключение скоростных моделей с ошибкой $\text{MAX}[\text{MAPE}]$ между извлеченной и восстановленной дисперсионной кривой, порог которой задан параметром `error_thr` (в долях процентов). Отброшенные модели проявятся как пропуски после интерполяции (если `fill_missing_values = False`). Рекомендация: `error_thr = 0.2`.

На следующем шаге каждая 1D скоростная модель (как для 2D, так и для 3D) одномерной интерполяцией приводится к одинаковой параметризации по глубине на заданный шаг `d_z` (в метрах). Максимальная глубина в метрах ограничивается параметром `max_depth`. Восстановленные скоростные модели, имеющие большую глубину, будут обрезаны до `max_depth`. Модели с меньшей глубиной будут продолжены последней константной скоростью до `max_depth`. Рекомендация: Определять `max_depth` по изображениям восстановленных скоростных моделей после инверсии.

Далее, восстановленные скоростные модели бинируются и интерполируются на заданные:

- `d_x`: шаг в пространственном направлении X в метрах. Рекомендация: выбирать `d_x` в зависимости от числа секторов в 3D в пределах $\frac{1}{4}$ - $\frac{1}{2}$ от размера шага между источниками в направлении X. В 2D определять `d_x` как $\frac{1}{2}$ от размера шага между источниками в направлении X.
- `d_y`: шаг в пространственном направлении Y в метрах. Рекомендация: В зависимости от числа секторов в 3D, выбирать `d_y` в пределах $\frac{1}{4}$ - $\frac{1}{2}$ от размера шага

между источниками в направлении Y . В 2D определять d_y как $\frac{1}{2}$ от размера шага между источниками в направлении Y .

Для 3D-случая, модели линейно интерполируются по глубинным слайсам с помощью двумерной интерполяции в направлении X - Y . В этом методе значение в каждой точке новой регулярной сетки вычисляется как взвешенная сумма значений в ближайших известных точках. Веса определяются на основе расстояния между точкой интерполяции и известными точками. Таким образом, на каждом глубинном слайсе создается регулярная сетка с шагами d_x и d_y , и для каждой точки этой сетки вычисляется значение скорости S -волн с использованием линейной интерполяции на основе известных значений скорости в ближайших точках. Использование линейной интерполяции обеспечивает достаточно гладкую и непрерывную скоростную модель, при этом не требуя значительных вычислительных ресурсов. Выбор шагов d_x и d_y позволяет контролировать разрешение конечной модели и адаптировать ее к особенностям исходных данных и геологическим задачам.

Для 2D возможны две опции, определяемые параметром `interp_dim`:

- "1d": интерполяция осуществляется только в направлении вдоль профиля.
- "2d": применяется двумерная интерполяция в направлении вдоль профиля и глубины.

Выбор между "1d" и "2d" зависит от характеристик обрабатываемых данных и поставленных геологических задач. Если ожидаются плавные изменения скоростей вдоль профиля, "1d" может быть достаточным. Если же предполагаются значительные изменения скоростей по глубине, "2d" может обеспечить более точную модель. Рекомендация: `interp_dim = 1d`.

Сглаживание скоростной модели

После интерполяции 2D модели сглаживаются в направлении X и Z , 3D модели сглаживаются в направлении X и Y . Степень гладкости интерполированной скоростной модели контролируется параметром `smooth_factor`. При увеличении коэффициента увеличивается гладкость. Рекомендация: `smooth_factor = 10`.

Применяемый алгоритм сглаживания [Garcia, 2010] основан на методе штрафovaných наименьших квадратов (penalized least squares method) и обеспечивает быстрое сглаживание данных в двух измерениях с помощью дискретного косинусного преобразования (discrete cosine transform). Благодаря использованию дискретного косинусного преобразования, алгоритм обеспечивает быстрое сглаживание даже для больших объемов данных.

Автоматический выбор степени сглаживания (если `smooth_factor` = пустое поле) осуществляется путем минимизации обобщенного критерия перекрестной проверки (generalized cross-validation score). Автоматический выбор степени сглаживания устраняет необходимость ручной настройки параметров.

Для обработки пропущенных и выброшенных значений реализована итеративно взвешенная устойчивая версия алгоритма (iteratively weighted robust version). Это часть алгоритма включается флагом `remove_outliers_smoothing`. Рекомендация: `remove_outliers_smoothing = True`. Итеративно взвешенная устойчивая версия алгоритма позволяет эффективно обрабатывать пропущенные и выброшенные значения (после

отбраковки по MAX[MAPE]). Алгоритм сглаживания также способен заполнить пропуски в данных (которые проявляются при отбраковке моделей с большой невязкой по дисперсионным кривым) на основе анализа соседних точек. Эта опция включена при `fill_missing_values = True`.

Обрезка площади и заполнение рельефа

Если задан отдельный файл с координатами полигона, то результирующий куб скоростной модели будет обрезан по этому полигону. Файл полигона площади для 3D в текстовом (.txt) формате 2 столбца: x, y. Если файл не задан то модель будет обрезана по рассчитаному выпуклому полигону.

Рельеф интерполируется на точки скоростной модели из значений, заданных в отдельной файле рельефа (обычно это рельеф в точках ПВ и ПП) в текстовом (.txt) формате 3 столбца: x, y, z. В отсутствии файла задается нулевой рельеф.

При необходимости, можно масштабировать значения рельефа с помощью параметра `scaler_to_elevation`. При этом, применяется следующая логика:

- Отрицательное значение коэффициента: значения рельефа делятся на абсолютное значение коэффициента.
- Положительное значение коэффициента: значения рельефа умножаются на коэффициент.
- Нулевое значение коэффициента: масштабирование не выполняется.

Форматы сохранения скоростной модели

На заключительном этапе постпроцессинга, полученная скоростная модель может быть сохранена в форматах SEG Y и FDM:

`save_seg y`: флаг, определяющий, сохранять ли скоростную модель в SEG Y-файл. Рекомендация: `save_seg y = True`.

`save_fdm`: флаг, определяющий, сохранять ли скоростную модель в FDM-файл. Рекомендация: `save_fdm = True`.

Описание формата SEG Y-файла. Сохранение в SEG Y-файл осуществляется с помощью библиотеки `seg yio`, которая задает спецификации файла, включая количество отсчетов, количество трасс, тип сортировки и формат данных:

- Шаг по глубине (`d_z`) записывается в бинарный заголовок SEG Y-файла в раздел “sample interval” (или `dt`) как $1 \text{ м} * 1000$ (метр умноженный на 1000, чтобы указать интервал в микросекундах).
- Регулярная сетка координат записывается в заголовках `CDP_X` и `CDP_Y`, а отметки рельефа в заголовке `ReceiverGroupElevation`.

Сохранение в SEG Y-файл позволяет представить скоростную модель в стандартном формате, совместимом с большинством программных пакетов для обработки и интерпретации сейсмических данных. Координаты и отметки рельефа сохраняются в заголовках трасс, что позволяет правильно позиционировать модель в пространстве.

Файл формата FDM состоит из двух основных частей: заголовка и данных скоростной модели.

1. Заголовок (Header): 512-байтный заголовок, содержащий 128 значений типа float32.

Хранит метаданные, описывающие геометрию и параметры скоростной модели:

x_size: Количество ячеек (бинов) по оси X (соответствует crossline).

y_size: Количество ячеек (бинов) по оси Y (соответствует inline).

z_size: Количество ячеек (бинов) по оси Z (соответствует глубине/времени).

x_step: Физический размер бина (шаг дискретизации) в метрах по оси X.

y_step: Физический размер бина (шаг дискретизации) в метрах по оси Y.

z_step: Физический размер бина (шаг дискретизации) в метрах по оси Z.

cfst: Номер первого crossline (обычно начинается с 1.0).

sfst: Номер первого inline (обычно начинается с 1.0).

cinc: Инкремент (шаг) номеров crossline (обычно 1.0).

sinc: Инкремент (шаг) номеров inline (обычно 1.0).

dist_unit: Код, указывающий единицы измерения расстояния (1=метры, 2=футы и т.д.).

angle_unit: Код, указывающий единицы измерения углов (1=градусы, 2=радианы и т.д.).

north_angle: Неиспользуемый параметр (предположительно, PI=3.141592...).

rot_angle: Угол поворота системы координат в радианах.

utm_x: UTM координата X (восточная) центра бина (1,1).

utm_y: UTM координата Y (северная) центра бина (1,1).

Все поля заголовка записываются как 32-битные числа с плавающей точкой (float32). UTM координаты хранятся в виде целой и дробной частей, начиная со смещения 96 байт.

2. Данные скоростной модели (Velocity Model Data):

Данные: 3D массив значений скоростей, представленных в формате float32 (4 байта на значение). Порядок хранения: X, Y, Z, то есть глубина (Z) меняется быстрее всего, затем inline (Y), и crossline (X) медленнее всего. Единицы измерения: метры в секунду (м/с).

Данный формат обеспечивает компактное хранение 3D скоростных моделей и содержит всю необходимую информацию для корректной загрузки и использования модели в программах сейсмического моделирования и обработки данных.

Встроенная визуализация скоростной модели

Встроенная визуализация скоростной модели выполняется средствами Python (библиотеки matplotlib и plotly). Изображения скоростной модели сохраняются в 2D пространстве по срезам XZ, YZ и XY, а также в 3D пространстве по срезам XZ и YZ. Количество срезов по каждой оси задается параметрами: num_xslices_3d, num_yslices_3d, num_zslices_3d. Пределы цветовой шкалы (скорости S-волны) регулируются двумя параметрами: vmin_in_model - минимальная скорость в цветовой шкале и vmax_in_model - максимальная скорость в цветовой шкале.

Далее на рис. 18 - 21 представлены примеры скоростной модели, полученной для инверсии Оккама и GWO для одних и тех же дисперсионных кривых и изображенной в различных плоскостях.

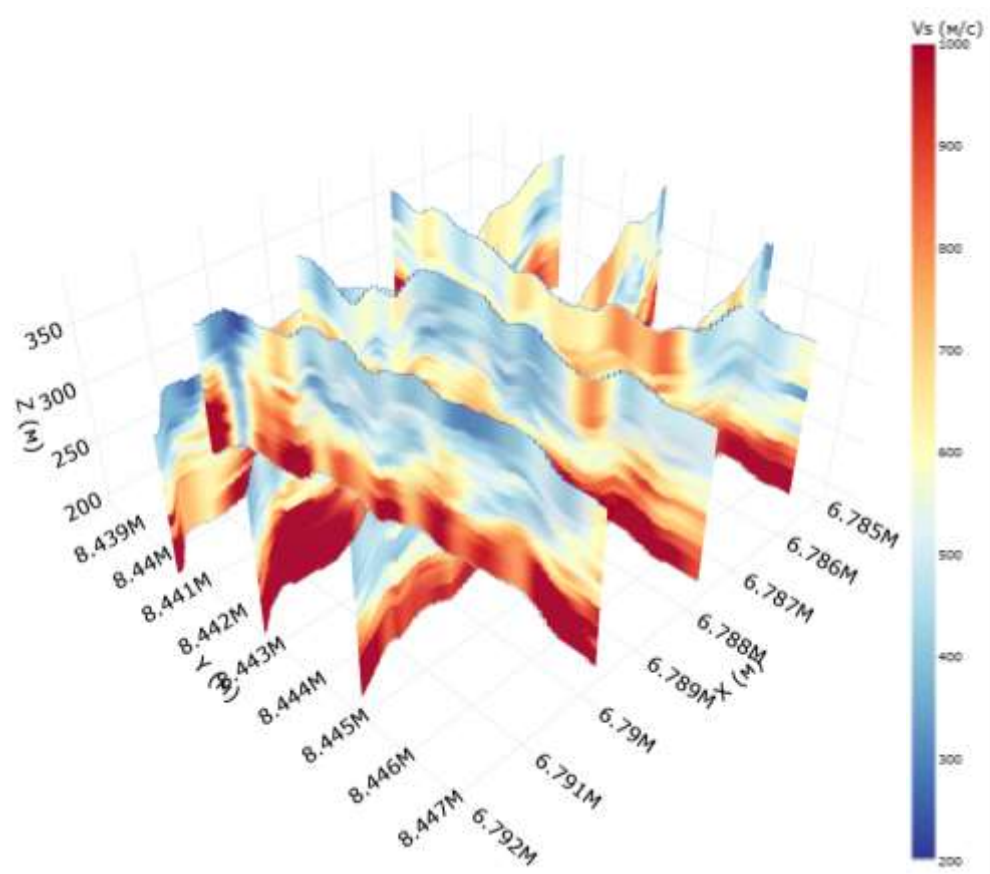
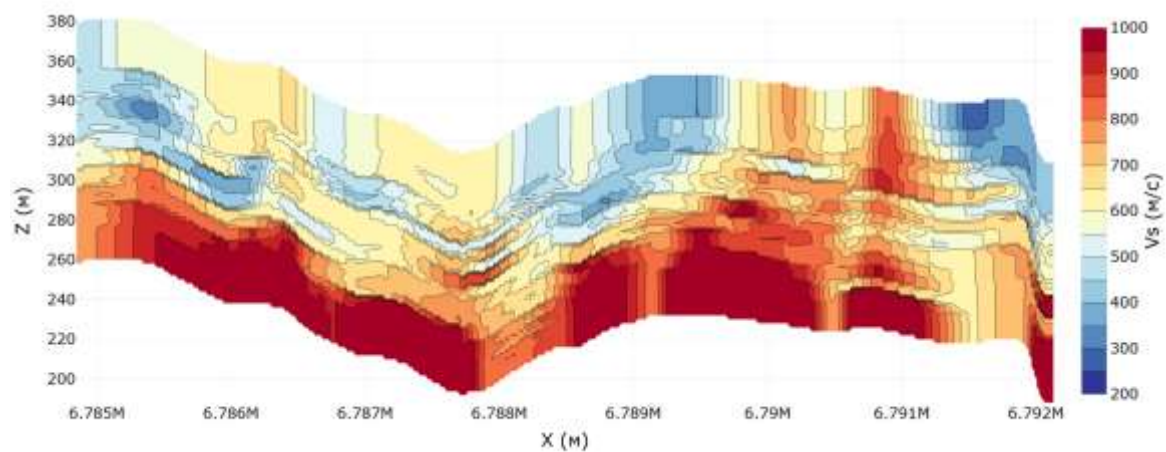


Рис. 18. Визуализация 3D скоростной модели в 3D-пространстве: 3 среза по X и Y. Метод инверсии - алгоритм Оккама (max_depth: 120, d_x: 20, d_y: 100, d_z: 2, smooth_factor: 1, vmin_in_model: 200, vmax_in_model: 1000).



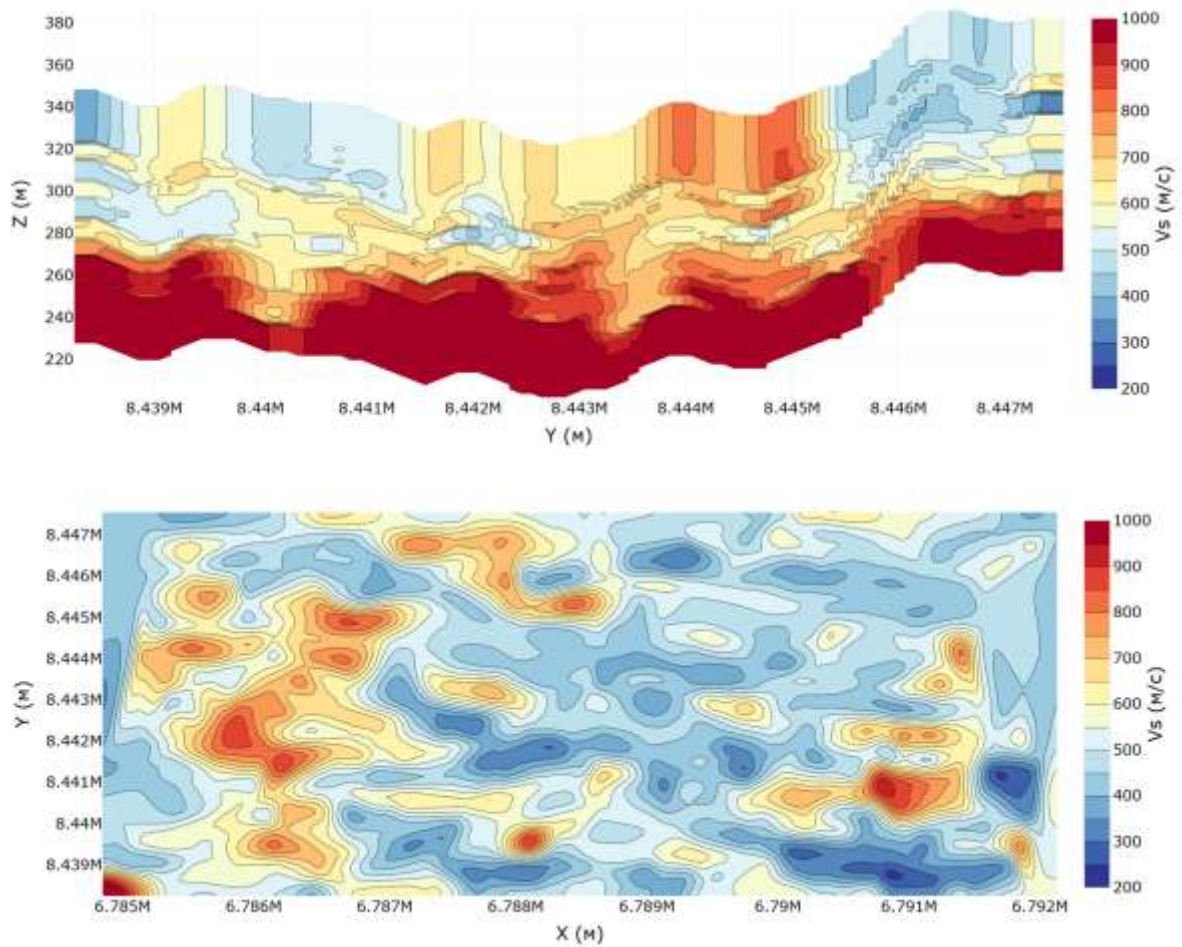


Рис. 19. Визуализация 3D скоростной модели в 2D-пространстве: срез XZ (сверху), срез по YZ (в центре) и срез по XY (снизу). Метод инверсии - алгоритм Оккама (max_depth: 120, d_x: 20, d_y: 100, d_z: 2, smooth_factor: 1, vmin_in_model: 200, vmax_in_model: 1000).

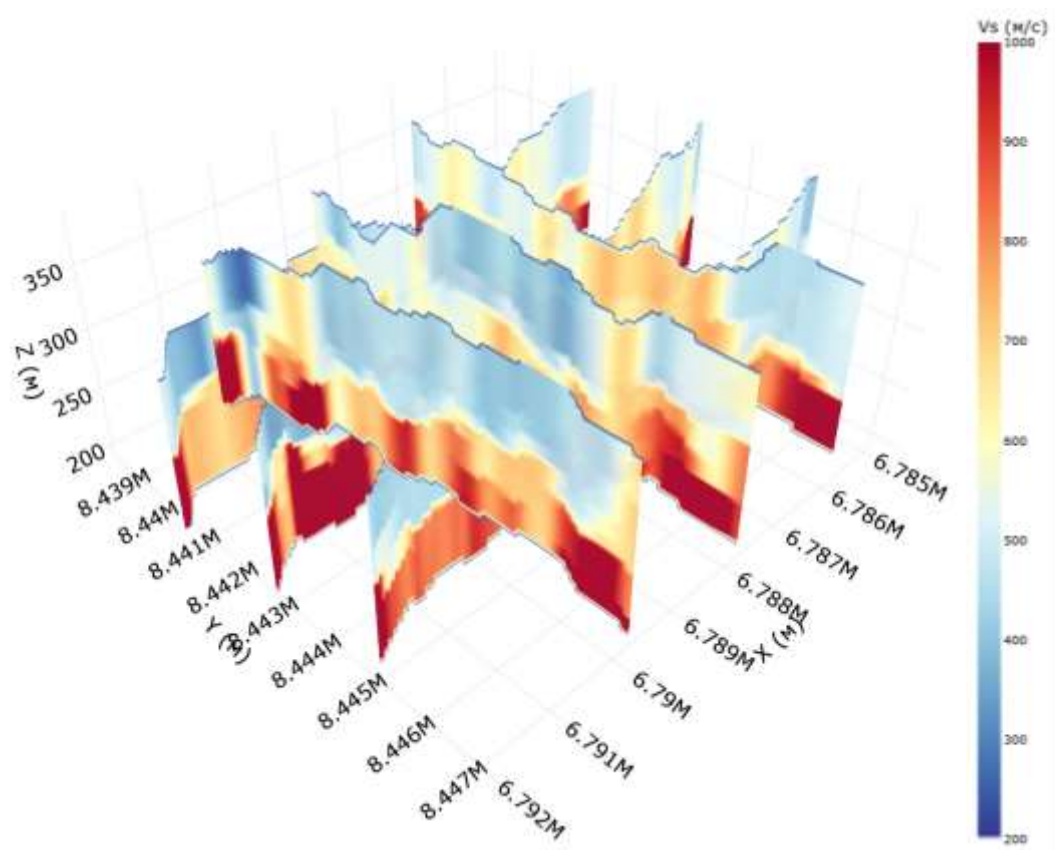
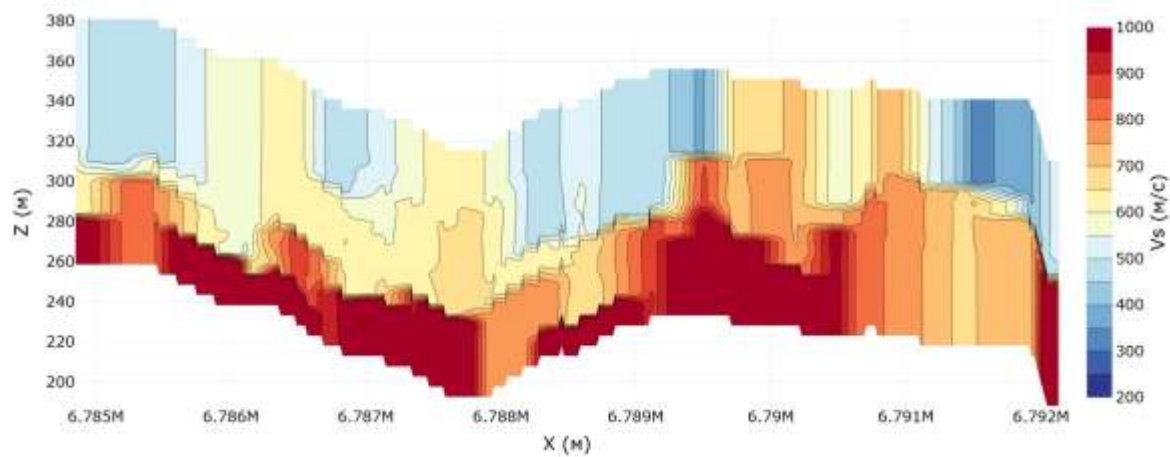


Рис. 20. Визуализация 3D скоростной модели в 3D-пространстве: 3 среза по X и Y.
 Метод инверсии - алгоритм GWO (max_depth: 120, d_x: 20, d_y: 100, d_z: 10,
 smooth_factor: 1, vmin_in_model: 200, vmax_in_model: 1000).



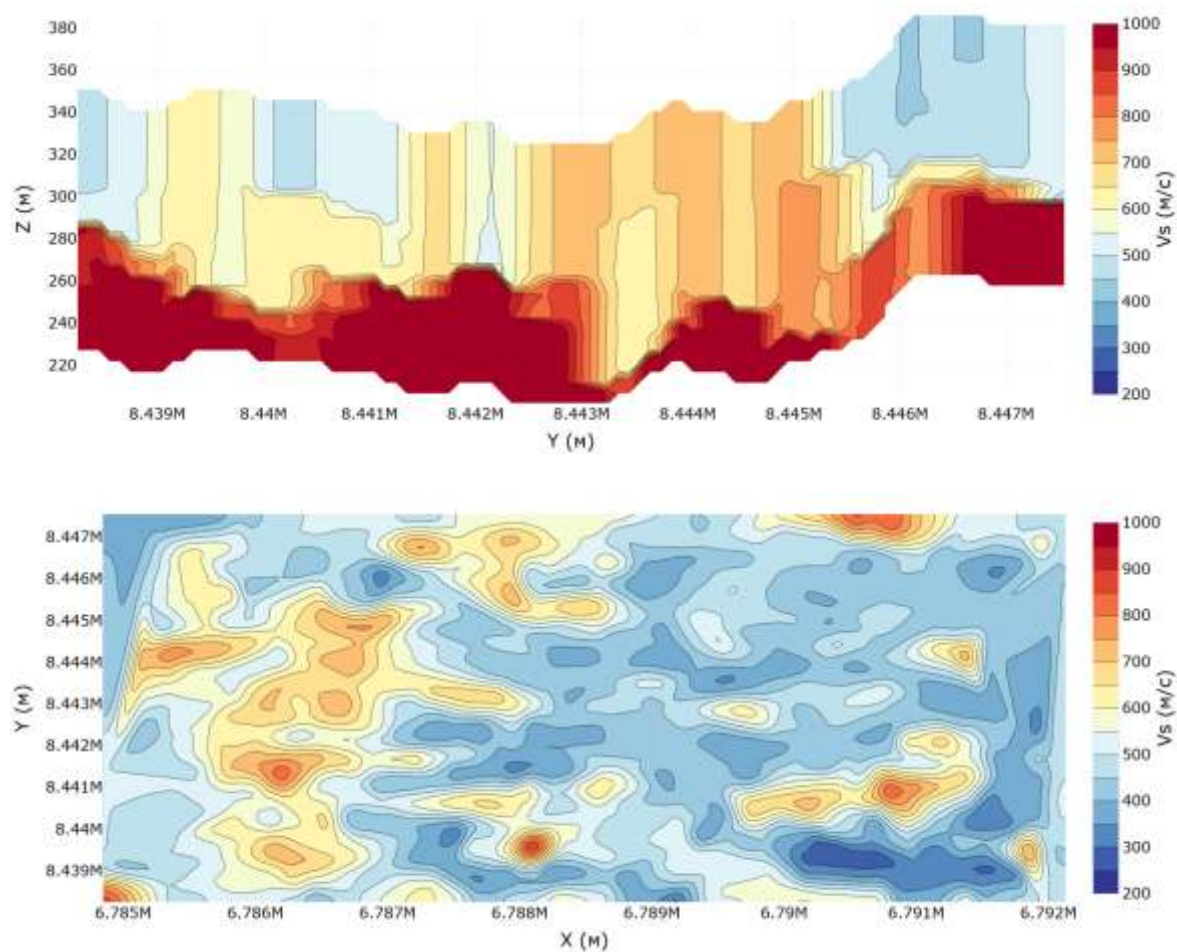


Рис. 21. Визуализация 3D скоростной модели в 2D-пространстве: срез XZ (сверху), срез по YZ (в центре) и срез по XY (снизу). Метод инверсии - алгоритм GWO (max_depth: 120, d_x: 20, d_y: 100, d_z: 2, smooth_factor: 1, vmin_in_model: 200, vmax_in_model: 1000).

Приложение №1. Таблица управляемых параметров

Абсолютный путь к конфиг-файлу задается отдельно перед запуском каждого подмодуля.

Название параметра в конфигурационном yaml-файле	Описание параметра	Принимаемые значения, по которым осуществляется проверка параметра
Препроцессинг		
path4ffid_file	<p>Абсолютный путь к текстовому файлу с FFID для каждой трассы.</p> <p>Необходим для ускорения чтения списка с FFID каждой трассы в случае использования входных файлов формата SEG-Y.</p> <p>Если не задан, то FFID извлекается из segy-файла с данными или набора данных СБД.</p> <p>Формат файла: в две колонки номер трассы и значение FFID.</p> <p><u>Примечание:</u> Если путь задан, то проверяется наличие файла.</p>	<p>String</p> <ol style="list-style-type: none"> Доступные значения для наименования директории. None
ffid_start	FFID (CDP) первого источника, с которого начинается обработка	<p>Integer</p> <p>ffid_start >= 0</p>
ffid_stop	FFID (CDP) последнего источника, который будет обработан.	<p>Integer</p> <p>ffid_stop > ffid_start</p>
ffid_increment	<p>Шаг между FFID (CDP), используемыми для обработки.</p> <p><u>Примечание:</u> Если ffid_increment == 0, то используются все доступные FFID для обработки, ffid_start и ffid_stop игнорируются.</p>	<p>Integer</p> <p>Range: >= 0</p>
num_sources_on_cpu	Количество FFID (CDP), которые будут обрабатываться на одном ядре процессора.	Integer

	<p>Число задействованных ядер CPU = $\text{length}([\text{ffid_start}:\text{ffid_increment}:\text{ffid_stop}]) // \text{num_sources_on_cpu}$</p> <p>Например, если вы обрабатываете 100 источников, $\text{ffid_increment}=1$, $\text{ffid_start}=1$, $\text{ffid_stop}=100$, и $\text{num_sources_on_cpu} = 10$, то будет использовано 10 ядер.</p> <p><u>Примечание:</u> Если $\text{num_sources_on_cpu} == 0$, то распаралеливание отключено, используется 1 ядро на все источники.</p>	<p>$\text{num_sources_on_cpu} \geq 0$</p>
data_dir	<p>Путь к директории с данными, файлу данных или идентификатор СБД.</p> <p>Набор данных СБД описывается в виде <code>sds:PROJECT/IDENT/VERSION</code></p> <p><u>Примечание:</u> Если в директории находится несколько файлов (ожидается что один файл - это один профиль 2Д или одна площадь 3Д), то файлы будут обработаны последовательно с текущими конфигурационными параметрами.</p>	<p>String</p> <p>Доступные значения для наименования директории.</p> <p><u>Примечание:</u> Проверяется наличие директории по заданному пути.</p>
scaler_to_coordinates	<p>Масштабирующий коэффициент для значений координат источников и приемников. Если отрицательный, то значения координат делятся на <code>scaler_to_coordinates</code>, если положительный то умножаются, если нуль, то координаты не масштабируются.</p>	<p>Float</p>
type_data	<p>Тип данных: 2D или 3D</p>	<p>String</p> <ol style="list-style-type: none"> 1. «2d» 2. «3d»
offset_min	<p>Минимальная длина базы наблюдения для обрезки трасс.</p> <p><u>Рекомендация:</u> задавать <code>offset_min ~</code></p>	<p>Float</p> <p>$\text{offset_min} \geq 0$.</p>

	offset_max*0.2	
offset_max	<p>Максимальная длина базы наблюдения для обрезки трасс.</p> <p><u>Примечание:</u> В методе SWI база наблюдения = (offset_max-offset_min).</p> <p>Результат инверсии относится к середине расстановки (common mid point) [cmp_x, cmp_y] = (offset_max-offset_min)/2. В 3В это геометрический центр сектора.</p> <p><u>Рекомендация:</u> offset_max задавать равным максимальному расстоянию визуально отслеживаемого сигнала поверхностной волны на исходных данных.</p>	<p>Float</p> <p>offset_max > offset_min.</p>
snr	<p>Отношение сигнал / помеха, ниже которого сейсмограммы отбраковываются до этапа спектрального анализа. Значение сигнала вычисляется как максимум нормированной функции взаимной корреляции (ФВК) между соседними трассами, ограниченными значениями vmin и vmax. Помеха вычисляется как 1 - максимум (ФВК). Такой расчет выбран в виду необходимости отбраковки сейсмограмм, не содержащих регулярный коррелируемый сигнал поверхностной волны.</p> <p><u>Примечание:</u> Значения snr всех сейсмограмм сохраняются в логах после препроцессинга</p> <p><u>Рекомендация:</u> выбирать значение snr на этапе подбора параметров для части данных по визуальному анализу сейсмограмм и значениям snr из лог-файла.</p>	<p>Float,</p> <p>snr > 0</p>
qc_preprocessing	<p>Флаг, определяющий сохранять ли нарезанные по базам наблюдения сейсмограммы в segy-файлы.</p> <p><u>Примечание:</u> В 2D имя нарезанных</p>	<p>Bool,</p>

	сейсмограмм формируется из названия профиля, координат точки привязки результата и флага ветви сейсмограммы (0 - левая ветвь, 1 - правая ветвь). В 3D и режиме sort_3d_order = csp имя формируется из названия профиля, номера FFID и номера сектора. В 3D и режиме sort_3d_order = cdp имя формируется из названия профиля, номера CDP.	True / False
parameters_3d: sort_3d_order	(Только для 3d) Тип сортировки данных 3D. «CSP» – сортировка по ОТВ (common shot point) «CDP» – сортировка по ОГТ (common depth point) <u>Рекомендация:</u> использовать CSP	String 1. «CSP» 2. «CDP»
num_sectors	(Только для 3d и sort3D_order == «CSP») Количество секторов для 3D сейсмограммы, в пределах каждого из них выбираются приемники для обработки. <u>Рекомендация:</u> num_sectors = 6	integer num_sectors > 0
Спектральный анализ		
spectral_metod	Метод расчета спектров. «SFK» – на основе преобразования Стоквелла «FKT» – на основе двумерного преобразования Фурье <u>Рекомендация:</u> использовать «SFK»	String 1. «SFK» 2. «FKT»
fmin	Минимальная частота расчета спектра в V-f области. <u>Рекомендация:</u> использовать fmin = 0	Float fmin >= 0

fmax	Максимальная частота расчета спектра в V-f области.	Float fmax > fmin
vmin	<p>Минимальная фазовая скорость для расчета спектра в V-f области.</p> <p><u>Примечание:</u> значение vmin меньше 50 в силу особенностей алгоритма значительно увеличивают время вычисления.</p> <p><u>Рекомендация:</u> использовать vmin = 50.</p>	Float vmin > 0
vmax	<p>Максимальная фазовая скорость для расчета спектра в V-f области.</p> <p><u>Примечание:</u> не рекомендуется значениями vmin и vmax выходить за пределы скоростей поверхностных волн, наблюдаемых на сейсмограмме .</p> <p><u>Рекомендация:</u> использовать vmax = 1500.</p>	Float vmax > vmin
path4dc_limits	<p>Абсолютный путь к csv-файлу с пределами извлечения фазовых скоростей для заданного набора частот.</p> <p>Используется если extract_dc_method == «max» или «dbscan».</p> <p>Формат csv-файла: 3 колонки - freq, v_min, v_max (частота, минимальное значение диапазона фазовой скорости, максимальное значение диапазона фазовой скорости).</p> <p><u>Примечание:</u> Пределы частот автоматически ограничиваются значениями fmin и fmax. Пределы скоростей автоматически ограничиваются значениями vmin и vmax. Значения скоростей линейно интерполируются для всего набора частот в спектре.</p>	<p>String</p> <p>Доступные значения для наименования директории.</p> <p>Допустимые значения в csv-файле:</p> <p>freq: Float, >= 0</p> <p>v_min: Float, >= 0</p> <p>v_max: Float > v_min</p>
extract_dc_method	Метод извлечения дисперсионной	String

	<p>кривой:</p> <p>«max» – поиск максимума на каждой частоте в пределах заданных скоростей. Пределы скоростей заданы в файле по пути path4dc_limits.</p> <p>«dbscan» – кластеризация дисперсионного изображения, выбор наибольшего кластера и расчет средней кривой по точкам кластера.</p> <p>«ae» - извлечение дисперсионных кривых с помощью предобученной сверточной нейронной сети типа AutoEncoder.</p> <p><u>Рекомендация:</u> использовать extract_dc_method = «ae».</p>	<p>1. «max»</p> <p>2. «ae»</p> <p>3. «dbscan»</p>
qc_spectral	<p>Флаг, определяющий сохранять ли дисперсионные изображения в png и segy-файлы.</p> <p><u>Примечание:</u> имя файлов дисперсионных изображений совпадает с именем сейсмограмм после препроцессинга.</p>	<p>Bool,</p> <p>True / False</p>
advanced: desired_nt	<p>Размер сейсмограммы по времени после паддинга.</p> <p><u>Рекомендация:</u> использовать desired_nt = 3000.</p>	<p>Integer</p> <p>$0 < \text{desired_nt} \leq 5000$</p>
advanced: desired_nr	<p>Размер сейсмограммы по пространству после паддинга.</p> <p><u>Рекомендация:</u> использовать desired_nr = 1000.</p>	<p>$0 < \text{desired_nr} \leq 1000$</p>
advanced: smooth_data	<p>Флаг, определяющий применение двумерного гладкого по краям окна к сейсмограмме перед трансформацией в спектральное изображение. Да – если True, нет – иначе.</p> <p><u>Примечание:</u> Процент сглаживания 10% (зашит в коде).</p>	<p>Bool,</p> <p>True / False</p>

	<u>Рекомендация:</u> smooth_data = True.	
advanced: width	(Только при spectral_metod = «SFK») Ширина импульса Гаусса . <u>Рекомендация:</u> width = 1.	Float, $0 < \text{width} \leq 100$
advanced: peak_fraction	Коэффициент использующийся при переходе от спектрального изображения к маске изображения перед извлечением дисперсионной кривой. Определяет какие амплитуды на каждой частоте из спектра перейдут в маску. Задается в долях от максимального значения амплитуды на каждой частоте. Например, при peak_fraction=0.8, на каждой частоте спектрального изображения в маску перейдут амплитуды только $\geq \text{max_ampl} * 0.8$. Параметр также используется для оценки энергии вдоль извлеченной кривой, значения с меньшей энергией будут отброшены. <u>Рекомендация:</u> peak_fraction = 0.9	Float, $0 < \text{peak_fraction} < 1$
advanced: cutoff_fraction	(Только при spectral_metod = «SFK» и extract_dc_method = «dbscan») Коэффициент использующийся при фильтрации незначительных кластеров при извлечении дисперсионных кривых. Определяет кластеры какого размера по оси частот будут отброшены. Задается в долях от размера максимального кластера по оси частот. Например, при cutoff_fraction=0.4, останутся только те кластеры, у которых размер по оси частот $\text{freq_size} \geq \text{max_freq_size} * 0.4$. <u>Примечание:</u> кластеры разных мод сортируются и нумеруются последовательно по расстоянию от точки [0,0] в пространстве фазовая скорость - частота.	Float, $0 < \text{cutoff_fraction} < 1$

	<u>Рекомендация:</u> peak_fraction = 0.4.	
advanced: dc_error_thr	<p>Коэффициент использующийся при отбраковке не гладких дисперсионных кривых. Определяет максимальный порог отличия между дисперсионной кривой и полученной по кривой полиномом четвертой степени для метрики MAPE (абсолютная процентная ошибка). Например, при dc_error_thr=0.3, будут забракованы те кривые, у которых $\text{MAX}(\text{MAPE}(\text{кривая} - \text{полином четвертой степени})) > 0.4$.</p> <p><u>Рекомендация:</u> dc_error_thr = 0.2</p>	Float, $0 < \text{dc_error_thr} < 1$
Инверсия дисперсионных кривых		
inversion_method	<p>Метод инверсии дисперсионной кривой:</p> <p>«оссам» – процедура локальной минимизации по алгоритму Оккама, в результате определяется гладкая скоростная модель (только значения Vs).</p> <p>«SSA» – процедура глобальной минимизации по алгоритму SSA, в результате определяется слоистая скоростная модель (значения Vs и мощности слоев).</p> <p>«GWO» – процедура глобальной минимизации по алгоритму GWO, в результате определяется слоистая скоростная модель (значения Vs и мощности слоев).</p> <p><u>Рекомендация:</u> inversion_method = «оссам»</p>	String <ol style="list-style-type: none"> 1. «oscam» 2. «ssa» 3. «gwo» 4. «fcnn»
niter	<p>Количество итераций для алгоритмов глобальной и локальной инверсии.</p> <p><u>Рекомендация:</u> niter = 80.</p>	Integer $10 \leq \text{niter} \leq 100$
max_num_modes	Максимальное число мод, которое будет использоваться при инверсии.	Integer $\text{max_num_modes} \geq$

	<u>Рекомендация:</u> max_num_modes = 1.	1
wavetype	<p>Тип поверхностной волны.</p> <p>«Rayleigh» – волна Релея.</p> <p>«Love» – волна Лява.</p> <p><u>Примечание:</u> метод SWI применим к обработке поверхностных волн Лява с аналогичным графом, но требуется возбуждение и регистрация SH-волн.</p> <p><u>Рекомендация:</u> wavetype = «Rayleigh».</p>	<p>String</p> <p>1. «rayleigh</p> <p>2. «love»</p>
veltype	<p>Тип скорости извлеченной дисперсионной кривой.</p> <p>«Phase» – фазовая скорость.</p> <p>«Group» – групповая скорость.</p> <p><u>Примечание:</u> в текущей реализации SWI извлекаются только фазовые скорости, но возможен запуск инверсии на групповых скоростях.</p> <p><u>Рекомендация:</u> veltype = «Phase».</p>	<p>String</p> <p>1. «phase»</p> <p>2. «group»</p>
path4vp_model	<p>Абсолютный путь к файлу со скоростной моделью Vp или отношением Vp / Vs.</p> <p>Возможные опции:</p> <p>1. Скоростная модель Vp или Vp / Vs задается через segy-файл. Указывается путь к файлу с расширением “.seggy” или “.sgy”. В таком случае требуется задание шага по глубине в dt в метрах*1000, и заголовки CDP_X, CDP_Y. До запуска инверсии выбирается ближайшая к дисперсионной кривой, полученной для точки [cmp_x, cmp_y] одномерная</p>	<p>String</p> <p>Доступные значения для наименования директории.</p> <p>Допустимые значения в csv-файле:</p> <p>depth, Float >=0</p> <p>vp, Float > 0</p> <p>vp2vs, Float > 1.4</p>

	<p>скоростная модель V_p или V_p / V_s из файла.</p> <p>2. Задание одномерной модели V_p или V_p / V_s через табличный csv-файл (константа для всей модели). Указывается путь к файлу с расширением “.csv”. Формат файла: 3 колонки - depth, v_p, v_p2v_s (глубина, значение скорости V_p, значение отношения V_p / V_s).</p> <p><u>Примечание:</u> в ходе инверсии заданная модель V_p или V_p/V_s интерполируется на глубины каждой восстанавливаемой скоростной модели V_s.</p>	
vp_model	<p>Тип задаваемой модели:</p> <p>«vp» – скорость P-волны.</p> <p>«vp2vs» – отношение скорости P-волны к скорости S-волны.</p> <p><u>Примечание:</u> определяет тип значения, читаемой из segu или csv файла.</p>	<p>String,</p> <p>1. «vp»</p> <p>2. «vp2vs»</p>
lock_vp	<p>Флаг, определяющий фиксировать ли при инверсии скоростную модель V_p, задаваемую в segu или csv файле, или нет.</p> <p><u>Примечание:</u> В случае True, скоростная модель V_p, заданная в segu или csv файле будет применяться для физического ограничения подбираемых скоростных моделей V_s на каждой итерации. В случае False, скоростная модель V_p, заданная в segu или csv файле будет использована для вычисления отношения v_p2v_s для начальной модели (локальная оптимизация) или диапазонов поиска (глобальная оптимизация) которое затем используется на для вычисления V_p из восстановленной V_s на остальных итерациях.</p>	<p>Bool,</p> <p>True / False</p>

	Рекомендация: если нет уверенности в модели Vp, то использовать lock_vp = False	
qc_inversion	<p>Флаг, определяющий сохранять ли изображения восстановленных скоростных моделей и дисперсионных кривых в png и segy-файлы.</p> <p><u>Примечание:</u> имя файлов дисперсионных изображений совпадает с именем сейсмограмм после препроцессинга и дисперсионных изображений.</p>	<p>Bool,</p> <p>True / False</p>
<p>local_search:</p> <p>nlay</p>	<p>Количество слоев для гладкой скоростной модели при инверсии Оккама.</p> <p><u>Рекомендация:</u> nlay = 10</p>	<p>Integer</p> <p>2 <= nlay <= 50</p>
<p>global_search:</p> <p>test_count</p>	<p>Число независимых запусков решения обратной задачи алгоритмами глобального поиска для одной дисперсионной кривой. Результирующая модель определяется как средняя для каждого распределения восстановленного параметра (скорости и мощности в слое).</p> <p><u>Примечание:</u> повышение параметра test_count уменьшает неоднозначность инверсии, но кратно увеличивает время решения.</p> <p><u>Рекомендация:</u> в зависимости от имеющихся ресурсов выбирать test_count в пределах 1 - 20.</p>	<p>Integer</p> <p>test_count >= 1</p>
<p>global_search:</p> <p>path4vs_limits</p>	<p>Путь к csv-файлу с заданными диапазонами поиска скоростей S-волны и мощностей слоев при инверсии.</p> <p>Формат таблицы в файле – 5 колонок: Vs_min, Vs_max, h_min, h_max, vp2vs. (Минимальное значение Vs в слое, максимальное значение Vs в слое, минимальная мощность слоя, максимальная мощность слоя,</p>	<p>String</p> <p>Доступные значения для наименования директории или пустое поле.</p> <p>Допустимые значения в csv-</p>

	<p>значение отношения V_p/V_s). Мощности последнего слоя задаются нулями. Количество строк – число слоев.</p> <p><u>Примечание:</u> Если path4vs_limits задан пустым диапазоны определяются автоматически с использованием параметра χ_i.</p>	<p>файле:</p> <p>vs_min, Float > 0</p> <p>vs_max, Float > vs_min</p> <p>h_min, Float > 0</p> <p>h_max, Float > h_min</p>
<p>global_search:</p> <p>χ_i</p>	<p>Параметр слоистости, влияет на определение количества слоев в алгоритмах глобальной оптимизации. Чем он меньше, тем больше число слоев для глобальной инверсии.</p> <p><u>Рекомендация:</u> $\chi_i = 2$</p>	<p>$1.2 \leq \chi_i \leq 5$</p>
<p>global_search:</p> <p>get_num_layers</p>	<p>Способ автоматического определения числа слоев и диапазонов по дисперсионным кривым для инверсии методом глобального поиска.</p> <p>«mean» - рассчитывается средняя дисперсионная кривая для всего набора и задается только одна слоистая модель (число слоев и диапазоны поиска);</p> <p>«classes» - набор кривых разбивается на некоторое число классов (согласно условию, что классы достаточно различаются друг от друга по заданной метрике) и диапазоны и число слоев рассчитываются в пределах одного класса;</p> <p>«every» - для каждой кривой свое число слоев (это очень дорого вычислительно, но можно получить более правильную модель).</p> <p><u>Примечание:</u> параметр используется только для inversion_method == «SSA» или «GWO» и, если path4vs_limits задан пустым.</p> <p><u>Рекомендация:</u> get_num_layers =</p>	<p>String</p> <ol style="list-style-type: none"> 1. «mean» 2. «classes» 3. «every»

	«every»	
Постпроцессинг		
max_depth	<p>Максимальная глубина в метрах для интерполированной 2D / 3D модели. Восстановленные скоростные модели с большой глубиной будут обрезаны до max_depth, с меньшей глубиной будут увеличены константой до max_depth.</p> <p><u>Рекомендация:</u> определять max_depth по изображениям восстановленных скоростных моделей в процессе подбора параметров.</p>	Float, max_depth > 0
d_x	<p>Шаг в пространственном направлении X в метрах, на который будут интерполированы восстановленные скоростные модели</p> <p><u>Рекомендация:</u> в зависимости от числа секторов в 3D d_x выбирать в пределах $\frac{1}{4}$ - $\frac{1}{2}$ от размера шага между источниками в направлении X. В 2D d_x определять как $\frac{1}{2}$ от размера шага между источниками в направлении X</p>	Float, d_x > 0
d_y	<p>Шаг в пространственном направлении Y в метрах, на который будут интерполированы восстановленные скоростные модели</p> <p><u>Рекомендация:</u> в зависимости от числа секторов в 3D d_y выбирать в пределах $\frac{1}{4}$ - $\frac{1}{2}$ от размера шага между источниками в направлении Y. В 2D d_y определять как $\frac{1}{2}$ от размера шага между источниками в направлении Y</p>	Float, d_y > 0
d_z	<p>Шаг по глубине в метрах, на который будут интерполированы восстановленные скоростные модели</p> <p><u>Примечание:</u> На первом этапе постпроцессинга каждая 1D скоростная модели как для 2D так и для 3D одномерной интерполяцией приводится к одинаковой параметризации по глубине. Затем для</p>	Float, d_z > 0

	<p>3D случая модели интерполируются по глубинным слайсам с помощью двумерной интерполяции в направлении X-Y. Для 2D возможны 2 опции, определяемые параметром interp_dim.</p> <p><u>Рекомендация:</u> d_z= 1</p>	
smooth_factor	<p>Коэффициент сглаживания интерполированной скоростной модели. При увеличении коэффициента увеличивается гладкость скоростной модели.</p> <p><u>Примечание:</u> 2Д модели сглаживается в направлении X и Z, 3Д сглаживаются в направлении X и Y.</p> <p><u>Рекомендация:</u> smooth_factor = 10</p>	<p>Float,</p> <p>smooth_factor > 0</p>
remove_outliers_smoothing	<p>Флаг определяющий сглаживать ли большие выбросы в скоростной модели с помощью алгоритма сглаживания.</p> <p><u>Рекомендация:</u> remove_outliers_smoothing = True</p>	<p>Bool,</p> <p>True / False</p>
fill_missing_values	<p>Флаг определяющий заполнять ли пропущенные значения (в результате отбраковки по error и по краям модели, где отсутствовали данные) в скоростной модели с помощью алгоритма сглаживания.</p> <p><u>Рекомендация:</u> fill_missing_values = True</p>	<p>Bool,</p> <p>True / False</p>
vmin_in_model	<p>Минимальная скорость в цветовой шкале при отрисовке скоростной модели</p> <p><u>Рекомендация:</u> vmin_in_model = 100</p>	<p>Float,</p> <p>vmin_in_model > 0</p>
vmax_in_model	<p>Максимальная скорость в цветовой шкале при отрисовке скоростной модели</p> <p><u>Рекомендация:</u> vmin_in_model = 1500</p>	<p>Float,</p> <p>vmax_in_model > 0</p>

save_seguy	<p>Флаг определяющий определяющий сохранять ли скоростную модель в segy-файл.</p> <p><u>Рекомендация:</u> save_seguy = True</p>	<p>Bool,</p> <p>True / False</p>
save_fdm	<p>Флаг определяющий определяющий сохранять ли скоростную модель в fdm-файл.</p> <p><u>Рекомендация:</u> save_seguy = True</p>	<p>Bool,</p> <p>True / False</p>
error_thr	<p>Порог ошибки (доли процентов) между извлеченной и восстановленной дисперсионной кривой выше который скоростные модели будут отброшены и проявятся как пропуски после интерполяции. При remove outliers smoothing = true эти пропуски будут заполнены с помощью алгоритма сглаживания.</p> <p><u>Примечание:</u> Ошибка вычисляется как MAX [MAPE (извлеченная кривая - восстановленная кривая)]</p> <p><u>Рекомендация:</u> error_thr = 0.2</p>	<p>Float,</p> <p>error_thr > 0</p>
path2elevation	<p>Путь для файла рельефа в текстовом (.txt) формате 3 столбца: x, y, z.</p> <p>В отсутствии файла задается нулевой рельеф.</p>	<p>String</p> <p>Доступные значения для наименования директории или пустое поле.</p>
scaler_to_elevation	<p>Масштабирующий коэффициент для значений рельефа. Если отрицательный, то значения рельефа делятся на scaler_to_elevation, если положительный то умножаются, если нуль, то рельеф не масштабируется</p>	<p>Float</p>
path2polygon	<p>Путь для файла полигона площади для 3D в текстовом (.txt) формате 2 столбца: x, y.</p> <p>В отсутствии файла будет применен выпуклый полигон</p>	<p>String</p> <p>Доступные значения для наименования директории или пустое поле.</p>

parameters_2d: interp_dim	<p>Выбор направления интерполяции для 2Д скоростной модели: “1d” или “2d”.</p> <p>При 1d интерполяция осуществляется только в направлении вдоль профиля</p> <p>При 2d применяется двумерная интерполяция в направлении вдоль профиля и глубины.</p> <p><u>Рекомендация:</u> interp_dim = 1d</p>	<p>String</p> <ol style="list-style-type: none"> 1. “1d” 2. “2d”
parameters_3d: num_xslices_3d	<p>Количество слайсов по оси X при отрисовке 3D модели в 3D-осях</p> <p><u>Рекомендация:</u> num_xslices_3d = 3</p>	<p>Integer</p> <p>num_xslices_3d >= 0</p>
parameters_3d: num_yslices_3d	<p>Количество слайсов по оси Y при отрисовке 3D модели в 3D-осях</p> <p><u>Рекомендация:</u> num_yslices_3d = 3</p>	<p>Integer</p> <p>num_yslices_3d >= 0</p>
parameters_3d: num_zslices_3d	<p>Количество слайсов по оси Z. Горизонтальные срезы рисуются только в 2D осях (в отдельных фалов изображений).</p> <p><u>Рекомендация:</u> num_yslices_3d = 3</p>	<p>Integer</p> <p>num_zslices_3d >= 0</p>

Приложение №2. Описание алгоритма SFK

Для обработки сейсмических данных широко используется непрерывное вейвлет преобразование (CWT) [Smith M. et al., 2008], которое в отличие от стандартного оконного преобразования Фурье (STFT), не требует предварительного выбора длины окна и имеет переменное частотно-временное разрешение. Преобразование Стоквелла (S-преобразование) обеспечивает частотно-временное разрешение сигнала схожее с CWT, но в отличие от него, сохраняет прямую связь со спектром Фурье. В работе [Kulesh, 2005] показана возможность оценки фазовых и групповых скоростей поверхностных волн на основе CWT. Аналогичный метод, основанный на S-преобразовании и позволяющий независимо друг от друга оценивать фазовые и групповые скорости поверхностных волн был предложен [Roohollah, 2013].

Обобщенное S-преобразование сигнала $d(t)$ определяется выражением [Stockwell et al., 1996]:

$$S[d(t)](\tau, f) = \int_{-\infty}^{+\infty} d(t) \frac{|f|}{\sqrt{2\pi\sigma}} e^{-\frac{(\tau-t)^2 f^2}{2\sigma^2}} e^{-i2\pi ft} dt, \quad (1)$$

где σ – масштабирующий параметр, контролирующий ширину вейвлета, f – частота, τ и t – временные переменные.

Рассмотрим два сигнала $d_1(t)$ и $d_2(t)$, зарегистрированных парой приемников, расположенных на одной линии с источником. В случае если сигнал состоит только из одной моды поверхностной волны, S-преобразование второго сигнала (записи второго приемника) связано с S-преобразованием первого сигнала [Roohollah, 2013]:

$$S[d_2(t)](\tau, f) = e^{-2\pi k(f)l} e^{-\lambda(f)l} S[d_1(t)](\tau - k'(f)l, f), \quad (2)$$

где $k(f)$ – волновое число, $\lambda(f)$ – частотно зависимый параметр затухания и l – расстояние между приемниками. На основе модели распространения поверхностных волн между одной парой приемников (2), возможно определять фазовую $v(f) = f/k(f)$ и групповую $u(f) = 1/k'(f)$ скорости. Как правило, зависимость волнового числа от частоты имеет линейный характер начиная с некоторой частоты, т.е. $k(f) = k_0 + \alpha f$. Групповая скорость в этом случае становится постоянной: $U(f) = 1/\alpha$, как в однородной среде. Если при этом k_0 не нулевое, то фазовая скорость не постоянна – она характеризует зависимость скорости S-волны от глубины.

Определение фазовых скоростей поверхностных волн более эффективно при использовании большего числа приемников. Применение f - k анализа позволяет разделить различные моды поверхностной волны и оценить их фазовые скорости независимо. В данной работе мы предлагаем метод позволяющий использовать модель распространения поверхностных волн на основе S-преобразования (2) при анализе многоканальных данных.

Рассмотрим нестационарный поверхностно-волновой пакет, распространяющийся вдоль линейной системы наблюдения. Применим S-преобразование (1) к каждой трассе, полученной сейсмограммы $d(x, t)$. Так мы получим трехмерное распределение по времени, расстоянию и частоте. Рассмотрим сечение полученного трехмерного распределения на фиксированной частоте:

$$\hat{d}_f(x, t) = S[d(x, t)](\tau, f, x). \quad (3)$$

Т. к. методика SWI предполагает однородное горизонтально-слоистое строение среды, будем считать, что групповая скорость поверхностных волн не зависит от расстояния x . При анализе псевдосейсмограмм на фиксированной частоте годограф поверхностно-волнового пакета образует прямую наклонную линию, выходящую из точки источника. Будем считать, что источник генерирует импульс в начале координат (как по времени, так и по пространству). Для каждой частоты f и групповой скорости u введем одномерную комплекснозначную функцию пространственной горизонтальной координаты:

$$\hat{d}_{f,u}(x) = \hat{d}_f\left(x, \frac{x}{u}\right) \quad (4)$$

Для получения волнового числа к функции (4) применим одномерное преобразование Фурье по пространственной переменной:

$$\Omega(u, f, k) = \int_{-\infty}^{+\infty} \hat{d}_{f,u}(x) e^{-2\pi i x k} dx. \quad (5)$$

Для получения из 3х-мерного распределения (5) традиционного f - k изображения необходимо выполнить поиск максимумов амплитуд функции для серии групповых скоростей, покрывающих область возможных наклонов пакета поверхностной волны:

$$D(f, k) = \max_u \Omega(u, f, k) \quad (6)$$

Вместо стандартных f - k или f - v изображений, полученных с помощью традиционного двумерного преобразования Фурье, для извлечения дисперсионной кривой мы предлагаем анализировать распределение (6). Локализация по времени на каждой частоте повышает соотношение сигнал/помеха, поскольку исключает шум от помех или от других волн, которые не попадают в эффективную ширину соответствующего вейвлета.

Следующим этапом предлагаемого подхода является пикирование дисперсионной кривой путем поиска максимумов амплитуд на каждой частоте:

$$k(f) = \operatorname{argmax}_k D(f, k). \quad (7)$$

Как показывают эксперименты по обработке реальных данных, дисперсионная кривая (7) имеет более гладкий характер, по сравнению с кривой, получаемой с помощью стандартного f - k преобразования. Заметим, что расчетные (теоретические) дисперсионные кривые в горизонтально-слоистых средах всегда гладкие. Этот факт свидетельствует об эффективности предлагаемого подхода. Далее приводится пример применения предлагаемого алгоритма извлечения дисперсионных кривых на синтетических данных.

Для демонстрации фильтрационных свойств предлагаемого метода рассмотрим горизонтально-слоистую модель среды с неоднородными включениями (рисунок 1а-1в). Неоднородности добавлялись случайным образом с увеличением их размера по мере увеличения глубины, при этом сохранялась средняя скорость в каждом слое.

Полное упругое волновое поле рассчитывалось с помощью конечно-разностной схемы на сдвинутых сетках [Virieux 1986]. Источник – вертикально направленный удар с формой импульса Риккера с $f_d = 30$ Гц. Сейсмограмма вертикальной компоненты

смещений представлена на рисунке 1г. Сейсмограмма состоит из 101 трассы с шагом между приемниками – 2 м. К данным был добавлен нормально распределенный шум Гаусса (рисунок 1д).

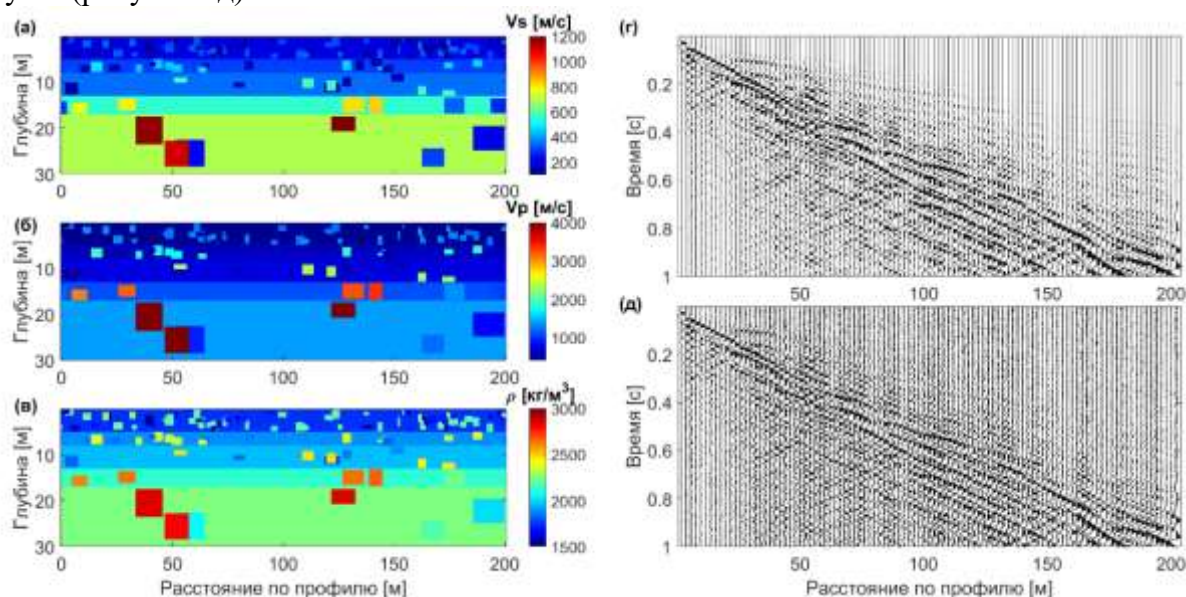


Рисунок 1. Модель среды и данные для синтетического теста: а – модель скоростей поперечной волны (м/с); б – модель скоростей продольной волны (м/с); в – модель плотности (кг/м³); г – сейсмограмма вертикальной компоненты смещения; д – сейсмограмма с добавлением нормально-распределенного шума Гаусса (SNR = 13 dB).

На рисунке 2 представлены $f-k$ и $v-f$ изображения, полученные с помощью стандартного двумерного преобразования Фурье и предлагаемого подхода на основе S-преобразования. Значения амплитуды нормируются – приведены в долях от максимального значения на каждой частоте. Предлагаемый метод усиливает пакет поверхностных волн путем фильтрации других волн на временно-частотном распределении. В результате получается более гладкое и «чистое» спектральное изображение. При использовании стандартного подхода на спектральном изображении наблюдаются многочисленные ложные локальные максимумы, искажающие положение и амплитуды спектральных пиков, соответствующих фундаментальной моде.

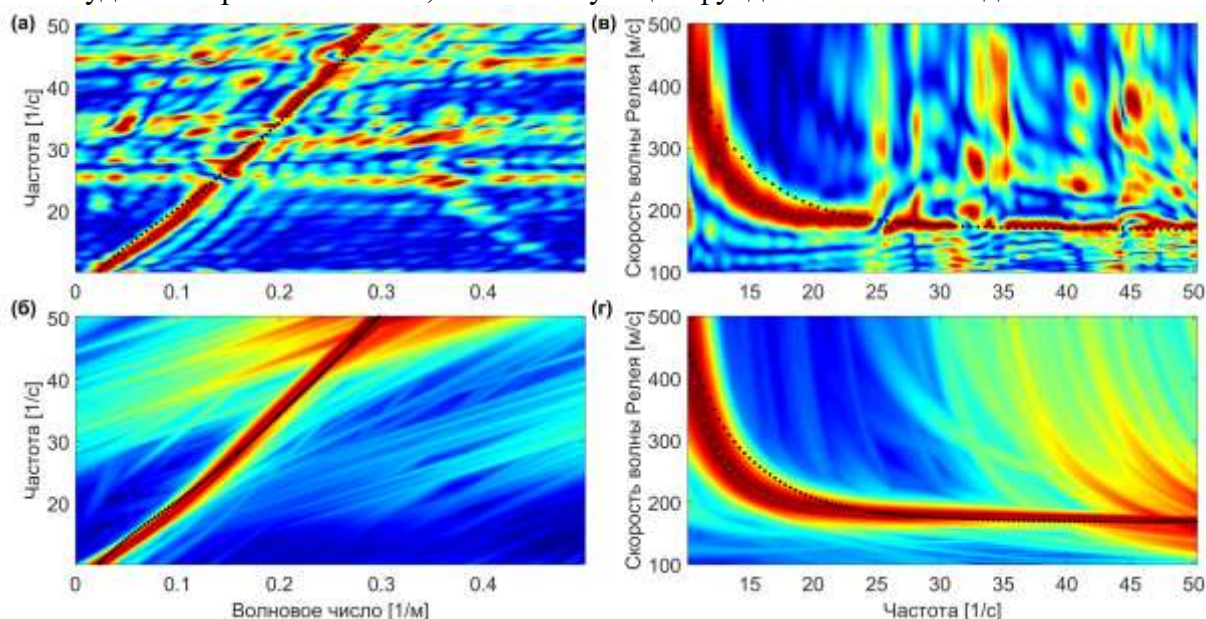


Рисунок 2. Спектральные образы сейсмограммы с ПВ0 м. а – $f-k$ спектр, полученный с помощью двумерного преобразования Фурье; б – $f-k$ спектр, полученный с помощью S-

преобразования; ν – спектр фазовых скоростей, полученный с помощью двумерного преобразования Фурье; γ – спектр фазовых скоростей, полученный с помощью S-преобразования. Черными точками нанесена аналитическая кривая, рассчитанная для среды со средними по слоям значениями упругих параметров, красными точками нанесены автоматически определенные максимумы.

Анализируя спектральные изображения рисунка 2, отметим, что кривая, полученная с помощью S-преобразования, находится в лучшем согласии с аналитической кривой, чем при стандартном подходе. Таким образом, предлагаемый подход позволяет автоматизировать процесс извлечения дисперсионных кривых поверхностных волн, как в случае простой горизонтально-слоистой среды, так и при наличии неоднородностей, которые приносят сложный шум в спектр и делают неоднозначным процесс пикирования.

Приложение №3. Описание алгоритма Оккама

В модуле SWI реализован новый подход к решению обратной задачи для дисперсионных кривых поверхностных волн, основанный на алгоритме Оккама [Constable et al., 1987]. Этот алгоритм среди множества допустимых решений выбирает максимально гладкое и позволяет избежать осцилляций, характерных для решений данного круга задач. Для построения алгоритма корректного решения обратной задачи необходимо учесть некоторые важные особенности процесса подбора дисперсионных кривых. К этим особенностям относятся: выбор алгоритма решения прямой задачи, построение начальной модели, обеспечивающей сходимость инверсии, правильный подбор сглаживающего параметра.

Решение прямой задачи

Для идеально упругой изотропной горизонтально-слоистой среды зависимость фазовой скорости от частоты (дисперсионную кривую) можно получить путем разрешения аналитических выражений, составленных для такой среды.

Рассмотрим постановку данной задачи для волны Релея. Пусть ось z направлена вертикально вниз, нижнее полупространство $z > 0$ занимает вертикально-неоднородная среда. Рассмотрим декартовы координаты и поверхностную волну, распространяющуюся в горизонтальном направлении возрастания x с частотой ω :

$$\mathbf{u}(x, y, z, t) = \mathbf{Z}(z) \exp[i(kx - \omega t)]. \quad (1)$$

Решение уравнений упругости вида (1) соответствующее волнам Рэлея, должно удовлетворять двум краевым условиям. Во-первых волны должны удовлетворять условиям на поверхности $z = 0$, где нормальные напряжения равны нулю. Во-вторых, на границе $z = \infty$ их амплитуды должны быть равны нулю из-за отсутствия источников на бесконечности. При таких ограничениях не существует нетривиального решения уравнения (1) для произвольной частоты ω и произвольного волнового числа k . Однако оно может существовать, если для данного ω k принимает некоторое частное значение, например $k_n(\omega)$. Это задача на собственные значения, здесь используется индекс n так как для данного $\mathbf{u}_n(z)$ у поверхностной волны (1) находится более чем одно значение k . Таким образом, для данной частоты ω поверхностные волны имеют единственным образом определенные волновые числа $k_0(\omega), k_1(\omega) \dots$. Другими словами, фазовые скорости $V_n = \frac{\omega}{k_n}$ фиксированы для данной частоты.

Для вывода дисперсионных кривых для волны Релея, рассмотрим физически обоснованное решение – смещения в горизонтальных (u и v) и вертикальной (w) осях [Аки, Ричардс, 1983]:

$$\begin{aligned} u &= r_1(k, z, \omega) \exp[i(kx - \omega t)], \\ v &= 0, \\ w &= ir_2(k, z, \omega) \exp[i(kx - \omega t)] \end{aligned} \quad (2)$$

и соответствующие им компоненты напряжений:

$$\begin{aligned}
\tau_{yz} &= \tau_{xy} = 0, \\
\tau_{xx} &= i \left[\lambda \frac{dr_2}{dz} + k(\lambda + 2\mu)r_1 \right] \exp[i(kx - \omega t)], \\
\tau_{yy} &= i \left[\lambda \frac{dr_2}{dz} + k\lambda r_1 \right] \exp[i(kx - \omega t)], \\
\tau_{zz} &= i \left[(\lambda + 2\mu) \frac{dr_2}{dz} + k\lambda r_1 \right] \exp[i(kx - \omega t)], \\
\tau_{zx} &= \mu \left[\frac{dr_1}{dz} - kr_2 \right] \exp[i(kx - \omega t)],
\end{aligned} \tag{3}$$

компоненты напряжения τ_{zx} и τ_{zz} непрерывны по глубине z , запишем их как

$$\begin{aligned}
\tau_{zx} &= r_3(k, z, \omega) \exp[i(kx - \omega t)], \\
\tau_{zz} &= ir_4(k, z, \omega) \exp[i(kx - \omega t)].
\end{aligned} \tag{4}$$

Уравнения (2) и (3) с учетом (4) подставляются в уравнение движения для изотропной идеально упругой вертикально-неоднородной среды. Тогда дифференциальные уравнения для вектора движения-напряжения (r_1, r_2, r_3, r_4) примет вид:

$$\frac{d}{dz} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix} = \begin{pmatrix} 0 & k & \mu^{-1}(z) & 0 \\ -k\lambda(z)[\lambda(z) + 2\mu(z)]^{-1} & 0 & 0 & [\lambda(z) + 2\mu(z)]^{-1} \\ k^2\zeta(z) - \omega^2\rho(z) & 0 & 0 & k\lambda(z)[\lambda(z) + 2\mu(z)]^{-1} \\ 0 & -\omega^2\rho(z) & -k & 0 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix}, \tag{5}$$

$$\text{где } \zeta(z) = \frac{4\mu(z)[\lambda(z) + \mu(z)]}{[\lambda(z) + 2\mu(z)]}.$$

Граничные условия для поверхностных волн, т.е. обращение в нуль напряжения на свободной поверхности и отсутствие напряжения на бесконечности, выражаются в том что

$$\begin{aligned}
r_1, r_2 \text{ и } l_1 &\rightarrow 0 \text{ при } z \rightarrow \infty, \\
r_3 = r_4 = l_2 &= 0 \text{ на свободной поверхности } (z = z_0).
\end{aligned} \tag{6}$$

Для конкретной частоты ω нетривиальные решения системы дифференциальных уравнений (5) при условии (6) существуют только для определенных $k = k_n(\omega)$.

Соотношение $\frac{\omega}{k_n}$ определяет фазовую скорость, а решение (5) – зависимость моды от z .

Аналогично выводиться система дифференциальных уравнений для вектора движения-напряжения для волн Лява, если вместо 3.1.2. рассматривать решение следующего вида:

$$\begin{aligned}
u &= 0, \\
v &= l_1(k, z, \omega) \exp[i(kx - \omega t)], \\
w &= 0.
\end{aligned}$$

Существует несколько способов решения задачи на собственные значения – собственные векторы (5). В модуле SWI используется программная реализация быстрого современного алгоритма Лая и Рикса [Lai, Rix, 1998], использующая решение методом

Томсона-Хаскелла [Thomson 1950; Haskell 1953], частного случая метода матричного пропагатора, применяемым для серии однородных слоев, подстилаемых однородным полупространством.

Построение начального приближения

Решение обратной задачи методами локальной оптимизации начинается с построения начального приближения. В результате тестирования метода на ряде синтетических данных и исследования алгоритма инверсии была дана оценка сходимости процесса в зависимости от выбора начального приближения. На рисунке 1 представлены результаты инверсии по синтетическим данным при разном отклонении стартовой модели от истинной модели среды и соответствующее им поведение функционалов невязки в зависимости от шага итерации. Как видно из поведения функционала глобальный минимум достигается, т.е. итерационный процесс решения обратной задачи сходится, при начальной модели, отличающейся от истинной не более чем на 45%.

Подобную точность задания стартового приближения можно достичь при использовании предложенного способа, схематично представленного на рисунке 2. Для задания стартовой модели используется наблюдаемая дисперсионная кривая и некоторые физические особенности поверхностных волн. Каждая точка на дисперсионной кривой имеет свою длину волны ($\lambda_i = V_i / f_i$), которая определяется типом используемого источника, характеристикой среды и параметрами системы наблюдений и скорость (V_i). Для перевода точки на дисперсионной кривой в модель среды используются переводные коэффициенты, подкрепляемые теорией классиков [Аки К., Ридчардс П., 1983; Викторов И.А., 1981].

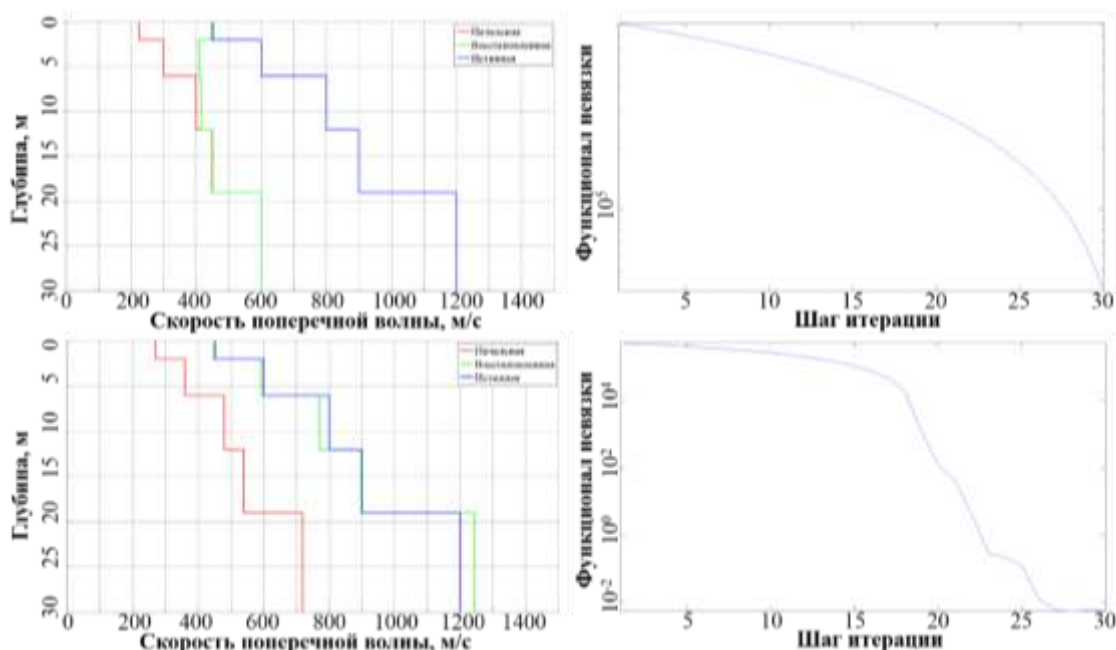


Рис. 1. Сверху слева результат инверсии при стартовой модели, отличающейся от истинной на 50%, на рисунке снизу слева на 40% для каждого слоя. Справа представлены функционалы невязки в зависимости от шага итерации для каждого случая.

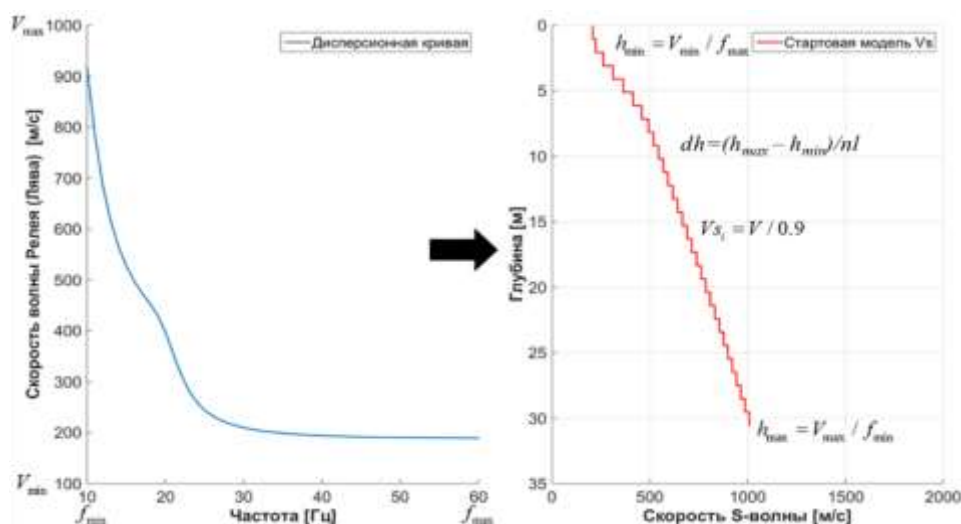


Рис. 2. Схема построения стартовой модели из наблюдаемой дисперсионной кривой.

Так, максимальная глубина исследования h_{max} примерно равна $1/3$ максимальной длины волны для волн Релея и $1/5$ для волн Лява. Повысить глубину исследования можно используя более мощные источники и более низкочастотные приемники. Связь скорости поперечной волны и волны Релея (или Лява) берется из расчета для однородной среды. При коэффициенте Пуассона $\nu \in [0; 0,5]$ скорость поверхностных волн меняется в пределах $V \in [0,87; 0,96] V_s$.

Так как инверсия Оккама представляет собой плавную инверсию, в которой число слоев заранее не определено шаг по слоям берется одинаковый: $dh = (h_{max} - h_{min}) / nl$, где nl – число слоев, который интерпретатор должен выбирать сам, тем самым добиваясь лучшего подбора конечной модели. Таким образом в зависимости от задачи можно подбирать очень гладкую (с большим числом слоев) модель или наоборот очень разрешенную (с малым числом слоев).

Ввиду не определенности параметра nl было проведено дополнительное исследование, результаты которого приведены, на рисунке 3. Здесь представлены варианты инверсии для синтетических данных, отличающиеся только количеством слоев стартовой модели. Как видно из рисунка качественный подбор модели обеспечивается при $nl \approx 10-30$, слишком маленькое значение ($nl < 10$) не может обеспечить максимальную сходимость кривых, в виду недостаточной дискретизации стартовой модели, и, напротив, использование слишком больших значений ($nl > 30$) не привносит новых данных в конечный результат, а только замедляет процесс инверсии.

Сглаживание функционала невязки

Решение линеаризованной обратной задачи для волны Релея или Лява (искомая невязка модели на каждой итерации) на основе подхода Оккама представляется уравнением [Lai C.G., 2005]:

$$dV_s = \left\{ \mu (\partial^T \partial) + \left[(J_s)_{V_{s_0}} \right]^T \left[(J_s)_{V_{s_0}} \right] \right\}^{-1} \left[(J_s)_{V_{s_0}} \right]^T (V_{R(L)}^{набл} - V_{R(L)}^{расч}(V_{s_0})),$$

где μ это множитель Лагранжа, который может быть интерпретирован как параметр сглаживания, ∂ - конечно-разностный дифференциальный оператор, имеющий вид:

$$\partial = \begin{bmatrix} 0 & & & & \\ -1 & 1 & & & 0 \\ & -1 & 1 & & \\ & & & \dots & \\ 0 & & & -1 & 1 \end{bmatrix},$$

$$(J_s)_{V_{S_0}} = \frac{\partial(V_{R(L)})_j}{\partial(V_S)_k} - \text{Якобиан, показывающий, как изменение одного из параметров}$$

среды, в данном случае используется скорость поперечной волны, влияет на изменение значений фазовой скорости волны Релея (или Лява). $V_{R(L)}^{\text{набл}}$ – наблюдаемая дисперсионная кривая, извлеченная из спектра сейсмограммы, $V_{R(L)}^{\text{расч}}(V_{S_0})$ – расчетная дисперсионная кривая для данной стартовой модели на каждой итерации. Частные производные для построения матрицы Якоби рассчитываются путем вычисления интегральных уравнений, составленных для однородных плоскопараллельных слоев, покрывающих однородное полупространство [Аки, Ричардс, 1983]. Конечная модель получается путем прибавления к начальному приближению V_{S_0} невязки модели dV_s :

$$V_S = V_{S_0} + dV_S$$

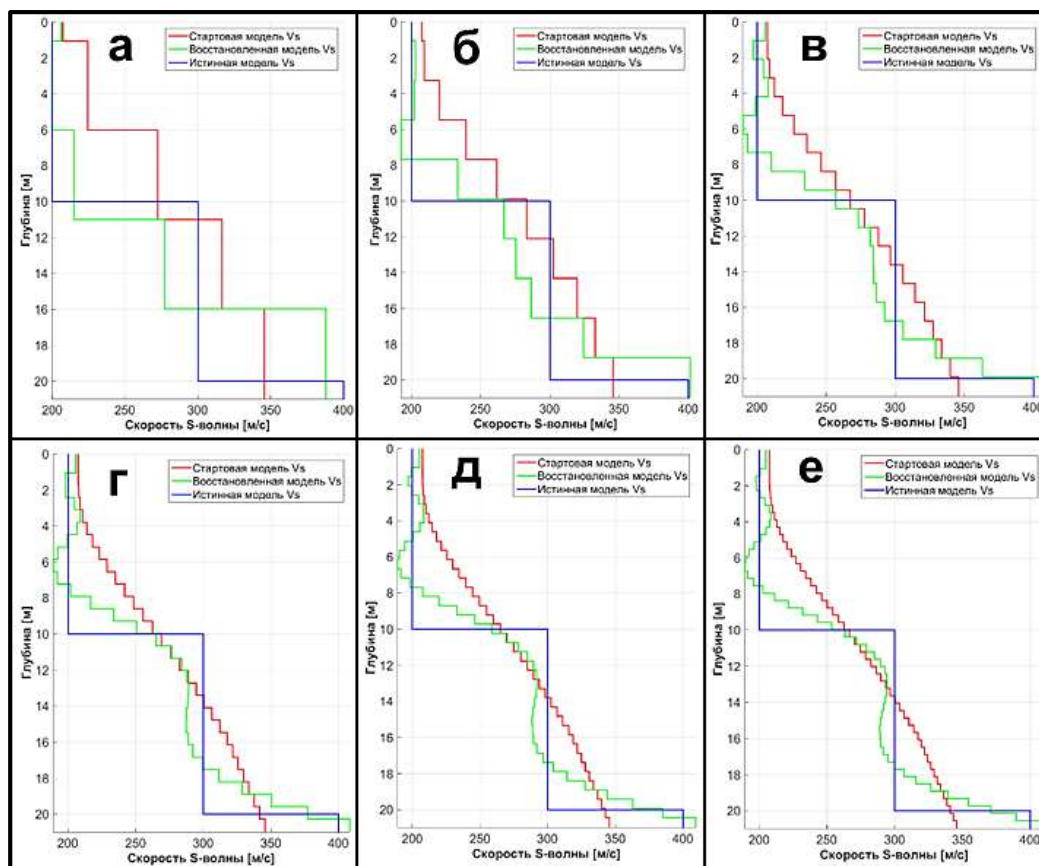


Рис. 3. Результаты решения обратной задачи в зависимости от количества слоев в стартовой модели: а – 5, б – 10, в – 20, г – 30, д – 40, е – 50 слоев.

Множитель Лагранжа путем ограничения функционала невязки двух кривых способствует его минимизации. Правильный подбор параметра μ способствует быстрой

минимизации функционала и более качественному решению обратной задачи. На рисунке 4 показаны значения функционала невязки как функции от множителя Лагранжа.

Определив, таким образом значения μ , которые доставляют минимум функционала на каждой из 6ти итераций, подставим их в решение обратной задачи. Результат решения на 6ой итерации представлен на рисунке 5. Его можно сравнить с решением представленном на рисунке 3г, где использовалось постоянное значение $\mu=0.05$, а количество итераций равнялось 50ти. Очевидно, что необходимо производить подбор параметра Лагранжа на каждой итерации. Такой подход позволит быстрее минимизировать функционал и обеспечит подбор восстановленной модели ближе к истинной. В модуле SWI реализован адаптивное изменения множителя Лагранжа в зависимости от итерации и значения невязки между дисперсионными кривыми.

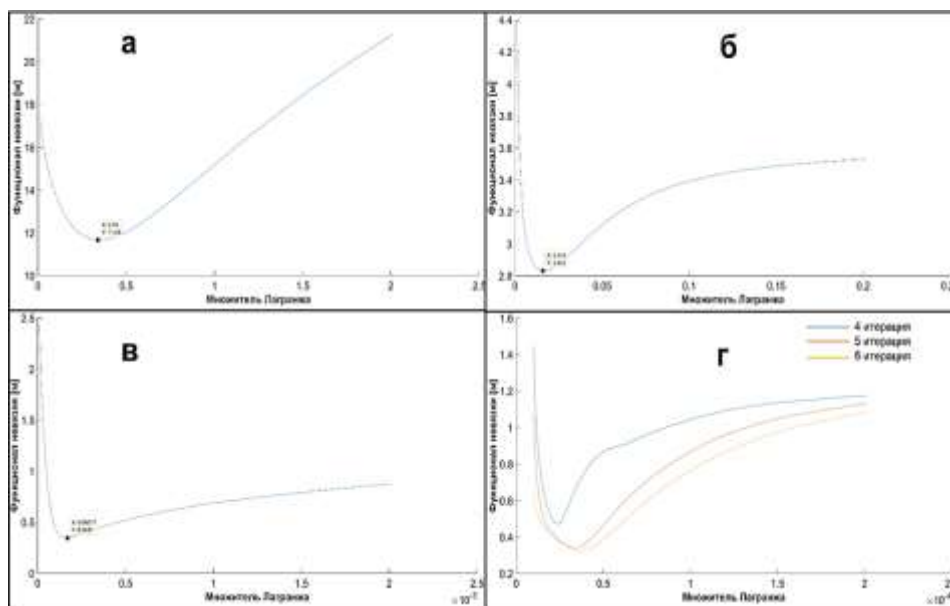


Рис. 4. Среднеквадратический функционал невязки между наблюдаемой и восстановленной дисперсионной кривой как функция от множителя Лагранжа для первых шести итераций: а – 1 итерация, б – 2 итерация, в – 3 итерация, г – 4-6 итерация.

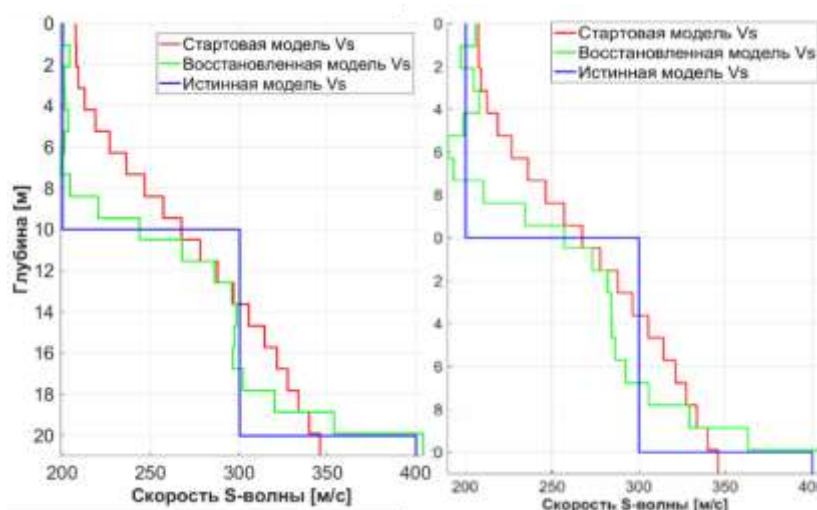


Рис. 5. Результат решения обратной задачи: слева – по заранее найденным значениям параметра Лагранжа (рис. 4) для первых 6 итераций, справа – решение при постоянном значении $\mu = 0,05$ для 50 итераций.

Приложение №4. Описание алгоритма SSA

В приложении описывается подход для оценки диапазонов подбираемых параметров в процессе инверсии методами глобального поиска (SSA и GWO), а также алгоритм SSA с примерами исследования на синтетических данных.

Описание подхода к оценке диапазонов поиска параметров скоростной модели

Частотно-зависимая фазовая скорость поверхностной волны рассматривается как функция от длины волны: $V_R(f) = V_R(V_R/f) = V_R(\lambda)$. Для оценки диапазонов глубин i -го слоя (d_{min}^i и d_{max}^i), в работе [Cox and Teague, 2016] введен эмпирический параметр Ξ , названный «коэффициент слоистости», который представляет собой множитель систематически повышающий мощность каждого слоя исходя из потенциальной толщины слоя выше. Значение Ξ определяет число слоев. Авторы работы [Cox and Teague, 2016] предложили следующий набор значений $\Xi = \{1.2, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 5.0, 7.0\}$ и показали, что выбор значения Ξ оказывает прямое влияние на результирующую модель при инверсии методом ближайшего соседа.

Следуя алгоритму [Cox and Teague, 2016], оцениваем диапазоны глубин для каждого слоя по итерационным формулам:

$$\begin{aligned} d_{min}^i &\approx \begin{cases} \lambda_{min}/3 & \text{при } i=1 \\ d_{max}^{i-1} & \text{при } i>1 \end{cases} \\ d_{max}^i &\approx \begin{cases} \lambda_{min}/2 & \text{при } i=1 \\ d_{min}^i + \Xi \cdot \lambda_{min}/2 & \text{при } i=2 \\ d_{min}^i + \Xi \cdot (d_{max}^{i-1} - d_{min}^{i-1}) & \text{при } i>2 \end{cases} \end{aligned} \quad (1)$$

Максимальная глубина (d_{max}^i) определяемой модели не превышает вертикальное разрешение (d_{res}), которое определяется как половина максимальной наблюдаемой длины волны ($\lambda_{max}/2$). Поэтому, итеративный процесс (1) останавливается когда $d_{max}^N > d_{res}$ и $i = N$, где N -ый слой – подстилающее полупространство. После определения диапазонов глубин границ рассчитываются диапазоны мощностей для каждого слоя:

$$\begin{cases} h_{min}^i = \lambda_{min}/3 \\ h_{max}^i = d_{max}^i - d_{min}^i \end{cases}$$

В [Cox and Teague, 2016] не приведен алгоритм оценки диапазонов V_S^i . Используется аппроксимационная зависимость $V_S(\lambda)$ от фазовой скорости волны Релея $V_R(\lambda)$ для однородной среды [Viktrov, 1967]:

$$V_S(\lambda) \approx 1.16V_R(\lambda)$$

Далее применяется используемое на практике для эмпирической инверсии [Richart et al., 1970] соответствие между глубиной d и наблюдаемой длиной волны λ :

$$d \approx \lambda/2. \quad (2)$$

В соответствии с (2) V_S можно рассматривать как функцию от глубины d . Для каждого i -го слоя при $i < N$ рассчитывается среднее значение скорости в диапазоне:

$$V_S^{i, \text{сред}} = V_S(d_{\text{сред}}^i),$$

где средняя глубина $d_{\text{сред}}^i$ – полусумма граничных глубин, определенных в (1):

$$d_{\text{сред}}^i = \frac{d_{\text{max}}^i + d_{\text{min}}^i}{2}.$$

Скорость в полупространстве определяется как $V_S^{N, \text{сред}} = 1.16V_R(f_{\text{min}})$, где f_{min} – минимальная наблюдаемая частота в дисперсионной кривой.

Для определения диапазонов V_S каждого слоя добавляется/вычитается постоянное значение ($100 \cdot \varepsilon$ м/с) к/из среднего значения ($V_S^{i, \text{сред}}$), т.е. $V_S^{i, \text{min}} = V_S^{i, \text{сред}} - 100 \cdot \varepsilon$ и $V_S^{i, \text{max}} = V_S^{i, \text{сред}} + 100 \cdot \varepsilon$. Дополнительно выполняется условие, что V_S^i не может быть меньше чем 100 м/с, т.е. $V_S^{i, \text{min}} = \max[V_S^{i, \text{сред}} - 100 \cdot \varepsilon, 100]$.

Описание алгоритма SSA

Алгоритм роя сальп (от англ. Salp Swarm Algorithm или SSA) – оптимизатор на основе биоинспирации, созданный для решения задач инженерного моделирования. Он вдохновлен видом колоний плавающих морских животных – сальпов, образующих изгибающиеся цепочки или спирали. Алгоритм SSA характеризуется простотой, гибкостью, отсутствием деривационных механизмов и способностью избегать локальных оптимумов. Алгоритм SSA является коллективным, т.е. в ходе оптимизации улучшается множество решений. В отличие от индивидуалистических, коллективные алгоритмы позволяют лучше избегать локальных решений и обмениваться информацией о пространстве поиска, однако требуют большего числа оценок функции невязки. SSA успешно применяется для решения различных оптимизационных задач и показывает конкурентоспособную производительность по сравнению с другими метаэвристиками [Mirjalili et al., 2017].

Математическая модель роя сальп

В основе оптимизационного алгоритма роя сальп (от англ. Salp Swarm Algorithm или SSA) лежит симуляция поведения колоний сальп, образующих изгибающиеся цепочки. При симуляции роевого поведения цепочка сальпов делится на две группы: лидера и последователей. Лидером является первый или несколько первых сальпов (агентов), а остальные считаются последователями. Поведение цепочки моделируется таким образом, что лидер движет группу, а последователи движутся за предыдущим звеном цепи. Важным этапом моделей роевого интеллекта является переход от разведки к нападению, т. е. момент, когда заканчивается поиск оптимума по всему пространству параметров и начинается исследование в локальной области. Авторы SSA описали такой переход параметром c_l^1 , который характеризует длину шага от оптимума на итерации l и уменьшается с каждой итерацией. Переход от разведки к нападению принято выполнять после прохождения 50% всех итераций.

По аналогии с другими роевыми методами оптимизации положение сальпа определяется в D -мерном пространстве переменных. Таким образом положение всех

агентов сохраняется в матрице X_{ld}^k , $l \in [1, L]$, $d \in [1, D]$, $k \in [1, K]$, где L – количество итераций, D – количество переменных данной задачи, K – количество агентов, из них M – количество лидеров. Количество агентов выбирается из размерности задачи: $K = 10 \cdot D$, где D – размерность задачи, которое складывается из числа восстанавливаемых параметров. Число итераций равнялось 100 для всех экспериментов в данном исследовании.

Предполагается, что в пространстве поиска находится цель, к которой движется цепочка агентов, называемая источником пищи G . В текущей постановке источником пищи является позиция значений векторов V_S и h восстанавливаемой модели. Начальное положение лидера и источника пищи инициализируется значениями из случайного равномерного распределения, заданного в диапазонах поиска решения. По этой причине стохастические методы инверсии от запуска к запуску могут давать отличающиеся решения. Для поиска наилучшего результата выполняется несколько запусков оптимизационной задачи. Например, авторы инверсии дисперсионных кривых с использованием метода GWO рекомендуют использовать 20 независимых запусков решения [Song et al., 2015].

Изменение положение лидера в процессе итераций определяется следующим выражением:

$$X_{ld}^m = \begin{cases} G_{ld} + c_l^1((ub_d - lb_d)c^2 + lb_d), & c^3 \geq 0.5 \\ G_{ld} - c_l^1((ub_d - lb_d)c^2 + lb_d), & c^3 < 0.5 \end{cases} \quad (3)$$

где X_{ld}^m – положение лидера $m \in [1, M]$ в пространстве параметра d на l -ой итерации, G_{ld} – положение пищи в пространстве параметра d на l -ой итерации, ub_d и lb_d соответственно означают верхнюю и нижнюю границу поиска в пространстве параметра d . Границы от итерации к итерации сохраняются. Параметр c_l^1 является наиболее важным, поскольку отвечает за переход от поиска к нападению и определяется выражением:

$$c_l^1 = 2e^{-\left(\frac{4l}{L}\right)^2} \quad (4)$$

Параметры c^2 и c^3 выбираются из случайного равномерного распределения, заданного в диапазоне $[0, 1]$. Параметр c^2 – отвечает за размер случайного шага лидера, а c^3 за направление шага. Движение лидеров всегда происходит по направлению к источнику пищи. Изменение положений сальпов-последователей описывается законом движения Ньютона:

$$X_{ld}^k = \frac{1}{2}at^2 + v_0t, \quad (5)$$

где $k > M$, X_{ld}^k – позиция k -го сальпа-последователя в пространстве параметра d на итерации l , $t = l * dt$ – время, v_0 – начальная скорость, $a = (v - v_0)/dt$ – ускорение, где $v = (x - x_0)/dt$. В задачи оптимизации время дискретизации является итерацией, следовательно $dt = 1$, с учётом того, что задача оптимизации начинается со стационарного положения $v_0 = 0$, изменение положения сальпов-последователей для итерации l имеет вид:

$$X_{ld}^k = \frac{((x - x_0) - v_0)}{dt} + v_0t = \frac{1}{2}(X_{ld}^{k-1} - X_{ld}^k) + X_{ld}^k = \frac{1}{2}(X_{ld}^{k-1} + X_{ld}^k) \quad (6)$$

Таким образом, сальп в пространстве параметра d на l -итерации от своего положения сдвигается к предыдущему в цепи сальпу на половину расстояния между ними, так происходит со всей группой последователей. Положение источника пищи G_{ld} обновляется после того, как все агенты изменили своё положение в соответствии с уравнениями (3) и (6), и являются оптимальными параметрами после l -итераций.

Описание оптимизационного процесса по алгоритму SSA

Использование математической модели для симуляции салп-цепей напрямую в задачах оптимизации не является возможным. Требуется модифицирование модели с целью ее адаптации для решения оптимизационных задач. Основной целью оптимизатора является выявление глобального оптимума. В контексте модели «рой салп» последователи двигаются за лидером, который также направляется к источнику пищи. Если источник пищи ассоциировать с глобальным оптимумом, то салп-цепь автоматически направляется к этому оптимуму. Тем не менее, существует сложность в том, что глобальный оптимум задачи оптимизации неизвестен. В таких ситуациях принимается предположение о том, что наилучшее решение, найденное на данный момент, считается глобальным оптимумом и используется как источник пищи, к которому салп-цепь стремится.

В текущем исследовании SSA адаптирован к задаче инверсии дисперсионных кривых фазовых скоростей. Рассматривается инверсия только дисперсионной кривой фундаментальной моды волны Релея для восстановления значений скорости поперечной волны (V_S) и мощностей слоёв (h) структуры верхней части геологического разреза. Реализация алгоритма выполнена на языке программирования Python с использованием библиотек *numpy*, *sklearn*, *disba*, *joblib* и *multiprocessing*. Оптимизационная задача направлена на поиск глобального минимума невязки между извлеченной и расчетной дисперсионными кривыми. Прямое моделирование дисперсионных кривых волны Релея осуществлялось с помощью библиотеки *disba* (Python), расчет в которой реализован матричным алгоритмом Дункина [Dunkin, 1965]. Псевдокод алгоритма SSA описывается следующим образом:

```

Инициализация агентов (салпов)  $X_{0d}^k$  в пределах  $ub$  и  $lb$ 
while (условие остановки, как правило, заданное количество итераций)
    Расчёт невязки для каждого агента
    Текущее положение пищи  $G_{0d}$  = поисковый агент с наименьшей невязкой
    Обновление  $c_0^1$ , согласно уравнению (2)
    for каждого агента ( $X_{0d}^k$ )
        if ( $d==1$ )
            Обновляется позиция лидера, согласно уравнению (1)
        else
            Обновляется позиция последователей, согласно уравнению (4)
        end
    end
    Поправки в положение агентов с учетом границ поиска
End
Return  $G_{0d}$ 

```

Из описания псевдокода следует, что процесс начинается с аппроксимации глобального оптимума путем инициализации множества агентов случайным образом. Затем для каждого агента вычисляется значение функционала невязки, выбирается агент с наименьшей невязкой, и его позиция назначается источником пищи. Далее обновляется коэффициент c^1 , затем обновляется положение лидера и положение последователей по каждой размерности задачи (G_{0d}). Если какой-то из агентов выходит за границы пространства поиска, то его позиция корректируется. Все шаги, кроме инициализации, выполняются итерационно до достижения максимальной итерации.

Следует отметить, что в процессе оптимизации источник пищи (G_{ld}) будет периодически обновляться, поскольку цепочка салп с высокой вероятностью находит

более оптимальное решение, исследуя пространство поиска. Положение лидера обновляется только на основе лучшего решения на текущий момент, при этом имеет случайный шаг. Это обеспечивает дополнительное исследование лидером пространство вокруг себя. Пересчет положения последователей относительно друг друга, позволяет им постепенно двигаться к лидеру, что предотвращает легкую стагнацию алгоритма SSA в локальных оптимумах. Адаптивное уменьшение коэффициента c^1 по мере итераций, позволяет алгоритму SSA совершить своевременный переход от исследования пространства поиска (разведка) к его эксплуатации (нападение).

Для наглядной визуализации описанной математической модели, выполняется моделирование задачи оптимизации. В качестве функции невязки использована L2 (евклидова) норма. Рассчитана дисперсионная кривая для горизонтально-слоистой среды, состоящей из 2 слоёв с параметрами V_S : 150, 700 м/с, мощность верхнего слоя $h = 5$ м. С целью получения визуализации для двумерной задачи (в смысле размерности пространства поиска) мощность слоя была зафиксирована (не подбиралась), т.е. поиск выполнялся в пространстве: V_S первого слоя – V_S второго слоя ($V_S^1 - V_S^2$). В качестве лидера был выбран первый сальп, остальные считались последователями и следовали за ним. Двадцать сальпов случайным образом размещены в плоскости параметров $V_S^1 - V_S^2$ (рис. 1, $l = 0$), на последующих итерациях видно, как рой формирует цепочку и как изменяется положение пищи (оптимума) (зелёный ромб рис. 1, $l \in [10, 80]$). Стоит заметить, что на 50 итерации (рис. 1, $l = 50$) сальпы переходят от режима поиска к режиму нападения, такое поведение обеспечивает баланс между устойчивостью решения задачи оптимизации и избегания попадания в локальные минимумы. На последних итерациях (рис. 1, $l \in [80, 99]$) решение сходится к истинному (отмечено красной звездой на рис. 1). Результаты моделирования, подтверждают, что цепочка способна эффективно преследовать движущийся источник пищи и может направляться к глобальному оптимуму, который изменяется в процессе оптимизации.

Далее качественно анализируются промежуточные результаты решения обратной оптимизационной задачи. Приводится срез семимерного (подбираются 4 скорости и 3 мощности) функционала невязки в плоскости V_S второго слоя – V_S третьего слоя ($V_S^2 - V_S^3$), история позиций всех поисковых агентов, а также кривые сходимости (значения невязки на каждой итерации) и точности (значения MAPE на каждой итерации) для метрики *MSLE* и *MedAE*. На изображениях поверхности функционалов невязки (рис. 2а, 2г) черными точками также изображена цепь поисковых агентов на 60 итерации.

Сопоставление результатов оптимизации выполняется на качественном уровне. Иллюстрации истории позиций поисковых агентов (рис. 2б, 2д) в SSA на всех итерациях показывают, как алгоритм исследует и осваивает пространство поиска, проверяя вероятные модели поиска во всей цепи. При рассмотрении истории позиций видно, что в ходе стохастического процесса с роевым поведением агентов выбирается наиболее перспективные области пространства поиска. Отметим, что история поиска напрямую зависит от модальности функционала невязки, а также принимает случайные формы при каждом запуске. Но тем не менее, алгоритм SSA может направлять поиск в перспективные области пространства пропорционально уровню сложности задачи. Кроме того, такое поведение способствует избеганию локальных оптимумов.

При рассмотрении истории поиска нельзя отделить процесс разведки от нападения. Их легко отделить на кривых сходимости и точности (рис. 2в, 2е), где после первой половины итераций происходит резкий спад значений, после которых флуктуация цепи и изменения параметров модели становятся незначительными. При этом анализируется весь диапазон от минимального до максимального значения невязки и точности из 100 независимых запусков решения обратной задачи.

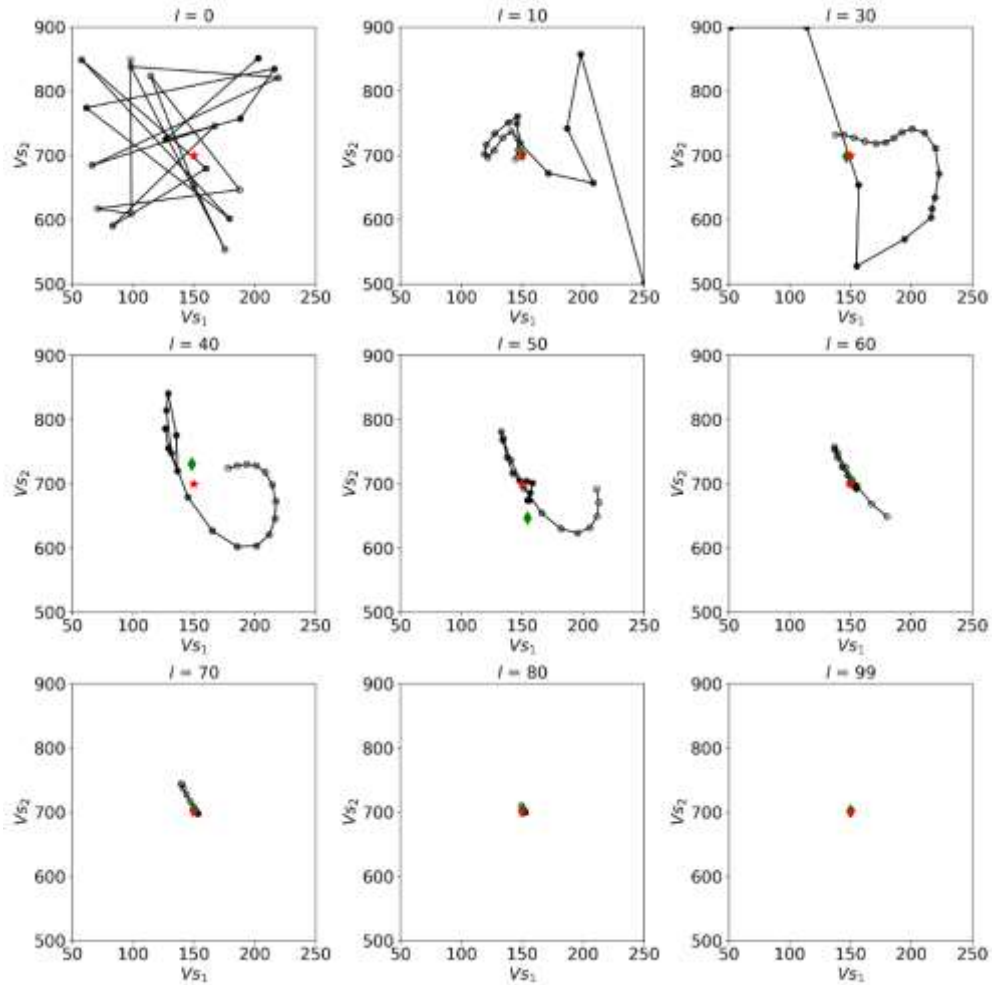


Рис. 1. Движение цепочки сальпов в ходе итерационного решения двумерной оптимизационной задачи в пространстве параметров $V_S^1 - V_S^2$ при $L = 100$ (количество итераций), $K = 20$ (число агентов), $M = 1$ (количество лидеров).

Поведение решения значительно отличается при использовании функционалов из разных групп: для *MSLE* наблюдаются существенно меньшие значения кривых сходимости и точности, чем для метрики *MedAE*. Кривые сходимости показывают, что для всех независимых запусков наблюдается общая тенденция к снижению, т.е. качество моделей улучшается пропорционально количеству итераций и задача минимизации решается успешно. Другой закономерностью является ухудшение невязки (точности) на начальных этапах оптимизации. Это связано с эффектом разведки, при котором сальпы случайно перемещаются по пространству. Но при приближении к этапу нападения кривая сходимости плавно декретируется пропорционально числу итераций.

Такое поведение кривых сходимости свидетельствует о многомодальности (о наличии множества локальных оптимумов) семимерного функционала невязки. Для унимодальных функционалов кривая сходимости очень плавная и устойчивая. Однако в случае многомодальных и составных функций SSA может не показывать никаких улучшений в течение нескольких итераций. Это объясняется тем, что в таких задачах SSA делает упор на разведку, что иногда приводит к выборке неперспективных областей, но повышает эффективность избегания локального оптимума. В связи с этой особенностью рекомендуется выполнять серию независимых запусков решения обратной задачи. Как было отмечено в описании алгоритма, рекомендуется использовать 20 независимых запусков решений [Song et. al, 2015].

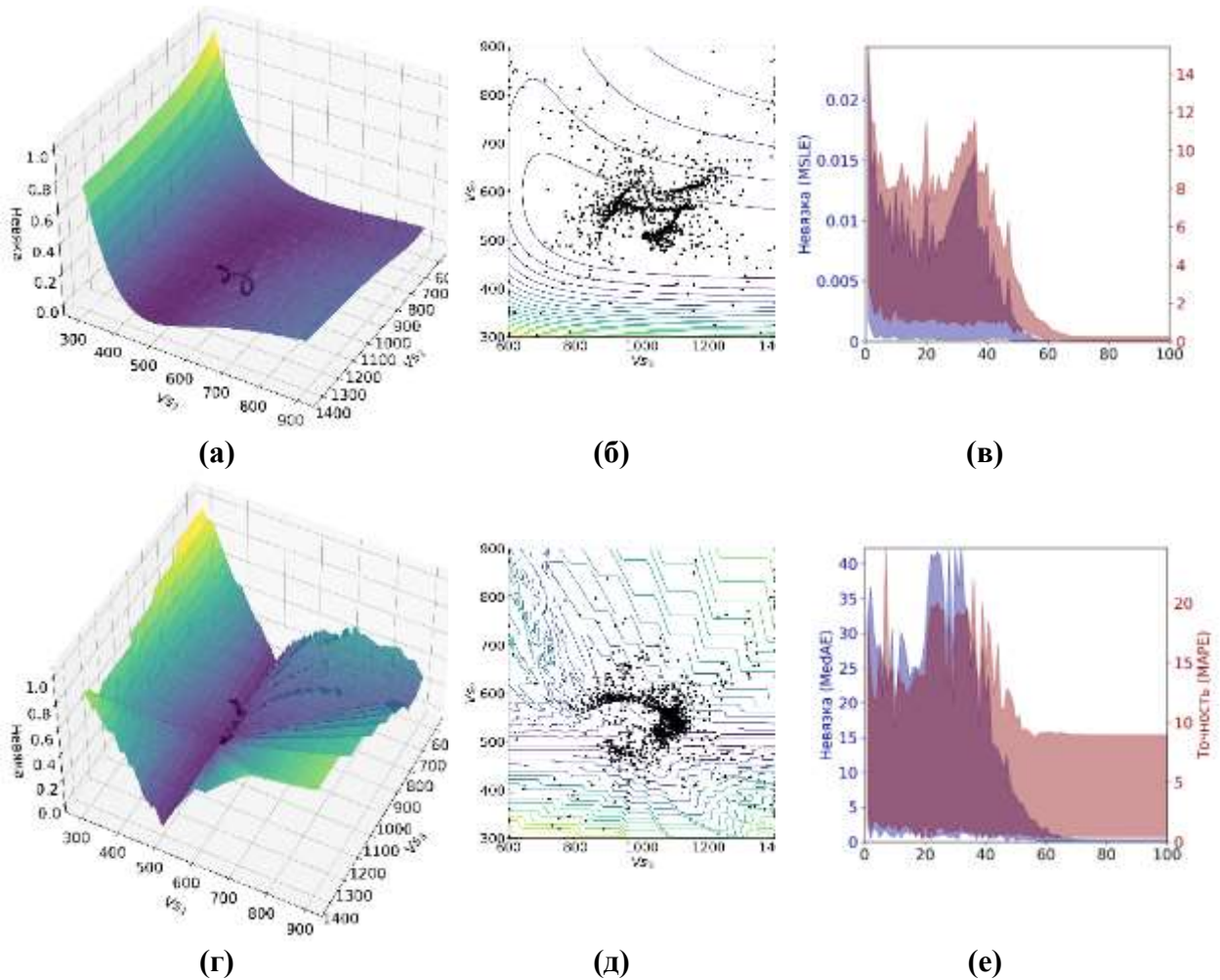


Рис. 2. Результат решения обратной задачи А на основе алгоритма SSA: поверхность функционала невязки (а, г), история позиций поисковых агентов (б, д), кривые сходимости и точности (в, е) для метрик $MSLE$ и $MedAE$.

Результатом решения рассматриваемой обратной задачи являются 100 независимых скоростных моделей S-волны. Анализ параметров распределения восстановленных значений характеризует устойчивость решения (области эквивалентности результатов). Для сопоставления области эквивалентности результатов также выполняется решение обратной задачи с помощью оптимизатора GWO [Song et al., 2015]. При этом используются такие же значения гиперпараметров: число поисковых агентов – 70, количество итераций – 100. Функционал невязки в обоих случаях – $MSLE$.

Так, на рис. 3 представлены результаты решения обратной задачи, полученные с помощью алгоритмов SSA (рис. 3а) и GWO (рис. 3б) для модели с повышением скорости по глубине. По распределениям восстановленных значений V_S и h рассчитывается математическое ожидание (\mathbb{E}) и стандартное отклонение (D), приведенные в табл. 1. Для такой статистически значимой выборки модель, полученная расчетом математического ожидания (красная пунктирная линия на рис. 3а и 3б), будет являться наилучшим решением обратной задачи. Для алгоритма SSA ошибка MAPE, измеряющая отклонение между восстановленной скоростной моделью, определенной на основе математического ожидания распределений параметров, и истинной скоростной моделью, составляет 0.1% и 0.3% при подборе V_S и h , соответственно. Эти же значения для алгоритма GWO: 0.1% и 0.0% соответственно. Как видно, ошибки восстановления при использовании обоих алгоритмов незначительные. Однако, алгоритм SSA обеспечивает более низкий разброс решения, определяемый стандартным отклонением восстановленных параметров, которое в среднем на 35% меньше для V_S и на 58% меньше для h , чем алгоритм GWO.

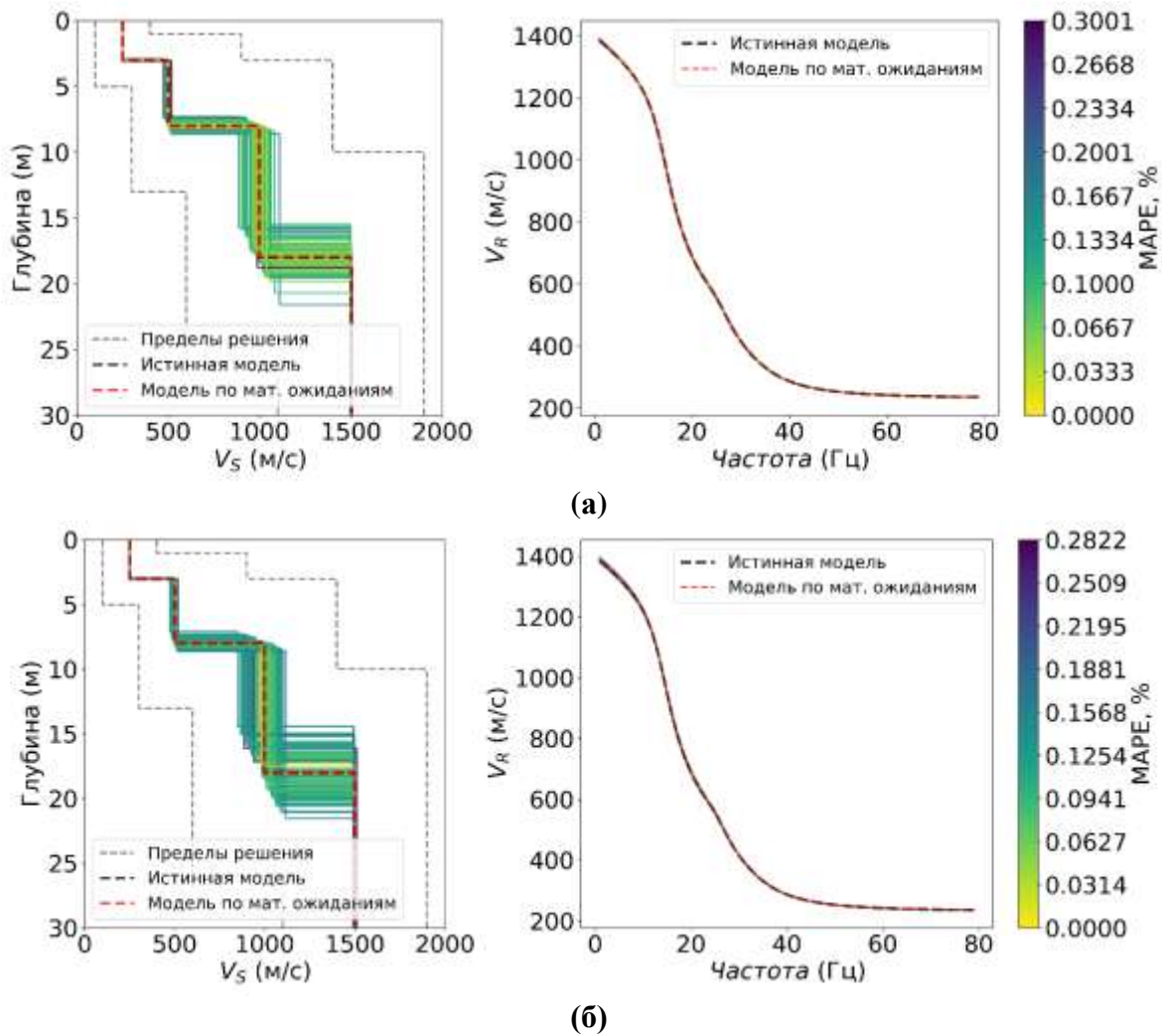


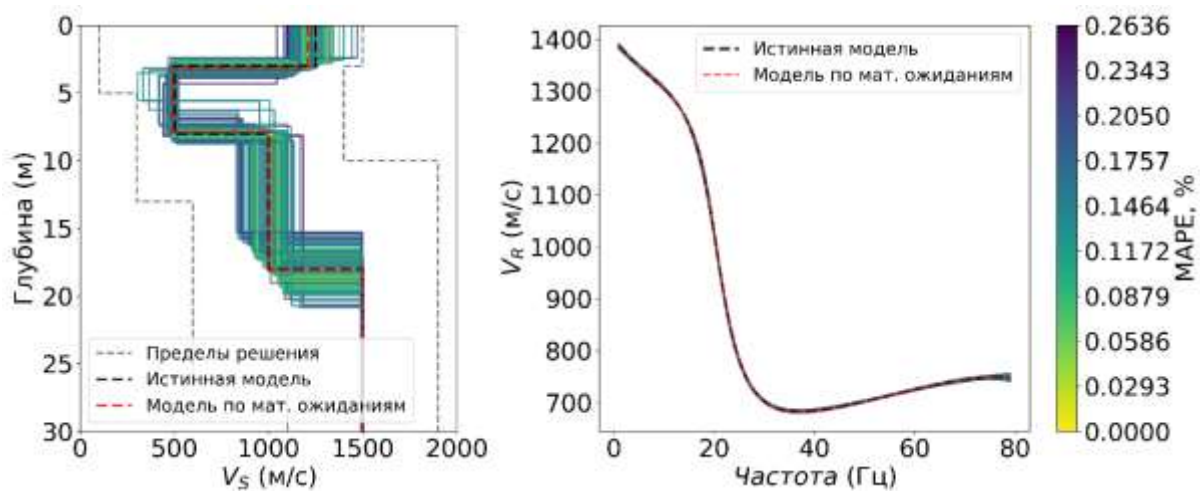
Рис. 3. Результат решения обратной задачи для скоростной модели с повышением скорости по глубине – скоростные модели (слева) и дисперсионные кривые фазовых скоростей (справа), полученные на основе алгоритма SSA (а) и GWO (б). В данном случае положение истинной (черной пунктирной кривой) и восстановленной (красной пунктирной кривой) скоростной модели совпадает.

Результат решения обратной задачи по восстановлению скоростной модели с наличием низкоскоростного слоя с использованием алгоритма SSA изображен на рис. 4а и с помощью оптимизатора GWO на рис. 4б. Статистические параметры восстановленных значений представлены в табл. 2. Ошибка MAPE между скоростной моделью, определяемой математическим ожиданием распределений восстановленных параметров, и истинной скоростной моделью при использовании алгоритма SSA составляет 1.6% и 4.9% при подборе V_S и h , соответственно. Эти же значения для алгоритма GWO: 3.4% и 7.3% соответственно. В основном значения ошибок также являются незначительными, однако, отметим, что в данном случае алгоритм GWO при восстановлении мощности выходит за пределы 5%-го стандарта, принятого во многих программных пакетах, тогда как результат SSA остается в пределах этой погрешности. В этом случае алгоритм SSA также обеспечивает меньший разброс решения: стандартное отклонение восстановленных параметров в среднем на 40% меньше для V_S и на 47% меньше для h , чем для алгоритма GWO.

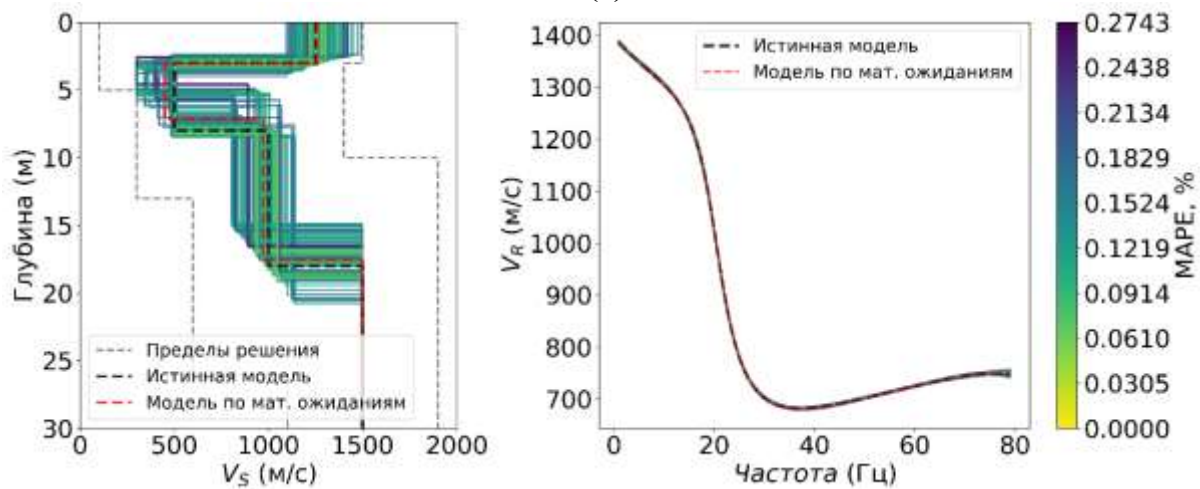
Таблица 1. Статистические характеристики (математическое ожидание – E и стандартное отклонение – D) восстановленных значений V_S и h при решении обратной задачи для модели А с помощью алгоритмов SSA и GWO.

Слой	Истинные значения		$E[V_S]$		$D[V_S]$		$E[h]$		$D[h]$	
	V_S (м/с)	h (м)	SSA	GWO	SSA	GWO	SSA	GWO	SSA	GWO
1	250	3	250.0	250.0	0.3	0.3	3.0	3.0	0.0	0.0
2	500	5	500.8	500.3	7.5	9.5	5.0	5.0	0.2	0.3
3	1000	10	996.8	997.0	40.0	63.7	9.9	10.0	0.9	1.5
4	1500	∞	1500	1499.5	2.1	3.3	–	–	–	–

Результаты синтетических экспериментов показывают, что для рассмотренных скоростных моделей в среднем ошибки восстановления не превышают 1.5% и 2.1% для V_S и h , соответственно, при использовании алгоритма SSA и не превышают 1.6% и 2.9% для V_S и h , соответственно, при использовании алгоритма GWO, что гарантирует приемлемую точность решения обратной задачи для обоих подходов. При этом, неопределенности результатов инверсии, полученных с помощью алгоритма SSA, значительно меньше (в среднем меньше на 48% для V_S и на 64% h), чем при использовании алгоритма GWO.



(a)



(b)

Рис. 4. Результат решения обратной задачи для модели В – скоростные модели (слева) и дисперсионные кривые фазовых скоростей (справа), полученные на основе алгоритма SSA (а) и GWO (б).

Таблица 4. Статистические характеристики (математическое ожидание – E и стандартное отклонение – D) восстановленных значений V_S и h при решении обратной задачи для модели В с помощью алгоритмов SSA и GWO.

Слой	Истинные значения		$E[V_S]$		$D[V_S]$		$E[h]$		$D[h]$	
	V_S (м/с)	h (м)	SSA	GWO	SSA	GWO	SSA	GWO	SSA	GWO
1	1250	3	1209.4	1257.3	78.1	91.7	3.2	3.0	0.3	0.4
2	500	5	486.6	448.9	32.7	75.7	4.7	4.1	0.6	1.2
3	1000	10	1007.2	971.1	79.2	87.1	10.2	10.4	1.1	1.2
4	1500	∞	1500.0	1498.4	2.3	2.3	–	–	–	–

Таким образом, алгоритм SSA адаптирован для решения задачи инверсии дисперсионных кривых поверхностных волн. Источником пищи «роя салп» здесь является позиция значений векторов скорости S-волны и мощности восстанавливаемой модели в многомерном пространстве функционала невязки, определяемого выбранной метрикой. Алгоритм SSA показывает способность эффективно находить глобальный оптимум, даже когда он изменяется в процессе оптимизации.

Анализ истории позиции поисковых агентов показывает, что алгоритм SSA эффективно исследует пространство решения, выбирая перспективные области, что помогает избегать локальных оптимумов. Поведение кривых сходимости и точности указывает на многомодальность функционала невязки, и в таких случаях SSA может не показывать улучшений в начальных итерациях. Это связано с этапом разведки, на котором поисковые агенты просто случайно перемещаются по пространству. Но при приближении к этапу нападения кривая сходимости плавно декретируется пропорционально числу итераций. Это позволяет избегать локальных оптимумов и повышает эффективность алгоритма. Исследование подчеркивает важность выбора подходящей метрики функционала невязки в задачах оптимизации и рекомендует проводить серию независимых запусков решения обратной задачи для учета стохастической природы алгоритма SSA.

Оба алгоритма, GWO (Grey Wolf Optimizer) и SSA (Salp Swarm Algorithm), демонстрируют хорошие результаты при решении обратной задачи SWI для восстановления скоростных моделей S-волн. Они обеспечивают высокую точность, с незначительными ошибками восстановления параметров, как скорости S-волны, так и мощности слоев. Однако, стоит отметить, что SSA превосходит GWO по стабильности и устойчивости решения, что проявляется в меньшем разбросе результатов и меньшей средней ошибке MAPE. Важно подчеркнуть, что, несмотря на эффективность обоих алгоритмов, они являются вычислительно требовательными, что необходимо учитывать при обработке больших объемов данных или при работе с моделями высокой сложности.

Список литературы

- 1 Аки К., Ричардс П. Количественная сейсмология: Теория и методы. Т. 1. Пер. С англ. – М.: Мир, 1983. – 520 с., ил.
- 2 Белоусов А. В. Стандартные оценки качества полевого сейсмического материала // Приборы и системы разведочной геофизики. – 2011. – Т. 37. – №. 3. – С. 31-36.
- 3 Викторов И. А. Звуковые поверхностные волны в твёрдых телах. – 1981.
- 4 Яблоков А. В. и др. Развитие методов инженерной сейсморазведки на основе данных поверхностных волн // 12th Conference and Exhibition Engineering Geophysics 2016. – European Association of Geoscientists & Engineers, 2016. – С. ср-482-00060.
- 5 Constable, S. C., R. L. Parker, and C. Constable. (1987). Occam's inversion: A practical algorithm for generating smooth models from electromagnetic sounding data. *Geophysics* 52(3), 289–300.
- 6 Cox B.R., Teague D.P. Layering ratios: a systematic approach to the inversion of surface wave data in the absence of a priori information // *Geophysical Journal International*, 2016, vol. 207, no. 1, pp. 422-438.
- 7 Garcia D. Robust smoothing of gridded data in one and higher dimensions with missing values // *Computational statistics & data analysis*. – 2010. – Т. 54. – №. 4. – С. 1167-1178.
- 8 Haskell, N. A. (1953). The dispersion of surface waves on multilayered media. *Bull Seismol Soc Am* 43(1), 17–34.
- 9 Kulesh M., Holschneider M., Diallo M. S., Xie Q., Scherbaum F. Modeling of wave dispersion using continuous wavelet transforms // *Pure and Applied Geophysics*. – 2005. – Т. 162. – №.
- 10 Lai C. G., Rix G. J. Simultaneous inversion of Rayleigh phase velocity and attenuation for near-surface site characterization. – School of Civil and Environmental Engineering, Georgia Institute of Technology, 1998. – С. 258.
- 11 Lai, C. G. (2005). Surface waves in dissipative media: Forward and inverse modelling. In: *Surface Waves in Geomechanics: Direct and Inverse Modelling for Soil and Rocks* (C. G. Lai and K. Wilmski, eds.). New York: Springer-Verlag, p. 385.
- 12 Mirjalili S., Gandomi A.H., Mirjalili S.Z., Saremi S., Faris H., Mirjalili S.M. Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems // *Advances in engineering software*, 2017, vol. 114, pp. 163-191.
- 13 Roohollah A. Surface Wave Analysis and its Application to the Calculation of Converted Wave Static Corrections: Дис. – University of Calgary, 2013.
- 14 Serdyukov A. S., Yablokov A. V., Duchkov A. A., Azarov A. A., Baranov V. D. Slant f-k transform of multichannel seismic surface wave data // *Geophysics*. – 2019. – Т. 84. – №. 1. – С. A19-A24.
- 15 Smith M., Perry G., Stein J., Bertrand A., Yu G. Extending seismic bandwidth using the continuous wavelet transform // *first break*. – 2008. – Т. 26. – №. 6. – С. 97-102.
- 16 Song X., Tang L., Zhao S., Zhang X., Li L., Huang J., Cai W., Grey Wolf Optimizer for parameter estimation in surface waves, *Soil Dynamics and Earthquake Engineering*, 2015, vol. 75, pp. 147-157.
- 17 Spitz S. Seismic trace interpolation in the FX domain // *Geophysics*. – 1991. – Т. 56. – №. 6. – С. 785-794.

18 Stockwell R. G., Mansinha L., Lowe R. P. Localization of the complex spectrum: the S transform //IEEE transactions on signal processing. – 1996. – T. 44. – №. 4. – C. 998-1001.

19 Thomson, W. T. (1950). Transmission of elastic waves through a stratified solid medium. *J Appl Phys* 21, 89–93.

20 Virieux J. P-SV wave propagation in heterogeneous media: Velocity-stress finite-difference method //Geophysics. – 1986. – T. 51. – №. 4. – C. 889-901.

21 Yablokov A., Lugovtsova Y., Serdyukov A. Uncertainty quantification of multimodal surface wave inversion using artificial neural networks // Geophysics, 2023, vol. 88, no. 2, pp. KS1-KS11.

22 Yablokov A.V., Serdyukov A.S., Loginov G.N., Baranov V.D. An artificial neural network approach for the inversion of surface wave dispersion curves // Geophysical Prospecting, 2021, vol. 69, no. 7, pp. 1405-1432.