

Сравнительный анализ численных методов решения обратной задачи об источнике распространения волн

Губер Алексей Владимирович

Шишленин Максим Александрович

XVII Международная конференция «Забабахинские научные
чтения»

ИМ СО РАН

20 мая 2025 г.

- 1 Введение
- 2 Матричный метод
- 3 PINN
- 4 Градиентный метод
- 5 Заключение

Восстановление источников акустических волн является актуальной и практически важной задачей:

- 1 медицинская акустическая томография (источник локализован, точный вид сигнала неизвестен);
- 2 морская акустика (внутренние волны, волны цунами и пр.);
- 3 геофизика (источники землетрясений, микросейсмический мониторинг, при условии, что параметры среды известны).

Постановка прямой задачи

Рассмотрим некоторый волновой процесс из пунктов 1, 2, 3 в области $\Omega = \{(x, y) : x \in (0, L_x), y \in (0, L_y)\}$:

$$u_{tt} = \operatorname{div}(c^2(x, y) \nabla u), \quad (x, y) \in \Omega, \quad t \in (0, T),$$

$$u|_{t=0} = q(x, y), \quad u_t|_{t=0} = 0,$$

$$u|_{\partial\Omega} = 0.$$

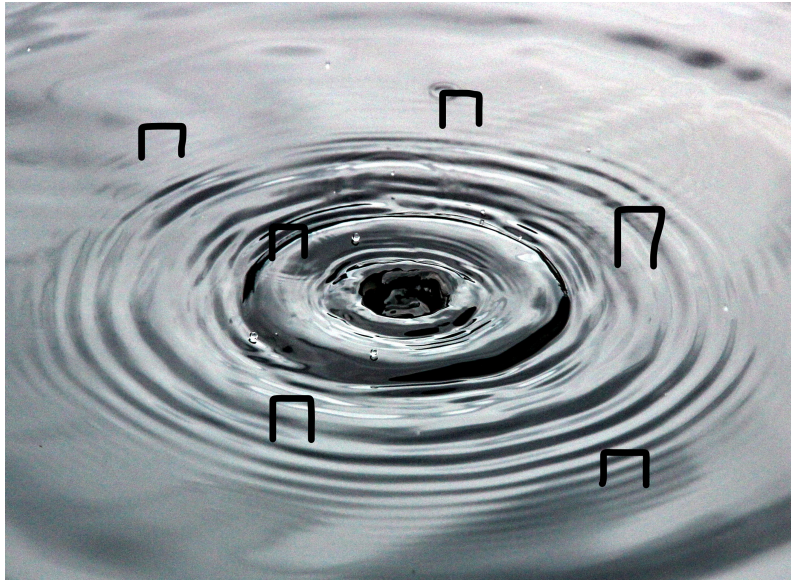
Здесь $c(x, y)$ – скорость распространения волн, $u(x, y, t)$ – давление.

Постановка обратной задачи

Обратная задача состоит в нахождении функции $q(x, y)$ из дополнительной информации:

$$u(x_n, y_n, t) = f_n(t), \quad n = \overline{1, N}.$$

Здесь (x_n, y_n) – координаты приёмников, N – количество приёмников.



Реализовать основные методы решения поставленной обратной задачи об источнике волн и провести их сравнительный анализ.

Реализовать основные методы решения поставленной обратной задачи об источнике волн и провести их сравнительный анализ.

Рассматриваемые методы:

- матричный метод;
- модифицированный матричный метод с Tensor-Train разложением;
- PINN;
- градиентный метод.

Метод заключается в сведении дискретной задачи к СЛАУ большой размерности с плохо обусловленной матрицей:

$$A_{total}\vec{Q} = \vec{F}.$$

Здесь матрица A_{total} имеет размер $(N_x + 1)(N_y + 1) \times N_t N$, N – количество приёмников.

М. А. Шишленин. Матричный метод в задачах определения источников колебаний. Сиб. электрон. матем. изв., 11 (2014), С.161–С.171.

Эта СЛАУ может быть решена методом усечённого SVD:

$$U\Sigma V^T \vec{q} = \vec{f}$$

$$\Sigma \vec{z} = U^T \vec{f} = \vec{g}$$

Тогда вектор \vec{z} может быть найден: $z_j = \frac{g_j}{\sigma_j}$ и если $\sigma_j = 0$, тогда $z_j = 0$.

С другой стороны решение СЛАУ $A\vec{x} = \vec{b}$ может быть получено методом сопряжённых градиентов:

$$\vec{p}^{(0)} = 0, s = 1$$

$$\vec{r}^{(s)} = \begin{cases} A^T (A\vec{x}^{(s)} - \vec{b}), & \text{если } s = 1, \\ \vec{r}^{(s-1)} + \frac{\vec{q}^{(s-1)}}{(\vec{p}^{(s-1)}, \vec{q}^{(s-1)})}, & \text{если } s \geq 2, \end{cases} \quad (1)$$

$$\vec{p}^{(s)} = \vec{p}^{(s-1)} + \frac{\vec{r}^{(s)}}{(\vec{r}^{(s)}, \vec{r}^{(s)})}$$

$$\vec{q}^s = A^T (A\vec{p}^{(s)}),$$

$$\vec{x}^{(s+1)} = \vec{x}^{(s)} - \frac{\vec{p}^{(s)}}{(\vec{p}^{(s)}, \vec{q}^{(s)})},$$

$$s = s + 1.$$

Lukyanenko, D.; Shinkarev, V.; Yagola, A. Accounting for Round-Off Errors When Using Gradient Minimization Methods. *Algorithms* 2022, 15, 324. <https://doi.org/10.3390/a15090324>

Исходя из предположения, что все операции выполняются корректно, имеем

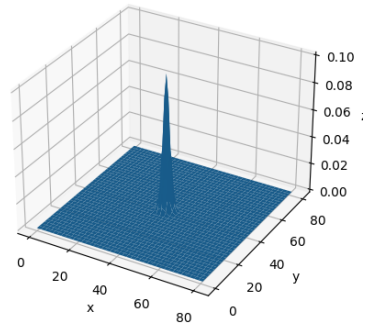
$$\vec{x}^{res} = \vec{x}^{(N_t N + 1)}.$$

Но можно учесть ошибку арифметических операций и использовать её в критерии остановки алгоритма:

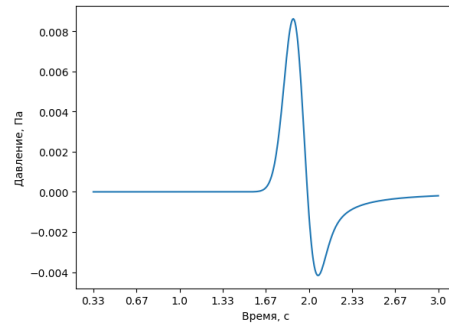
$$jj\vec{r}^{(s)}jj^2 > \sigma_s^2 \Delta^2,$$

где σ_s^2 – дисперсия ошибки $jj\vec{r}^{(s)}jj^2$ и $\Delta = 10^{-16}$.

Матричный метод

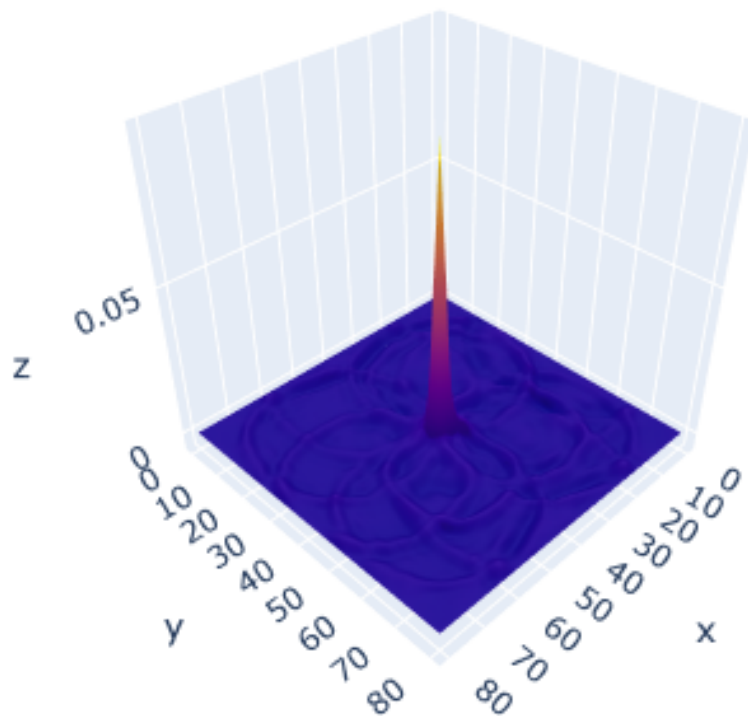


. 1:



. 2:

()



. 3:

Матричный метод. Tensor-Train разложение

Тензор $A = A(i_1, \dots, i_d)$ – это многомерный массив, где $i_k = \overline{1, n_k}$.

Разложение вида $A(i_1, \dots, i_d) = G_1(i_1) \dots G_d(i_d)$, где матрица $G_k(i_k)$ имеет размер $r_{k-1} \times r_k$ называется Tensor-Train разложением (форматом).

Oseledets, Ivan. (2011). Tensor-Train Decomposition. SIAM J. Scientific Computing. 33. 2295-2317. 10.1137/090752286.

Возможные операции над тензорами в этом формате:

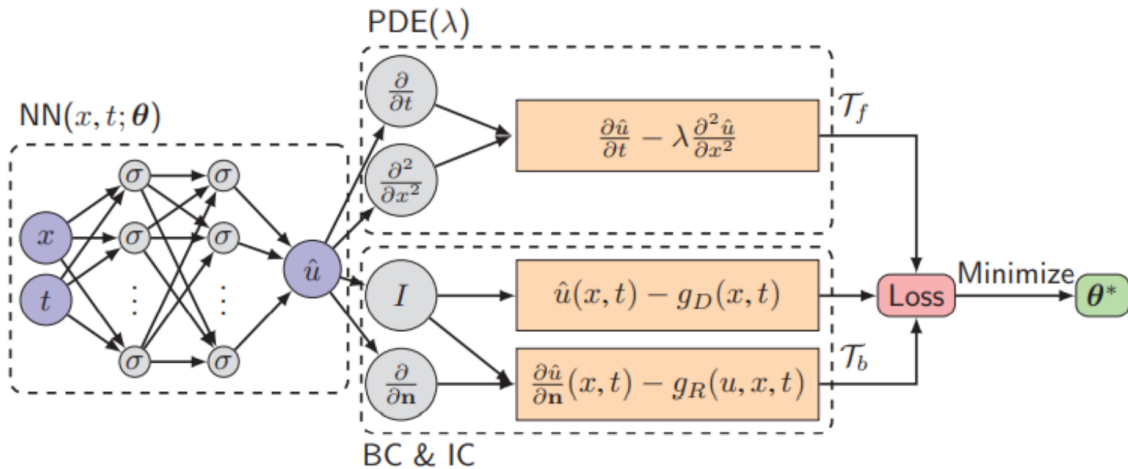
- 1 сложение;
- 2 умножение;
- 3 округление;
- 4 восстановление по $O(dnr^2)$ элементам.

Матричный метод. Tensor-Train разложение

Операция	Сложность
Сложение тензоров	$O(dnr^2 + dr^4)$
Многомерная свертка	$O(dnr + dr^3)$
Поэлементное произведение	$O(dnr^4)$
Скалярное произведение	$O(dnr^2 + dr^4)$
Норма	$O(dnr^2 + dr^4)$

. 4:

Физически обусловленные нейронные сети (PINN) представляют собой тип аппроксиматоров функций общего назначения, которые могут внедрять знания о любых физических законах, управляющих заданным набором данных, в процесс обучения и могут быть описаны с помощью уравнений в частных производных.



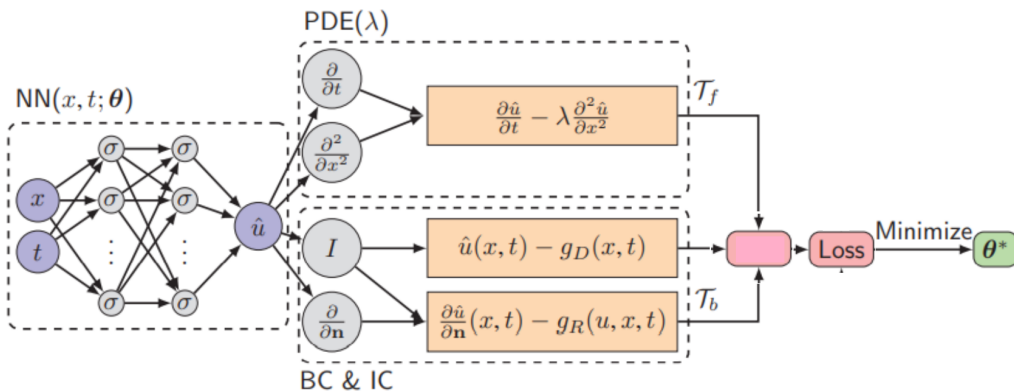
. 5: PINN.

В таком виде эта модель не позволяет определить начальные условия, как нам надо.

Пусть решение уравнения имеет вид

$$\Phi(x, t, \theta) = A(x, y, t) + Z(x, y, t)F(x, t, \theta).$$

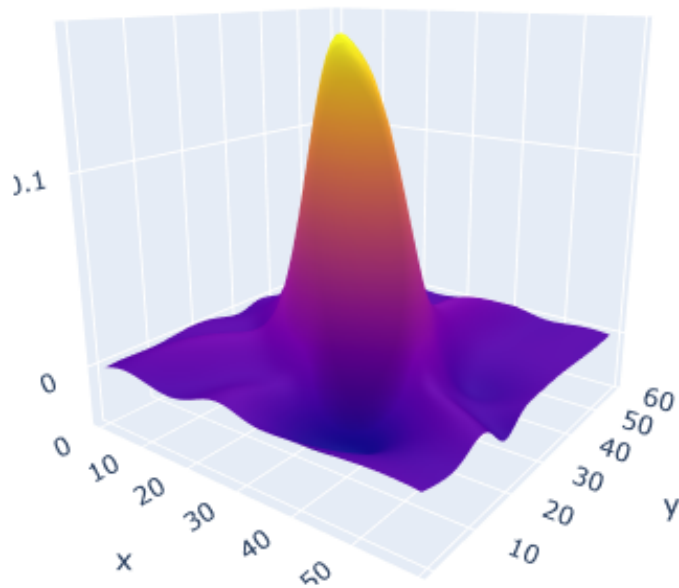
Функция $A(x, y, t)$ должна удовлетворять начальным и граничным условиям и $Z(x, y, t) = 0$ на границе. Значит, искомая функция $q(x, y)$ может быть представлена в виде нейронной сети в функции $Z(x, y, t)$.



. 6:

PINN

Результат – решение обратной задачи требует > 10 тыс. точек и 60 тыс. итераций, $MSE = 10^{-7}$.



. 7:

PINN (DeepXDE)

Sirignano, J., Spiliopoulos, K. (2018). DGM: A deep learning algorithm for solving partial differential equations. *Journal of Computational Physics*, 375, 1339-1364.

Решение обратной задачи сводится к минимизации функционала

$$J(q) = \int_0^T \sum_{i=0}^N [u(x_i, y_i, t, q) - f_i(t)]^2 dt \quad \min_q.$$

Тогда решение может быть получено итеративно:

$$q^{(n+1)} = q^{(n)} - \alpha_n J'(q^{(n)}),$$

где $q^{(0)}$ – начальное приближение, α_n – параметр спуска.

В нашем случае имеем

$$J^0(q) = c^{-2}(x, y) \varphi_t|_{t=0},$$

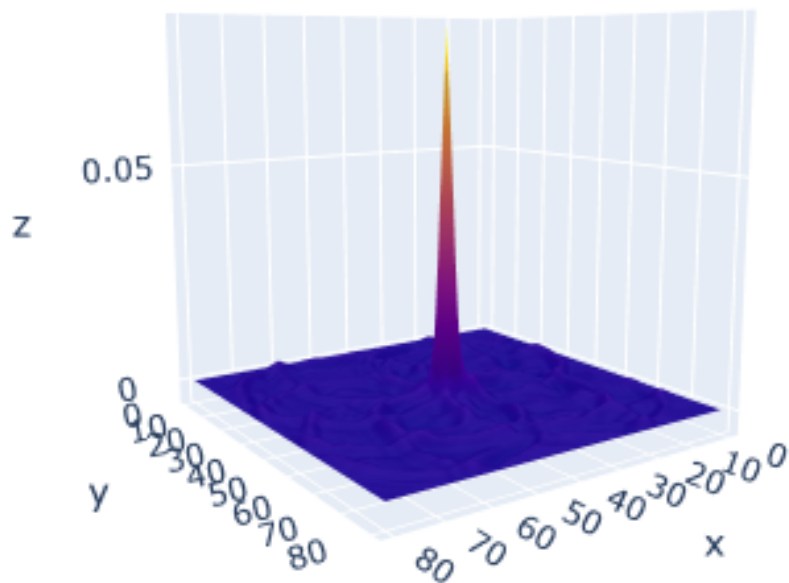
где φ – решение сопряжённой задачи

$$c^{-2}(x, y) \varphi_{tt} - \Delta \varphi = \sum_{i=1}^N 2 [u(x_i, y_i, t) - f_i(t)] \delta(x - x_i, y - y_i),$$

$$\varphi|_{t=T} = 0,$$

$$\varphi_t|_{t=0} = 0,$$

$$\varphi|_{\partial\Omega} = 0.$$



. 8:

Метод	ит.	$N_{x,y}$	N_t	Время, с	Ошибка
Matrix (SVD)	-	80	90	786	0.006
Matrix (CG)	-	80	90	377	0.006
TT	-	80	90	128	0.007
PINN	70000	-	-	7826	0.028
Gradient	10000	80	90	365	0.011

1:

AMD Ryzen 9 5900X, NVIDIA GeForce RTX 4070

Q&A

alexej.guber@yandex.ru