

Применение физически-информированной нейронной сети для модельных задач гидрологии

Кошелев К.Б., Стрижак С.В.

ИСП РАН, г. Москва

Международная конференция «Марчуковские научные чтения 2024»
(МНЧ-2024)
07-11 октября 2024 г.

Содержание

- 1D уравнения по теории мелкой воды
- Постановка прямой задачи
- Настройка гиперпараметров для прямой задачи
- Результаты обучения PINN
- Постановка обратной задачи. Результаты решения
- 2D стационарные уравнения по модели мелкой воды
- Преимущества и недостатки использования PINN
- Заключение

1D уравнения по теории мелкой воды

уравнение неразрывности

$$\frac{\partial W}{\partial t} + \frac{\partial Q}{\partial x} = q$$

уравнение движения с использованием формулы Маннинга для учета трения о дно

$$\frac{\partial Q}{\partial t} + \frac{\partial}{\partial x} \left(\frac{Q^2}{W} \right) + gW \frac{\partial z}{\partial x} + \frac{gn^2 Q |Q|}{WR^{4/3}} = 0$$

Здесь t – время, $x \in [0, X]$ – продольная координата вдоль русла, W – площадь живого сечения, Q – расход, q – боковая приточность на единицу длины, g – ускорение свободного падения, δ – отметка дна, n – коэффициент шероховатости по формуле Маннинга, R – гидравлический радиус, $z = \delta + H$ – уровень поверхности воды, H – глубина воды.

Полагаются известными следующие функции, определяющие морфологические параметры:

$w = w(x, h)$ $h = h(x, w)$ $n = n(x)$ $\delta = \delta(x)$ $q = q(x, t)$, где w – площадь сечения, соответствующая расстоянию до отметки дна h .

Начальные условия имеют вид: $W(x, 0) = W_0(x)$ $Q(x, 0) = Q_0(x)$

Граничные условия для спокойного течения: $Q(0, t) = \hat{Q}(t)$ и $H(X, t) = \hat{H}(t)$

Постановка прямой задачи

Рассмотрим стационарное течение в прямоугольном канале шириной D с постоянным коэффициентом шероховатости n и заданными строго положительными значениями расхода $Q(x)$ и площади живого сечения $W(x)$.

Для прямоугольного канала величины глубины и гидравлического радиуса вычисляются по формулам:

$$H(x) = \frac{W(x)}{D} \text{ и } R(x) = \frac{W(x)}{D+2 \cdot H(x)}$$

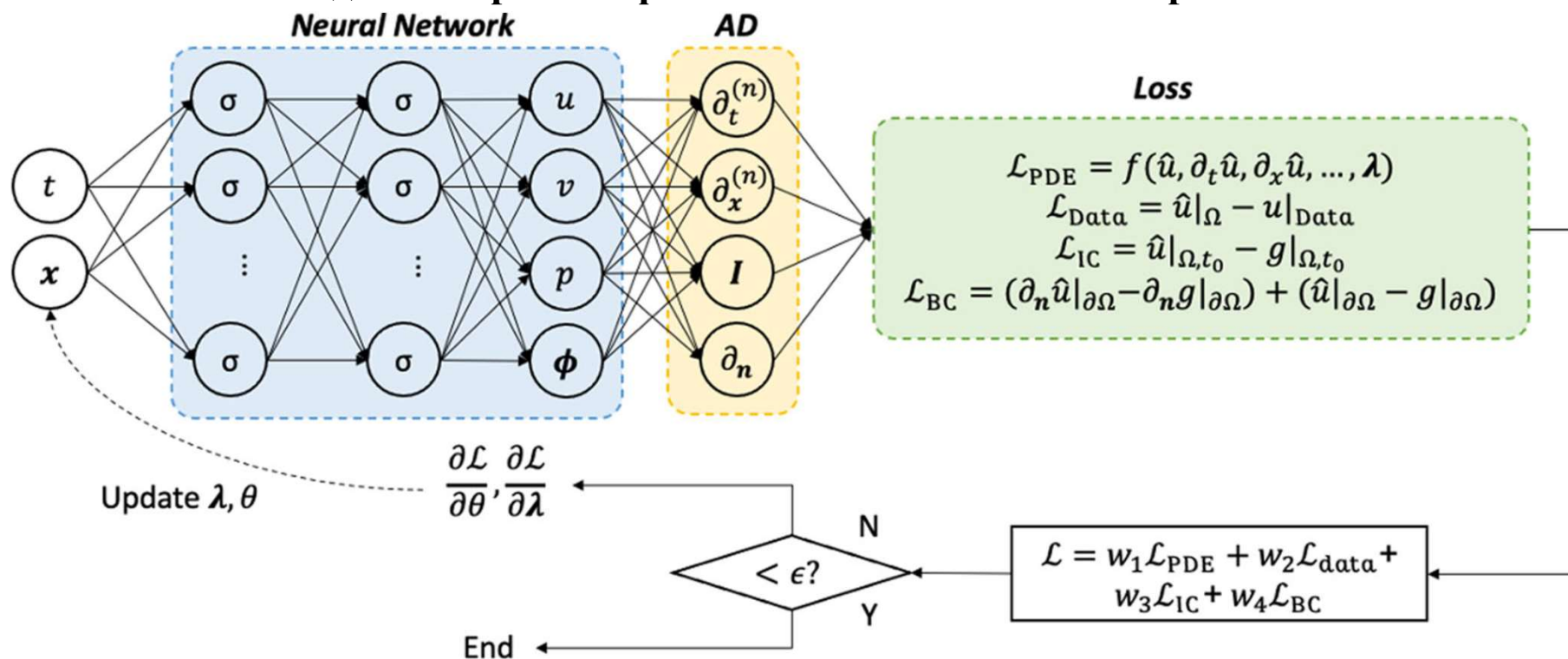
Для выполнения уравнения неразрывности необходимо задать боковую приточность как $q(x) = \frac{dQ}{dx}$

Для выполнения уравнения движения надо вычислить отметку дна по формуле:

$$\delta(x) = H(0) - H(x) - \int_0^x \left(\frac{\frac{d(Q(x)^2/W(x))}{dx}}{gW(x)} + \frac{n^2 Q(x)^2}{W(x)^2 R^{4/3}} \right) dx$$

Для обучения нейронной сети использовались уравнения с приведенными замыкающими соотношениями для боковой приточности и отметки дна.

Блоки для построения физически-обоснованной нейронной сети



Note: $\hat{u} = [u, v, p, \phi]$, $\mathbf{x} = [x, y]$, θ : weights/biases, λ : unknown PDE parameters, $w_i, i = 1, 2, \dots, 4$: weights

PINN (Physics-Informed Neural Networks) состоит из **первого блока** с невязкой (остаточных слагаемых) для дифференциального уравнения, для начальных и граничных условий. Входы для нейронной сети преобразуются в выходы сети. Параметры нейронной сети θ определяется как минимум функции потерь (Loss function).

Второй блок - это сеть с физическими данными, которая берет выходное поле u и вычисляет производную, используя заданные уравнения. Также оцениваются граничные/начальные условия, используются размеченные данные наблюдений (вычислений, эксперимента).

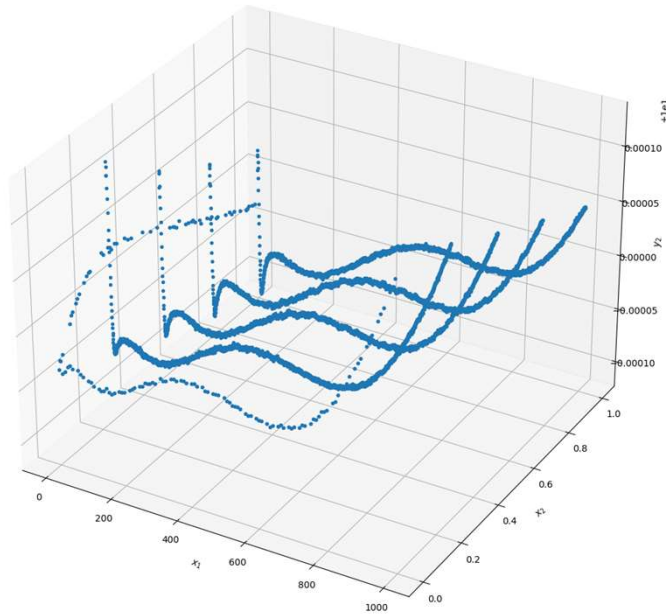
Последним шагом является механизм формирования обратной связи, который минимизирует функцию потерь, используя оптимизатор, в соответствии с некоторой скоростью обучения, чтобы получить параметры NN θ

Настройка гиперпараметров для прямой задачи

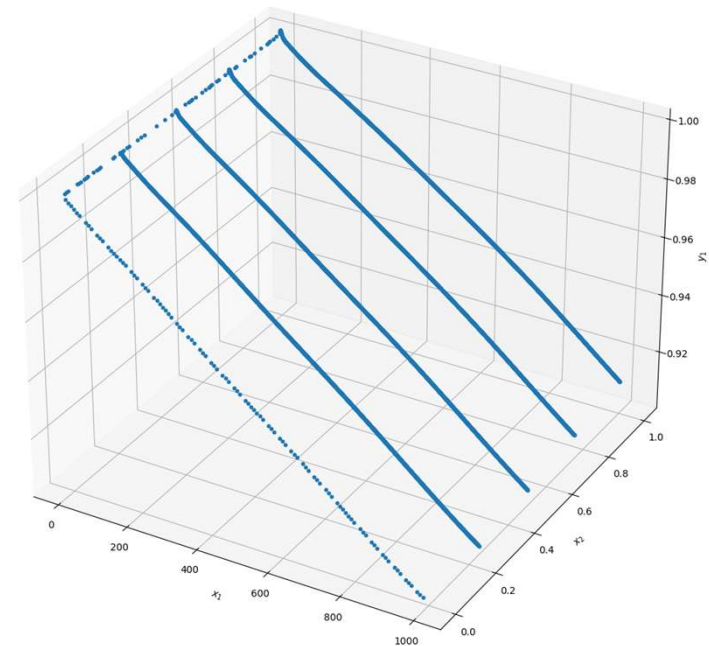
Для решения задачи использовался скрипт на языке python с использованием пакетов DeepXDE и TensorFlow.

```
data = dde.data.TimePDE(geomtime,pde,
    [boundary_condition_Q, boundary_condition_w, initial_condition_Q, initial_condition_w],
    num_domain=25001,
    num_boundary=100,
    num_initial=1000,
    num_test=10000
)
net = dde.nn.FNN([2] + 4 * [100] + [2], "tanh", "Glorot normal")
model = dde.Model(data, net)model.compile("adam", lr=1e-3,
    loss_weights=[1, 1, 10000, 10000, 10000, 10000]
)
model.train(iterations=50000)
model.compile("L-BFGS", loss_weights=[1, 1, 10000, 10000, 10000, 10000])
losshistory, train_state = model.train()
```

Результаты тренировки PINN для 1D уравнений Сен-Венана в прямоугольном канале длиной 1000 м в течение 1 суток



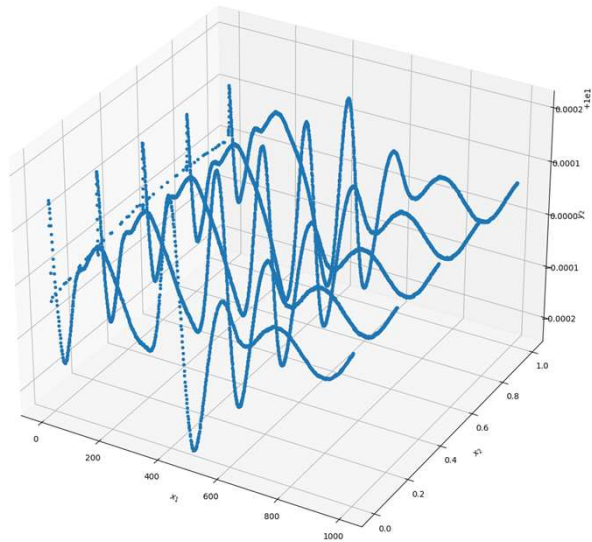
Отклонение полученной площади живого сечения от 10 m^2



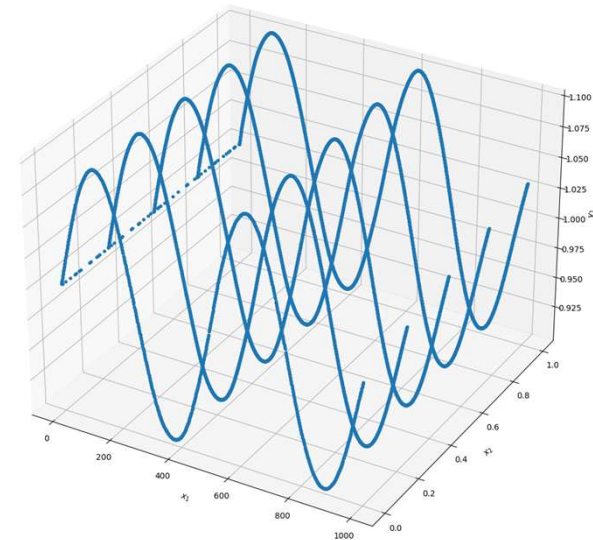
Распределение полученного расхода по длине канала и времени

Уровень дна канала подобран таким образом, чтобы стационарным решением были постоянная площадь живого сечения (10 m^2) и расход, изменяющийся по линейному закону от $1 \text{ m}^3/\text{с}$ до $0.9 \text{ m}^3/\text{с}$.

Результаты тренировки PINN для 1D уравнений Сен-Венана в прямоугольном канале длиной $L=1000$ м в течение 1 суток



Отклонение полученной площади живого сечения от 10 м^2



Распределение полученного расхода по длине канала и времени при $k=4$

Уровень дна канала подобран таким образом, чтобы стационарным решением были постоянная площадь живого сечения (10 м^2) и расход, изменяющийся по закону $Q = 1 + 0,1 \cdot \sin\left(\frac{k\pi x}{L}\right)$, где x – расстояние вдоль русла, k - параметр

Постановка обратной задачи. Результаты решения

Обратная задача была реализована для определения расхода во входном створе Q_0 и коэффициента шероховатости n . Начальное приближение было задано в 2 раза больше искомого.

При нахождении коэффициента шероховатости пришлось уменьшить скорость обучения $lr=0.0001$, при этом ошибка в величине коэффициента шероховатости могла достигать 5%.

2D стационарные уравнения по модели мелкой воды

$$\frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0$$

$$\frac{\partial(hu^2 + 0.5 \cdot gh^2)}{\partial x} + \frac{\partial huv}{\partial x} = -gh \frac{\partial z}{\partial x} - g \frac{n^2 |u| u}{h^{1/3}} + h\mu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

$$\frac{\partial huv}{\partial x} + \frac{\partial(hv^2 + 0.5 \cdot gh^2)}{\partial x} = -gh \frac{\partial z}{\partial y} - g \frac{n^2 |u| v}{h^{1/3}} + h\mu \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right)$$

Уровень поверхности воды

$$H = h + z$$

Поверхность дна канала определяется уравнением

$$z = Z(x, y)$$

Здесь x, y – координаты вдоль и поперек канала, h – глубина воды, u, v – компоненты осредненной по глубине скорости вдоль координат x и y , g – ускорение силы тяжести, n – коэффициент шероховатости поверхности дна по формуле Маннинга, μ – вихревая вязкость.

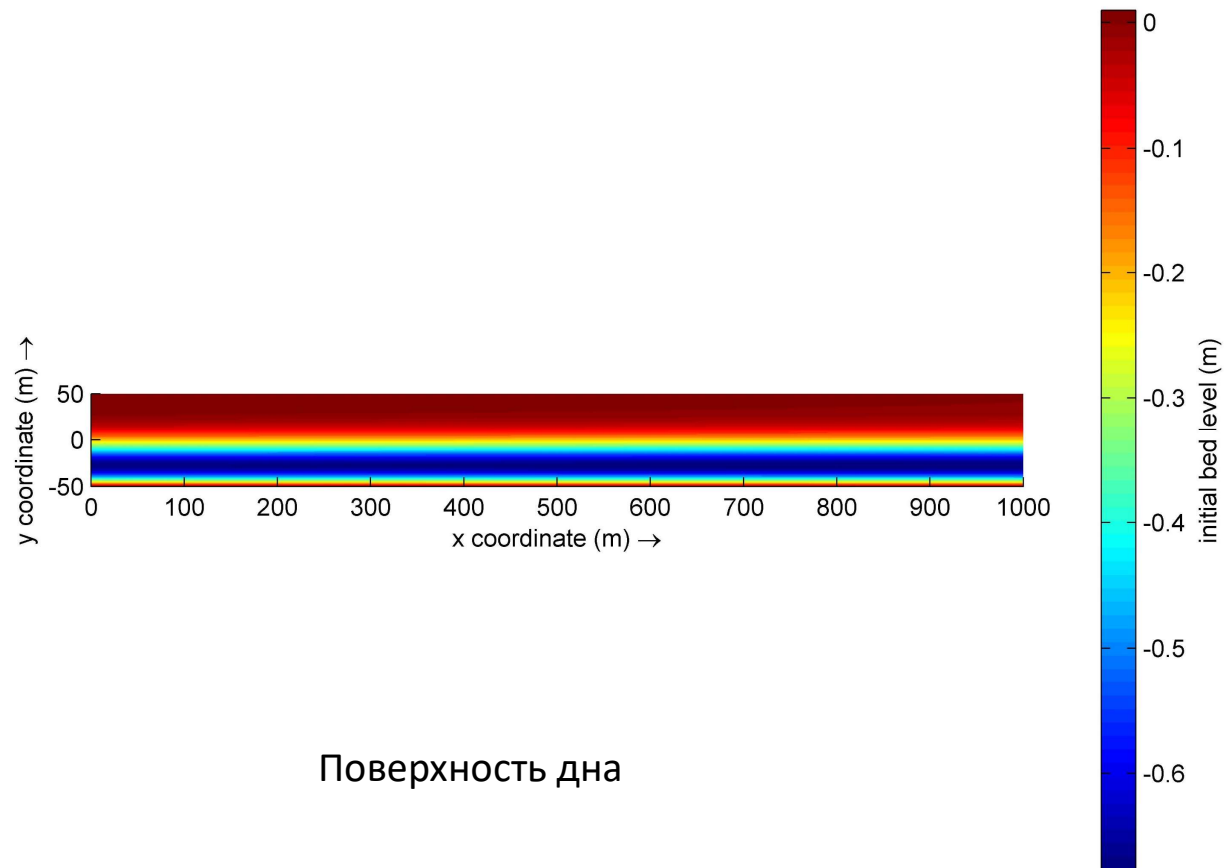
Граничные условия

$$Q_{/x=0} = Q_{inlet}, H_{/x=L} = H_{outlet}$$

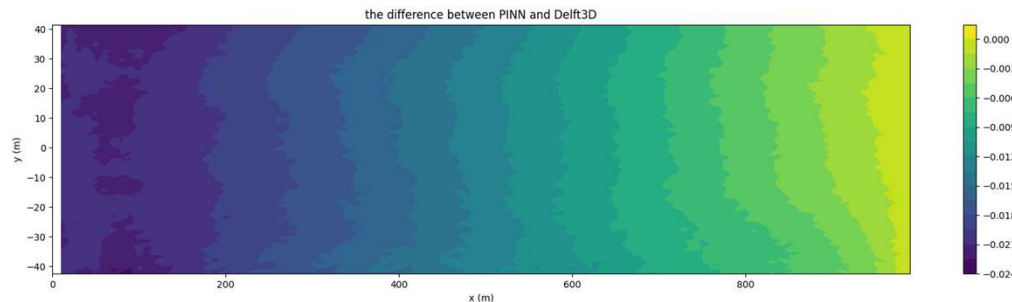
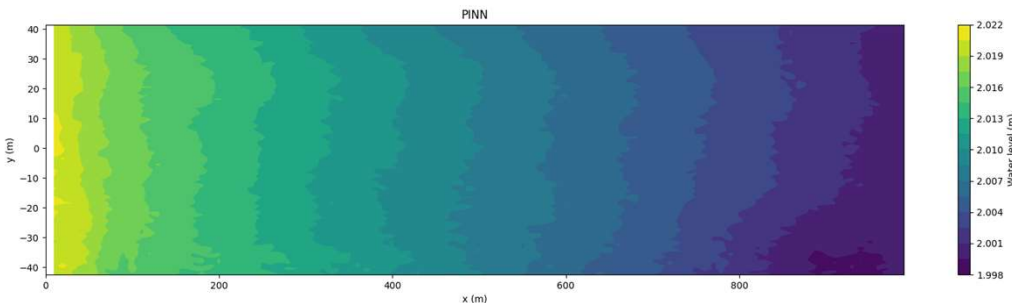
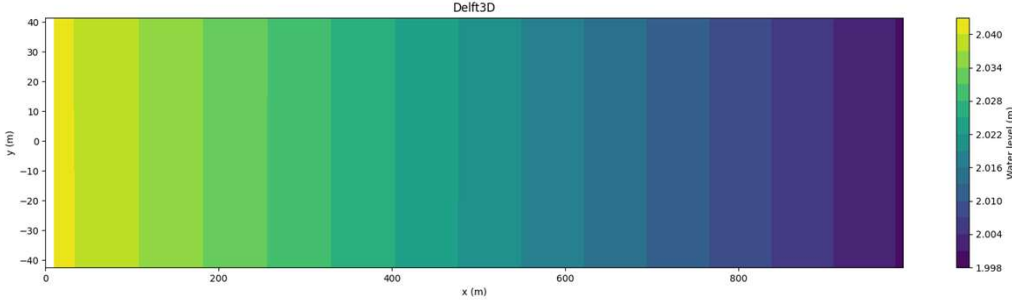
Параметры тестового расчета

$$z = 0.00001 \cdot (L - x) - 0.05 \cdot y \cdot e^{-y^2/1000}$$

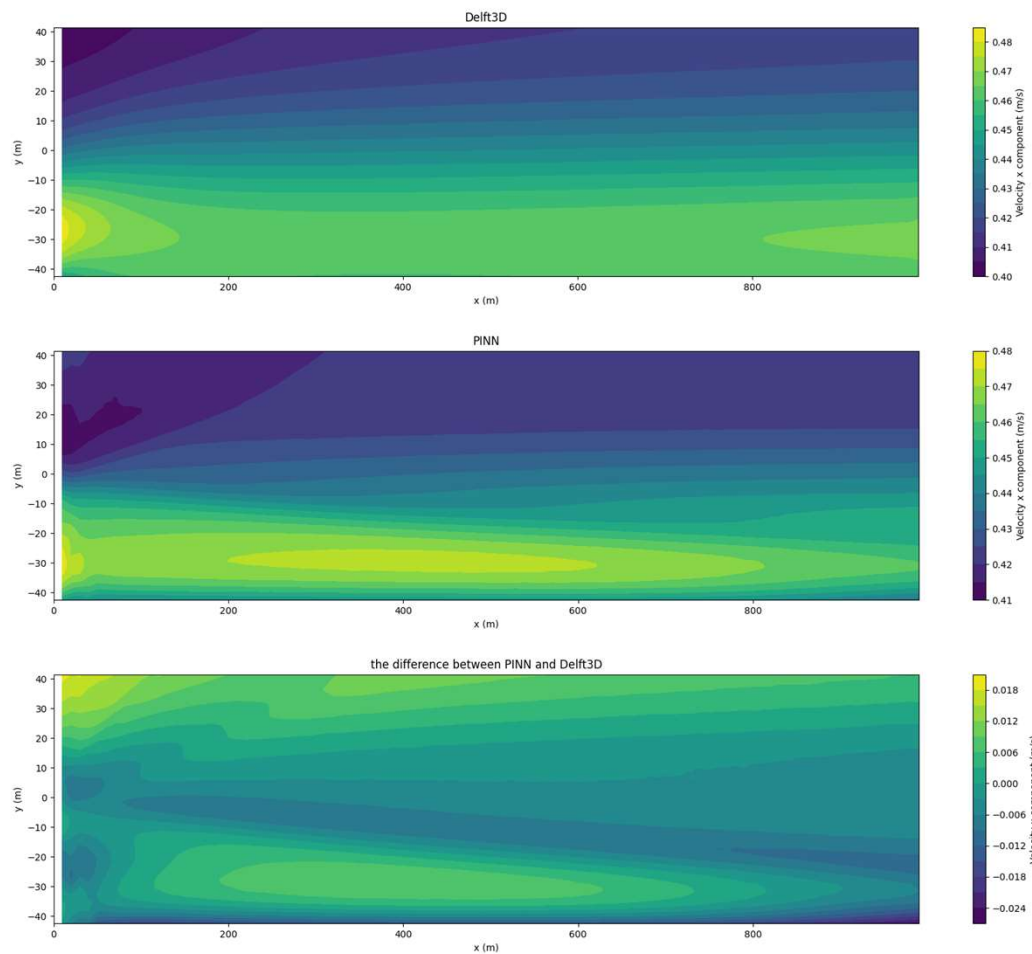
$$L = 1000, B = 100, Q_{/x=0} = 100, H_{/x=L} = 2, g = 9.81, n = 0.025, \mu = 1$$



Сравнительные результаты расчетов. Уровень поверхности воды



Сравнительные результаты расчетов. x компонента скорости



Преимущества и недостатки использования PINN

Зачем надо использовать подход с применением PINN, если все ранее упомянутые задачи можно без труда решить традиционными вычислительными методами?

Преимущества использования PINN

1. Можно обойтись без умения решать краевые задачи.
2. Существует готовый механизм для решения обратных задач.
3. Решение задач с использованием PINN обычно воспринимается положительно.

Недостатки использования PINN

1. Чрезмерные требования к вычислительным ресурсам. Обязательно использование GPU.
2. Кодирование дифференциальных уравнений еще не универсально.
3. Требуется опыт в использовании гиперпараметров.
4. Численный метод решения дифференциальных уравнений недостаточно устойчивый.
5. Поиск необходимых наборов данных для обучения в случае реальных водотоков не является простым.
6. Не удастся воспользоваться традиционными способами отладки компьютерных программ.

Заключение

Преимущества использования PINN пока ограничиваются множеством недостатков и издержек. Только ликвидация или смягчение этих недостатков приведет к распространению использования данного подхода для решения реальных задач.

Спасибо за внимание

