

浅水方程

MG21210021 李庆春

2023 年 9 月 13 日

1 2D 浅水方程简述

$$\begin{cases} \frac{du}{dt} - fv = -g \frac{d\eta}{dx} + \frac{\tau_x}{\rho_0 H} - \kappa u \\ \frac{dv}{dt} + fv = -g \frac{d\eta}{dy} + \frac{\tau_y}{\rho_0 H} - \kappa v \\ \frac{d\eta}{dt} + \frac{d(\eta+H)u}{dx} + \frac{d(\eta+H)v}{dy} = \sigma - w \end{cases} \quad (1)$$

以上是二维浅水方程表达式，其中动量方程是线性的，连续方程是非线性的，自变量 x, y, t 的含义是显然的，应变变量 u 是水平方向的流速， v 是垂直方向的流速， η 是水面动态海拔；其中 $f = f_0 + \beta y$ 是全维度变换的科里奥利斯参数； κ 是跟摩擦相关的系数； $\tau_x \tau_y$ 是跟风应力相关的系数； σ 是跟质量源相关的系数； w 是跟质量汇相关的系数；先不考虑这些额外的系数，将方程 1 简化为：

$$\begin{cases} \frac{du}{dt} = -g \frac{d\eta}{dx} \\ \frac{dv}{dt} = -g \frac{d\eta}{dy} \\ \frac{d\eta}{dt} = -\frac{d(\eta+H)u}{dx} - \frac{d(\eta+H)v}{dy} \end{cases} \quad (2)$$

等价于

$$\begin{cases} \frac{du}{dt} = -g \frac{d\eta}{dx} \\ \frac{dv}{dt} = -g \frac{d\eta}{dy} \\ \frac{d\eta}{dt} = -u \frac{d\eta}{dx} - v \frac{d\eta}{dy} - (\eta + H) \left(\frac{du}{dx} + \frac{dv}{dy} \right) \end{cases} \quad (3)$$

该方程的 CFL 条件为：

$$\begin{cases} dt \leq \frac{\min(dx, dy)}{\sqrt{gH}} \\ \alpha \ll 1 \quad (\text{if coriolis is used}) \end{cases} \quad (4)$$

其中 dx, dy 是网格间距， g 是重力加速度， H 是静水深度。

下面是把代码贴给 GPT，反向翻译出来的公式，与上面是吻合的，所以代码没问题

$$\begin{cases} \frac{\partial u}{\partial t} = -g \frac{\partial \eta}{\partial x} \\ \frac{\partial v}{\partial t} = -g \frac{\partial \eta}{\partial y} \\ \frac{\partial \eta}{\partial t} = -\frac{\partial}{\partial x} ((\eta + H)u) - \frac{\partial}{\partial y} ((\eta + H)v) \end{cases} \quad (5)$$

2 1D 浅水方程简述

$$\begin{cases} \frac{\partial}{\partial t}h + \frac{\partial}{\partial x}hv = 0 \\ \frac{\partial}{\partial t}hv + \frac{\partial}{\partial x}(hv^2 + \frac{gh^2}{2}) = 0 \end{cases} \quad (6)$$

或

$$\begin{cases} \frac{\partial h}{\partial t} = -h\frac{\partial v}{\partial x} - v\frac{\partial h}{\partial x} \\ h\frac{\partial v}{\partial t} + v\frac{\partial h}{\partial t} = -2hv\frac{\partial v}{\partial x} - (v^2 + gh)\frac{\partial h}{\partial x} \end{cases} \quad (7)$$

第一个方程式质量守恒方程，第二个方程是动量守恒方程；对应的积分形式如下：

$$\begin{cases} \frac{d}{dt} \int_{x_0}^{x_1} h dx = h(x_0, t)v(x_0, t) - h(x_1, t)v(x_1, t) = \int_{x_0}^{x_1} -\frac{\partial}{\partial x}(hv) dx \\ \frac{d}{dt} \int_{x_0}^{x_1} hvd x = h(x_0, t)v^2(x_0, t) + \frac{gh^2(x_0, t)}{2} - h(x_1, t)v^2(x_1, t) - \frac{gh^2(x_1, t)}{2} \end{cases} \quad (8)$$

对于浅水方程，对于每个区间 $[x_i, x_{i+1}]$ ，需要存储质量的平均值和动量的平均值：

$$\begin{cases} H_i = \frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} h dx \\ M_i = \frac{1}{\Delta x} \int_{x_i}^{x_{i+1}} hvd x \end{cases} \quad (9)$$

把 H_i, M_i 带入积分形式：

$$\begin{cases} \frac{d}{dt} H_i \Delta x = h(x_i, t)v(x_i, t) - h(x_{i+1}, t)v(x_{i+1}, t) \\ \frac{d}{dt} M_i \Delta x = h(x_i, t)v^2(x_i, t) - h(x_{i+1}, t)v^2(x_{i+1}, t) + \frac{gh^2(x_i, t)}{2} - \frac{gh^2(x_{i+1}, t)}{2} \end{cases} \quad (10)$$

对于任意的 $x \in (x_i, x_{i+1})$ ，我们可以把 $h(x, t)$ 和 $v(x, t)$ 看成常值函数，因为：

$$\begin{cases} h(x, t) \approx H_i(t) \\ h(x, t)v(x, t) \approx M_i(t) \\ v(x, t) \approx \frac{M_i(t)}{H_i(t)} \end{cases} \quad (11)$$

在边界点 x_i 和 x_{i+1} 上，我们可以通过对边界点左右两个区间的守恒量取平均值，对于 FVM 以及其他同类数值解函数存在不连续性的数值方法，这样给定两个相邻的控制体积上的函数解，求两个控制体积交界处的函数值的策略被称为数值通量（numerical flux）。这样简单取两个值的平均的通量被称为中央通量（central flux），不过它只是众多通量选择中的一种。在方程比较简单并且解函数比较平滑的时候，中央通量还是比较可靠的。但是对于 SWE，中央通量是数值不稳定（numerically unstable）的。为了解决这一问题，这里我们采用 Lax-Friedrichs 通量。Lax-Friedrichs 通量很容易计算：分别取左边和右边两区间上的值计算通量，取二值的平均，再加上一修正参数，即系统中的波速乘以该守恒量在两个区间上的差再除以二。这一修正参数被称为数值消散（numerical dissipation）。

$$h(x_i, t)v(x_i, t) \approx \frac{M_{i-1} + M_i}{2} + wavespeed \cdot \frac{H_{i-1} - H_i}{2} \quad (12)$$

对于边界点 x_0 和 x_n ，当我们使用反射强边界条件时，我们可以人为地设置两个假想体积（ghost cell）。比如，我们可以认为在区间 $[x_0, x_1]$ 的左边还有一个区间： $[x_{-1}, x_0]$ 。并且 $H_{-1} =$

$H_0, M_{-1} = -M_0$ 。也就是说：我们认为在 $[x_{-1}, x_0]$ 上的水和 $[x_0, x_1]$ 上的水有相同的高度但是相反的运动方向。形象一点来说的话，那就是我们可以认为 x_0 上有一面镜子，只不过这个“镜子”反射的不是光，而是动量。

所以，我们得到最终的 FVM 格式：

$$\begin{cases} \frac{d}{dt} H_i(t) = \frac{1}{\Delta x} [h(x_i, t)v(x_i, t) - h(x_{i+1}, t)v(x_{i+1}, t)] \\ \frac{d}{dt} M_i(t) = \frac{1}{\Delta x} \{ [h(x_i, t)v^2(x_i, t) + \frac{gh^2(x_i, t)}{2}] - [h(x_{i+1}, t)v^2(x_{i+1}, t) + \frac{gh^2(x_{i+1}, t)}{2}] \} \end{cases} \quad (13)$$

其中

$$\begin{cases} h(x_i, t)v(x_i, t) = \frac{M_{i-1} + M_i}{2} + c \cdot \frac{H_{i-1} - H_i}{2} \\ h(x_i, t)v^2(x_i, t) + \frac{gh^2(x_i, t)}{2} = \frac{\frac{M_{i-1}^2}{H_{i-1}} + \frac{M_i^2}{H_i}}{2} + \frac{gH_{i-1}^2 + gH_i^2}{4} + c \cdot \frac{M_{i-1} - M_i}{2} \end{cases} \quad (14)$$

其中 c 为波速， $H_{-1} = H_0, H_{n+1} = H_n, M_{-1} = -M_0, M_{n+1} = -M_n$

CFL 条件： $c \cdot \Delta t < \Delta x$

所以最终格式为：

$$\begin{cases} \frac{d}{dt} H_i(t) = \frac{1}{2\Delta x} [M_{i-1} - M_{i+1} + c \cdot (H_{i-1} - 2H_i + H_{i+1})] \\ \frac{d}{dt} M_i(t) = \frac{1}{2\Delta x} [(\frac{M_{i-1}^2}{H_{i-1}} - \frac{M_{i+1}^2}{H_{i+1}}) + \frac{g}{2}(H_{i-1}^2 - H_{i+1}^2) + c(M_{i-1} - 2M_i + M_{i+1})] \end{cases} \quad (15)$$

2.1 暑期 7 月工作

2.1.1 样本数据生成

为了提升训练样本的代表性，对训练样本的波进行复杂化。训练数据由数值方法生成，所以首先确定数值方法的精度，对比了两种网格划分（单位 1 区间划分的小区数）：红色实线的是划分 200 个小区间，蓝色虚线的是划分 400 个小区间，由下图可见，二者的相对误差 ≤ 0.015 ，所以在后续的样本生成过程中，就采用 400 去划分。

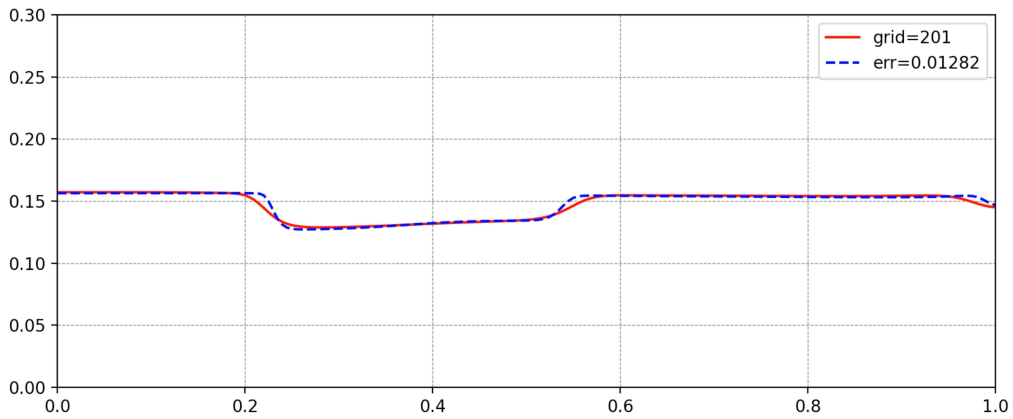


图 1: 不同网格划分的精度对比

训练样本数据的生成逻辑如下：

1. 通过高斯径向基函数，随机生成 5000 个振荡较为激烈的波函数。
2. 以每个函数为初值，利用数值方法，解出时间为 $[0, 6]$ 的数值解，经过这个步骤会生成 5000 个样本序列，每个序列的数据结构为 601×202 ，601 为时间 $[0, 8]$ 离散点个数，101 为空间 $[0, 1]$ 离散点个数，因为每个点对应两个数据：高度和速度，所以是 202。（这里为什么选择时间区间 $[0, 6]$ ，而不是 $[0, 10]$ ，是由于波在演变到时间 $t=6$ 之后就变得很平缓，几乎没有什么太大的振荡了。如下图所示）

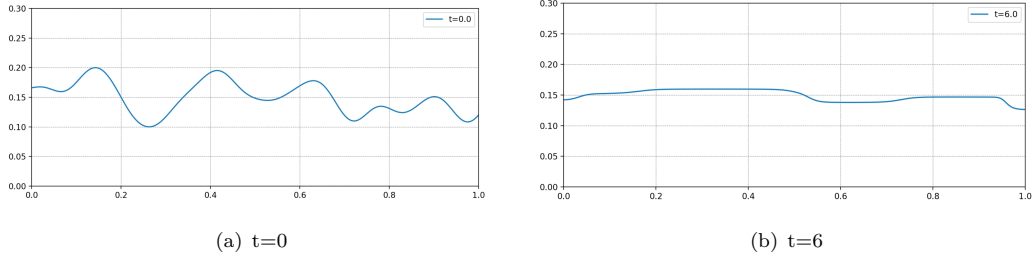


图 2: 浅水波随时间演变趋势

3. 对每个序列在时间 $[0, 5]$ 上，随机选取 n 个作为初值，并且在初值后面选取 m 个观测。观测点不妨就设为 $x=0.5$ ，观测时间间隔为 0.1。 n 和 m 的取值视情况而定。把初值和观测值合并到一起，作为 DeepONet 的分支网输入。

注：以上生成样本数据是很耗时的，所以要采用多进程技术，并且放到工作站上去执行。

2.1.2 神经网络训练

首先验证可行性，用 pinn 训练，看看复杂的波形神经网络是否可以训练。训练区间 $x \in [0, 1], t \in [0, 1]$ ，初值不妨选择第一个样本序列的 $t=0$ 的值，观测点不妨就设为 $x=0.5$ ，观测时间间隔为 0.1，取后续的 10 个观测值。结合半边值条件 $v=0$ 。物理信息点选择 10000 个。训练到神经网络的均方误差 $\leq 5 \times 10^{-6}$ ，根据经验，应该要训练到 $\leq 10^{-6}$ 才是好的，可能由于波形复杂，总是陷入局部最优解，所以经过几次参数调整，未达到 $\leq 10^{-6}$ ，但整体结果还是可以的，训练结果如图 3：

接下来用 DeepONet 训练，训练逻辑如下：

1. 对 5000 个样本序列采样，每个样本序列随机选取一个时间点的值作为初值；
2. 针对上述的每个初值，对后续的时间做观测采样，观测点去 $x=0.5$ ，后续每隔时间 0.1 才一个观测值，共采 10 个。和初值合并，作为分支网的输入；
3. 边值条件为 $v=0$ ；
4. 物理信息点选取 5000 个，使用超立方采样；

训练的均方误差： $mse \leq 5 \times 10^{-5}$ ，结果如图 4：

可以看出，结果不是很理想，只能看出大致的趋势是吻合的，这从一定程度上说明了可行性。至于为什么拟合的效果不好，总结原因如下：

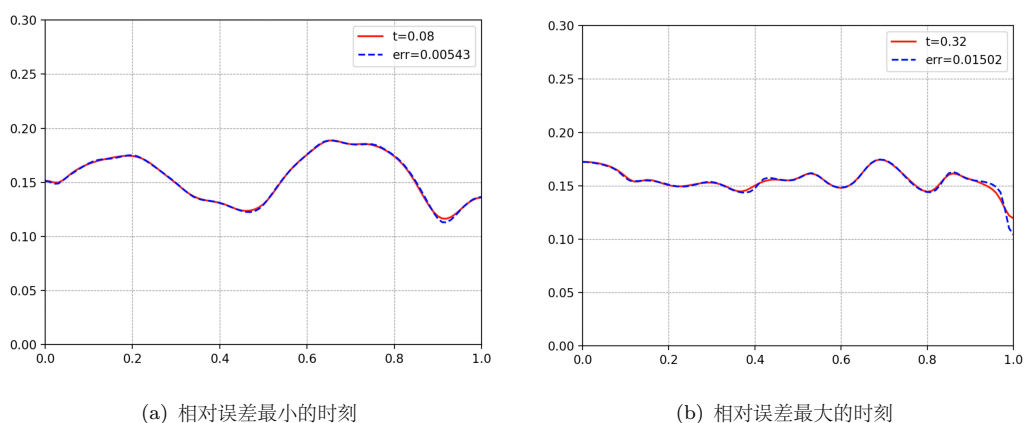


图 3: pinn 训练结果

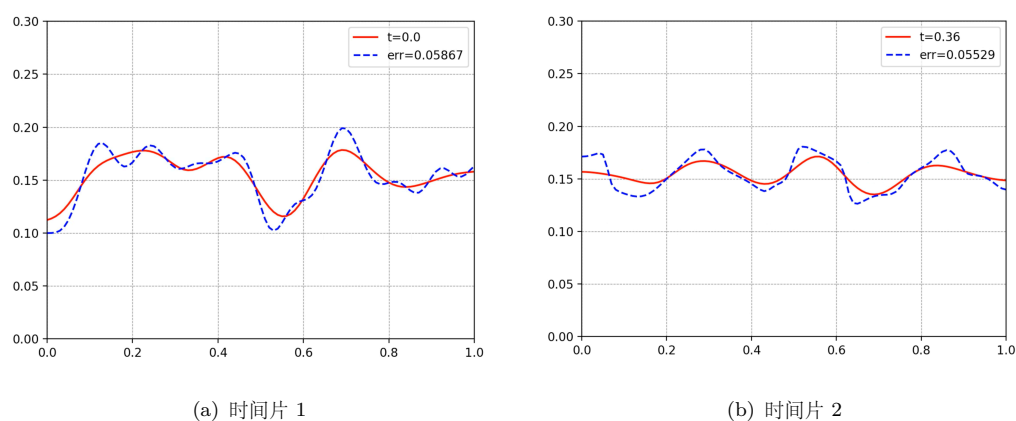


图 4: DeepONet 训练结果

1. 训练样本不具代表性，样本序列是 5000 个，每个样本序列对应约 5000 个离散时间点的数据，所以大概 2500w，但是训练数据只用了其中的 5000 个。
2. 物理信息点的选取偏少，根据经验对于区间 $[0, 1] \times [0, 1]$ ，物理信息点的个数最好为 10000 个，但是由于显存的限制，选取了 5000 个，在此情况下，显存使用量是 5982MiB。
3. 神经网络的训练可能容易陷入局部最优解，均方误差总是达不到理想值 10^{-6} 。
4. 训练的波形变得复杂了，同时分支网络的输入维度也变大了。之前的训练分支网只输入高度特征，现在还需要输入速度特征。同时加上些许观测。训练的复杂度提升了。

针对上述问题，后续我将做如下工作：

1. 适当缩小样本空间，由于浅水波的演变性质，随着时间的推移，水波趋于平缓，所以可以只关注前面时间的波，比如只看时间 $[0, 3]$ 之内的。样本序列也缩小下范围，比如先看 100 个时间序列，能否拟合的好。（这个也正在不停的调参数，训练过程中）
2. 训练数据懒加载，由于真正的训练数据是由分支网的输入数据和主干网的输入数据做卡氏积而得的，所以数据量会很大，消耗很多的显存。可以改变思路，只把分支网的原始

训练数据和主干网的原始训练数据放入显存，在实际训练过程中，实时计算每个 batch 的训练数据。（这样会大大加大训练时长，之前做个实验，训练时间大约变为原来的 4 倍，如果这样还能接受，不确定随着数据量的增大，训练时长是线性增加还是指数增加）

3. 尝试采用随机特征方法（RFM）去规避伸进网络陷入局部以及达不到期望收敛阶的问题。（这篇文章粗读了一遍，暂时没太明白原理，不过看样子是有希望的，需要后续继续精读，对比代码细致的看）。

3 RFM 研究

3.1 随机特征函数

什么是随机特征函数（random feature function, RFF）？随机特征函数就是特征向量随机生成的函数。对于机器学习方法来说，就是网络权值随机初始化的步骤。因此从机器学习的框架下看，RFM 就是利用一个定义在 Ω 上的网络基函数 $\{\phi_m\}$ 的线性组合来表示数值解

$$u_M(\mathbf{x}) = \sum_{m=1}^M u_m \phi_m(\mathbf{x})$$

$$\phi_m(\mathbf{x}) = \sigma(\mathbf{k}_m \cdot \mathbf{x} + b_m)$$

其中 \mathbf{k}_m, b_m 就是随机生成后固定的内层参数，而 σ 是非线性激活函数，为方程求解提供非线性的部分。

$$Lu = f, u = \sum_{m=1}^M u_m \phi_m(\mathbf{x})$$

$$Lu = \sum_{m=1}^M u_m L\phi_m(\mathbf{x}) = f$$

$$\sum_{m=1}^M u_m L\phi_m(\mathbf{x}_j) = f_j$$

$$\mathbf{A}\mathbf{u} = \mathbf{f}$$

问题就转换为了最小二问题，其中 \mathbf{u} 就是最后待求解的外层权重参数，这个算法有潜在的问题，就是如果没有源项，即 $\mathbf{f}=0$ ，则求出来的 $\mathbf{u}=0$ 。

3.2 局部 RFF 与单位分解

上述随机特征函数是全局定义的，但微分方程的解常常有小尺度的局部变化，因此 RFM 考虑在多个局部构造局部随机特征函数，再用单位分解（partition of unity）技术将它们组合。具体来说，首先取定单位分解函数的中心点 $\{\mathbf{x}_n\}_{n=1}^{M_p} \subset \Omega$ ，在这 M_p 个局部构造仿射变换
注：这里把定义域划分为 M_p 个小区间， $\{\mathbf{x}_n\}$ 是每个小区间的中点

$$\tilde{\mathbf{x}} = \frac{1}{r_n}(\mathbf{x} - \mathbf{x}_n), \quad n = 1, \dots, M_p, \quad (16)$$

这个仿射变换将 $[x_{n1} - r_{n1}, x_{n1} + r_{n1}] \times \cdots \times [x_{nd} - r_{nd}, x_{nd} + r_{nd}]$ 的小局部映射到统一的区间 $[-1, 1]^d$ ，以便实现局部特征的拟合。而单位分解函数的构造依赖于这个仿射变换，常取为

$$\psi_n^a(x) = \mathbb{I}_{-1 \leq \tilde{x} < 1},$$

或

$$\psi_n^b(x) = \mathbb{I}_{[-\frac{5}{4}, -\frac{3}{4}]}(\tilde{x}) \frac{1 + \sin(2\pi\tilde{x})}{2} + \mathbb{I}_{[-\frac{3}{4}, \frac{3}{4}]}(\tilde{x}) + \mathbb{I}_{[\frac{3}{4}, \frac{5}{4}]}(\tilde{x}) \frac{1 - \sin(2\pi\tilde{x})}{2}$$

紧接着，在每个局部定义 J_n 个随机特征函数

$$\phi_{nj}(\mathbf{x}) = \sigma(\mathbf{k}_{nj} \cdot \tilde{\mathbf{x}} + b_{nj}), \quad j = 1, \cdots, J_n, \quad (17)$$

注：这里类似一层全连接隐藏层，只不过把输入标准化到 $[-1, 1]$ 之间， $\tilde{\mathbf{x}}$ 的维度是输入维度， J_n 是输出维度。

则最终数值解就是将局部随机特征函数通过单位分解函数组合起来

$$u_M(\mathbf{x}) = \sum_{n=1}^{M_p} \psi_n(\mathbf{x}) \sum_{j=1}^{J_n} u_{nj} \phi_{nj}(\mathbf{x})$$

注：右边的 Σ 对应的是一个全连接网络， J_n 表示这个网络的隐层维度（神经元个数），左边的 Σ 表示用单位分解技术将它们组合。 M_p 表示划分的区间个数，对应着神经网络的个数。

3.3 参考文献

1. 介绍 RFM 的：J. Chen, X. Chi, W. E, and Z. Yang, Bridging traditional and machine learning based algorithms for solving pdes: The random feature method, Journal of Machine Learning, 1 (2022), pp. 268–298.
2. 关于神经网络 MSE 缺陷的文章：C. Wang, S. Li, D. He, and L. Wang, Is L^2 Physics Informed Loss Always Suitable for Training Physics Informed Neural Network?. Advances in Neural Information Processing Systems, 35 (2022), pp. 8278–8290.