Complete flow characterization from snapshot PIV, fast probes and physics-informed neural networks

Álvaro Moreno Soto, Alejandro Güemes and Stefano Discetti
Prepared by Álvaro Moreno Soto

Introduction

This document includes a set of instructions for the correct implementation of the scripts used for the increase of temporal resolution of PIV experiments and their further regularization via physics-informed neural networks as detailed in the publication under the same name and authors as this document, DOI https://doi.org/10.1016/j.cma.2023.116652

The datasets

Datasets are classified in two separate folders at Zenodo, each of them referred to a different set of experiments documented in the main article. For validation of the proposed architecture, the dataset *Pinball* is included, whereas for the application on a real experiment we add the dataset *Airfoil2D*. The datasets consist of three files: the original data at low-time resolution (***_PIV.mat*), the fields estimated using the MLP (***_MLP.mat*) and the time-resolved fields enhanced with the PINN (***_PINN.mat*), where **** stands either for *Pinball* or *Airfoil2D*, accordingly. The two latter can be computed using the Python scripts **MLP.py** and **PINN.py** available at GitHub and following the procedure describe below.

MLP.py implements the enhancement of time resolution via the estimation of temporal modes according to the input of subsequent velocity measurements given by pointwise fast probes during one convective flow throughtime. The script runs with the data contained in file ***_PIV.mat* which includes the following components:

- &_DNS: reference field obtained by direct numerical simulation, used for reference purposes,
- &_PIV: PIV dataset
- &_probe: measurements by fast probes,

where & stands for time and Cartesian coordinates T, X, Y, and velocity and pressure components U, V, P. Bear in mind that test cases are further indicated by *_test*. For convenience, the matrix which contains the association probe-sequence to PIV snapshot is already computed and named *MLP_PIV*.

The output of this first script is a *mat* file containing the velocity field reconstructed on the PIV grid over the test dataset and now with full time resolution. This file corresponds to ***_MLP.mat*, containing time, Cartesian components on the original PIV grid and the corresponding velocity field.

PINN.py implements the physics-informed neural network. It receives as input the matrix ***_MLP.mat* (and when necessary, ***_PIV.mat* for reference purposes). PINN regularizes the velocity field by imposition of the Navier-Stokes equations while keeping as reference the MLP-given velocity field. The output of the algorithm is a full time-resolved domain which has been regularized by physics constraints and which is able to disclose further fluid features, such as the pressure gradient or the full pressure field if a pressure reference value is available. The final reconstruction is also better spatially-resolved, since the output grid in which the velocity and pressure fields are computed is significantly finer than the original PIV one. The output of the script is the corresponding file ***_PINN.mat* containing time, Cartesian components on a high-spatially-resolved grid and the corresponding time-resolved velocity fields.

For convenience, all files ***_PIV.mat, **_MLP.mat* and ***_PINN.mat* are included in the Zenodo repository, even though only ***_PIV.mat* is necessary to satisfactorily run both scripts in order 1) MLP, 2) PINN. Note that for a proper execution, the datasets need to be allocated in the same folder as the Python scripts.

Particularity of the *Pinball* dataset

Whereas the data contained in *Airfoil2D* folder has been obtained by real PIV experiments on a NACA 0018 profile, the PIV information for the pinball cases have been synthetically obtained by artificially seeding particles over the data provided by DNS and computing a moving average with a 32-px interrogation window with a 50% overlap. Additionally, Gaussian noise up to 20% has been added on top of the average velocity values. The different cases for validation and comparison are detailed below and included in independent subfolders:

- Case 1: MLP + PINN with PIV data obtained by direct interpolation of DNS over PIV grid (no interrogation window required for this case)
- Case 21: MLP + PINN with PIV data computed with a 32-px interrogation window
- Case 22: MLP + PINN with PIV data computed with a 64-px interrogation window
- Case 31: MLP + PINN with PIV data computed with a 32-px interrogation window and additional Gaussian noise up to 5%
- Case 32: MLP + PINN with PIV data computed with a 32-px interrogation window and additional Gaussian noise up to 10%
- Case 33: MLP + PINN with PIV data computed with a 32-px interrogation window and additional Gaussian noise up to 20%