Machine Learning in the Fourier Domain: Fourier Neural Operators

Michele Pascale

Department of Medical Physics and Bioengineering
University College London

April 3, 2023

Conventional vs. Data-Driven Methods

- Conventional methods such as FEM and FDM rely on discretisation of space, resulting in a resolution trade-off: coarse grids, fast but inaccurate and fine grids accurate but computationally slow.
- Data-driven methods (using NNs) attempt to learn trajectories from data for the family of equations. Classic NN approaches learn finite-dimensional mappings, hence unable to learn mesh-invariant solutions.

Neural Operators

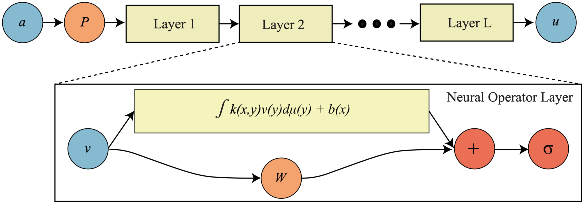
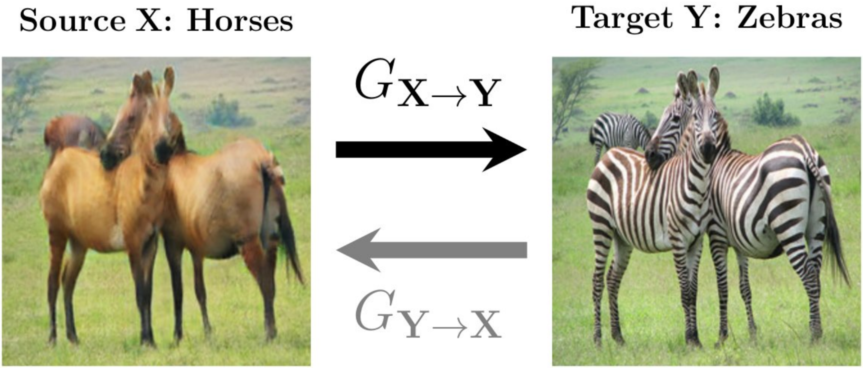


Figure: Neural Operator (base architecture)

- Some of the first work to propose learning mesh-free, infinite-dimensional operators using a Neural Network.
- Only needs training once, needs no knowledge of PDE and the single set of learned network parameters has ability to transfer solutions between meshes.
- Neural Operators alone cannot rival efficiency of finite-dimensional CNN approaches, due to cost of evaluating integral operators.

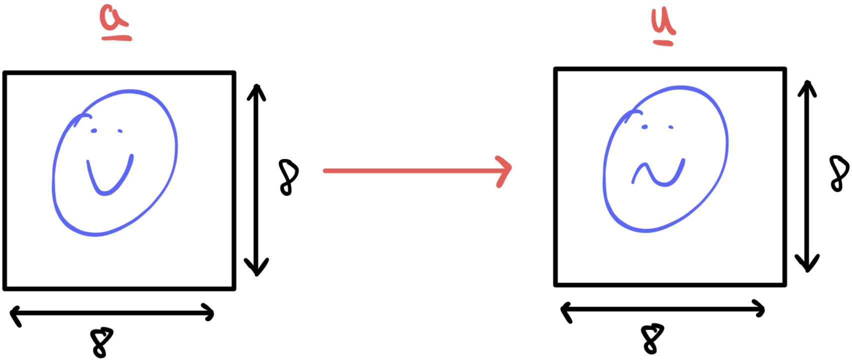
Fourier Neural Operators (Overview)

- FNO was the first work that learned resolution-invariant solution operators on Navier-Stokes equations.
- Builds on the notion of Operator Learning and can be intuitively described as a traditional Image-to-Image problem in ML:



Why not learn in the image domain?

In image space, you work at a **fixed resolution** with no flexibility after network training.



The input **a** and output from network **u** here can be considered as the input and solution state respectively for a given PDE.

Definition 1 (Input / Output Space)

Let $D \subset \mathbb{R}^d$ be a *bounded, open set* and $\mathcal{A} = \mathcal{A}(D; \mathbb{R}^{d_a})$ and $\mathcal{U} = \mathcal{U}(U; \mathbb{R}^{d_b})$ be *separable Banach Spaces* of function-taking values in $\mathbb{R}^{d_a}$ and $\mathbb{R}^{d_b}$ respectively.

Definition 2 (Solution Operator)

Furthermore, let $G^* : \mathcal{A} \to \mathcal{U}$ be a (typically) non-linear map between the two spaces. Here, $\mathcal{A}$ and $\mathcal{U}$ are the input and output spaces respectively and G^* is the Neural Operator, or "mapping function" between the input and output (solution) space.

Since this is a data-driven approach, suppose we have some observations $\{a_j, u_j\}$ $(j = 1 \ldots N)$, where $a_j \sim \mu$ is an i.i.d sequence from the probability measure μ supported on $\mathcal{A}$ and $u_j = G^*(a_j)$ is possibly corrupted with noise.

Definition 3 (Parametric Mapping Function)

We want to build an approximation of the mapping G^* via construction of a parametric map:

$$G : \mathcal{A} \times \Theta \to \mathcal{U} \text{ or equivalently, } G_\Theta : \mathcal{A} \to \mathcal{U}$$

for some finite-dimensional parameter space Θ by choosing $\theta^* \in \Theta$ so that $G(\cdot, \theta^*) = G_{\theta^*} \approx G^*$.

The goal here is to learn to minimise a cost functional $\mathcal{C} : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$:

$$\min_{\theta \in \Theta} \mathbb{E}_{a \sim \mu}[\mathcal{C}(G(a, \theta), G^*(a))]$$

FNO Theory - Kernel Integral Operator

Definition 4 (Kernel Integral Operator $\mathcal{K}$)

$$(\mathcal{K}(a;\phi)v_t)(x) := \int_D k(x, y, a(x), a(y); \phi)v_t(y)dy, \quad \forall x \in D$$

where $\mathcal{K}(a;\phi)$ is chosen to be a kernel integral transformation parameterised by a neural network.

Definition 5 (Iterative Updates)

We define the iterative update representation as follows:
$v_{t+1}(x) = \sigma(Wv_t(x) + (\mathcal{K}(a;\phi)v_t)(x))$

Now, dependence on $a(x)$ is not ideal in the context of NNs (can't propagate result through contiguous layers $v_1 \mapsto v_2 \mapsto \cdots \mapsto v_{t-1}$)...

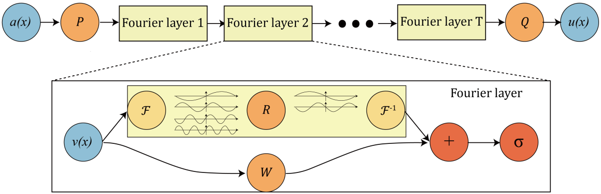
FNO Theory - Fourier Integral Operator

If we remove dependence on $a(x)$ and impose $\mathcal{K}_\phi(x, y) = \mathcal{K}_\phi(x - y)$ we can apply the convolution theorem to attain:

Definition 6 (Fourier Integral Operator)

$$(\mathcal{K}(a;\phi)v_t)(x) = \mathcal{F}^{-1}(R_\phi \cdot (\mathcal{F}(v_t)))(x), \quad \forall x \in D$$

Recall that a convolution in Fourier space is simply a multiplication operation and the kernel integral operator $\mathcal{K}$ has implicitly been deconstructed into a learnable parameter matrix R_ϕ (which is a Fourier transform of a periodic function) which is trained within the FNO network, multiplied by the Fourier transform of v_t (illustrated in the bottom row of the figure below).



Culling of top Fourier modes acts somewhat as a regularisation technique to help improve generalisation (in the context of Navier Stokes equations only?)