POD-NN

REFERENCE: Hesthaven, Ubbiali, "Non-intrusive reduced order modeling of nonlinear problems using neural networks", JCP, 2018.

GOAL: provide a FAST ONLINE PHASE even for problems with nonlinearities of non-affine structures.

How? : AVOID PROJECTION!!! Using NN!

PROBLEM FORMULATION

given $\mu \in \mathcal{P}$ find $u(\mu) \in V$ s.t.

$$\mathcal{G}(u(\mu);\mu) = 0 \qquad (1) \qquad \text{(strong)}$$

In weak formulation we have:

$$g(u(\mu), v; \mu) = 0 \qquad \forall v \in V \qquad (2)$$

FOM DISCRETIZATION

$V_\delta \subset V \rightarrow$ given $\mu \in \mathcal{P}$ find $u_\delta(\mu) \in V_\delta$ s.t.

$$g(u_\delta(\mu), v; \mu) = 0 \qquad \forall v \in V_\delta$$

ALGEBRAICALLY

$$\underbrace{G_\delta(\underline{u}_\delta(\mu);\mu)}_{\text{RESIDUAL VECTOR}} = 0 \in \mathbb{R}^{N_\delta} \qquad (3)$$

$\hookrightarrow G_\delta(\underline{u}_\delta(\mu);\mu)_i = g(u_\delta(\mu), \varphi_i; \mu)$

where $\{\varphi_i\}_{i=1}^{N_\delta}$ is a FOM BASIS

ROM SPACE

1. BUILD THE SPACE
2. PROJECT (3) AND SOLVE

How can I avoid projection?

$$\mu \xmapsto{NN} \underline{u}_N \in \mathbb{R}^N \quad (N \ll N_\delta)$$

where N is ROM DIMENSION
N_δ is FOM DIMENSION

ROM PROJECTION STRATEGY

ROM space $V_N = \text{span}\underbrace{\{\psi_1, \dots, \psi_N\}}_{\text{related to the snapshots}} \underbrace{\subset}_{\text{linear}} V_\delta$

The reduced solution $u_N(\mu) = \sum_{\ell=1}^{N} u_{rb}^{\ell}(\mu)\psi_\ell$

$\Rightarrow \quad \underline{u}_N(\mu) = [u_{rb}^1, \dots, u_{rb}^N]^T \in \mathbb{R}^N$

To find $u_N(\mu)$ we usually project.
$\mathbb{B} = [\psi_1, \dots, \psi_N] \in \mathbb{R}^{N_\delta \times N}$

$$\underbrace{0 =}_{\text{COMPONENT-WISE}} G_N(u_N(\mu); \mu)_i = g\left(\sum_{\ell=1}^{N} u_{rb}^{\ell}(\mu)\psi_\ell, \psi_i; \mu\right) \quad \begin{array}{l} \ell = 1, \dots, N \\ i = 1, \dots, N \end{array}$$

$$\underbrace{\Rightarrow}_{\text{MATRIX NOTATION}} G_N(u_N(\mu); \mu) = \mathbb{B}^T G_\delta(\underbrace{\mathbb{B} u_N(\mu)); \mu}_{\text{each time}})$$

THIS is BAD!!! || I have: to compute $\mathbb{B}u_N(\mu)$
assemble G_δ
project G_δ

STANDARD SOLUTIONS: EIM
DEIM
TENSORS ...

NEW SOLUTION? POD-NN

Before POD-NN let us talk about ROM proj.

In the FOM-ROM case $V_N \subset V_\delta$

$V_N = \text{span}\{\psi_1, \dots, \psi_N\}$, thus

V_N is generated by the columns of $\mathbb{B}$

In general, an orthogonal projector is an operator s.t.

$$\text{Ker}(\mathbb{P}) = \text{Range}(\mathbb{P})^\perp$$

Namely $\underbrace{\mathbb{P}x \in M}_{\text{a subspace of a bigger space}} \Rightarrow (\mathbb{I} - \mathbb{P})x \in M^\perp$

$\mathbb{P}$ is an orthogonal projector if
$\mathbb{P}^2 = \mathbb{P}$ and $\mathbb{P} = \mathbb{P}^T$.

In our case $M = V_N$

$\mathbb{P}_2 = \underbrace{\mathbb{B}\mathbb{B}^T}$ is an orthogonal projector w.r.t. L^2-inner product

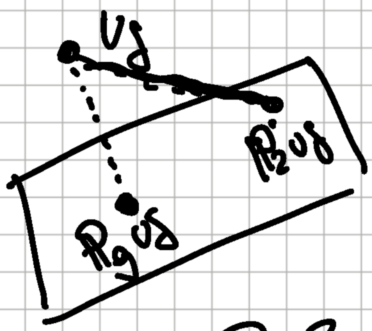
with $\mathbb{B}$ - L^2-orthogonal $(\psi_i, \psi_e)_{L^2} = 0$ $\quad [\, y^T x \sim (y, x)_{L^2} \,]$

If we do not have the L^2-orthogonality between the basis we can use H^1-semi-norm-orthogonality

Our basis are built $(\nabla\psi_i, \nabla\psi_e)_{L^2} = 0$

we need to define a new-projector

$\mathbb{P}_2 = \mathbb{V}\mathbb{V}^T$ whe $\mathbb{V} = \underbrace{\mathbb{X}_\delta^{1/2}}_{\text{matrix of the inner product}} \mathbb{B}$



V_N is built w.r.t. H^1 semi-norm
or better, the energy norm!

$\mathbb{P}_\delta$? $\quad \mathbb{P}_\delta x = \underset{y \in \mathrm{Range}(\mathbb{P}_\delta)}{\arg\min} \|x - y\|_X^2$

$$\mathbb{P}_\delta v_\delta(\mu) = \mathbb{B}(\mathbb{B}^T X_\delta \mathbb{B})^{-1} \mathbb{B}^T X_\delta v_\delta(\mu)$$

$$\parallel$$

$$\mathbb{B}\, u_N(\mu)$$

↳ we have a representation of $u_N(\mu)$
WITHOUT SOLVING THE SYSTEM
IF WE HAVE THE SNAPSHOT!!!

Let us relate this concept with NN

$$\pi^{NN}(\mu) = \underline{u}_{rb}^{NN} \in \mathbb{R}^N$$

To train this problem I need DATA!

If I have a POD I can use:

- the parameters of the POD ($\mu \in \mathcal{P}_{train}$)
- the related snapshots
- $\mathbb{B}$
- X_δ

and I can define the following loss

$$\sum_{\mu \in \mathcal{P}_{train}} \|\pi^{NN}(\mu) - \underline{u}_N\|_X^2$$

$$\underline{u}_N(\bar{\mu}) = \underbrace{(\mathbb{B}^T \mathbb{X}_\delta \mathbb{B})^{-1}}_{\mathbb{X}_N^{-1}} \mathbb{B}^T \mathbb{X}_\delta u_\delta(\bar{\mu})$$

In order to find $\underline{u}_N(\bar{\mu})$ $(\bar{\mu} \in \mathcal{P}_{train})$
I solve the following
system

$$\boxed{\mathbb{X}_N \underline{u}_N(\mu) = \mathbb{B}^T \mathbb{X}_\delta u_\delta(\mu)}$$

You still need a <u>COSTLY OFFLINE PHASE</u>

BUT online you have $\mu^* \xmapsto[\pi^{NN}(\mu^*)]{} \underline{u}_N(\mu^*)$