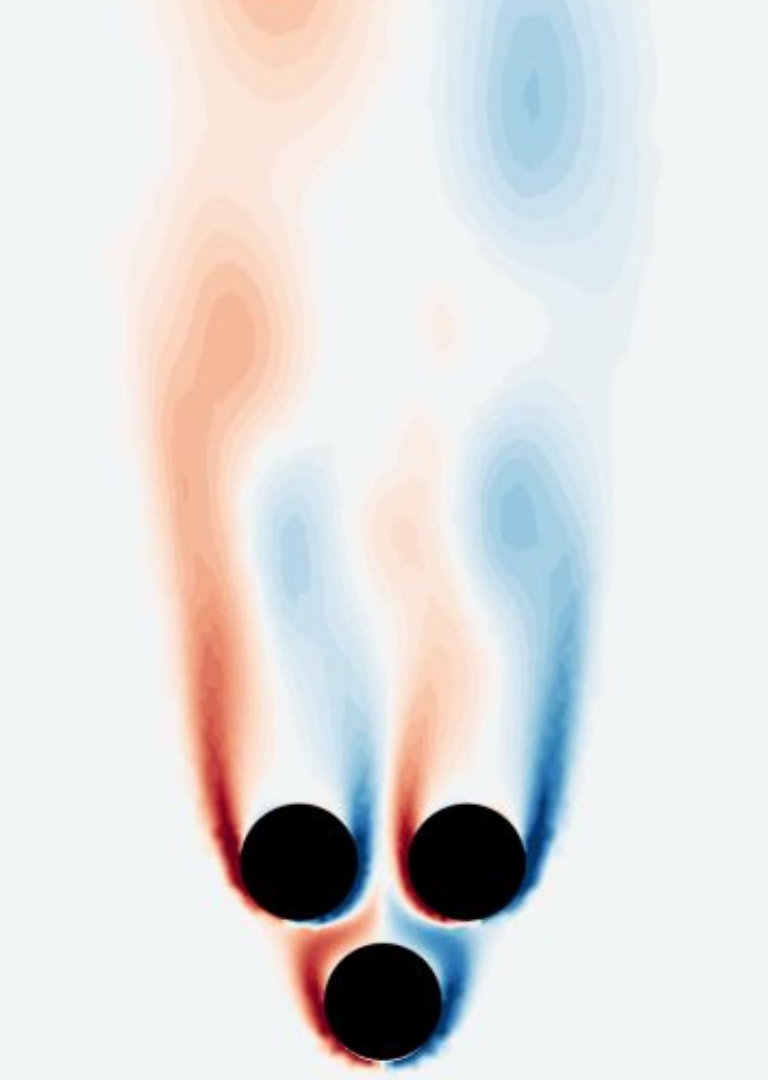


**CD-ROM:
Complemented
Deep-Reduced
Order Model**

22 November 2023





SystemX program: Simulation/machine learning hybrid modelling

- Emmanuel MENIER (SystemX-INRIA)
- Michele Alessandro BUCCI (SafranTech-INRIA)
- Mouadh YAGOUBI (SystemX)
- Lionel MATHELIN (CNRS)
- Marc SCHOENAUER (INRIA)

Menier, Emmanuel, et al. "CD-ROM: Complemented Deep-Reduced order model." *Computer Methods in Applied Mechanics and Engineering* 410 (2023): 115985.

Agenda

01

Introduction and motivations

02

ROM closure

03

Deep Learning solution

04

Results

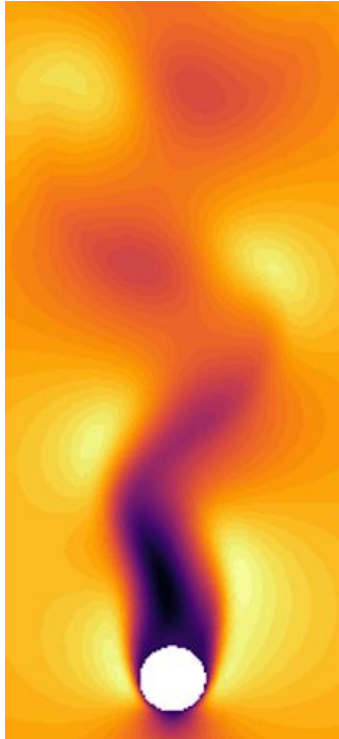
05

Conclusions & Perspectives

Introduction and motivation

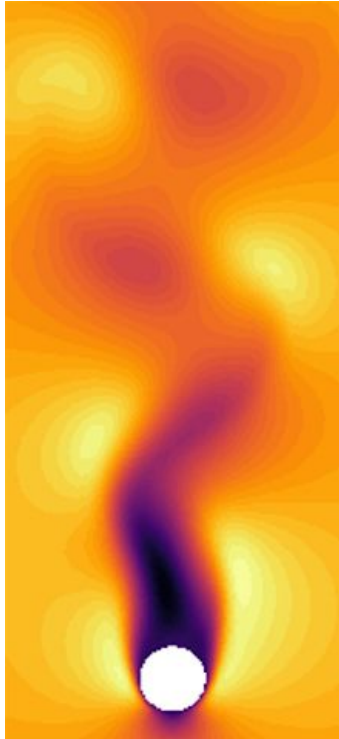
POD–Galerkin ROMs

First principle simulations



- **Numerical simulations are computationally demanding:**
 - Physical reasons: small scales structures and fast events require tiny spatial and temporal discretisation
 - Numerical reasons: stability of numerical schemes, dispersion or dissipation errors
 - Complex geometries
- **Generally the computational cost does not scale with the intrinsic dimension of the problem**
 - e.g a limit cycle might be described with a few Fourier modes

First principle simulations



- **Numerical simulations are computationally demanding:**
 - Physical reasons: small scales structures and fast events require tiny spatial and temporal discretisation
 - Numerical reasons: stability of numerical schemes, dispersion or dissipation errors
 - Complex geometries
- **Generally the computational cost does not scale with the intrinsic dimension of the problem**
 - e.g a limit cycle might be described with a few Fourier modes



Reduced Order Model

Reduced Order Model

Find a suitable subspace where the dynamics holds

- Given $\mathbf{u} \in \mathbb{R}^n$, $\mathbf{U} \in \mathbb{R}^{n \times N}$ and $\mathbf{a} \in \mathbb{R}^N$

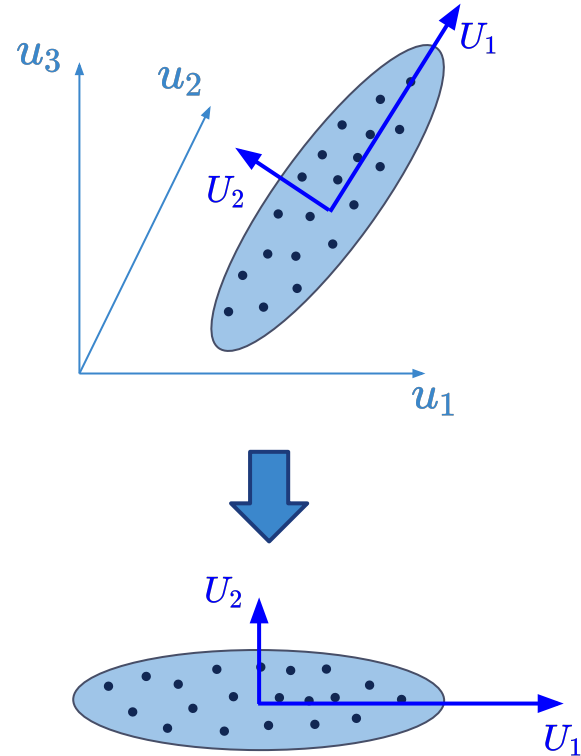
Galerkin projection

$$\mathbf{u}(x, t) = \sum_i^N U_i(x) a_i(t) = \mathbf{U}(x) \mathbf{a}(t)$$

- If $\langle U_i, U_j \rangle = \delta_{ij}$ then

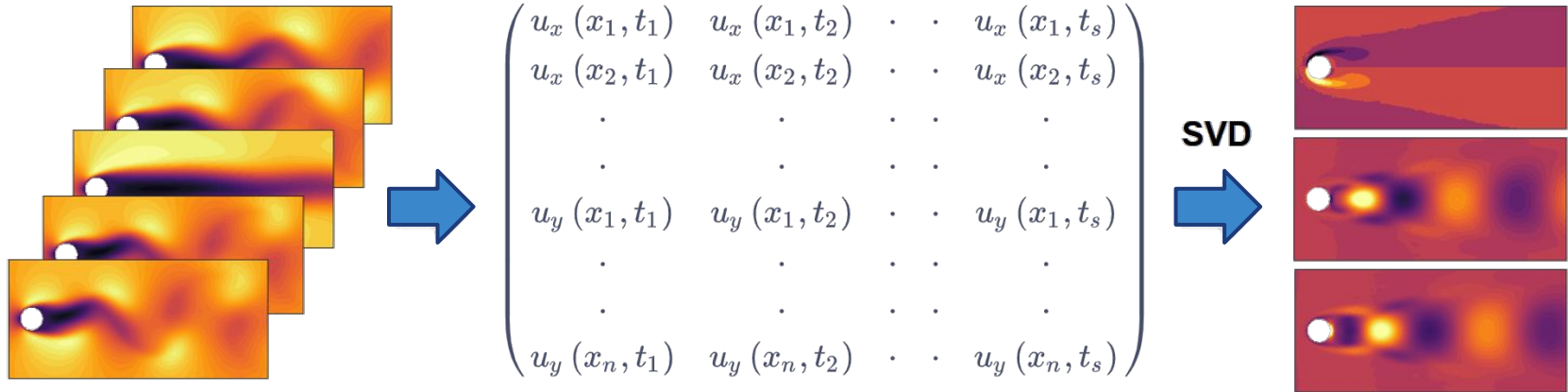
$$\mathbf{P} = \mathbf{U}\mathbf{U}^T$$

Is a projector operator that can be used to recover the minimal subspace where the dynamics holds.



POD

Proper Orthogonal Decomposition can be used to extract principal modes from a physical simulation

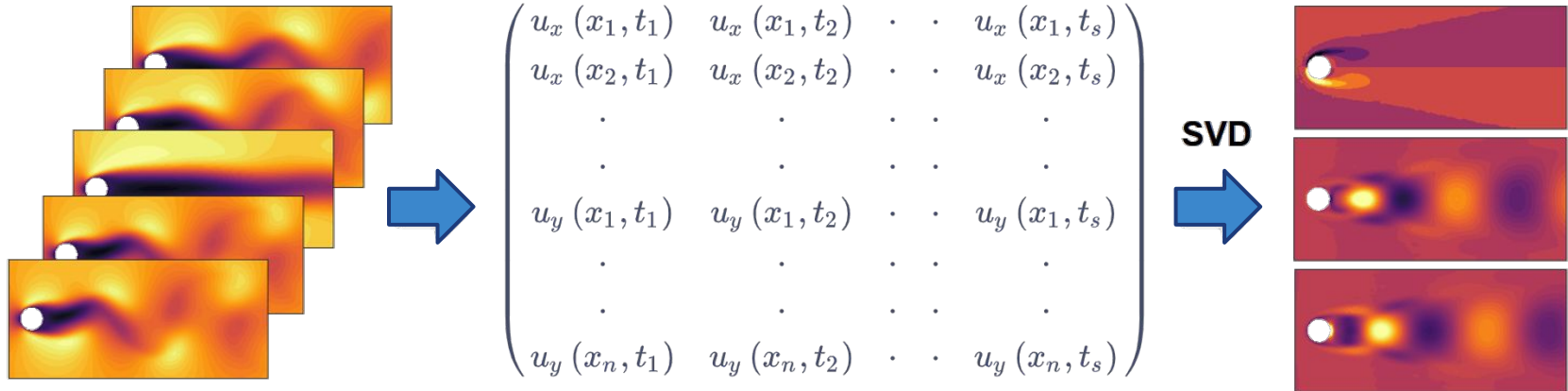


The POD base \mathbf{U} is truncated to the first k modes according to some energy norm, such that:

$$\|\mathbf{P}\mathbf{u}\|_F \gg \|(I - \mathbf{P})\mathbf{u}\|_F$$

POD

Proper Orthogonal Decomposition can be used to extract principal modes from a physical simulation



The POD base \mathbf{U} is truncated to the first k modes according to some energy norm, such that:

$$\|\mathbf{P}\mathbf{u}\|_F \gg \|\cancel{(I - \mathbf{P})\mathbf{u}}\|_F \text{ neglected}$$

POD-Galerkin ROM

If the dynamics of the FOM is described by:

$$\frac{d\mathbf{u}}{dt} = g(\mathbf{u})$$

The associated POD-Galerkin ROM with N ODEs reads:

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + \text{residuals}$$

Residuals come from POD base truncation.

- Might affect **transient dynamics**. Dangerous if the ROM is used in a control scenario (e.g. MPC)
- If not small enough, errors can cumulate and might lead dynamics to **diverge!**

POD-Galerkin ROM

If the dynamics of the FOM is described by:

$$\frac{d\mathbf{u}}{dt} = g(\mathbf{u})$$

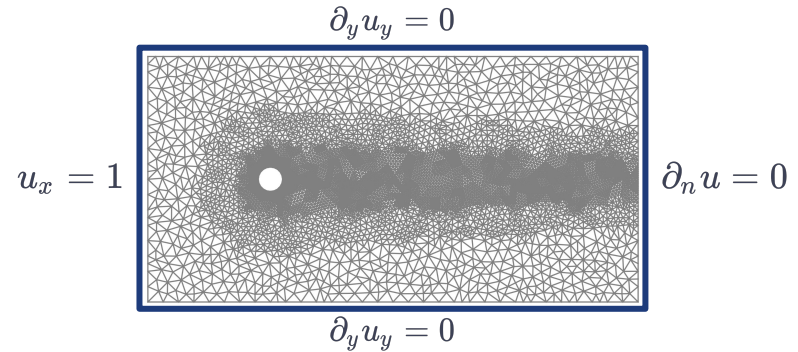
The associated POD-Galerkin ROM with N ODEs reads:

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + \text{residuals}$$

Residuals come from POD base truncation.

- Might affect **transient dynamics**. Dangerous if the ROM is used in a control scenario (e.g. MPC)
- If not small enough, errors can cumulate and might lead dynamics to **diverge!**

Incompressible flow around 2D cylinder case*

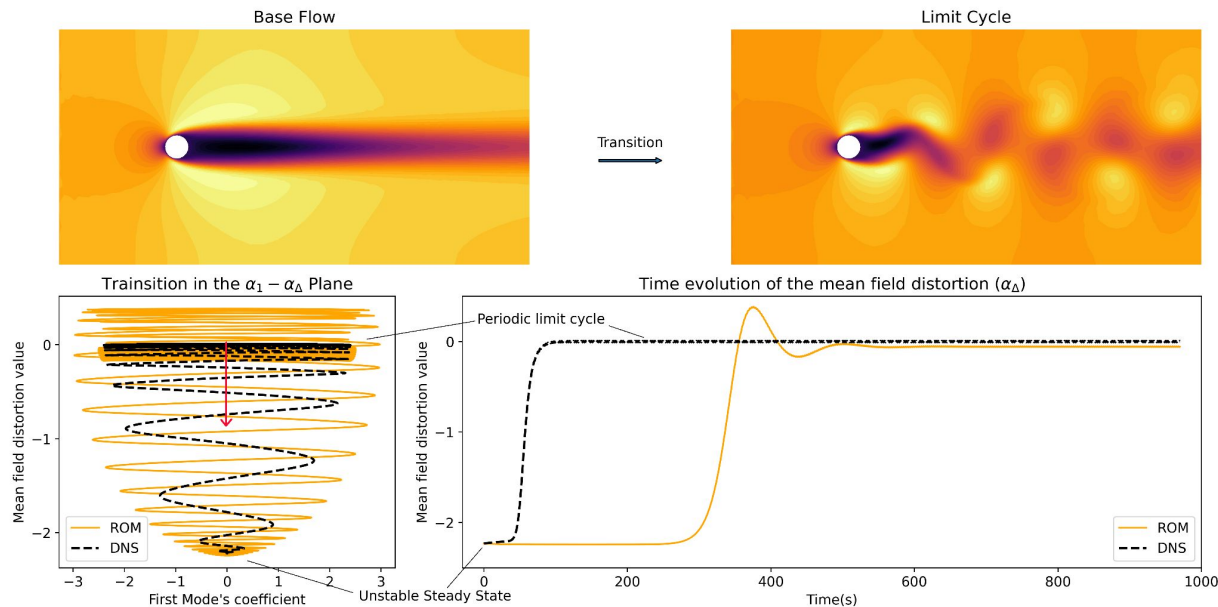


ROM describing the dynamics from the unstable base flow ($\dot{\mathbf{u}} = \mathbf{0}$) at $t = 0$ to the stable periodic cycle at $Re = 100$.

*BR Noack et al. "A hierarchy of low-dimensional models for the cylinder wake." JFM (2003)

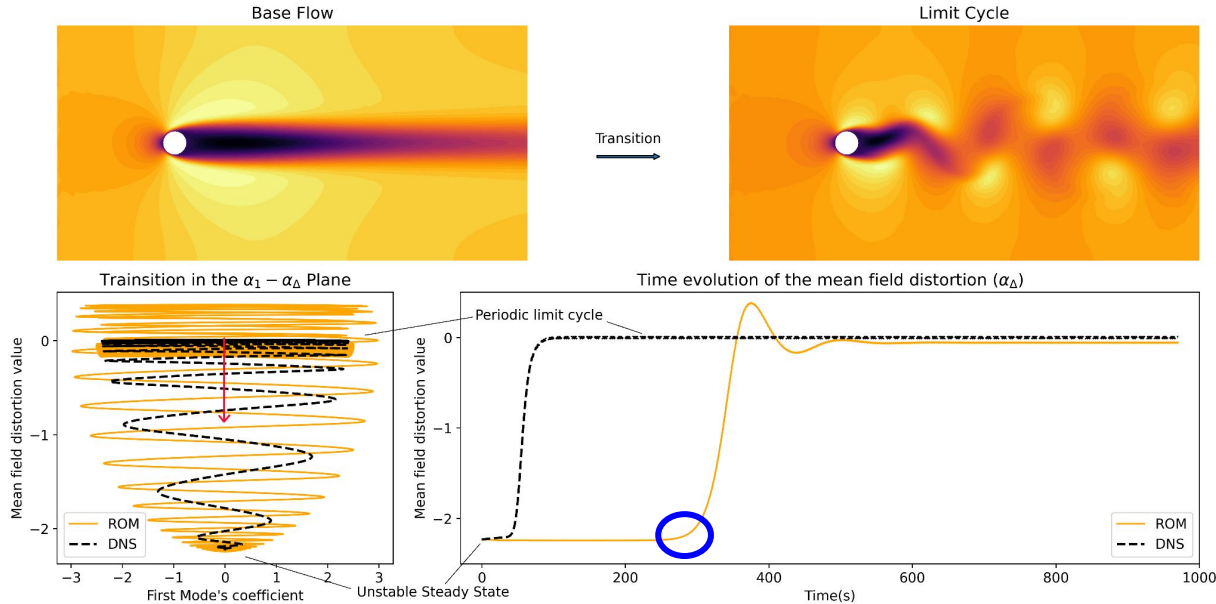
ROM performance: Flow around 2D Cylinder

2 POD modes + the shift mode were used to recover the ROM



ROM performance: Flow around 2D Cylinder

2 POD modes + the shift mode were used to recover the ROM

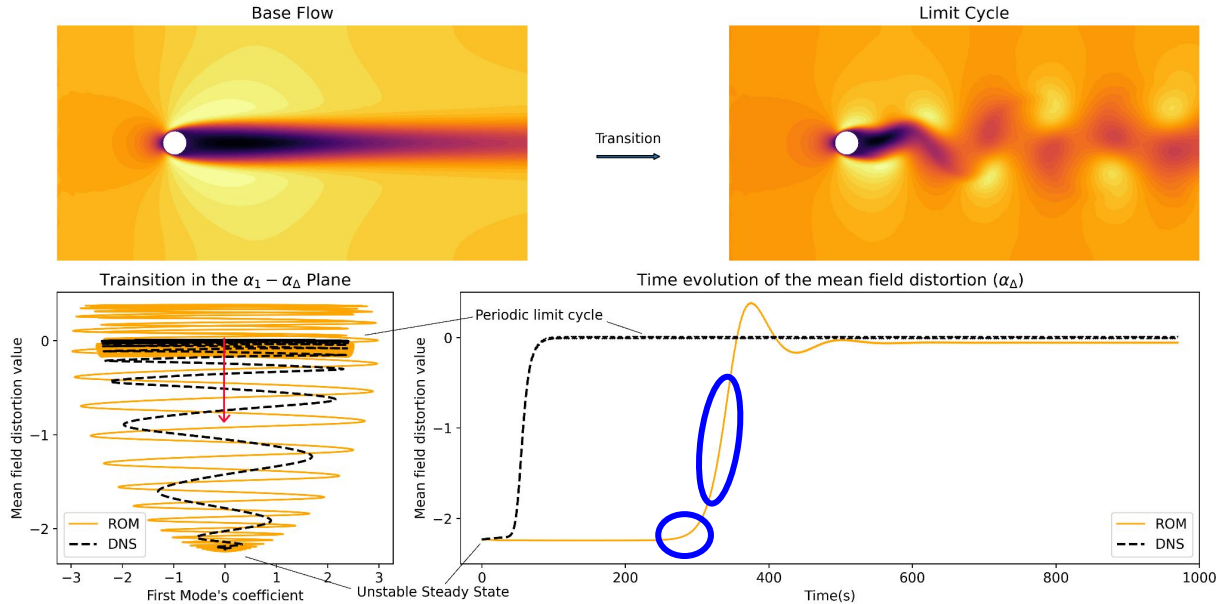


Errors:

- Delayed onset of the instability

ROM performance: Flow around 2D Cylinder

2 POD modes + the shift mode were used to recover the ROM



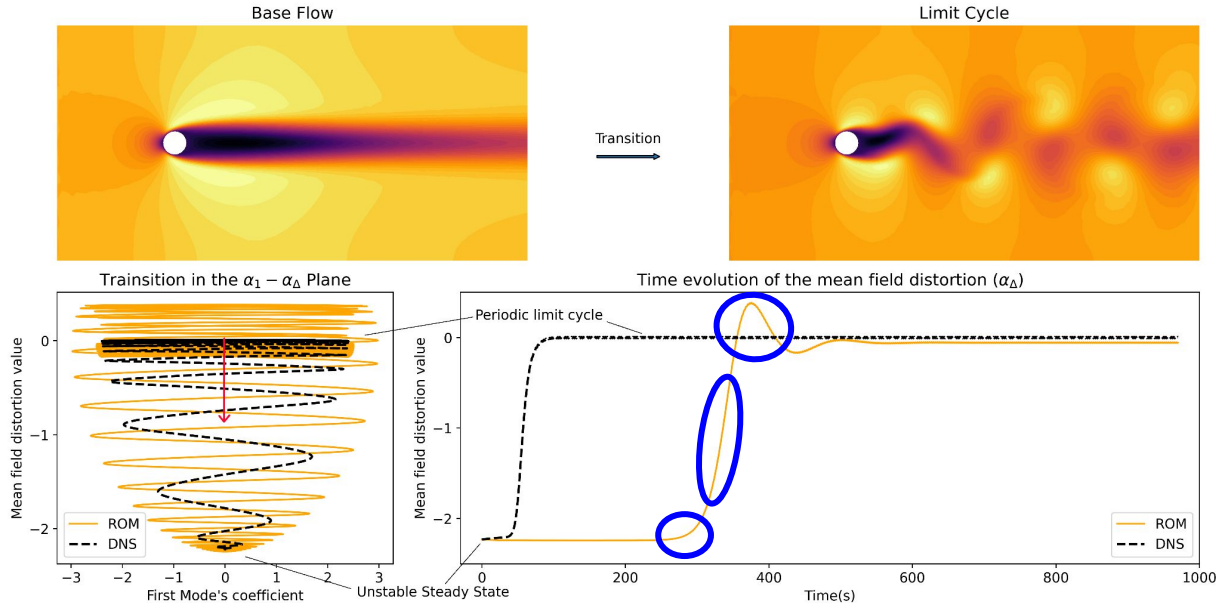
Errors:

- Delayed onset of the instability
- Wrong growth rate (might be recovered with a recalibration*)

*Kaptanoglu, Alan A., et al.
 "Promoting global stability in data-driven models of quadratic nonlinear dynamics." PRF (2021)

ROM performance: Flow around 2D Cylinder

2 POD modes + the shift mode were used to recover the ROM



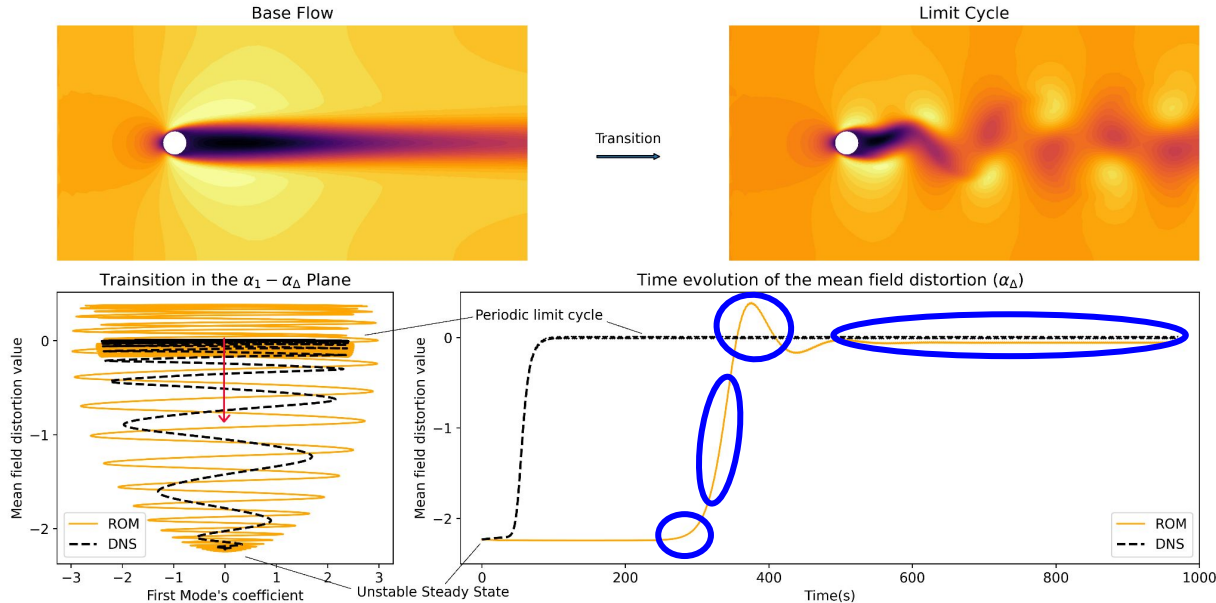
Errors:

- Delayed onset of the instability
- Wrong growth rate (might be recovered with a recalibration*)
- Energy overshoot

*Kaptanoglu, Alan A., et al.
 "Promoting global stability in data-driven models of quadratic nonlinear dynamics." PRF (2021)

ROM performance: Flow around 2D Cylinder

2 POD modes + the shift mode were used to recover the ROM



Errors:

- Delayed onset of the instability
- Wrong growth rate (might be recovered with a recalibration*)
- Energy overshoot
- Mismatch on the periodic cycle

*Kaptanoglu, Alan A., et al.
 "Promoting global stability in data-driven models of quadratic nonlinear dynamics." PRF (2021)

ROM closure

Unboxing the residual

Unboxing the residual

- **Reduce the errors:**
 - Increasing the POD modes (**not desirable**)
 - Use more sophisticated ROMs than POD-Galerkin (generally more **computationally demanding**)
 - Try to **model the residual introducing a closure term**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + f_\theta(a)$$

Unboxing the residual

- **Reduce the errors:**
 - Increasing the POD modes (**not desirable**)
 - Use more sophisticated ROMs than POD-Galerkin (generally more **computationally demanding**)
 - Try to **model the residual introducing a closure term**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + f_\theta(a)$$

Introduce a closure $f_\theta(a)$ is not a new idea

NN:

- Hesthaven, Jan S., and Stefano Ubbiali. "Non-intrusive reduced order modeling of nonlinear problems using neural networks." JCP (2018)
- Barnett, Joshua et al. "Neural-network-augmented projection-based model order reduction for mitigating the Kolmogorov barrier to reducibility." JCP (2023)

Filter:

- Wells, David, et al. "An evolve-then-filter regularized reduced order model for convection-dominated flows." IJNMF (2017)

...

Unboxing the residual

- **Reduce the errors:**
 - Increasing the POD modes (**not desirable**)
 - Use more sophisticated ROMs than POD-Galerkin (generally more **computationally demanding**)
 - Try to **model the residual introducing a closure term**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + f_\theta(a)$$

- **The residual:**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}\mathbf{U}^T \mathbf{u} + (\mathbf{I} - \mathbf{U}\mathbf{U}^T) \mathbf{u})$$

Introduce a closure $f_\theta(a)$ is not a new idea

NN:

- Hesthaven, Jan S., and Stefano Ubbiali. "Non-intrusive reduced order modeling of nonlinear problems using neural networks." JCP (2018)
- Barnett, Joshua et al. "Neural-network-augmented projection-based model order reduction for mitigating the Kolmogorov barrier to reducibility." JCP (2023)

Filter:

- Wells, David, et al. "An evolve-then-filter regularized reduced order model for convection-dominated flows." IJNMF (2017)

...

Unboxing the residual

- **Reduce the errors:**
 - Increasing the POD modes (**not desirable**)
 - Use more sophisticated ROMs than POD-Galerkin (generally more **computationally demanding**)
 - Try to **model the residual introducing a closure term**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + f_\theta(a)$$

- **The residual:**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}\mathbf{U}^T \mathbf{u} + (I - \mathbf{U}\mathbf{U}^T) \mathbf{u}) \approx \mathbf{U}^T g(\mathbf{U}\mathbf{U}^T \mathbf{u}) + \mathbf{U}^T g_{\mathbf{U}\mathbf{U}^T \mathbf{u}} (I - \mathbf{U}\mathbf{U}^T) \mathbf{u}$$

- $g_{\mathbf{U}\mathbf{U}^T \mathbf{u}}$ is the jacobian of the FOM around the state $\mathbf{U}\mathbf{U}^T \mathbf{u}$

Introduce a closure $f_\theta(a)$ is not a new idea

NN:

- Hesthaven, Jan S., and Stefano Ubbiali. "Non-intrusive reduced order modeling of nonlinear problems using neural networks." JCP (2018)
- Barnett, Joshua et al. "Neural-network-augmented projection-based model order reduction for mitigating the Kolmogorov barrier to reducibility." JCP (2023)

Filter:

- Wells, David, et al. "An evolve-then-filter regularized reduced order model for convection-dominated flows." IJNMF (2017)

...

Unboxing the residual

- **Reduce the errors:**
 - Increasing the POD modes (**not desirable**)
 - Use more sophisticated ROMs than POD-Galerkin (generally more **computationally demanding**)
 - Try to **model the residual introducing a closure term**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + f_\theta(a)$$

- **The residual:**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}\mathbf{U}^T \mathbf{u} + (I - \mathbf{U}\mathbf{U}^T) \mathbf{u}) \approx \mathbf{U}^T g(\mathbf{U}\mathbf{U}^T \mathbf{u}) + \mathbf{U}^T g_{\mathbf{U}\mathbf{U}^T \mathbf{u}} (I - \mathbf{U}\mathbf{U}^T) \mathbf{u}$$

- $g_{\mathbf{U}\mathbf{U}^T \mathbf{u}}$ is the jacobian of the FOM around the state $\mathbf{U}\mathbf{U}^T \mathbf{u}$
- **The uncorrected ROMs**

Introduce a closure $f_\theta(a)$ is not a new idea

NN:

- Hesthaven, Jan S., and Stefano Ubbiali. "Non-intrusive reduced order modeling of nonlinear problems using neural networks." JCP (2018)
- Barnett, Joshua et al. "Neural-network-augmented projection-based model order reduction for mitigating the Kolmogorov barrier to reducibility." JCP (2023)

Filter:

- Wells, David, et al. "An evolve-then-filter regularized reduced order model for convection-dominated flows." IJNMF (2017)

...

Unboxing the residual

- **Reduce the errors:**
 - Increasing the POD modes (**not desirable**)
 - Use more sophisticated ROMs than POD-Galerkin (generally more **computationally demanding**)
 - Try to **model the residual introducing a closure term**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}a) + f_\theta(a)$$

- **The residual:**

$$\frac{da}{dt} = \mathbf{U}^T g(\mathbf{U}\mathbf{U}^T \mathbf{u} + (I - \mathbf{U}\mathbf{U}^T) \mathbf{u}) \approx \mathbf{U}^T g(\mathbf{U}\mathbf{U}^T \mathbf{u}) + \mathbf{U}^T g_{\mathbf{U}\mathbf{U}^T \mathbf{u}}(I - \mathbf{U}\mathbf{U}^T) \mathbf{u}$$

- $g_{\mathbf{U}\mathbf{U}^T \mathbf{u}}$ is the jacobian of the FOM around the state $\mathbf{U}\mathbf{U}^T \mathbf{u}$
- **The uncorrected ROMs**
- **The residual** accounts for the dynamics of the **complement** of $\mathbf{U}\mathbf{U}^T$ on $\mathbf{U}\mathbf{U}^T$

Introduce a closure $f_\theta(a)$ is not a new idea

NN:

- Hesthaven, Jan S., and Stefano Ubbiali. "Non-intrusive reduced order modeling of nonlinear problems using neural networks." JCP (2018)
- Barnett, Joshua et al. "Neural-network-augmented projection-based model order reduction for mitigating the Kolmogorov barrier to reducibility." JCP (2023)

Filter:

- Wells, David, et al. "An evolve-then-filter regularized reduced order model for convection-dominated flows." IJNMF (2017)

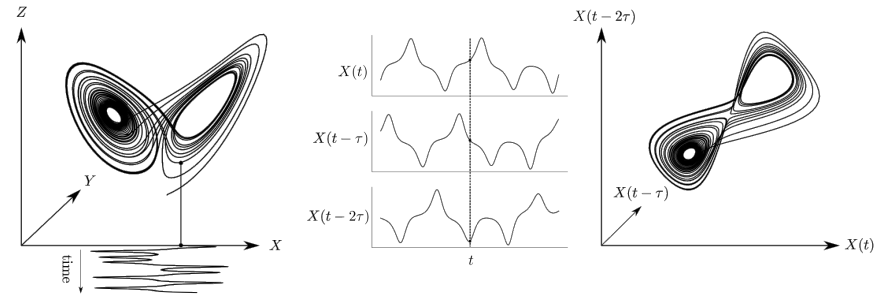
...

The unobserved dynamics

- **Goal: Recover the unobserved dynamics from the observed POD modes**

The unobserved dynamics

- **Goal: Recover the unobserved dynamics from the observed POD modes**
 - According to the **Takens theorem**:
 - Delayed coordinates of **observable coordinates** used to recover the unobserved phase space.
 - The reconstructed phase space is a diffeomorphisms of the original one i.e. the topology is preserved but not the geometric shape.



The unobserved dynamics

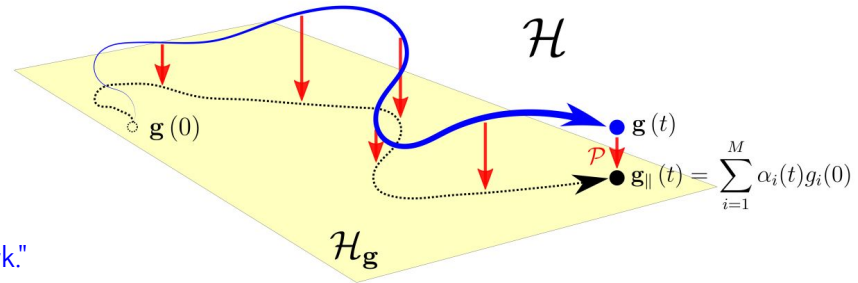
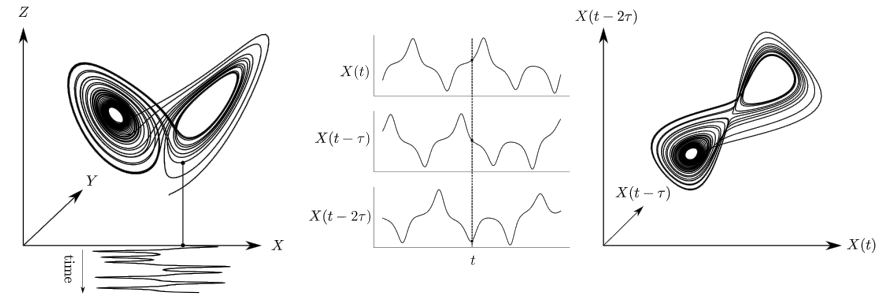
■ Goal: Recover the unobserved dynamics from the observed POD modes

- According to the **Takens theorem**:
 - Delayed coordinates of **observable coordinates** used to recover the unobserved phase space.
 - The reconstructed phase space is a diffeomorphism of the original one i.e. the topology is preserved but not the geometric shape.
- The **Mori-Zwanzig formalism** provides a closed form for the dynamics of partially observed systems.

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_0^t K(\mathbf{a}(s)) ds + F(t)$$

$\mathbf{a}(t)$ being the reduced coordinates

Yen Ting Lin, et al. "Data-driven learning for the Mori-Zwanzig formalism: a generalization of the Koopman learning framework." SIAM Journal on Applied Dynamical Systems (2021)



Demystify the MZ formalism

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_0^t K(\mathbf{a}(s)) ds + F(t)$$

Demystify the MZ formalism

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_0^t K(\mathbf{a}(s)) ds + F(t)$$

- The **markovian term**: describes the dynamics of the FOM system in the observed space \Rightarrow POD-ROM

Demystify the MZ formalism

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_0^t K(\mathbf{a}(s)) ds + F(t)$$

- The **markovian term**: describes the dynamics of the FOM system in the observed space \Rightarrow POD-ROM
- The **non-markovian term**: It is a convolution of past informations and it describes the projection of the unobserved dynamics onto the observed space.

Demystify the MZ formalism

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_0^t K(\mathbf{a}(s)) ds + F(t)$$

- The **markovian term**: describes the dynamics of the FOM system in the observed space \Rightarrow POD-ROM
- The **non-markovian term**: It is a convolution of past informations and it describes the projection of the unobserved dynamics onto the observed space.
- The **source term**: accounts for the incomplete knowledge of the initial condition and system dynamics.

Demystify the MZ formalism

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_0^t K(\mathbf{a}(s)) ds + F(t)$$

- The **markovian term**: describes the dynamics of the FOM system in the observed space \Rightarrow POD-ROM
- The **non-markovian term**: It is a convolution of past informations and it describes the projection of the unobserved dynamics onto the observed space.
- The **source term**: accounts for the incomplete knowledge of the initial condition and system dynamics.

Many data-driven system identification methods can be framed in the MZ formalism (e.g. ARMAX). Convolution on past informations is a standard procedure also for generative models in Deep Learning (e.g. WaveNet, LSTM, Transformers)

Demystify the MZ formalism

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_0^t K(\mathbf{a}(s)) ds + F(t)$$

- The **markovian term**: describes the dynamics of the FOM system in the observed space \Rightarrow POD-ROM
- The **non-markovian term**: It is a convolution of past informations and it describes the projection of the unobserved dynamics onto the observed space.
- The **source term**: accounts for the incomplete knowledge of the initial condition and system dynamics.

Many data-driven system identification methods can be framed in the MZ formalism (e.g. ARMAX). Convolution on past informations is a standard procedure also for generative models in Deep Learning (e.g. WaveNet, LSTM, Transformers)



We need some assumptions to effectively apply the MZ formalism

Hypothesis

1. The FOM is observable via the first k POD modes:
We didn't prove it but, higher harmonic POD modes are enslaved to dominant POD modes. The dynamics of the last POD modes are non-linearly correlated with the first POD modes*
2. The dynamics associated to the unobserved POD modes is dissipative**

*Loiseau, Jean-Christophe, and Steven L. Brunton. "Constrained sparse Galerkin regression." JFM (2018)

*Callaham, Jared L., et al. "On the role of nonlinear correlations in reduced-order modelling." JFM (2022)

**Ahmed, Shady E., et al. "On closures for reduced order models—A spectrum of first-principle to machine-learned avenues." PoF (2021).

Hypothesis

1. The FOM is observable via the first k POD modes:
We didn't prove it but, higher harmonic POD modes are enslaved to dominant POD modes. The dynamics of the last POD modes are non-linearly correlated with the first POD modes*
2. The dynamics associated to the unobserved POD modes is dissipative**

Ignore the source term



$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t K(\mathbf{a}(s)) ds$$

*Loiseau, Jean-Christophe, and Steven L. Brunton. "Constrained sparse Galerkin regression." JFM (2018)

*Callaham, Jared L., et al. "On the role of nonlinear correlations in reduced-order modelling." JFM (2022)

**Ahmed, Shady E., et al. "On closures for reduced order models—A spectrum of first-principle to machine-learned avenues." PoF (2021).

Hypothesis

1. The FOM is observable via the first k POD modes:
We didn't prove it but, higher harmonic POD modes are enslaved to dominant POD modes. The dynamics of the last POD modes are non-linearly correlated with the first POD modes*
2. The dynamics associated to the unobserved POD modes is dissipative**

Ignore the source term



$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t K(\mathbf{a}(s)) ds$$

Exponential kernel



$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t \mathbf{a}(s) e^{\lambda(s-t)} ds, \quad \lambda \in \mathbb{R}_+$$

*Loiseau, Jean-Christophe, and Steven L. Brunton. "Constrained sparse Galerkin regression." JFM (2018)

*Callaham, Jared L., et al. "On the role of nonlinear correlations in reduced-order modelling." JFM (2022)

**Ahmed, Shady E., et al. "On closures for reduced order models—A spectrum of first-principle to machine-learned avenues." PoF (2021).

Hypothesis

1. The FOM is observable via the first k POD modes:
We didn't prove it but, higher harmonic POD modes are enslaved to dominant POD modes. The dynamics of the last POD modes are non-linearly correlated with the first POD modes*
2. The dynamics associated to the unobserved POD modes is dissipative**

Ignore the source term



$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t K(\mathbf{a}(s)) ds$$

Exponential kernel



$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t \mathbf{a}(s) e^{\lambda(s-t)} ds, \quad \lambda \in \mathbb{R}_+$$

The exponential kernel is not the unique choice. One possibility is to use Recurrent NNs*** aiming at predict $residual_t = RNN(\mathbf{a}_t, residual_{t-1})$. \Rightarrow hard to train if ROM is unstable and diverges after few iterations.

*Loiseau, Jean-Christophe, and Steven L. Brunton. "Constrained sparse Galerkin regression." JFM (2018)

*Callaham, Jared L., et al. "On the role of nonlinear correlations in reduced-order modelling." JFM (2022)

**Ahmed, Shady E., et al. "On closures for reduced order models—A spectrum of first-principle to machine-learned avenues." PoF (2021).

***Wang, Qian, et al.. "Recurrent neural network closure of parametric POD-Galerkin reduced-order models based on the Mori-Zwanzig formalism." JCP (2020)

The exponential kernel

- **It is a convenient choice:**

- To correlate the residual with recent observations (and not far in the past)
- Well describe a dissipative dynamics as long as $\lambda \in \mathbb{R}_+$
- It is a filter of width $1/\lambda$. Already used in LES* and SFD**
- Easy to integrate with an augmented ODE.

$$\begin{aligned}\dot{\mathbf{a}} &= ROM(\mathbf{a}) + \mathbf{y} \\ \dot{\mathbf{y}} &= \mathbf{a} - \lambda \mathbf{y}\end{aligned}$$

\mathbf{y} is like the **memory** gate for recurrent Neural Networks or Reservoir Computing

- \mathbf{a}, \mathbf{y} have the same dimension. This might introduce an information bottleneck
- We introduced an encoding $\mathbf{E} : \mathbb{R}^N \rightarrow \mathbb{R}^{n_E}$ and decoding $\mathcal{R} : \mathbb{R}^{n_E} \rightarrow \mathbb{R}^N$ map

*Pruett, C. D., et al. "The temporally filtered Navier–Stokes equations: properties of the residual stress." PoF (2003)

**Åkervik, Espen, et al. "Steady solutions of the Navier–Stokes equations by selective frequency damping." PoF (2006)

Deep Learning solution CD-ROM formulation

CD-ROM

$$\begin{aligned}\dot{\mathbf{a}} &= ROM(\mathbf{a}) + \mathcal{R}_{\theta_R}(\mathbf{y}) \\ \dot{\mathbf{y}} &= \mathbf{E}_{\theta_E}(\mathbf{a}) - \Lambda \mathbf{y}\end{aligned}$$

- $\mathbf{E}_{\theta_E}, \mathcal{R}_{\theta_R}$ are MLP and Λ is a trainable diagonal $\mathbb{R}_+^{n_E \times n_E}$ matrix

CD-ROM

$$\begin{aligned}\dot{\mathbf{a}} &= ROM(\mathbf{a}) + \mathcal{R}_{\theta_R}(\mathbf{y}) \\ \dot{\mathbf{y}} &= \mathbf{E}_{\theta_E}(\mathbf{a}) - \Lambda \mathbf{y}\end{aligned}$$

- $\mathbf{E}_{\theta_E}, \mathcal{R}_{\theta_R}$ are MLP and Λ is a trainable diagonal $\mathbb{R}_+^{n_E \times n_E}$ matrix

▪ The training procedure:

$$\begin{aligned}\arg \min_{\theta_E, \theta_R, \Lambda} &= J(\mathbf{a}) = \|\mathbf{a}_{target} - \mathbf{a}\| \\ \text{s.t. } &\dot{\mathbf{a}} = \text{CD-ROM}(\mathbf{a}, \theta_E, \theta_R, \Lambda)\end{aligned}$$

- The target **residuals don't need to be computed** to train the MLPs. The distance between the target trajectory ($\mathbf{a}_{target}(t) = \mathbf{U}^T \mathbf{u}_{DNS}(t)$) and the predicted one is considered as loss function.

CD-ROM

$$\begin{aligned}\dot{\mathbf{a}} &= ROM(\mathbf{a}) + \mathcal{R}_{\theta_R}(\mathbf{y}) \\ \dot{\mathbf{y}} &= \mathbf{E}_{\theta_E}(\mathbf{a}) - \Lambda \mathbf{y}\end{aligned}$$

- $\mathbf{E}_{\theta_E}, \mathcal{R}_{\theta_R}$ are MLP and Λ is a trainable diagonal $\mathbb{R}_+^{n_E \times n_E}$ matrix

▪ The training procedure:

$$\begin{aligned}\arg \min_{\theta_E, \theta_R, \Lambda} &= J(\mathbf{a}) = \|\mathbf{a}_{target} - \mathbf{a}\| \\ \text{s.t. } &\dot{\mathbf{a}} = \text{CD-ROM}(\mathbf{a}, \theta_E, \theta_R, \Lambda)\end{aligned} \quad \longrightarrow \quad \begin{aligned}L(\mathbf{a}(t), \mathbf{a}^\dagger(t), \theta_E, \theta_R, \Lambda) &= \\ J(\mathbf{a}(t)) - \mathbf{a}^\dagger(t) \int_0^T \dot{\mathbf{a}}(t) - \text{CD-ROM}(\mathbf{a}(t), \theta_E, \theta_R, \Lambda) dt\end{aligned}$$

- The target **residuals don't need to be computed** to train the MLPs. The distance between the target trajectory ($\mathbf{a}_{target}(t) = \mathbf{U}^T \mathbf{u}_{DNS}(t)$) and the predicted one is considered as loss function.
- The augmented Lagrangian is minimized to solve the constrained optimization problem*
 - The **gradients are back propagated through the ROM model** \Rightarrow the ROM sensitivity is taken into account
 - Straightforward to compute **vector-Jacobian products with automatic differentiation**
 - A portion of the true trajectory is used to **initialize the memory \mathbf{y}**

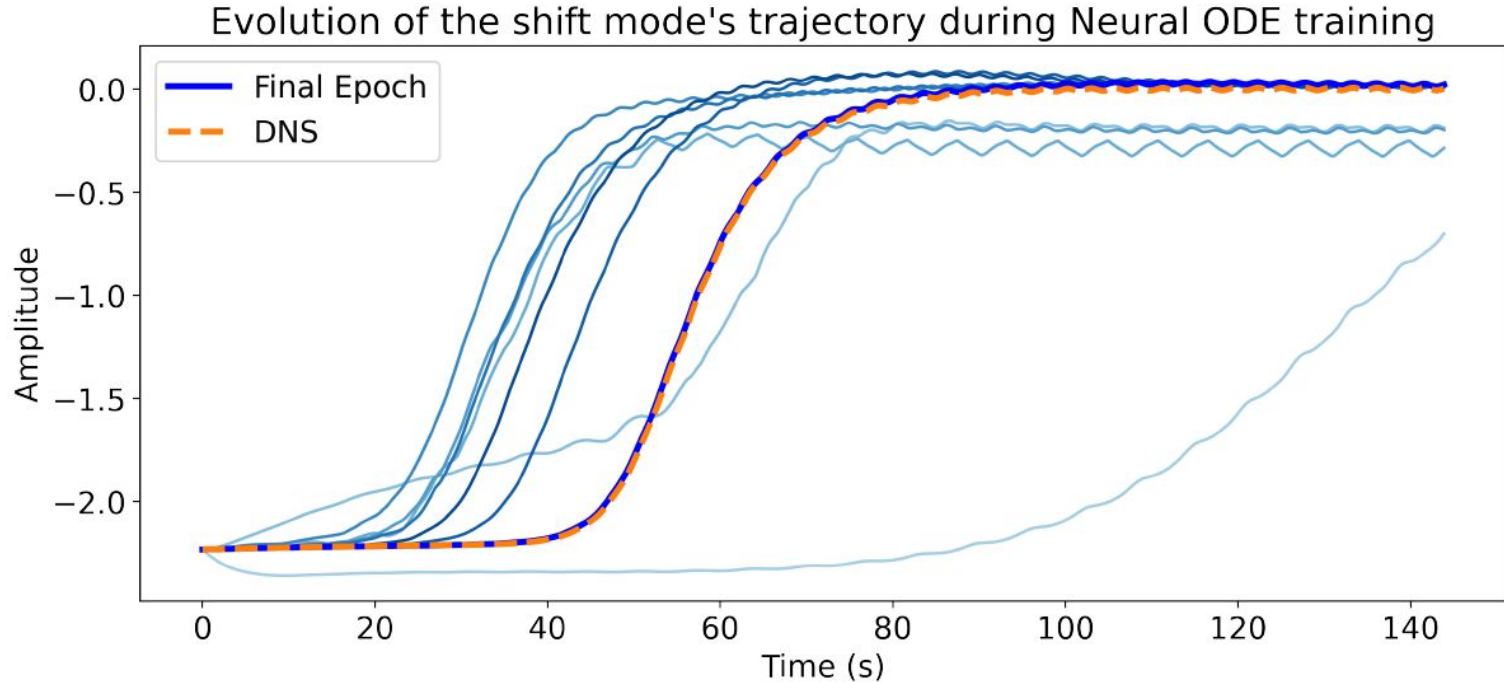
*Chen, Ricky TQ, et al. "Neural ordinary differential equations." Advances in neural information processing systems (2018).

Results

- Flow around 2D cylinder
- Fluidic Pinball
- Parametric Kuramoto-Sivashinsky

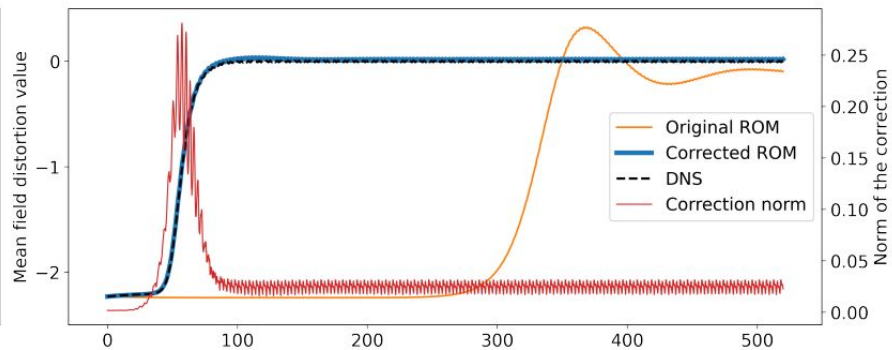
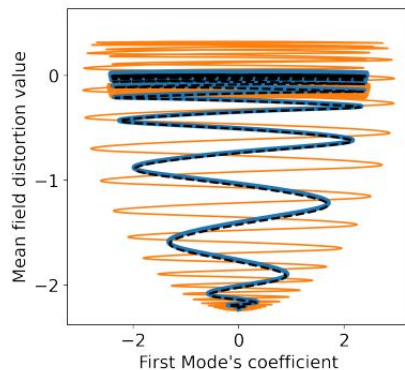
Flow around 2D cylinder

2 POD modes + the shift mode were used to train the ROM correction

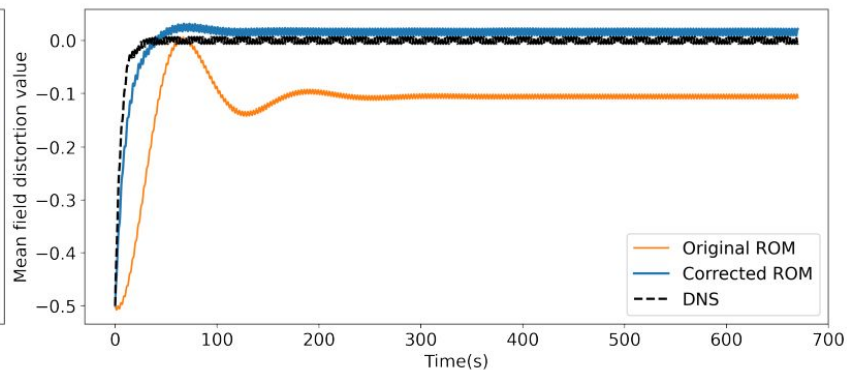
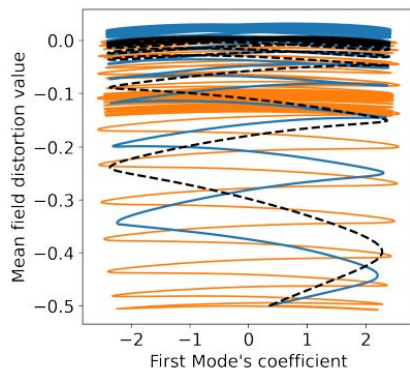


Flow around 2D cylinder

Train

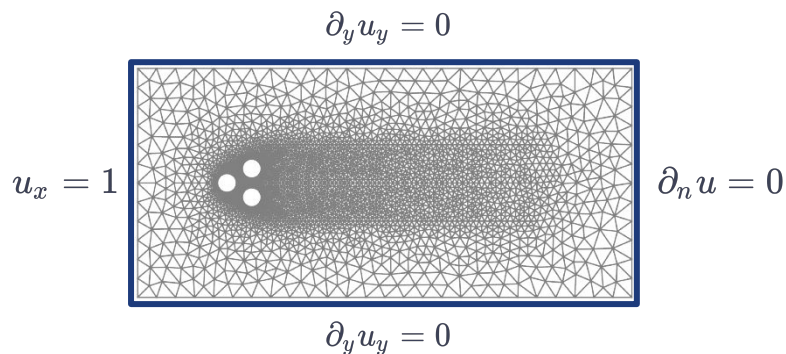


Test



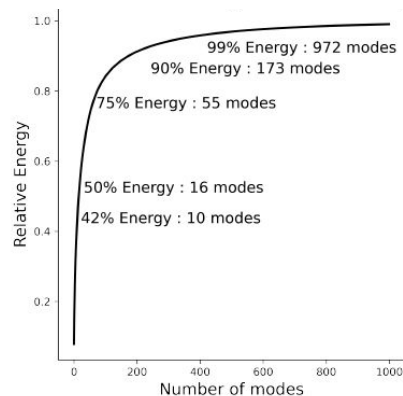
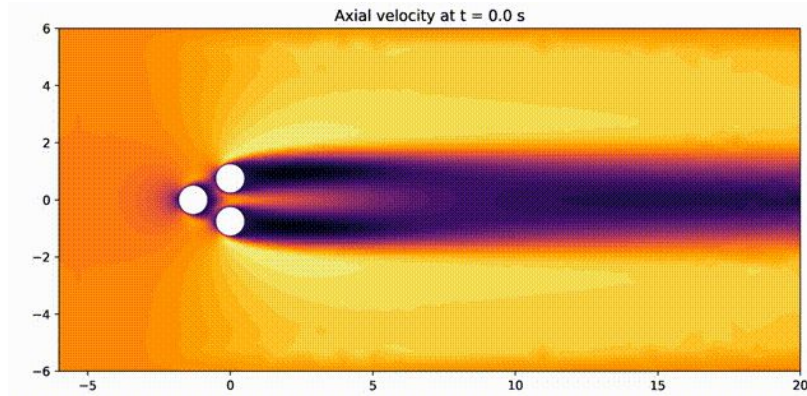
Fluidic Pinball

Chaotic Fluidic Pinball*



CD-ROM describing the chaotic dynamics of the flow behind 3 cylinders at $Re=130$

*N. Deng, et al. "Low-order model for successive bifurcations of the fluidic pinball." JFM (2019)

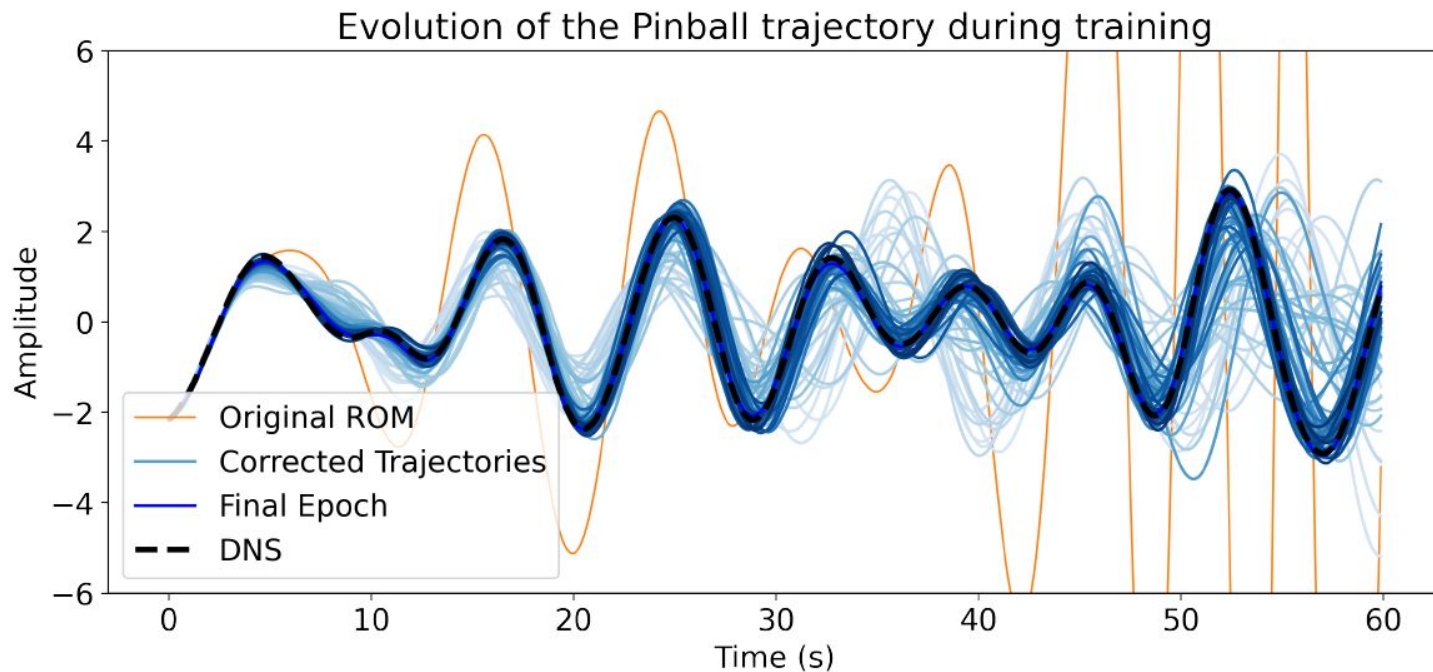


More than 900 POD modes are necessary to capture the 99% of the informations. Not so ROM!

Dataset: long chunked DNS trajectory

Fluidic Pinball

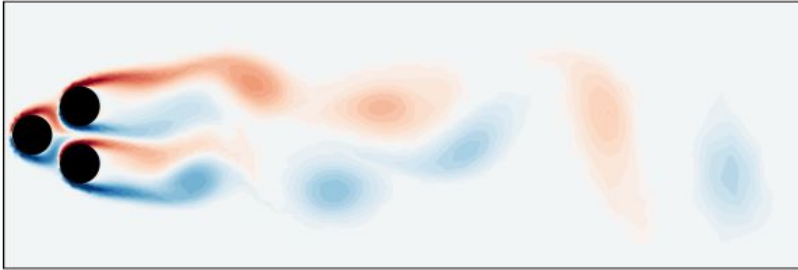
10 POD modes were used to train the ROM correction



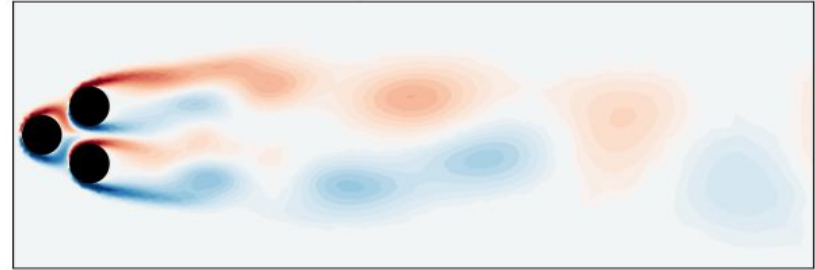
Fluidic Pinball

Vorticity field comparison after 110 time units

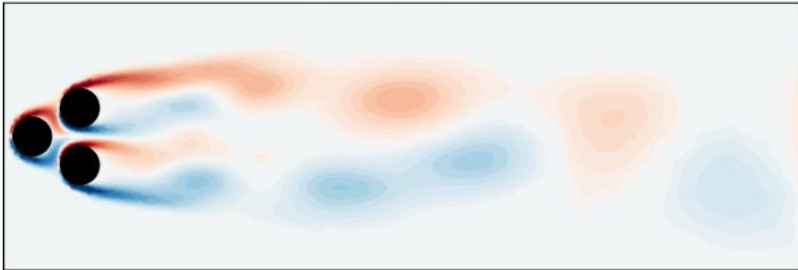
DNS



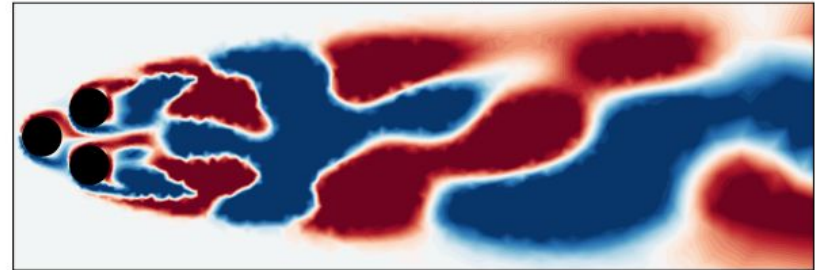
Projected DNS



Corrected ROM

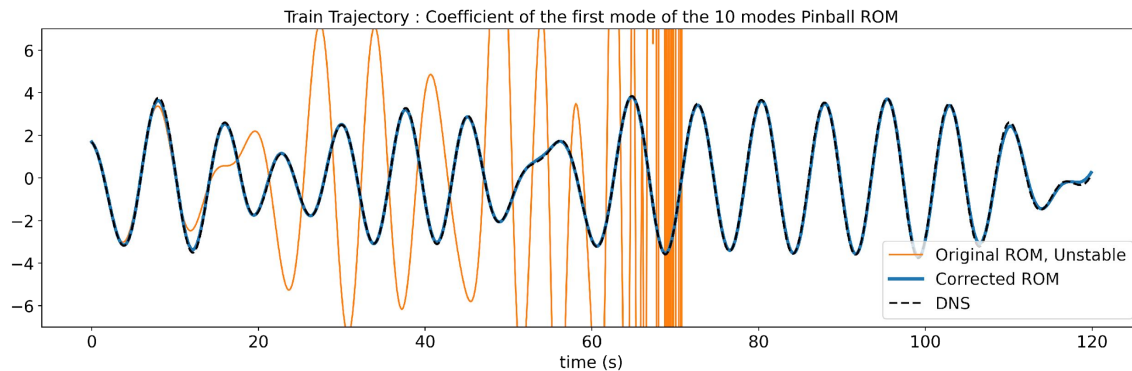


Uncorrected ROM (diverged)

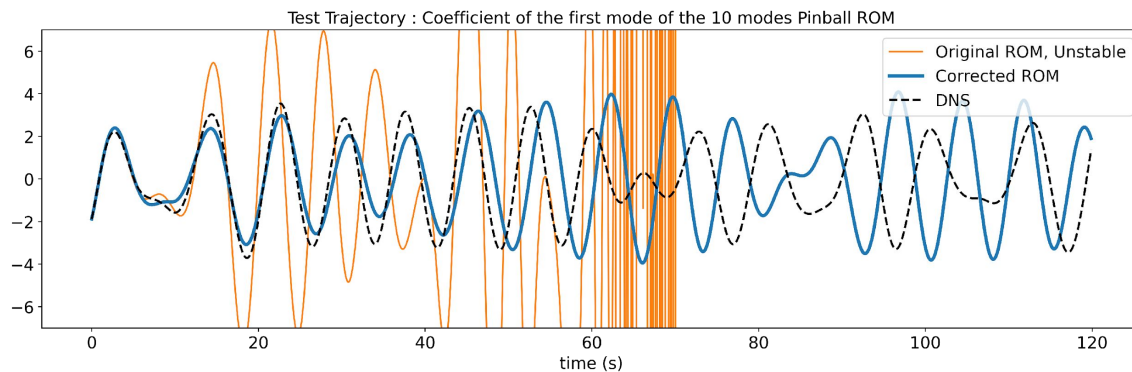


Fluidic Pinball

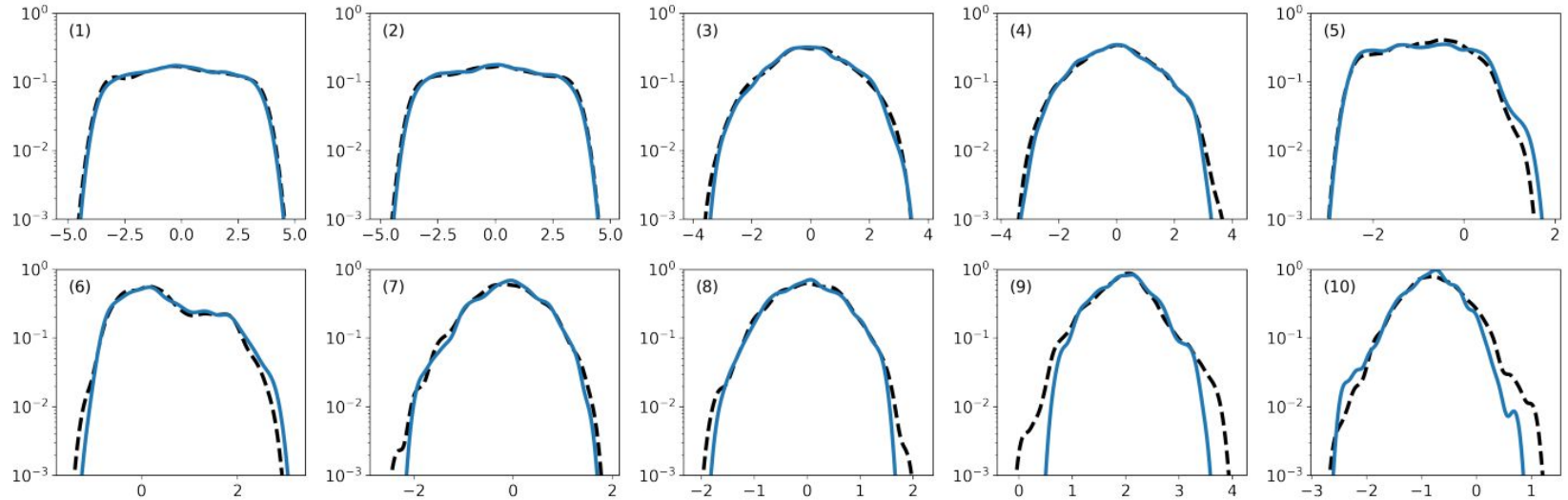
Train



Test



Fluidic Pinball

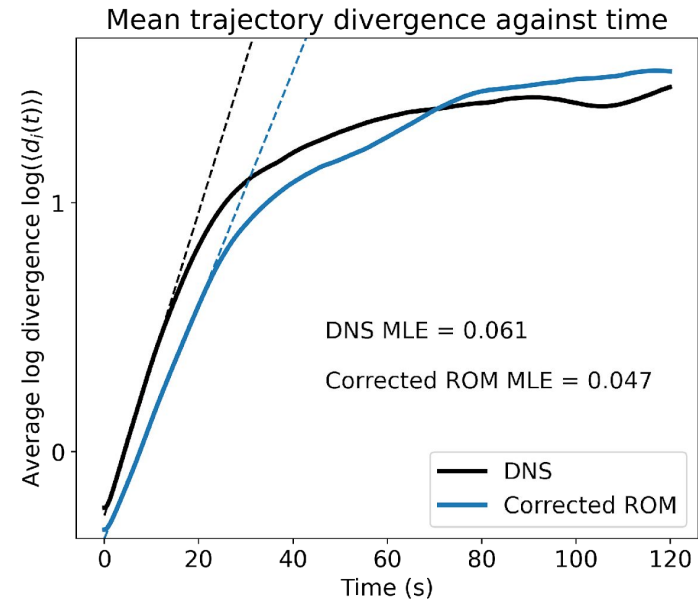
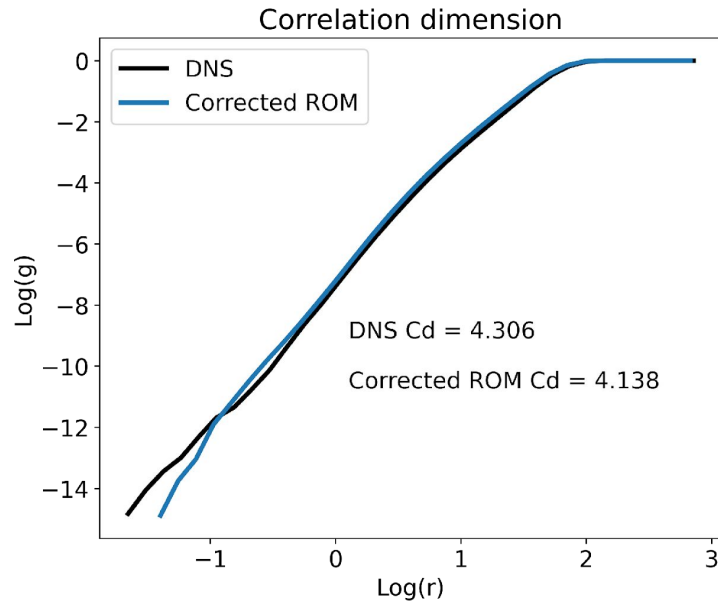


Estimated probability density functions for the coefficient of each mode. Plain blue line: Statistics of the trajectory simulated with the **corrected model**; dotted black line: Statistics of the projected DNS data. Labels refer to the mode index.

Fluidic Pinball

$$C(\varepsilon) = \lim_{N \rightarrow \infty} \frac{1}{N^2} \sum_{\substack{i,j=1 \\ i \neq j}}^N \Phi(\varepsilon - \|x_i - x_j\|)$$

$$\delta \mathbf{X}(t) \approx e^{\lambda t} \delta \mathbf{X}_0$$

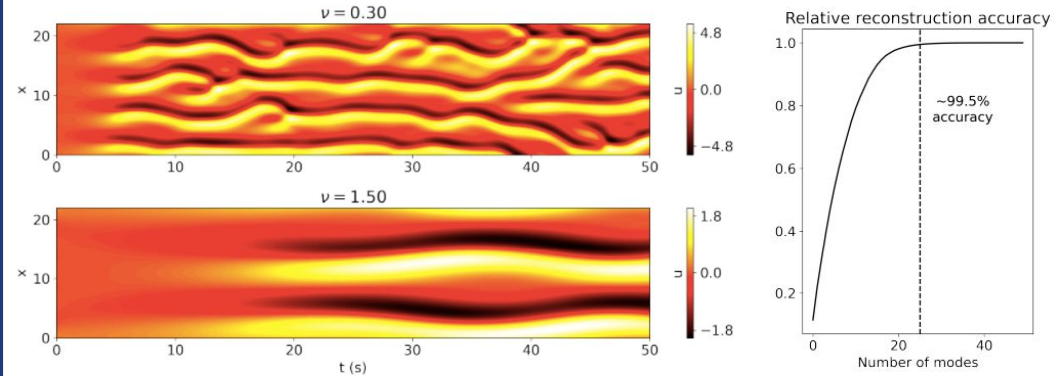


Parametric Kuramoto-Sivashinsky

1D Kuramoto-Sivashinsky

$$\frac{\partial u}{\partial t} = -\frac{1}{2} \nabla \cdot u^2 - \Delta u - \nu \Delta^2 u$$

- ν viscosity like parameter $\in [0.3, 1.5]$
- Periodic BC
- 513 Fourier modes



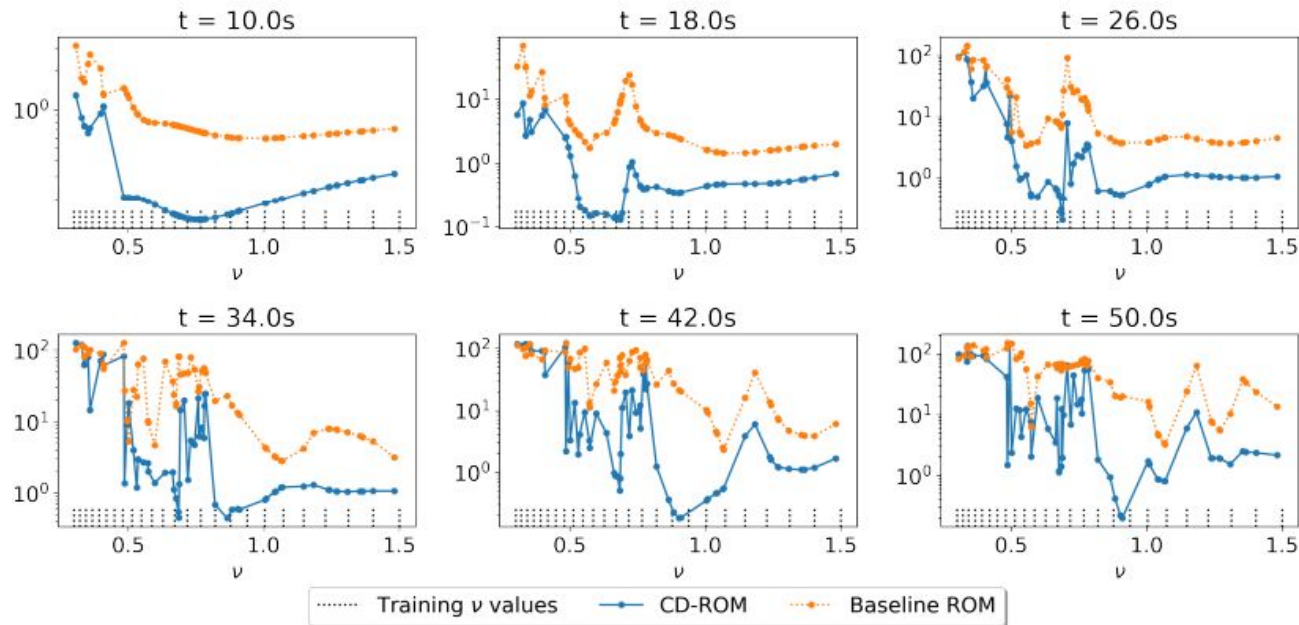
$$\begin{aligned} \frac{d}{dt} \mathbf{a} &= -\frac{1}{2} \mathbf{a}^\top \tilde{\mathcal{Q}} \mathbf{a} - \tilde{\mathcal{L}} \mathbf{a} - \nu \tilde{\mathcal{L}}^2 \mathbf{a} + \mathcal{R}(\mathbf{y}, \nu; \boldsymbol{\theta}_{\mathcal{R}}), \\ \frac{d}{d} \mathbf{y} &= \mathbf{E}(\mathbf{a}, \nu; \boldsymbol{\theta}_{\mathbf{E}}) - \Lambda \mathbf{y}. \end{aligned}$$

- For low ν values the dynamics is chaotic
- 25 parameter values log-uniformly sampled
- 25 POD modes
- Parameter as input in the NNs

Parametric Kuramoto-Sivashinsky

$$d(t, \nu) = \|\mathbf{a}(t, \nu) - \mathbf{a}^*(t, \nu)\|_2$$

Computed for unseen ν



Conclusions & perspectives

Conclusions

■ Conclusions:

- A new correction has been proposed leveraging on MZ formalism
- The proposed method can be integrated in time with standard time-marching schemes → important in control scenario
- For all cases the same data used to recover the POD modes has been used to train the MLPs
- Works well also with unstable ROM since the sensitivity of the ROM is considered during the training
- The time scales used to correct the ROM are interpretable via $1/\lambda$

■ Perspectives:

- Better exploit the MZ-formalism to have a more theoretically grounded architecture
- Explore new temporal kernels

Further conclusions

markovian **non-markovian**

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t \mathbf{a}(s) e^{\lambda(s-t)} ds, \quad \lambda \in \mathbb{R}_+$$

Menier, E., et al. CD-ROM: Complemented Deep-Reduced order model. CMAME (2023)

- The **markovian** term is the **POD-Galerkin ROM**
- The **non-markovian** kernel takes care of the **truncation errors**

Further conclusions

markovian **non-markovian**

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t \mathbf{a}(s) e^{\lambda(s-t)} ds, \quad \lambda \in \mathbb{R}_+$$

Menier, E., et al. CD-ROM: Complemented Deep-Reduced order model. CMAME (2023)

- The **markovian** term is the **POD-Galerkin ROM**
- The **non-markovian** kernel takes care of the **truncation errors**

Menier, E., et al. Continuous Methods: Adaptively intrusive reduced order model closure. NEURIPS workshop (2022)

- The **markovian** term is the **incomplete ROM**
- The **non-markovian** kernel takes care of the **non-reducible terms**

Further conclusions

markovian **non-markovian**

$$\frac{d}{dt} \mathbf{a}(t) = \Omega(\mathbf{a}(t)) + \int_{-\infty}^t \mathbf{a}(s) e^{\lambda(s-t)} ds, \quad \lambda \in \mathbb{R}_+$$

Menier, E., et al. CD-ROM: Complemented Deep-Reduced order model. CMAME (2023)

- The **markovian** term is the **POD-Galerkin ROM**
- The **non-markovian** kernel takes care of the **truncation errors**

Menier, E., et al. Continuous Methods: Adaptively intrusive reduced order model closure. NEURIPS workshop (2022)

- The **markovian** term is the **incomplete ROM**
- The **non-markovian** kernel takes care of the **non-reducible terms**

Menier, E., et al. Interpretable learning of effective dynamics for multiscale systems. arXiv preprint arXiv:2309.05812 (2023)

- The **markovian** term is the approximation of the **Koopman operator** with a finite number of observables
- The **non-markovian** kernel takes care of the **finite number of observables**

Check out his poster!

**POWERED
BY TRUST**
