Introduction à la réduction de modèles en Dynamique des Fluides

Laurent Cordier & Lionel Mathelin

TDMF, November 29, 2017, Orsay

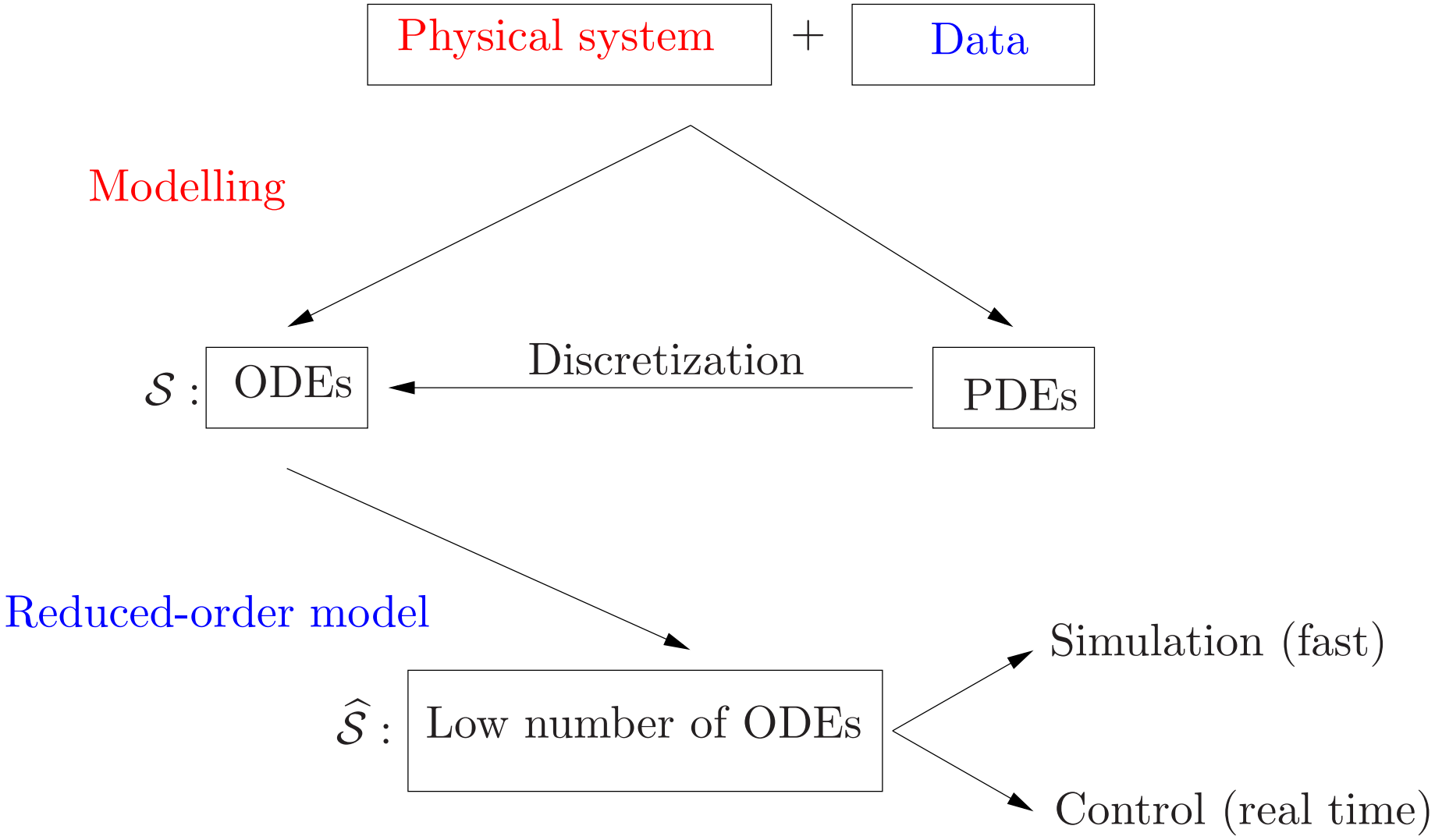
Reduced-Order Modelling

- Ex. from Spalart et al. (1997): wing considered at cruising flight conditions *i.e.* $\mathrm{Re} = \mathcal{O}(10^7)$. Converged solution obtained for
  - about 10^{11} grid points, about 5×10^6 time steps.

    40 years for the first LES of a wing !!
- Nearly impossible to solve numerically problems where
  - either, a great number of resolution of the state equations is necessary (continuation methods, parametric studies, optimization problems or optimal control,. . .),
  - either a solution in real time is searched (active control in closed-loop control for instance).
- Objective: reduce the number of degrees of freedom.

  In fluid mechanics/turbulence :
  - Prandtl boundary layer equations,
  - RANS models ($k - \epsilon$, $k - \omega$),
  - Large Eddy Simulation (LES),
  - Low-order dynamical system based on *POD* (Lumley, 1967),
  - Reduced-order models based on balanced, DMD and/or global modes.

Reduced-Order Modelling



Outline

Outline

Eigenvalue Decomposition

Definition and application

- For $S \in \mathbb{C}^{n \times n}$, $\boldsymbol{v}_i \in \mathbb{C}^n$ and $\lambda_i \in \mathbb{C}$ are eigen-vectors/-values if:

$$S\,V = V\,\Lambda,$$

with $V = (\boldsymbol{v}_1,\ \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n) \in \mathbb{C}^{n \times n}$ and $\Lambda = \mathrm{diag}\,(\lambda_1, \lambda_2, \ldots, \lambda_n)$.

- If S has n linearly independent eigenvectors $\boldsymbol{v}_i$ then

$$S = V\,\Lambda\,V^{-1} \quad \text{eigendecomposition of } A$$

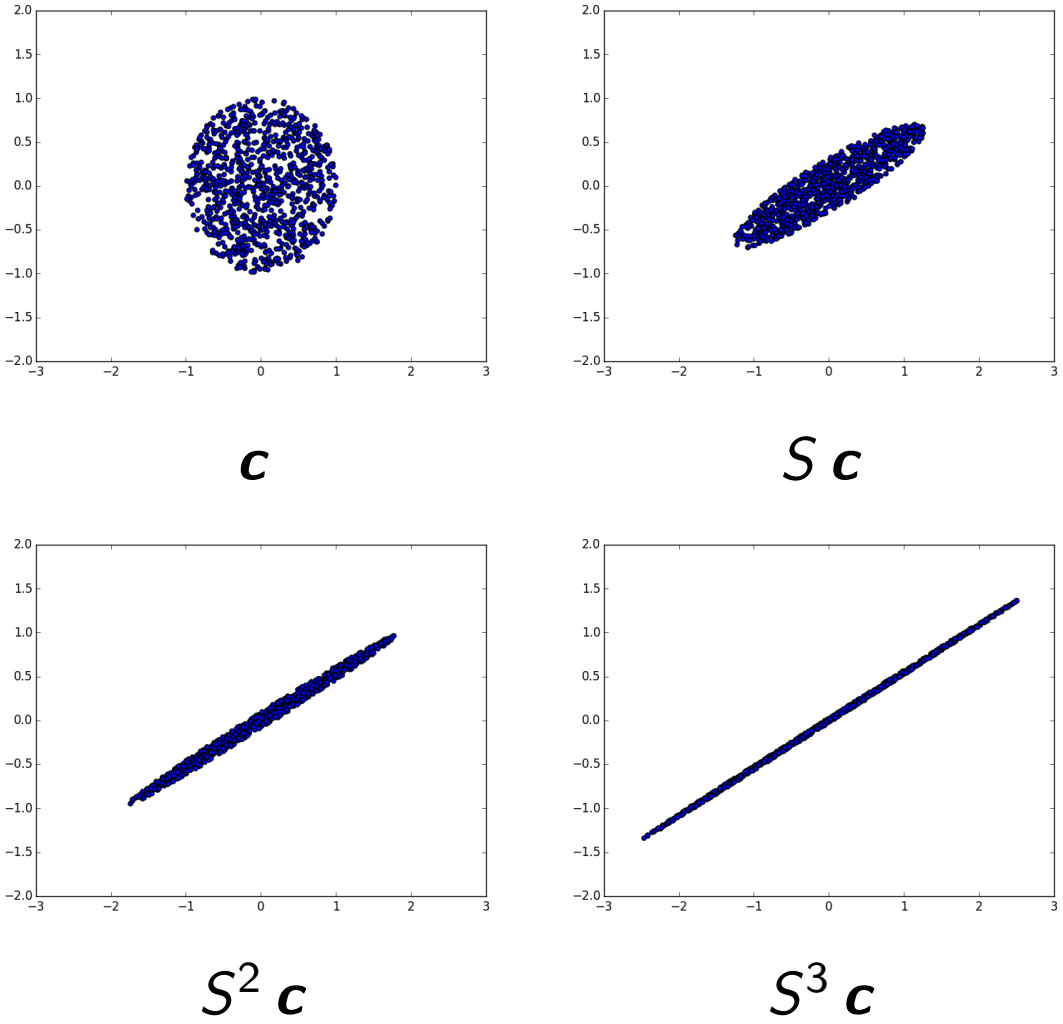
- Linear dynamical systems

$$\dot{\boldsymbol{x}} = S\,\boldsymbol{x}.$$

$$\begin{aligned} \boldsymbol{x}\,(t) &= \exp\,(S\,t)\ \boldsymbol{x}\,(t_0)\,, \\ &= V\ \exp\,(\Lambda\,t)\ V^{-1}\,\boldsymbol{x}\,(t_0)\,. \end{aligned}$$

  - $\mathrm{Re}(\lambda_k)$: growth rate (> 0) ; decay rate (< 0)
  - $\mathrm{Im}(\lambda_k)$: frequency
  - System stable if $\mathrm{Re}(\lambda_k) < 0 \quad \forall k$

Let $\mathcal{C} = \{\boldsymbol{c}_i \mid \|\boldsymbol{c}_i\|_2 = 1\}$ and $S = \begin{pmatrix} 1.2 & 0.4 \\ 0.5 & 0.5 \end{pmatrix}$.



Eigenvectors capture the directions in which vectors can grow or shrink.

Singular Value Decomposition (SVD)

Definition

$$\boxed{S = U\Sigma V^H} \in \mathbb{C}^{N_x \times N_t} \qquad \text{with}$$

- $U \in \mathbb{C}^{N_x \times N_x}$ unitary: $UU^H = U^H U = I_{N_x}$

  Left singular vectors: $U = (u_1, u_2, \cdots, u_{N_x})$
- $V \in \mathbb{C}^{N_t \times N_t}$ unitary: $VV^H = V^H V = I_{N_t}$

  Right singular vectors: $V = (v_1, v_2, \cdots, v_{N_t})$
- Σ 'diagonal' matrix

Singular values: $\Sigma = \text{diag}(\sigma_1, \cdots, \sigma_p, 0 \cdots, 0)$ with $p = \min(N_x, N_t)$

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_p = 0 \quad \text{where} \quad r = \text{rank}(S) \leq p.$$

$$\Sigma = \begin{pmatrix} \Sigma_p & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \quad ; \quad \Sigma_p = \begin{pmatrix} \sigma_1 & 0 & 0 \\ \vdots & \ddots & 0 \\ 0 & \cdots & \sigma_p \end{pmatrix}$$

SVD

Example for $\boxed{N_x < N_t}$ i.e. $p = N_x$

$S = U\Sigma V^H$ where S has more columns than rows.

$$S = \begin{pmatrix} \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_{N_x} \end{pmatrix} \left(\begin{array}{cccc|cccc} \sigma_1 & & & & 0 & \cdots & \cdots & 0 \\ & \ddots & & & \vdots & & & \vdots \\ & & \ddots & & \vdots & & & \vdots \\ & & & \ddots & \vdots & & & \vdots \\ & & & \sigma_{N_x} & 0 & \cdots & \cdots & 0 \end{array}\right) \left(\begin{array}{c} \boldsymbol{v}_1^H \\ \vdots \\ \vdots \\ \boldsymbol{v}_{N_x}^H \\ \hline \boldsymbol{v}_{N_x+1}^H \\ \vdots \\ \vdots \\ \boldsymbol{v}_{N_t}^H \end{array}\right)$$

SVD

Example for $\boxed{N_x > N_t}$ i.e. $p = N_t$

$S = U\Sigma V^H$ where S has more rows than columns.

$$S = \begin{pmatrix} \boldsymbol{u}_1 & \cdots & \boldsymbol{u}_{N_t} & \boldsymbol{u}_{N_t+1} \cdots & \boldsymbol{u}_{N_x} \end{pmatrix} \left(\begin{array}{ccccc} \sigma_1 & & & & \\ & \ddots & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \sigma_{N_t} \\ \hline 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & & & & \vdots \\ \vdots & & & & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \end{array}\right) \begin{pmatrix} \boldsymbol{v}_1^H \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \boldsymbol{v}_{N_t}^H \end{pmatrix}.$$

Truncated SVD approximations

Dyadic expansion

⋆ If $r = \text{rank}(S)$, then the SVD of $S \in \mathbb{C}^{N_x \times N_t}$ can be written as

$$S = \begin{pmatrix} \underline{U}_{N_x \times r} & \overline{U}_{N_x \times (N_t - r)} \end{pmatrix} \begin{pmatrix} \underline{\Sigma}_{r \times r} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \underline{V}_{N_t \times r} & \overline{V}_{N_t \times (N_t - r)} \end{pmatrix}^H$$

$$S = \underline{U}_{N_x \times r} \underline{\Sigma}_{r \times r} \underline{V}^H_{N_t \times r}$$

$$\boxed{S = \sigma_1 \, \boldsymbol{u}_1 \boldsymbol{v}_1^H + \sigma_2 \, \boldsymbol{u}_2 \boldsymbol{v}_2^H + \cdots + \sigma_r \, \boldsymbol{u}_r \boldsymbol{v}_r^H .}$$

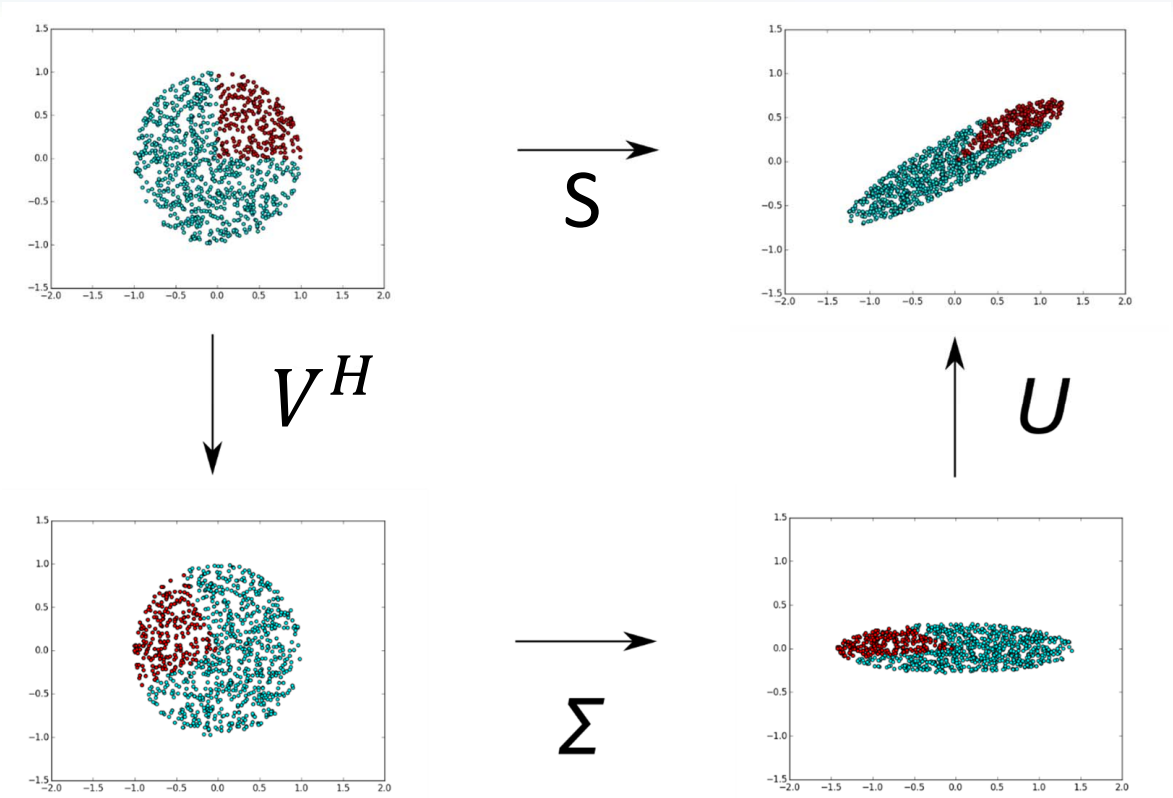
⋆ If we truncate to $k < r$ terms, then

$$S_k = U_k \Sigma_k V_k^H = \sigma_1 \, \boldsymbol{u}_1 \boldsymbol{v}_1^H + \sigma_2 \, \boldsymbol{u}_2 \boldsymbol{v}_2^H + \cdots + \sigma_k \, \boldsymbol{u}_k \boldsymbol{v}_k^H .$$

S_k is an approximation of the matrix S. How good is it?

SVD

Geometric interpretation



- Columns $\boldsymbol{u_i}, i = 1, \cdots, r$ define an orthonormal basis of S
- Columns $\boldsymbol{v_i}, i = 1, \cdots, r$ define an orthonormal basis of S^H
- Singular values σ_i indicate amplification factors

$\Longrightarrow$ SVD: combination of rotations and dilatation.

SVD and eigenvalue problems

Properties

- Classical POD (Lumley, 1967)

$$SS^H = \left(U\Sigma V^H\right)\left(V\Sigma^H U^H\right) = U\Sigma \underbrace{V^H V}_{I_{N_t}} \Sigma^H U^H$$

$$= U\Sigma^2 U^H = U\Lambda U^H$$

$\Longrightarrow \left(SS^H\right) U = U\Sigma^2 = U\Lambda$, *i.e.* columns of U ev's of $SS^H \in \mathbb{C}^{N_x \times N_x}$

- Snapshot POD (Sirovich, 1987)

$$S^H S = \left(V\Sigma^H U^H\right)\left(U\Sigma V^H\right) = V\Sigma^H \underbrace{U^H U}_{I_{N_x}} \Sigma V^H$$

$$= V\Sigma^2 V^H$$

$$= V\Lambda V^H$$

$\Longrightarrow \left(S^H S\right) V = V\Sigma^2 = V\Lambda$, *i.e.* columns of V ev's of $S^H S \in \mathbb{C}^{N_t \times N_t}$

- Singular values

$$\sigma_i = \sqrt{\lambda_i(S^H S)} = \sqrt{\lambda_i(SS^H)} \quad i = 1, \cdots, r$$

Low rank approximation of S

Eckart-Young theorem

$\forall S \in \mathbb{C}^{N_x \times N_t}$, determine $S_k \in \mathbb{C}^{N_x \times N_t}$ such that $\text{rank}(S_k) = k < \text{rank}(S)$.

Criterion:

minimization of the Frobenius norm of the error $S - S_k$.

Theorem: Eckart-Young

$$\boxed{\min_{\text{rank}\,(X) \leq k} \|S - X\|_F = \|S - S_k\|_F = \sqrt{\sum_{i=k+1}^{r} \sigma_i^2(S)}}$$

with $\quad S_k = U \begin{pmatrix} \Sigma_k & 0 \\ 0 & 0 \end{pmatrix} V^H = \sigma_1 \boldsymbol{u}_1 \boldsymbol{v}_1^H + \sigma_2 \boldsymbol{u}_2 \boldsymbol{v}_2^H + \cdots + \sigma_k \boldsymbol{u}_k \boldsymbol{v}_k^H$

and $||S||_F = \sqrt{\sum_{i=1}^{N_x} \sum_{j=1}^{N_t} s_{ij}^2} = \sqrt{\sum_{i=1}^{r} \sigma_i^2}$.

Remark : This theorem establishes a relationship between the rank k of the approximation, and the singular values of S.

Outline

Proper Orthogonal Decomposition

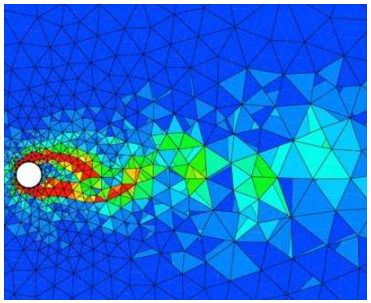
- Also known as:
  - Karhunen-Loève decomposition: Karhunen (1946), Loève (1945) ;
  - Principal Component Analysis: Hotelling (1953) ;
  - Singular Value Decomposition: Golub and Van Loan (1983).
- Applications include:
  - Random variables (Papoulis, 1965) ;
  - Image processing (Rosenfeld and Kak, 1982) ;
  - Signal analysis (Algazi and Sakrison, 1969) ;
  - Data compression (Andrews, Davies and Schwartz, 1967) ;
  - Process identification and control (Gay and Ray, 1986) ;
  - Optimal control (Ravindran, 2000 ; Hinze et Volkwein 2004 ; Bergmann, 2004)

  and of course in fluid mechanics
- Introduced in turbulence by Lumley (1967)

Lumley J.L. (1967) : The structure of inhomogeneous turbulence. *Atmospheric Turbulence and Wave Propagation*, ed. A.M. Yaglom & V.I. Tatarski, pp. 166-178.

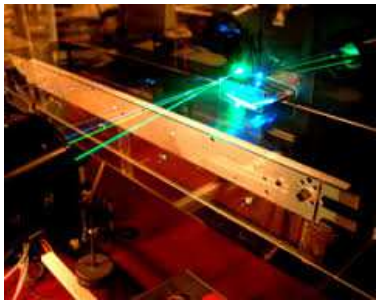
From data to Snapshot Data Matrix

Simulations



- Velocity fields
- Pressure fields
- Vorticity fields
- Tracers

Experiments



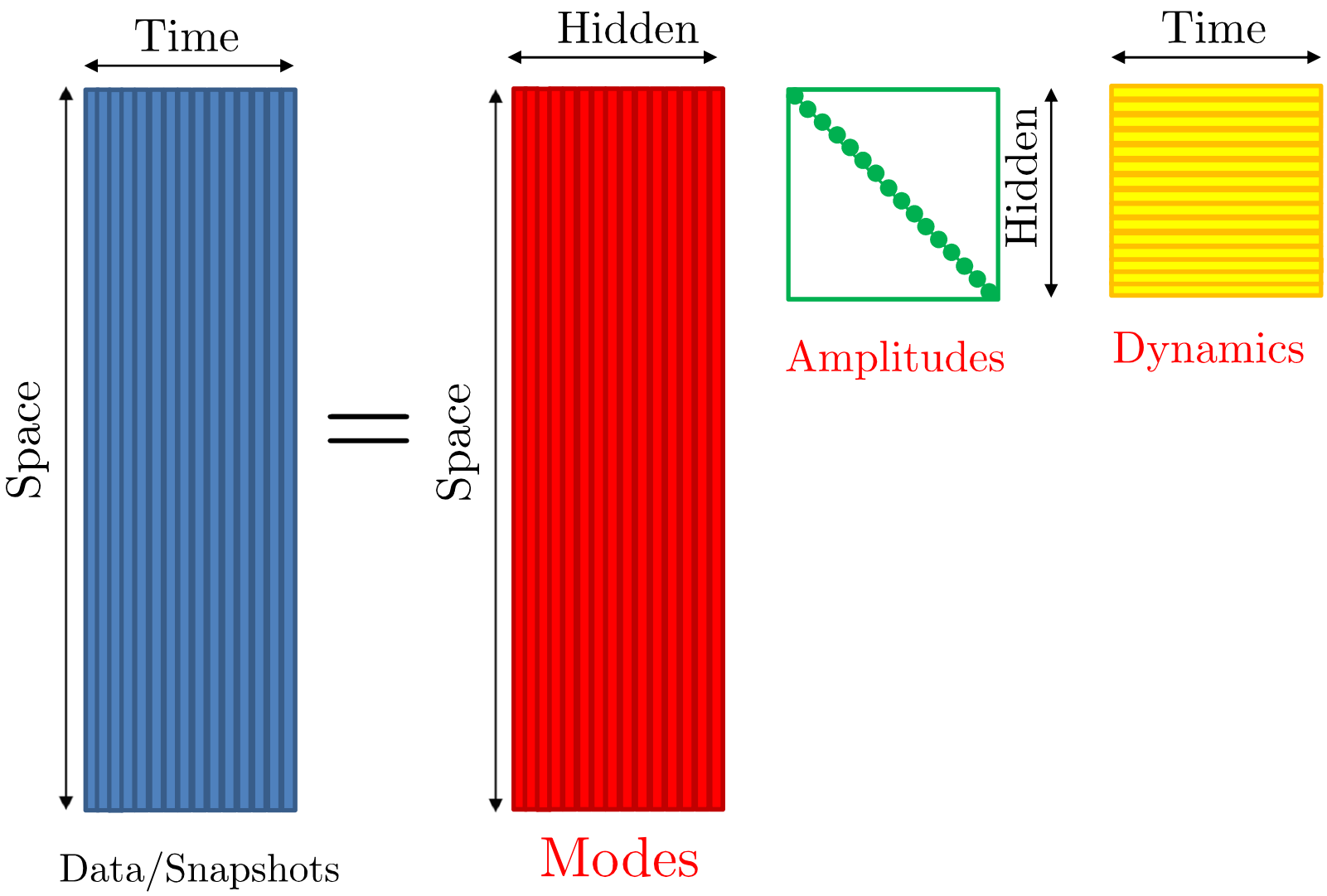
- PIV
- Hot-wires
- LDV
- Visualizations

$$\Rightarrow \quad S = \begin{array}{c} \text{Time} \\ \text{Space} \end{array} \begin{pmatrix} \\ \\ \\ \end{pmatrix}$$

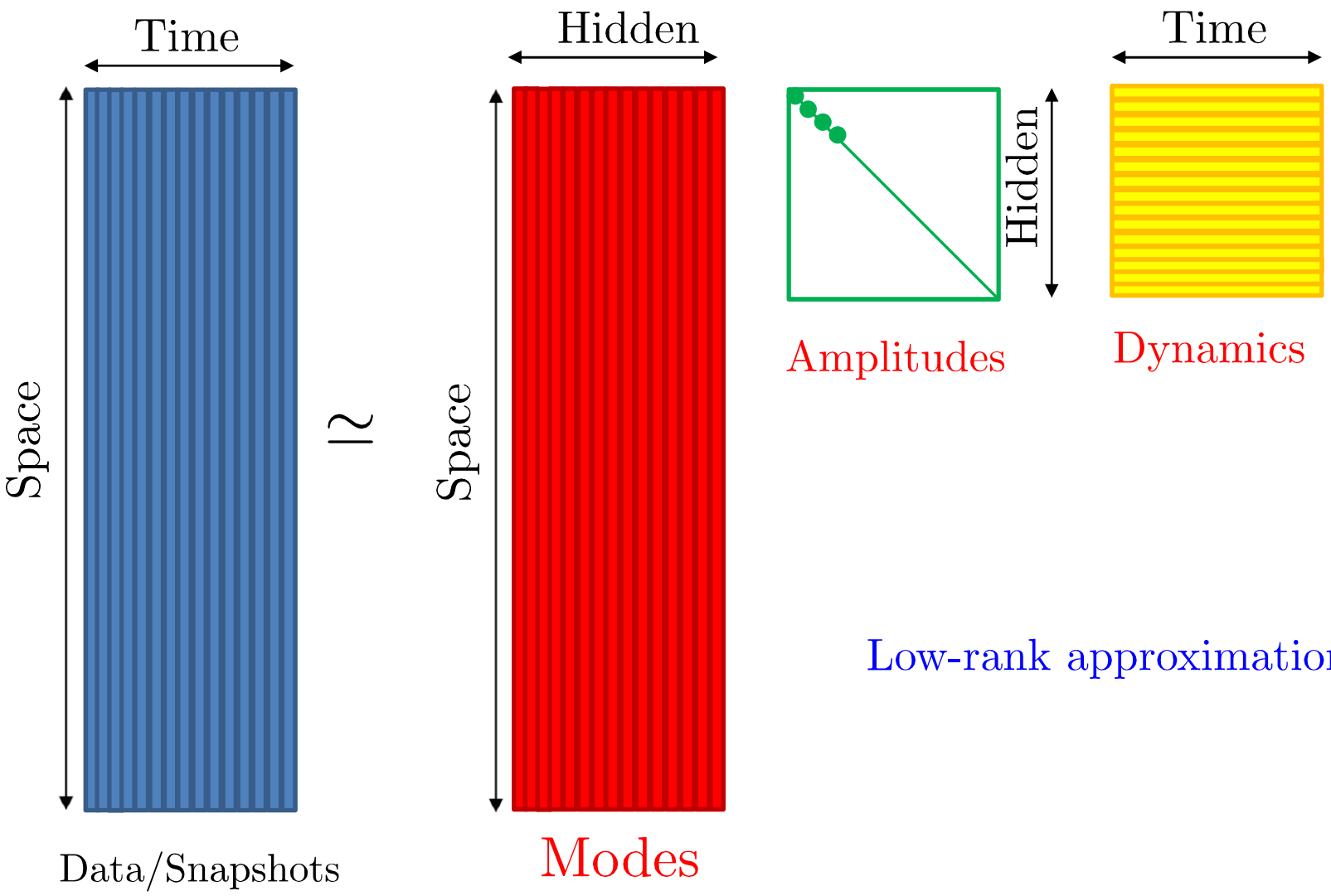
Data/Snapshots

Thanks P. Schmid for the inspiration !

Data analysis as a matrix decomposition



Model reduction: exploit the redundancy



Low-rank approximation

Snapshot Data Matrix

Vectorial case (n_c components)

$$\boldsymbol{u} = (u_1, \cdots, u_{n_c}) \;;\; \boldsymbol{x} = (x_1, \cdots, x_{n_x}) \;;\; \boldsymbol{t} = (t_1, \cdots, t_{N_t}) \;;\; N_x = n_x \times n_c$$

$$S = \left(\begin{array}{c|c|c|c|c}
u_1(x_1, t_1) & u_1(x_1, t_2) & \cdots & u_1(x_1, t_{N_t-1}) & u_1(x_1, t_{N_t}) \\
u_2(x_1, t_1) & u_2(x_1, t_2) & \cdots & u_2(x_1, t_{N_t-1}) & u_2(x_1, t_{N_t}) \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
u_{n_c}(x_1, t_1) & u_{n_c}(x_1, t_2) & \cdots & u_{n_c}(x_1, t_{N_t-1}) & u_{n_c}(x_1, t_{N_t}) \\
\hline
u_1(x_2, t_1) & u_1(x_2, t_2) & \cdots & u_1(x_2, t_{N_t-1}) & u_1(x_2, t_{N_t}) \\
u_2(x_2, t_1) & u_2(x_2, t_2) & \cdots & u_2(x_2, t_{N_t-1}) & u_2(x_2, t_{N_t}) \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
u_{n_c}(x_2, t_1) & u_{n_c}(x_2, t_2) & \cdots & u_{n_c}(x_2, t_{N_t-1}) & u_{n_c}(x_2, t_{N_t}) \\
\hline
\vdots & \vdots & \vdots & \vdots & \vdots \\
\hline
u_1(x_{N_x}, t_1) & u_1(x_{N_x}, t_2) & \cdots & u_1(x_{N_x}, t_{N_t-1}) & u_1(x_{N_x}, t_{N_t}) \\
u_2(x_{N_x}, t_1) & u_2(x_{N_x}, t_2) & \cdots & u_2(x_{N_x}, t_{N_t-1}) & u_2(x_{N_x}, t_{N_t}) \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
u_{n_c}(x_{N_x}, t_1) & u_{n_c}(x_{N_x}, t_2) & \cdots & u_{n_c}(x_{N_x}, t_{N_t-1}) & u_{n_c}(x_{N_x}, t_{N_t})
\end{array}\right) \in \mathbb{R}^{N_x \times N_t}$$

The POD basis problem in $\mathbb{R}^{N_x}$

- Find a k dimensional subspace $V_k^{\mathrm{POD}} = \mathrm{span}\left(\boldsymbol{\Phi}_1, \cdots, \boldsymbol{\Phi}_k\right)$ s.t.

$$\min_{\Pi_{\mathrm{POD}}} \sum_{i=1}^{N_t} \|\boldsymbol{u}(\boldsymbol{x}, t_i) - \Pi_{\mathrm{POD}} \boldsymbol{u}(\boldsymbol{x}, t_i)\|^2_{\mathbb{R}^{N_x}} \qquad s.t. \quad \|\boldsymbol{\Phi}_k\|^2_{\mathbb{R}^{N_x}} = 1$$

or equivalently

$$\max_{\Pi_{\mathrm{POD}}} \sum_{i=1}^{N_t} \|\Pi_{\mathrm{POD}} \boldsymbol{u}(\boldsymbol{x}, t_i)\|^2_{\mathbb{R}^{N_x}} \qquad s.t. \quad \|\boldsymbol{\Phi}_k\|^2_{\mathbb{R}^{N_x}} = 1$$

with Π_{POD}: orthogonal projector on V_k^{POD}, and

$$\Pi_{\mathrm{POD}} \boldsymbol{u}(\boldsymbol{x}, t_i) = \sum_{j=1}^{k} \left(\boldsymbol{u}(\boldsymbol{x}, t_i), \boldsymbol{\Phi}_j(\boldsymbol{x})\right)_{\mathbb{R}^{N_x}} \boldsymbol{\Phi}_j(\boldsymbol{x}).$$

- Solutions:

$$\left(SS^{\mathsf{T}}\right) \boldsymbol{\Phi}_i = \lambda_i \boldsymbol{\Phi}_i, \quad i = 1, \cdots, k, \ \ i.e. \ \boxed{V_k^{\mathrm{POD}} \equiv U_k}$$

The POD basis problem with a weighted inner product

- Weighted inner product: W symmetric, positive definite

$$(\boldsymbol{\psi}_1, \boldsymbol{\psi}_2)_W = \boldsymbol{\psi}_1^{\mathsf{T}} W \boldsymbol{\psi}_2 = \left(W^{1/2}\boldsymbol{\psi}_1, W^{1/2}\boldsymbol{\psi}_2 \right)_{\mathbb{R}^{N_x}}$$

- Find a k dimensional subspace $V_k^{\mathrm{POD}} = \mathrm{span}\left(\boldsymbol{\Phi}_1, \cdots, \boldsymbol{\Phi}_k\right)$ s.t.

$$\max_{\Pi_{\mathrm{POD}}} \sum_{i=1}^{N_t} \|\Pi_{\mathrm{POD}} \boldsymbol{u}(\boldsymbol{x}, t_i)\|_W^2 \qquad s.t. \quad \|\boldsymbol{\Phi}_k\|_W^2 = 1$$

- Solutions:

$$\left(\tilde{S}\tilde{S}^{\mathsf{T}}\right) \tilde{\boldsymbol{\Phi}}_i = \lambda_i \tilde{\boldsymbol{\Phi}}_i, \quad i = 1, \cdots, k$$

with

$$\boxed{\tilde{S} = W^{1/2} S} \qquad \text{and} \qquad \boxed{\tilde{\boldsymbol{\Phi}}_i = W^{1/2} \boldsymbol{\Phi}_i}$$

Dynamic Mode Decomposition (DMD)

The Arnoldi approach

Hypothesis 1

Schmid (2010)

$\exists A \in \mathbb{R}^{N_x \times N_x}$, linear operator, such that

$$\boldsymbol{u}_{k+1} = A\boldsymbol{u}_k, \quad \forall k \in [1, N-1] \quad \Longrightarrow$$

$$U_2^N = \{\boldsymbol{u}_2, \ldots, \boldsymbol{u}_N\} = AU_1^{N-1} = A\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{N-1}\}$$

Objective:

Determine the eigenvectors/values of A, without knowing A.

Hypothesis 2

- $\{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{N-1}\}$ linearly independent.
- $\boldsymbol{u}_N = c_1\boldsymbol{u}_1 + \cdots + c_{N-1}\boldsymbol{u}_{N-1} + \boldsymbol{r}$.

Dynamic Mode Decomposition (DMD)

The Arnoldi approach

Schmid (2010)

Combining Hyp. 1 and Hyp. 2

$$AU_1^{N-1} = U_1^{N-1}C + \boldsymbol{r}\boldsymbol{e}_{N-1}^T$$

Similarity transformation

with C the Companion matrix:

$$C = \begin{pmatrix} 0 & \dots & 0 & c_1 \\ 1 & \dots & 0 & c_2 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 1 & c_{N-1} \end{pmatrix}$$

c_i can be found by pseudo-inverse of U_1^{N-1}.

$$\boldsymbol{u}_N = U_1^{N-1}\boldsymbol{c} \Rightarrow \boxed{\boldsymbol{c} = \left(U_1^{N-1}\right)^+ \boldsymbol{u}_N}$$

Eigen-elements of A

If $C\boldsymbol{y}_i = \lambda_i \boldsymbol{y}_i$ then $A\boldsymbol{\Phi}_i \approx \lambda_i \boldsymbol{\Phi}_i$,

with $\boldsymbol{\Phi}_i = U_1^{N-1}\boldsymbol{y}_i$ defined up to a constant.

Reconstruction using Comp. matrix properties:

$$\boxed{\boldsymbol{u}_k = \sum_{i=1}^{N-1} \boldsymbol{\Phi}_i \lambda_i^{k-1}}$$

Dynamic Mode Decomposition (DMD) The SVD-based approach

- Use of pseudo-inverse

$$U_2^N = AU_1^{N-1} \quad \Longrightarrow \quad A = U_2^N \left(U_1^{N-1}\right)^+$$

- SVD of U_1^{N-1}

$$U_1^{N-1} = U_r \Sigma_r V_r^H \quad \Longrightarrow \quad \left(U_1^{N-1}\right)^+ = V_r \Sigma_r^+ U_r^H$$

- Similarity matrix of A

$$A = U_2^N V_r \Sigma_r^+ U_r^H \quad \Longrightarrow \quad \boxed{U_r^H A U_r = U_r^H U_2^N V_r \Sigma_r^+ = S_r}$$

- Eigen-elements of A (*Tu et al., 2014*)

$$\text{If} \quad S_r \boldsymbol{y}_i = \lambda_i \boldsymbol{y}_i \quad \text{then} \quad A\boldsymbol{\Phi}_i \approx \lambda_i \boldsymbol{\Phi}_i$$

with

$$\boldsymbol{\Phi}_i = \lambda_i^{-1} U_2^N V_r \Sigma_r^+ \boldsymbol{y}_i$$

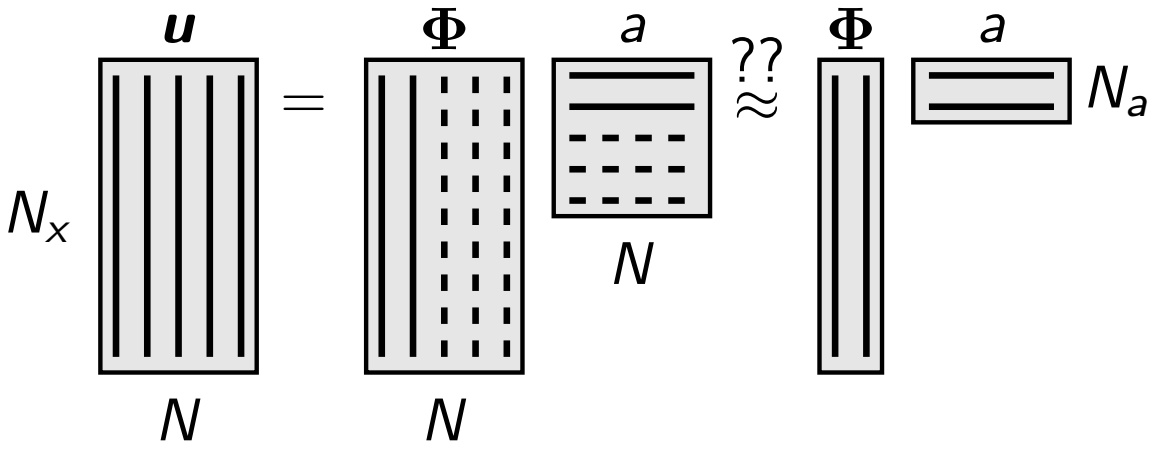
DMD

Modes' selection

How to perform a truncation?

$$\boldsymbol{u}_k = \sum_{i=1}^{N-1} \boldsymbol{\Phi}_i(\boldsymbol{x}) a_i(t_k) \qquad \text{Complete basis.}$$

Modes' selection



- POD / Balanced truncation: Modes sorted by eigenvalues.
- DMD: Choice not obvious!

DMD

Modes' selection

How to perform a truncation in DMD?

$$\boldsymbol{u}_k = \sum_{i=1}^{N-1} \boldsymbol{\Phi}_i \lambda_i^{k-1} \qquad N-1 \text{ modes with linear dynamics behavior.}$$

Modes' selection: Choice depends on the objective.

DMD

Modes' selection

How to perform a truncation in DMD?

$$\boldsymbol{u}_k = \sum_{i=1}^{N-1} \boldsymbol{\Phi}_i \lambda_i^{k-1}$$

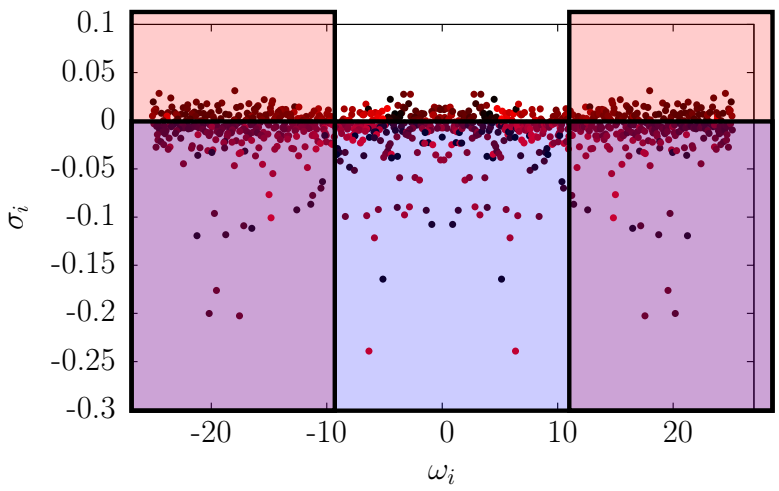
$N-1$ modes with linear dynamics behavior.

Modes' selection: Choice depends on the objective.

Frequency / Growth rate:

$\lambda_i^{k-1} = e^{(\sigma_i + i\omega_i)t_k}$ with

$$\omega_i = \frac{\arg(\lambda_i)}{\Delta t} \; ; \; \sigma_i = \frac{\log(|\lambda_i|)}{\Delta t}$$



DMD

Modes' selection

How to perform a truncation in DMD?

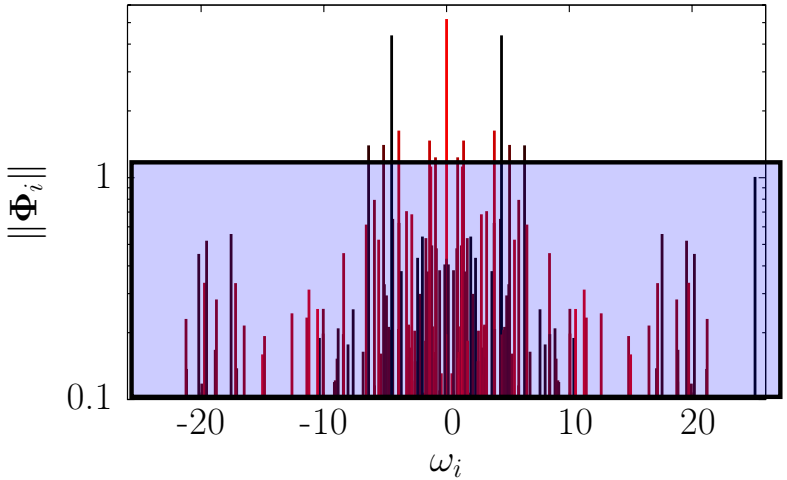
$$\boldsymbol{u}_k = \sum_{i=1}^{N-1} \boldsymbol{\Phi}_i \lambda_i^{k-1}$$

$N-1$ modes with linear dynamics behavior.

Modes' selection: Choice depends on the objective.

Mode amplitude:

$$A_i = \|\boldsymbol{\Phi}_i\|^2$$



DMD

Modes' selection

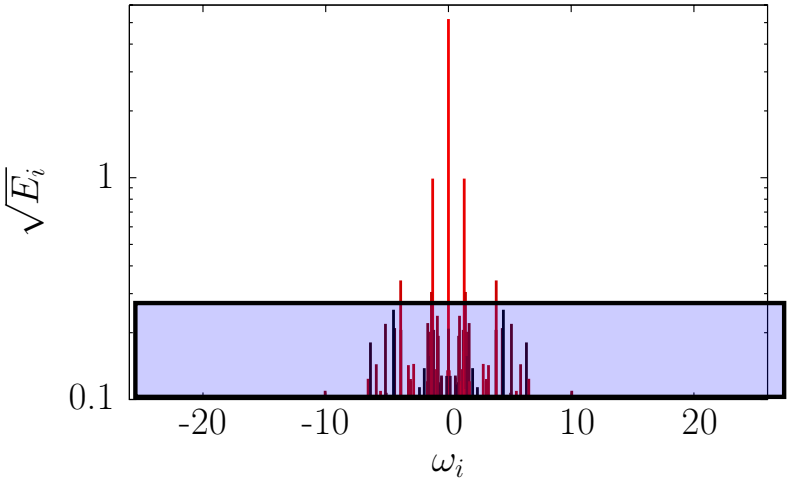
How to perform a truncation in DMD?

$$\boldsymbol{u}_k = \sum_{i=1}^{N-1} \boldsymbol{\Phi}_i \lambda_i^{k-1} \qquad N-1 \text{ modes with linear dynamics behavior.}$$

Modes' selection: Choice depends on the objective.

Energy contribution:

$$E_i = \frac{1}{T}\int_0^T \left\| \boldsymbol{\Phi}_i \lambda_i^{t/\Delta t} \right\|^2 \mathrm{d}t$$
$$= \|\boldsymbol{\Phi}_i\|^2 \frac{e^{2\sigma_i T} - 1}{2\sigma_i T}$$



DMD

Modes' selection

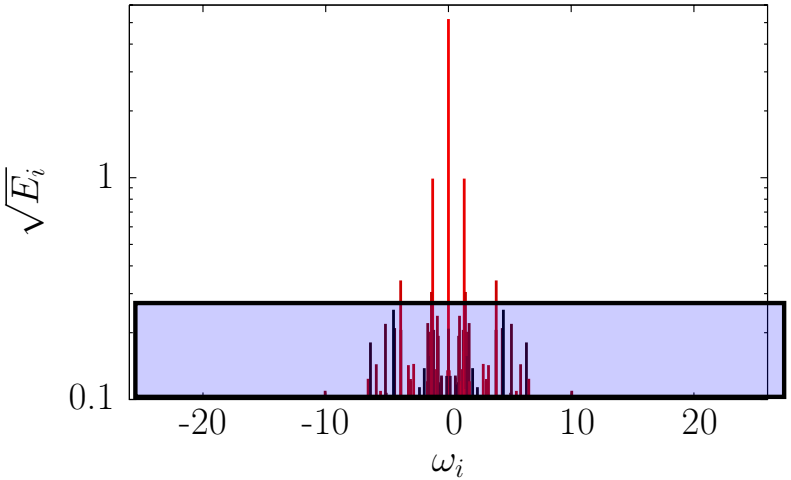
How to perform a truncation in DMD?

$$\boldsymbol{u}_k = \sum_{i=1}^{N-1} \boldsymbol{\Phi}_i \lambda_i^{k-1} \qquad N-1 \text{ modes with linear dynamics behavior.}$$

Modes' selection: Choice depends on the objective.

Energy contribution:

$$E_i = \frac{1}{T}\int_0^T \left\| \boldsymbol{\Phi}_i \lambda_i^{t/\Delta t} \right\|^2 \mathrm{d}t$$
$$= \|\boldsymbol{\Phi}_i\|^2 \frac{e^{2\sigma_i T} - 1}{2\sigma_i T}$$



Non-orthogonality of modes $\Longrightarrow$ Difficulty of modes' selection.

Variants of DMD

Optimized DMD

Optimized DMD: *Chen et al. (2012)*

- $\boldsymbol{u}_k = \sum_{i=1}^{N_a} \hat{\boldsymbol{\Phi}}_i \hat{\lambda}_i^{k-1} + \boldsymbol{r}_k$ with $N_a \ll N-1$
- Find the best $(\hat{\boldsymbol{\Phi}}_i, \hat{\lambda}_i)$ such that $\Gamma = \sum_{k=1}^{N} \|\boldsymbol{r}_k\|^2$ minimal.

Minimize the residual under the linear dynamics constraint
Computationally expensive. $\Rightarrow$ Analytical gradient computation.

- Other variants:
  - Low-rank and sparse DMD (*Jovanović et al., 2012*).
  - Optimal mode decomposition (*Goulart et al., 2012*).
  - Chronos-Koopman analysis (*Cammilleri et al., 2013*).
  - Compressive sampling DMD (*Brunton et al., 2013*).
  - Extended DMD (*Williams et al., 2015*).

DMD

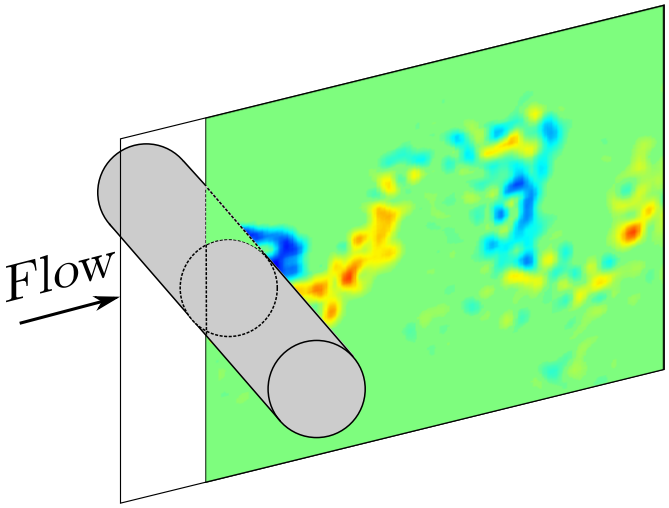
DMD vs. Optimized DMD

N. Bénard

Data: PIV data of a cylinder wake $Re = 13000$.

Classical DMD:

- $N = 1000$.
- 25 periods of vortex shedding.
- $N_a = 7$ modes selected with E_i criterion.



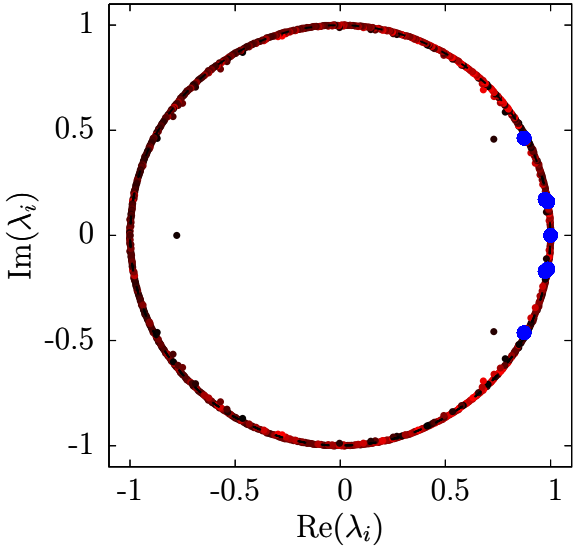
Optimized DMD:

- $N = 256$.
- 6 periods of vortex shedding.
- $N_a = 7$ Optimized DMD modes.

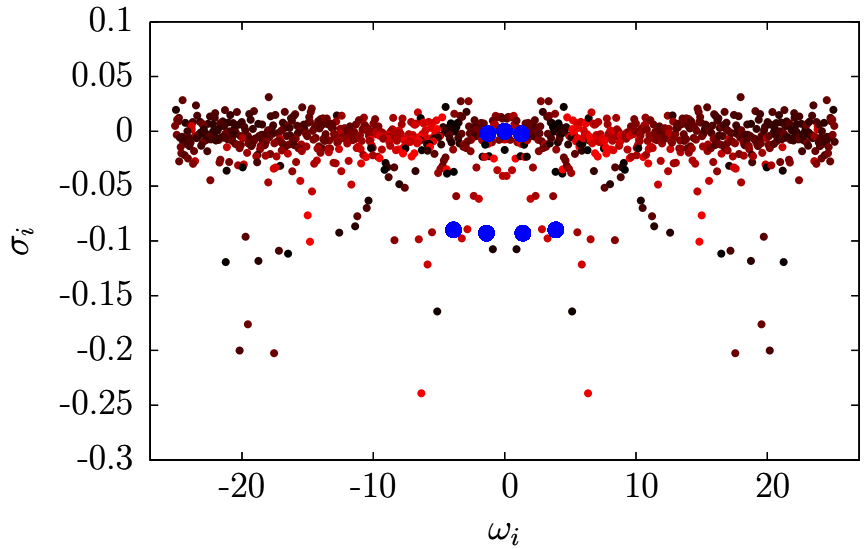
DMD

DMD vs. Optimized DMD

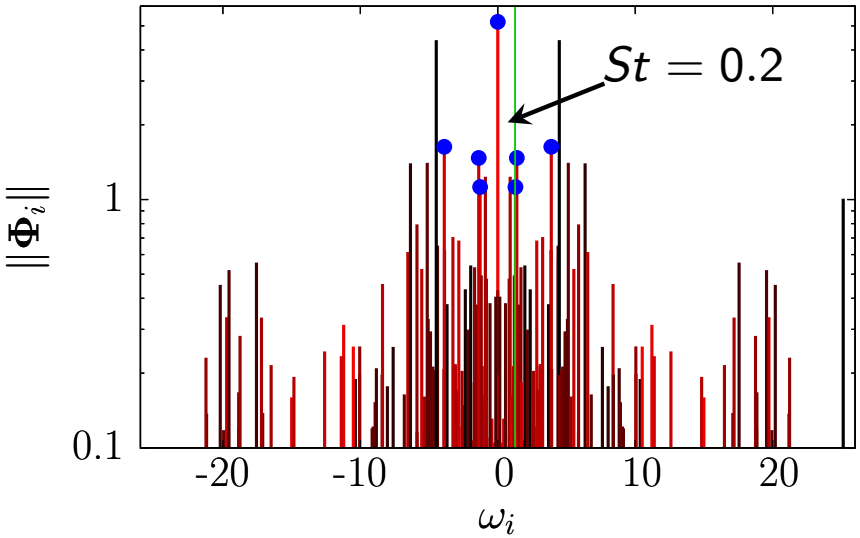
DMD eigenvalues:



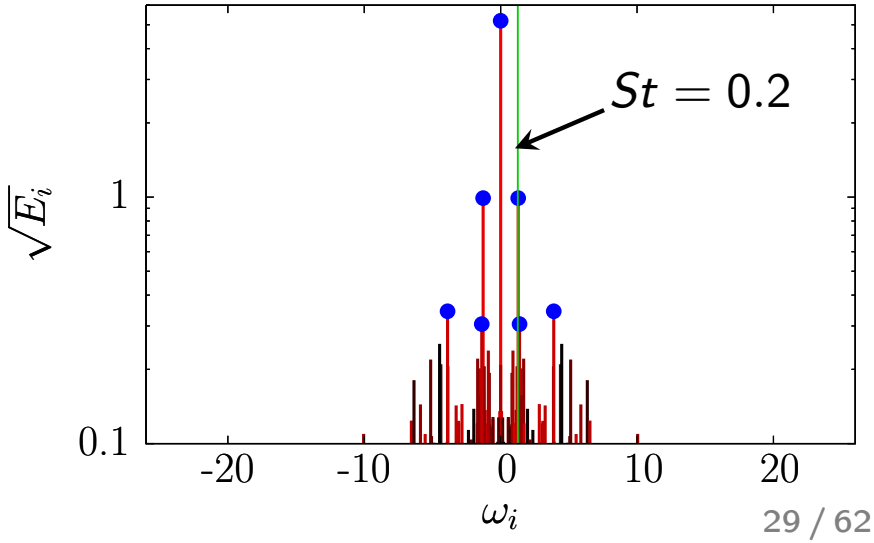
Frequencies/growth rates:



Modes amplitude:



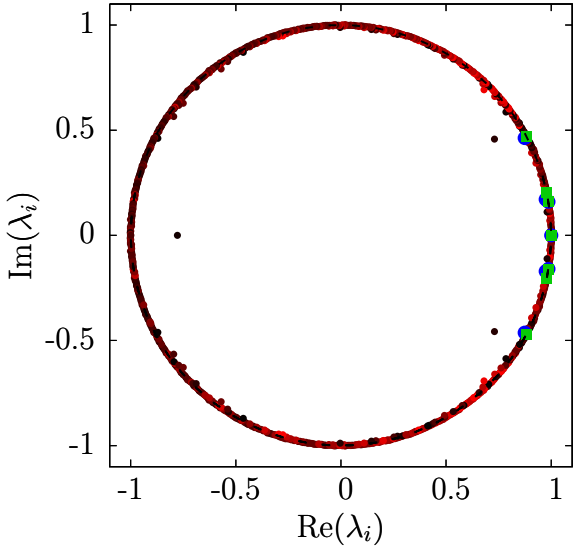
Energy contribution:



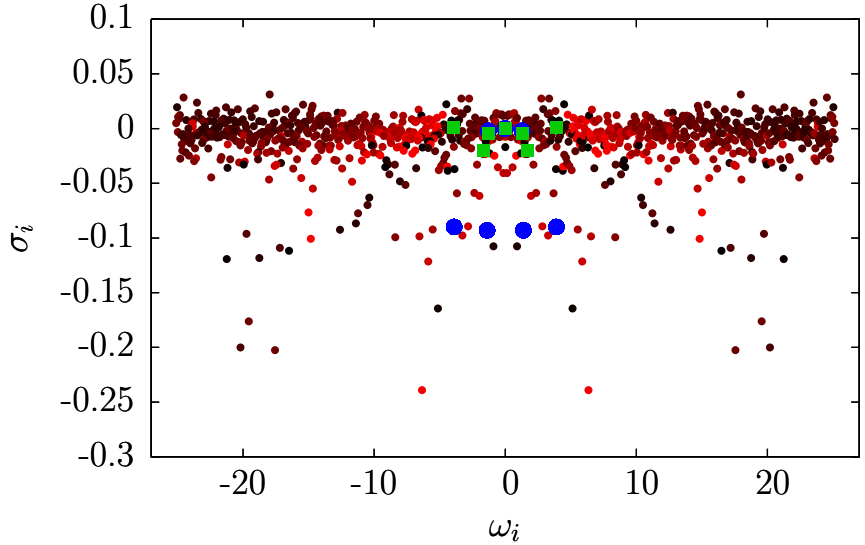
DMD

DMD vs. Optimized DMD

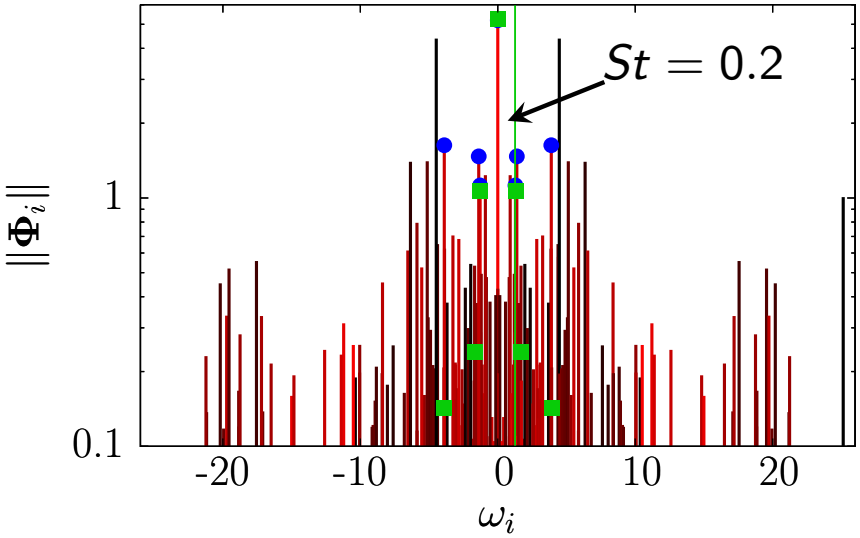
DMD eigenvalues:



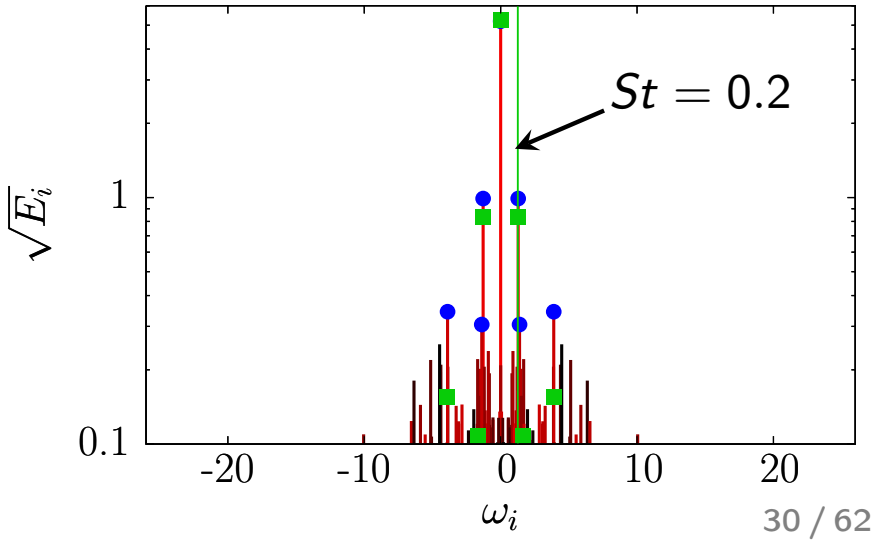
Frequencies/growth rates:



Modes amplitude:



Energy contribution:



DMD

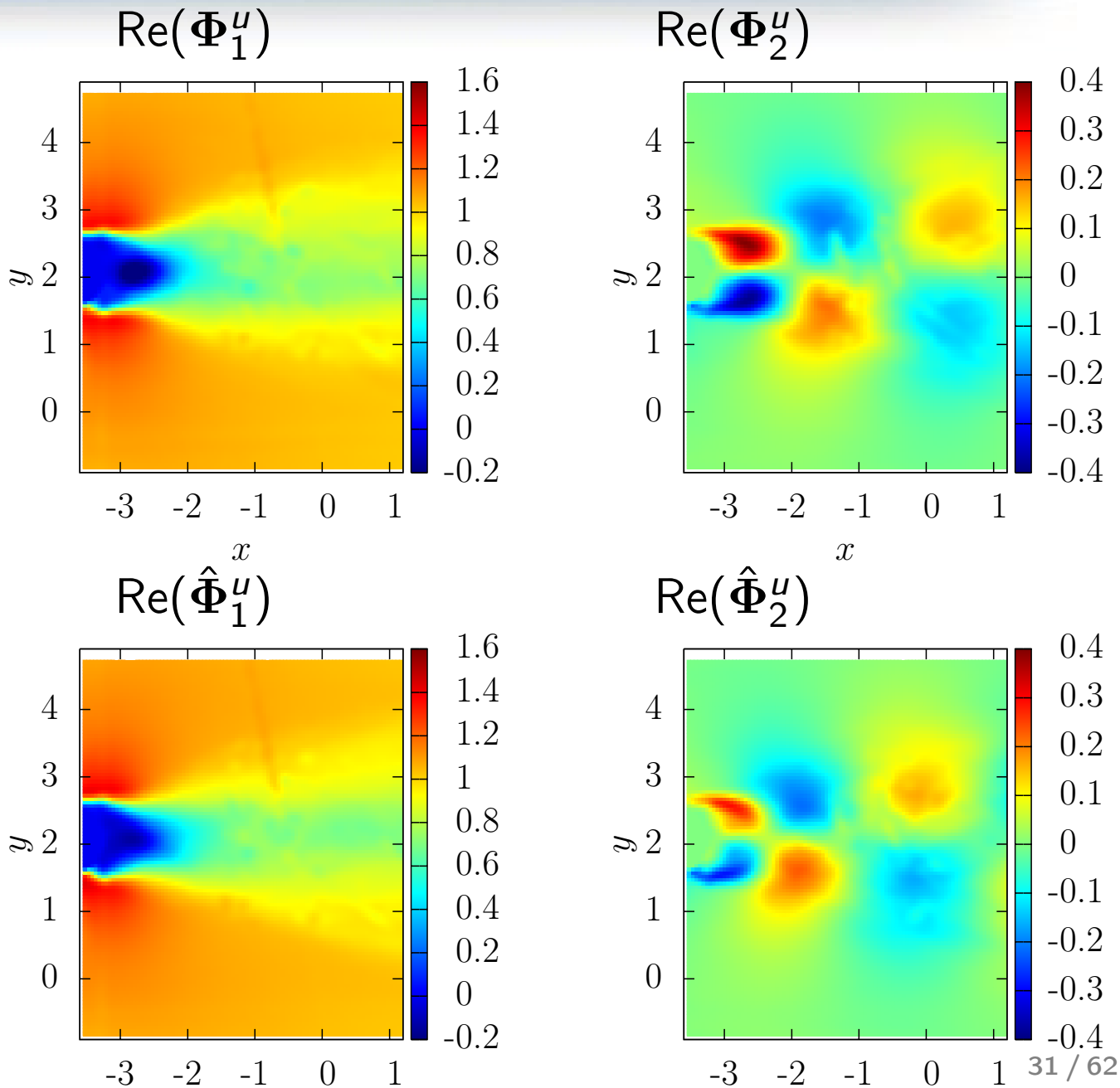
DMD vs. Optimized DMD

Classical DMD

Modes with higher energy contribution.

Optimized DMD

Selected DMD modes as initial condition.



DMD

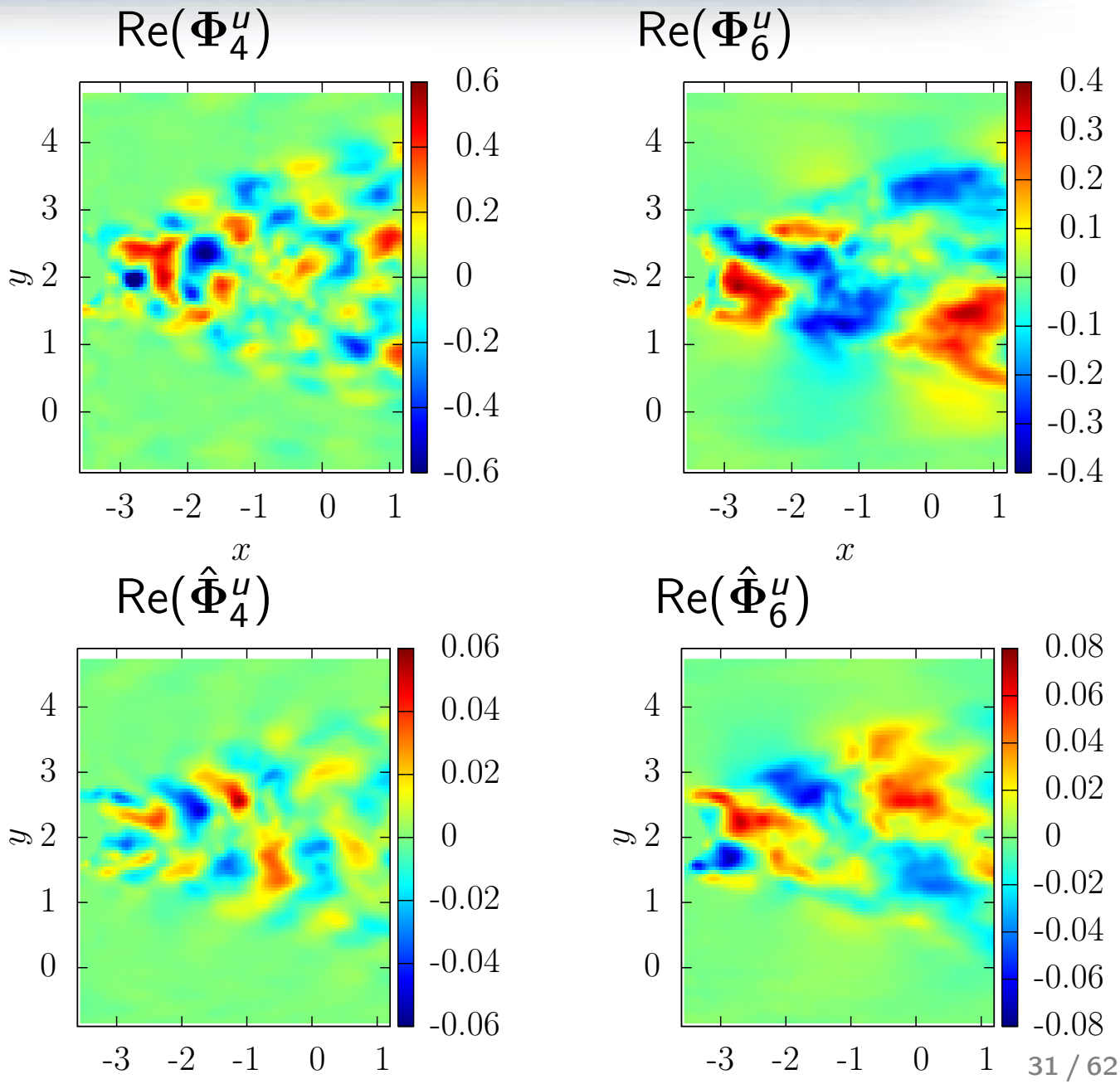
DMD vs. Optimized DMD

Classical DMD

Modes with higher energy contribution.

Optimized DMD

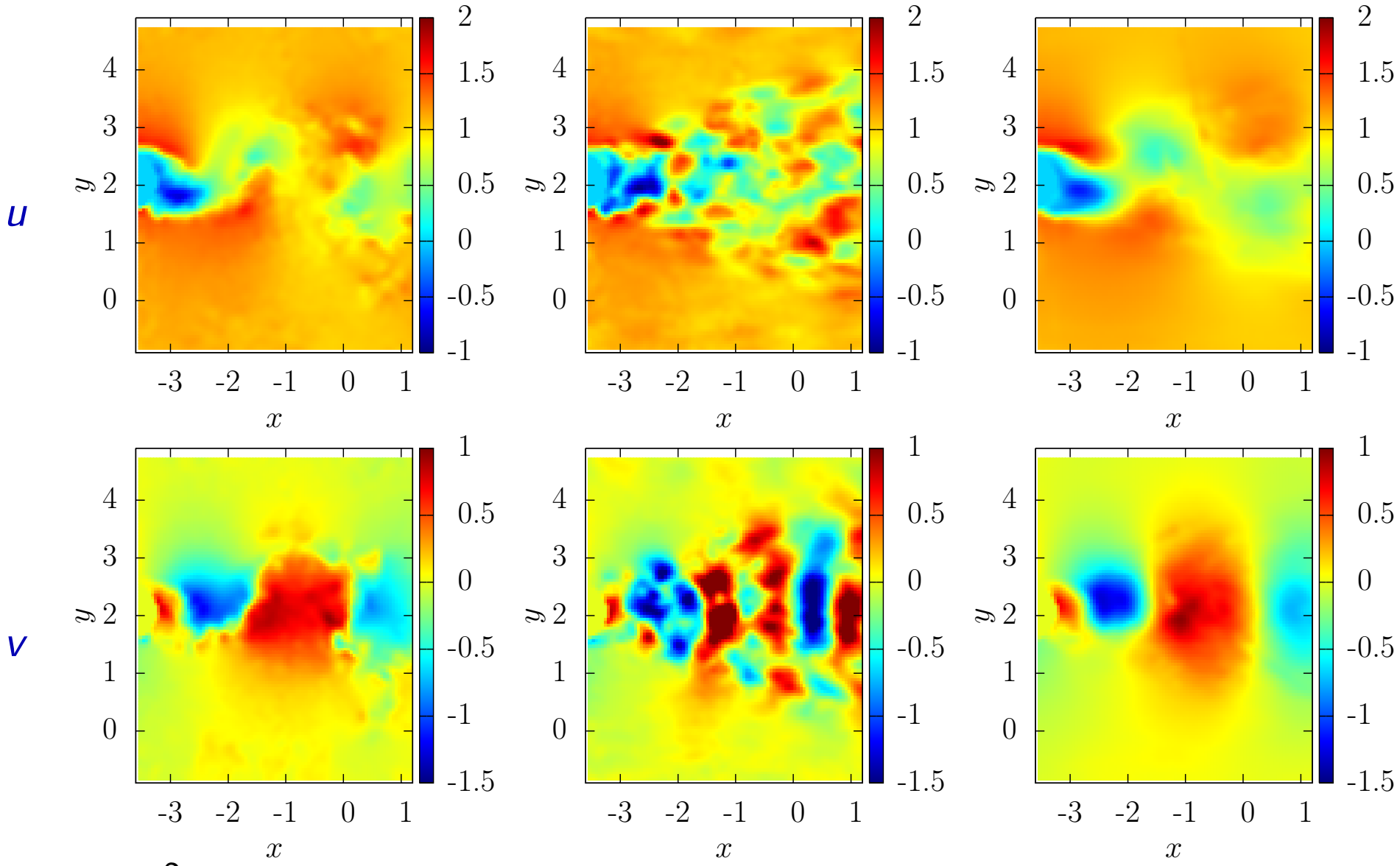
Selected DMD modes as initial condition.



DMD vs. Optimized DMD

5th snapshot reconstruction

Original snapshots: Classical DMD: Optimized DMD:



L^2-norm error: 45.6% 15.6%

k-means algorithm

Input : $\{\boldsymbol{v}^m\}$, set of snapshots
Input : K, number of clusters
Output: $\boldsymbol{c}_1, \cdots, \boldsymbol{c}_K$, centroids

0. Initialize K means $\boldsymbol{c}_1^{(0)}, \cdots, \boldsymbol{c}_K^{(0)}$ (random, kmeans++);

for $l \leftarrow 0$ to L do

1. Assignment step;
   Assign each snapshot to the nearest cluster;

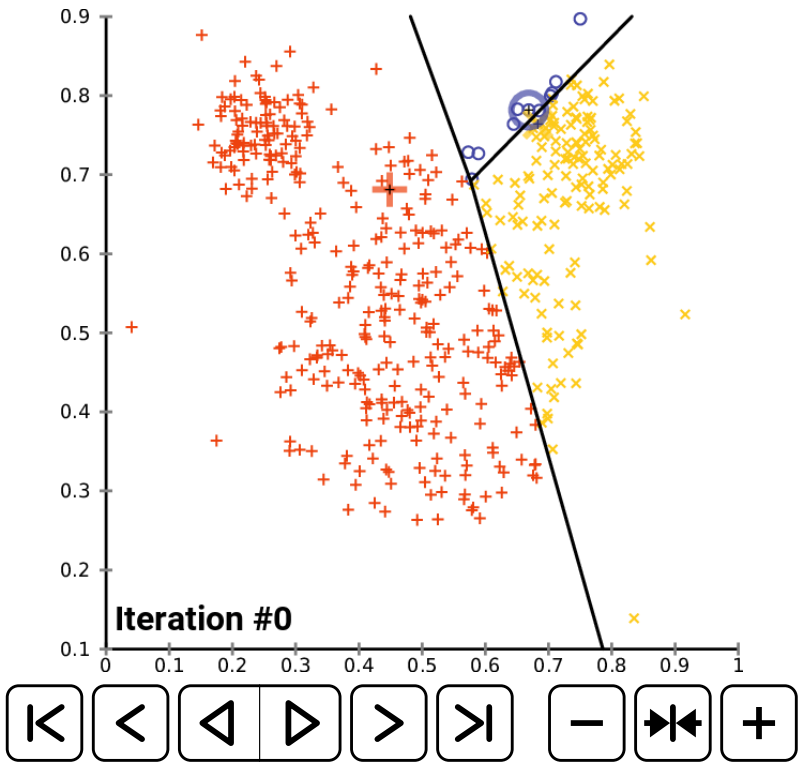
$$\mathcal{C}_k^{(l)} = \{\boldsymbol{v}^m : \|\boldsymbol{v}^m - \boldsymbol{c}_k^{(l)}\|^2 \leq \|\boldsymbol{v}^m - \boldsymbol{c}_j^{(l)}\|^2 \quad \forall j \in [1:K]\}$$

2. Update step;
   Compute new means (centroids);

$$\boldsymbol{c}_k^{(l+1)} = \frac{1}{|\mathcal{C}_k^{(l)}|} \sum_{\boldsymbol{v}^m \in \mathcal{C}_k^{(l)}} \boldsymbol{v}^m$$

3. Test convergence;

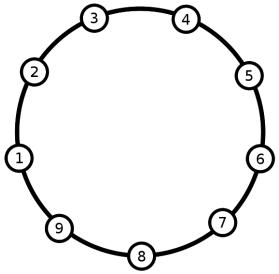
end

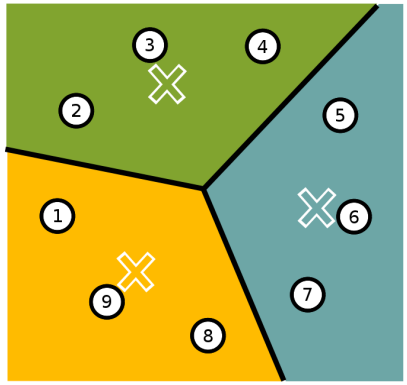


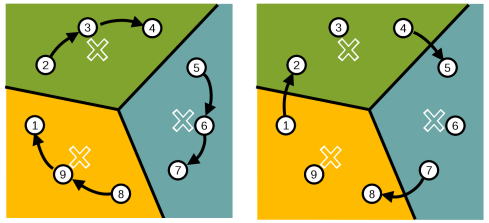
Cluster-based Reduced-Order Modelling

	Kinematics	Dynamics
1 **Data**	**2** **Discretised state space** (cluster analysis)	**3** Construct **cluster transition matrix** → Markov model

Discrete snapshots of a limit cycle

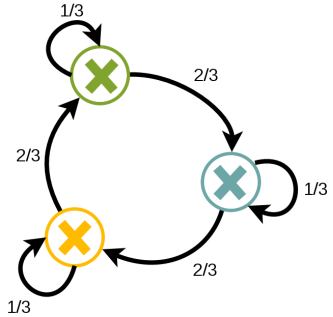






$$\boldsymbol{p}^l = P^l\,\boldsymbol{p}^0$$

P_{jk}	k=1 1→j	k=2 2→j	k=3 3→j
j=1	2/3	0	1/3
j=2	1/3	2/3	0
j=3	0	1/3	2/3



Time resolved velocity snapshots

$$\boldsymbol{v}^m(\boldsymbol{x}) := \boldsymbol{v}(\boldsymbol{x}, t_m)$$

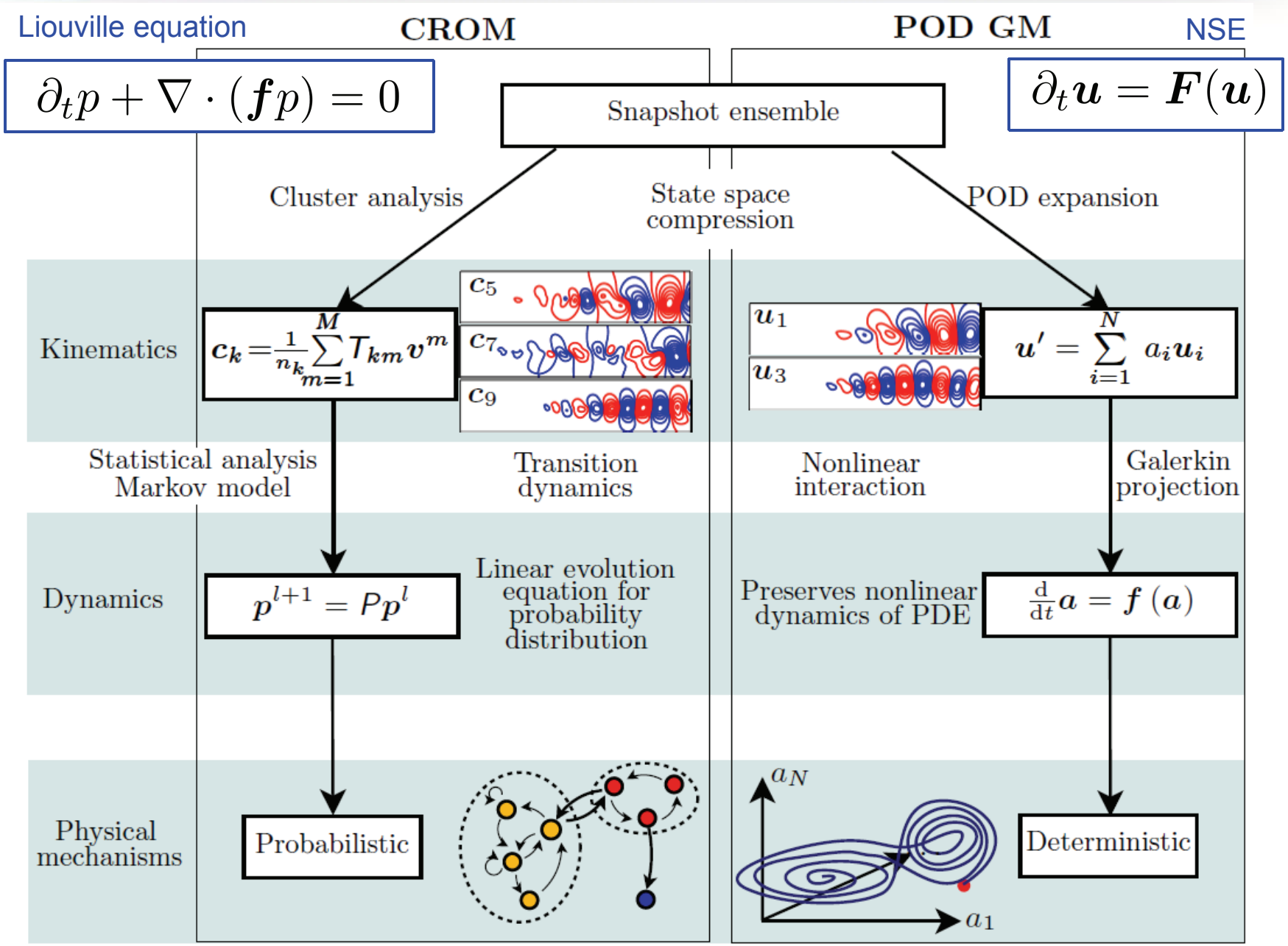
Cluster analysis (k-means)
(Steinhaus 1956, MacQueen 1967)

$$J = \sum_{k=1}^{K} \sum_{\boldsymbol{v}^m \in \mathcal{C}_k} \|\boldsymbol{v}^m - \boldsymbol{c}_k\|_\Omega^2$$

Discrete-time Markov model

$$\boldsymbol{p}^l = P^l\,\boldsymbol{p}^0 \quad \text{with} \quad P_{jk} = \frac{n_{jk}}{n_k}$$

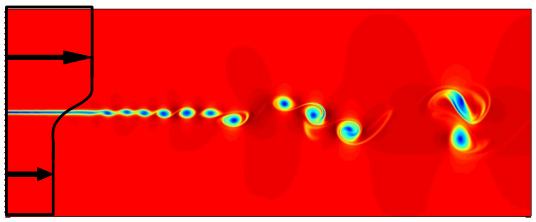
Comparison CROM vs. POD GM



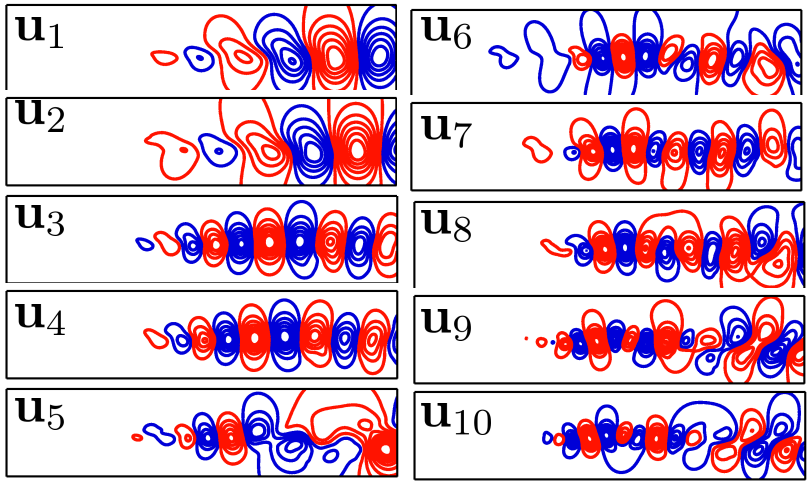
CROM

Mixing layer

1. Data
   - 2D incompressible
   - $Re = 500$
   - $M = 2000$ snapshots



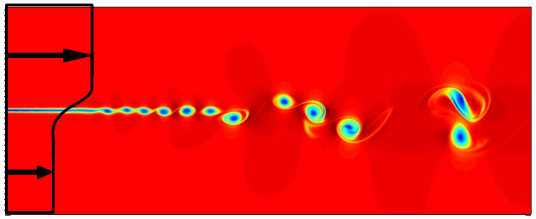
Snapshot POD modes



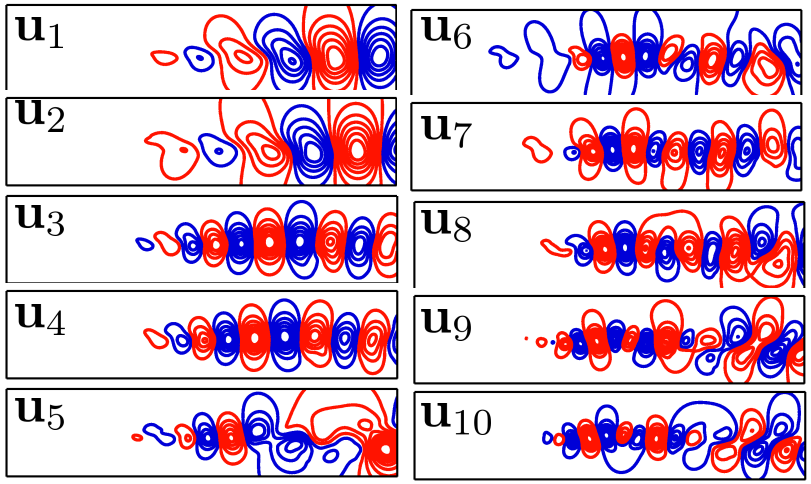
CROM

Mixing layer

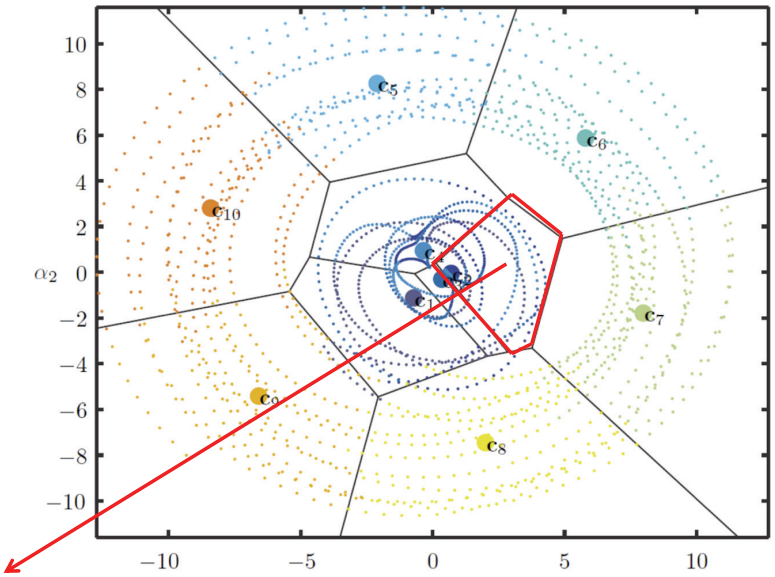
1. Data
   - 2D incompressible
   - $Re = 500$
   - $M = 2000$ snapshots



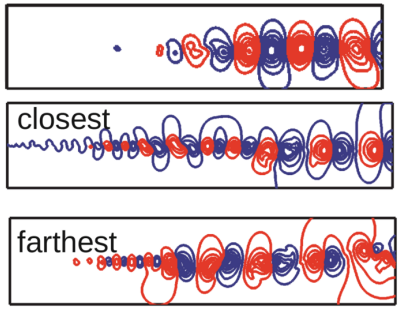
Snapshot POD modes



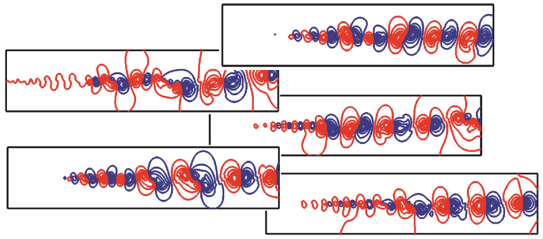
2. Cluster analysis ($K = 10$)



Cluster centroid c2:



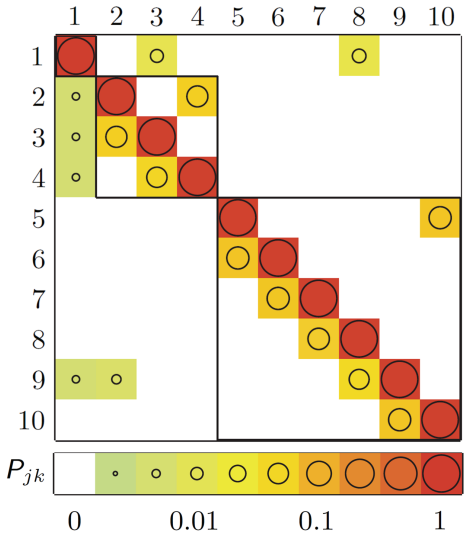
Some snapshots in this cluster:

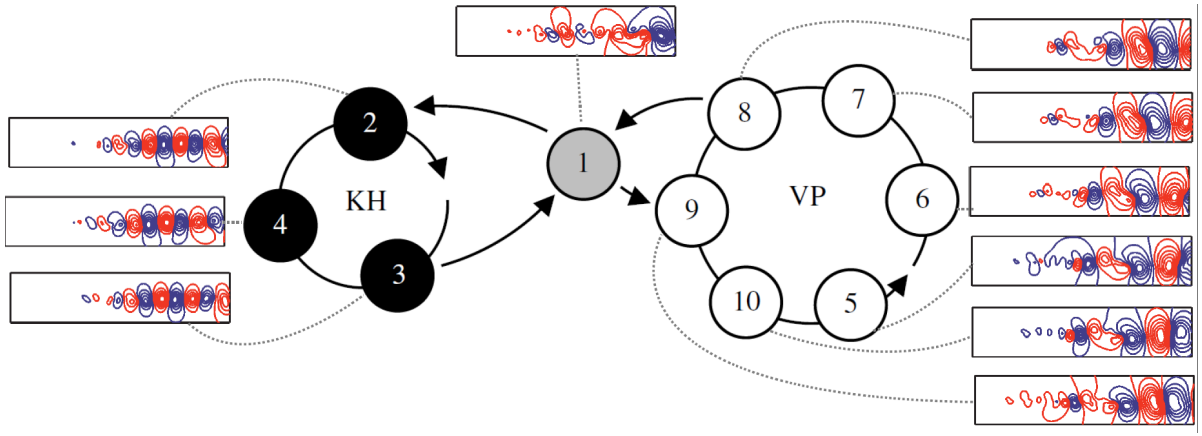


CROM

Mixing layer

3. Cluster transition matrix and simplified cluster transitions





- Identification of two shedding regimes:
  KH: Kelvin Helmoltz and VP: Vortex pairing
- Flipper cluster c_1 acts as a switch between both regimes

Outline

Global modes

- Flow dynamics:

$$\dot{\boldsymbol{q}} = \boldsymbol{f}(\boldsymbol{q}). \tag{1}$$

- Hypothesis: Steady base flow $\boldsymbol{Q}$

$$\boldsymbol{q}(x, y, z, t) = \boldsymbol{Q}(x, y, z) + \epsilon \boldsymbol{q}'(x, y, z, t) \quad \text{with} \quad \epsilon \ll 1 \tag{2}$$

- Substitute (1) into (2), expand in Taylor series, at order 1

$$\dot{\boldsymbol{q}}' = A\boldsymbol{q}' \quad \text{with } A \text{ Jacobian matrix of } \boldsymbol{f} \text{ at } \boldsymbol{q}$$

- Different levels of expansion for $\boldsymbol{q}(x, y, z, t)$

$$\boldsymbol{Q}(x, y, z) + \epsilon \left\{ \hat{\boldsymbol{q}}(x, y, z) \exp\left[-\jmath \Omega t\right] + \text{c.c.} \right\} \qquad \text{3D global modes}$$

$$\boldsymbol{Q}(x, y) + \epsilon \left\{ \hat{\boldsymbol{q}}(x, y) \exp\left[\jmath \left(\beta z - \Omega t\right)\right] + \text{c.c.} \right\} \qquad \text{2D global modes}$$

$$\boldsymbol{Q}(y) + \epsilon \left\{ \hat{\boldsymbol{q}}(y) \exp\left[\jmath \left(\alpha x + \beta z - \Omega t\right)\right] + \text{c.c.} \right\} \qquad \text{Local stability}$$

- 3D global modes leads to generalized eigenvalue problem

$$\boxed{-\jmath \Omega \hat{\boldsymbol{q}} = A \hat{\boldsymbol{q}}}$$

2D global modes

Incompressible Navier-Stokes (1)

- Incompressible Navier-Stokes $\boldsymbol{u} = (u, v, w)$

$$\partial_t \boldsymbol{u} + (\boldsymbol{u} \cdot \boldsymbol{\nabla})\, \boldsymbol{u} = -\boldsymbol{\nabla} p + \frac{1}{Re} \boldsymbol{\Delta} \boldsymbol{u}$$
$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0,$$

- Base flow equations $\boldsymbol{Q}(x, y) = (\boldsymbol{U}, P) = (U, V, 0, P)$

$$(\boldsymbol{U} \cdot \boldsymbol{\nabla})\, \boldsymbol{U} = -\boldsymbol{\nabla} P + \frac{1}{Re} \boldsymbol{\Delta} \boldsymbol{U}$$
$$\boldsymbol{\nabla} \cdot \boldsymbol{U} = 0.$$

- Perturbation equations $\boldsymbol{q'}(x, y, z, t) = (u', v', w', p')$

$$\partial_t \boldsymbol{u'} + (\boldsymbol{u'} \cdot \boldsymbol{\nabla})\, \boldsymbol{U} + (\boldsymbol{U} \cdot \boldsymbol{\nabla})\, \boldsymbol{u'} = -\boldsymbol{\nabla} p' + \frac{1}{Re} \boldsymbol{\Delta} \boldsymbol{u'}$$
$$\boldsymbol{\nabla} \cdot \boldsymbol{u'} = 0.$$

- Hypothesis: Base flow homogeneous in the transverse direction

$$\boldsymbol{q'}(x, y, z, t) = \frac{1}{2} \left\{ (\hat{u}, \hat{v}, \hat{w}, \hat{p})\, (x, y) \exp\left[\jmath \beta z + \sigma t\right] + \text{c.c.} \right\} \text{ with } \sigma \in \mathbb{C}$$

2D global modes

Incompressible Navier-Stokes (2)

$$\boxed{A\,\hat{\boldsymbol{q}} = \sigma\, B\,\hat{\boldsymbol{q}}} \quad \text{with} \quad \hat{\boldsymbol{q}} = (\hat{\boldsymbol{u}}, p) = (\hat{u}, \hat{v}, \jmath\hat{w}, \hat{p}) \quad \text{global mode.}$$

$$A = \begin{pmatrix} \mathcal{D} - \mathcal{C} - \partial_x U & -\partial_y U & 0 & -\partial_x \\ -\partial_x V & \mathcal{D} - \mathcal{C} - \partial_y V & 0 & -\partial_y \\ 0 & 0 & \mathcal{D} - \mathcal{C} & \beta \\ \partial_x & \partial_y & \beta & 0 \end{pmatrix}$$

and

$$B = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

where

$$\mathcal{D} = \frac{1}{Re}\left(\partial_{x^2} + \partial_{y^2} - \beta^2\right) \qquad \text{viscous diffusion of perturbation}$$

$$\mathcal{C} = U\partial_x + V\partial_y \qquad \text{advection by base flow}$$

Koopman operator

(*Koopman, 1931*)

- Nonlinear dynamical system $\boldsymbol{f} : \mathcal{M} \longrightarrow \mathcal{M}$ ($\mathcal{M}$ finite dimensional)

$$\boldsymbol{\mathcal{X}}_{k+1} = \boldsymbol{f}(\boldsymbol{\mathcal{X}}_k)$$

- Let $g : \mathcal{M} \to \mathbb{R}$ be a scalar observable. $\mathcal{K}_{\boldsymbol{f}}$ Koopman operator

$$\mathcal{K}_{\boldsymbol{f}} g(\boldsymbol{\mathcal{X}}_k) := g(\boldsymbol{f}(\boldsymbol{\mathcal{X}}_k)) = g \circ \boldsymbol{f}(\boldsymbol{\mathcal{X}}_k) = g(\boldsymbol{\mathcal{X}}_{k+1}).$$

- $\mathcal{K}_{\boldsymbol{f}}$: linear operator of infinite dimension

$$\mathcal{K}_{\boldsymbol{f}}(\alpha_1 g_1(\boldsymbol{\mathcal{X}}_k) + \alpha_2 g_2(\boldsymbol{\mathcal{X}}_k)) = \alpha_1 \mathcal{K}_{\boldsymbol{f}} g_1(\boldsymbol{\mathcal{X}}_k) + \alpha_2 \mathcal{K}_{\boldsymbol{f}} g_2(\boldsymbol{\mathcal{X}}_k)$$

- Eigenfunctions and eigenvalues

$$\mathcal{K}_{\boldsymbol{f}}\, \phi^{(j)}(\boldsymbol{\mathcal{X}}_k) = \lambda^{(j)} \phi^{(j)}(\boldsymbol{\mathcal{X}}_k)$$

- Let define $\boxed{z^{(j)} = \phi^{(j)}(\boldsymbol{\mathcal{X}})}$ nonlinear change of coordinates. We have:

$$z_{k+1}^{(j)} = \phi^{(j)}(\boldsymbol{\mathcal{X}}_{k+1}) = \phi^{(j)}(\boldsymbol{f}(\boldsymbol{\mathcal{X}}_k)) = \mathcal{K}_{\boldsymbol{f}} \phi^{(j)}(\boldsymbol{\mathcal{X}}_k) = \lambda^{(j)} \phi^{(j)}(\boldsymbol{\mathcal{X}}_k) = \lambda^{(j)} z_k^{(j)}$$

Dynamics linear in $z^{(j)}$; $\mathcal{K}_{\boldsymbol{f}}$ may have enough eigenfunctions !!!

Koopman operator

Connection with DMD

- Let $\boldsymbol{g} : \mathcal{M} \to \mathbb{R}^p$ be a vectorial observable. We have:

$$\boldsymbol{g}(\boldsymbol{\mathcal{X}}_k) = \sum_{j=1}^{+\infty} \phi_j(\boldsymbol{\mathcal{X}}_k)\boldsymbol{k}_j \quad \text{with} \quad \boldsymbol{k}_j : \text{Koopman modes}$$

- We can show that:

$$\boldsymbol{g}(\boldsymbol{\mathcal{X}}_k) = \sum_{j=1}^{+\infty} \phi_j(\boldsymbol{\mathcal{X}}_k)\boldsymbol{k}_j = \sum_{j=1}^{+\infty} \mathcal{K}_{\boldsymbol{f}}^{k-1}\phi_j(\boldsymbol{\mathcal{X}}_1)\boldsymbol{k}_j = \sum_{j=1}^{+\infty} \lambda_j^{k-1}\phi_j(\boldsymbol{\mathcal{X}}_1)\boldsymbol{k}_j$$

$\Longrightarrow$ Koopman modes can be obtained by DMD algorithm.

Reduced-Order Modelling

Dynamical systems $\mathcal{S}$ and $\widehat{\mathcal{S}}$

- Full-order model (FOM)

$$\mathcal{S}: \begin{cases} \dot{\boldsymbol{\mathcal{X}}}(t) = \boldsymbol{f}\left(\boldsymbol{\mathcal{X}}(t), \boldsymbol{c}(t)\right), & \text{where} \quad \boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_{\mathcal{X}}} \\ \boldsymbol{\mathcal{Y}}(t) = \boldsymbol{g}\left(\boldsymbol{\mathcal{X}}(t), \boldsymbol{c}(t)\right), & \text{where} \quad \boldsymbol{\mathcal{Y}} \in \mathbb{R}^{n_{\mathcal{Y}}}. \end{cases}$$

- Reduced-order model (ROM)

$$\widehat{\mathcal{S}}: \begin{cases} \dot{\widehat{\boldsymbol{\mathcal{X}}}}(t) = \widehat{\boldsymbol{f}}\left(\widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)\right), & \text{where} \quad \widehat{\boldsymbol{\mathcal{X}}} \in \mathbb{R}^{n_k} \quad \text{with} \quad \boxed{n_k \ll n_{\mathcal{X}}} \\ \widehat{\boldsymbol{\mathcal{Y}}}(t) = \widehat{\boldsymbol{g}}\left(\widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)\right), & \text{where} \quad \widehat{\boldsymbol{\mathcal{Y}}} \in \mathbb{R}^{n_{\mathcal{Y}}}. \end{cases}$$

- Requirements for deriving $\widehat{\mathcal{S}}$
  1. low approximation error $\forall \boldsymbol{c}$ *i.e.*

$$\|\boldsymbol{\mathcal{Y}} - \widehat{\boldsymbol{\mathcal{Y}}}\| < \epsilon \times \|\boldsymbol{c}\| \qquad \text{with } \epsilon \text{ a tolerance}$$

$$\Longrightarrow \text{Need computable error bound estimates!!}$$

  2. stability and passivity (no generation of energy) preserved ;
  3. procedure of model reduction numerically stable and efficient ;
  4. if possible, automatic generation of models.

Reduced-Order Modelling

Projection method (Petrov-Galerkin)

- We introduce W_1 and W_2, two biorthogonal matrices of size $\mathbb{R}^{n_{\mathcal{X}} \times n_k}$, such that $\boxed{W_2^H Q W_1 = I_{n_k}}$ where $Q \in \mathbb{R}^{n_{\mathcal{X}} \times n_{\mathcal{X}}}$ is the weight matrix.
- We consider: i) the projection $\boldsymbol{\mathcal{X}} = W_1 \widehat{\boldsymbol{\mathcal{X}}}$ and ii) $\widehat{\boldsymbol{\mathcal{Y}}} \simeq \boldsymbol{\mathcal{Y}}$.
- Algorithm:
  1. $\boldsymbol{\mathcal{X}} \simeq W_1 \widehat{\boldsymbol{\mathcal{X}}}$

$$\boldsymbol{\mathcal{R}} = W_1 \dot{\widehat{\boldsymbol{\mathcal{X}}}}(t) - \boldsymbol{f}\left(W_1 \widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)\right),$$
$$\widehat{\boldsymbol{\mathcal{Y}}}(t) = \boldsymbol{g}\left(W_1 \widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)\right).$$

  2. Petrov-Galerkin projection: $W_2^H Q \boldsymbol{\mathcal{R}} = 0_{n_k}$ *i.e.*

$$\widehat{\mathcal{S}} : \begin{cases} \dot{\widehat{\boldsymbol{\mathcal{X}}}}(t) = \widehat{\boldsymbol{f}}(\widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)) = W_2^H Q\, \boldsymbol{f}(W_1 \widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)), \\ \widehat{\boldsymbol{\mathcal{Y}}}(t) = \widehat{\boldsymbol{g}}(\widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)) = \boldsymbol{g}(W_1 \widehat{\boldsymbol{\mathcal{X}}}(t), \boldsymbol{c}(t)), \end{cases}$$

For $W_1 \neq W_2$: oblique projection.
For $W_1 \equiv W_2$: Galerkin projection (orthogonal projection).

Reduced-Order Modelling

Projection method: choice of W_1 and W_2

▷ For linear systems, various projection methods exist:

1. Krylov methods (Gugercin et Antoulas, 2006)
   proj. on the Krylov subspace of the controllability gramian: identification of the moments of the transfer function.
2. Balanced realizations
   proj. on dominant modes of the controllability and observability gramians
   - Balanced Truncation (Moore, 1981) ; Balanced POD (Rowley, 2005)
3. Instability methods
   proj. on global modes and adjoint global modes (Sipp, 2008)

▷ For non-linear systems: *a posteriori* methods

1. Proper Orthogonal Decomposition or POD (Lumley 1967 ; Sirovich 1987)
   proj. on the subspace determined with snapshots of the system.
2. Dynamic Mode Decomposition (Schmid, 2010)

POD Reduced-order model

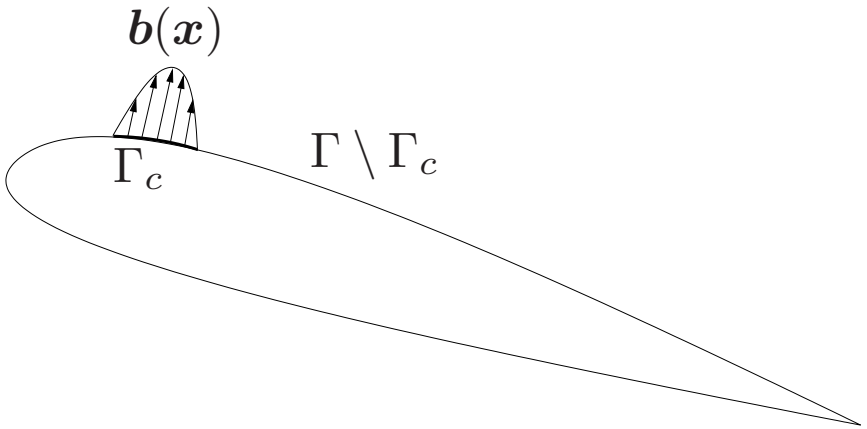
Generalities

▷ Boundary control of the Navier-Stokes equations $(\boldsymbol{x} \in \Omega \text{ and } t \geq 0)$

$$\begin{cases} \dfrac{\partial \boldsymbol{u}}{\partial t} = \boldsymbol{f}(\boldsymbol{u}, P) \\ \boldsymbol{u}(\boldsymbol{x},\ t=0) = \boldsymbol{u}_0(\boldsymbol{x}) \quad (I.C.) \\ \boldsymbol{u}(\boldsymbol{x},\ t) = \gamma(t)\boldsymbol{b}(\boldsymbol{x}) \quad \text{for } \boldsymbol{x} \in \Gamma_c, \quad (B.C.) \\ \boldsymbol{u}(\boldsymbol{x},\ t) = \boldsymbol{h}(\boldsymbol{x}) \quad \text{for } \boldsymbol{x} \in \Gamma \setminus \Gamma_c \quad (B.C.). \end{cases}$$

where

$$\boldsymbol{f}(\boldsymbol{u}, P) = -(\boldsymbol{u} \cdot \boldsymbol{\nabla})\,\boldsymbol{u} - \boldsymbol{\nabla} p + \frac{1}{\mathrm{Re}}\Delta \boldsymbol{u}.$$



POD Reduced-order model

Choice of the decomposition variable

▷ B.C. independent of time, *i.e.* $\boldsymbol{u}(\boldsymbol{x},\, t) = \boldsymbol{u}_{\mathrm{BC}}(\boldsymbol{x})$ on Γ

- $\mathcal{U} = \{\boldsymbol{u}(\boldsymbol{x},\, t_1),\, \cdots,\, \boldsymbol{u}(\boldsymbol{x},\, t_{N_t})\}$
- $\boldsymbol{u_m}(\boldsymbol{x})$: ensemble average of $\mathcal{U}$ (time average)

$$\boldsymbol{u_m}(\boldsymbol{x}) = \frac{1}{N_t}\sum_{k=1}^{N_t} \boldsymbol{u}(\boldsymbol{x}, t_k)$$

- $\mathcal{U}' = \{\boldsymbol{u}(\boldsymbol{x},\, t_1) - \boldsymbol{u_m}(\boldsymbol{x}),\, \cdots,\, \boldsymbol{u}(\boldsymbol{x},\, t_{N_t}) - \boldsymbol{u_m}(\boldsymbol{x})\}$
- $\boldsymbol{u}(\boldsymbol{x},\, t) - \boldsymbol{u_m}(\boldsymbol{x})$ is solenoidal
- $\boldsymbol{u}_{\mathrm{POD}}(\boldsymbol{x},\, t) = \boldsymbol{u}(\boldsymbol{x},\, t) - \boldsymbol{u_m}(\boldsymbol{x})$ verify homogeneous B.C. i.e.

$$\boxed{\boldsymbol{\Phi}_i(\boldsymbol{x})|_{\boldsymbol{x}\in\Gamma} = 0}.$$

- $\boldsymbol{u}(\boldsymbol{x},\, t) = \boldsymbol{u_m}(\boldsymbol{x}) + \sum_{i=1}^{N_{\mathrm{POD}}} a_i(t)\boldsymbol{\Phi}_i(\boldsymbol{x})$.

POD Reduced-order model

Choice of the decomposition variable

▷ B.C. dependent of time, *i.e.* $\boldsymbol{u}(\boldsymbol{x},\, t) = \boldsymbol{u}_{\mathrm{BC}}(\boldsymbol{x}, t)$ on Γ

- $\mathcal{U} = \{\boldsymbol{u}(\boldsymbol{x},\, t_1),\, \cdots,\, \boldsymbol{u}(\boldsymbol{x},\, t_{N_t})\}$
- $\boldsymbol{u_m}(\boldsymbol{x})$: ensemble average of $\mathcal{U}$ (time average)
- $\mathcal{U}' = \{\boldsymbol{u}(\boldsymbol{x},\, t_1) - \gamma(t_1)\boldsymbol{u_c}(\boldsymbol{x}) - \boldsymbol{u_m}(\boldsymbol{x}),\, \cdots,\, \boldsymbol{u}(\boldsymbol{x},\, t_{N_t}) - \gamma(t_{N_t})\boldsymbol{u_c}(\boldsymbol{x}) - \boldsymbol{u_m}(\boldsymbol{x})\}$
- $\boxed{\boldsymbol{u}(\boldsymbol{x},\, t) = \boldsymbol{u_m}(\boldsymbol{x}) + \gamma(t)\boldsymbol{u_c}(\boldsymbol{x}) + \sum_{i=1}^{N_{\mathrm{POD}}} a_i(t)\boldsymbol{\Phi}_i(\boldsymbol{x})}$ where

$$\begin{aligned} \boldsymbol{u_c}(\boldsymbol{x}) &= \boldsymbol{b}(\boldsymbol{x}) \quad && \text{on } \Gamma_c \text{ and} \\ \boldsymbol{u_c}(\boldsymbol{x}) &= 0 && \text{on } \Gamma \setminus \Gamma_c. \end{aligned}$$

- $\boldsymbol{u}_{\mathrm{POD}}(\boldsymbol{x},\, t) = \boldsymbol{u}(\boldsymbol{x},\, t) - \boldsymbol{u_m}(\boldsymbol{x}) - \gamma(t)\boldsymbol{u_c}(\boldsymbol{x})$ verify homogeneous B.C. i.e.

$$\boxed{\boldsymbol{\Phi}_i(\boldsymbol{x})|_{\boldsymbol{x}\in\Gamma} = 0}.$$

POD Reduced-order model

Galerkin projection (1)

- Galerkin Projection of the Navier-Stokes equations onto the POD basis:

$$\left(\boldsymbol{\Phi}_i,\, \frac{\partial \boldsymbol{u}}{\partial t} - \boldsymbol{f}(\boldsymbol{u}, P)\right)_\Omega = \left(\boldsymbol{\Phi}_i,\, \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\boldsymbol{\nabla})\,\boldsymbol{u} + \boldsymbol{\nabla} p - \frac{1}{\mathrm{Re}}\Delta \boldsymbol{u}\right)_\Omega = 0 \quad \forall i$$

$$\Longrightarrow \left(\boldsymbol{\Phi}_i,\, \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u}\right)_\Omega = \left(\boldsymbol{\Phi}_i,\, -\boldsymbol{\nabla} p + \frac{1}{\mathrm{Re}}\Delta \boldsymbol{u}\right)_\Omega .$$

- Integration by parts (Green formula):

$$\left(\boldsymbol{\Phi}_i,\, \frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u}\right)_\Omega = (p,\, \boldsymbol{\nabla}\cdot\boldsymbol{\Phi}_i)_\Omega - \frac{1}{\mathrm{Re}}\left((\boldsymbol{\nabla}\otimes\boldsymbol{\Phi}_i)^T,\, \boldsymbol{\nabla}\otimes\boldsymbol{u}\right)_\Omega$$
$$- [p\,\boldsymbol{\Phi}_i]_\Gamma + \frac{1}{\mathrm{Re}}[(\boldsymbol{\nabla}\otimes\boldsymbol{u})\boldsymbol{\Phi}_i]_\Gamma .$$

with $[\boldsymbol{a}]_\Gamma = \int_\Gamma \boldsymbol{a}\cdot\boldsymbol{n}\,\mathrm{d}\boldsymbol{x}$ and $(\overline{\overline{A}},\, \overline{\overline{B}})_\Omega = \int_\Omega \overline{\overline{A}} : \overline{\overline{B}}\, d\Omega = \sum_{i,j} \int_\Omega A_{ij} B_{ji}\, \mathrm{d}\boldsymbol{x}.$

POD Reduced-order model

Galerkin projection (2)

- We decompose the velocity fields on N_{POD} modes:

$$\boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{u_m}(\boldsymbol{x}) + \gamma(t)\,\boldsymbol{u_c}(\boldsymbol{x}) + \sum_{k=1}^{N_{\mathrm{POD}}} a_k(t)\boldsymbol{\Phi}_k(\boldsymbol{x}).$$

- Dynamical system with N_{gal} ($\ll N_{\mathrm{POD}}$) modes kept:

$$\begin{aligned}\frac{d\,a_i(t)}{d\,t} =& \mathcal{A}_i + \sum_{j=1}^{N_{\mathrm{gal}}} \mathcal{B}_{ij}\,a_j(t) + \sum_{j=1}^{N_{\mathrm{gal}}}\sum_{k=1}^{N_{\mathrm{gal}}} \mathcal{C}_{ijk}\,a_j(t)a_k(t)\\ &+ \mathcal{D}_i\,\frac{d\,\gamma}{d\,t} + \left(\mathcal{E}_i + \sum_{j=1}^{N_{\mathrm{gal}}} \mathcal{F}_{ij}\,a_j(t)\right)\gamma + \mathcal{G}_i\gamma^2\end{aligned}$$

$$a_i(0) = (\boldsymbol{u}(\boldsymbol{x},0) - \boldsymbol{u_m}(\boldsymbol{x}) - \gamma(0)\,\boldsymbol{u_c}(\boldsymbol{x}),\ \boldsymbol{\Phi}_i(\boldsymbol{x}))_\Omega.$$

$\mathcal{A}_i$, $\mathcal{B}_{ij}$, $\mathcal{C}_{ijk}$, $\mathcal{D}_i$, $\mathcal{E}_i$, $\mathcal{F}_{ij}$ et $\mathcal{G}_i$ depend only on $\boldsymbol{\Phi}$, $\boldsymbol{u_m}$, $\boldsymbol{u_c}$ and Re.

- Dynamics predicted by the POD ROM may be not sufficiently accurate
  $\Longrightarrow$ need of identification techniques (Data Assimilation)

POD Reduced-order model

Coefficients for $\gamma = 0$

$$\mathcal{A}_i = -\left(\boldsymbol{\Phi}_i, \left(\boldsymbol{u_m} \cdot \boldsymbol{\nabla}\right) \boldsymbol{u_m}\right)_\Omega - \frac{1}{\mathrm{Re}} \left(\boldsymbol{\nabla} \boldsymbol{\Phi}_i, \boldsymbol{\nabla} \boldsymbol{u_m}\right)_\Omega + \frac{1}{\mathrm{Re}} \left[\boldsymbol{\Phi}_i \, \boldsymbol{\nabla} \boldsymbol{u_m}\right]_\Gamma$$

$$\begin{aligned}\mathcal{B}_{ij} = &-\left(\boldsymbol{\Phi}_i, \left(\boldsymbol{u_m} \cdot \boldsymbol{\nabla}\right) \boldsymbol{\Phi}_j\right)_\Omega - \left(\boldsymbol{\Phi}_i, \left(\boldsymbol{\Phi}_j \cdot \boldsymbol{\nabla}\right) \boldsymbol{u_m}\right)_\Omega \\ &-\frac{1}{\mathrm{Re}} \left(\boldsymbol{\nabla} \boldsymbol{\Phi}_i, \boldsymbol{\nabla} \boldsymbol{\Phi}_j\right)_\Omega + \frac{1}{\mathrm{Re}} \left[\boldsymbol{\Phi}_i \, \boldsymbol{\nabla} \boldsymbol{\Phi}_j\right]_\Gamma\end{aligned}$$

$$\mathcal{C}_{ijk} = -\left(\boldsymbol{\Phi}_i, \left(\boldsymbol{\Phi}_j \cdot \boldsymbol{\nabla}\right) \boldsymbol{\Phi}_k\right)_\Omega$$

POD Reduced-order model

Coefficients for $\gamma \neq 0$

$$\mathcal{D}_i = -\left(\boldsymbol{\Phi}_i, \boldsymbol{u_c}\right)_\Omega$$

$$\mathcal{E}_i = -\left(\boldsymbol{\Phi}_i, \left(\boldsymbol{u_m} \cdot \boldsymbol{\nabla}\right) \boldsymbol{u_c}\right)_\Omega - \left(\boldsymbol{\Phi}_i, \left(\boldsymbol{u_c} \cdot \boldsymbol{\nabla}\right) \boldsymbol{u_m}\right)_\Omega - \frac{1}{\mathrm{Re}} \left(\boldsymbol{\nabla} \boldsymbol{\Phi}_i, \boldsymbol{\nabla} \boldsymbol{u_c}\right)_\Omega + \frac{1}{\mathrm{Re}} \left[\boldsymbol{\Phi}_i \, \boldsymbol{\nabla} \boldsymbol{u_c}\right]_\Gamma$$

$$\mathcal{F}_{ij} = -\left(\boldsymbol{\Phi}_i, \left(\boldsymbol{\Phi}_j \cdot \boldsymbol{\nabla}\right) \boldsymbol{u_c}\right)_\Omega - \left(\boldsymbol{\Phi}_i, \left(\boldsymbol{u_c} \cdot \boldsymbol{\nabla}\right) \boldsymbol{\Phi}_j\right)_\Omega$$

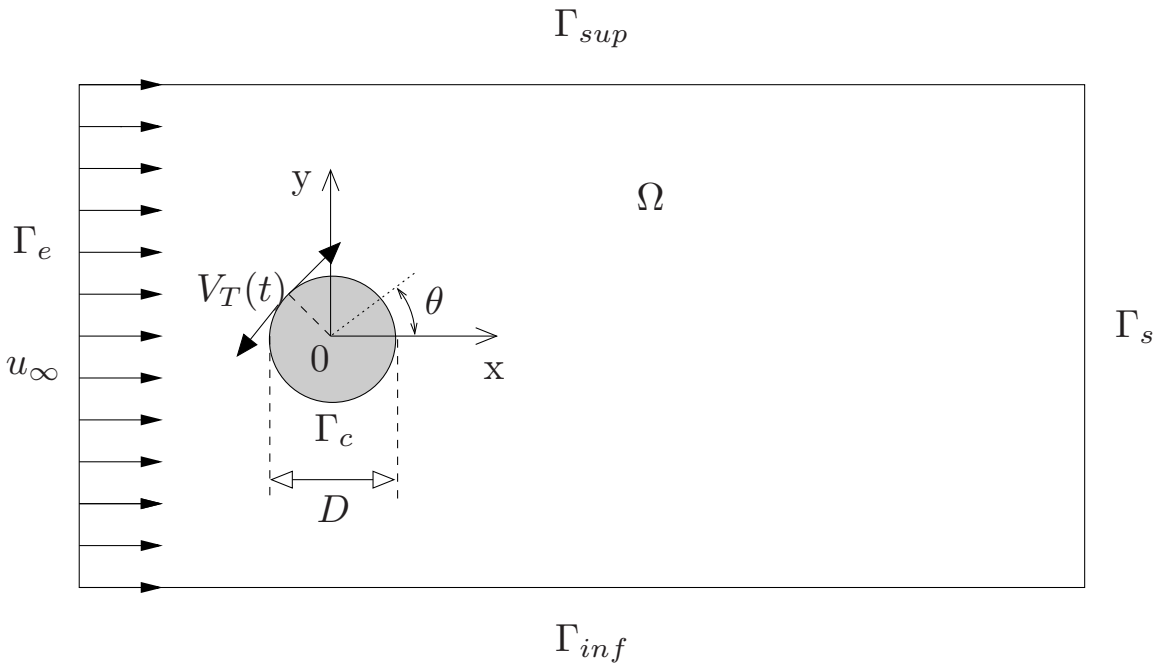
$$\mathcal{G}_i = -\left(\boldsymbol{\Phi}_i, \left(\boldsymbol{u_c} \cdot \boldsymbol{\nabla}\right) \boldsymbol{u_c}\right)_\Omega$$

Cylinder wake flow

Configuration

- Two dimensional flow around a circular cylinder at Re = 200
- Viscous, incompressible and Newtonian fluid
- Cylinder oscillation with a tangential velocity $\gamma(t)$

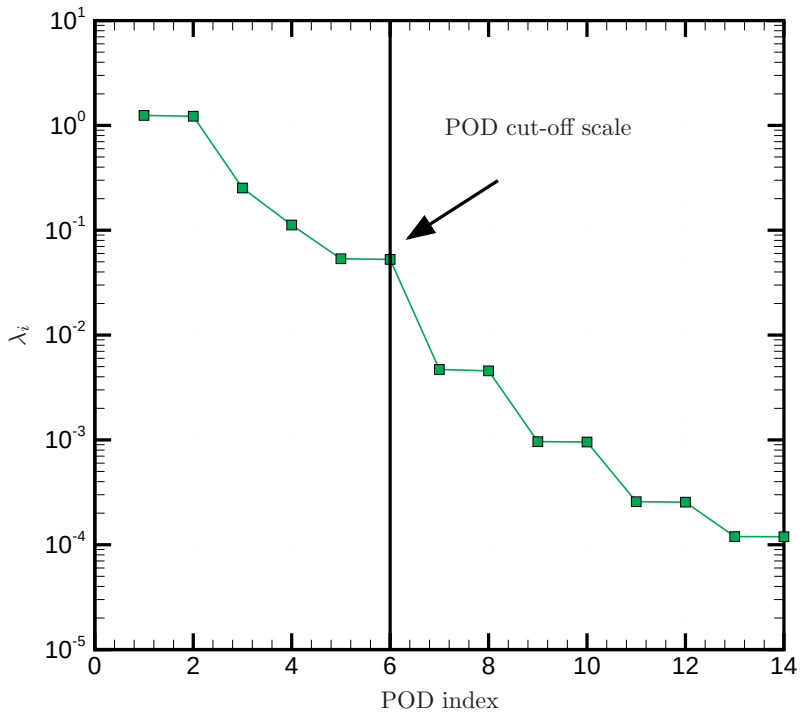
$$\gamma(t) = \frac{V_T}{u_\infty} = A \sin(2\pi St_f t)$$

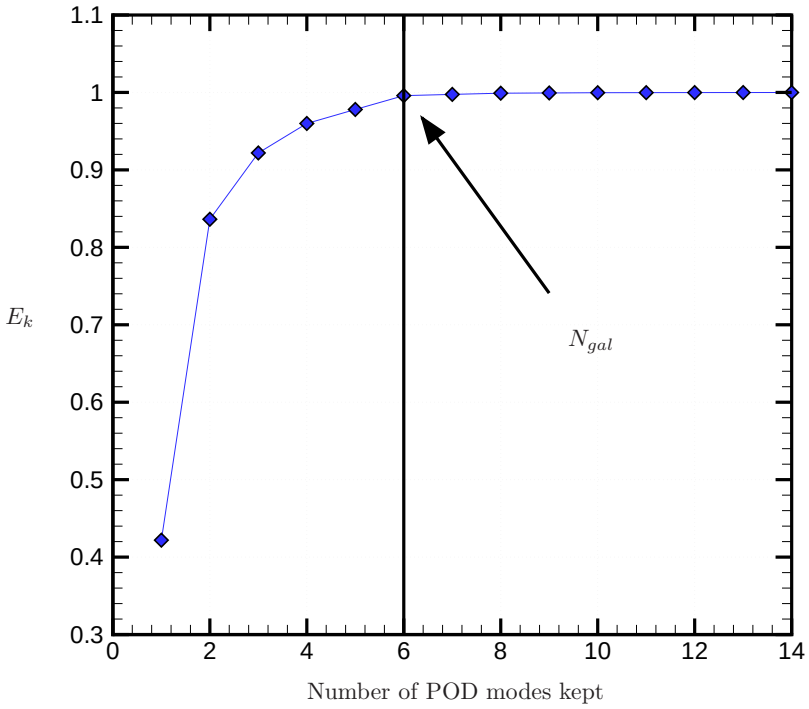


POD of the controlled wake flow ($\gamma \neq 0$) $A = 2$ and $St_f = 0,5$

- 361 snapshots taken uniformly over $T = 18$
- Energetic Content: $E_k = \sum_{i=1}^{k} \lambda_i \Big/ \sum_{i=1}^{N_{\text{POD}}} \lambda_i$

Objective: Determine POD truncation with 99% of relative energy





$$N_{gal} = \arg\min_k E_k \text{ such that } E_{N_{\text{gal}}} > 99\% \quad \Rightarrow \quad \boldsymbol{N_{\text{gal}}} = 6\ !$$

POD of the controlled wake flow ($\gamma \neq 0$)

Velocity modes

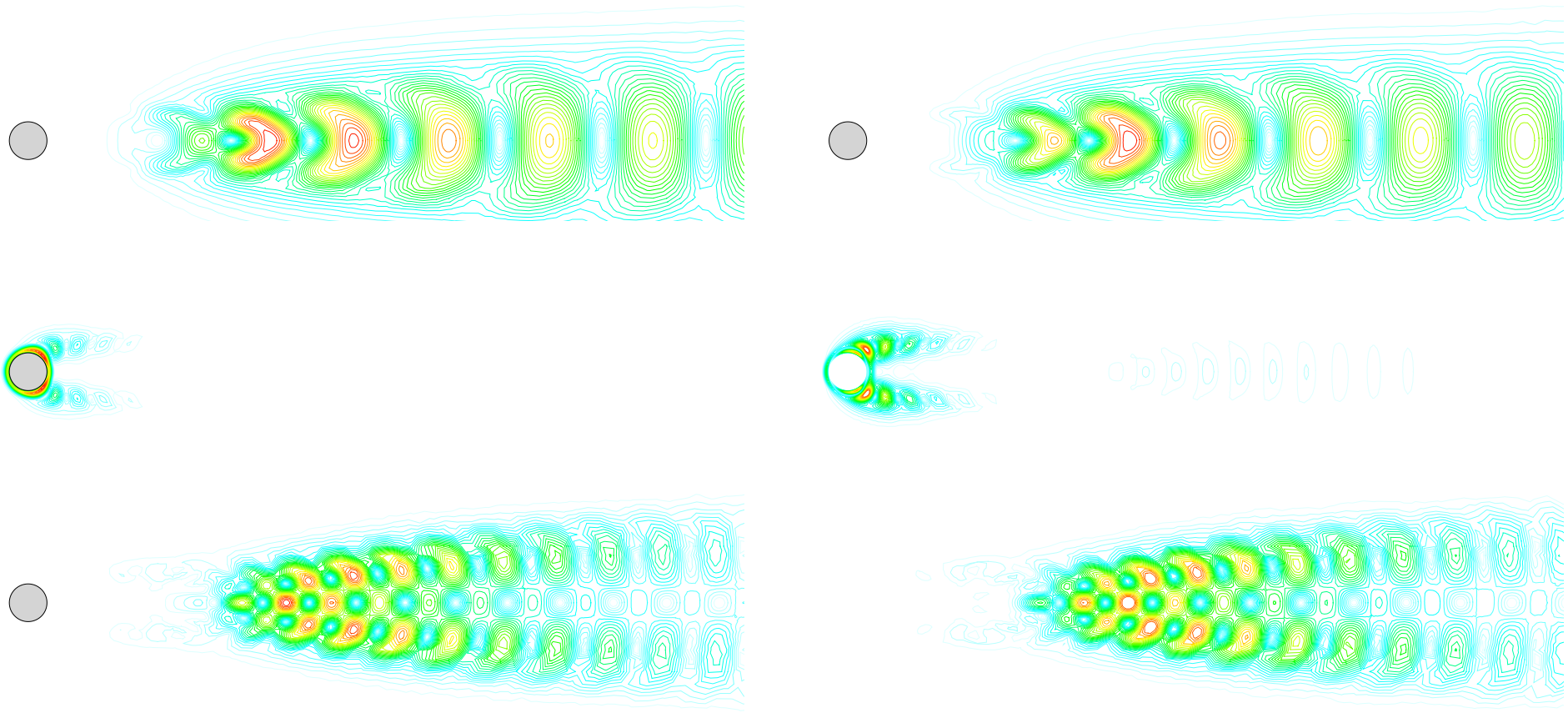


Fig. : *Iso-values of the first 6 POD modes*

$\gamma(t) = A\sin(2\pi St_f t)$ *with* $A = 2$ *and* $St_f = 0,5$.

POD of the controlled wake flow ($\gamma \neq 0$) Integration and calibration

Reconstruction errors of POD ROM $\Rightarrow$ time amplification of the modes

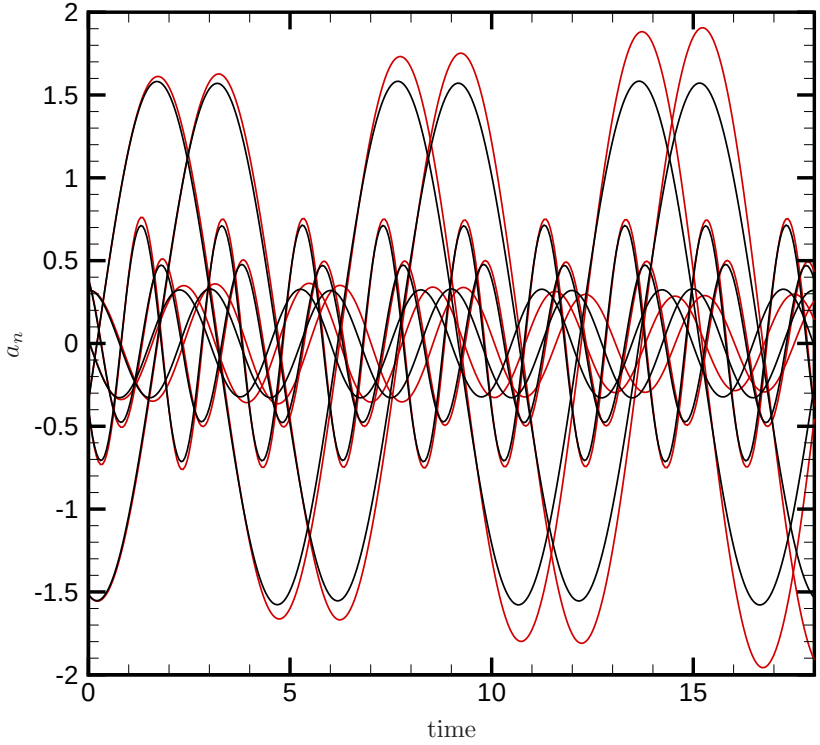


Fig. : *Time evolution of the first 6 POD modes ($A = 2$ and $St_f = 0,5$).*

▷ Reasons:

- Extraction of large scale structures carrying energy
- Main of the dissipation contained in the small structures

▷ Solutions:

- Identification method, Data Assimilation for instance

——— projection (Navier-Stokes) : $a^P(t)$
——— prediction before identification (POD ROM)

Outline

Perspectives

- For linear models
  - Balanced Truncation
  - Balanced Proper Orthogonal Decomposition (BPOD)
  - Eigensystem Realization Algorithm (ERA)
- Non linear dimensionality reduction methods
  - Kernel Principal Component Analysis (K-PCA)
  - MultiDimensional Scaling (MDS)
  - Isomap
  - Locally Linear Embedding (LLE)
- High-Order Principal Component Analysis (HO-PCA)
- Resolvent analysis
- ...
- ...

Outline

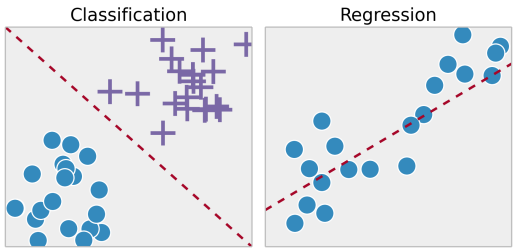
Machine Learning

Sub categories

1. Supervised Learning

   Learn a mapping from inputs $\boldsymbol{x}$ to outputs $\boldsymbol{y}$ given a labeled set $\mathcal{D}_{\mathrm{SL}} = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^{N}$.

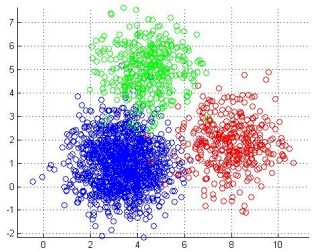
   - Classification or pattern recognition
   - Regression Genetic Programming



2. Unsupervised Learning

   Given only inputs $\mathcal{D}_{\mathrm{UL}} = \{\boldsymbol{x}_i\}_{i=1}^{N}$, discover "interesting patterns"

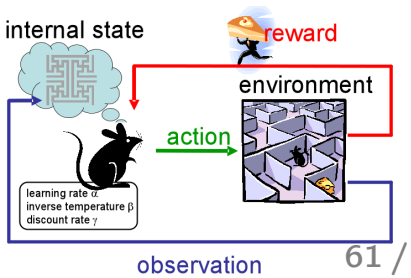
   - Clustering: CROM
   - Dimensionality Reduction: PCA, POD, DMD

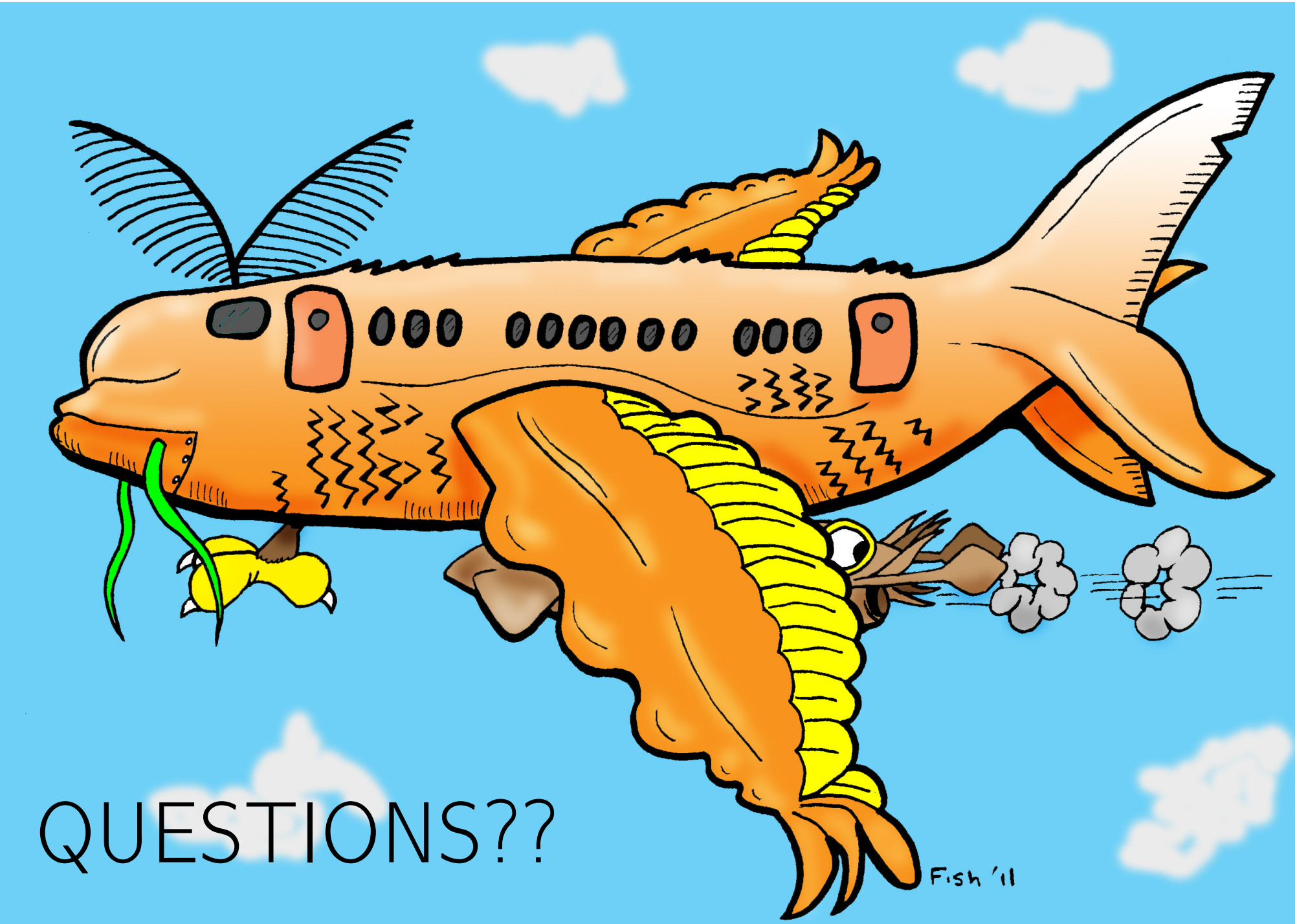


3. Reinforcement Learning

   How to take actions in an environment so as to maximize a cumulative reward.

   Discretized and continuous RL





QUESTIONS??