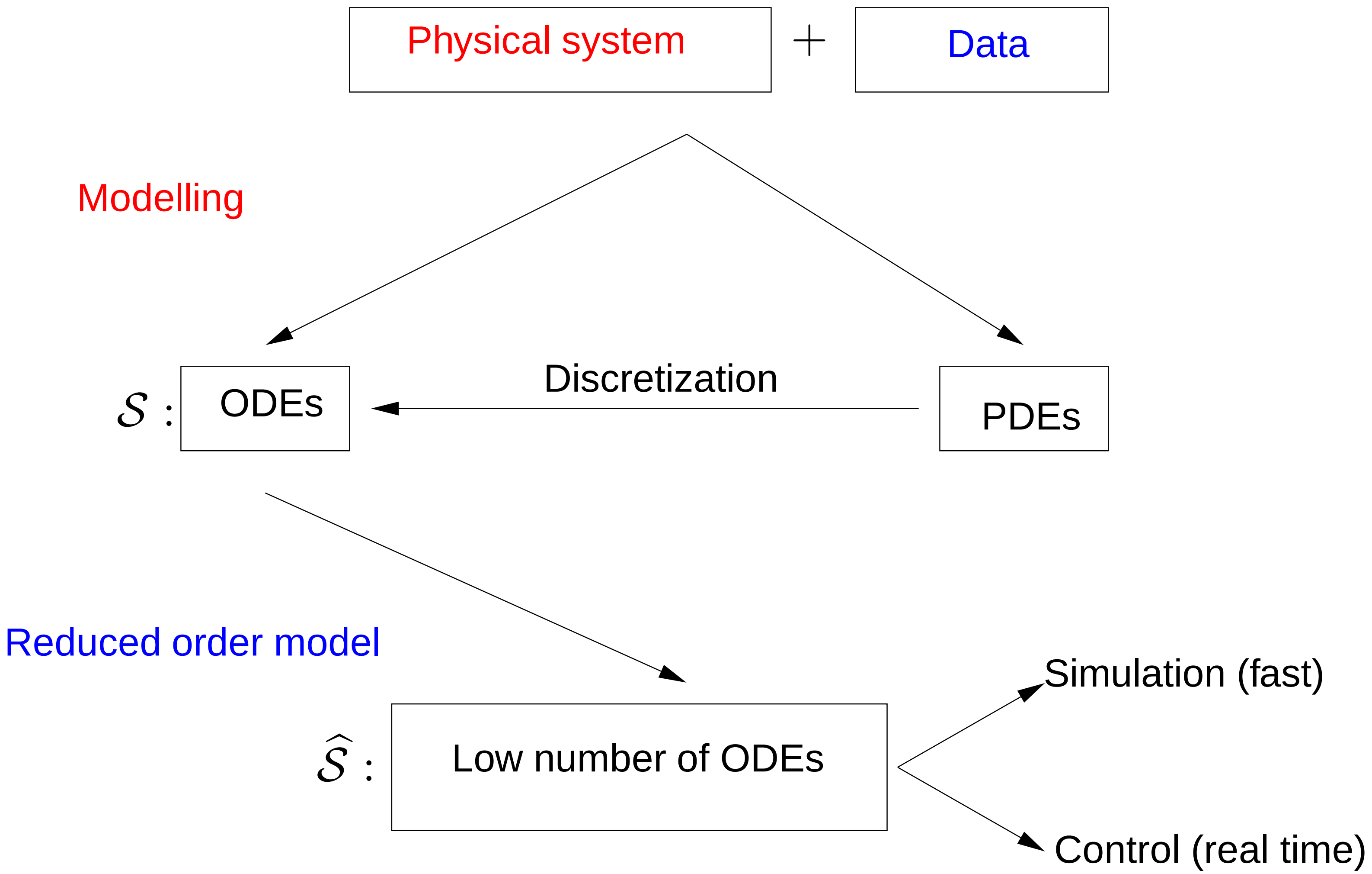
Model reduction by POD

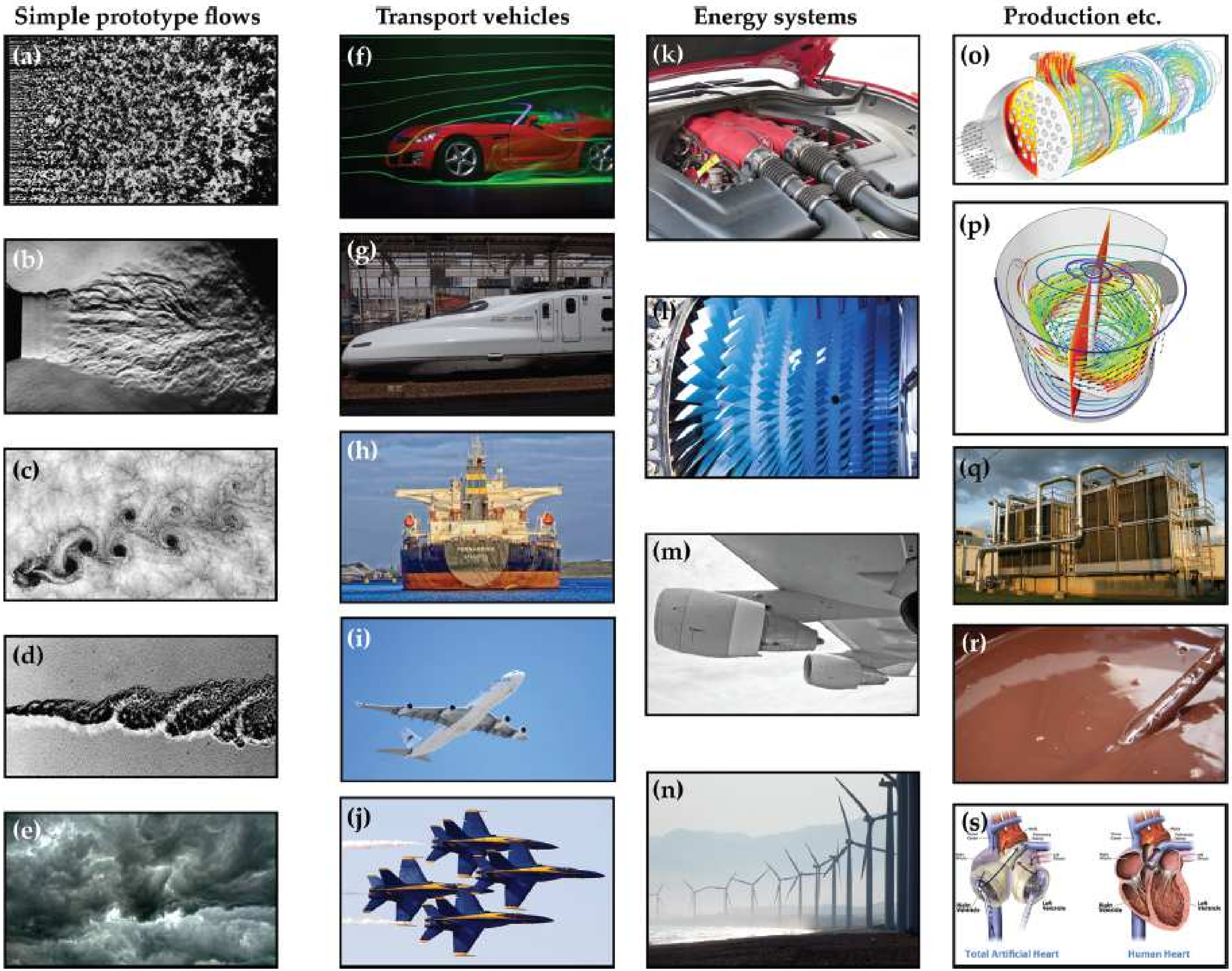
Laurent Cordier

Institut Pprime

ANF Réduction de dimension, September 26, 2017, Ile d'Oléron

Reduced-Order Modelling





From Brunton, Noack, AMR, 2015

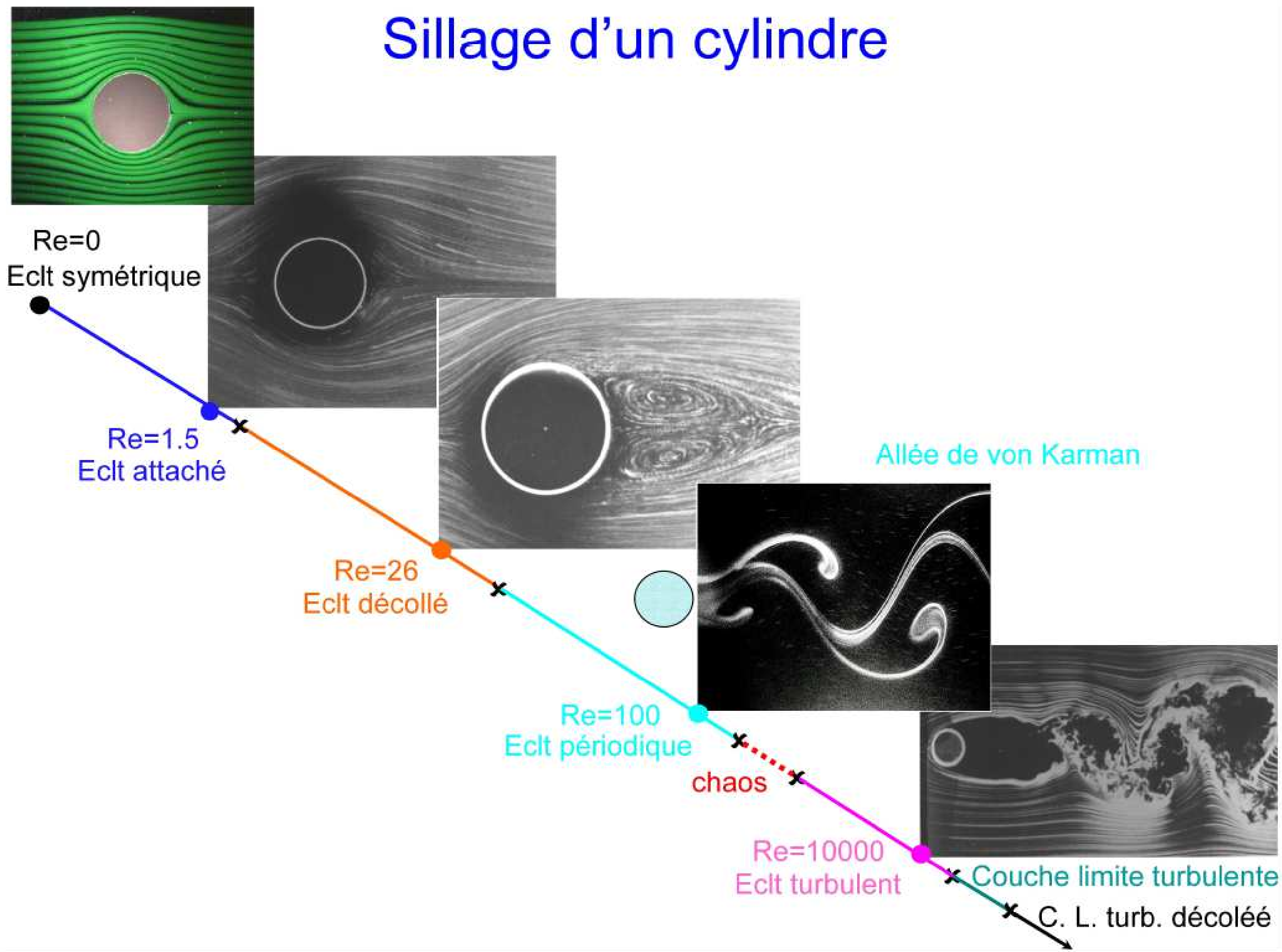


Fig. : *From Gallaire (2009)*

Ecoulement « naturel » :

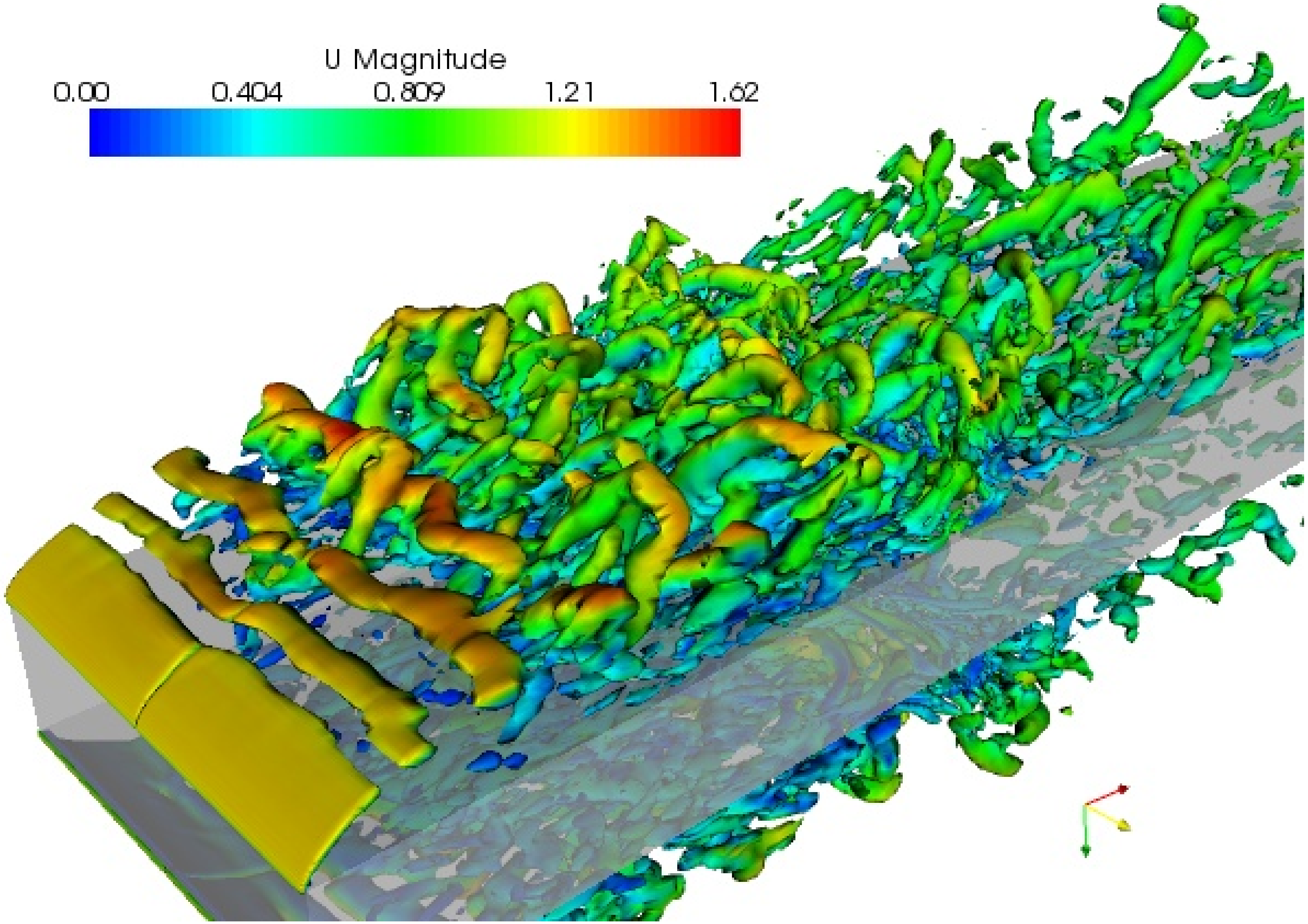
isoW0.avi

Ecoulement contrôlé de manière optimale par rotation :

isoWopt.avi

Ecoulement contrôlé de manière optimale par oscillation verticale du cylindre :

Cylinder_control_oscillation_MPEG4.avi



From L. Mathelin (LIMSI)

Reduced-Order Modelling

- Ex. from Spalart et al. (1997): wing considered at cruising flight conditions *i.e.* 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 *Proper Orthogonal Decomposition* (Lumley, 1967),
- Reduced-order models based on balanced and/or global modes.

Proper Orthogonal Decomposition

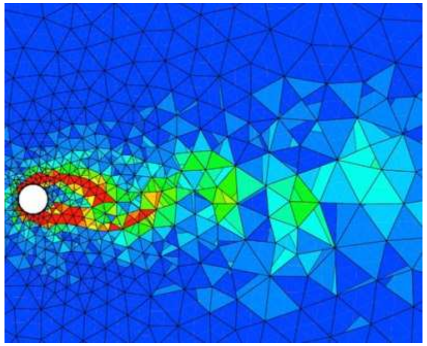
- Also known as:
  - Karhunen-Loève decomposition: Karhunen (1946), Loève (1945) ;
  - Principal Component Analysis: Hotelling (1953) ;
  - Singular Value Decomposition: Golub and Van Loan (1983).
- Applications include:
  - Random variables (Papoulis, 1965) ;
  - Image processing (Rosenfeld and Kak, 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.
- Two possibilities of presentation:
  1. Mathematical framework: SVD
  2. Turbulence framework: Hilbert-Schmidt theory

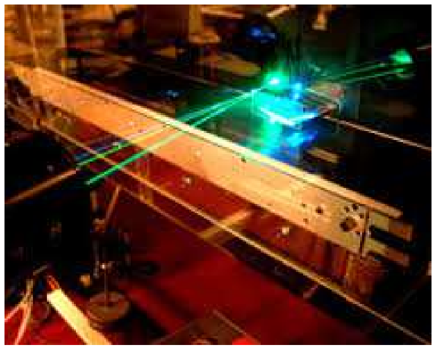
From data to Snapshot Data Matrix

Simulations



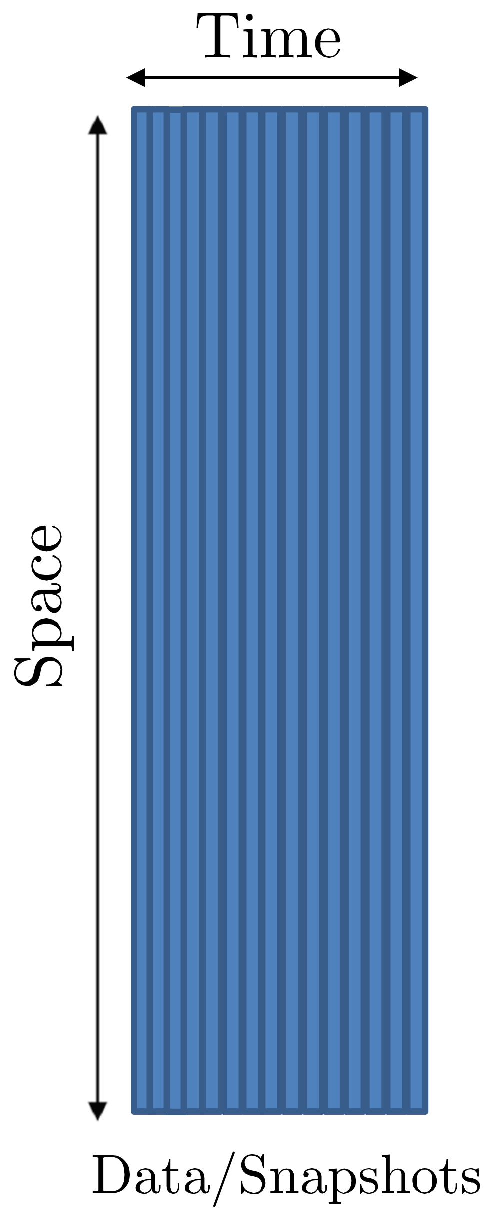
- Velocity fields
- Pressure fields
- Vorticity fields
- Tracers

Experiments



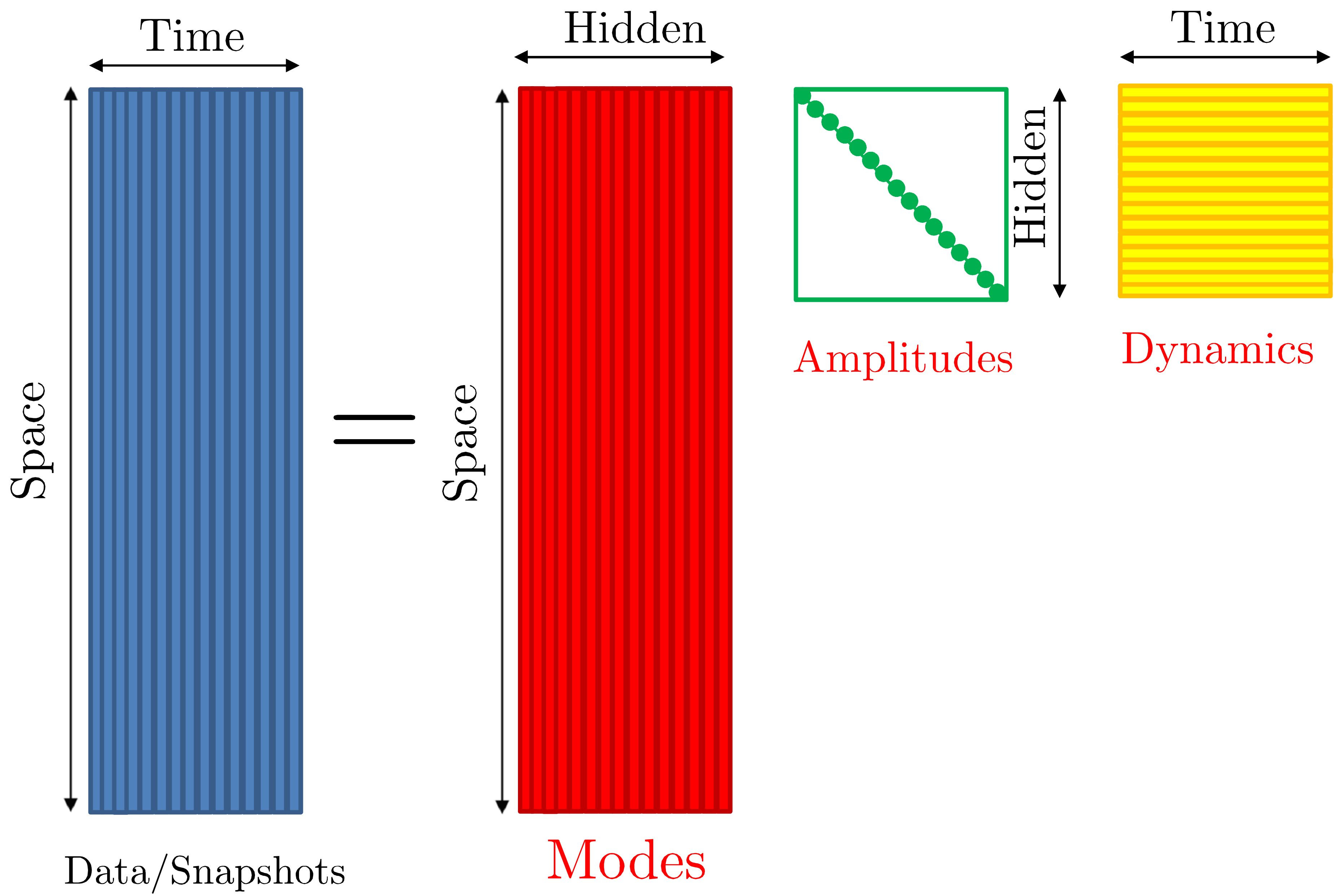
- PIV
- Hot-wires
- LDV
- Visualizations

$\Longrightarrow$ $S =$

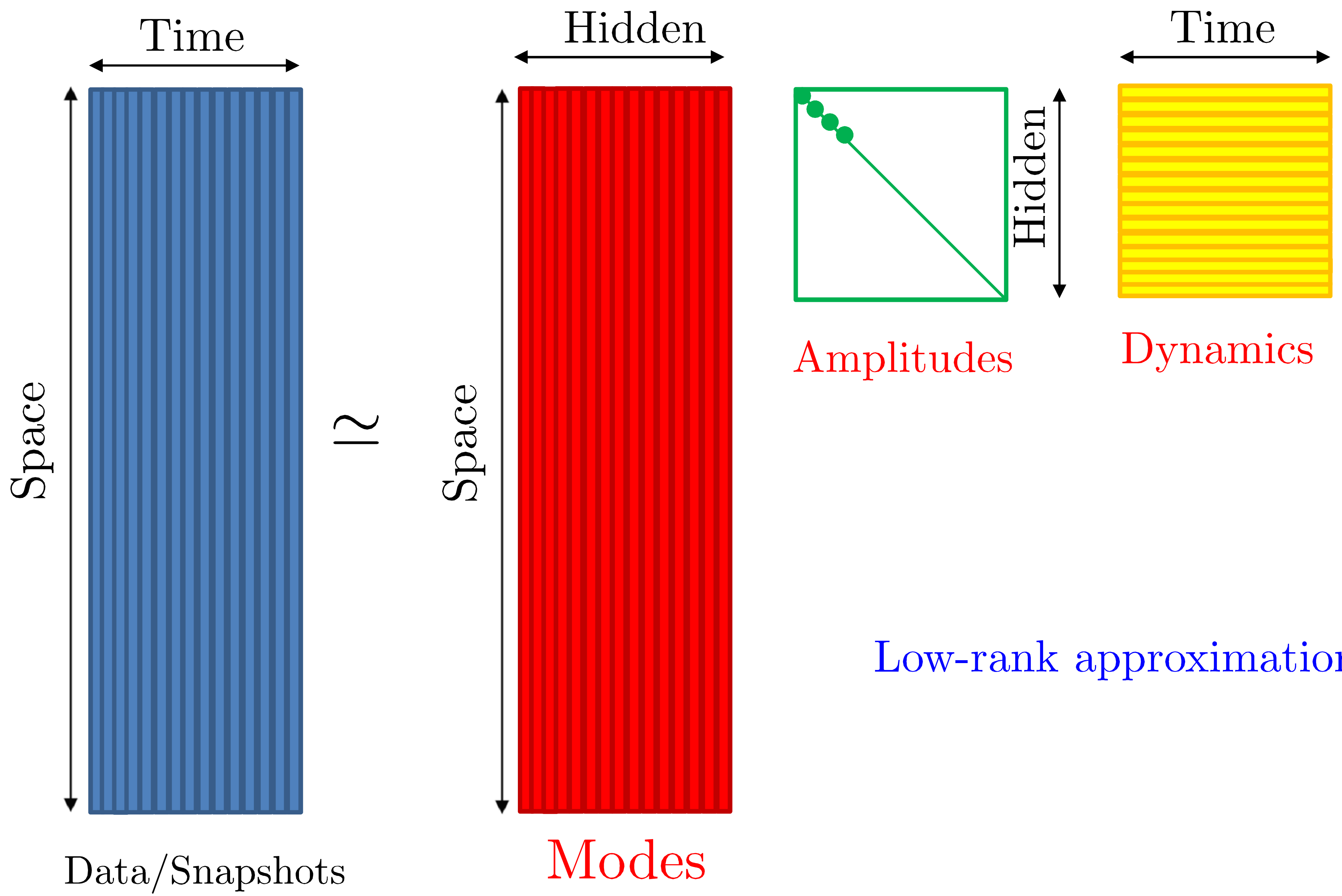


Thanks P. Schmid for the inspiration !

Data analysis as a matrix decomposition



Model reduction: exploit the redundancy



Low-rank approximation

Snapshot Data Matrix

$$\boldsymbol{u} = (u_1, u_2, \cdots, u_{n_c}) \;\; ; \;\; \boldsymbol{x} = (x_1, x_2, \cdots, x_{n_x}) \;\; ; \;\; \boldsymbol{t} = (t_1, t_2, \cdots, t_{N_t}) \;\; ; \;\; N_x = n_x \times n_c$$

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

The POD basis problem

- Given a collection of N_t functions $\boldsymbol{u}(\boldsymbol{x}, t_i)$
- Find a k dimensional subspace $V_k^{\text{POD}} = \text{span}\left(\boldsymbol{\phi}^{(1)}, \cdots, \boldsymbol{\phi}^{(k)}\right)$ which minimizes

$$\mathcal{J}(\Pi_{\text{POD}}) = \sum_{i=1}^{N_t} \|\boldsymbol{u}(\boldsymbol{x}, t_i) - \Pi_{\text{POD}}\boldsymbol{u}(\boldsymbol{x}, t_i)\|_{\Omega}^2$$

where $\mathcal{J}$ is the mean squared error.
Π_{POD} is the orthogonal projector on the space spanned by the functions $\{\boldsymbol{\phi}^{(i)}\}_{i=1}^{k}$.

- Minimizing $\mathcal{J}$ is equivalent to minimize

$$\mathcal{J}(\boldsymbol{\phi}^{(1)}, \cdots, \boldsymbol{\phi}^{(k)}) = \sum_{i=1}^{N_t} \|\boldsymbol{u}(\boldsymbol{x}, t_i) - \sum_{j=1}^{k}\left(\boldsymbol{u}(\boldsymbol{x}, t_i), \boldsymbol{\phi}^{(j)}(\boldsymbol{x})\right)_{\Omega} \boldsymbol{\phi}^{(j)}(\boldsymbol{x})\|_{\Omega}^2.$$

- The functions $\phi^{(j)}$ are orthonormal, i.e.

$$\left(\boldsymbol{\phi}^{(k_1)}, \boldsymbol{\phi}^{(k_2)}\right)_{\Omega} = \int_{\Omega} \boldsymbol{\phi}^{(k_1)}(\boldsymbol{x}) \cdot \boldsymbol{\phi}^{(k_2)}(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x} = \delta_{k_1 k_2} = \begin{cases} 0 & \text{for } k_1 \neq k_2, \\ 1 & \text{for } k_1 = k_2, \end{cases}$$

- The solutions of the minimization problem are given by the truncated Singular Value Decomposition of length k of S.

Singular Value Decomposition (SVD)

Definition

$$\boxed{S = U \Sigma V^H} \in \mathbb{C}^{N_x \times N_t}$$ 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 $p = \min(N_x, N_t)$

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

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

▷ SVD and eigenvalue problems

1. Singular values

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

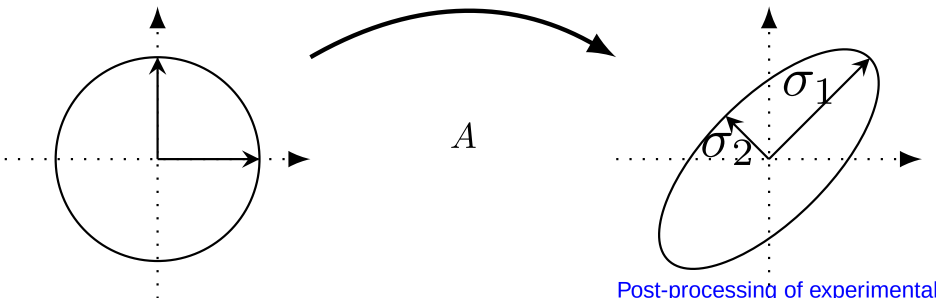
2. $\left(S^H S\right) V = V\Sigma^2 = V\Lambda$, hence columns of V are ev's of $S^H S \in \mathbb{C}^{N_t \times N_t}$
3. $\left(S S^H\right) U = U\Sigma^2 = U\Lambda$, hence columns of U are ev's of $S S^H \in \mathbb{C}^{N_x \times N_x}$

▷ 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 in the sense that

$$S\boldsymbol{v_i} = U\Sigma V^H \boldsymbol{v_i} = U\Sigma \boldsymbol{e_i} = \sigma_i \boldsymbol{u_i} \quad i = 1, \cdots, r$$

which shows that S maps input $\boldsymbol{v_i}$ to output $\boldsymbol{u_i}$ with amplification σ_i.



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

$$S = \begin{pmatrix} \boldsymbol{u_1} & \cdots & \boldsymbol{u_{N_x}} \end{pmatrix} \left(\begin{array}{ccccc|cccc} \sigma_1 & & & & & 0 & \cdots & \cdots & 0 \\ & \ddots & & & & \vdots & & & \vdots \\ & & \ddots & & & \vdots & & & \vdots \\ & & & \ddots & & \vdots & & & \vdots \\ & & & & \sigma_{N_x} & 0 & \cdots & \cdots & 0 \end{array}\right) \begin{pmatrix} \boldsymbol{v_1}^H \\ \vdots \\ \vdots \\ \boldsymbol{v_{N_x}}^H \\ \hline \boldsymbol{v_{N_x+1}}^H \\ \vdots \\ \vdots \\ \boldsymbol{v_{N_t}}^H \end{pmatrix}$$

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

$$S = \begin{pmatrix} \boldsymbol{u_1} & \cdots & \boldsymbol{u_{N_t}} & \boldsymbol{u_{N_t+1}} \cdots & \boldsymbol{u_{N_x}} \end{pmatrix} \begin{pmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & \sigma_{N_t} \\ \hline 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & & & & \vdots \\ \vdots & & & & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{v_1}^H \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \boldsymbol{v_{N_t}}^H \end{pmatrix}.$$

▷ Truncated approximations

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

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

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

$$\boxed{S = \sigma_1 \, \boldsymbol{u_1 v_1^H} + \sigma_2 \, \boldsymbol{u_2 v_2^H} + \cdots + \sigma_r \, \boldsymbol{u_r v_r^H} \, .}$$

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

$$S_k = \sigma_1 \, \boldsymbol{u_1 v_1^H} + \sigma_2 \, \boldsymbol{u_2 v_2^H} + \cdots + \sigma_k \, \boldsymbol{u_k v_k^H} \, .$$

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

▷ Norms

⋆ 2-induced norm: $||S||_2 = \max_{||x||_2 = 1} ||Sx||_2 = \sigma_1$.

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

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

Criterion:

minimization of the norm (2-norm or Frobenius norm) of the error $E = S - S_k$.

Theorem: Eckart-Young

$$\min_{\text{rank}\,(X) \leq k} \|S - X\|_2 = \|S - S_k\|_2 = \sigma_{k+1}(S)$$

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

with $S_k = U \begin{pmatrix} \Sigma_k & 0 \\ 0 & 0 \end{pmatrix} V^H = \sigma_1 \boldsymbol{u_1} \boldsymbol{v_1}^H + \sigma_2 \boldsymbol{u_2} \boldsymbol{v_2}^H + \cdots + \sigma_k \boldsymbol{u_k} \boldsymbol{v_k}^H$

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

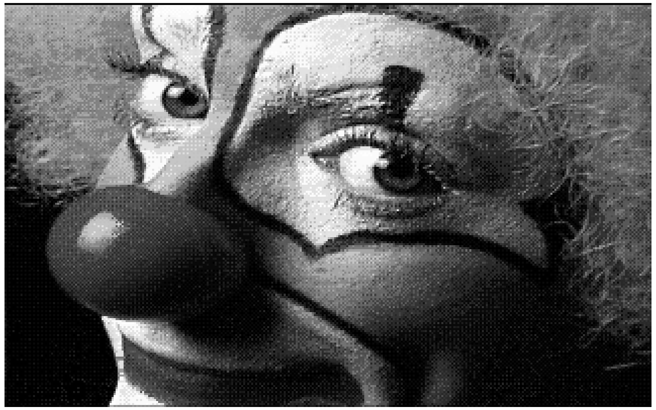
Image compression by truncated SVD

- Consider an image with $n_i \times n_j$ pixels. This image can be stored as a matrix $S \in \mathbb{R}^{n_i \times n_j}$ where s_{ij} contains the grey level of pixel (i, j).
- Memory: 4 bytes per pixel $\Longrightarrow 4 \times n_i \times n_j$ bytes
- Eckart-Young th.: an approximation of S with k singular modes writes

$$S_k = \sigma_1 \boldsymbol{u_1} \boldsymbol{v_1}^H + \sigma_2 \boldsymbol{u_2} \boldsymbol{v_2}^H + \cdots + \sigma_k \boldsymbol{u_k} \boldsymbol{v_k}^H, \quad \text{with} \quad \|S - S_k\|_2 = \sigma_{k+1}(S).$$

- Size reduction
  - Store $\sigma_1, \cdots, \sigma_k, \boldsymbol{u_1}, \cdots, \boldsymbol{u_k}$ and $\boldsymbol{v_1}^H, \cdots, \boldsymbol{v_k}^H$ in place of S
  - Memory $4 \times k \times (1 + n_i + n_j)$ bytes
  - Indicators of savings
    - ⋆ Compression factor: $C_k = \dfrac{n_i \, n_j}{k \, (1 + n_i + n_j)}$
    - ⋆ Data storage: $D_k = \dfrac{1}{C_k}$
    - ⋆ Retained "energy": $E_{\text{SVD}}(k) = \dfrac{\sum_{i=1}^{k} \sigma_i^2(S)}{\sum_{i=1}^{r} \sigma_i^2(S)}$

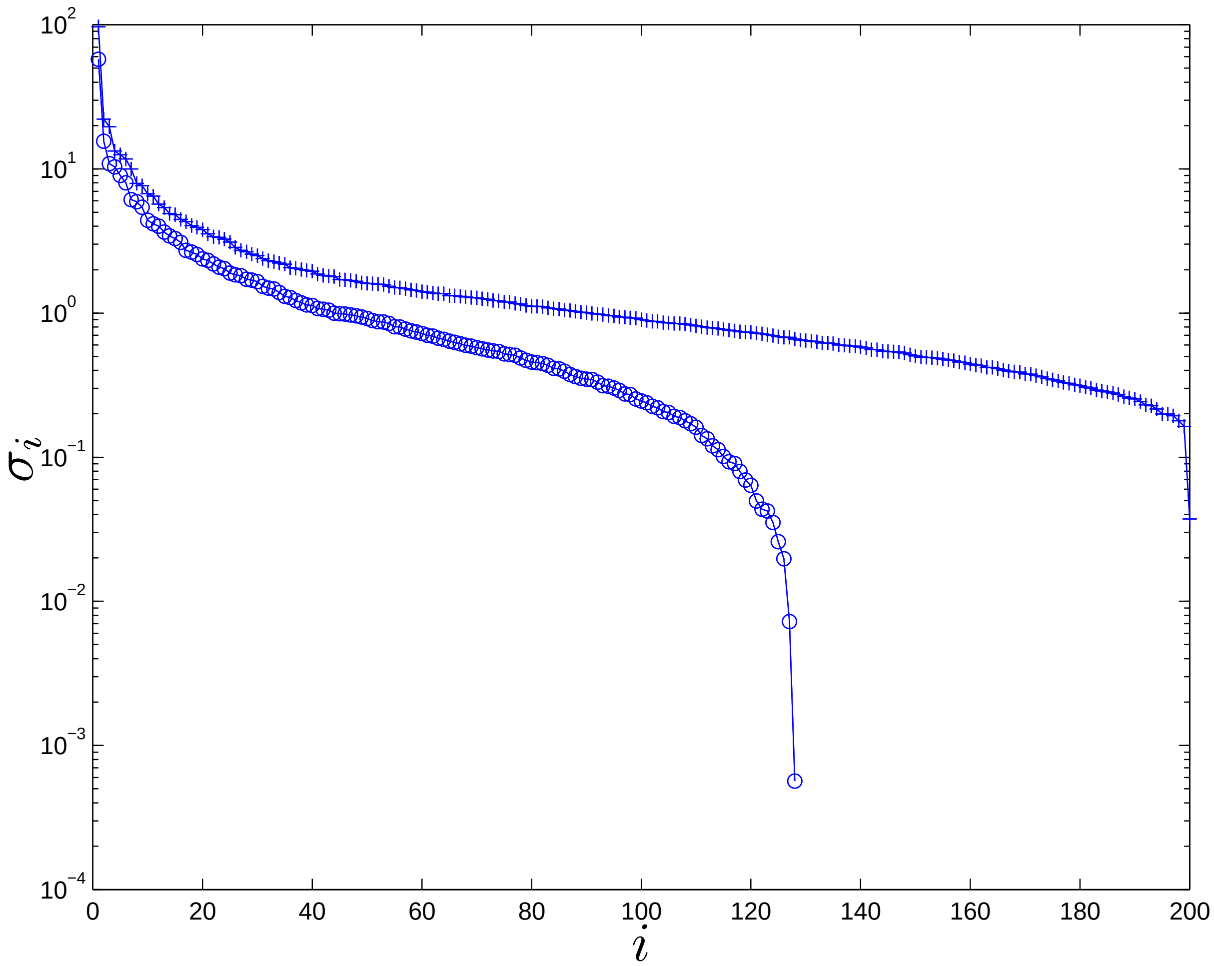
Image compression by truncated SVD



(a) Clown: matrix 200×330, rank: 200, size: 258 kb

(b) Trees: matrix 128×128, rank: 128, size: 64 kb

Image compression by truncated SVD

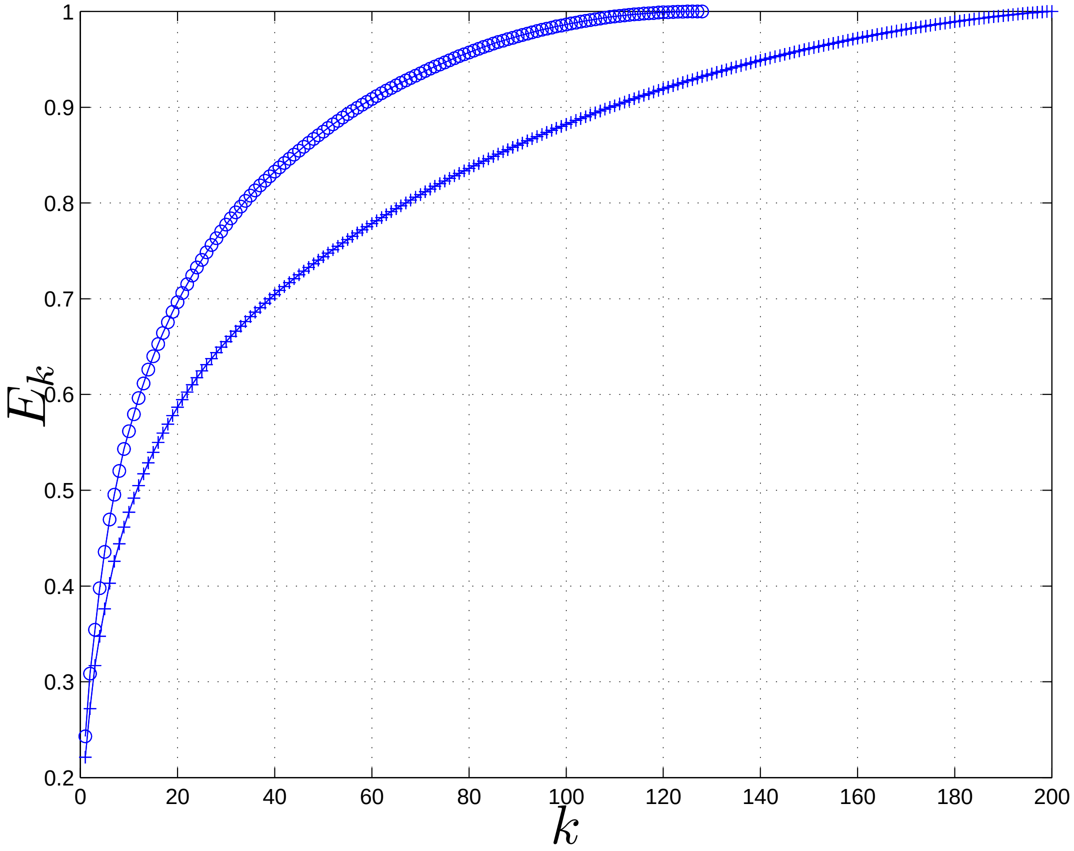


○: "Trees" image ; $+$: "Clown" image

Faster decrease of σ_i for the "Trees" than for the "Clown".

Image compression by truncated SVD

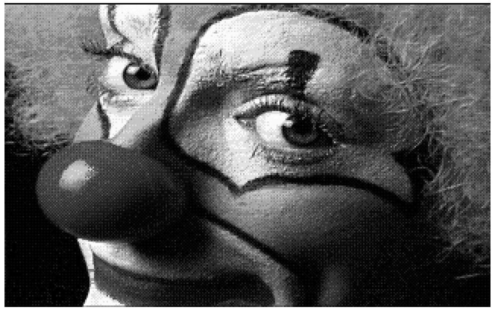
For an approximation of level k: $E_{\text{SVD}}(k) = \dfrac{\sum_{i=1}^{k} \sigma_i^2(S)}{\sum_{i=1}^{r} \sigma_i^2(S)}$



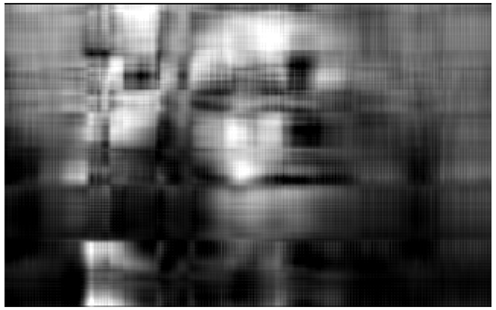
○: "Trees" image ; $+$: "Clown" image

"Trees" image easier to represent with a low-rank approximation than the "Clown" image.

Image compression by truncated SVD



(c) Original image



(d) $k = 6$; $D_k = 4.8\%$

(e) $k = 12$; $D_k = 9.6\%$

(f) $k = 20$; $D_k = 16\%$

Image compression by truncated SVD

"Trees"

(g) Original image

(h) $k = 6$; $D_k = 9.4\%$

(i) $k = 12$; $D_k = 18.8\%$

(j) $k = 20$; $D_k = 31.2\%$

▷ Solve equation

$$\frac{\partial u}{\partial t} = \frac{1}{\text{Re}} \frac{\partial^2 u}{\partial x^2} - u \frac{\partial u}{\partial x} \quad \forall x \in]0;1[\quad \text{and} \quad t \in]0;T[$$

with

$$\begin{aligned} u(x,0) &= \sin(\pi x) & \forall x \in]0;1[\quad (IC) \\ u(0,t) = u(1,t) &= 0 & \forall t \in]0;T] \quad (BC) \end{aligned}$$

▷ Analytical solution

$$u_a(x,t) = \frac{2\pi}{\text{Re}} \frac{\sum_{n=1}^{\infty} a_n \, n \, \sin(n\pi x) \exp\left(-n^2 \pi^2 t/\text{Re}\right)}{a_0 + \sum_{n=1}^{\infty} a_n \, \cos(n\pi x) \, \exp\left(-n^2 \pi^2 t/\text{Re}\right)}$$

where a_n are Fourier coefficients.

Application to the 1D Burgers equation

- Numerical parameters for the POD analysis
  - Re $= 10$,
  - $T = 0.1$ and $\Delta t = 10^{-4}$ i.e. $N_t = 1000$ snapshots in the data base,
  - $x \in [0; 1]$ and $\Delta x = \frac{1-0}{N_x - 1}$ with $N_x = 100$.

Matlab

- 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.*

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

$\Longrightarrow$ 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.

- We introduce W_1 and W_2, two biorthogonal matrices of size $\mathbb{R}^{n_{\mathcal{X}} \times n_k}$, such that $W_2^H Q W_1 = I_{n_k}$ 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: $W_2^H Q \boldsymbol{\mathcal{R}} = \mathbf{0}_{n_k}$ *i.e.*

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

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

▷ 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)

▷ Solve equation

$$\frac{\partial u}{\partial t} = \frac{1}{\text{Re}}\frac{\partial^2 u}{\partial x^2} - u\frac{\partial u}{\partial x} \quad \forall x \in]0;1[\quad \text{and} \quad t \in]0;T[$$

with

$$\begin{aligned} u(x,0) &= \sin(\pi x) & \forall x \in]0;1[\quad (IC) \\ u(0,t) = u(1,t) &= 0 & \forall t \in]0;T] \quad (BC) \end{aligned}$$

▷ Analytical solution

$$u_a(x,t) = \frac{2\pi}{\text{Re}} \frac{\sum_{n=1}^{\infty} a_n\, n\, \sin(n\pi x) \exp\left(-n^2\pi^2 t/\text{Re}\right)}{a_0 + \sum_{n=1}^{\infty} a_n\, \cos(n\pi x)\, \exp\left(-n^2\pi^2 t/\text{Re}\right)}$$

Matlab

Proper Orthogonal Decomposition (POD)

▷ Introduced in turbulence by Lumley (1967).

▷ Method of information compression

▷ Look for a realization $\boldsymbol{\Phi}(\boldsymbol{X})$ which is closer, in an average sense, to realizations $\boldsymbol{u}(\boldsymbol{X})$ with $\boldsymbol{X} = (\boldsymbol{x}, t) \in \mathcal{D} = \Omega \times \mathbb{R}^+$

▷ $\boldsymbol{\Phi}(\boldsymbol{X})$ solution of the problem:

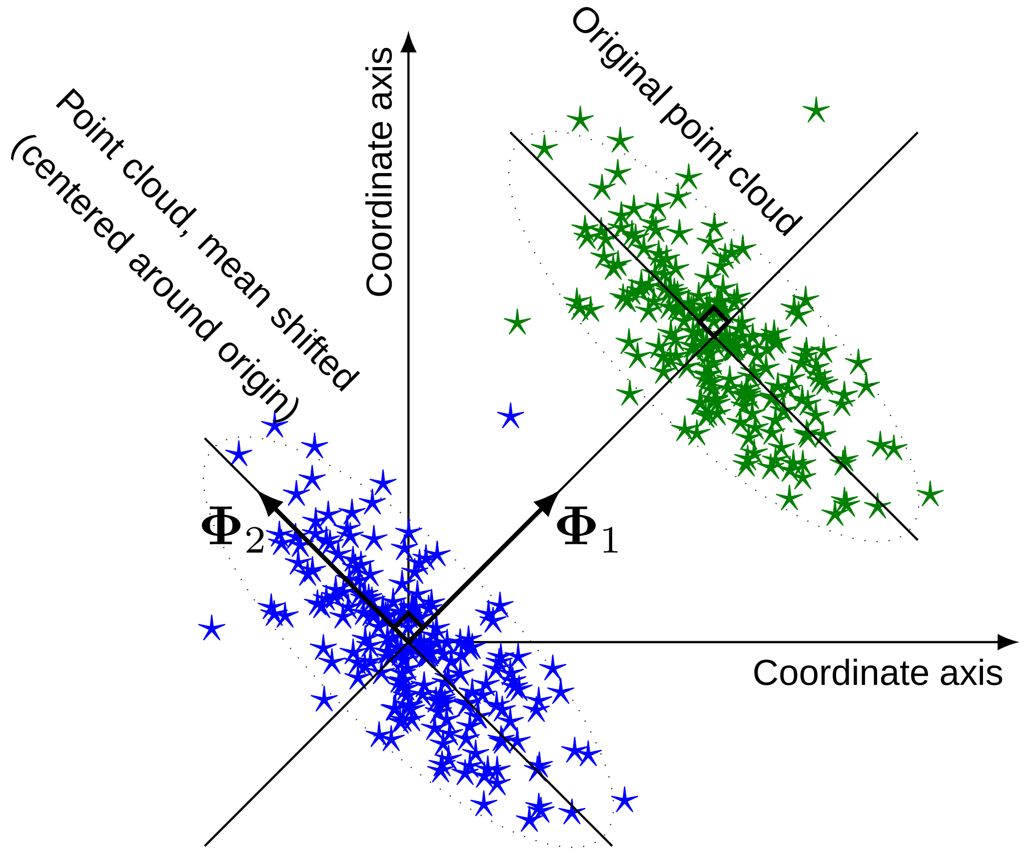
$$\max_{\boldsymbol{\Phi}} \langle |(\boldsymbol{u}, \boldsymbol{\Phi})|^2 \rangle \quad \text{s.t.} \quad \|\boldsymbol{\Phi}\|^2 = 1.$$

This is a constrained optimization problem !

▷ Optimal convergence in a given norm of $\boldsymbol{\Phi}(\boldsymbol{X})$

⇒ Dynamical order reduction of the ensemble data is guaranteed (Eckart-Young theorem).

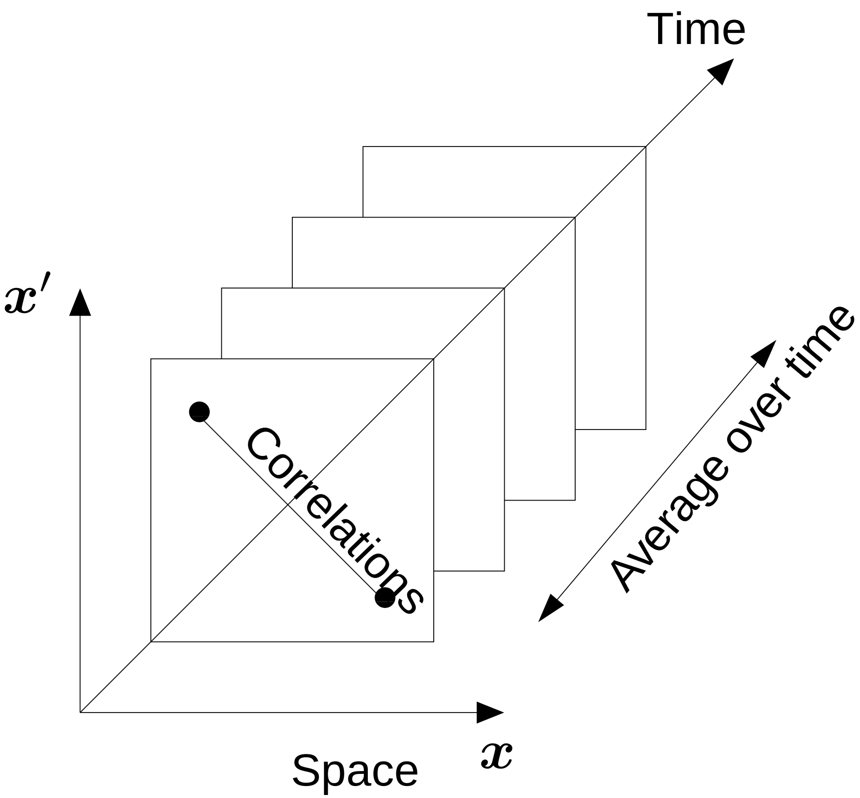
No results for the POD-based ROM !



POD approaches depend on:

- the inner product $(\cdot, \cdot)$ Not discussed here
  - L^2
  - H^1
  - ...
- the variable $\boldsymbol{X}$ used
  - spatial $\boldsymbol{x} = (x, y, z)$
  - temporal t
  - control parameters $\boldsymbol{c}$, for instance Reynolds number ...
- the averaging operation $\langle . \rangle$
  - spatial
  - temporal
- the input collection Not discussed here

  $\Longrightarrow$ interest of using sampling methods in the control parameter space:
  - Latin Hypercube Sampling
  - Centroidal Voronoi Tessellation
  - ...



- $\boldsymbol{X} = \boldsymbol{x} = (x, y, z)$
- $\langle . \rangle = \frac{1}{T} \int_T . \, \mathrm{d}t$

  i.e. temporal average (evaluated as ensemble average).

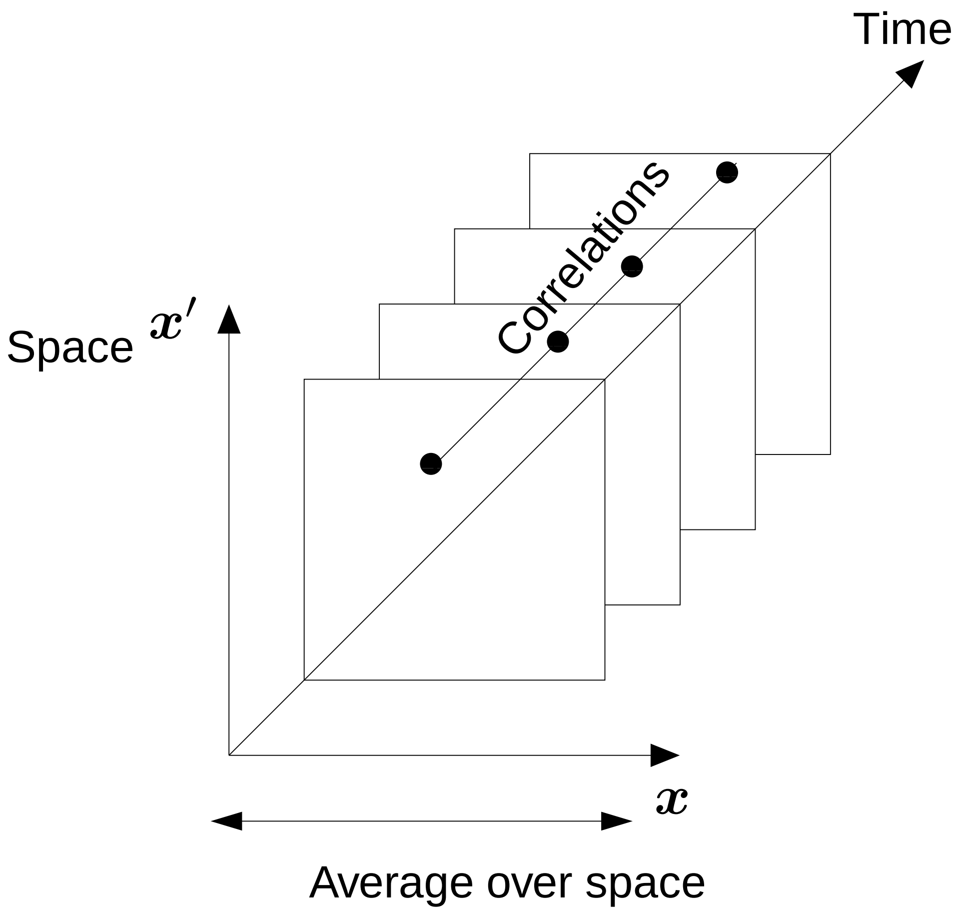
Fredholm equation:

$$\boxed{\sum_{j=1}^{n_c} \int_{\Omega} R_{ij}(\boldsymbol{x}, \boldsymbol{x}')\, \Phi_j^{(n)}(\boldsymbol{x}')\, \mathrm{d}\boldsymbol{x}' = \lambda^{(n)}\, \Phi_i^{(n)}(\boldsymbol{x})}$$

where $R_{ij}(\boldsymbol{x}, \boldsymbol{x'})$ is the two-point spatial correlation tensor defined as:

$$R_{ij}(\boldsymbol{x}, \boldsymbol{x'}) = \frac{1}{T} \int_T u_i(\boldsymbol{x}, t) u_j(\boldsymbol{x'}, t)\, \mathrm{d}\, t = \sum_{n=1}^{N_{\text{POD}}} \lambda^{(n)} \Phi_i^{(n)}(\boldsymbol{x}) \Phi_j^{(n)*}(\boldsymbol{x'})$$

- Eigenvectors are space dependent.
- Size: $N_{\text{POD}} = N_x \times n_c$



- $\boldsymbol{X} = (t)$
- $\langle . \rangle = \int_{\Omega} . \, \mathrm{d}\boldsymbol{x}$

i.e. spatial average.

Fredholm equation:

$$\int_T C(t,t')a^{(n)}(t')\,\mathrm{d}t' = \lambda^{(n)}a^{(n)}(t)$$

where $C(t,t')$ is the two-point temporal correlation tensor defined as:

$$C(t,t') = \frac{1}{T}\int_\Omega u_i(\boldsymbol{x},t)u_i(\boldsymbol{x},t')\,\mathrm{d}\boldsymbol{x} = \frac{1}{T}\sum_{n=1}^{N_{\text{POD}}} a^{(n)}(t)a^{(n)*}(t')$$

- Eigenvectors are time dependent.
- No cross correlations.
- Linear independence of the snapshots assumed.
- Size: $N_{\text{POD}} = N_t$.

▷ Recall: For the classical POD, $N_{\text{POD}} = N_x \times n_c$

$\Longrightarrow$ Snapshot POD reduces drastically computational effort when $N_x \gg N_t$.

What is the typical situation?

- For experimental data:
  long time history with moderate spatial resolution

  $\Longrightarrow$ Two-point spatial correlation tensor $R_{ij}(\boldsymbol{x}, \boldsymbol{x}')$ well converged

  Exception: data sets obtained from Particle Image Velocimetry

- For numerical simulation data:
  much higher spatial resolution but a moderate time history

  $\Longrightarrow$ Two-point temporal correlation tensor $C(t, t')$ well converged

- Consequences:
  - Classical POD generally used with experimental data,
  - Snapshot POD generally used with numerical data.

1. Each space-time realization $u_i(\boldsymbol{x}, t)$ can be expanded into orthogonal eigenfunctions $\Phi_i^{(n)}(\boldsymbol{x})$ with uncorrelated coefficients $a^{(n)}(t)$:

$$u_i(\boldsymbol{x}, t) = \sum_{n=1}^{N_{\text{POD}}} a^{(n)}(t) \Phi_i^{(n)}(\boldsymbol{x}).$$

2. Spatial modes $\boldsymbol{\Phi}^{(n)}(\boldsymbol{x})$ are orthonormal:

$$\left(\boldsymbol{\Phi}^{(n)}, \boldsymbol{\Phi}^{(m)}\right)_{\Omega} = \int_{\Omega} \boldsymbol{\Phi}^{(n)}(\boldsymbol{x}) \cdot \boldsymbol{\Phi}^{(m)}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} = \delta_{nm}.$$

3. Temporal modes $a^{(n)}(t)$ are orthogonal:

$$\frac{1}{T} \int_{T} a^{(n)}(t) a^{(m)*}(t) \, \mathrm{d}t = \lambda^{(n)} \delta_{nm}.$$

- Spatial basis functions $\Phi_i^{(n)}(\boldsymbol{x})$ can be estimated as:

$$\Phi_i^{(n)}(\boldsymbol{x}) = \frac{1}{T\,\lambda^{(n)}} \int_T u_i(\boldsymbol{x}, t) a^{(n)*}(t)\,\mathrm{d}t$$

i.e. as a linear combination of instantaneous velocity fields.
$\Longrightarrow \Phi_i^{(n)}(\boldsymbol{x})$ possess all the properties of $u_i(\boldsymbol{x}, t)$ that can be written as linear and homogeneous equations.

- Ex: for an incompressible flow

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 \Longrightarrow \boldsymbol{\nabla} \cdot \boldsymbol{\Phi}^{(n)} = 0 \quad \forall n = 1, \cdots, N_{\text{POD}}$$

- Ex: boundary conditions

If they are homogeneous, then they are satisfied by each of the eigenfunctions individually, else use of specific methods.

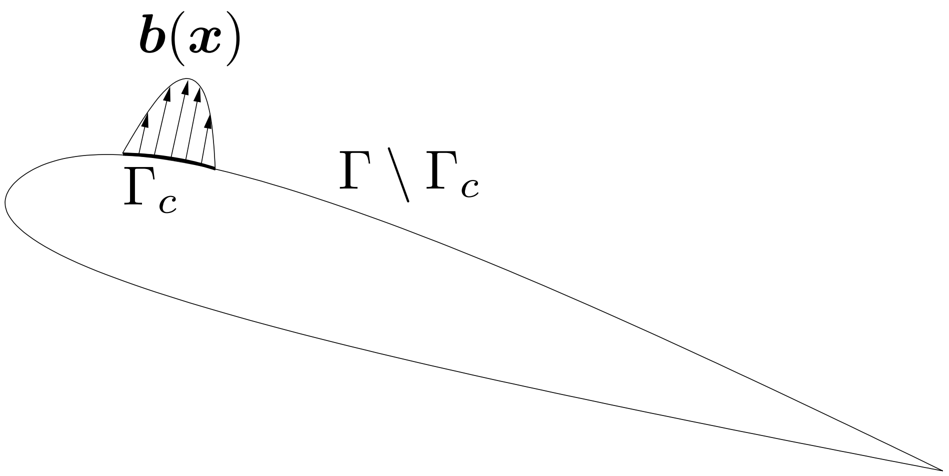
▷ Navier-Stokes equations written symbolically as: $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{x}, t)$ with $\boldsymbol{x} \in \Omega$ and $t \geq 0$

$$\frac{\partial \boldsymbol{u}}{\partial t} = \boldsymbol{f}(\boldsymbol{u}, P)$$

$$\boldsymbol{u}(\boldsymbol{x},\, t=0) = \boldsymbol{u}_0(\boldsymbol{x}) \quad (I.C.)$$

$$\boldsymbol{u}(\boldsymbol{x},\, t) = \gamma(t)\boldsymbol{b}(\boldsymbol{x}) \quad \text{for } \boldsymbol{x} \in \Gamma_c, \quad (B.C.)$$

$$\boldsymbol{u}(\boldsymbol{x},\, t) = \boldsymbol{h}(\boldsymbol{x}) \quad \text{for } \boldsymbol{x} \in \Gamma \setminus \Gamma_c \quad (B.C.).$$



▷ B.C. independent of time, *i.e.* $\boldsymbol{u}(\boldsymbol{x},\, t) = \boldsymbol{u}_{\text{BC}}(\boldsymbol{x})$ 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
- $\boldsymbol{u}_{\text{POD}}(\boldsymbol{x},\, t) = \boldsymbol{u}(\boldsymbol{x},\, t) - \boldsymbol{u_m}(\boldsymbol{x})$ verify homogeneous B.C. i.e.

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

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

▷ B.C. dependent of time, *i.e.* $\boldsymbol{u}(\boldsymbol{x},\,t)=\boldsymbol{u}_{\text{BC}}(\boldsymbol{x},t)$ 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)
- $\mathcal{U}'=$ $\{\boldsymbol{u}(\boldsymbol{x},\,t_1)-\gamma(t_1)\boldsymbol{u_c}(\boldsymbol{x})-\boldsymbol{u_m}(\boldsymbol{x}),\,\cdots,\,\boldsymbol{u}(\boldsymbol{x},\,t_{N_t})-\gamma(t_{N_t})\boldsymbol{u_c}(\boldsymbol{x})-\boldsymbol{u_m}(\boldsymbol{x})\}$
- $\boldsymbol{u}(\boldsymbol{x},\,t)=\boldsymbol{u_m}(\boldsymbol{x})+\gamma(t)\boldsymbol{u_c}(\boldsymbol{x})+\sum_{i=1}^{N_{\text{POD}}}a_i(t)\boldsymbol{\Phi}_i(\boldsymbol{x})$ where

$$\boldsymbol{u_c}(\boldsymbol{x})=\boldsymbol{b}(\boldsymbol{x}) \quad \text{on } \Gamma_c \text{ and}$$

$$\boldsymbol{u_c}(\boldsymbol{x})=\boldsymbol{0} \quad \text{on } \Gamma\setminus\Gamma_c.$$

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

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

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

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

- 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

$$[\boldsymbol{a}]_\Gamma = \int_\Gamma \boldsymbol{a}\cdot\boldsymbol{n}\,\mathrm{d}\boldsymbol{x} \qquad \text{and}$$

$$(\overline{\overline{A}},\ \overline{\overline{B}})_\Omega = \int_\Omega \overline{\overline{A}} : \overline{\overline{B}}\, d\Omega = \sum_{i,\,j}\int_\Omega A_{ij}B_{ji}\,\mathrm{d}\boldsymbol{x}.$$

POD Reduced-order model

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

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

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

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

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

$\mathcal{A}_i$, $\mathcal{B}_{ij}$, $\mathcal{C}_{ijk}$, $\mathcal{D}_i$, $\mathcal{E}_i$, $\mathcal{F}_{ij}$ 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

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

$$\mathcal{B}_{ij} = -\left(\boldsymbol{\Phi}^{(i)}, (\boldsymbol{u_m}\cdot\boldsymbol{\nabla})\,\boldsymbol{\Phi}^{(j)}\right)_\Omega - \left(\boldsymbol{\Phi}^{(i)}, \left(\boldsymbol{\Phi}^{(j)}\cdot\boldsymbol{\nabla}\right)\boldsymbol{u_m}\right)_\Omega$$

$$-\frac{1}{\text{Re}}\left(\boldsymbol{\nabla}\boldsymbol{\Phi}^{(i)}, \boldsymbol{\nabla}\boldsymbol{\Phi}^{(j)}\right)_\Omega + \frac{1}{\text{Re}}\left[\boldsymbol{\Phi}^{(i)}\,\boldsymbol{\nabla}\boldsymbol{\Phi}^{(j)}\right]_\Gamma$$

$$\mathcal{C}_{ijk} = -\left(\boldsymbol{\Phi}^{(i)}, \left(\boldsymbol{\Phi}^{(j)}\cdot\boldsymbol{\nabla}\right)\boldsymbol{\Phi}^{(k)}\right)_\Omega$$

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

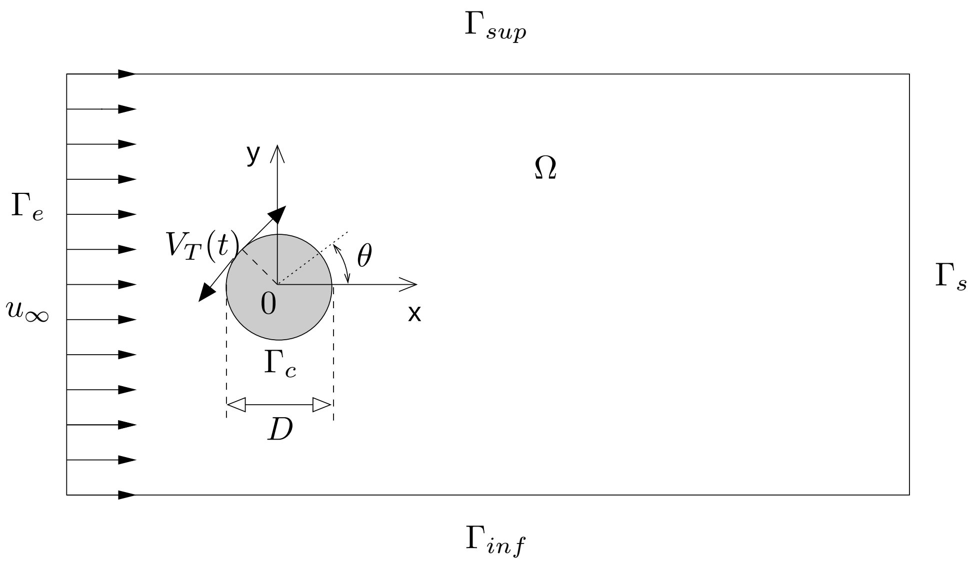
$$\mathcal{E}_i = -\left(\mathbf{\Phi}^{(i)}, (\boldsymbol{u_m} \cdot \boldsymbol{\nabla})\, \boldsymbol{u_c}\right)_\Omega - \left(\mathbf{\Phi}^{(i)}, (\boldsymbol{u_c} \cdot \boldsymbol{\nabla})\, \boldsymbol{u_m}\right)_\Omega$$
$$-\frac{1}{\mathrm{Re}}\left(\boldsymbol{\nabla}\mathbf{\Phi}^{(i)},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

$$\mathcal{F}_{ij} = -\left(\mathbf{\Phi}^{(i)}, \left(\mathbf{\Phi}^{(j)} \cdot \boldsymbol{\nabla}\right) \boldsymbol{u_c}\right)_\Omega - \left(\mathbf{\Phi}^{(i)}, (\boldsymbol{u_c} \cdot \boldsymbol{\nabla})\, \mathbf{\Phi}^{(j)}\right)_\Omega$$

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

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

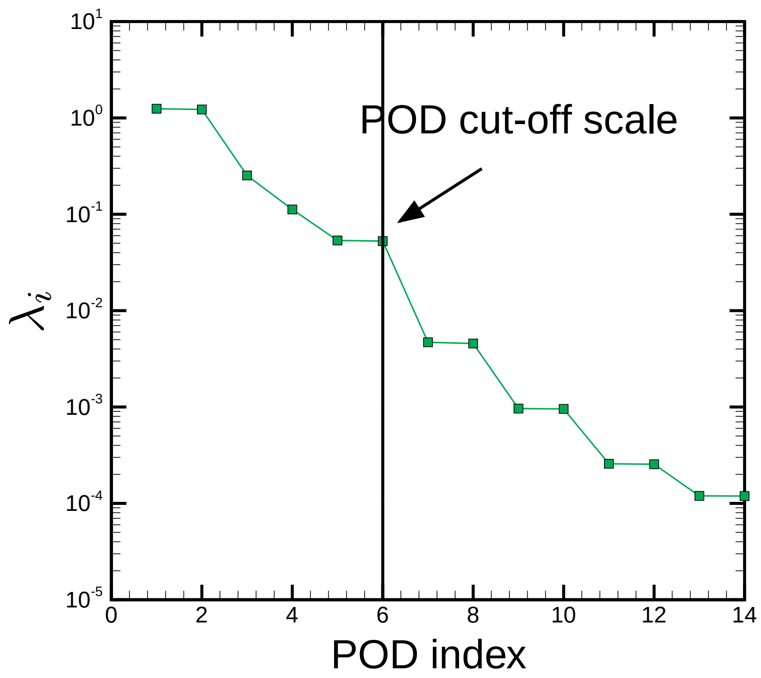
$$\gamma(t) = \frac{V_T}{u_\infty} = A \sin(2\pi St_f t)$$

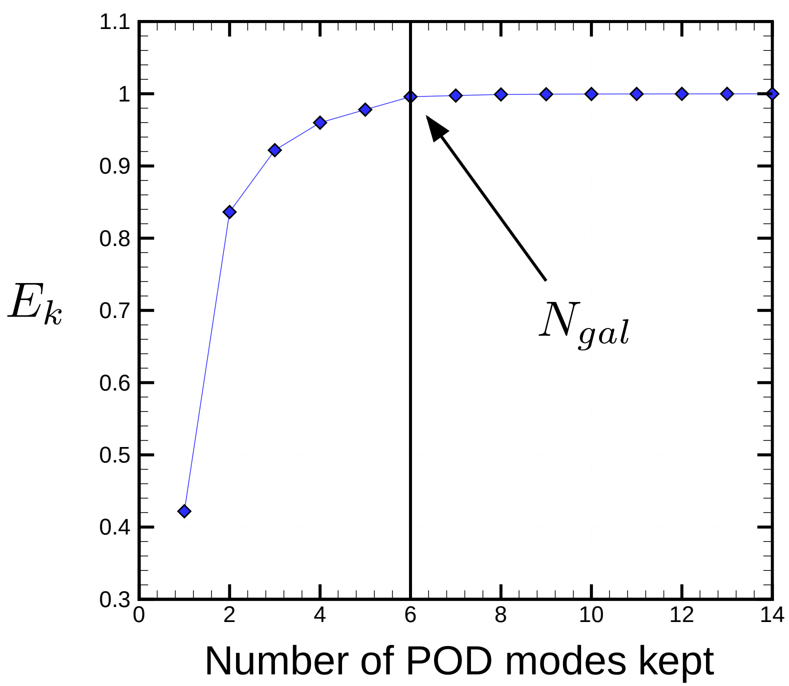


POD of the controlled wake flow ($\gamma \neq 0$) $A = 2$ and $St_f = 0,5$

- 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





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

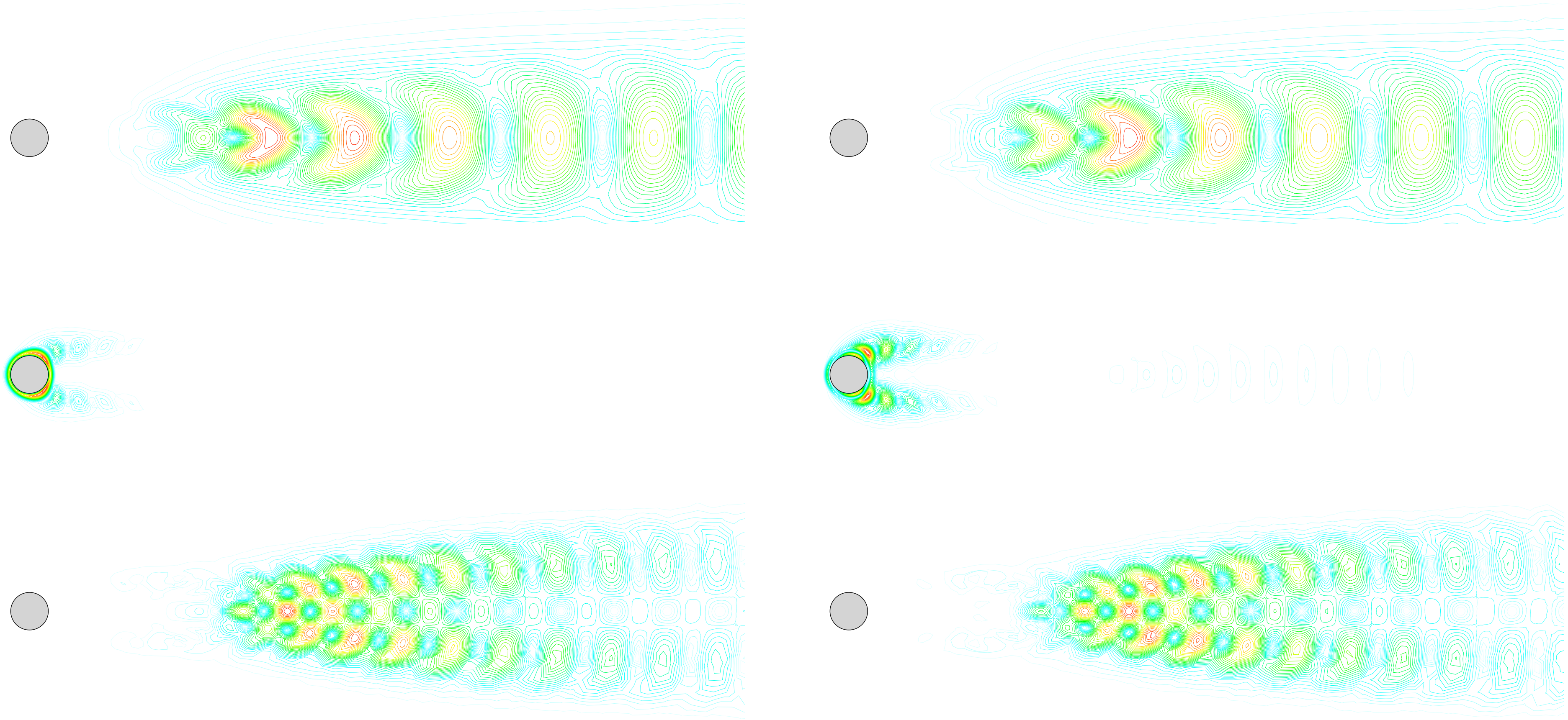


Fig. : *Iso-values of the first 6 POD modes*

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

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

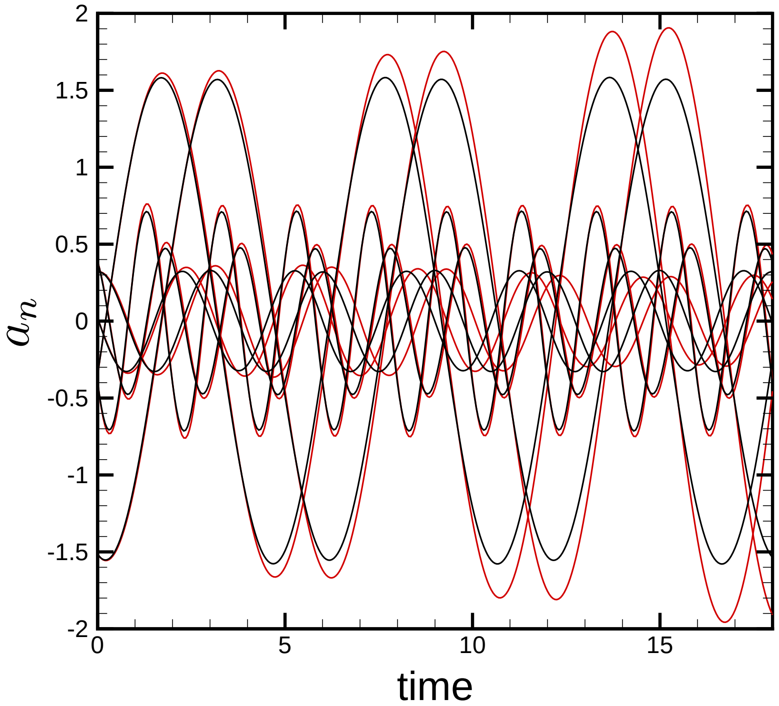


Fig. : *Time evolution of the first 6 POD modes ($A = 2$ and $St_f = 0,5$).*

▷ 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)