UNIVERSIDAD
POLITÉCNICA
DE MADRID

etsiae
UPM

Acceleration of data-driven modal decomposition using unsupervised machine learning techniques

Jesús Garicano-Mena
Dpto. Matemática Aplicada - ETSI Aeronáutica y del Espacio (UPM)
Center for Computational Simulation

Objectives of this talk

1) Describe classical Dynamic Mode Decomposition (DMD)

2) DMD as one of many data-driven modal decomposition techniques

3) DMD as a matrix factorization technique

4) Accelerating DMD methods

Objectives of this talk

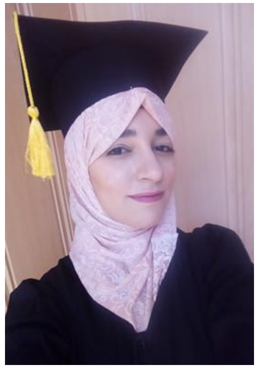
1) Describe classical Dynamic Mode Decomposition (DMD)

2) DMD as one of many data-driven modal decomposition techniques

3) DMD as a matrix factorization technique

4) Accelerating DMD methods

B. Begiashvili *et al.,* Data-driven modal decomposition methods as feature detection techniques for flow problems: a critical assessment, Phys. Fluids 35, 041301 (2023).
Li, et al., Dynamic Mode Decomposition Analysis of Spatially Agglomerated Flow Databases, Energies, vol. 13 (9), 2020.



N. Groun

B. Li

S. Le Clainche

B. Begiashvili

E. Valero

J. Garicano Mena

B. Begiashvili *et al.,* Data-driven modal decomposition methods as feature detection techniques for flow problems: a critical assessment, Phys. Fluids 35, 041301 (2023).

Li, et al., Dynamic Mode Decomposition Analysis of Spatially Agglomerated Flow Databases, Energies, vol. 13 (9), 2020.

Introduction

Dynamic Mode Decomposition

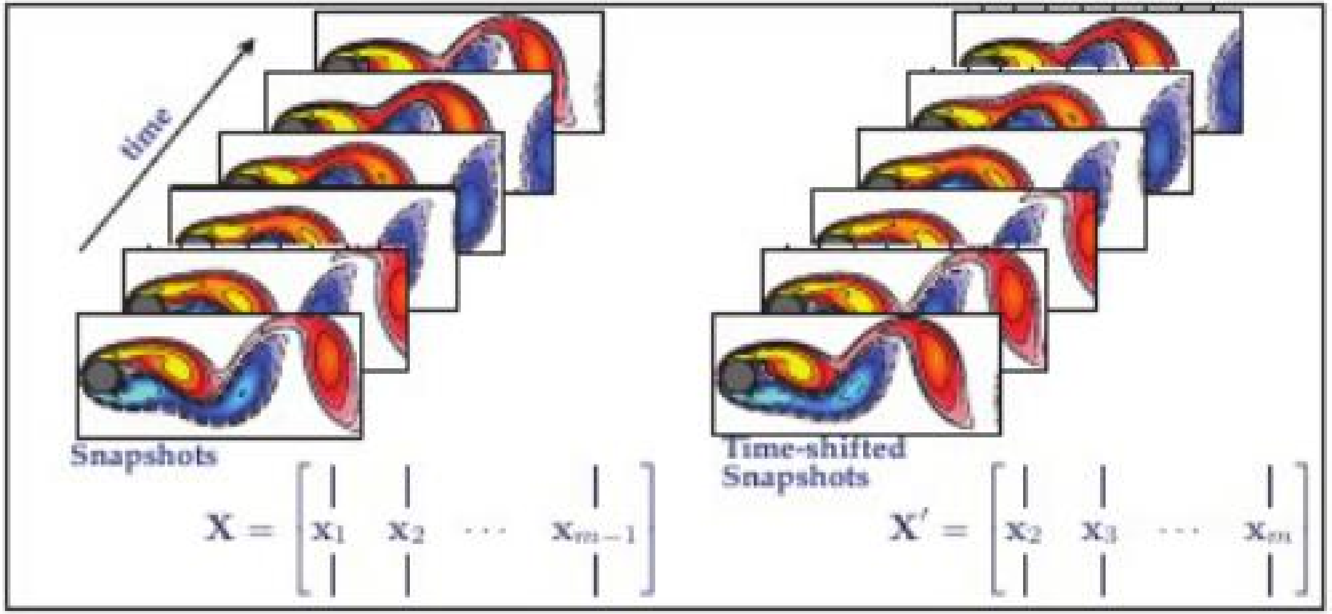
Alleviating the computational cost

Conclusions

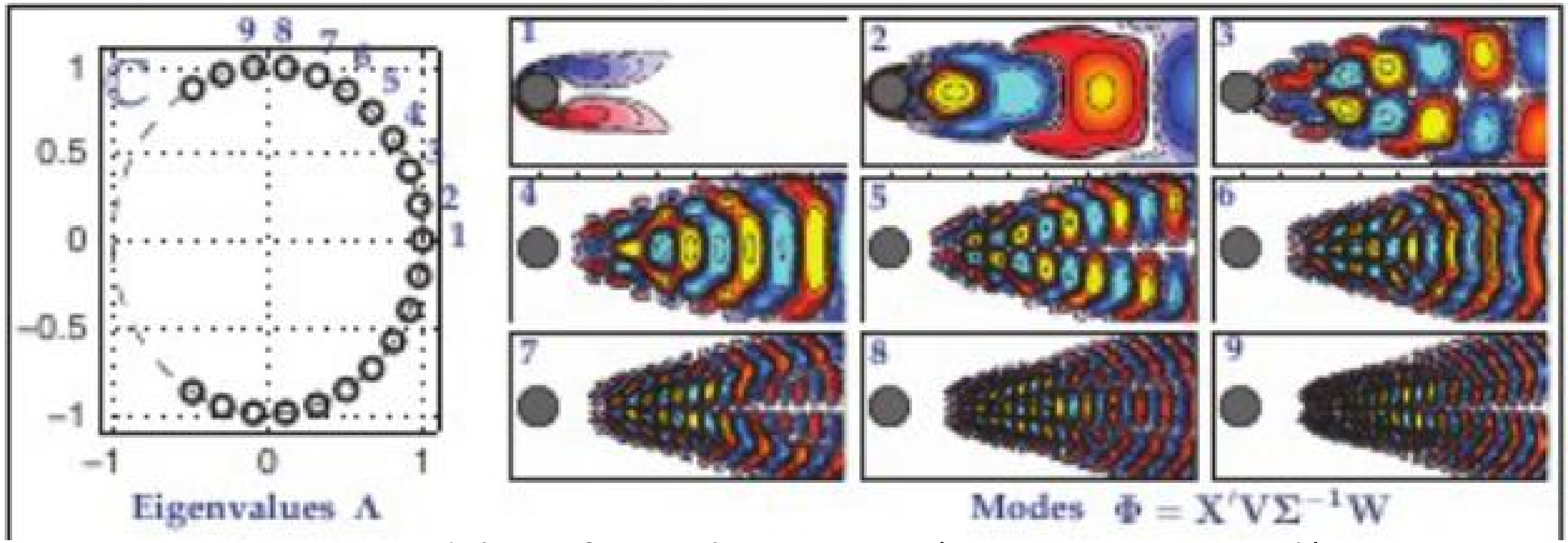
Introduction:

data-driven modal decomposition techniques

Dynamic Mode Decomposition

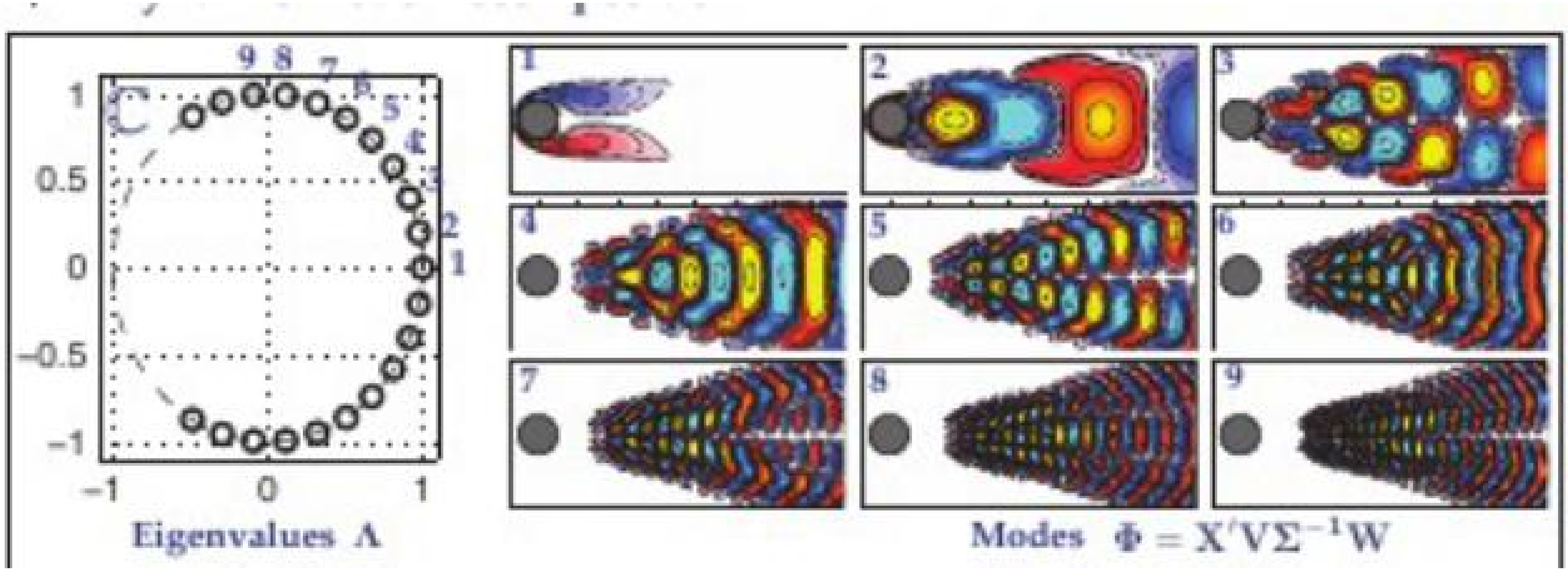


Data Analysis

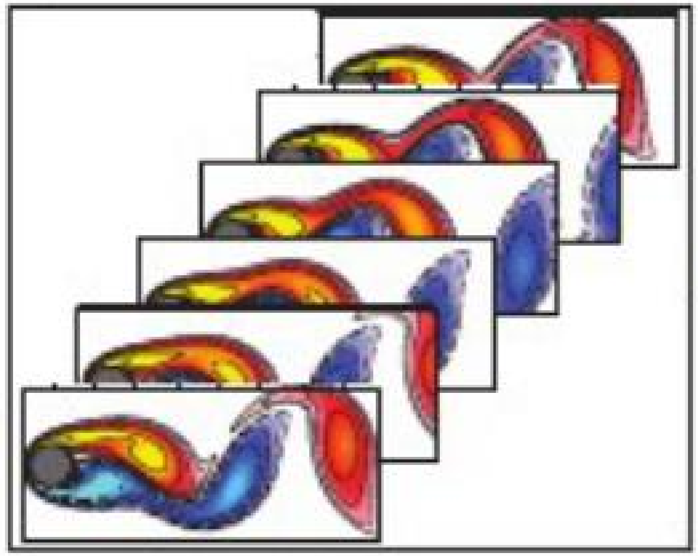


Credit: Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton, Kutz, *et al.*)

Dynamic Mode Decomposition



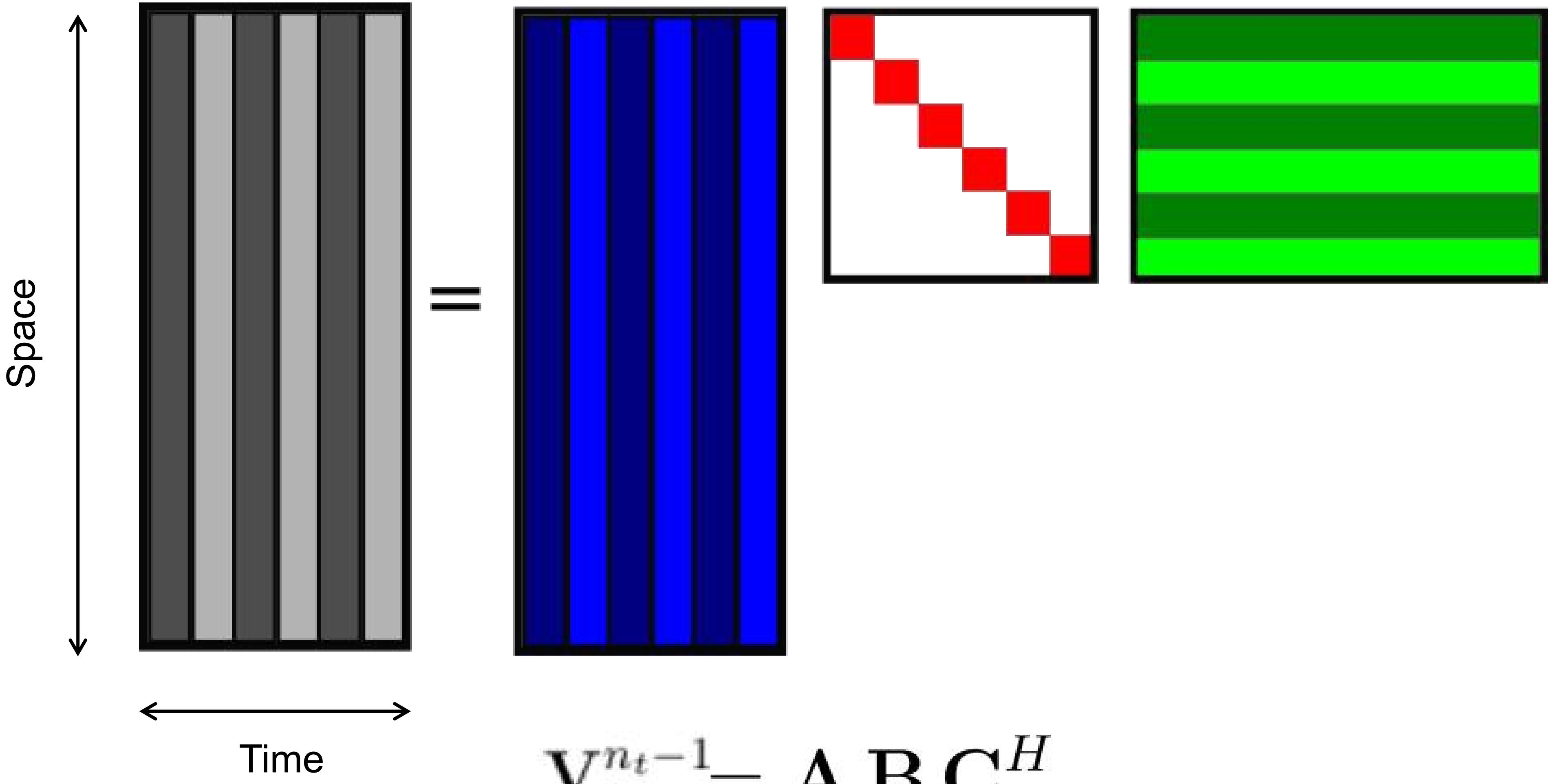
Synthesis (Reconstruction)



$$\mathbf{X} \approx \overset{\text{modes}}{\begin{bmatrix} | & | & \cdots \\ \phi_1 & \phi_2 & \cdots \\ | & | & \cdots \end{bmatrix}} \overset{\text{amplitudes}}{\begin{bmatrix} b_1 & 0 & \cdots \\ 0 & b_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}} \overset{\text{dynamics}}{\begin{bmatrix} 1 & \lambda_1 & \cdots & \lambda_1^{m-2} \\ 1 & \lambda_2 & \cdots & \lambda_2^{m-2} \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix}}$$

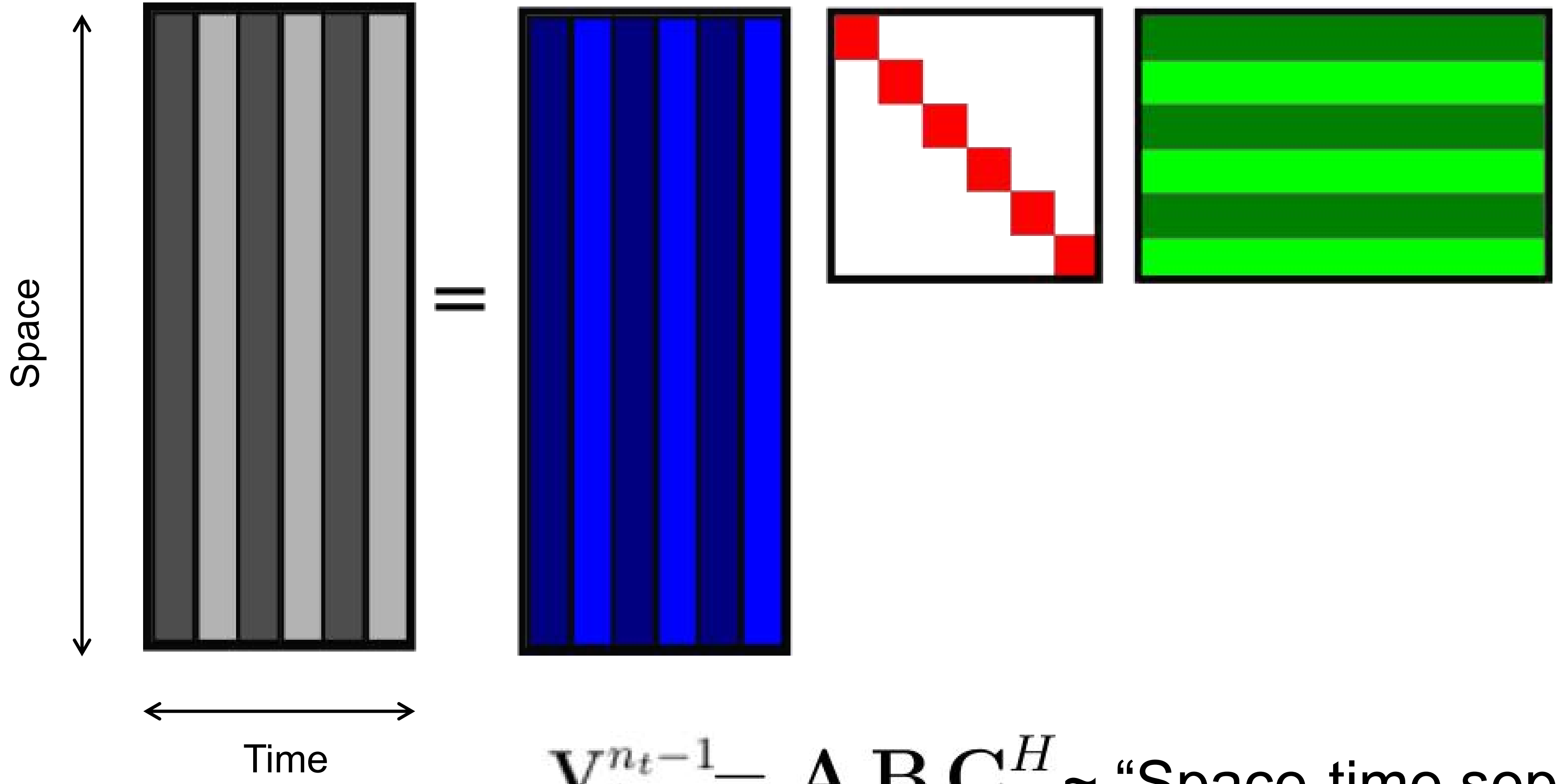
Credit: Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton, Kutz, *et al.*)

Matrix factorization techniques



$$\mathbf{V}_1^{n_t-1} = \mathbf{A}\,\mathbf{B}\,\mathbf{C}^H$$

Matrix factorization techniques



$$\mathbf{V}_1^{n_t-1} = \mathbf{A}\,\mathbf{B}\,\mathbf{C}^H$$ ~ "Space-time separation"

Dynamic Mode Decomposition

Temporal data sequence acquired at a fixed sampling frequency $1/\Delta t$

Arrange data in a (typically) tall and skinny matrix

$$\mathbf{V}_1^{n_t} = \begin{bmatrix} \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \underbrace{\blacksquare}_{\mathbf{v}(t_0)} & \underbrace{\blacksquare}_{\mathbf{v}(t_1)} & \cdots & \underbrace{\blacksquare}_{\mathbf{v}(t_{n_t-1})} & \underbrace{\blacksquare}_{\mathbf{v}(t_{n_t})} \end{bmatrix} \in \mathbb{R}^{n_p \times n_t}$$

P.J. Schmid, J. Fluid Mech., vol. 656, 2010

Dynamic Mode Decomposition

Temporal data sequence acquired at a fixed sampling frequency $1/\Delta t$

Arrange data in a (typically) tall and skinny matrix

$$\mathbf{V}_1^{n_t} = \begin{bmatrix} \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \underbrace{\blacksquare}_{\mathbf{v}(t_0)} & \underbrace{\blacksquare}_{\mathbf{v}(t_1)} & \cdots & \underbrace{\blacksquare}_{\mathbf{v}(t_{n_t-1})} & \underbrace{\blacksquare}_{\mathbf{v}(t_{n_t})} \end{bmatrix} \in \mathbb{R}^{n_p \times n_t} \quad \boxed{n_p \gg n_t}$$

P.J. Schmid, J. Fluid Mech., vol. 656, 2010

Dynamic Mode Decomposition

Temporal data sequence acquired at a fixed sampling frequency $1/\Delta t$

Arrange data in a (typically) $\boxed{\text{tall and skinny matrix}}$

$$
\mathbf{V}_1^{n_t} = \begin{array}{c}
\begin{bmatrix}
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\
\blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare
\end{bmatrix} \\
\begin{array}{ccccc}
\mathbf{v}(t_0) & \mathbf{v}(t_1) & & \mathbf{v}(t_{n_t-1}) & \mathbf{v}(t_{n_t})
\end{array}
\end{array}
\in \mathbb{R}^{n_p \times n_t} \quad \boxed{n_p \gg n_t}
$$

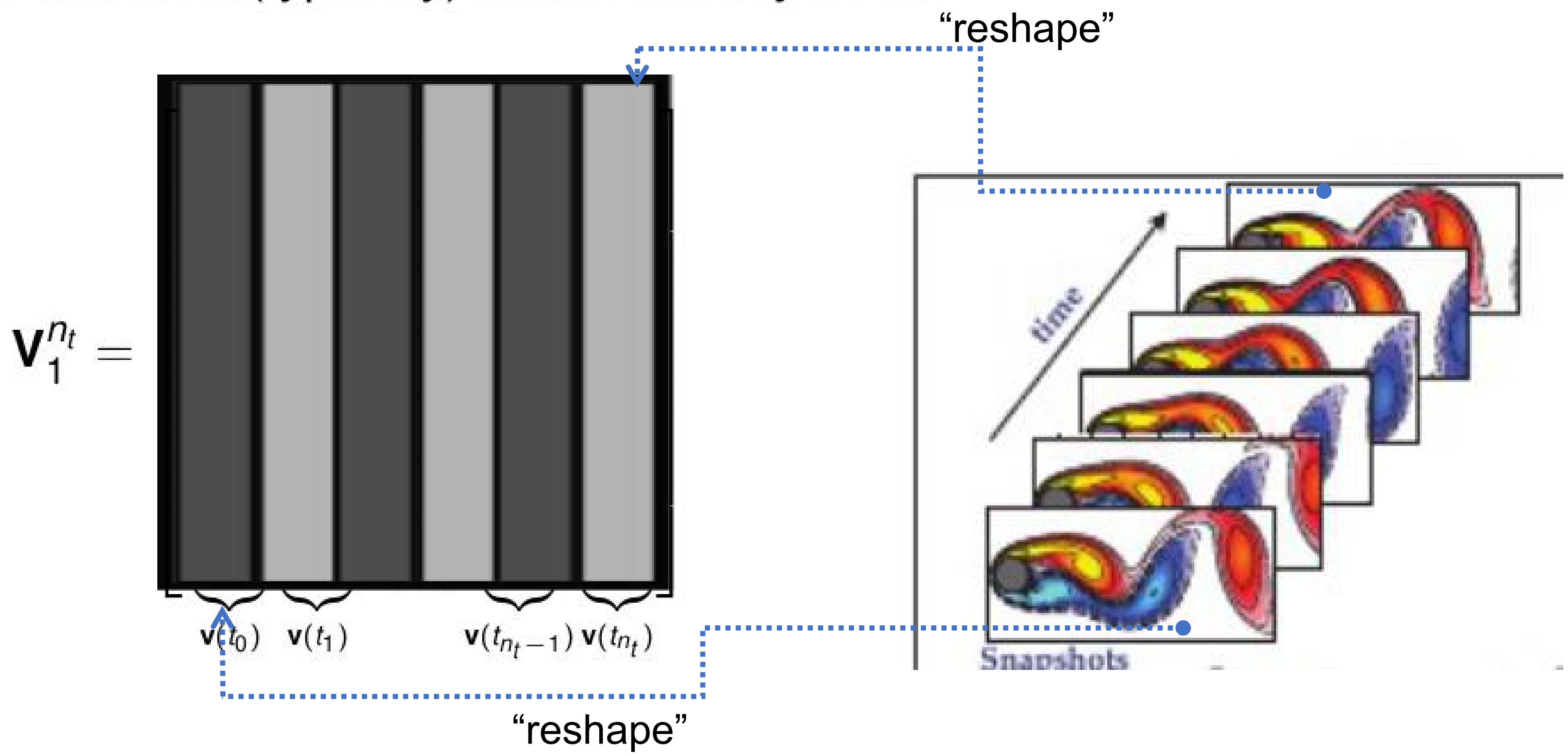
$$n_p = n_s \times n_{vars}$$

P.J. Schmid, J. Fluid Mech., vol. 656, 2010

Dynamic Mode Decomposition

Temporal data sequence acquired at a fixed sampling frequency $1/\Delta t$

Arrange data in a (typically) tall and skinny matrix



P.J. Schmid, J. Fluid Mech., vol. 656, 2010

Dynamic Mode Decomposition

From the complete data set

$$\mathbf{V}_1^{n_t} = [\mathbf{v}_1,\ \mathbf{v}_2,\ \ldots,\ \mathbf{v}_{n_t-1},\ \mathbf{v}_{n_t}] \in \mathbb{R}^{n_p \times n_t}$$

Identify data subsequences as:

$$\mathbf{V}_1^{n_t-1} = [\mathbf{v}_1,\ \mathbf{v}_2,\ \ldots,\ \mathbf{v}_{n_t-1}] \equiv \mathbf{X} \quad \text{and} \quad \mathbf{V}_2^{n_t} = [\mathbf{v}_2,\ \mathbf{v}_3,\ \ldots,\ \mathbf{v}_{n_t-1},\ \mathbf{v}_{n_t}] \equiv \mathbf{Y}$$

Dynamic Mode Decomposition

From the complete data set

$$\mathbf{V}_1^{n_t} = [\mathbf{v}_1,\ \mathbf{v}_2,\ \ldots,\ \mathbf{v}_{n_t-1},\ \mathbf{v}_{n_t}] \in \mathbb{R}^{n_p \times n_t}$$

Identify data subsequences as:

$$\mathbf{V}_1^{n_t-1} = [\mathbf{v}_1,\ \mathbf{v}_2,\ \ldots,\ \mathbf{v}_{n_t-1}] \equiv \mathbf{X} \quad \text{and} \quad \mathbf{V}_2^{n_t} = [\mathbf{v}_2,\ \mathbf{v}_3,\ \ldots,\ \mathbf{v}_{n_t-1},\ \mathbf{v}_{n_t}] \equiv \mathbf{Y}$$

Assume a linear relationship:

$$\mathbf{v}(t_{k+1}) = \mathcal{A}\,\mathbf{v}(t_k)$$

Dynamic Mode Decomposition

From the complete data set

$$\mathbf{V}_1^{n_t} = [\mathbf{v}_1, \, \mathbf{v}_2, \, \ldots, \, \mathbf{v}_{n_t-1}, \, \mathbf{v}_{n_t}] \in \mathbb{R}^{n_p \times n_t}$$

Identify data subsequences as:

$$\mathbf{V}_1^{n_t-1} = [\mathbf{v}_1, \, \mathbf{v}_2, \, \ldots, \, \mathbf{v}_{n_t-1}] \equiv \mathbf{X} \quad \text{and} \quad \mathbf{V}_2^{n_t} = [\mathbf{v}_2, \, \mathbf{v}_3, \, \ldots, \, \mathbf{v}_{n_t-1}, \, \mathbf{v}_{n_t}] \equiv \mathbf{Y}$$

Assume a linear relationship:

$$\mathbf{v}(t_{k+1}) = \mathcal{A}\,\mathbf{v}(t_k)$$

Koopman assumption

Dynamic Mode Decomposition

From the complete data set

$$\mathbf{V}_1^{n_t} = [\mathbf{v}_1,\, \mathbf{v}_2,\, \ldots,\, \mathbf{v}_{n_t-1},\, \mathbf{v}_{n_t}] \in \mathbb{R}^{n_p \times n_t}$$

Identify data subsequences as:

$$\mathbf{V}_1^{n_t-1} = [\mathbf{v}_1,\, \mathbf{v}_2,\, \ldots,\, \mathbf{v}_{n_t-1}] \equiv \mathbf{X} \quad \text{and} \quad \mathbf{V}_2^{n_t} = [\mathbf{v}_2,\, \mathbf{v}_3,\, \ldots,\, \mathbf{v}_{n_t-1},\, \mathbf{v}_{n_t}] \equiv \mathbf{Y}$$

Assume a linear relationship:

$$\mathbf{V}_2^{n_t} = \mathcal{A}\, \mathbf{V}_1^{n_t-1}$$

$$\mathcal{A} \in \mathbb{R}^{n_p \times n_p}$$

Dynamic Mode Decomposition

How does one identify $\mathcal{A}$? Well, you don't need to.

$$\mathbf{V}_1^{n_t-1} \equiv \mathbf{X} \stackrel{SVD}{=} \mathbf{L}_0\, \mathbf{S}_0\, \mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j}\, \mathbf{l}_{0,j} \cdot \mathbf{r}_{0,j}^T$$

The columns of $L_0 \Rightarrow$ Left Singular Vectors

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

How does one identify $\mathcal{A}$? Well, you don't need to.

$$\mathbf{V}_1^{n_t-1} \equiv \mathbf{X} \stackrel{SVD}{=} \mathbf{L}_0\,\mathbf{S}_0\,\mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j}\,\mathbf{l}_{0,j} \cdot \mathbf{r}_{0,j}^T$$

The columns of $L_0 \Rightarrow$ Left Singular Vectors (some consider these as POD modes)

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

How does one identify $\mathcal{A}$? Well, you don't need to.

$$\mathbf{V}_1^{n_t-1} \equiv \mathbf{X} \stackrel{SVD}{=} \mathbf{L}_0\, \mathbf{S}_0\, \mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j}\, \mathbf{l}_{0,j} \cdot \mathbf{r}_{0,j}^T$$

The columns of $L_0 \Rightarrow$ Left Singular Vectors (some consider these as POD modes ... if the temporally averaged state is substracted)

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

How does one identify $\mathcal{A}$? Well, you don't need to.

$$\mathbf{V}_1^{n_t-1} \equiv \mathbf{X} \overset{SVD}{=} \mathbf{L}_0\, \mathbf{S}_0\, \mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j}\, \mathbf{l}_{0,j} \cdot \mathbf{r}_{0,j}^T$$

The columns of $L_0 \Rightarrow$ Left Singular Vectors

Project into the reduced Left Singular Vector (sub)space and manipulate

$$\mathbf{L}_0\, \mathbf{Y}\, \mathbf{R}_0\, \mathbf{S}_0^{-1} = \mathbf{L}_0^T\, \mathcal{A}\, \mathbf{L}_0 \equiv \widetilde{\mathcal{A}} \in \mathbb{R}^{r \times r}$$

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

How does one identify $\mathcal{A}$? Well, you don't need to.

$$\mathbf{V}_1^{n_t-1} \equiv \mathbf{X} \overset{SVD}{=} \mathbf{L}_0\, \mathbf{S}_0\, \mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j}\, \mathbf{l}_{0,j} \cdot \mathbf{r}_{0,j}^T$$

The columns of L_0 ⇨ Left Singular Vectors

Project into the reduced Left Singular Vector (sub)space and manipulate

$$\mathbf{L}_0\, \mathbf{Y}\, \mathbf{R}_0\, \mathbf{S}_0^{-1} = \mathbf{L}_0^T\, \mathcal{A}\, \mathbf{L}_0 \equiv \widetilde{\mathcal{A}} \in \mathbb{R}^{r\times r}$$

Assumption: $\tilde{\mathcal{A}}$ conveys most of the information codified in $\mathcal{A}$

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

How does one identify $\mathcal{A}$? Well, you don't need to.

$$\mathbf{V}_1^{n_t-1} \equiv \mathbf{X} \stackrel{SVD}{=} \mathbf{L}_0\, \mathbf{S}_0\, \mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j}\, \mathbf{l}_{0,j} \cdot \mathbf{r}_{0,j}^T$$

The columns of L_0 ⇨ Left Singular Vectors

Project into the reduced Left Singular Vector (sub)space and manipulate

$$\mathbf{L}_0\, \mathbf{Y}\, \mathbf{R}_0\, \mathbf{S}_0^{-1} = \mathbf{L}_0^T\, \mathcal{A}\, \mathbf{L}_0 \equiv \widetilde{\mathcal{A}} \in \mathbb{R}^{r\times r}$$

Assumption: $\tilde{\mathcal{A}}$ conveys most of the information codified in $\mathcal{A}$,

but $r \leq n_t << n_p$

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

How does one identify $\mathcal{A}$? Well, you don't need to.

$$\mathbf{V}_1^{n_t-1} \equiv \mathbf{X} \overset{SVD}{=} \boxed{\mathbf{L}_0\, \mathbf{S}_0\, \mathbf{R}_0^T} = \boxed{\sum_{j=1}^{r} s_{0,j}\, \mathbf{l}_{0,j} \cdot \mathbf{r}_{0,j}^T}$$

The columns of $L_0 \Rightarrow$ Left Singular Vectors

Project into the reduced Left Singular Vector (sub)space and manipulate

$$\mathbf{L}_0\, \mathbf{Y}\, \mathbf{R}_0\, \mathbf{S}_0^{-1} = \mathbf{L}_0^T\, \mathcal{A}\, \mathbf{L}_0 \equiv \widetilde{\mathcal{A}} \in \mathbb{R}^{r \times r}$$

Assumption: $\tilde{\mathcal{A}}$ conveys most of the information codified in $\mathcal{A}$,

but $r \leq n_t << n_p$

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

Assumption $\tilde{A}$ conveys most of the information codified in $\mathcal{A}$

Dynamic modes and Ritz values are obtained via eigendecomposition

$$\widetilde{\mathcal{A}}\,\mathbf{w}_j = \mathbf{w}_j\;\mu_j$$

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

Assumption $\tilde{\mathcal{A}}$ conveys most of the information codified in $\mathcal{A}$

Dynamic modes and Ritz values are obtained via eigendecomposition

$$\widetilde{\mathcal{A}}\,\mathbf{W} = \mathbf{W}\,\mathbf{D}_{\mu}$$

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

Assumption $\tilde{A}$ conveys most of the information codified in $\mathcal{A}$

Dynamic modes and Ritz values are obtained via eigendecomposition

$$\widetilde{\mathcal{A}}\,\mathbf{W} = \mathbf{W}\,\mathbf{D}_{\mu}$$

Dynamic Modes are obtained projecting back into the LSV space

$$\mathbf{\Phi} = \mathbf{L}_0\,\mathbf{W}$$

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

Assumption $\tilde{A}$ conveys most of the information codified in $\mathcal{A}$

Dynamic modes and Ritz values are obtained via eigendecomposition

$$\widetilde{\mathcal{A}}\,\mathbf{W} = \mathbf{W}\,\mathbf{D}_{\mu}$$

Dynamic Modes are obtained projecting back into the LSV space

$$\mathbf{\Phi} = \mathbf{L}_0\,\mathbf{W}$$

Temporal interpolation/extrapolation is possible:

$$\mathbf{v}(\mathbf{r},t) = \sum_{j=1}^{r} \alpha_j\,\mathbf{\Phi}_j\,e^{\lambda_j\,t}$$ where $\lambda_j = \dfrac{\log(\mu_j)}{\Delta t}$

... once the α_j's are known.

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

Assumption $\tilde{\mathcal{A}}$ conveys most of the information codified in $\mathcal{A}$

Dynamic modes and Ritz values are obtained via eigendecomposition

$$\widetilde{\mathcal{A}}\,\mathbf{W} = \mathbf{W}\,\mathbf{D}_\mu$$

Dynamic Modes are obtained projecting back into the LSV space

$$\mathbf{\Phi} = \mathbf{L}_0\,\mathbf{W}$$

Temporal interpolation/extrapolation is possible:

$$\mathbf{v}(\mathbf{r},t) = \sum_{j=1}^{r} \alpha_j\,\mathbf{\Phi}_j\,e^{\lambda_j\,t} = \mathbf{\Phi}\,\mathbf{D}_\alpha\,\mathbf{V}_\lambda \text{ where } \lambda_j = \frac{\log(\mu_j)}{\Delta t}$$

... once the α_j's are known.

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

Assumption $\tilde{\mathcal{A}}$ conveys most of the information codified in $\mathcal{A}$

Dynamic modes and Ritz values are obtained via eigendecomposition

$$\widetilde{\mathcal{A}}\,\mathbf{W} = \mathbf{W}\,\mathbf{D}_\mu$$

Dynamic Modes are obtained projecting back into the LSV space

$$\mathbf{\Phi} = \mathbf{L}_0\,\mathbf{W}$$

Temporal interpolation/extrapolation is possible:

$$\mathbf{v}(\mathbf{r},t) = \boxed{\sum_{j=1}^{r} \alpha_j\,\mathbf{\Phi}_j\,e^{\lambda_j t}} = \boxed{\mathbf{\Phi}\,\mathbf{D}_\alpha\,\mathbf{V}_\lambda} \text{ where } \lambda_j = \frac{\log(\mu_j)}{\Delta t}$$

... once the α_j's are known.

P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

Dynamic Mode Decomposition

Assumption $\tilde{\mathcal{A}}$ conveys most of the information codified in $\mathcal{A}$

Dynamic modes and Ritz values are obtained via eigendecomposition

$$\widetilde{\mathcal{A}}\,\mathbf{W} = \mathbf{W}\,\mathbf{D}_\mu$$

Dynamic Modes are obtained projecting back into the LSV space

$$\boxed{\mathbf{\Phi} = \mathbf{L}_0\,\mathbf{W}}$$

Temporal interpolation/extrapolation is possible:

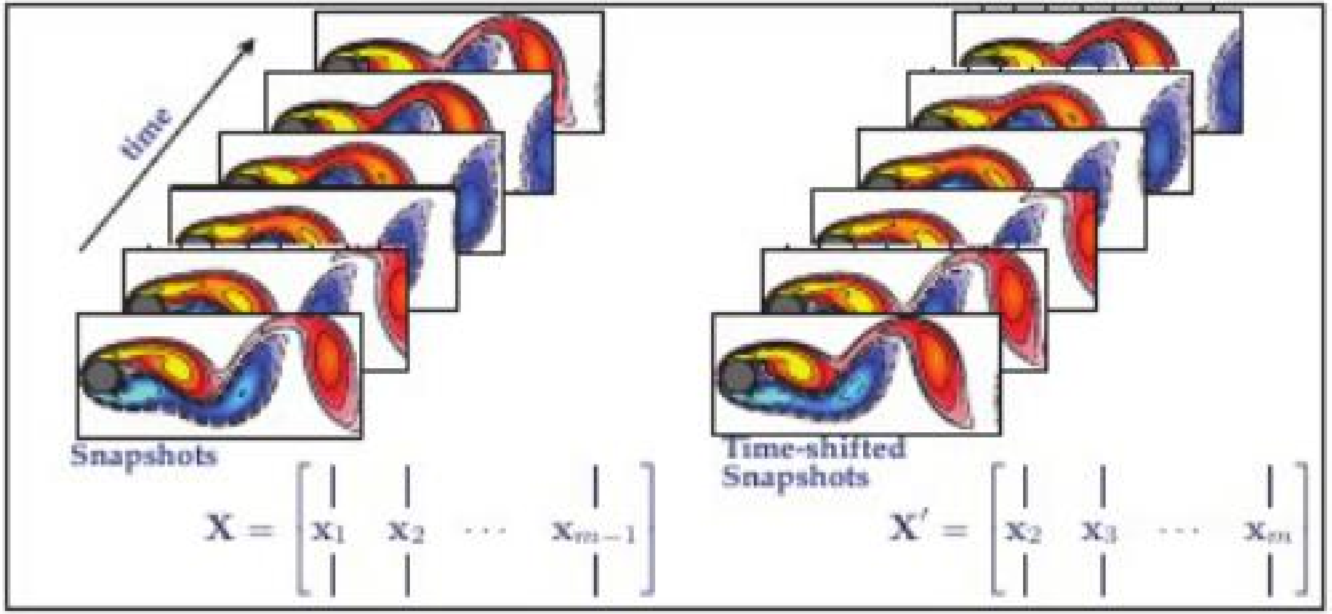
$$\mathbf{v}(\mathbf{r},t) = \boxed{\sum_{j=1}^{r} \alpha_j\,\mathbf{\Phi}_j\,e^{\lambda_j t}} = \boxed{\mathbf{\Phi}\,\mathbf{D}_\alpha\,\mathbf{V}_\lambda} \text{ where } \lambda_j = \frac{\log(\mu_j)}{\Delta t}$$

... once the α_j's are known.

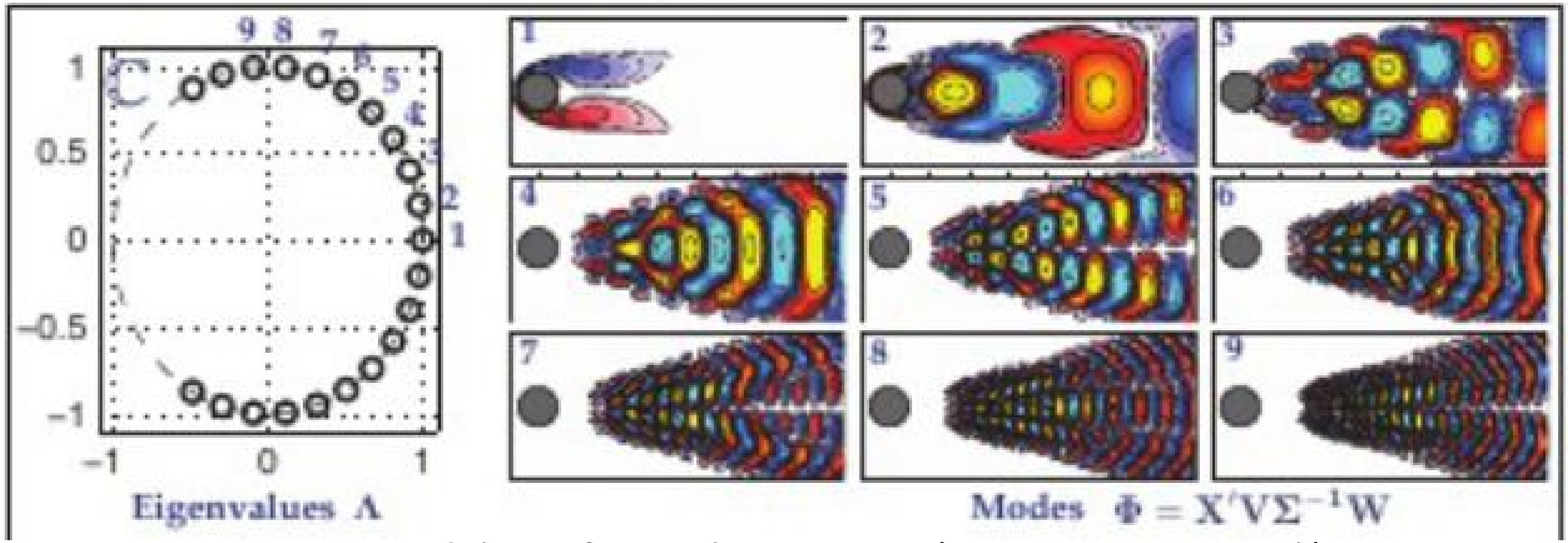
P.J. Schmid, Dynamic Mode Decomposition of numerical & experimental data, J. Fluid Mech., vol. 656, 2010.

An example

Dynamic Mode Decomposition

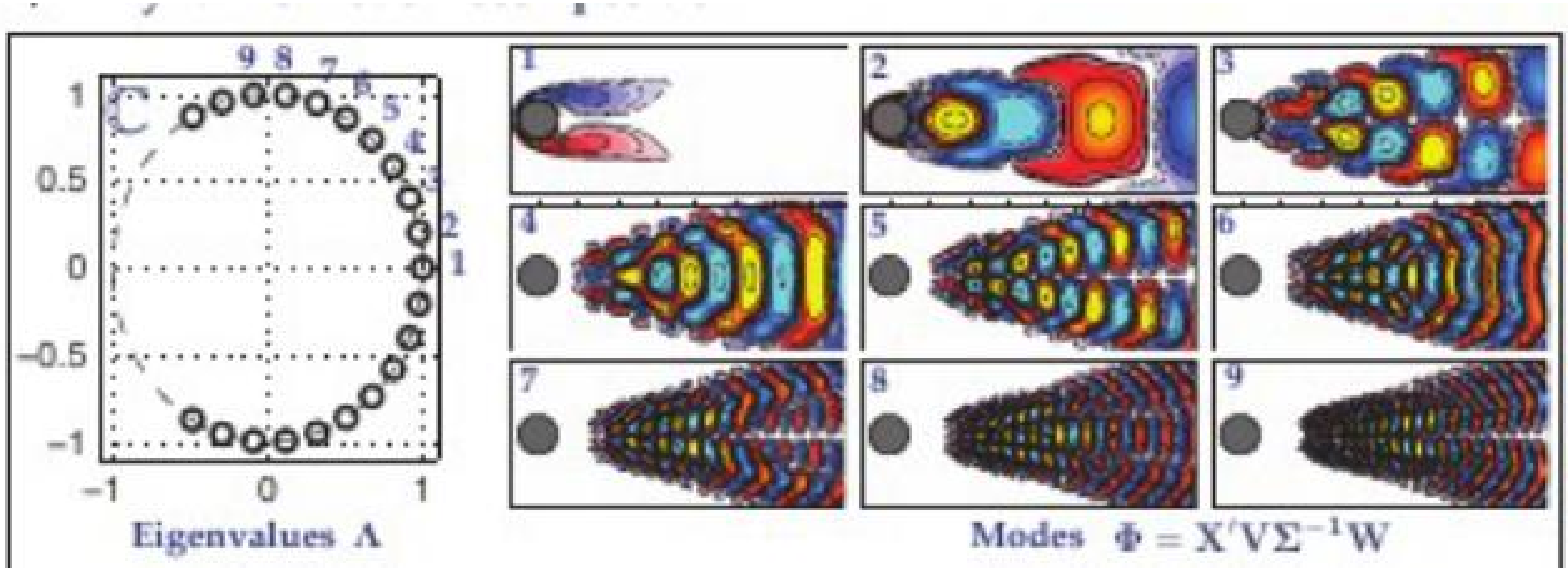


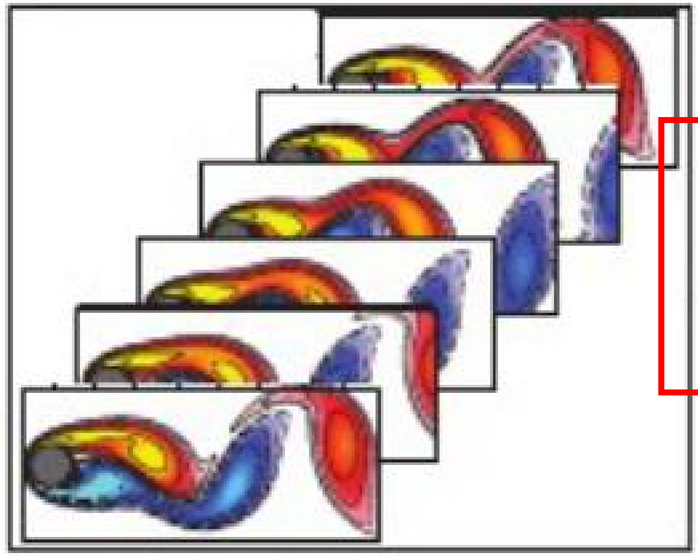
Data Analysis



Credit: Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton, Kutz, *et al.*)

Dynamic Mode Decomposition



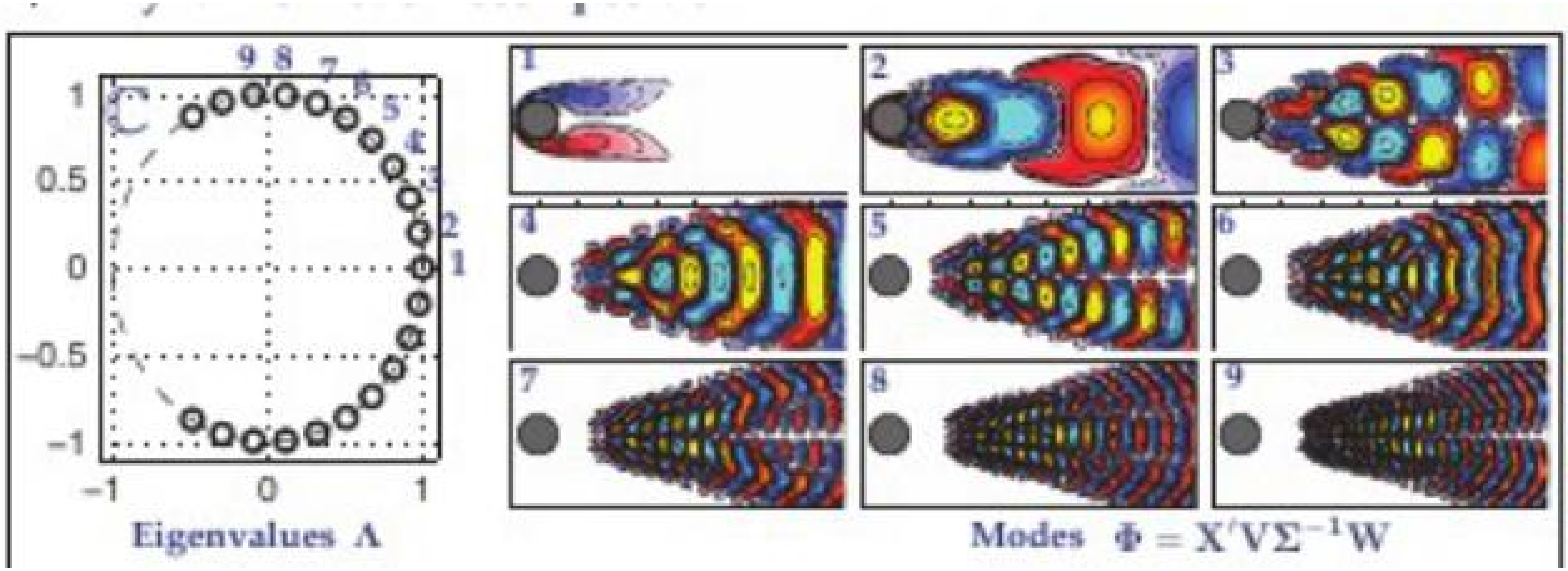


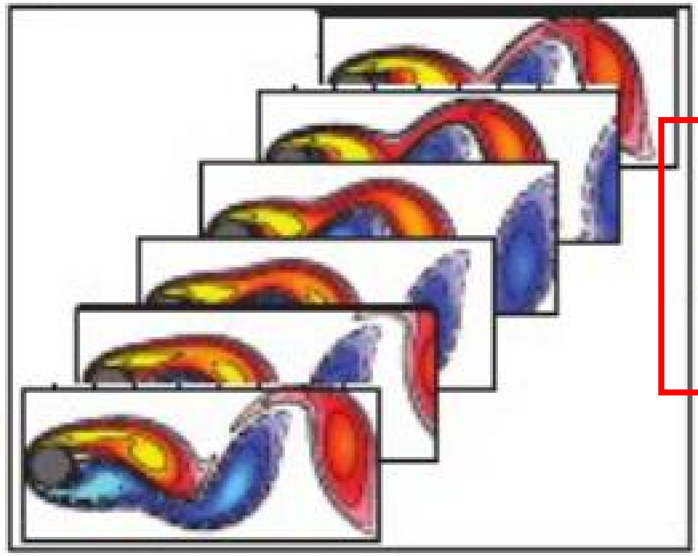
$$\begin{array}{cccc} & \text{modes} & \text{amplitudes} & \text{dynamics} \\ \mathbf{X} \approx & \begin{bmatrix} | & | & \cdots \\ \phi_1 & \phi_2 & \cdots \\ | & | & \cdots \end{bmatrix} & \begin{bmatrix} b_1 & 0 & \cdots \\ 0 & b_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} & \begin{bmatrix} 1 & \lambda_1 & \cdots & \lambda_1^{m-2} \\ 1 & \lambda_2 & \cdots & \lambda_2^{m-2} \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix} \end{array}$$

Synthesis (Reconstruction)

Credit: Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton & Kutz)

Dynamic Mode Decomposition





Synthesis (Reconstruction)

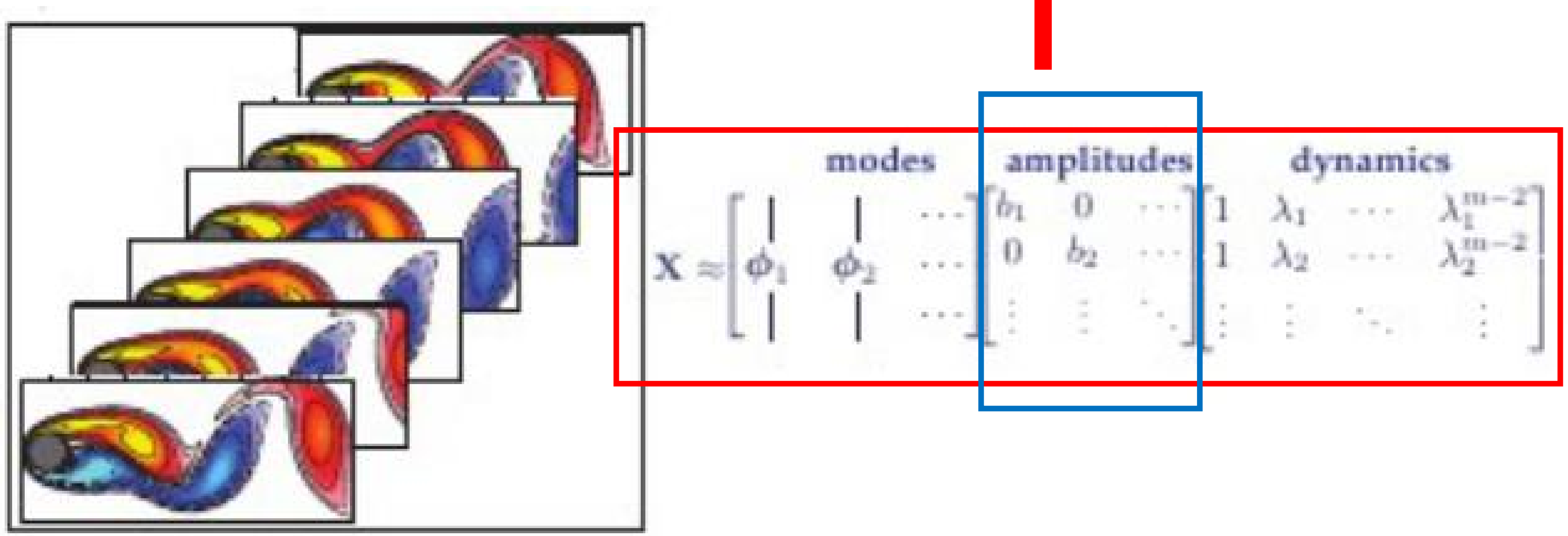
$$
\mathbf{X} \approx \overset{\text{modes}}{\begin{bmatrix} | & | & \cdots \\ \phi_1 & \phi_2 & \cdots \\ | & | & \cdots \end{bmatrix}} \overset{\text{amplitudes}}{\begin{bmatrix} b_1 & 0 & \cdots \\ 0 & b_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}} \overset{\text{dynamics}}{\begin{bmatrix} 1 & \lambda_1 & \cdots & \lambda_1^{m-2} \\ 1 & \lambda_2 & \cdots & \lambda_2^{m-2} \\ \vdots & \vdots & \ddots & \vdots \end{bmatrix}}
$$

Credit: Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton & Kutz)

Dynamic Mode Decomposition

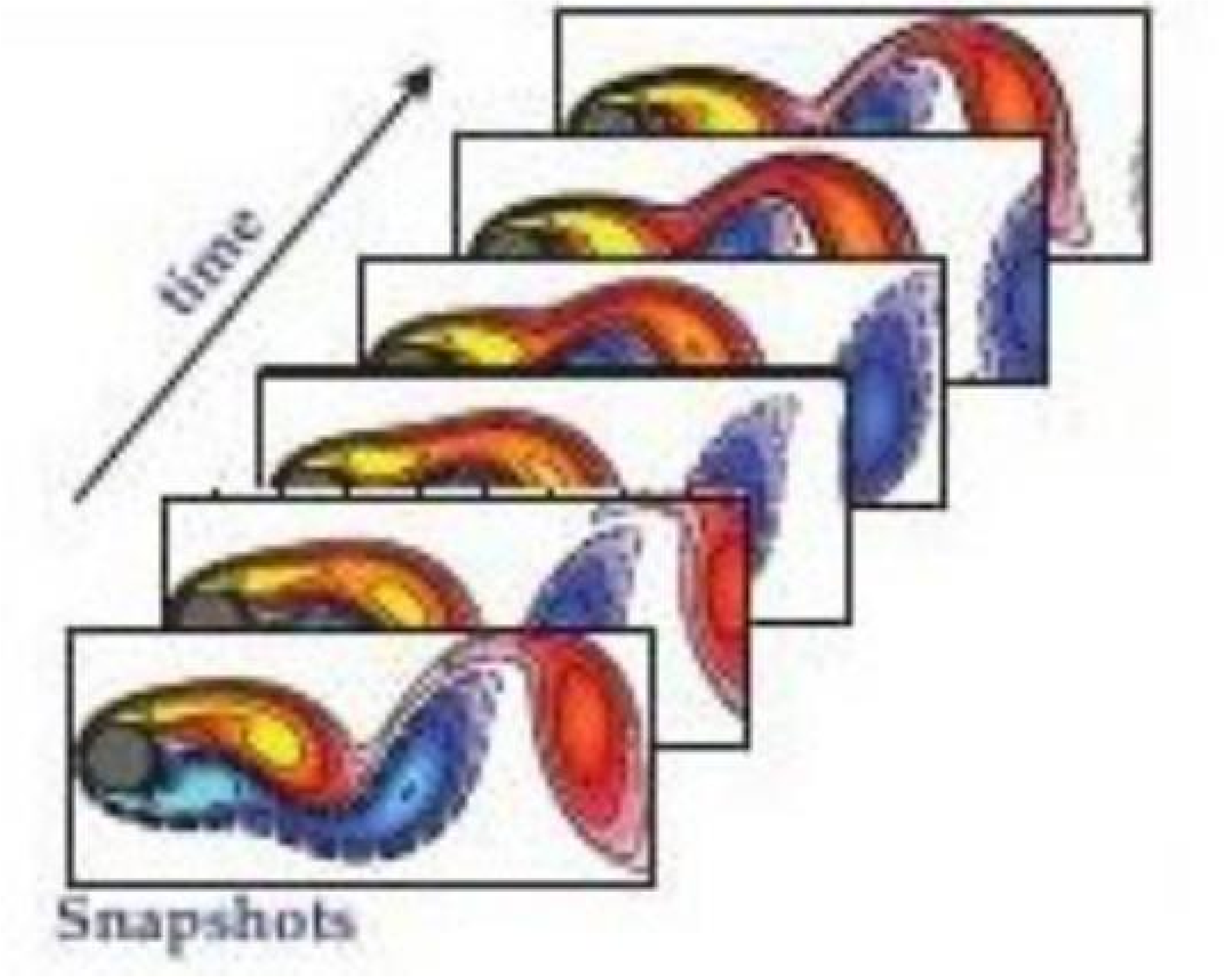
$$V = \mathbf{\Phi}\, D_{\alpha}\, M_{\mu}$$

Synthesis (Reconstruction)



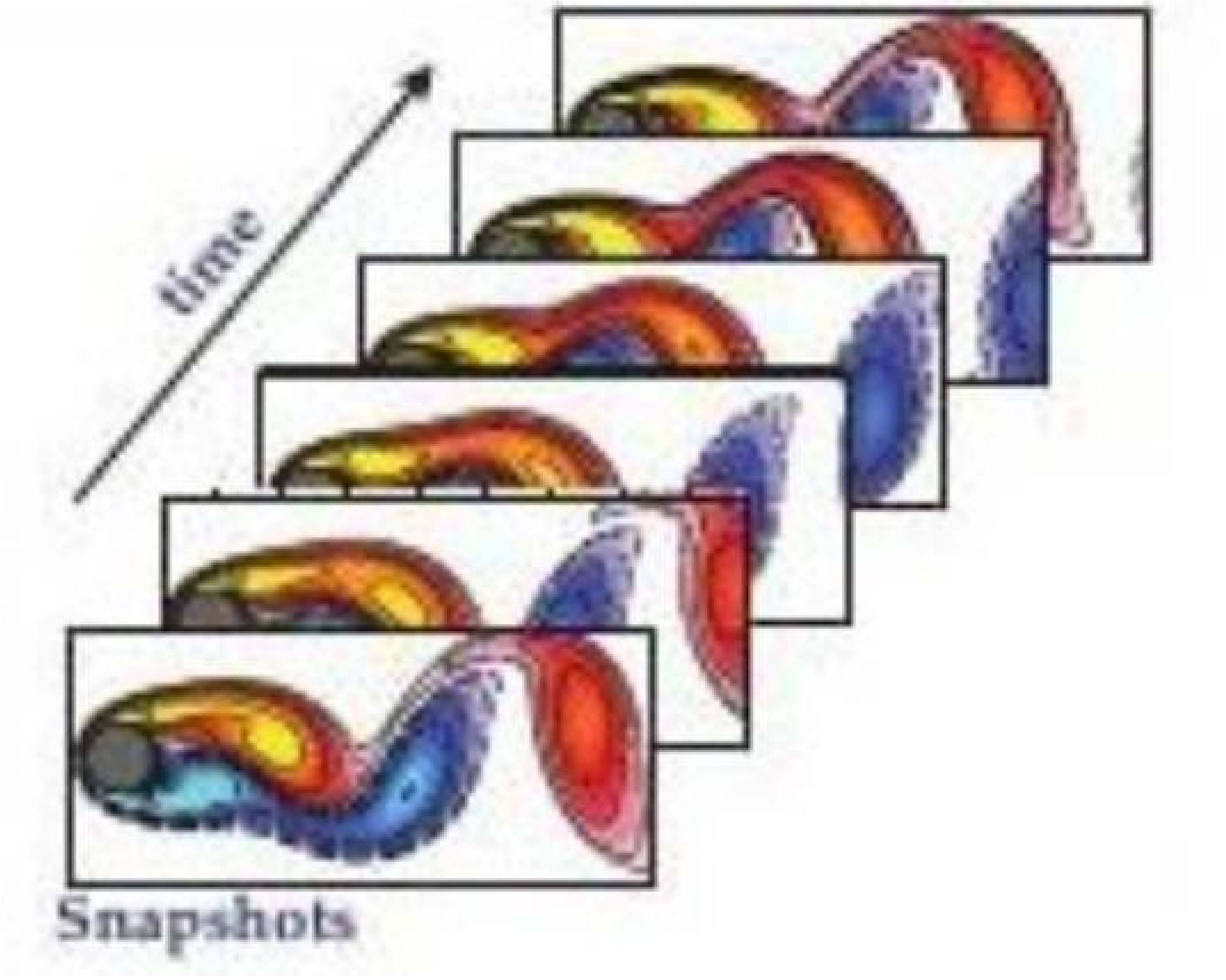
Credit: Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton & Kutz)

Database: Re_D=100 flow around a cylinder

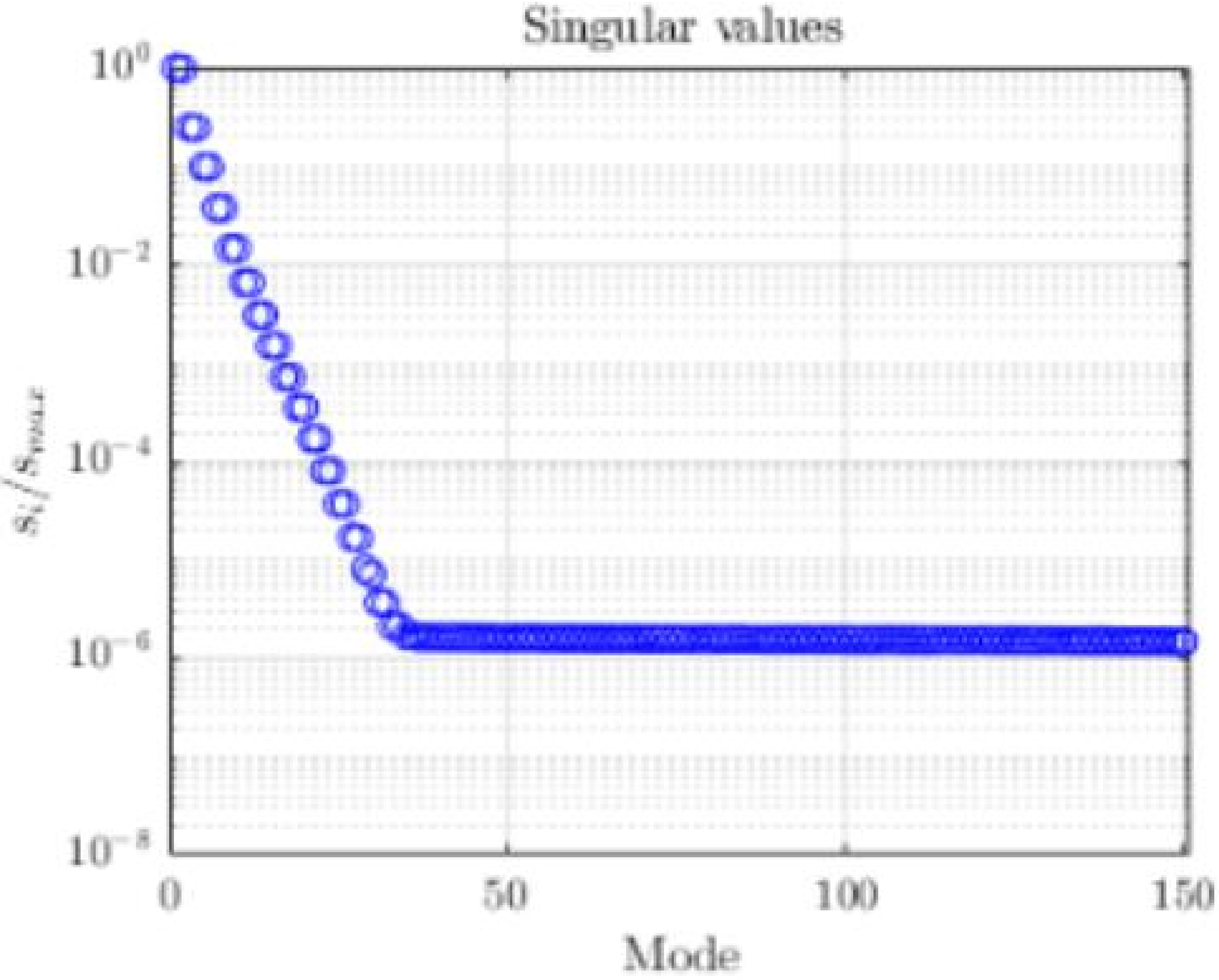


Database: Re_D=100 flow around a cylinder

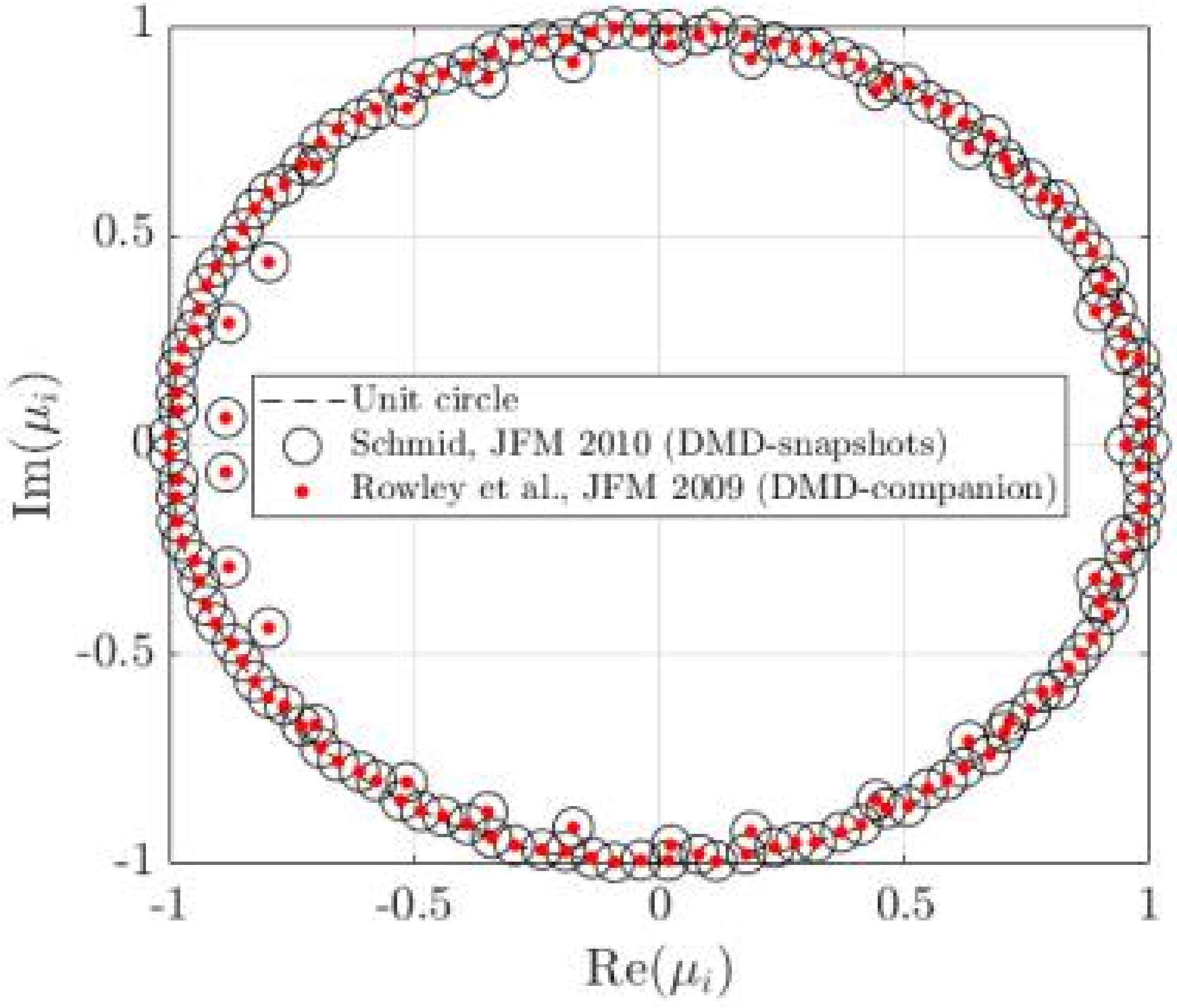
“Gentle” problem.



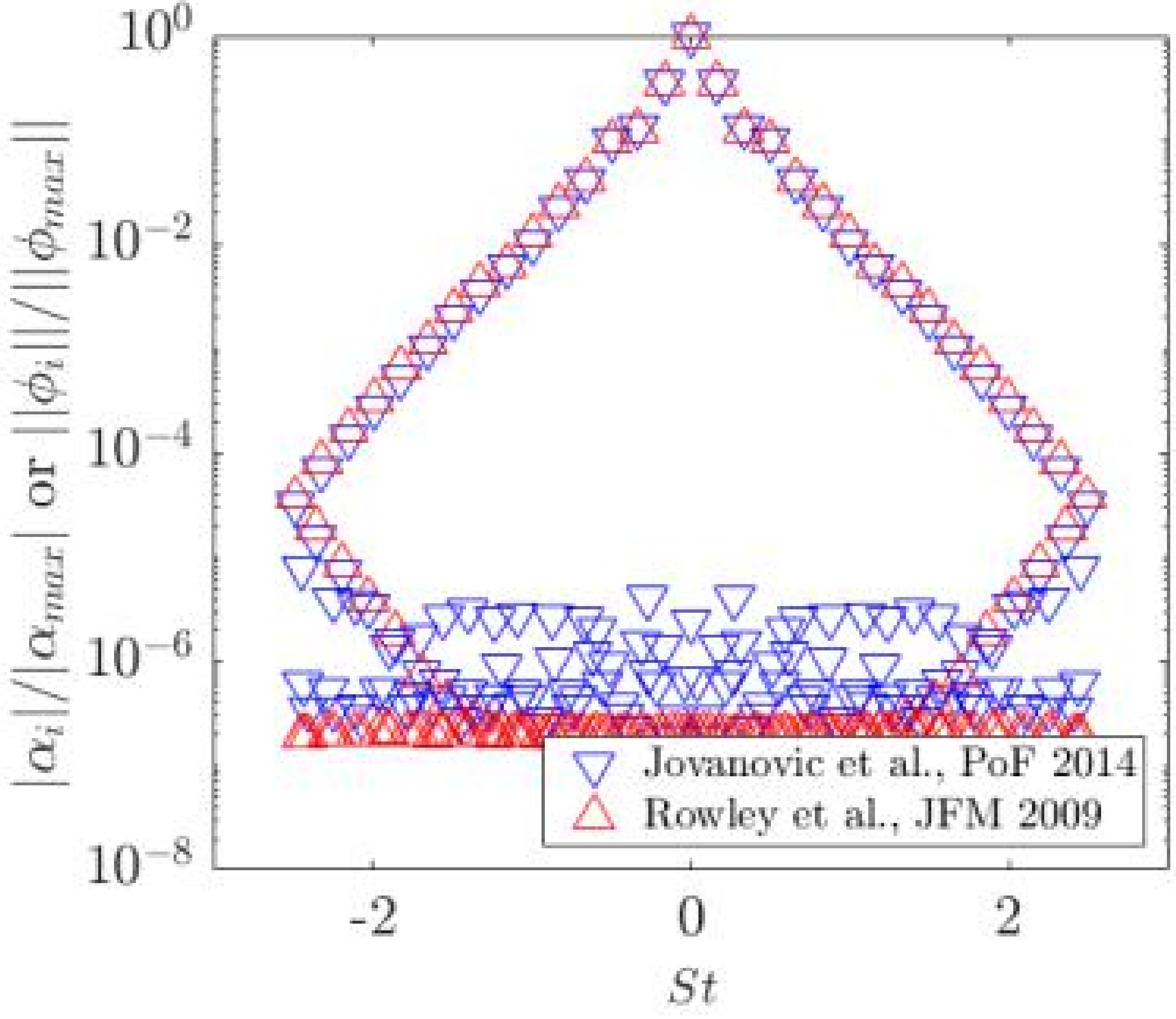
Database: Re_D=100 flow around a cylinder



Database: Re_D=100 flow around a cylinder



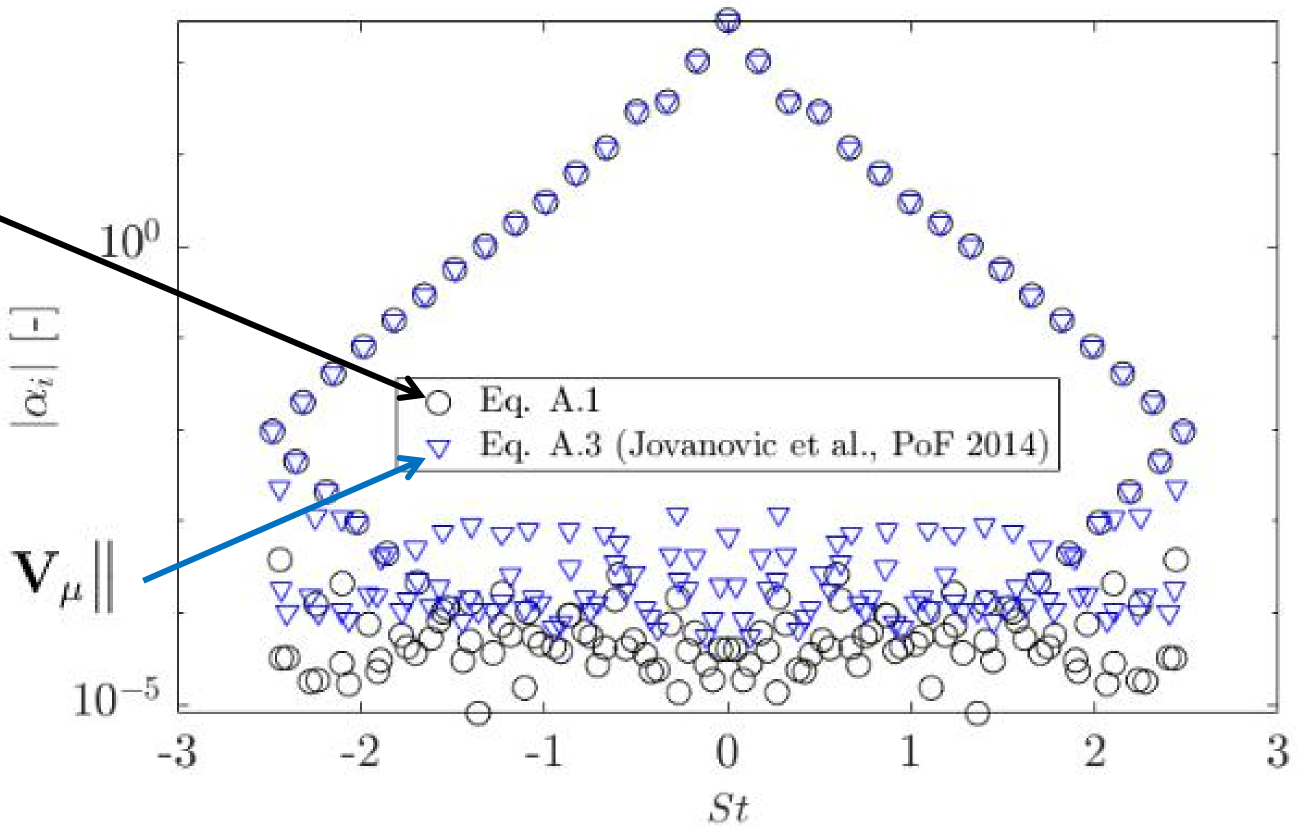
Database: Re_D=100 flow around a cylinder



Database: Re_D=100 flow around a cylinder

$$\boldsymbol{\alpha} = \boldsymbol{\Phi}^{+}\,\mathbf{v}_1$$

$$\min_{\boldsymbol{\alpha}} \left\| \mathbf{C}_1^{n_t-1} - \mathbf{W}\,\mathbf{D}_{\alpha}\,\mathbf{V}_{\mu} \right\|$$



Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton, Kutz, *et al.*)

M.R. Jovanovic, P.J. Schmid, J.W. Nichols, Sparsity-promoting dynamic mode decomposition, Phys. Fluids, vol. 26 (2), 2014.

Database: Re_D=100 flow around a cylinder

Method	Reconstruction error	Computing time $[s]$
Companion **DMD**	N.A.	4.72×10^{-2}
$\boldsymbol{\alpha} = \boldsymbol{\Phi}^{+}\, \mathbf{v}_1$	1.64×10^{-6}	1.85
$\min\limits_{\boldsymbol{\alpha}} \left\|\mathbf{C}_1^{n_t-1} - \mathbf{W}\, \mathbf{D}_\alpha\, \mathbf{V}_\mu\right\|$	5.85×10^{-13}	6.53×10^{-4}

Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems (Brunton, Kutz, *et al.*)

M.R. Jovanovic, P.J. Schmid, J.W. Nichols, Sparsity-promoting dynamic mode decomposition, Phys. Fluids, vol. 26 (2), 2014.

Dynamic Mode Decomposition can be understood both

- as a data-driven modal decomposition technique, and
- as matrix factorization technique

Dynamic Mode Decomposition can be understood both

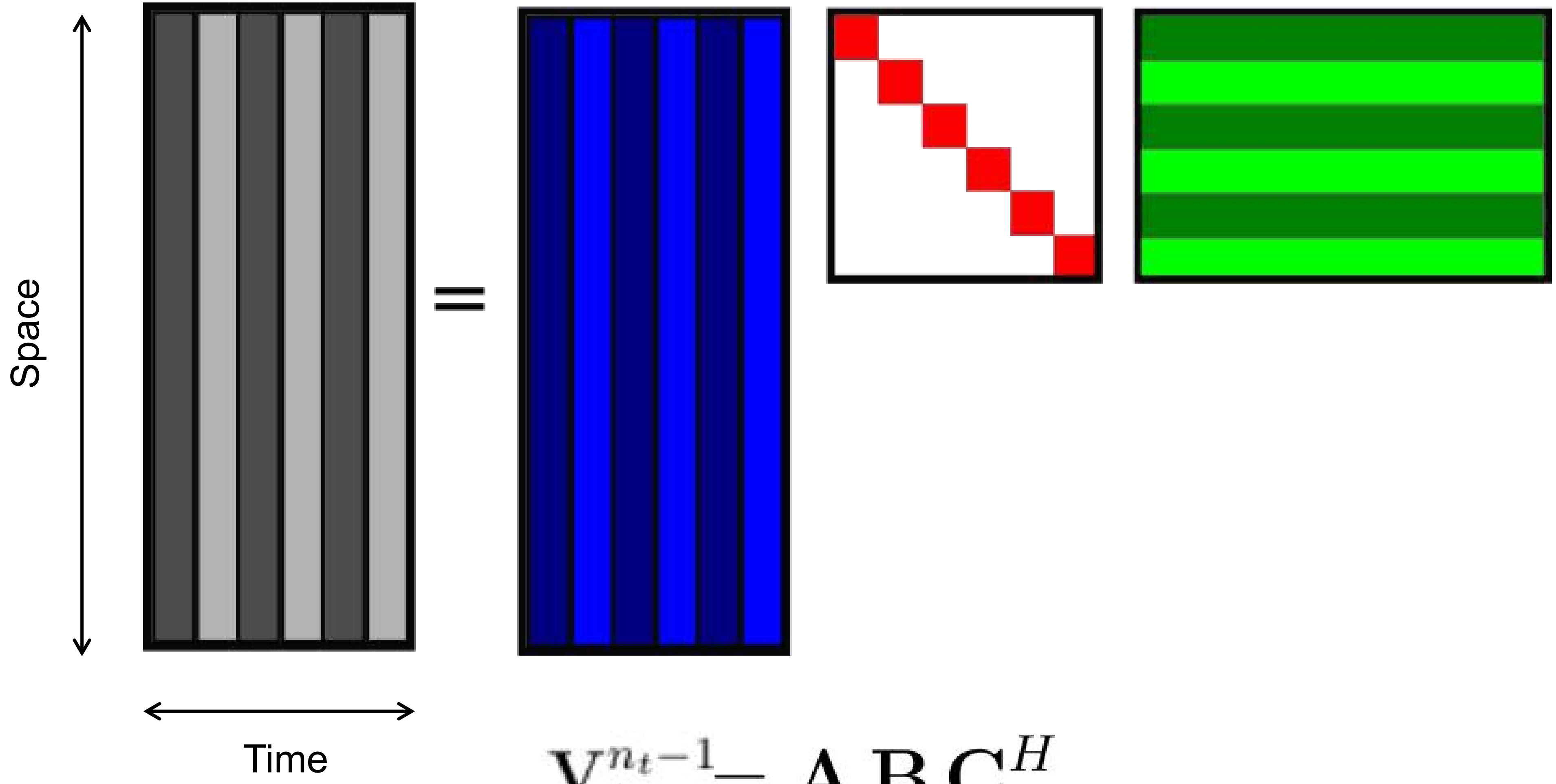
- as a data-driven modal decomposition technique, and
- as matrix factorization technique

This leads to connections with other decomposition techniques:

FFT, Spectral POD, multi-scale POD, multi-resolution DMD ...

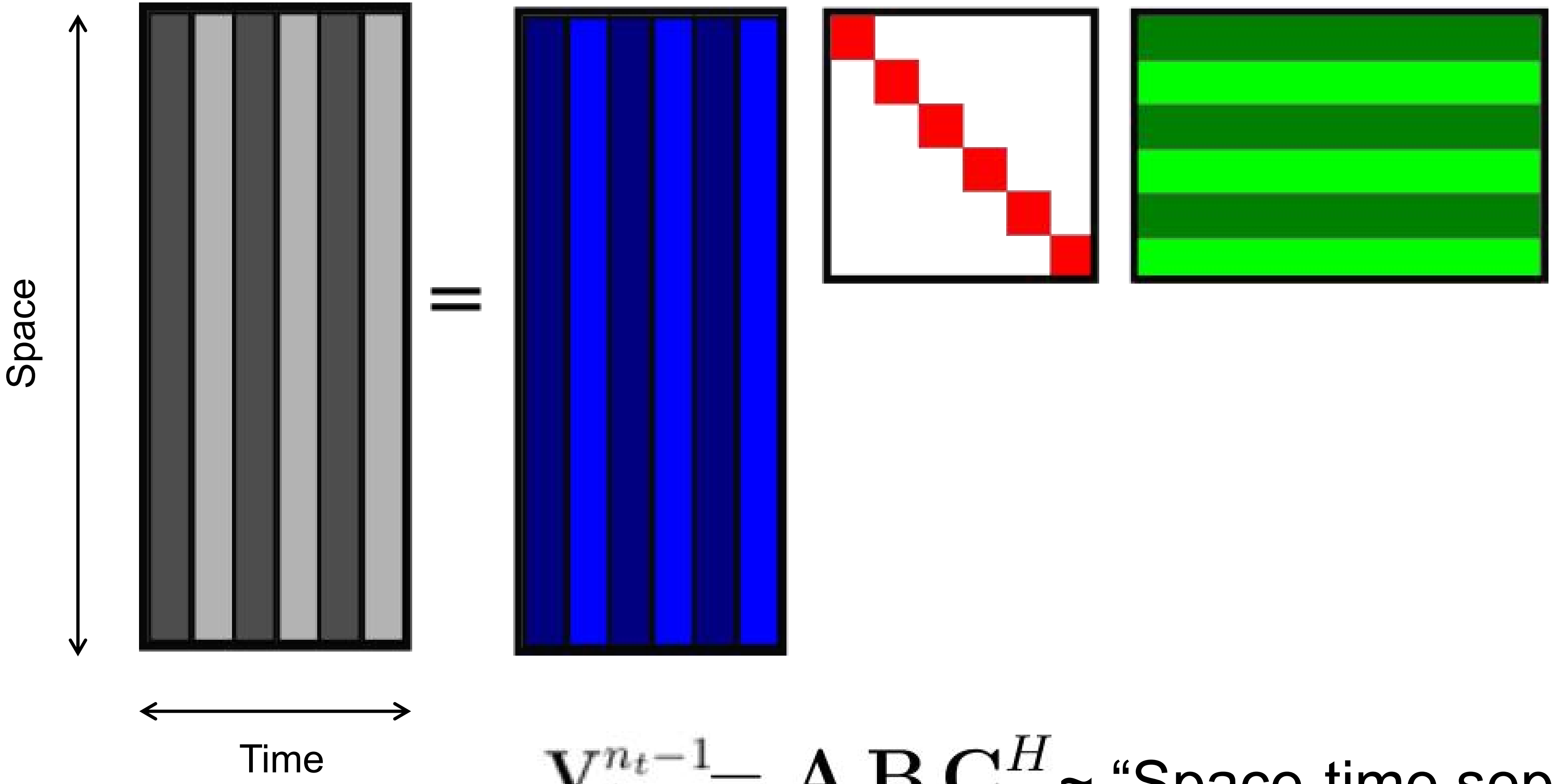
B. Begiashvili *et al.,* Data-driven modal decomposition methods as feature detection techniques for flow problems: a critical assessment, Phys. Fluids 35, 041301 (2023).

Matrix factorization techniques



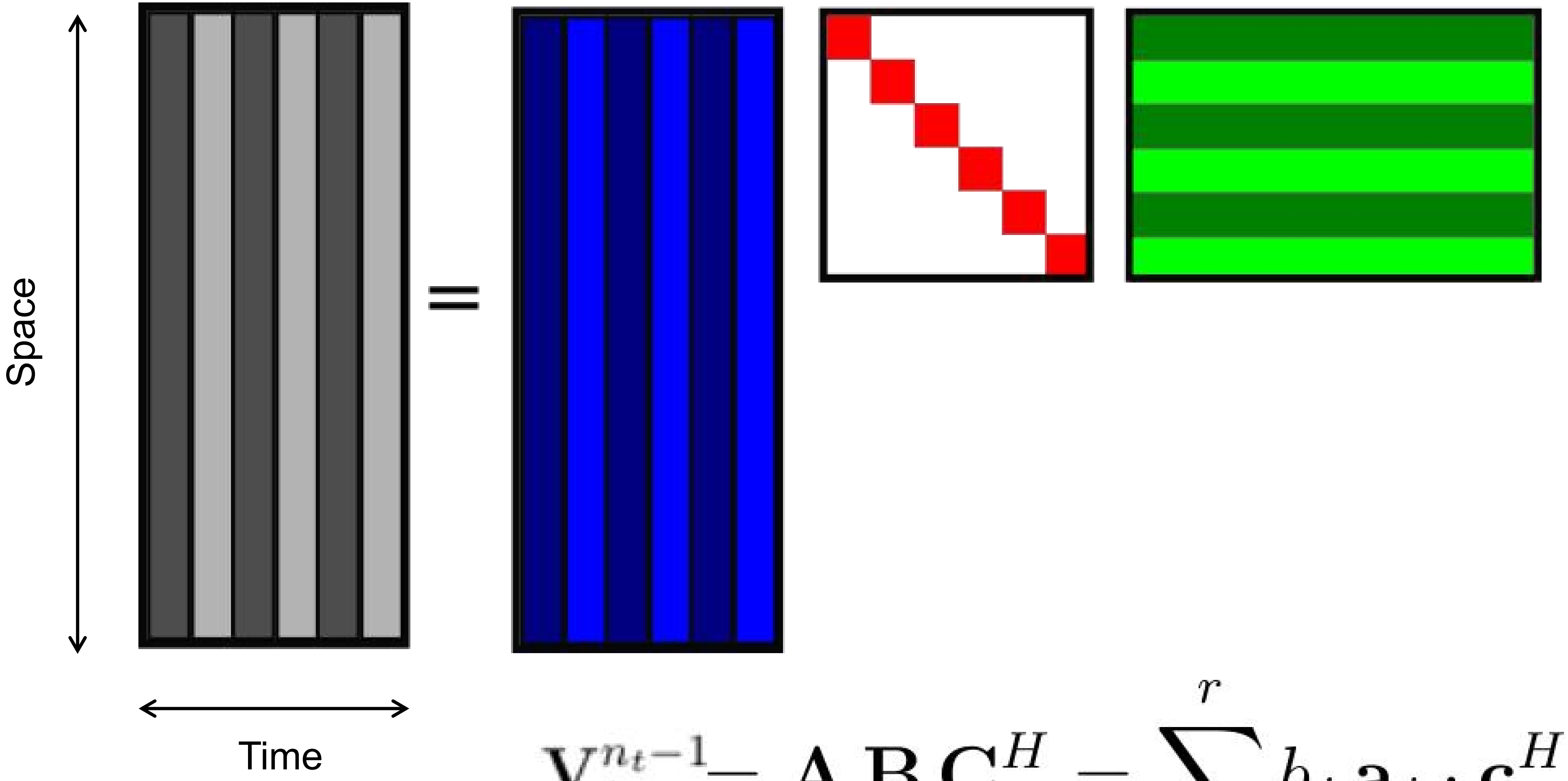
$$\mathbf{V}_1^{n_t-1} = \mathbf{A}\,\mathbf{B}\,\mathbf{C}^H$$

Matrix factorization techniques



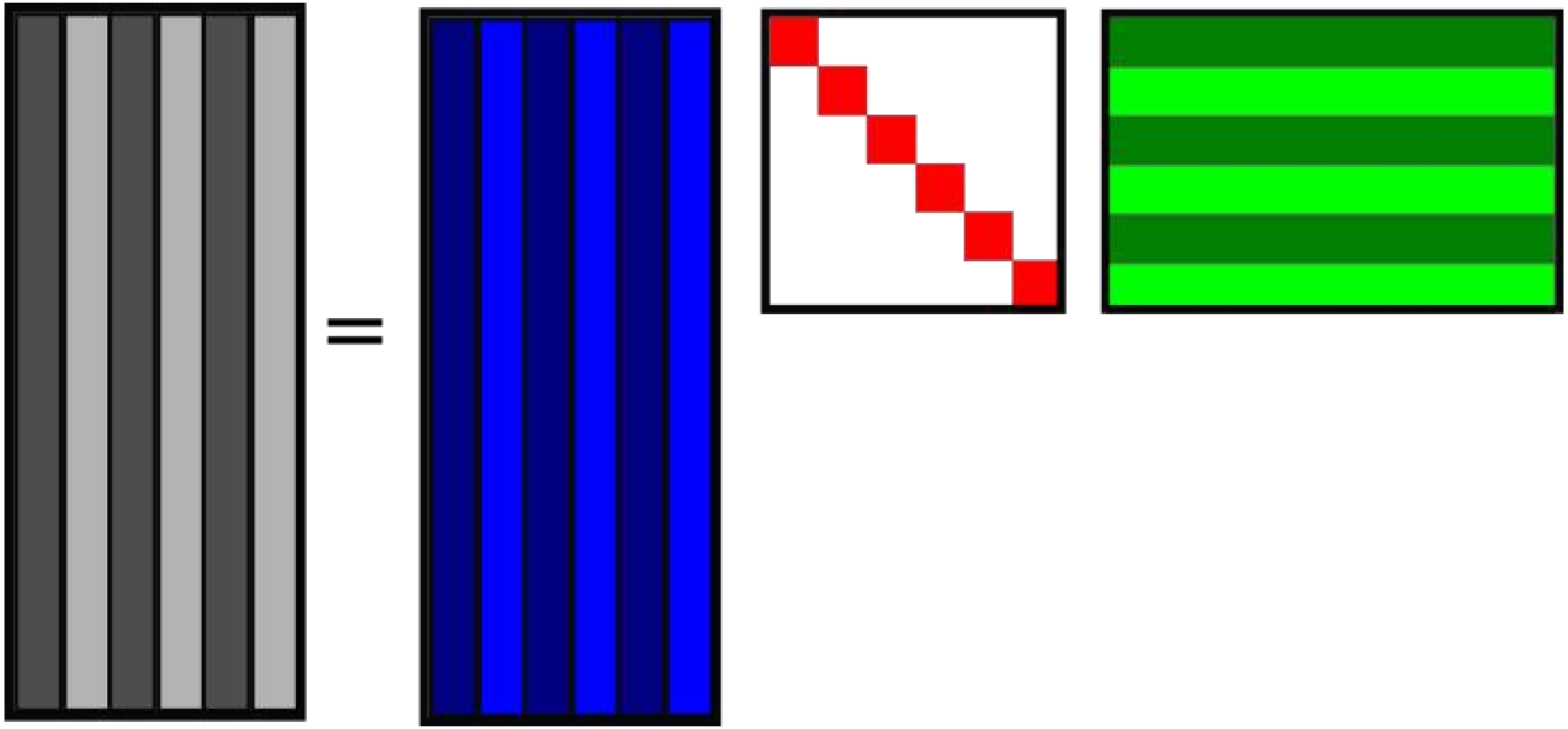
$\mathbf{V}_1^{n_t-1} = \mathbf{A}\,\mathbf{B}\,\mathbf{C}^H$ ~ “Space-time separation”

Matrix factorization techniques



$$\mathbf{V}_1^{n_t-1} = \mathbf{A}\,\mathbf{B}\,\mathbf{C}^H = \sum_{j=1}^{r} b_j\,\mathbf{a}_j \cdot \mathbf{c}_j^H$$

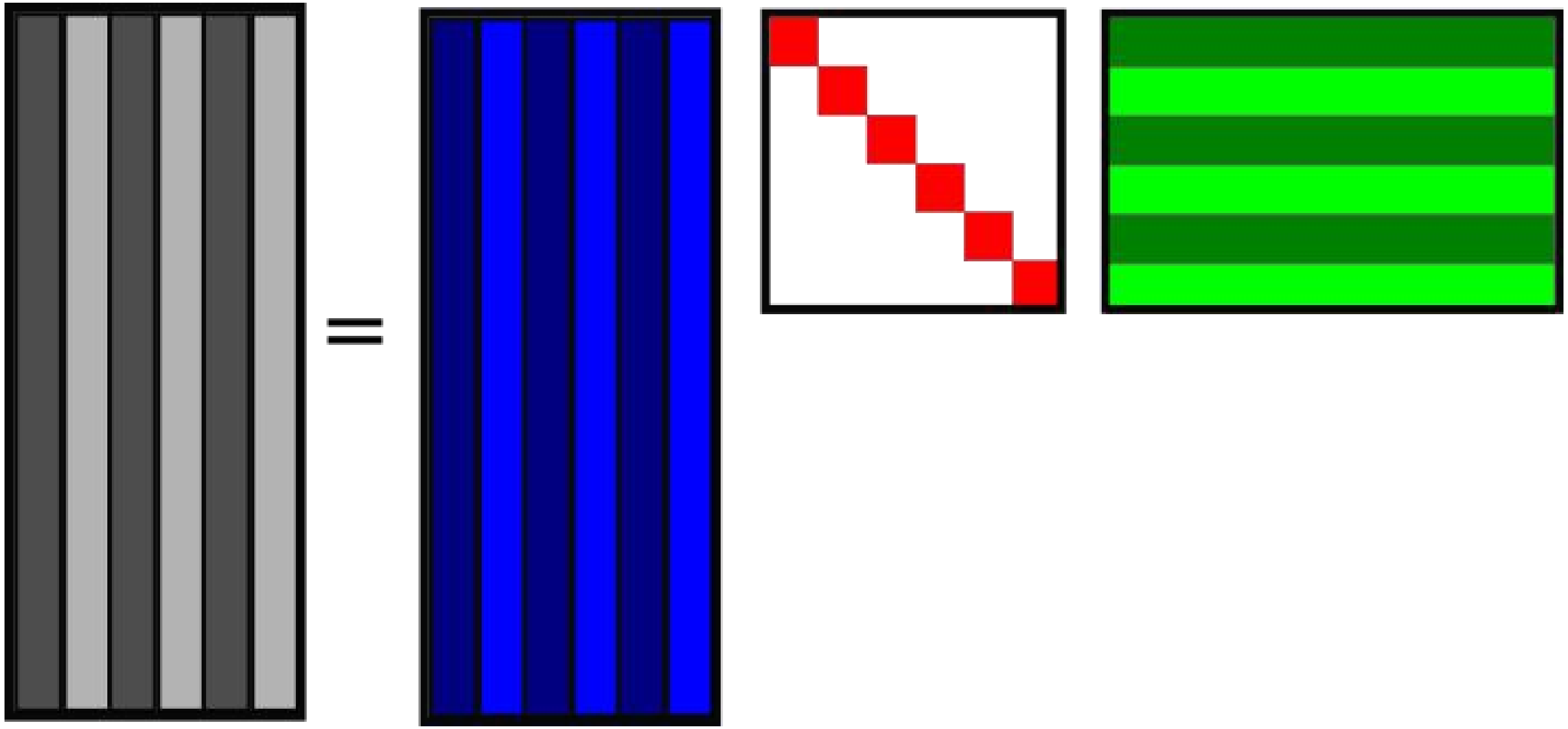
SVD/DMD are both matrix factorization techniques



$$\mathbf{V}_1^{n_t-1} \stackrel{SVD}{=} \mathbf{L}_0\,\mathbf{S}_0\,\mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j}\,\mathbf{l}_{0,j}\cdot\mathbf{r}_{0,j}^T$$

Matrix L_0: Left Singular Vectors (space, real), $L_0^H L_0 = I_m$
Matrix S_0: Singular Values (energy, real)
Matrix R_0: Right Singular Vectors (time, real, mixes frequencies), $R_0^H R_0 = I_n$

SVD/DMD are both matrix factorization techniques



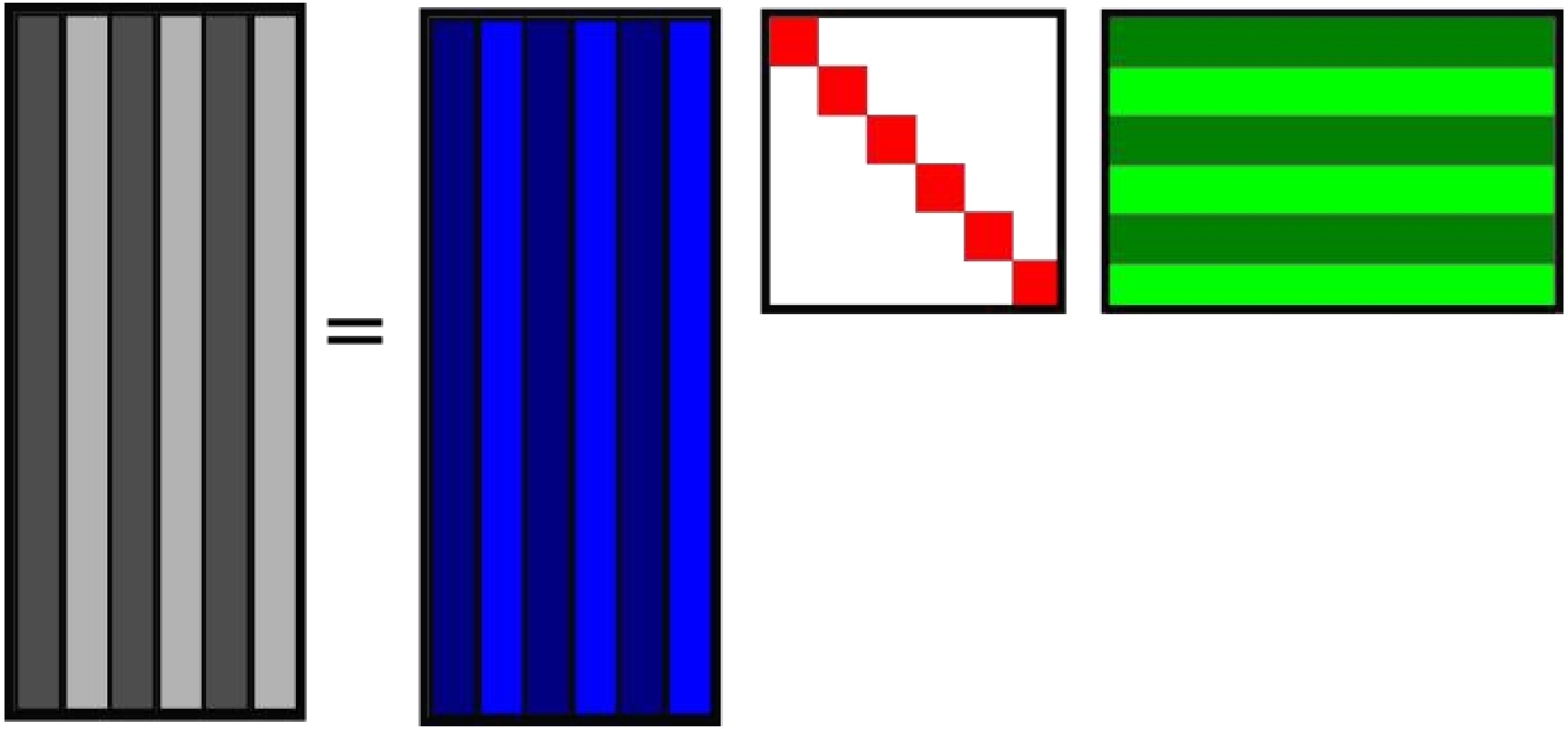
$$\mathbf{V}_1^{n_t-1} \overset{DMD}{=} \mathbf{\Phi}\,\mathbf{D}_\alpha\,\mathbf{V}_\mu = \sum_{j=1}^{r} \alpha_j\,\boldsymbol{\phi}_j \cdot \boldsymbol{\mu}^j$$

Matrix **Φ**: Dynamic Modes (space, complex)

Matrix D_α: Amplitudes (complex unless careful, meaning? importance?)

Matrix M_μ: Vandermonde matrix (time, complex, distinct frequencies)

SVD/DMD are both matrix factorization techniques

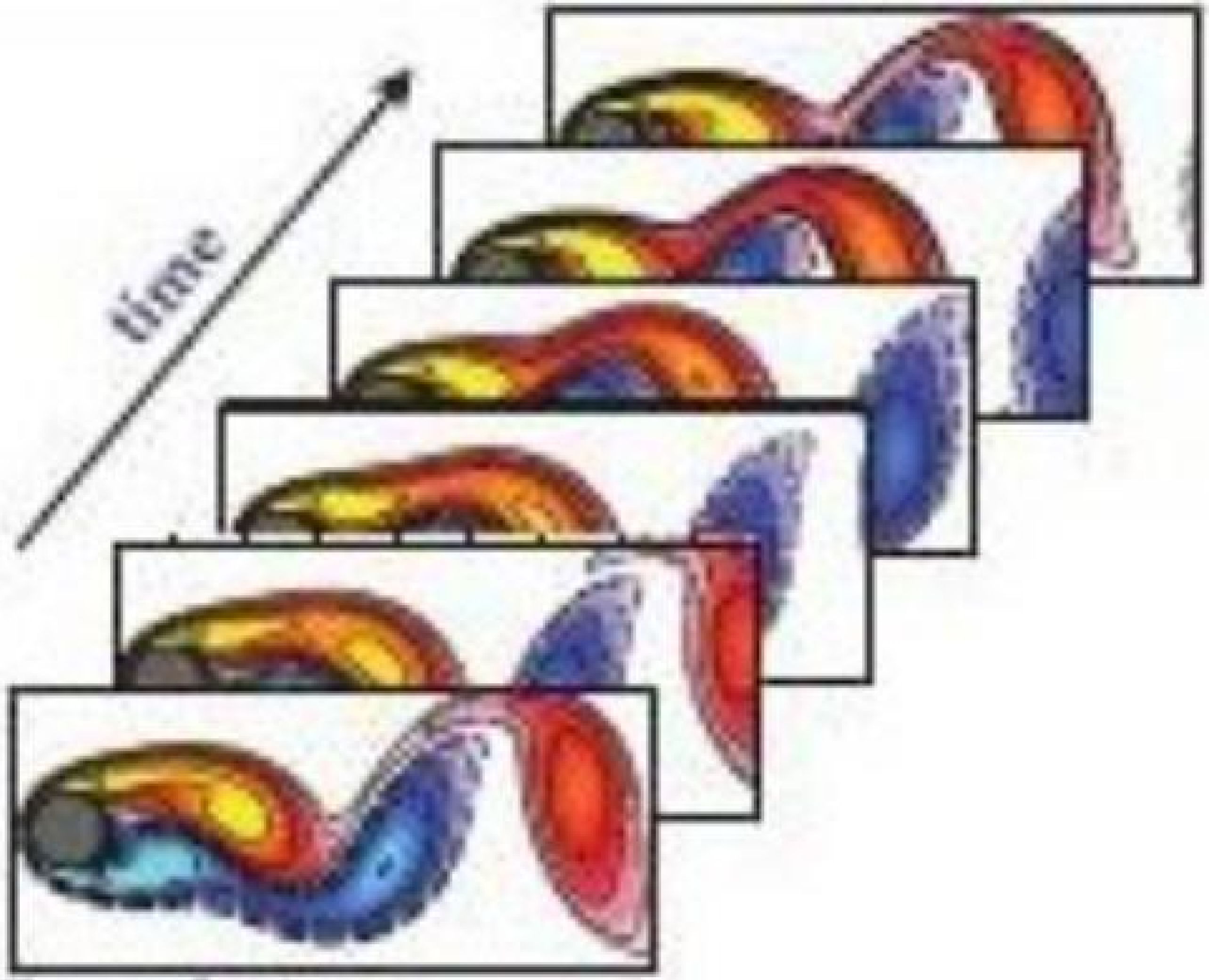


$$\mathbf{V}_1^{n_t} \stackrel{SVD}{=} \mathbf{L}_0 \boxed{\mathbf{S}_0 \, \mathbf{R}_0^T}$$

Scaled Chronos Matrix

$$\boxed{C_1^{n_t} \equiv \mathbf{S}_0 \, \mathbf{R}_0^T = \sum_{j=1}^{r} s_{0,j} \, \mathbf{e}_j \cdot \mathbf{r}_{0,j}^T}$$

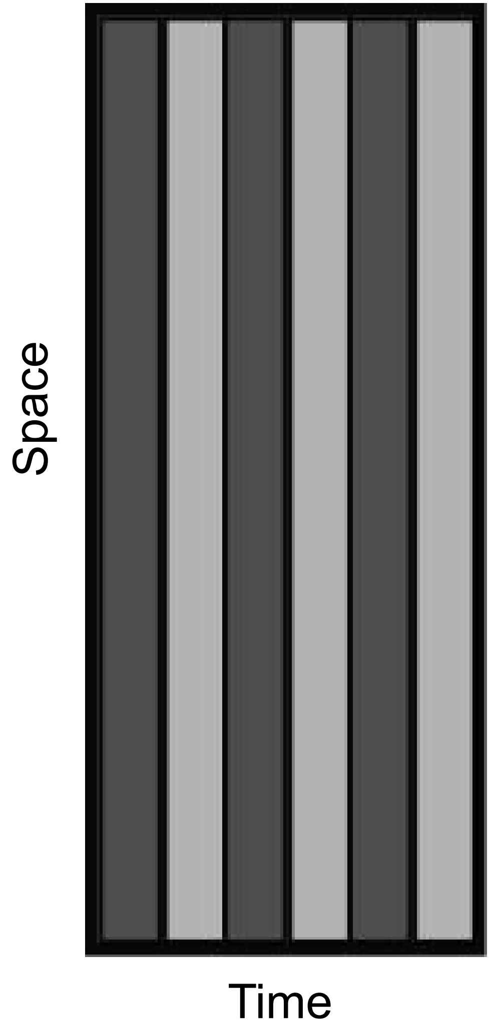
Alleviating the computational cost



Snapshots

There are many (n_p) spatial points in the system;

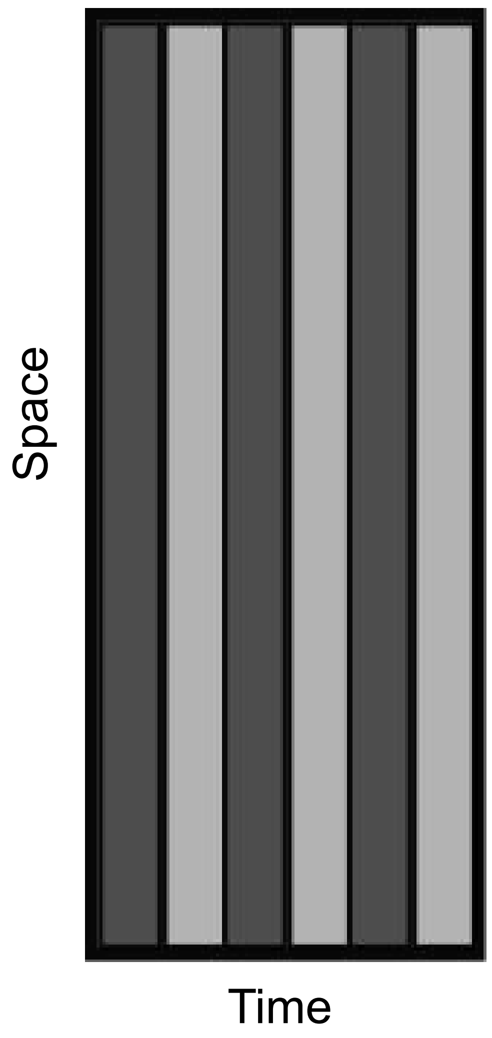
most of them will have coherent temporal variations.



There are many (n_p) spatial points in the system;

most of them will have coherent temporal variations.

Can few data points be representative of
the whole database temporal behaviour?



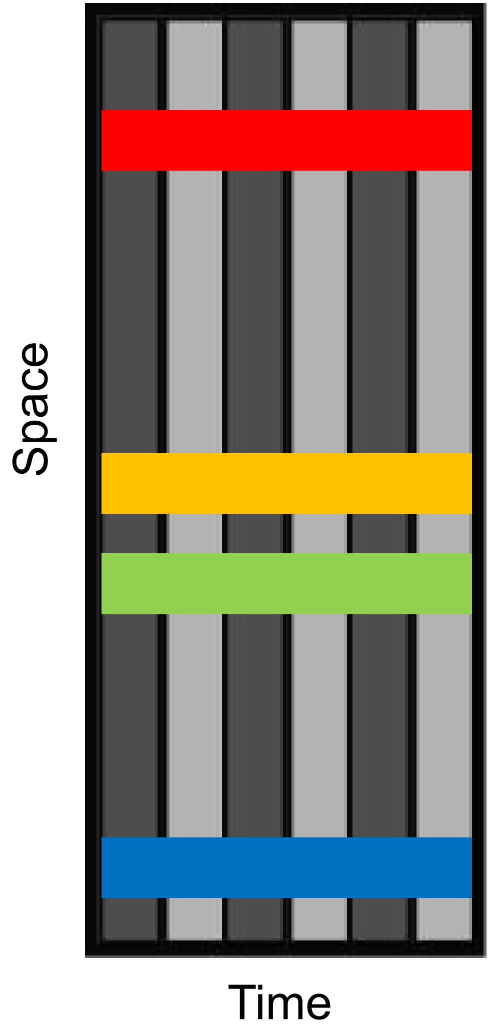
There are (n_p) many spatial points in the system;

most of them will have coherent temporal variations.

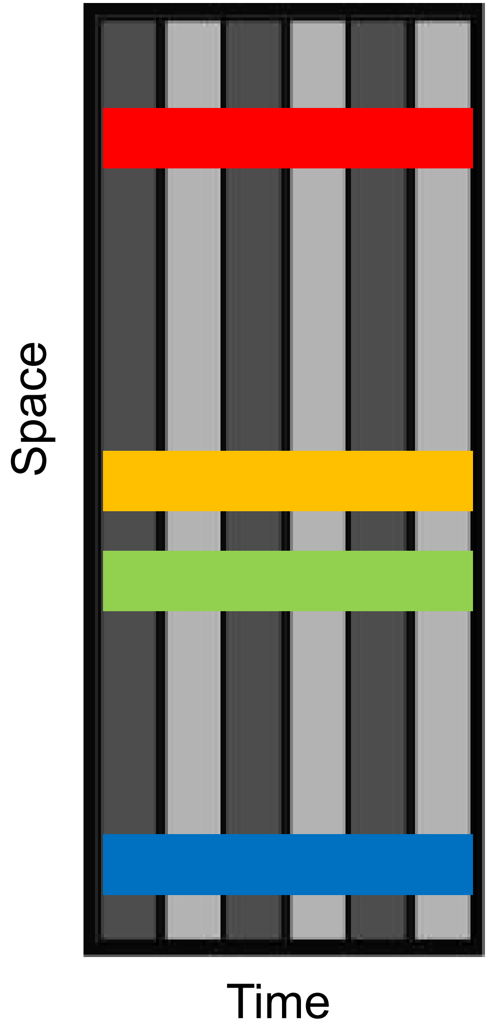
Can few data points be representative of
the whole database temporal behaviour?

Yes!

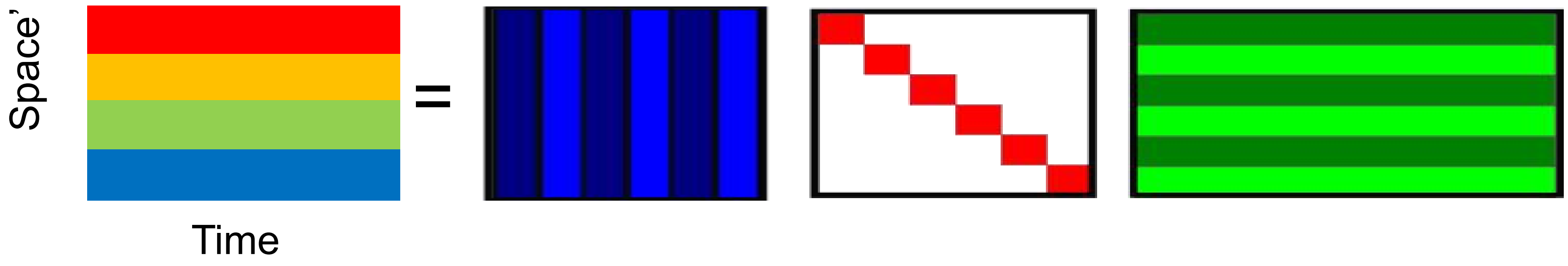
Those representative data points can be identified using *clustering* algorithms (Unsupervised Machine Learning).



Guéniat, Mathelin & Pastur, Phys. Fluids, vol. 27 (2), 2015.
Li, Garicano-Mena, Zheng & Valero, Energies, vol. 13 (9), 2020.



Guéniat, Mathelin & Pastur, Phys. Fluids, vol. 27 (2), 2015.
Li, Garicano-Mena, Zheng & Valero, Energies, vol. 13 (9), 2020.



Guéniat, Mathelin & Pastur, Phys. Fluids, vol. 27 (2), 2015.
Li, Garicano-Mena, Zheng & Valero, Energies, vol. 13 (9), 2020.

How can one identify those modes that have a most representative temporal variation?

Group those points that have comparable pdf's.

Guéniat, Mathelin & Pastur, Phys. Fluids, vol. 27 (2), 2015.

How can one identify those modes that have a most representative temporal variation?

Group those points that have comparable pdf's.

It is not easy to identify a pdf from discrete data, though.

Guéniat, Mathelin & Pastur, Phys. Fluids, vol. 27 (2), 2015.

How can one identify those modes that have a most representative temporal variation?

Group those points that have comparable pdf's.

It is not easy to identify a pdf from discrete data, though.

Take N_m statistical moments as representative of the pdf.

Guéniat, Mathelin & Pastur, Phys. Fluids, vol. 27 (2), 2015.

Spatially Agglomerated DMD analysis

1. Compute the first $\widetilde{N}_M$ (estimated) statistical moments, with $1 \leq \widetilde{N}_M \ll n_t$

2. Arrange those moments into matrix $\bar{\mathbf{M}} \in \mathbb{R}^{n_p \times \widetilde{N}_M}$

3. Spatial Agglomeration
   feed $\bar{\mathbf{M}}$ to clustering algorithm
   retrieve reduced database

4. Perform DMD analysis on spatially reduced database
   retrieve Ritz values & DMD modes

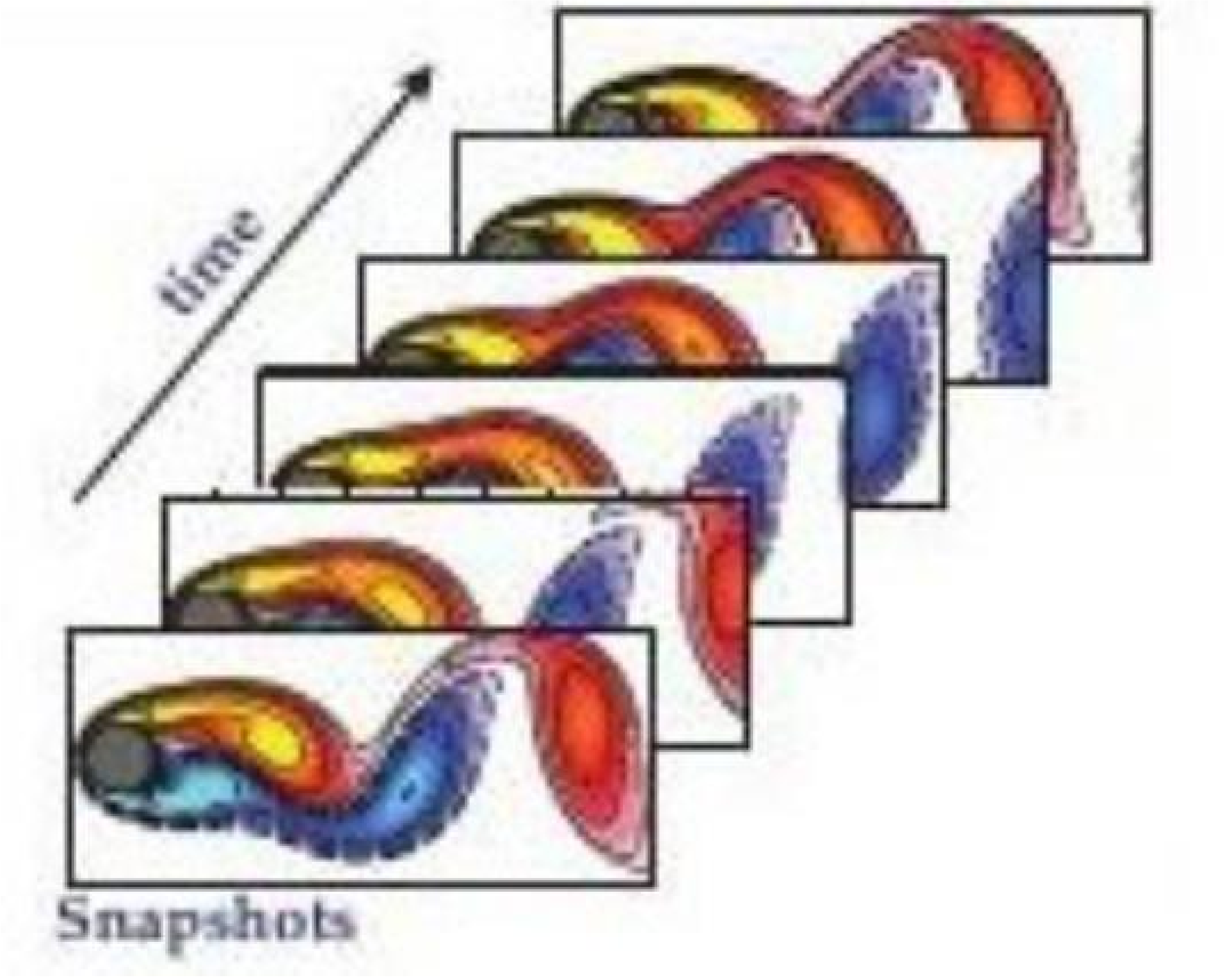
5. Reconstruct original DMD modes

Guéniat, Mathelin & Pastur, Phys. Fluids, vol. 27 (2), 2015.

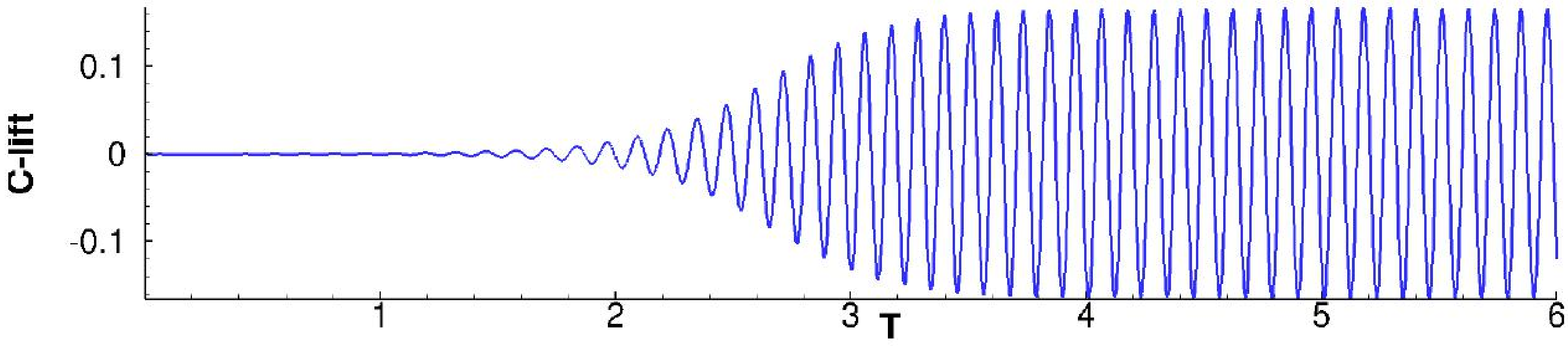
Algorithms	Complexity	Parameters
K-Means (`scikit-learn`)	Spatial: $\mathcal{O}(n_p(\widetilde{N_M}+\widetilde{n_p}))$. Temporal: $\mathcal{O}(n_p\widetilde{n_p}I)$.	-
Mini-batch K-Means (`scikit-learn`)	See above.	-
K-Means (`SciPy`)	See above.	-
K-Means++ (`SciPy`)	See above.	-
DBSCAN (`scikit-learn`)	Spatial: $\mathcal{O}(n_p)$. Temporal: $\mathcal{O}(n_p^2 t_d)$.	$d_{max}=2.2$, $n_{min}=2$.
HDBSCAN (`HDBSCAN`)	Spatial: $\mathcal{O}(n_p\widetilde{N}_M)$. Temporal: $\mathcal{O}(n_p^2\widetilde{N}_M)$.	$n_{min}=2$.
C-Means (`SciPy/skfuzzy`)	Similar to K-means, affected by fuzzifier [45].	$N_{cluster}=250$, $I_{max}=1000$.
Gaussian Mixture (`scikit-learn`)	Spatial: $\mathcal{O}(n_p k_G \widetilde{N}_M^3)$. Temporal: $\mathcal{O}(n_p k_G \widetilde{N}_M^3)$ [46,47].	$k_G=50$.
Mean Shift (`scikit-learn`)	Spatial: $\mathcal{O}(n_p\widetilde{N}_M)$. Temporal: $\mathcal{O}(n_p^2 I)$ [48].	$BW=15000/\widetilde{n_p}$.
Affinity Propagation (`scikit-learn`)	Spatial: $\mathcal{O}(n_p^2)$. Temporal: $\mathcal{O}(n_p^2 I)$.	-
Agglomerative Clustering (`scikit-learn`)	Spatial: $\mathcal{O}(n_p^2)$. Temporal: $\mathcal{O}(n_p^3)$.	`flag` $=$ *average*.
BIRCH (`scikit-learn`)	Spatial: $\mathcal{O}(n_p\widetilde{N_M})$. Temporal: $\mathcal{O}(n_p\widetilde{N_M})$.	-

Li, Garicano-Mena, Zheng & Valero, Dynamic Mode Decomposition Analysis of Spatially Agglomerated Flow Databases, Energies, vol. 13 (9), 2020.

Database: Re_D=60 flow around a cylinder

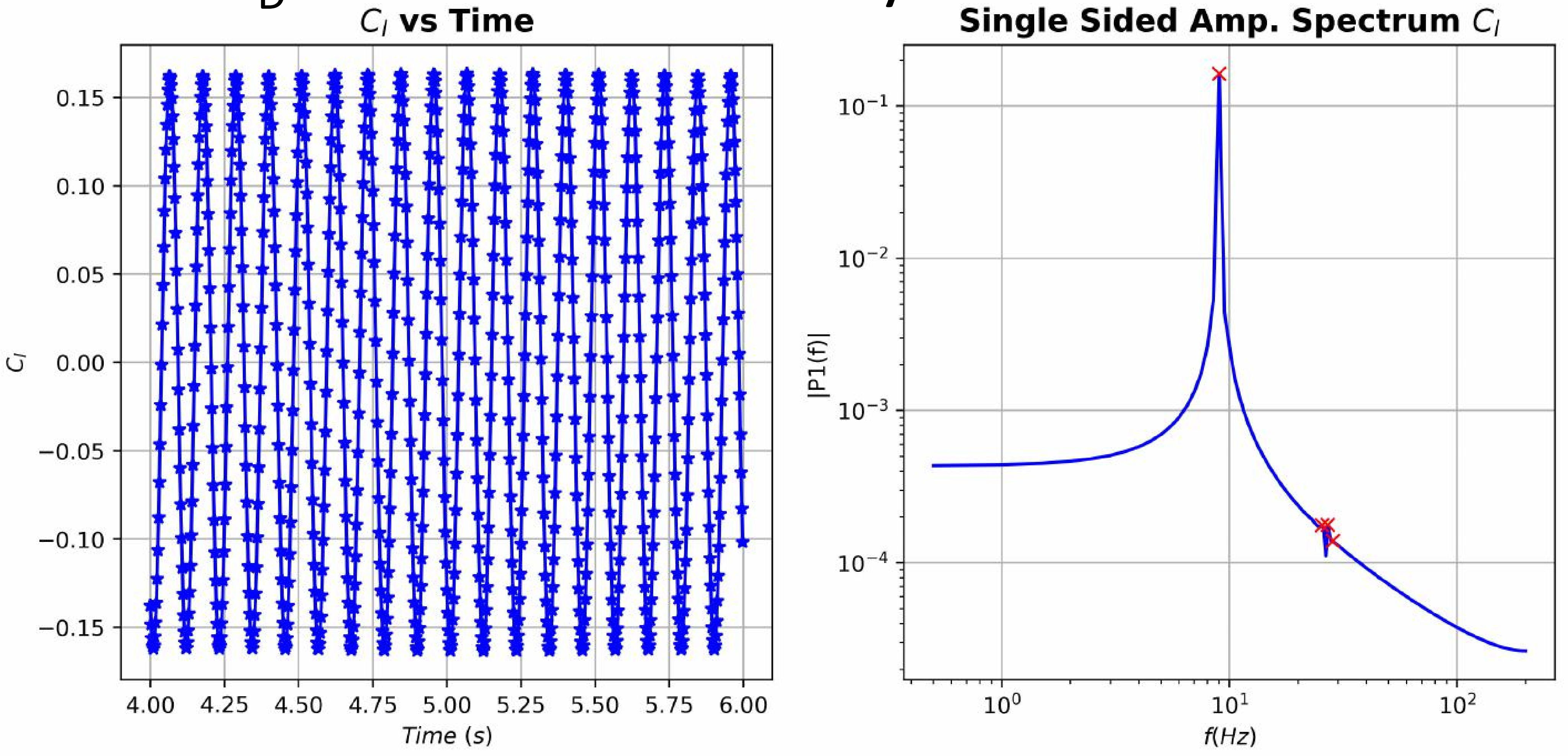


Database: Re_D=60 flow around a cylinder



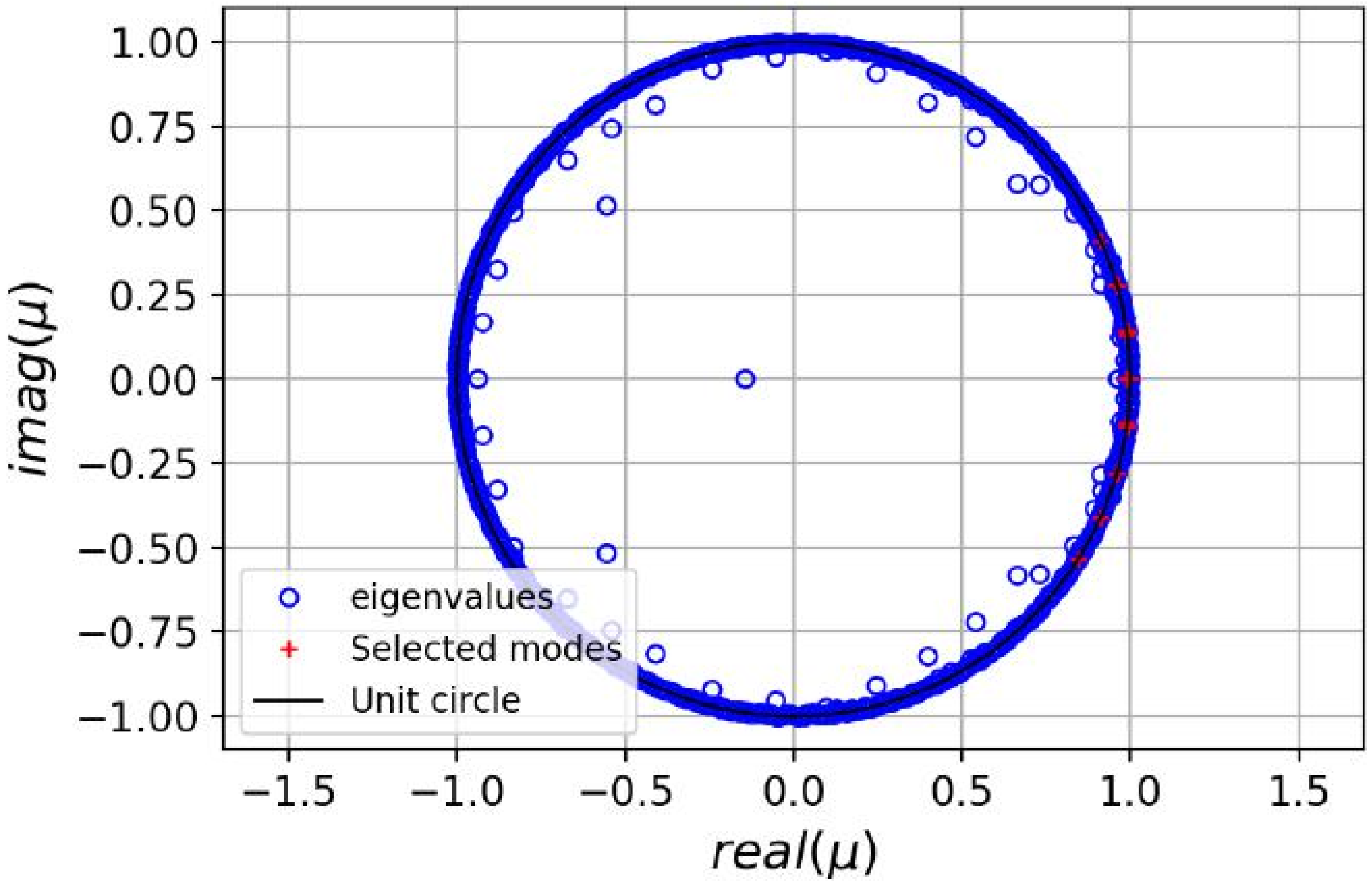
$Re_D = 60$ cylinder flow: Evolution of lift coefficient C_l from equilibrium steady solution to the limit cycle.

Database: Re_D=60 flow around a cylinder



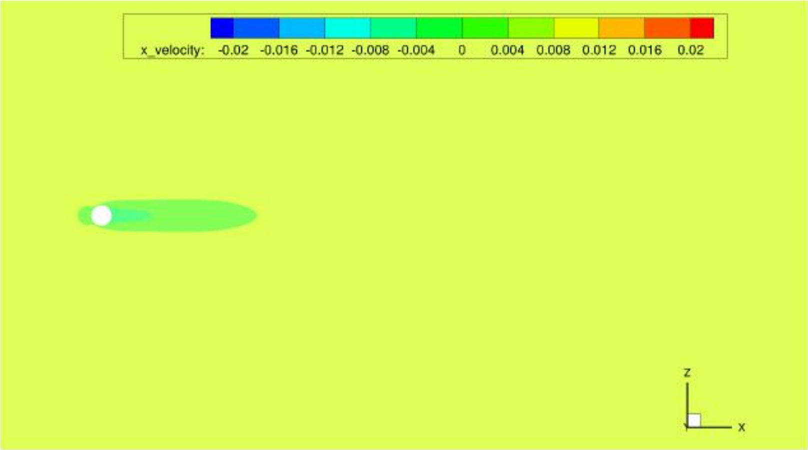
$Re_D = 60$ cylinder flow: C_l *vs* time and associated **FFT** spectrum.

Database: Re_D=60 flow around a cylinder

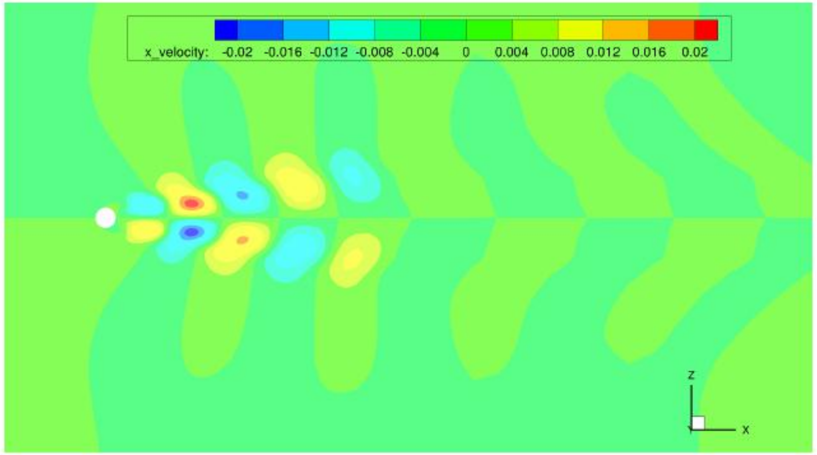


DMD spectrum (Ritz values)

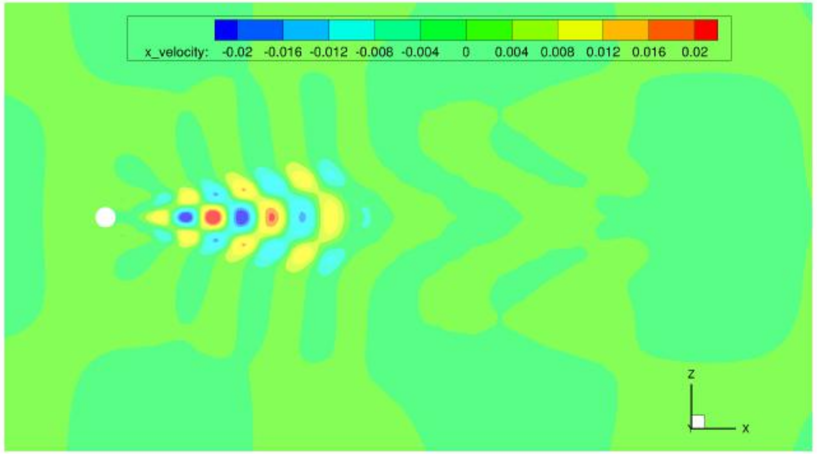
Database: Re_D=60 flow around a cylinder



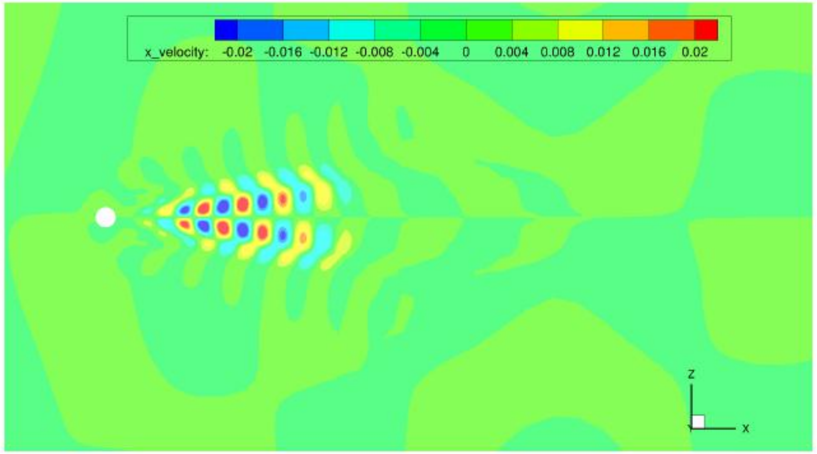
(a) Mode 0 : $f = 0.00,\ \sigma = 5.06 \times 10^{-6}$



(b) Mode 1-2: $f = 8.98,\ \sigma = 1.74 \times 10^{-3}$



(c) Mode 3-4: $f = 17.97, \sigma = 3.60 \times 10^{-3}$



(d) Mode 5-6: $f = 26.95, \sigma = 1.57 \times 10^{-3}$

$Re_D = 60$ cylinder flow: most relevant **DMD** modes, and corresponding frequencies and growth rates.

Database: Re_D=60 flow around a cylinder

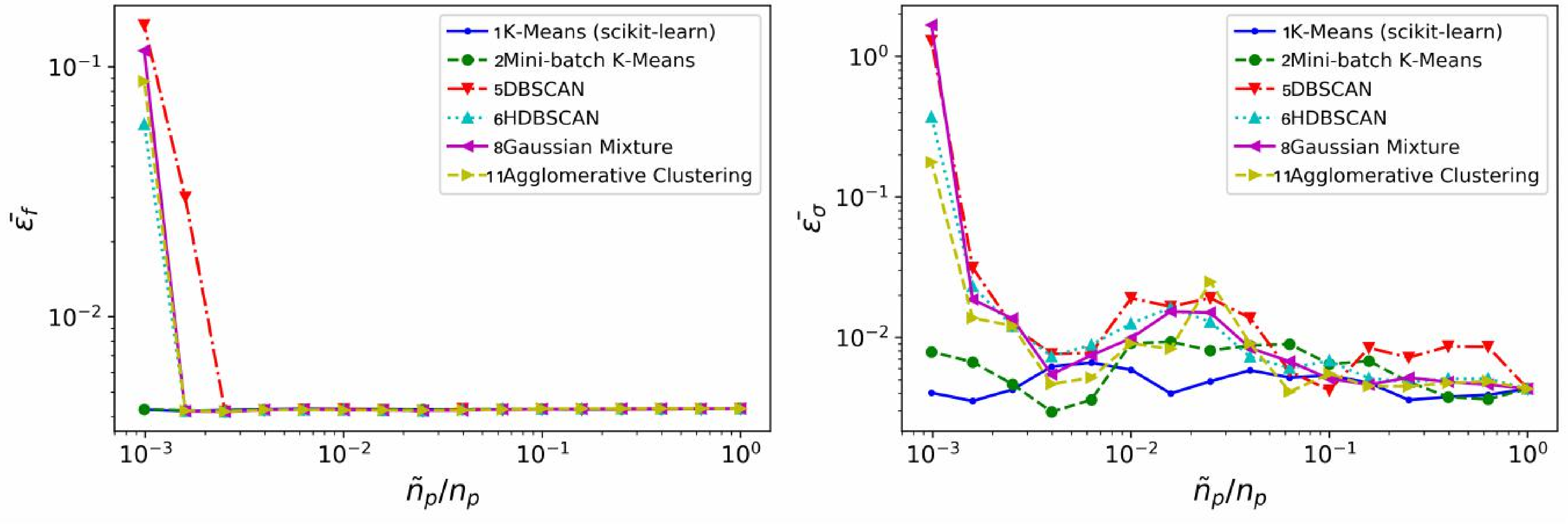
$$\mathbf{F} = \left[f_0, f_1, f_2, f_3\right]$$

$$\boldsymbol{\Sigma} = \left[\sigma_0, \sigma_1, \sigma_2, \sigma_3\right]$$

$$\varepsilon_f = \frac{\|\mathbf{F} - \mathbf{F}_C\|_2}{\|\mathbf{F}_C\|_2}$$

$$\varepsilon_\sigma = \frac{\|\boldsymbol{\Sigma}\|_2}{4}$$

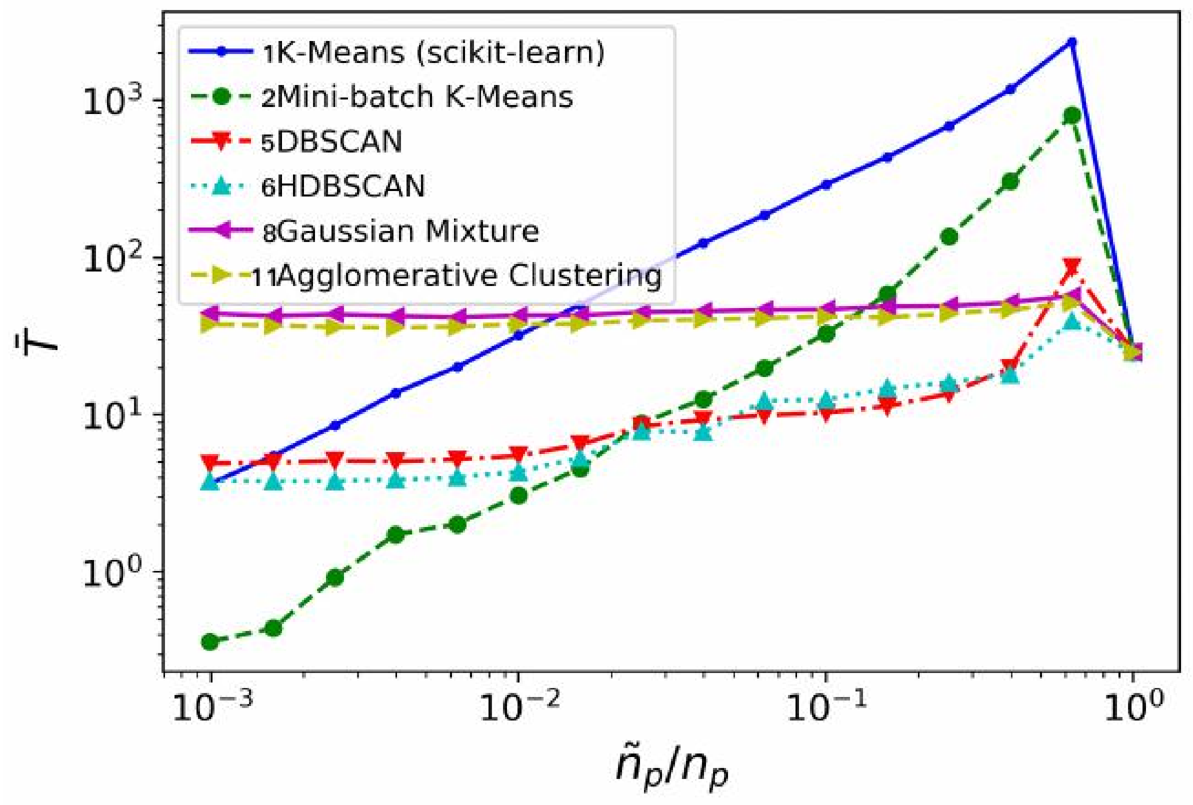
Database: Re_D=60 flow around a cylinder



(a) ε_f **(b)** ε_σ

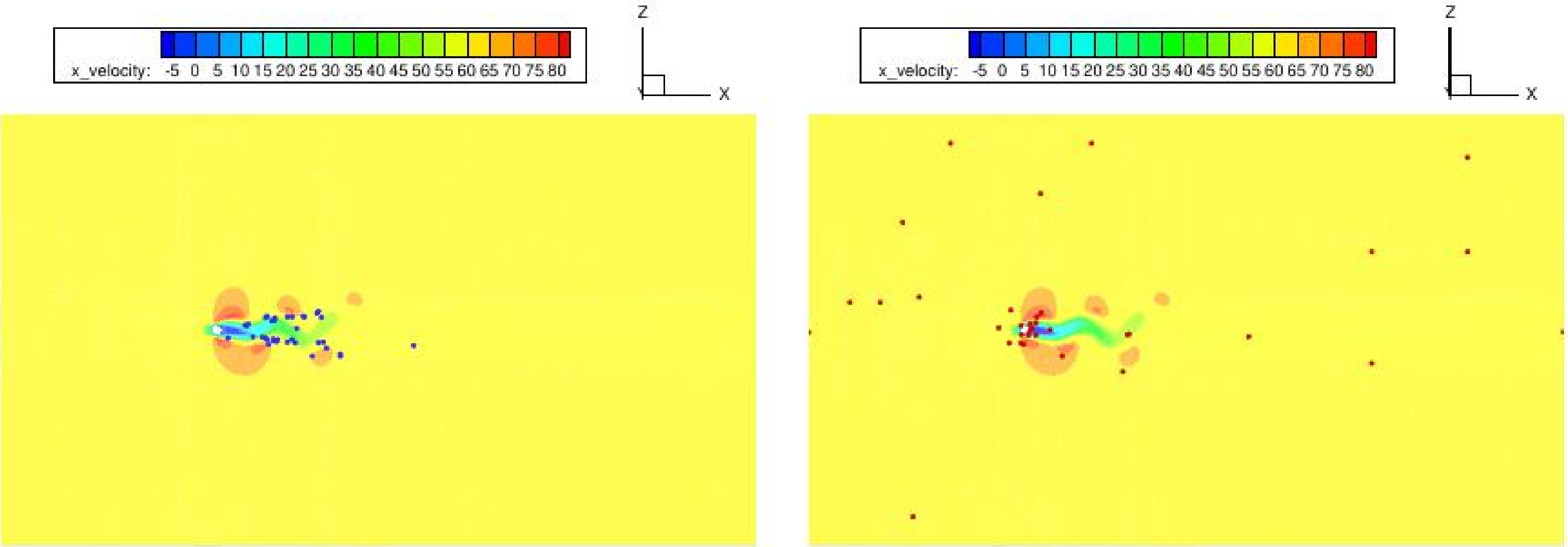
$Re_D = 60$ cylinder flow: Normalized errors ε_f (Eq. (12)) and ε_σ (Eq. (13)) committed on capturing the top 4 frequencies f_i and corresponding growth rate σ_i with different clustering algorithms over spatial reduction $\widetilde{n_p}/n_p < 1$.

Database: Re_D=60 flow around a cylinder



$Re_D = 60$ cylinder flow: time invested (in seconds), for different clustering algorithms (averaged over 10 realizations).

Database: Re_D=60 flow around a cylinder

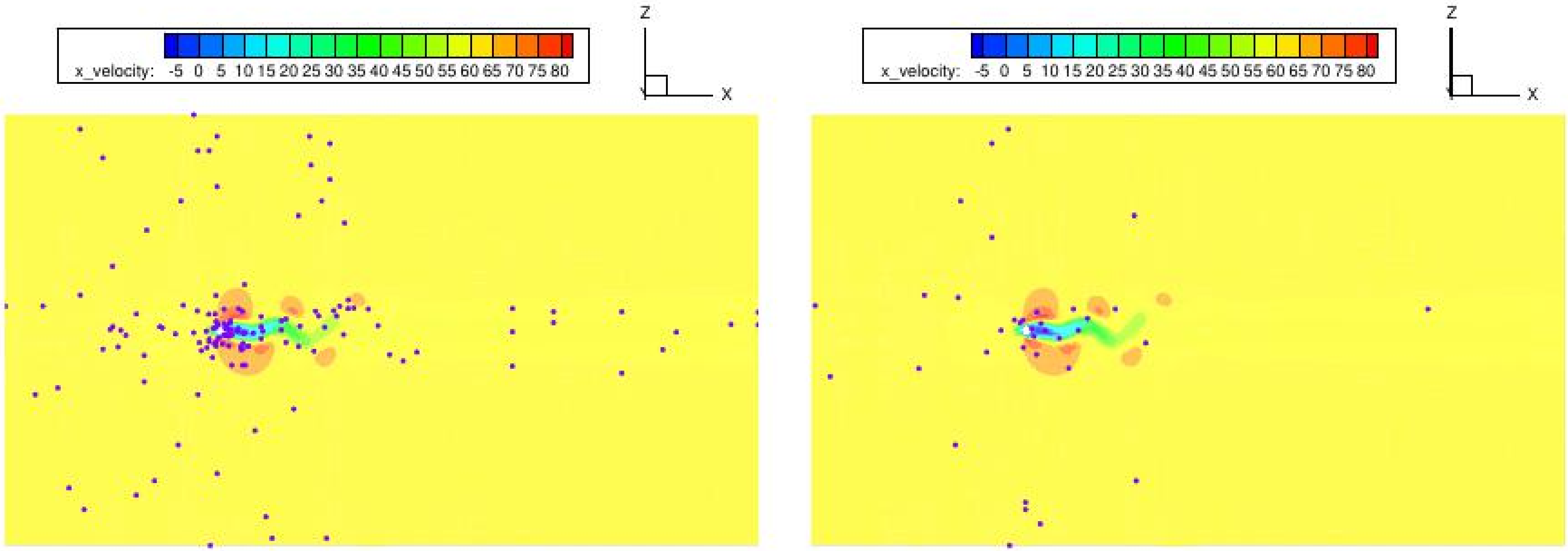


(a) K-means, $\widetilde{n_p}/n_p = 0.1\%$ ($\varepsilon_f = 4.26 \times 10^{-3}, \varepsilon_\sigma = 4.23 \times 10^{-3}$).

(b) DBSCAN, $\widetilde{n_p}/n_p = 0.1\%$ ($\varepsilon_f = 0.31, \varepsilon_\sigma = 5.37$).

$Re_D = 60$ cylinder flow: The distribution of centroids/cores from different clustering algorithms, with spatial reduction $\widetilde{n_p}/n_p < 0.4\%$.

Database: Re_D=60 flow around a cylinder

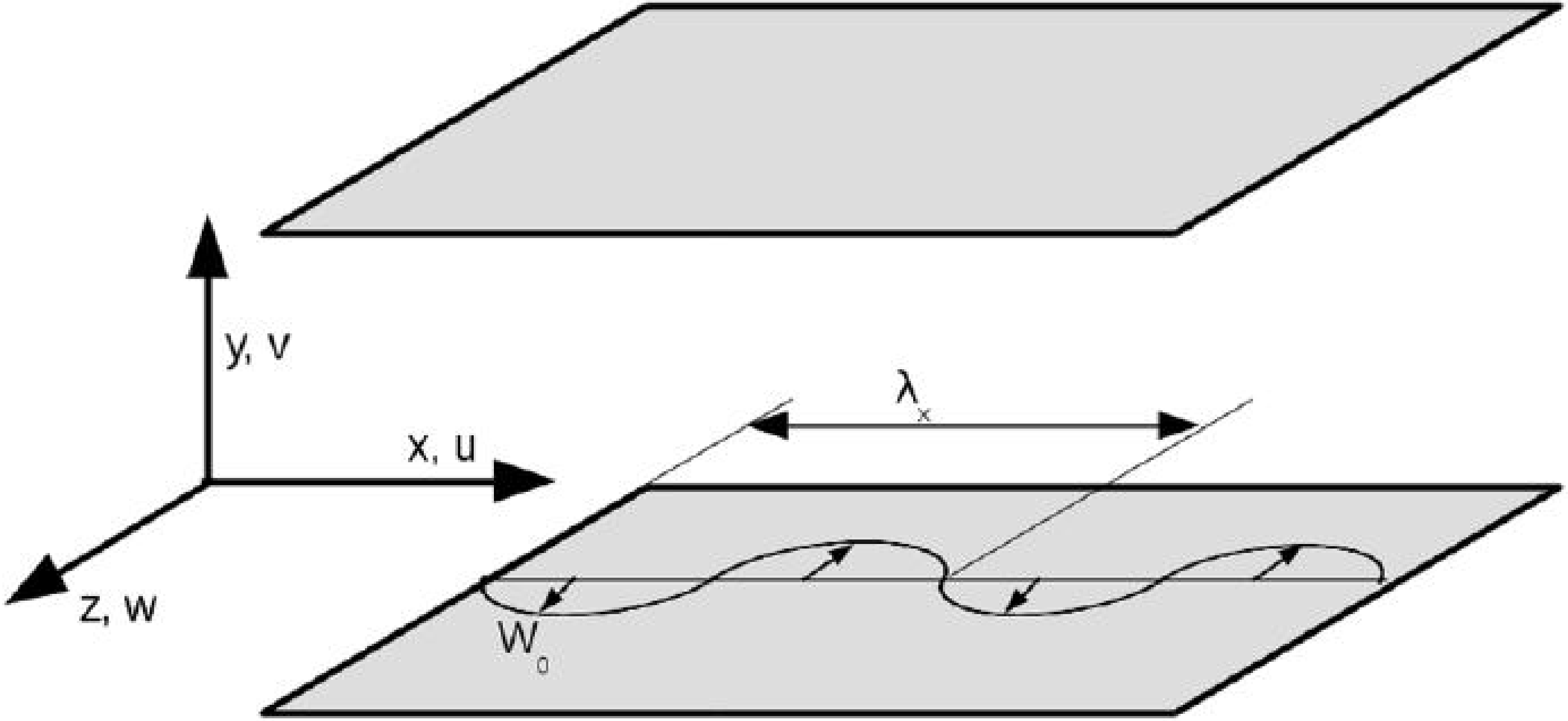


(c) Gaussian Mixture, $\widetilde{n_p}/n_p = 0.4\%$ ($\varepsilon_f = 4.20 \times 10^{-3}, \varepsilon_\sigma = 5.02 \times 10^{-3}$)

(d) Gaussian Mixture, $\widetilde{n_p}/n_p = 0.1\%$ ($\varepsilon_f = 0.55, \varepsilon_\sigma = 14.5$).

$Re_D = 60$ cylinder flow: The distribution of centroids/cores from different clustering algorithms, with spatial reduction $\widetilde{n_p}/n_p < 0.4\%$.

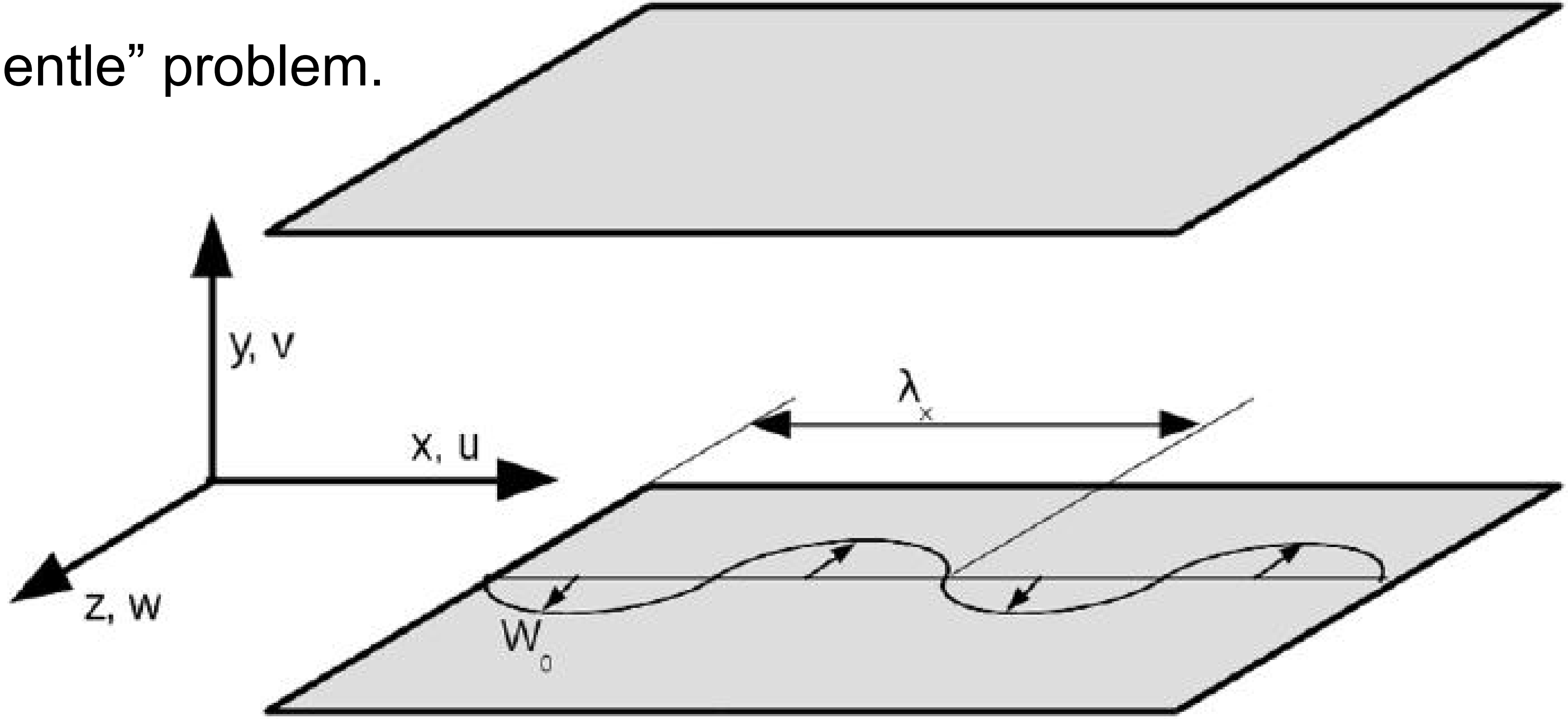
Database: $Re_c \approx 3600$ turbulent channel flow



Domain and system of reference for the channel flow problem. The domain is periodic along *x* and *z* directions; bulk flow is along *x* direction.

Database: $Re_c \approx 3600$ turbulent channel flow

Not so "gentle" problem.

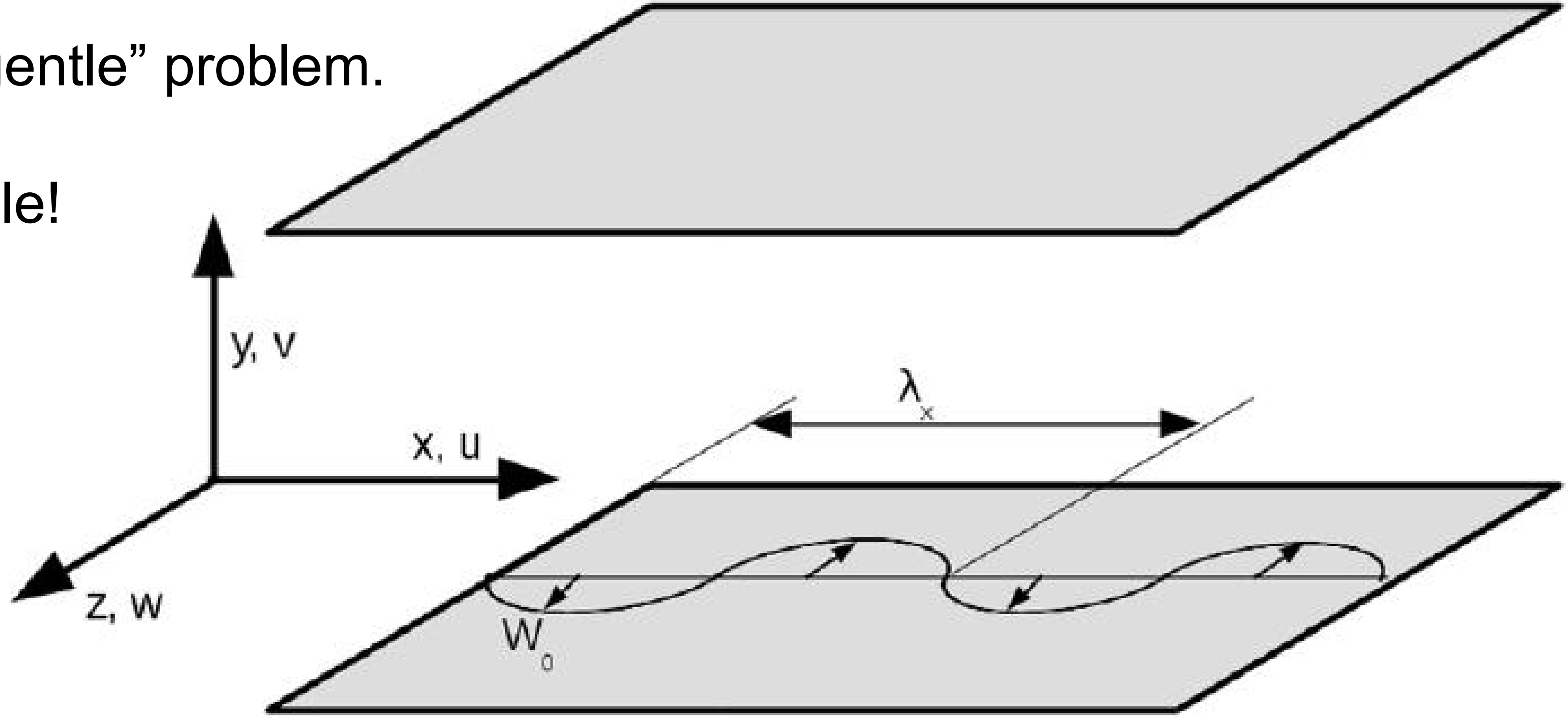


Domain and system of reference for the channel flow problem. The domain is periodic along *x* and *z* directions; bulk flow is along *x* direction.

Database: $Re_c \approx 3600$ turbulent channel flow

Not so "gentle" problem.

Multi-scale!



Domain and system of reference for the channel flow problem. The domain is periodic along *x* and *z* directions; bulk flow is along *x* direction.

Database: Re_c≈3600 turbulent channel flow

	L_x/δ	L_y/δ	L_z/δ	n_x	n_y	n_z	Re_c	u_c	W_0/u_c	λ_x/L_x	u_τ
Standard	π	2	$\pi/2$	96	101	96	3678.7	0.7699	...	...	0.042 33
Actuated							3732.7	0.7812	0.5	1	0.031 36

Forcing	Snapshots stored n_s	Δt^s	Memory (GB)
Constant flow rate	1200	0.156 25	32

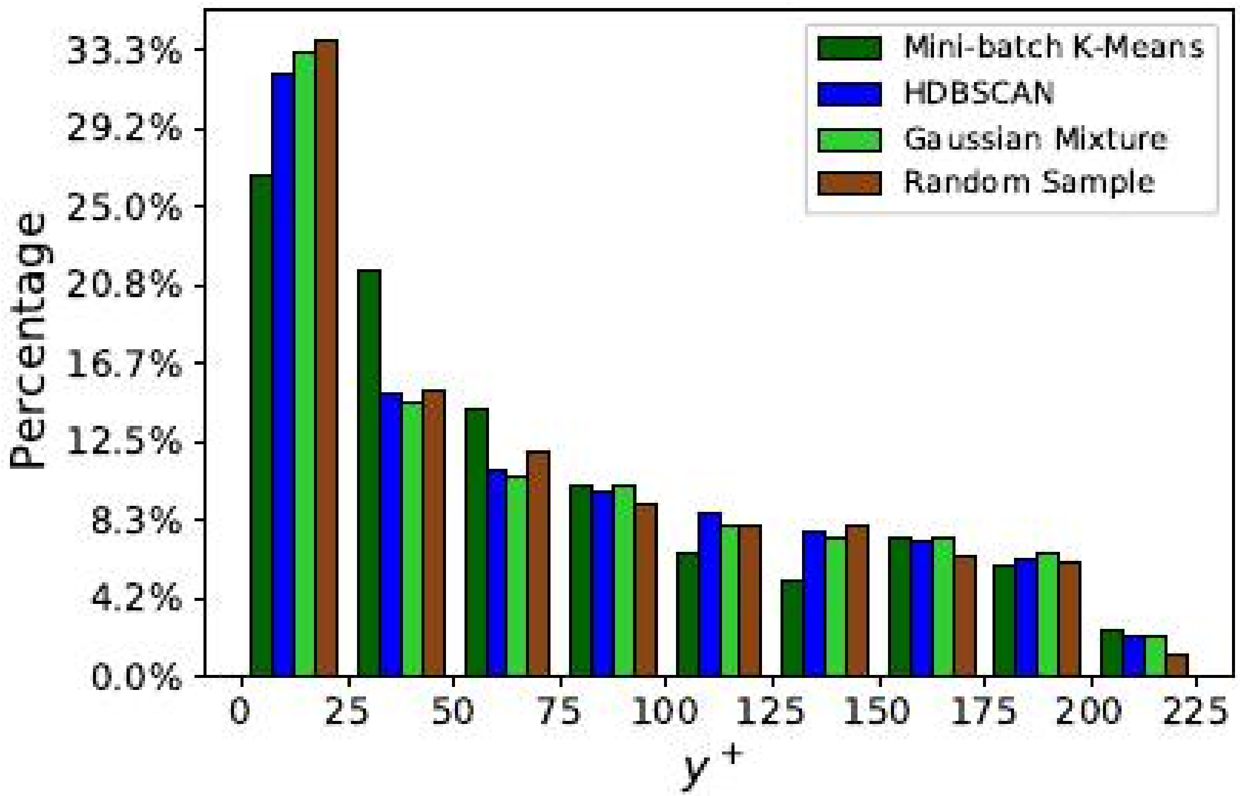
Database: Re_c≈3600 turbulent channel flow

"Small" database

	L_x/δ	L_y/δ	L_z/δ	n_x	n_y	n_z	Re_c	u_c	W_0/u_c	λ_x/L_x	u_τ
Standard	$\boxed{\pi}$	2	$\boxed{\pi/2}$	96	101	96	3678.7	0.7699	...	...	0.042 33
Actuated							3732.7	0.7812	0.5	1	0.031 36

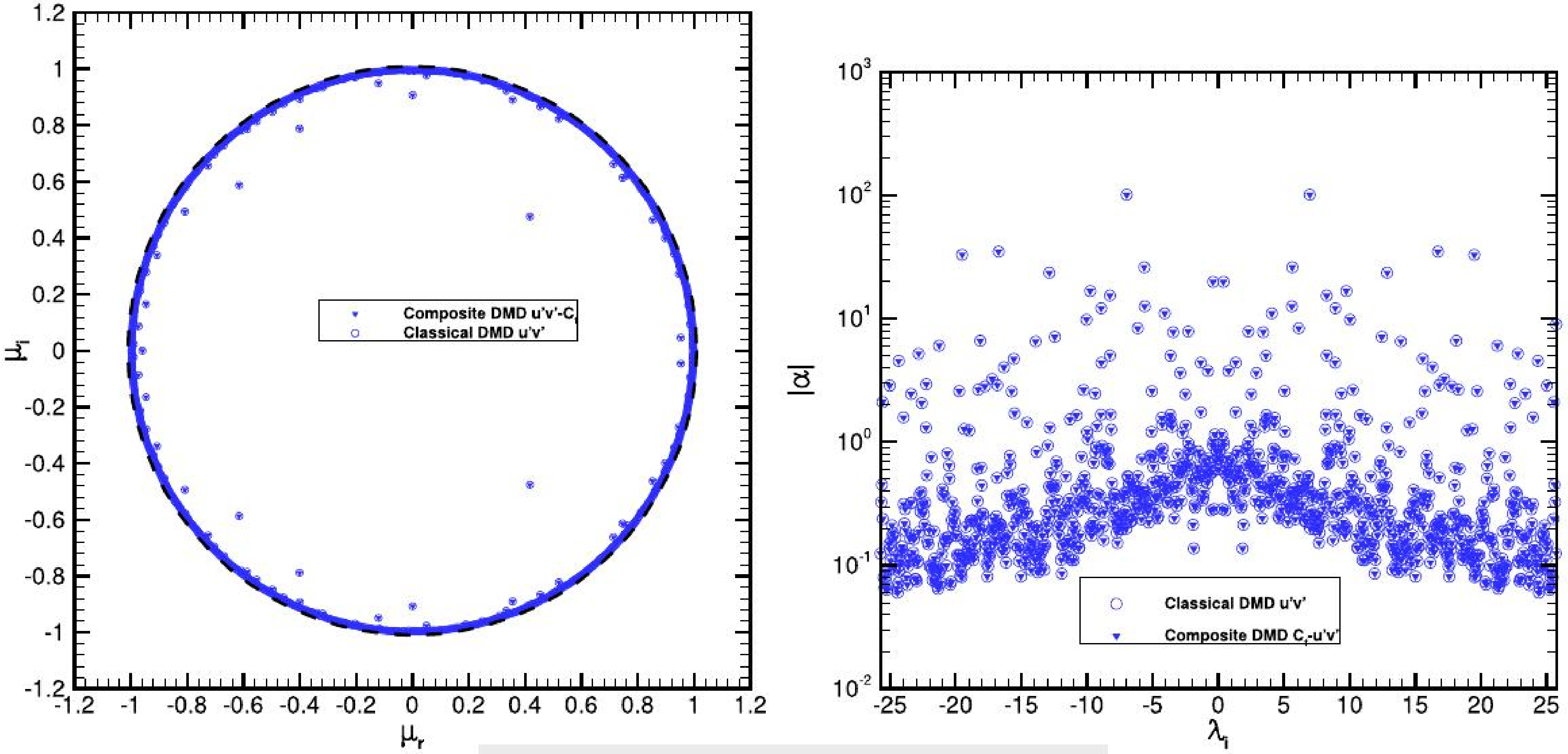
Forcing	Snapshots stored n_s	Δt^s	Memory (GB)
Constant flow rate	1200	0.156 25	$\boxed{32}$

Database: $Re_c \approx 3600$ turbulent channel flow



$Re_\tau = 200$ turbulent channel flow: distribution of centroids along y^+ for different agglomeration strategies.

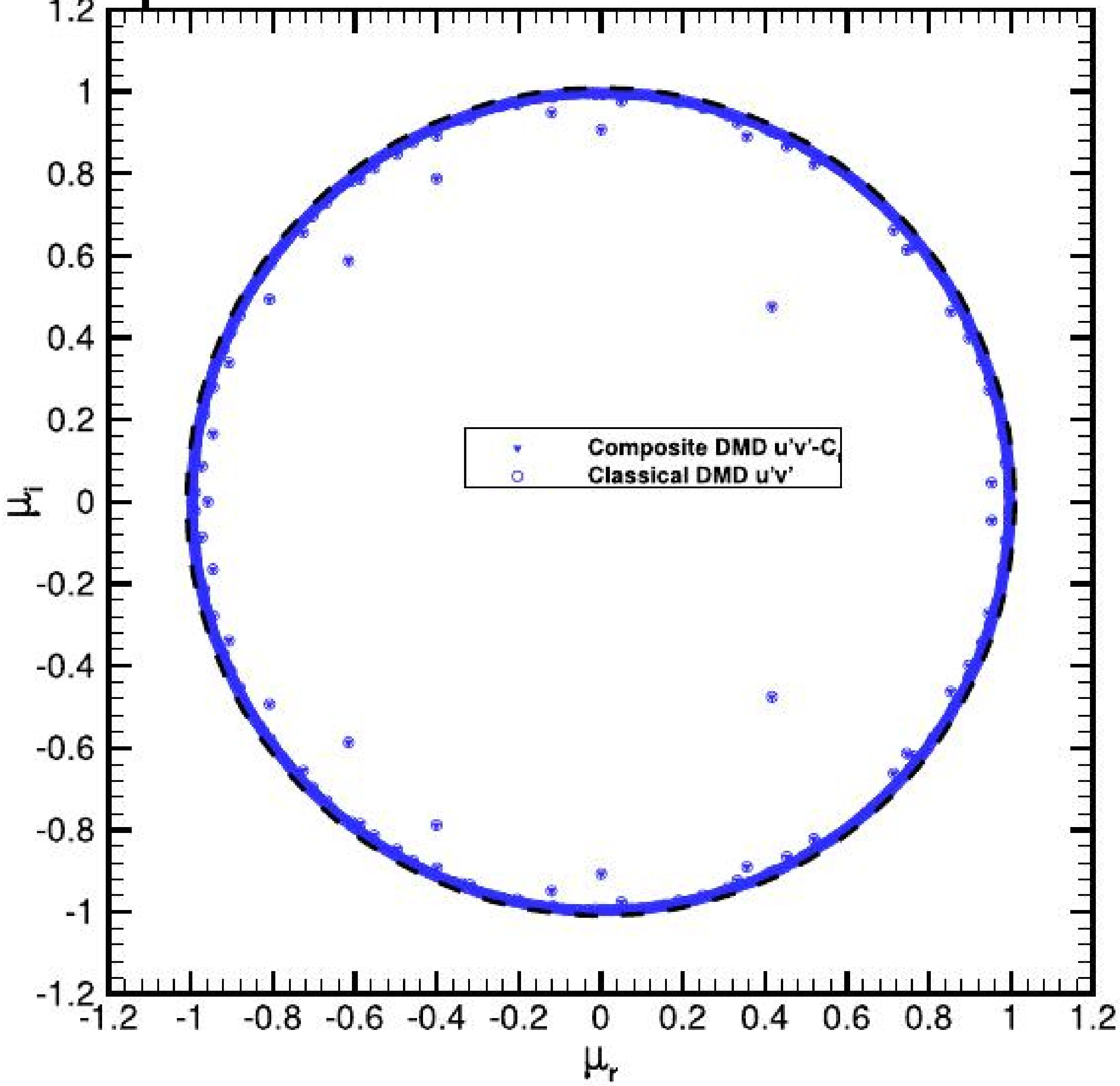
Re_c≈3600 TCF Composite DMD



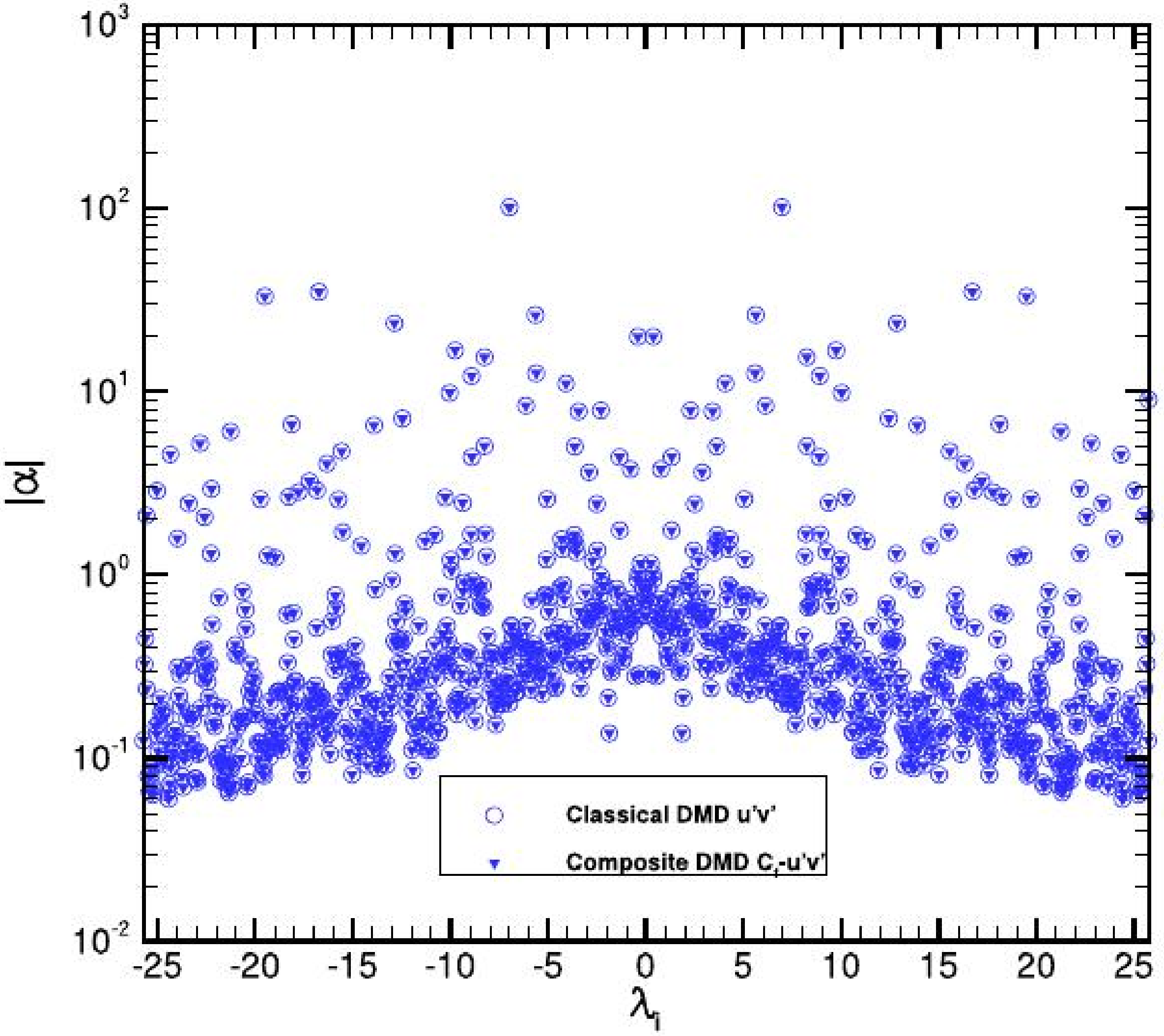
(a) (b)

Standard channel DMD spectra obtained from analysis based on $u'v'$ and composite C_f-$u'v'$ snapshots: (a) μ-plane representation with locus $|\mu| = 1$ in dashed line, and (b) amplitude $|\alpha_i|$ vs angular pulsation $\Im(\lambda_i)$.

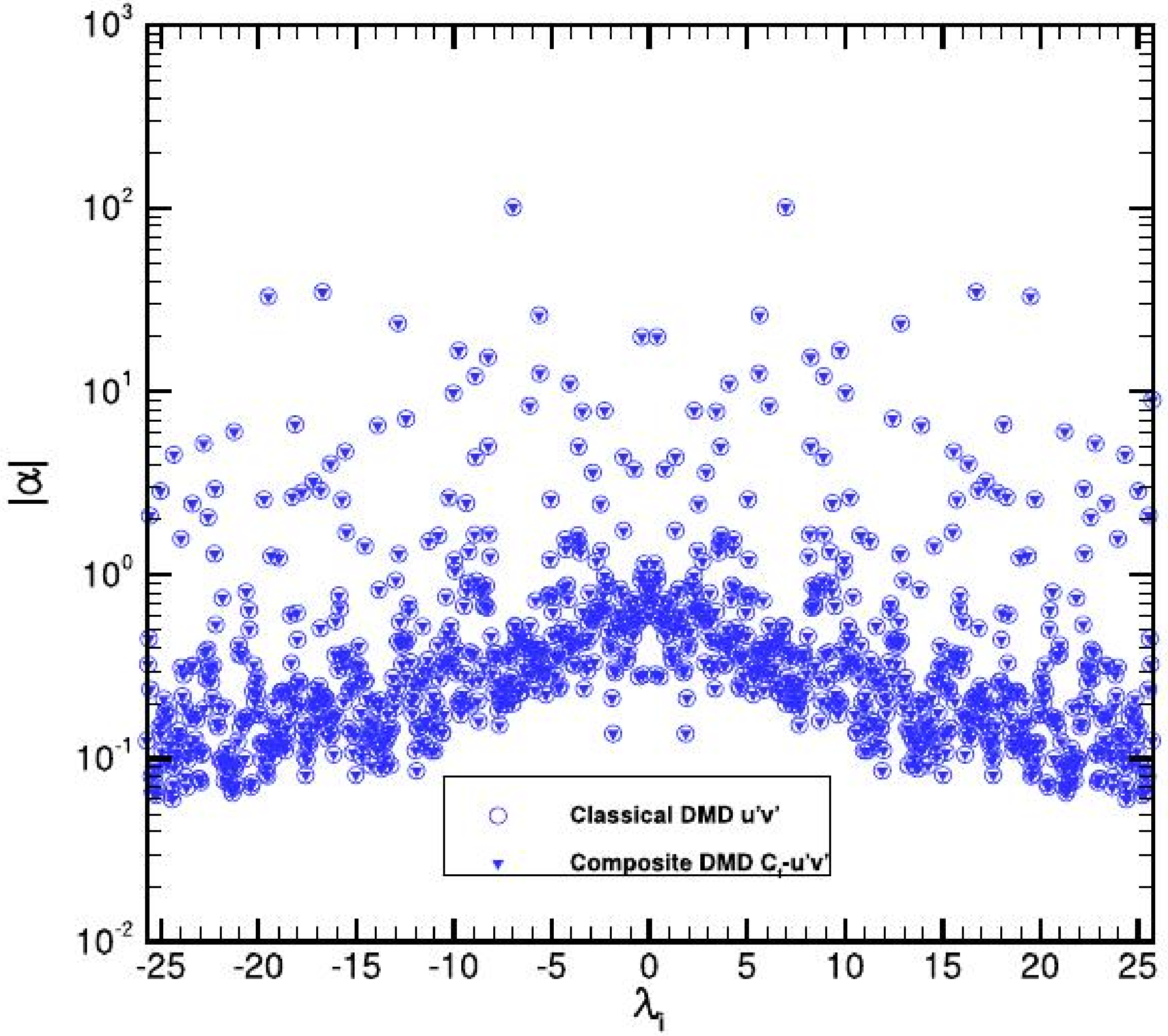
Re_c≈3600 TCF Composite DMD



$Re_c \approx 3600$ TCF Composite DMD



$Re_c \approx 3600$ TCF Composite DMD

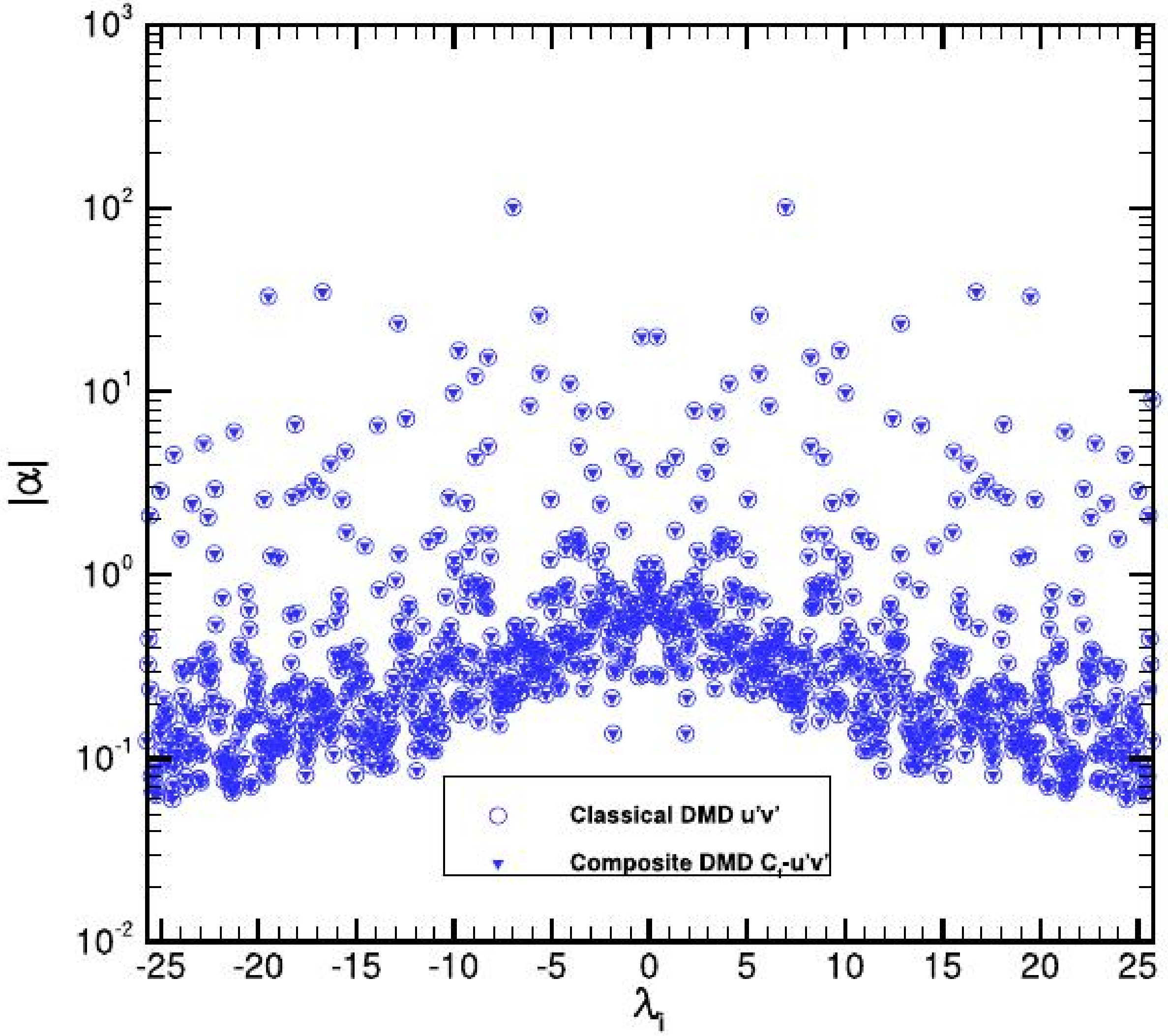


Choose those modes that have largest α_i

and use

$$\mathbf{v}(\vec{r},t) = \sum_{j=1}^{n_t-1} \alpha_j \, \Phi_j(\vec{r}) \, e^{\lambda_j t} \text{ with } \lambda_j = \frac{\log \mu_j}{\Delta t}$$

Re_c≈3600 TCF Composite DMD



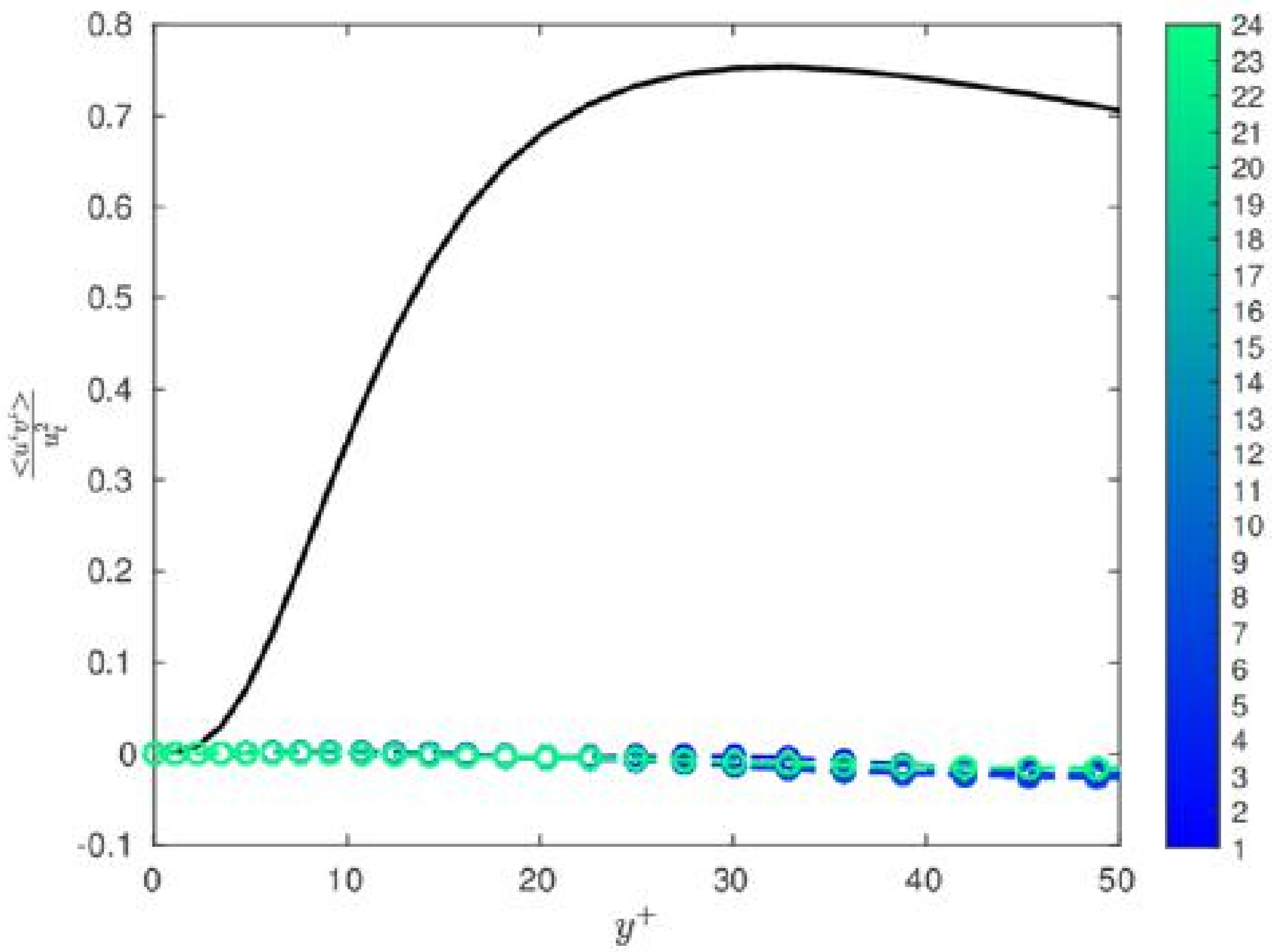
Choose those modes that have largest α_i

and use

$$\mathbf{v}(\vec{r},t) = \sum_{j=1}^{\overset{r_1}{\cancel{n_t-1}}} \alpha_j \, \Phi_j(\vec{r}) \, e^{\lambda_j t} \text{ with } \lambda_j = \frac{\log \mu_j}{\Delta t}$$

$$V = \mathbf{\Phi} \, D_{\alpha} \, M_{\mu}$$

$Re_c \approx 3600$ TCF Composite DMD



Composite DMD

Temporal data sequence acquired at a fixed sampling frequency $1/\Delta t$

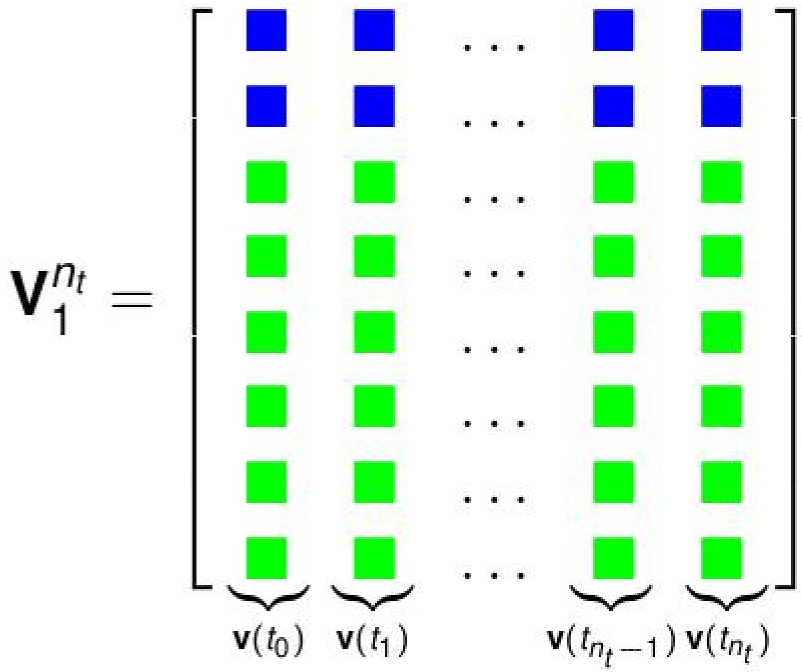
Arrange data in a (typically) tall and skinny matrix

$$\mathbf{V}_1^{n_t} = \left[\begin{array}{ccccc} \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \cdots & \blacksquare & \blacksquare \\ \underbrace{\blacksquare}_{\mathbf{v}(t_0)} & \underbrace{\blacksquare}_{\mathbf{v}(t_1)} & \cdots & \underbrace{\blacksquare}_{\mathbf{v}(t_{n_t-1})} & \underbrace{\blacksquare}_{\mathbf{v}(t_{n_t})} \end{array} \right]$$

P.J. Schmid, J. Fluid Mech., vol. 656, 2010

Composite DMD

Temporal data sequence acquired at a fixed sampling frequency $1/\Delta t$
Instantaneous data groups two (or more) different variables
into single column vector $\mathbf{v}(t_k)$
Arrange data in a (typically) tall and skinny matrix



T. Sayadi *et al.* , J. Fluid Mech., vol. 748, 2014

Composite DMD

$$C_f = \underbrace{\frac{12}{Re_b}}_{\text{Laminar}} + \underbrace{12 \int_0^1 2\,(1-y) < -u'\,v' > \,d\,y}_{\text{Turbulent}}$$

Applies to fully developed turbulent channel

K. Fukagata, K. Iwamoto and N. Kasagai, Phys. Fluids vol. 14, 2002

Composite DMD

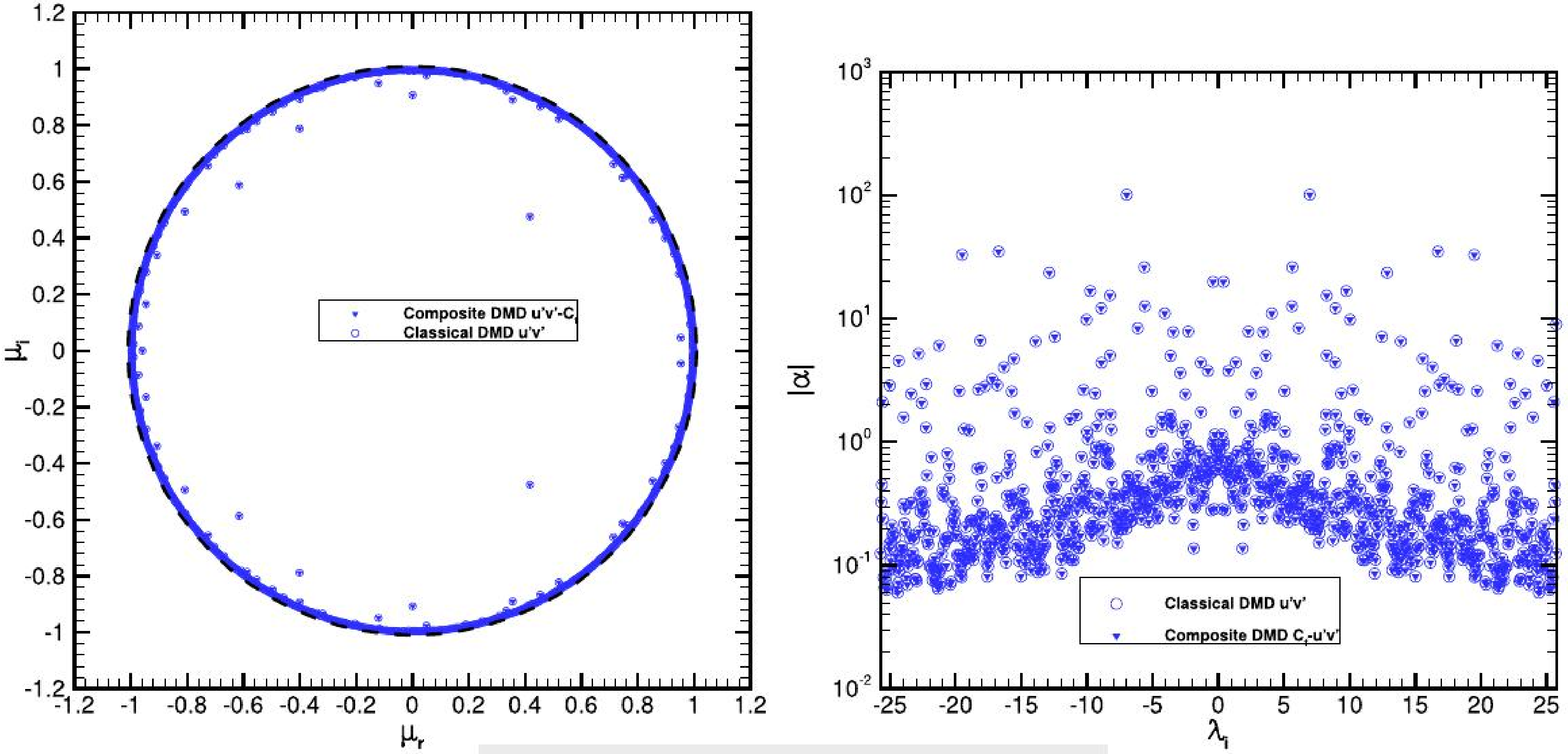
Analysis on composed $C_f(t_k)$& $u'v'(r, t_k)$ fields

$$\beta_i \equiv (\phi_i \cdot \mathbf{e}_{C_f})\alpha_i$$

$$\langle u'v' \rangle^{DMD}(y) = \frac{1}{n_s \Delta t^s} \sum_{i=1}^{n_r} \alpha_i \langle \phi_i - (\phi_i \cdot \mathbf{e}_{C_f})\mathbf{e}_{C_f} \rangle \int_0^{n_s \Delta t^s} e^{\lambda_i t} dt.$$

Garicano-Mena, Li, Ferrer & Valero, A composite dynamic mode decomposition analysis of turbulent channel flows, Phys. Fluids, vol. 31, 2019.

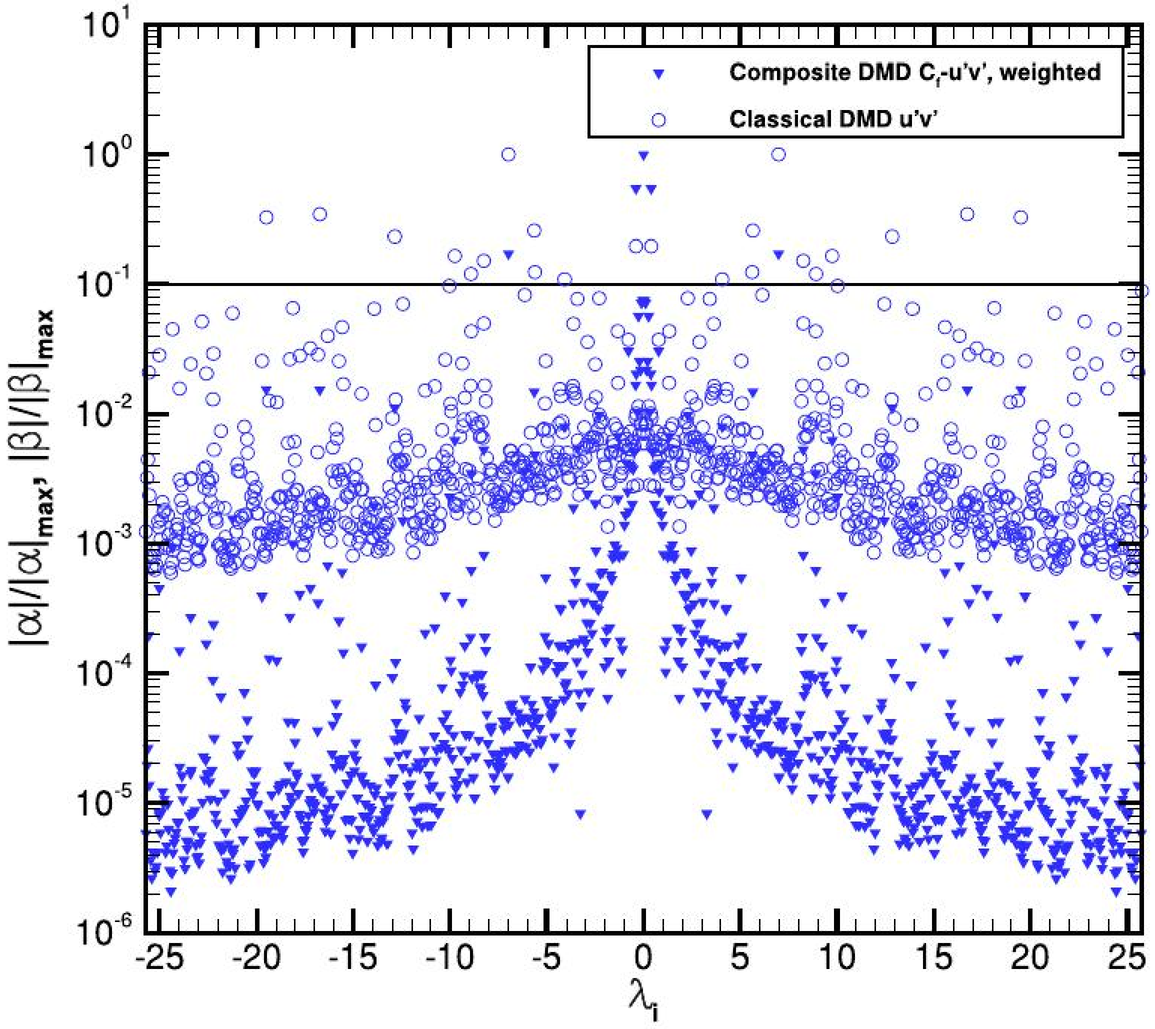
Re$_c$≈3600 TCF Composite DMD



(a) (b)

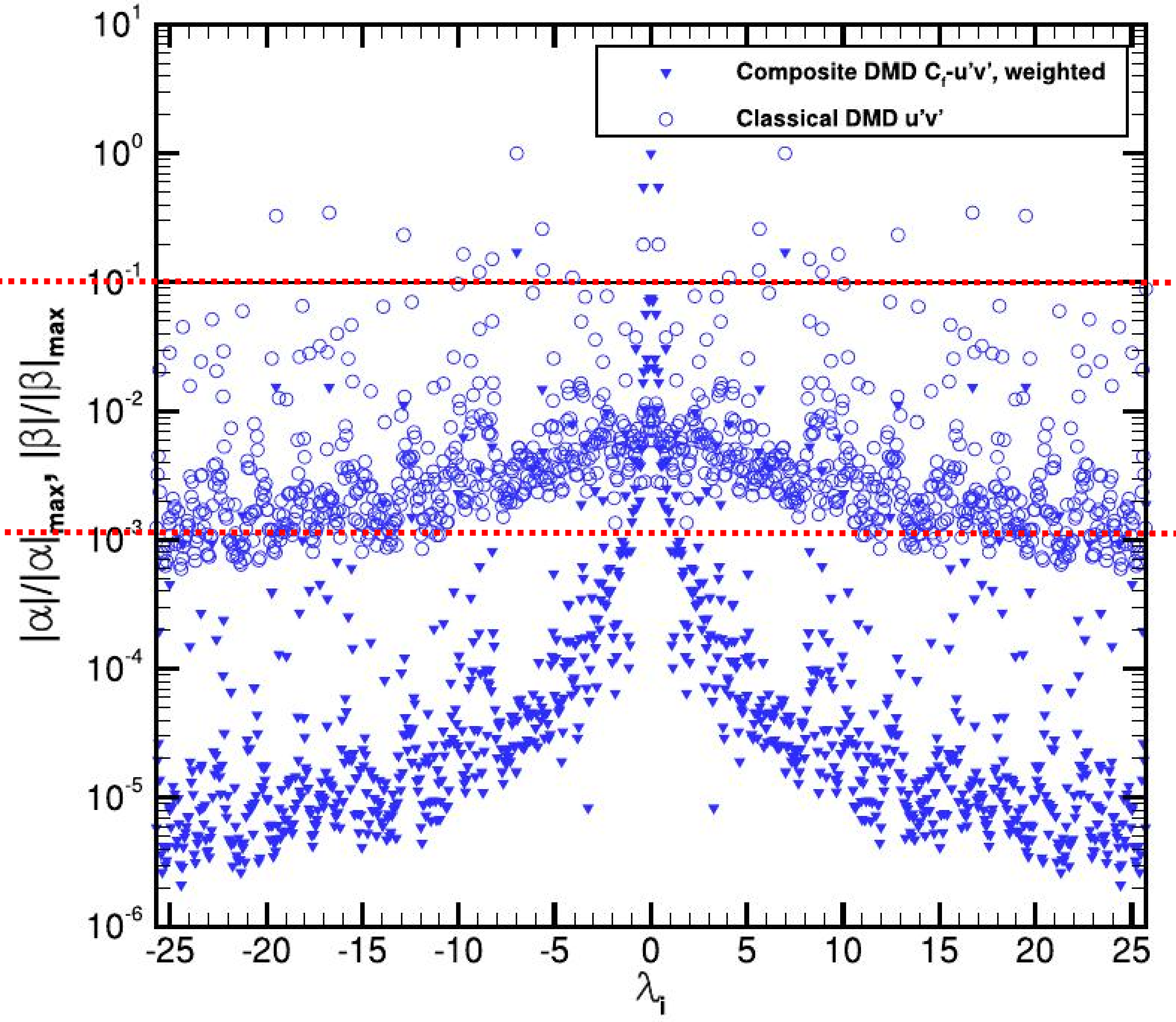
Standard channel DMD spectra obtained from analysis based on $u'v'$ and composite C_f-$u'v'$ snapshots: (a) μ-plane representation with locus $|\mu| = 1$ in dashed line, and (b) amplitude $|\alpha_i|$ vs angular pulsation $\Im(\lambda_i)$.

Composite Parallel DMD



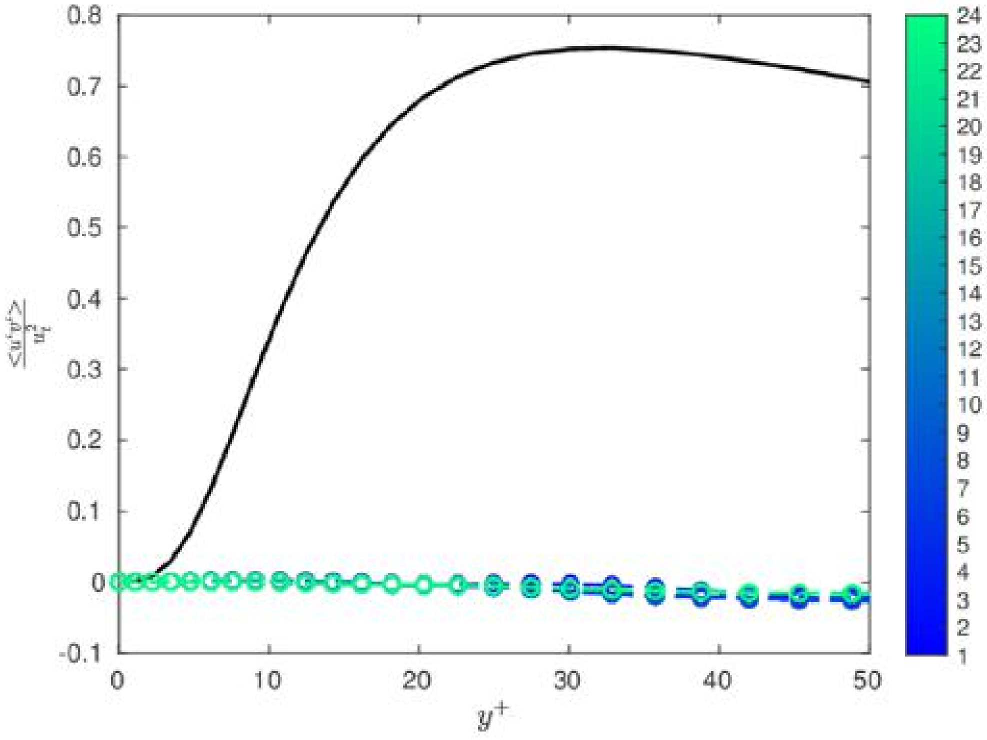
Standard channel DMD spectra obtained from analysis based on $u'v'$ and composite C_f-$u'v'$ snapshots: amplitudes $|\alpha_i|/|\alpha_{max}|$ and $|\beta_i|/|\beta_{max}|$ vs angular pulsation $\Im(\lambda_i)$.

Composite Parallel DMD



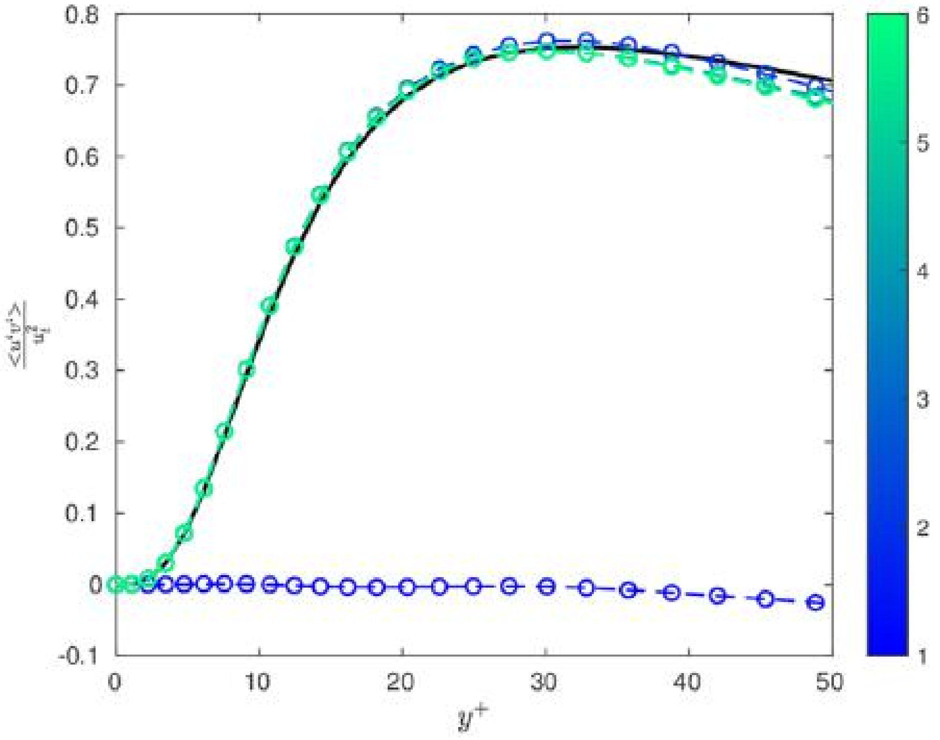
Standard channel DMD spectra obtained from analysis based on $u'v'$ and composite C_f-$u'v'$ snapshots: amplitudes $|\alpha_i|/|\alpha_{max}|$ and $|\beta_i|/|\beta_{max}|$ vs angular pulsation $\Im(\lambda_i)$.

Composite Parallel DMD



(a) Classical **DMD**, expansion w/

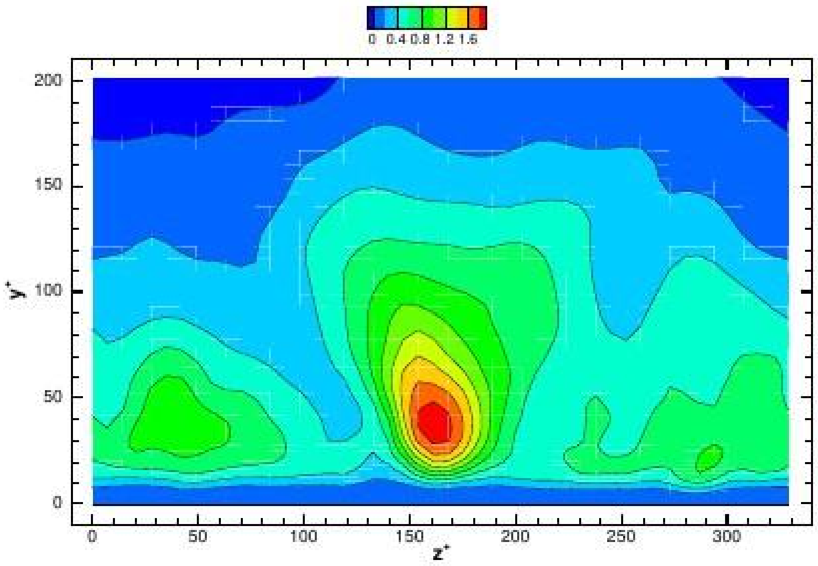
$\frac{|\alpha_i|}{|\alpha_{max}|} > 10\%,\ n_r = 24.$



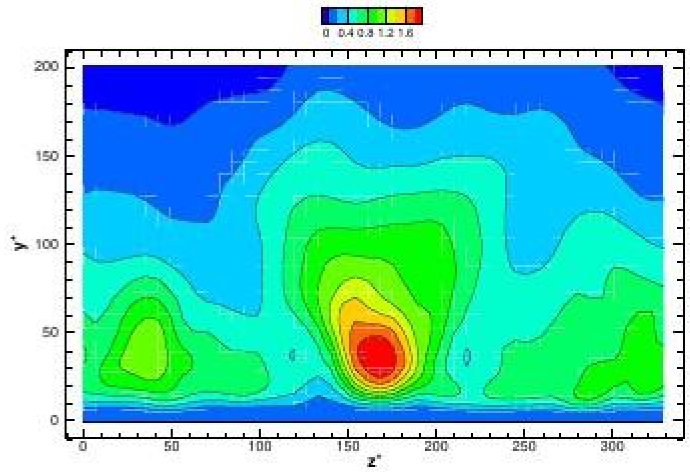
(b) Composite **DMD**, expansion w/

$\frac{|\beta_i|}{|\beta_{max}|} > 10\%,\ n_r = 6\ .$

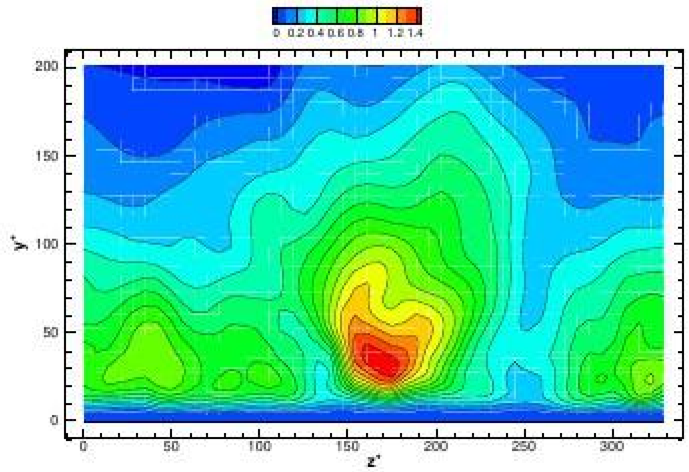
Database: Re_c≈3600 turbulent channel flow



(a) Complete database.



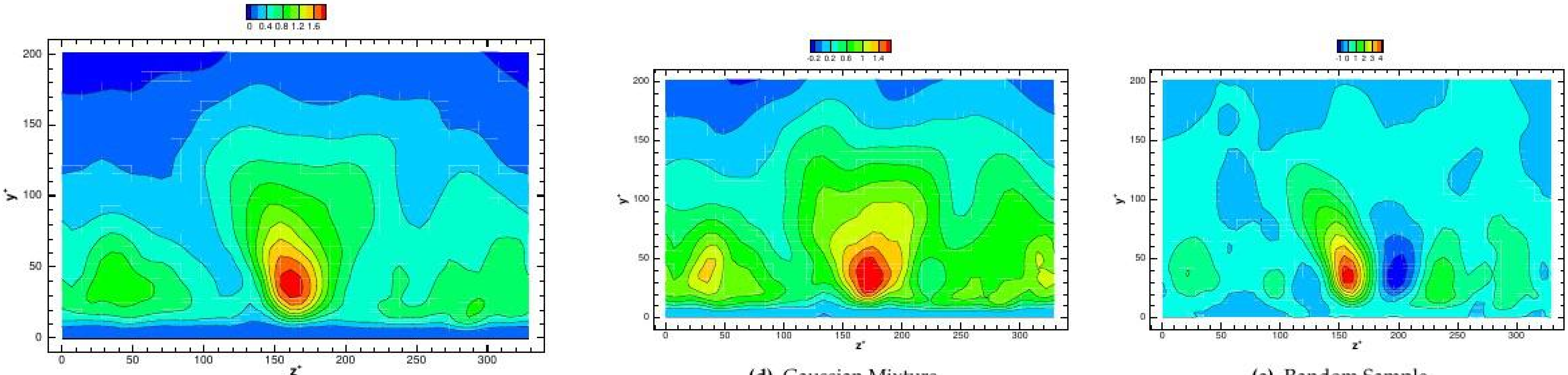
(b) Mini-Batch K-means.



(c) HDBSCAN.

$Re_\tau = 200$ turbulent channel flow: **DMD** reconstruction of the field $-\frac{\langle u'v' \rangle}{u_\tau^2}(y^+, z^+)$

Database: $Re_c \approx 3600$ turbulent channel flow



(a) Complete database.

(d) Gaussian Mixture.

(e) Random Sample.

$Re_\tau = 200$ turbulent channel flow: **DMD** reconstruction of the field $-\frac{\langle u'v' \rangle}{u_\tau^2}(y^+, z^+)$

Spatial Agglomeration & DMD

DMD analysis applied on Spatially Agglomerated Databases

allows effective reduction of computational costs

while retaining accurate results.

Spatial Agglomeration & DMD

DMD analysis applied on Spatially Agglomerated Databases

allows effective reduction of computational costs

while retaining accurate results.

Succesful example of combination of

(Unsupervised) Machine Learning Algorithms

&

data-driven modal decomposition techniques.

Li, Garicano-Mena, Zheng & Valero, Dynamic Mode Decomposition Analysis of Spatially Agglomerated Flow Databases, Energies, vol. 13 (9), 2020.

Conclusions

Conclusions

Data-driven modal decomposition techniques $\rightleftharpoons$ matrix factorization:

SVD is univoquely defined,
but mixes frequencies.

DMD identifies distinct frequencies,
but needs solving an optimization problem.

Conclusions

Data-driven modal decomposition techniques $\rightleftharpoons$ matrix factorization:

SVD (POD) is univoquely defined,
but mixes frequencies.

DMD identifies distinct frequencies,
but needs solving an optimization problem.
Fortunately, a closed solution is available.

Conclusions

Computational cost of data-driven modal decomposition analysis:

Potentially expensive & large memory footprint.

Conclusions

Computational cost of Data-driven modal decomposition analysis:

Potentially expensive & large memory footprint.

Strategies to alleviate the computational cost available:
- Spatial Agglomeration,
- Memory-distributed parallelism (not discussed today).

Conclusions

More sophisticated variants available:

Higher-Order SVD (tensor formulation):
very robust against noise.

Conclusions

More sophisticated variants available:

Higher-Order SVD (tensor formulation):
very robust against noise.

Higher-Order DMD:
uses a generalized, d-lagged Koopman assumption
capable of unravel very complex dynamics.

$$\mathbf{v}(t_{k+1}) = \mathcal{A}\,\mathbf{v}(t_k)$$

Conclusions

More sophisticated variants available:

Higher-Order SVD (tensor formulation):
very robust against noise.

Higher-Order DMD:
uses a generalized, d-lagged Koopman assumption
capable of unravel very complex dynamics.

$$\mathbf{v}(t_{k+1}) = \mathcal{A}\,\mathbf{v}(t_k)$$

$$\mathbf{v}(t_{k+d}) = \mathcal{A}_1\,\mathbf{v}(t_k) + \mathcal{A}_2\,\mathbf{v}(t_{k+1}) \ldots + \mathcal{A}_d\,\mathbf{v}(t_{k+d-1})$$

Conclusions

More sophisticated variants available:

Higher-Order SVD (tensor formulation):
very robust against noise.

Higher-Order DMD:
uses a generalized, d-lagged Koopman assumption
capable of unravel very complex dynamics

Applications also beyond fluid dynamics,
e.g. Medical Imaging (echocardiography videos, MRI data)

Groun, Le Clainche et al., Higher order dynamic mode decomposition: From fluid dynamics to heart disease analysis, Computers in Biology and Medicine, Vol 144, 2022.
Groun, Le Clainche et al., A novel data-driven method for the analysis and reconstruction of cardiac cine MRI, Computers in Biology and Medicine, Vol. 151, 2022.

Conclusions

More sophisticated variants available:

Higher-Order SVD (tensor formulation):
very robust against noise.

Higher-Order DMD:
uses a generalized, d-lagged Koopman assumption
capable of unravel very complex dynamics

Combinations with Neural Network technology.

Conclusions

More sophisticated variants available:

Higher-Order SVD (tensor formulation):
very robust against noise.

Higher-Order DMD:
uses a generalized, d-lagged Koopman assumption
capable of unravel very complex dynamics.

Combination with Neural Network technology.

No restriction on data origin.

Objectives of this talk?

1) Describe classical Dynamic Mode Decomposition (DMD)

2) DMD as one of many data-driven modal decomposition techniques

3) DMD as a matrix factorization technique

4) Accelerating DMD methods

Acknowledgements

- Grant TED2021-129774B-C21, funded by MCIN/AEI/10.13039/501100011033 and by the European Union "NextGenerationEU"/PRTR (DigitHEART)

- Grant PLEC2022-009235, funded by MCIN/AEI/10.13039/501100011033 (CardioAging)

Financiado por la Unión Europea NextGenerationEU

Plan de Recuperación, Transformación y Resiliencia

Thank you for your attention

Questions?