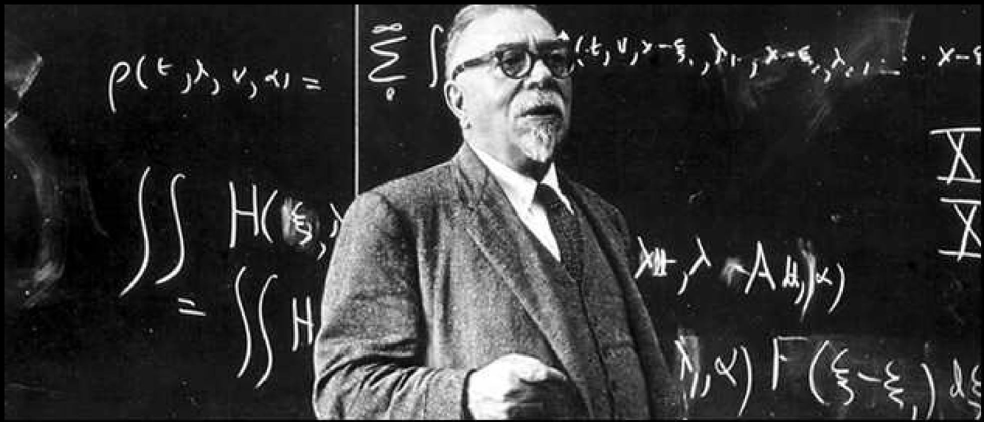
Mathematical Methods for Turbulence Control
Motivation & Overview



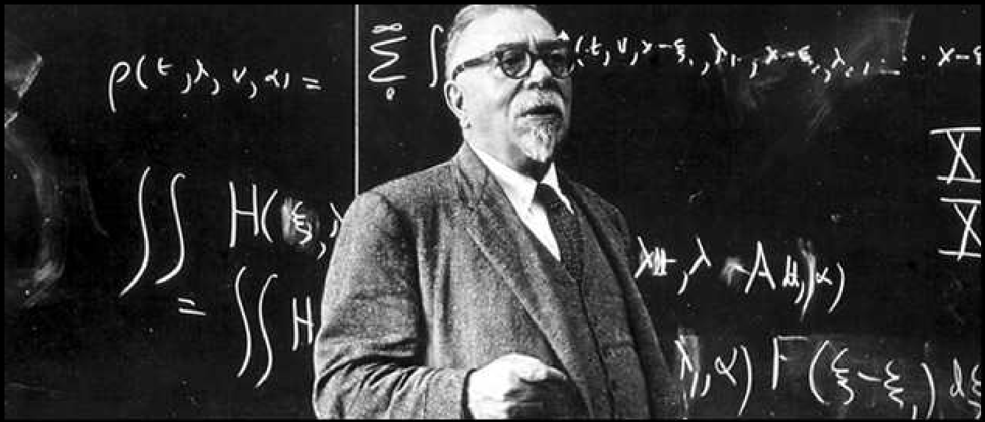
vs

Bernd Noack

HIT, China & TU Berlin

TU Berlin, Germany, 2020-03-02

Machine Learning
for Fluid Mechanics
Motivation & Overview



vs

Bernd Noack

HIT, China & TU Berlin

TU Berlin, Germany, 2020-03-02

My CV

Focus:	**Feedback turbulence control** **e.g. for cars, airplanes, combustion**
1985–1992	**Physics study + PhD (Göttingen)**
-1998	**PostDoc in Göttingen** (DLR, Uni, Max-Planck-Institut)
-2000	**United Techn. Research Center** (Think tank of Pratt, Carrier, Otis, ...)
-2009	**PostDoc $\mapsto$ Professor** TU Berlin, Germany
2010–2020	**Research Director at CNRS** Paris (LIMSI) & Poitiers (P')
2015–2019	**Professor, TU Braunschweig**
2017–now	**Professor, TU Berlin**
2020	**Professor, HIT**

Textbooks: **Reduced Order Modelling for Flow Control**
Machine Learning Control $\mapsto$ Turbulence

Audience

Language	☐ Mathematics
	☐ English
Level	☐ $\mapsto$Batchelor
	☐ $\mapsto$Master
	☐ $\mapsto$PhD + beyond
Study	☐ Engineering
	☐ Physics
	☐ Mathematics
	☐ Other
Background	☐ Fluid mechanics
(visited lectures)	☐ Control theory
	☐ Machine learning
Requested	☐ Navier-Stokes equations, ODE, PDE
Beneficial . .	☐ Any of my previous lectures

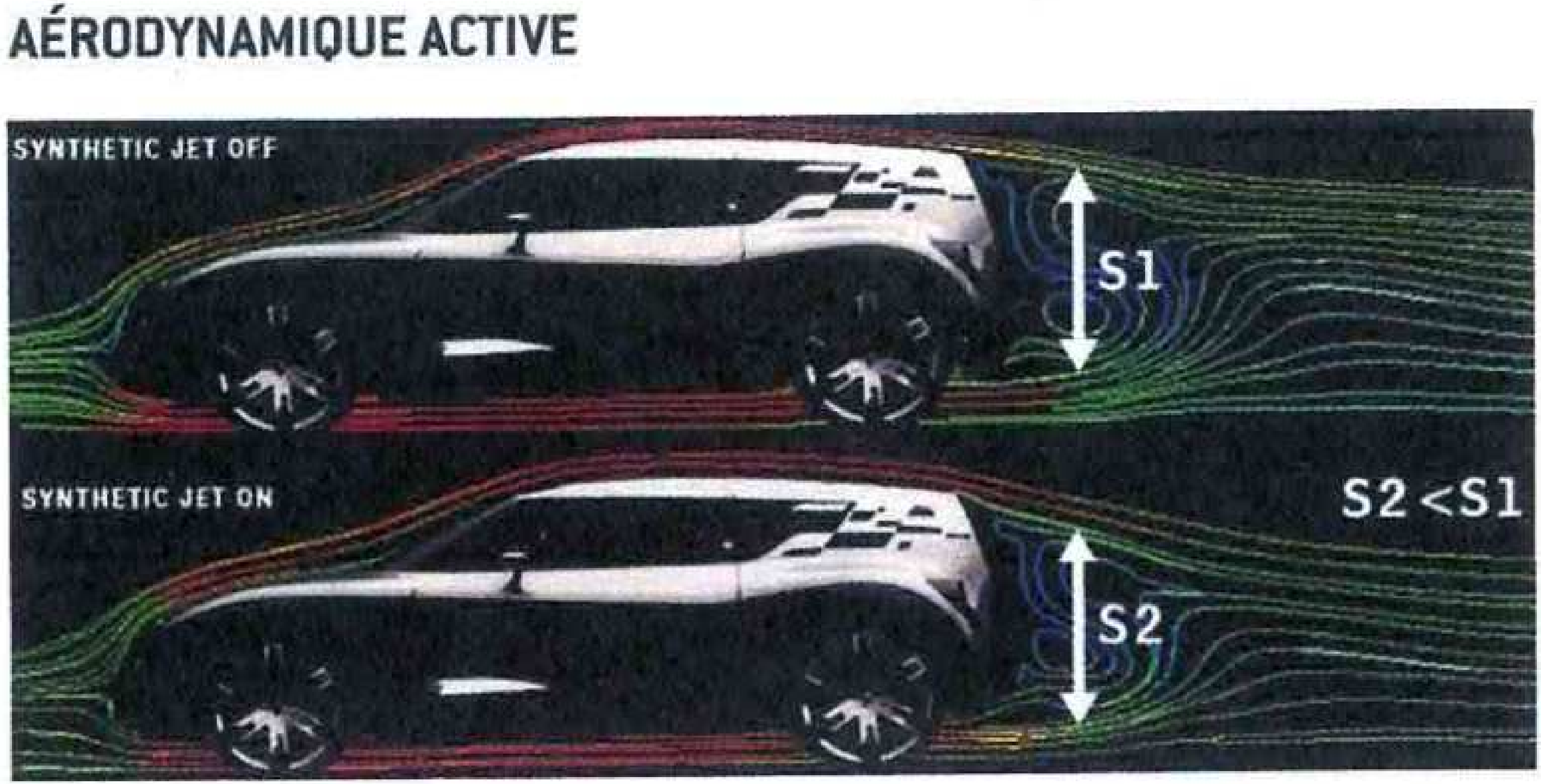
Turbulence control $\mapsto$ transport vehicles

Control strategies for drag reduction

- aerodynamic design
- passive (e.g. spoilers)
- active, open-loop
  (e.g. periodic blowing)
- active, closed-loop
  (largest opportunities!)

Renault Altica 2006 $\mapsto$

Renault Altica – Article in R & D 06/2004

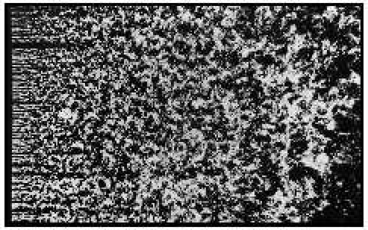
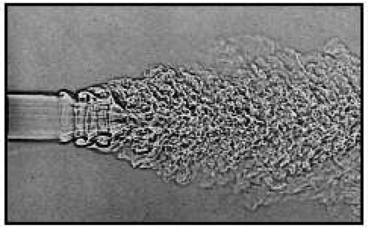


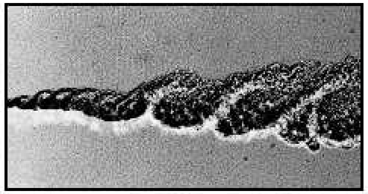
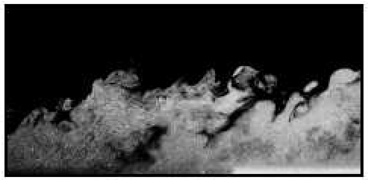
Active flow control with synthetic jets:

- 20% drag reduction at 90km/h;
- 1l fuel saving per 100 km;
- only 10 Watt actuation energy.

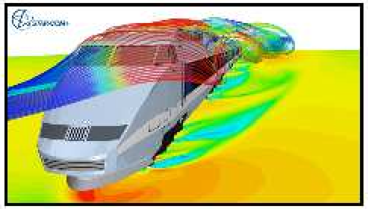
Turbulence control $\mapsto$ myriad applications

Simple prototype flows

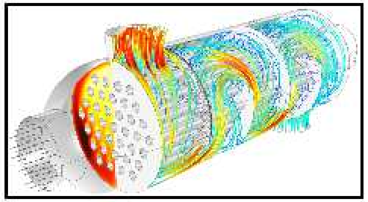
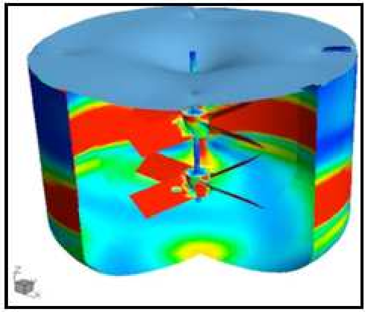
Transport vehicles

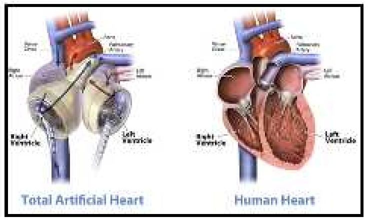




Energy systems




Production etc.

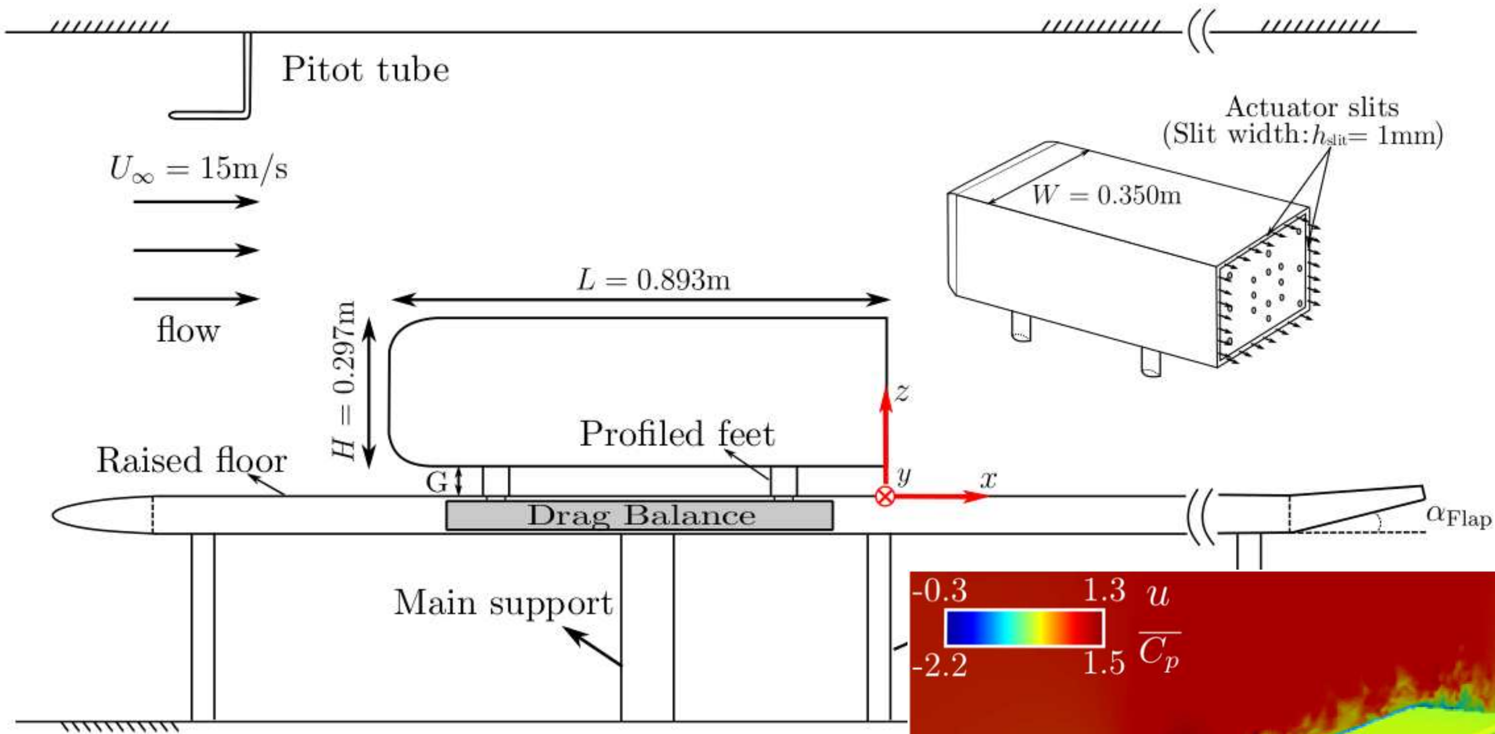



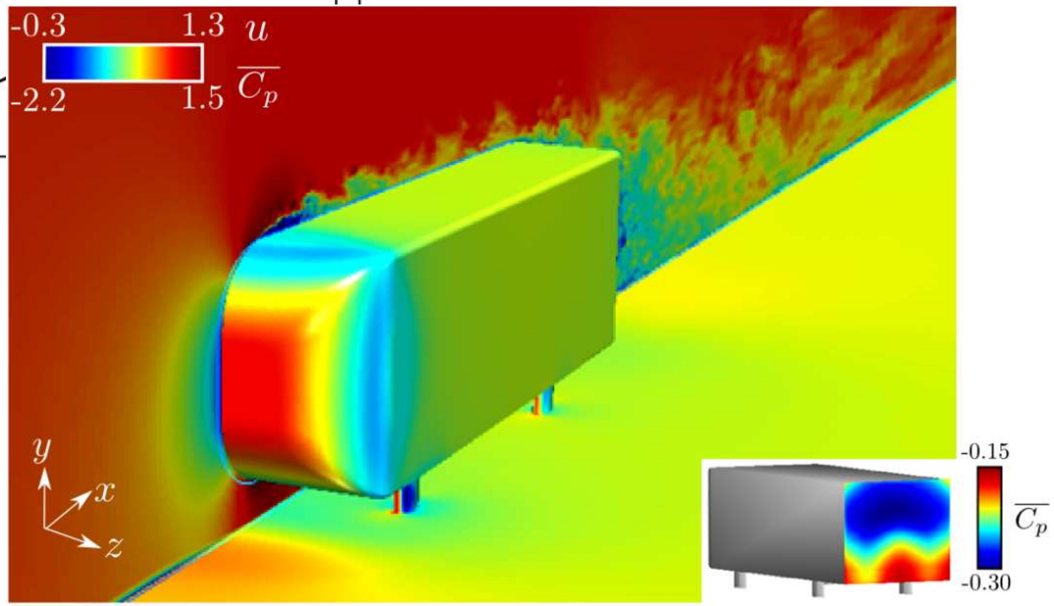




Paris

Drag reduction of simplied car model

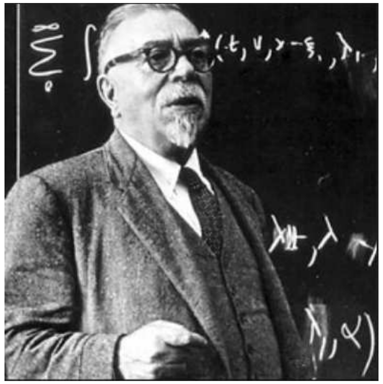
Barros, et al. 2016 JFM & Östh et al. 2014 JFM

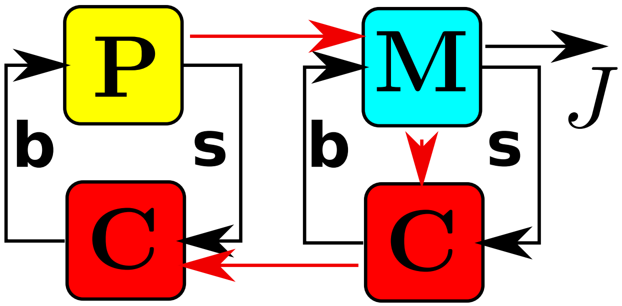




LES simulation
of same configuration →
Jan Östh et al. 2014 JFM

Model-based control





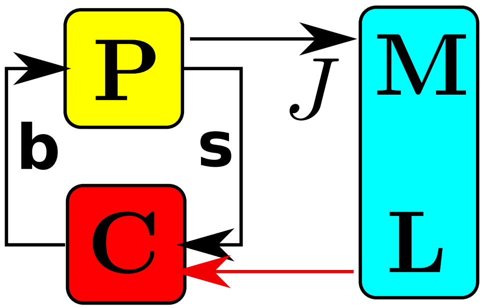
Build model:

$$\frac{da}{dt} = F(a,b)$$
$$s = G(a,b)$$

Derive control:

$$b = K(s)$$

Machine learning control



Define cost function:

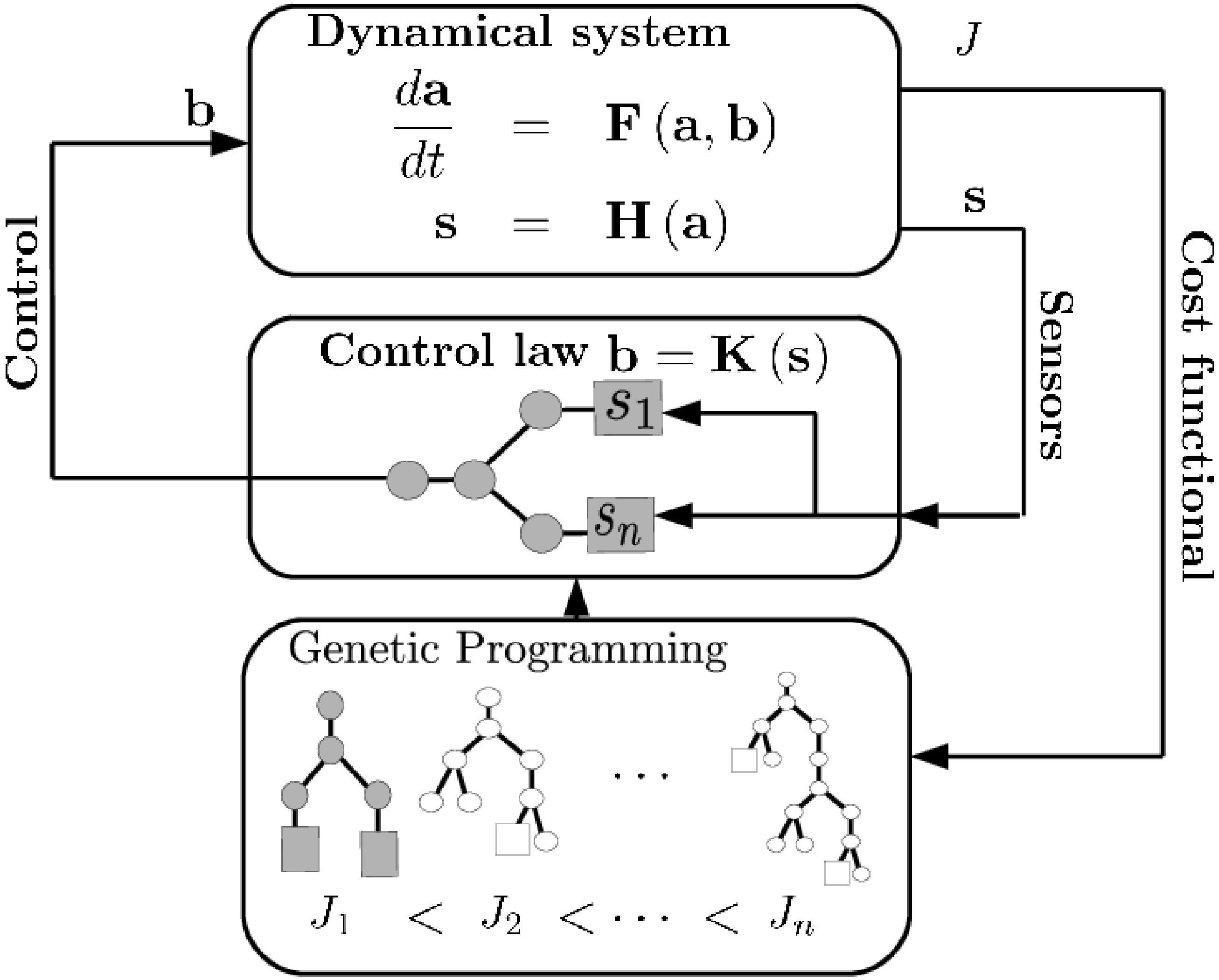
$$J = J_a + J_b = min$$

Solve regression problem:

$$\boxed{K_{opt}(s) = \arg\min J\,[K(s)]}$$

Machine learning control

Duriez, Brunton & Noack 2016 Springer, Wahde 2008



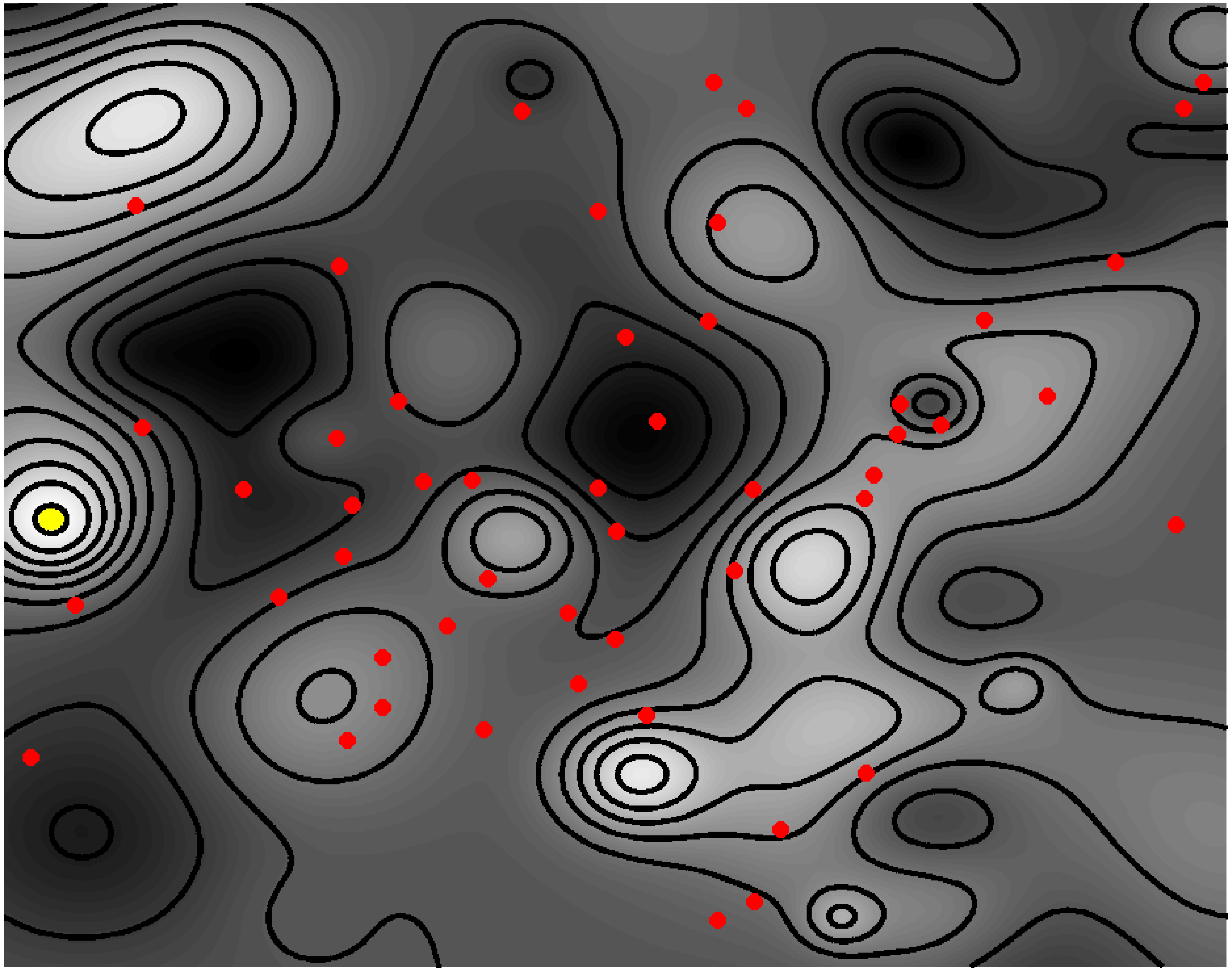
Regression problem: Find $b = K(s)$ so that $J = min$

Regression method = Genetic programming

Machine learning control III

[≡] *Duriez, Brunton & Noack 2016 Springer,* [≡] *Gautier et al. 2015 JFM*

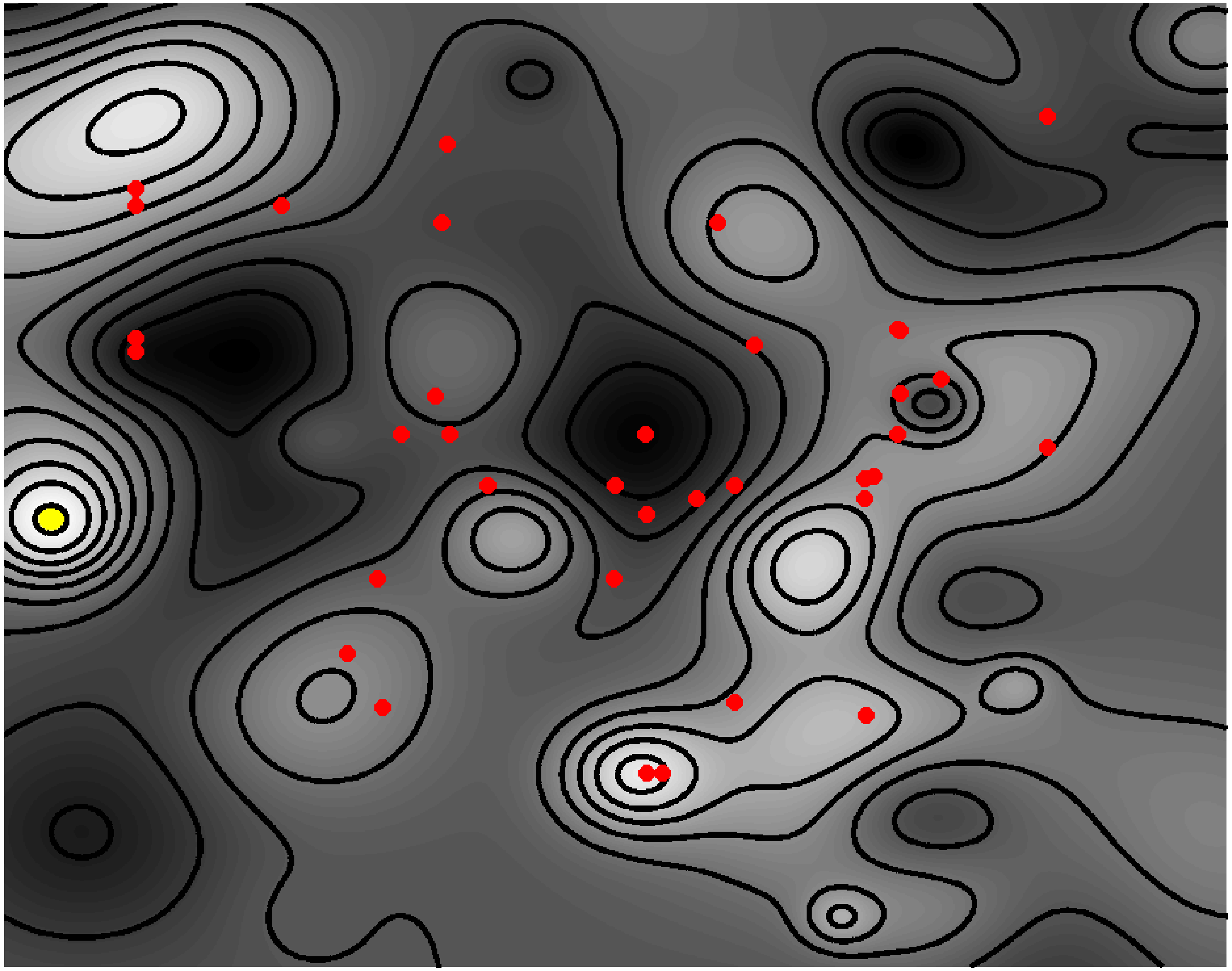
$n = 1$



Machine learning control III

≡ Duriez, Brunton & Noack 2016 Springer, ≡ Gautier et al. 2015 JFM

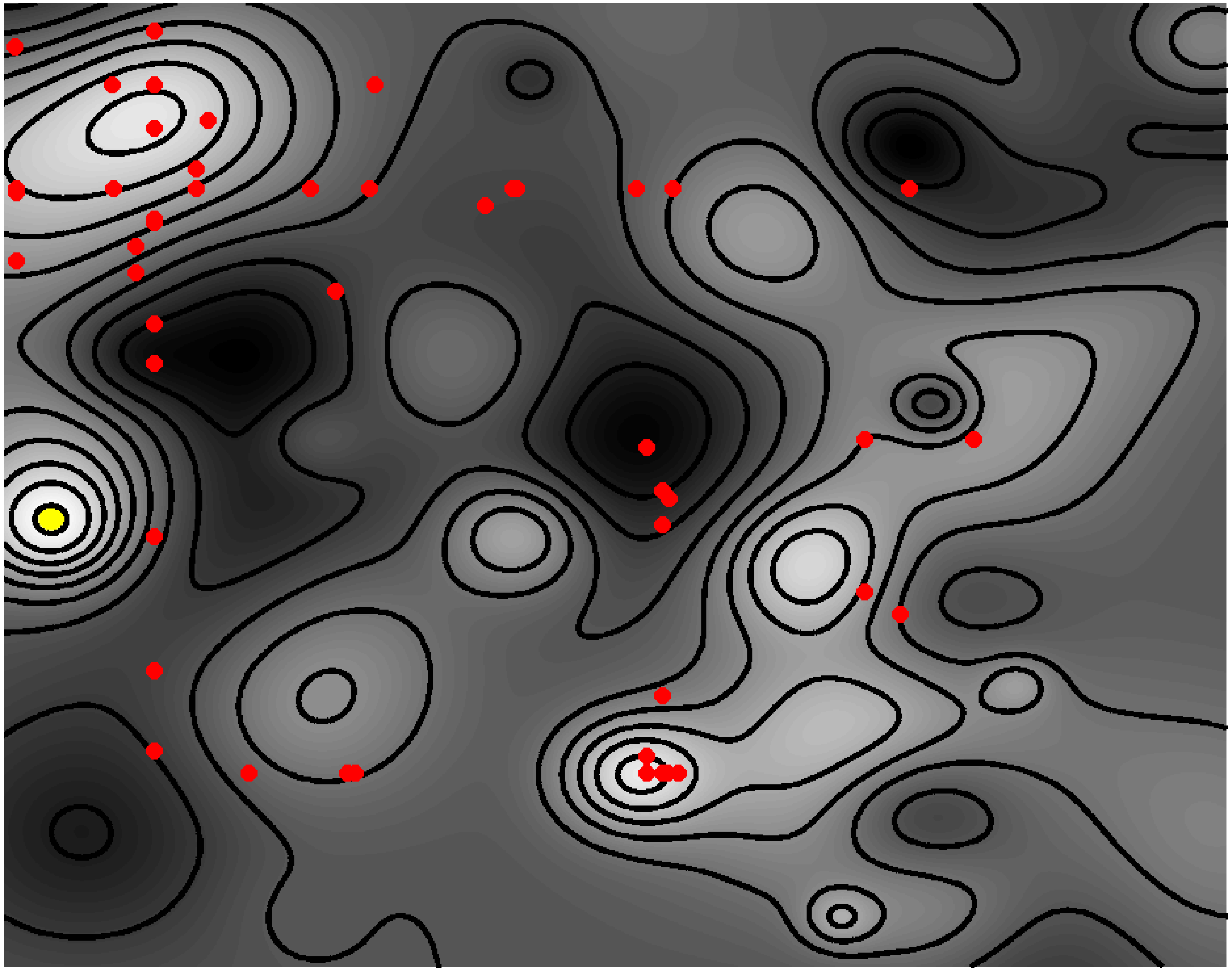
$n = 2$



Machine learning control III

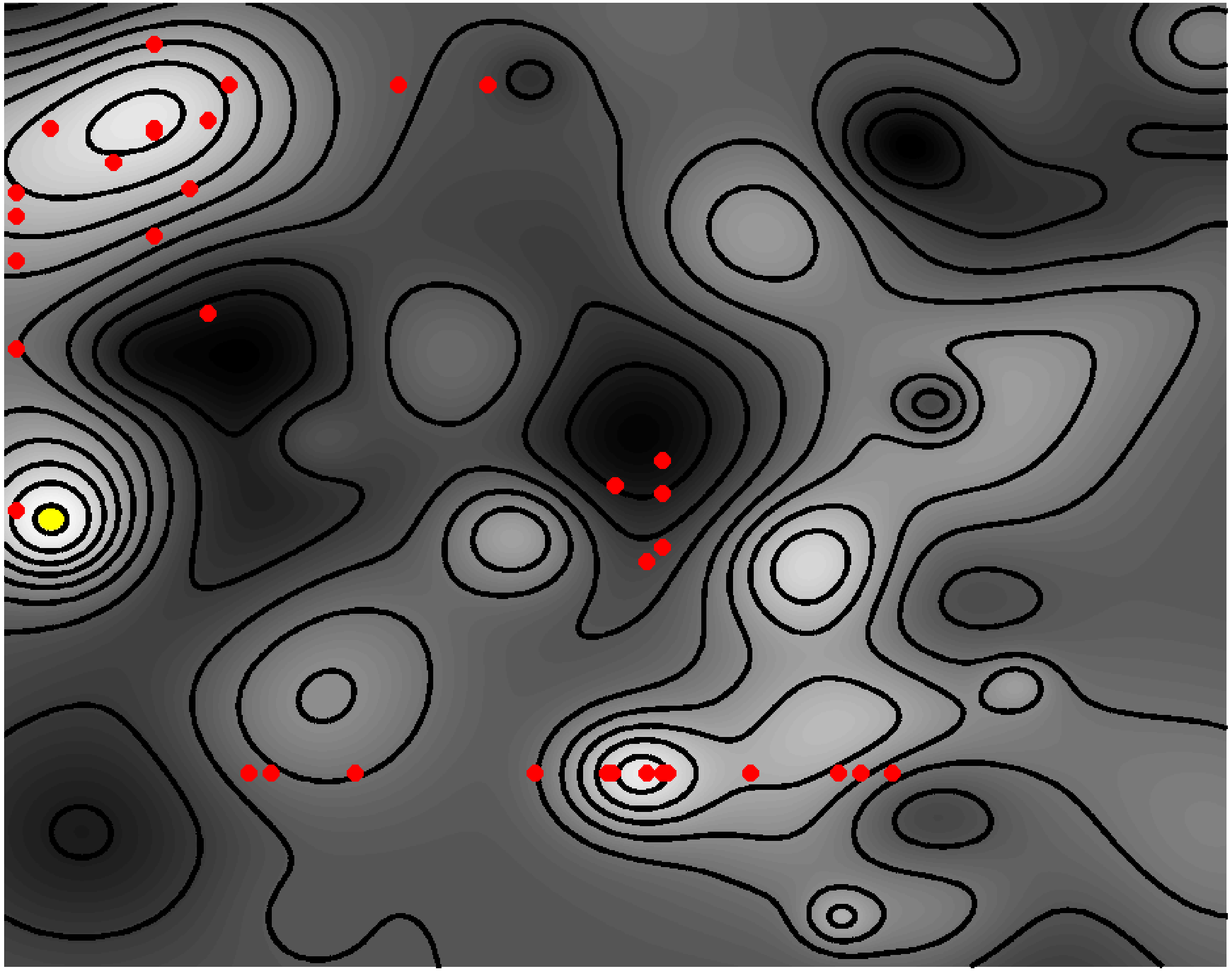
≡ Duriez, Brunton & Noack 2016 Springer, ≡ Gautier et al. 2015 JFM

$n = 3$



Machine learning control III

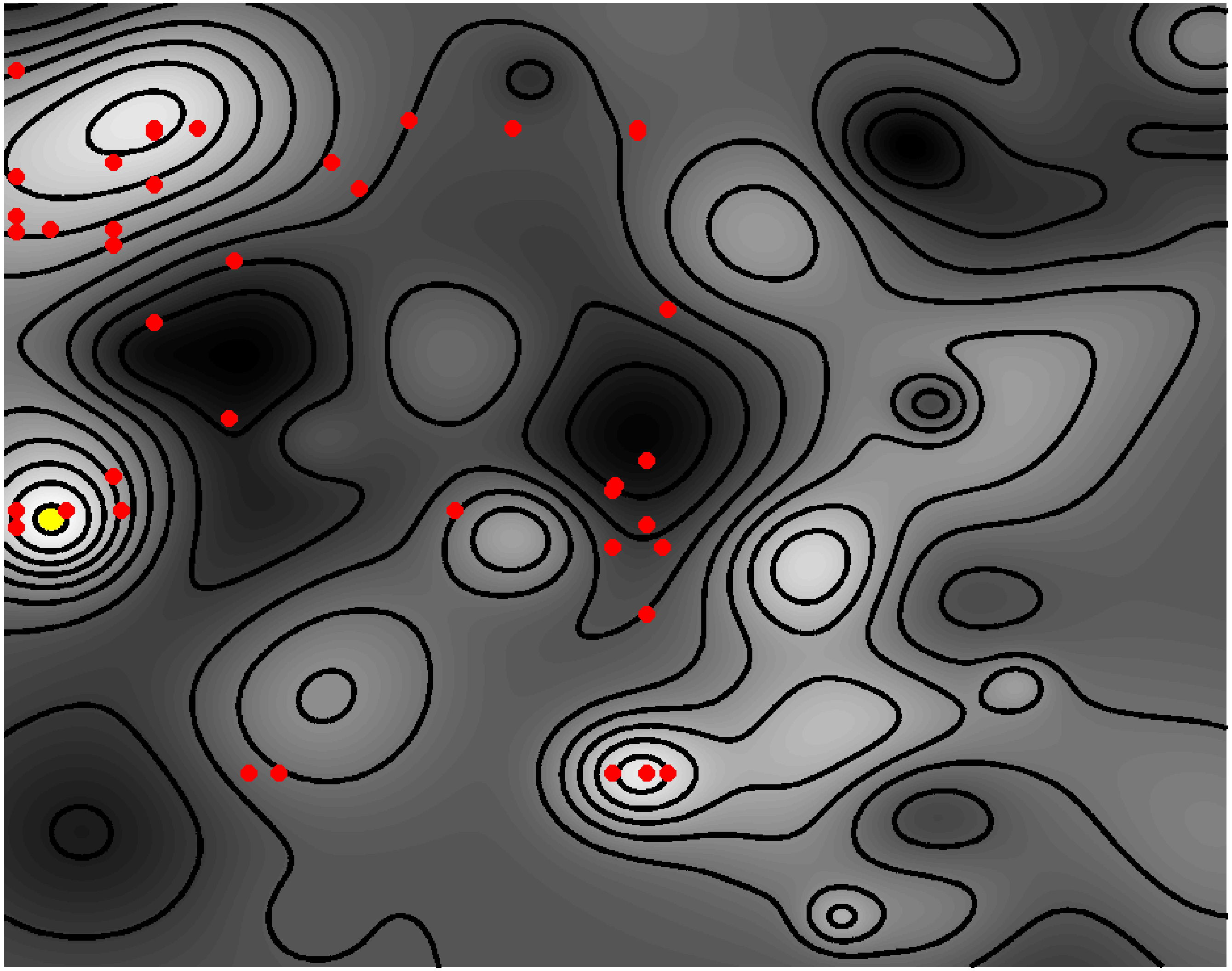
$n = 4$



Machine learning control III

≡ *Duriez, Brunton & Noack 2016 Springer,* ≡ *Gautier et al. 2015 JFM*

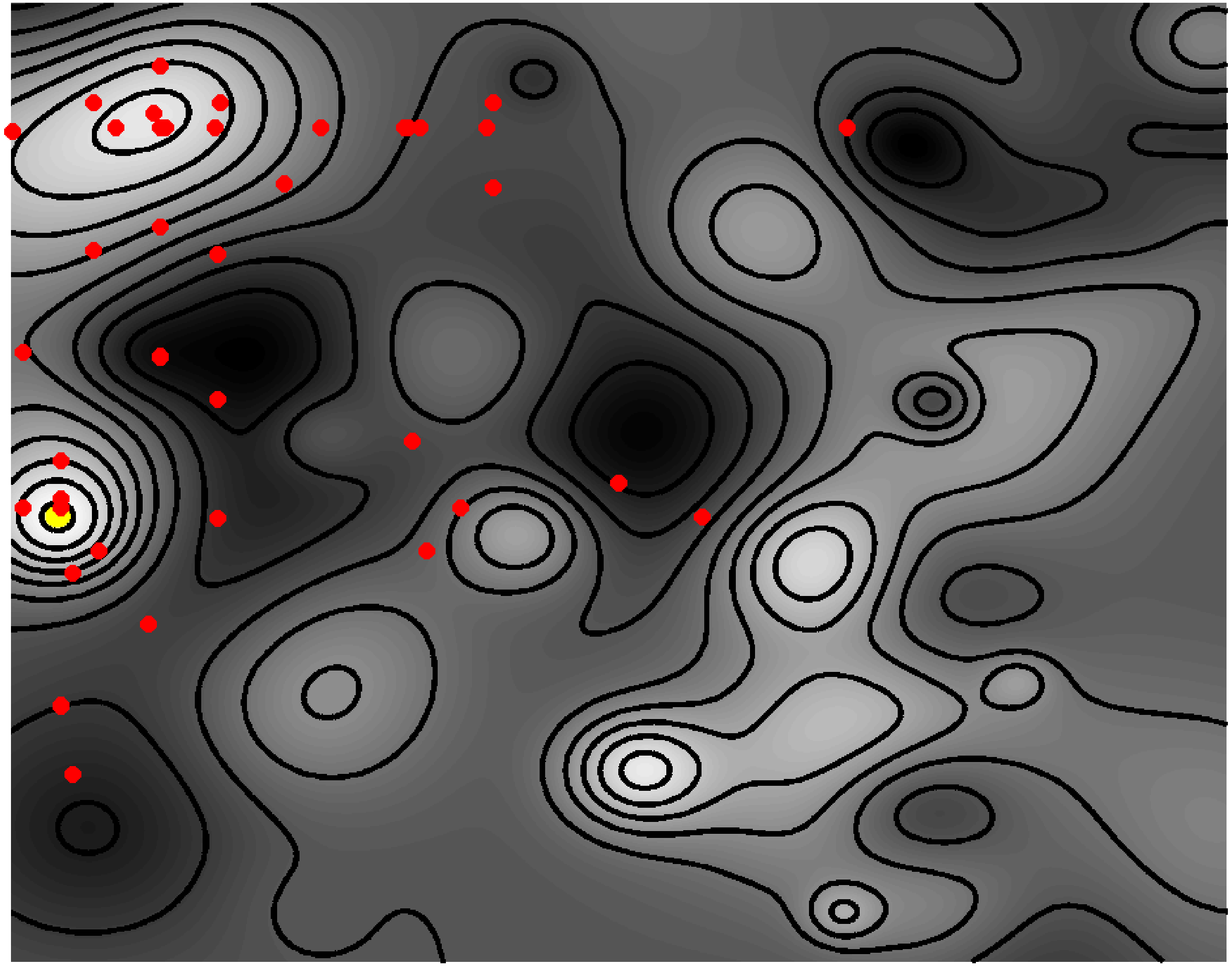
$n = 5$



Machine learning control III

≡ *Duriez, Brunton & Noack 2016 Springer,* ≡ *Gautier et al. 2015 JFM*

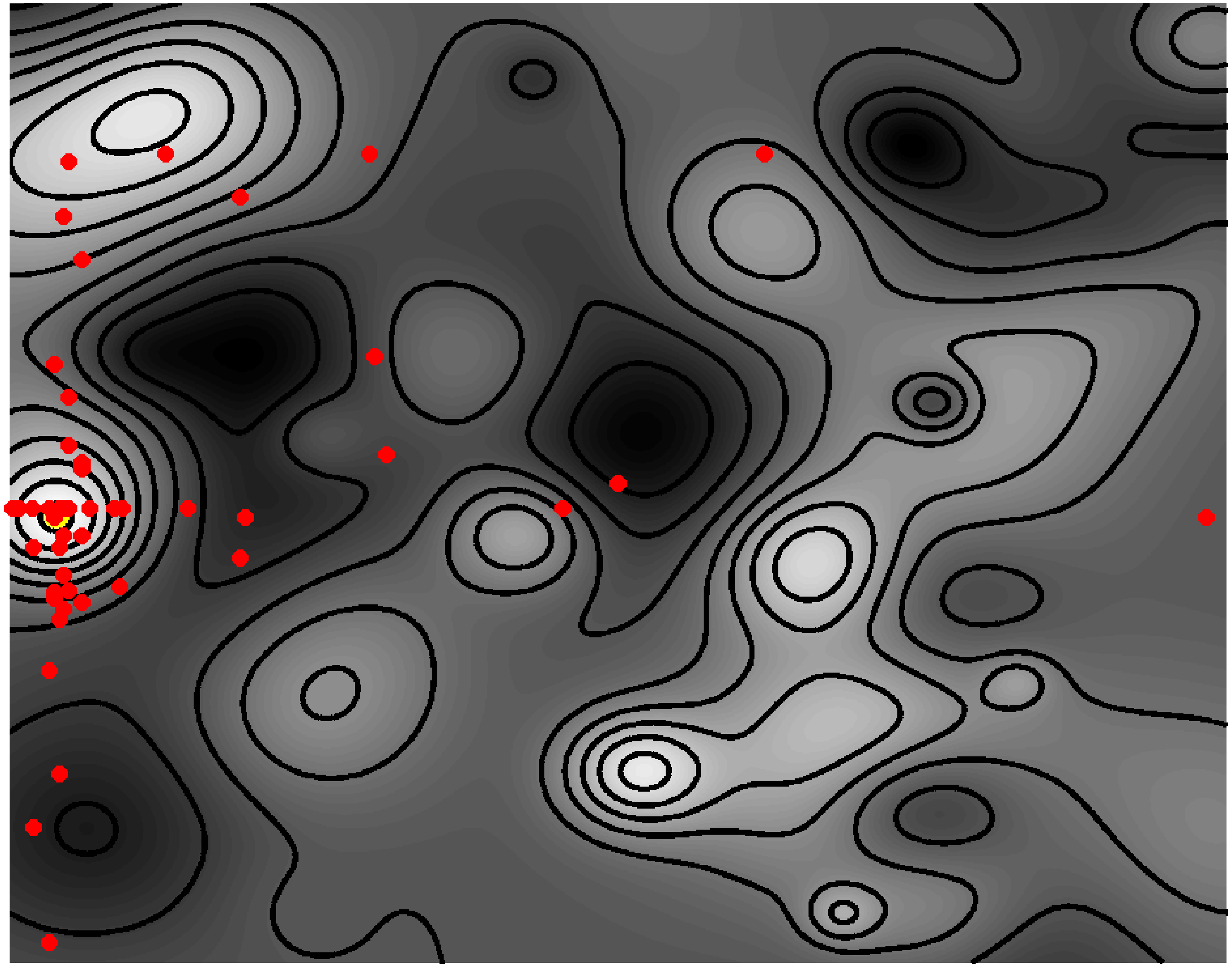
$n = 10$



Machine learning control III

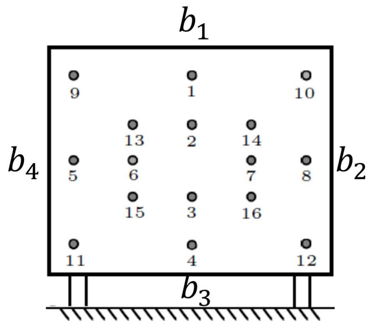
≡ Duriez, Brunton & Noack 2016 Springer, ≡ Gautier et al. 2015 JFM

$n = 20$



MLC-based drag reduction

Li+ 2017 EF & Barros+ 2016 JFM



Experiment: $Re = 3 \times 10^5$

MIMO control problem:

Ansatz $\boldsymbol{b} = \boldsymbol{K}(\boldsymbol{s})$

Drag reduction: 22%

Energy investment: 3%

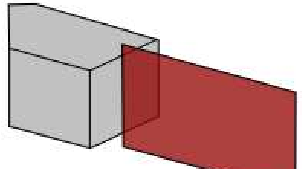
MLC application

Testing time < 1 hour

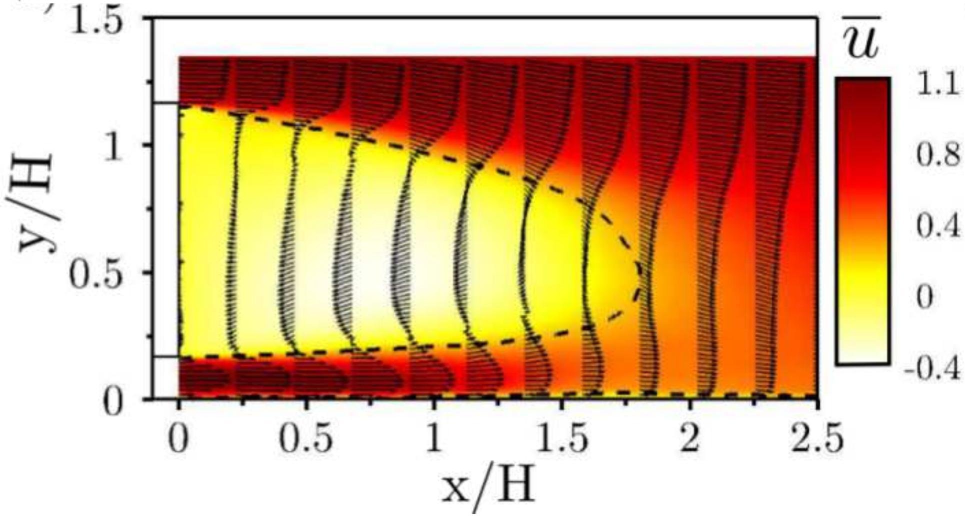
MLC law:

$b_1 = b_2 = b_3 = b_4 = b$

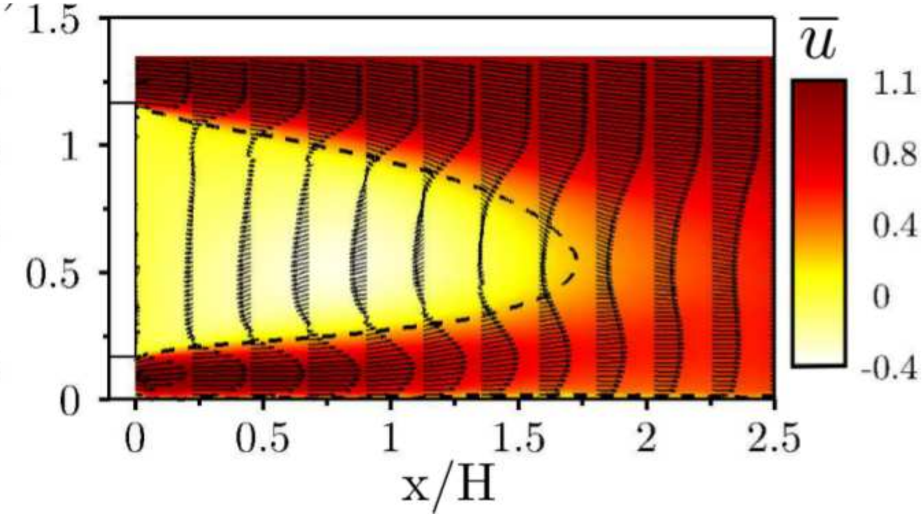
$b = H\left[\tanh\tanh(s'_4 - 0.1)\right]$



Unforced

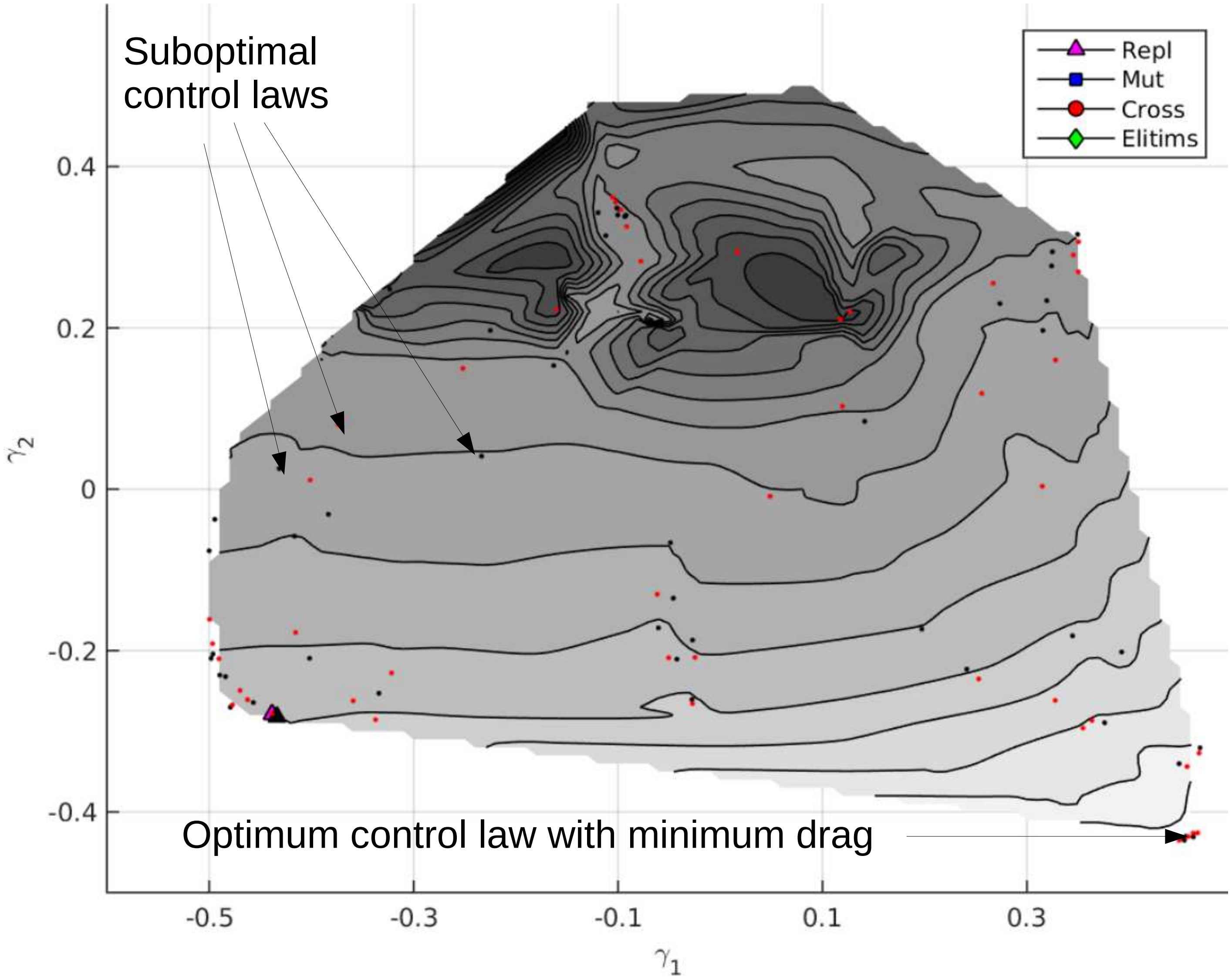


MLC controlled



Proximity plot for MLC of car model

≡ *2017 Kaiser+ FSSIC* ≡ *2016 Kaiser+ TCFD*



MLC with 5 generations with 50 control laws each.

More

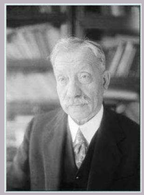
Core methods for ∼90% of your tasks

Noack+ 2011 Springer (↦ROM); Duriez+ 2016 Springer (↦MLC)

Classical approach

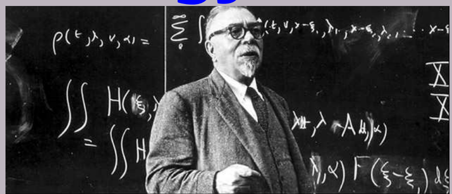
(1) Start with a model

POD modeling



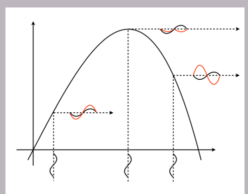
(2) Control for your model

Energy-based control



(3) Control without a model

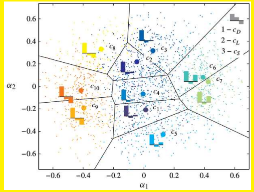
Extremung seeking



Machine learning

(4) Visualize your data

Proximity maps

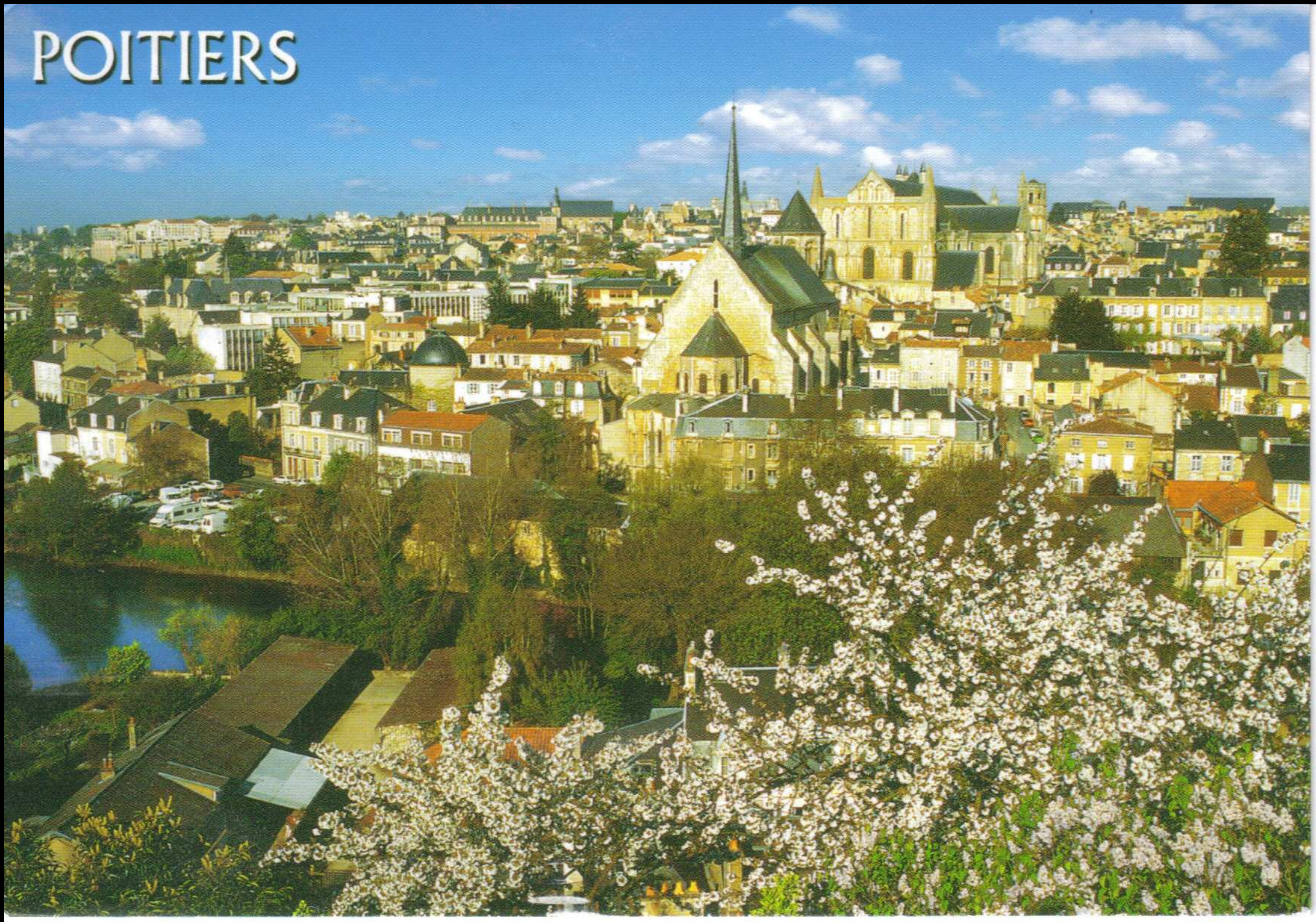


(5) Group your data in 'bins'

Clustering

(6) Learn control law

Genetic programming


POITIERS

Machine Learning — Definition

Machine learning is a subfield of computer science that evolved from the study of pattern recognition and computational learning theory in artificial intelligence. Machine learning explores the construction and study of **algorithms that can learn from and make predictions on data.** [Wikipedia]

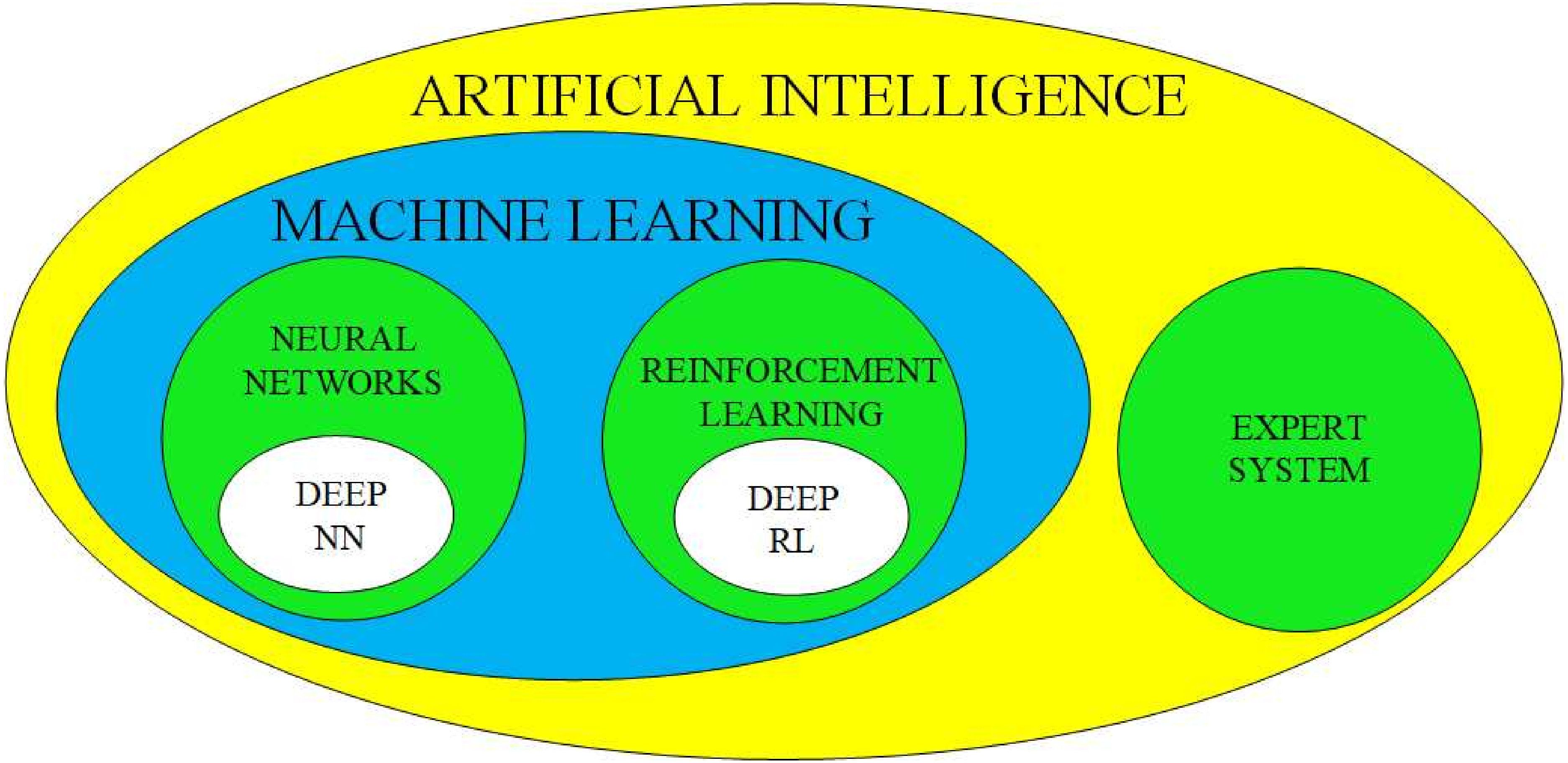
Tasks

(1) Supervised learning $\Rightarrow$ input-output map (with teacher)
(e.g. artifical neural network, genetic programming, ...);

(2) Unsupervised learning $\Rightarrow$ hidden structure in input
(e.g. k-means clustering);

(3) Reinforcement learning $\Rightarrow$ learning control by rewards

Machine Learning versus Artificial Intelligence



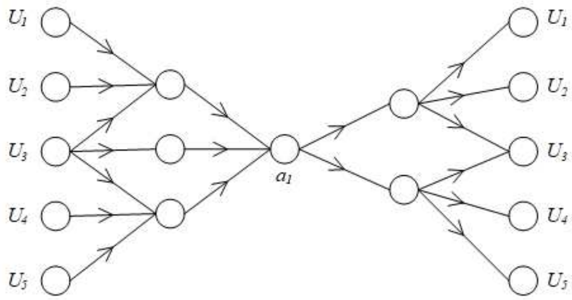
Reinforcement learning $\Rightarrow$ AlphaGo victory over world champion 2017
Neural Networks $\Rightarrow$ speach recognition and automated translation

Machine Learning Methods

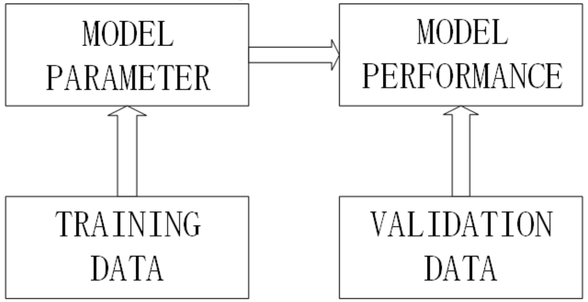
Regression problem: Find $a \mapsto b = f(a)$ which minimizes $J(f) = min$ based on data $a^m, b^m, m = 1, \ldots, M$ or based on testing $J(f)$.

Meta

Autoencoder

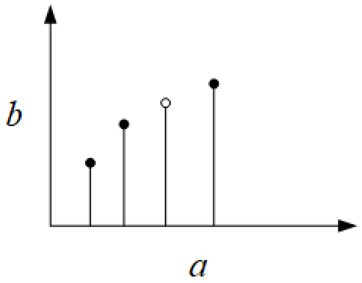


Learning from data

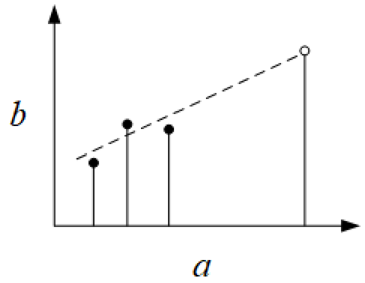


Function fitting

Interpolation

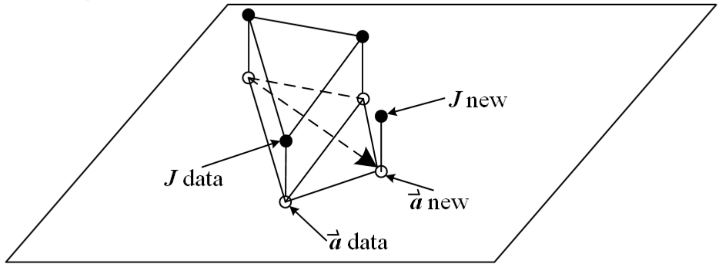


Extrapolation

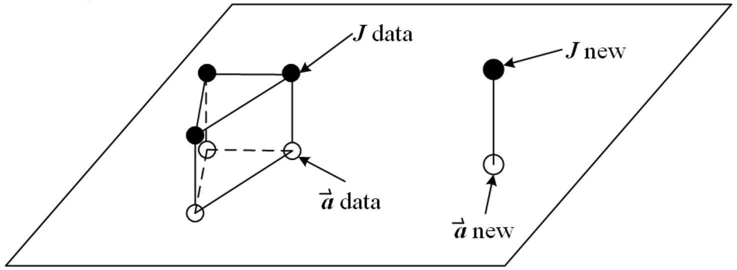


Variational problem

Exploitation



Exploration



AI = ML + expert systems / natural language recognition

Fluid dynamics — Regression problems

Regression problem: Find a function $\boldsymbol{a} \mapsto \boldsymbol{b} = \boldsymbol{f}(\boldsymbol{a})$ which minimizes $J(\boldsymbol{f}) = min$ based on data $\boldsymbol{a}^m, \boldsymbol{b}^m, m = 1, \ldots, M$ *(1st kind)* or based on testing $J(\boldsymbol{f})$ *(2nd kind)*.

Examples:

(1)	**Closure relation:**	$\boldsymbol{u}(\boldsymbol{x}, t) \mapsto \nu_S$
(2)	**Prediction:**	$\boldsymbol{u}(\boldsymbol{x}, t) \mapsto \boldsymbol{u}(\boldsymbol{x}, t + \Delta t)$
(3)	**State estimation:**	$\boldsymbol{s}(t) \mapsto \hat{\boldsymbol{u}}(\boldsymbol{x}, t)$
(4)	**Control law:**	$\boldsymbol{s}(t) \mapsto \boldsymbol{b} = \boldsymbol{K}(\boldsymbol{s})$ s.t. $J = min$
(5)	**Parametrics**	$\boldsymbol{p} \mapsto \widehat{J}(\boldsymbol{p})$ (response surface)
($\star$)	**Autoencoder**	$\boldsymbol{u} \mapsto \boldsymbol{a} \in R^N$ (Encoder)
	(Example POD)	$\boldsymbol{a} \mapsto \hat{\boldsymbol{u}} = \sum a_i \boldsymbol{u}_i$ (Decoder)

ML solvers for FD problems

Meta	• Low-dimensional flow representation • simplifies regression problems *Methods: POD, DMD, LLE, Clustering*
Regression problem of 1st kind Fitting	• **Estimation** $\boldsymbol{s}(t) \mapsto \hat{\boldsymbol{u}}(\boldsymbol{x}, t)$ • **Prediction** $\boldsymbol{u}(\boldsymbol{x}, t) \mapsto \boldsymbol{u}(\boldsymbol{x}, t + \Delta t)$ • **Response surface** $\boldsymbol{p} \mapsto \hat{J}(\boldsymbol{p})$ *Methods: K-NN, Cluster-based representations, Taylor expansion, SINDy, ANN, ...*
Regr. probl. of 2nd kind Variational p.	• **Closure relation** $\boldsymbol{u}(\boldsymbol{x}, t) \mapsto \nu_S$ • **Control law** $\boldsymbol{s}(t) \mapsto \boldsymbol{b}$ s.t. $J = min$ *Methods: Genetic programming, genetic algorithm (parameters)*

Preview course—Selected Keywords

Autoencoders	Regression 1	Regression 2	Principles ML
—Analysis	—Modeling	—Control	—Big data
• Proximity map • Clustering • POD, SPOD • DMD, RDMD • LLE	• Interpolation • Extrapolation • K NN • Krigging • Neural networks • SVM • Linear, Quad • Thikonov • Sparse ID	• Exploitation • Exploration • DVM • Simplex • Gen. alg. • Gen. progr. • Evol. alg. • Particle swarm	• No free lunch theorem • Induction example • In-sample error • Out-of-sample error • Data versus model complexity • Overfitting • Underfitting • Crossvalidation • Priors • Constrained optimization

Literature

(1) **S.J. Brunton & B.R. Noack 2015**
Closed-loop turbulence control
—progress and challenges,
Applied Mechanics Reviews

(2) **K. J. Angström & R. M. Murray 2010**
Feedback Systems: An Introduction
for Scientists and Engineers.
Princeton University Press. *[Control theory]*

(3) **M. Wahde 2008**
Biologically Inspired Optimization Methods:
An Introduction. WIT Press.
[Genetic algorithm, linear genetic programming, ...]

(4) **T. Duriez, S. L. Brunton & B. R. Noack 2016**
Machine Learning Control
—Taming Nonlinear Dynamics and Turbulence, Springer

(5) **B. R. Noack, M. Morzyński & G. Tadmor 2011**
Reduced-Order Modeling for Flow Control, Springer

Machine Learning for Fluid Mechanics
Features and Autoencoders

Bernd Noack

HIT, China & TU Berlin

TU Berlin, Germany, 2020-03-02

Preview course

A) Features $u \mapsto \gamma$ ($\dim(\gamma) = 2, 3$)
and reduced-order representations
$u \mapsto a \mapsto \hat{u} \approx u$ where $\dim(\mathrm{a}) \ll \dim(\mathrm{u}, \hat{\mathrm{u}})$

B) Principles of machine learning

C) Regression problem of first type
$\mathbf{(a^m, b^m), m = 1, \ldots, M \Rightarrow b = K(a)}$

D) Regression problem of second type + MLC
$\mathbf{K^\star} = \mathrm{argmin}_{\mathbf{K}} \mathbf{J(K)}, \quad \mathbf{b = K(a)}$

E) Reduced-order modeling
Model-based control

Overview

1. **Introduction**

   . . . . *From da Vinci to LLE ...*

2. **Proximity map**

   . . . . *Cartographing high-dimensional data ...*

3. **Locally Linear Embedding**

   . . . . . . *Autoencoding on manifolds / The future of ROR*

4. **Proper Orthogonal Decomposition**

   . . . . *Autoencoding on subspace with minimum residuum*

5. **Other modal expansions**

   . . . . *Alternatives to POD*

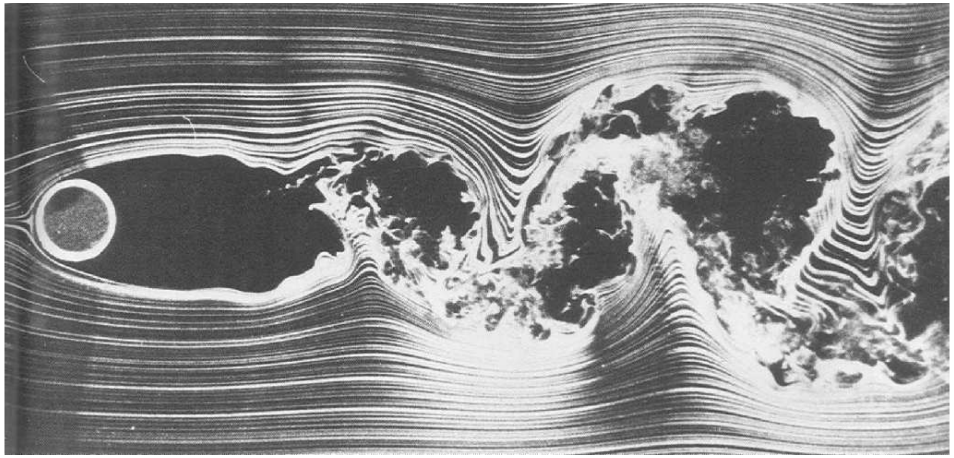
Overview

Path to low-dimensional models

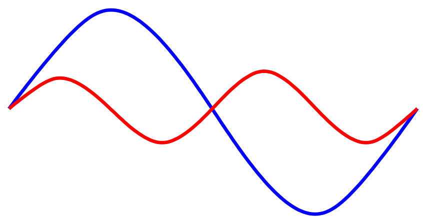
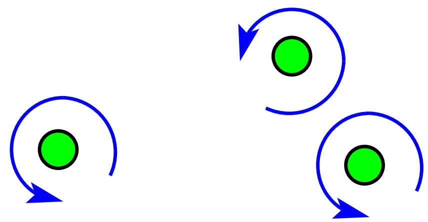
Low-dimensional coherent structures

Smoke visualization at $Re = 10000$

[van Dyke, Album of Fluid Motion]



Modeling approaches

	Eulerian view	**Lagrangian view**
coherent structures		
variable	velocity $\mathbf{u}$	vorticity ω
kinematics	$\mathbf{u} = \sum_i a_i(t)\mathbf{u}_i(\mathbf{x})$ Galerkin approximation	$\omega = \sum_i \Gamma_i \boldsymbol{\Omega}(\mathbf{x} - \mathbf{x}_i)$ vortex configuration
dynamics	$\frac{da_i}{dt} = f_i(a_1, \ldots)$ **Galerkin model**	$\frac{d\mathbf{x}_i}{dt} = \mathbf{u}(\{\Gamma_i, \mathbf{x}_i\})$ **vortex model**

Milestones of low-dimensional modeling

Leonardo da Vinci (1452–1519)

H.L.F. von Helmholtz (1821–1894)

Boris G. Galërkin (1871–1945)

Edward N. Lorenz (1917–2008)

- $\sim$ 1500 **da Vinci** $\in$ **visualization community**: First drawings of **coherent structures** (vortices) in the flow behind obstacles

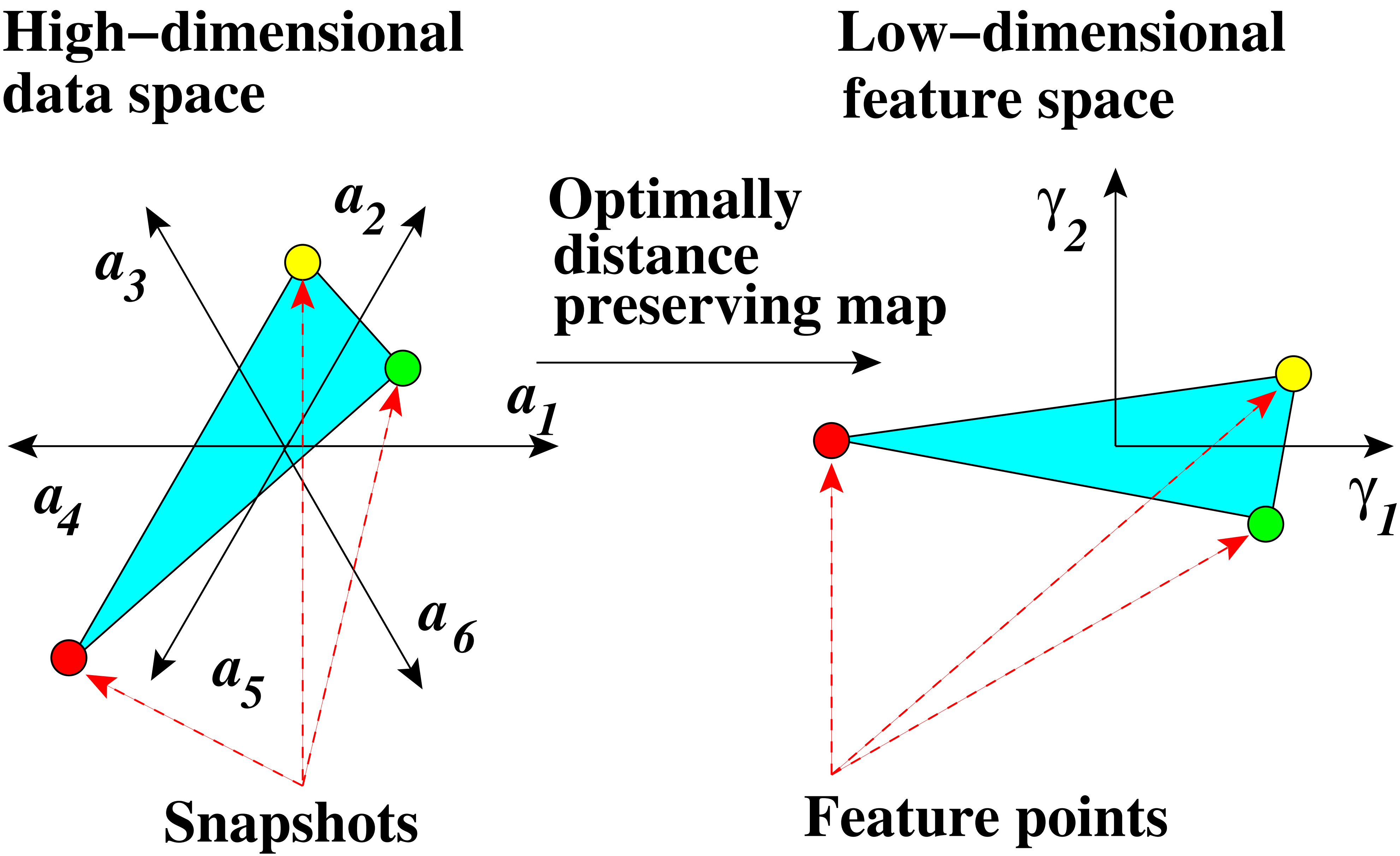
— *Euler (1755), Navier (1822), Stokes (1845)* —

- 1869 **von Helmholtz**: Theoretical foundation of the **vortex methods** with Helmholtz vortex laws
- 1915 **Galërkin**: Pioneering work on the **Galerkin methods**
- 1963 **Lorenz**: 3-dim. model for Rayleigh-Benard convection which lead to the "**Lorenz attractor**"

Overview

Proximity map — The idea

Kaiser+ 2014 JFM



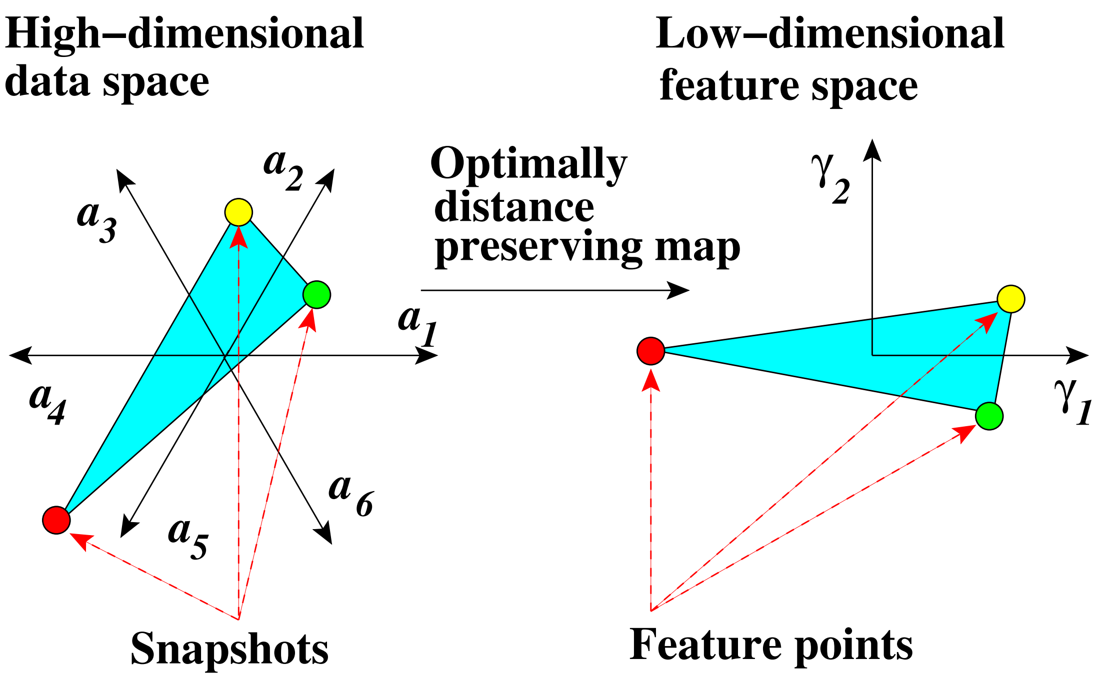
Proximity map — The optimization problem

Kaiser+ 2014 JFM

Optimization problem: Find optimally distance preserving mapping $\boldsymbol{\gamma} = \boldsymbol{T}\boldsymbol{a}$ from high-dimensional data space $\boldsymbol{a} \in R^N$ to low-dimensional feature space $\boldsymbol{\gamma} \in R^K$, $K \ll N$.

$$\begin{aligned}
\boldsymbol{a}^m &\in R^N, \\
& \quad m = 1, \ldots, M \\
\boldsymbol{\gamma}^m &= \boldsymbol{T}\boldsymbol{a}^m \in R^K, \\
D^{mn} &= \|\boldsymbol{a}^m - \boldsymbol{a}^n\| \\
\delta^{mn} &= \|\boldsymbol{\gamma}^m - \boldsymbol{\gamma}^n\| \\
\epsilon^{mn} &= |\delta^{mn} - D^{mn}| \\
E &= \sum_{m,n=1}^{M} (\epsilon^{mn})^2
\end{aligned}$$

Find T so that $E \overset{!}{=}$ min



Proximity map — The algorithm

Kaiser+ 2014 JFM

Executive summary (professors): google/baidu (Classical multidimensional scaling, wikipedia)

Executive summary (students): `cmdscale` in Matlab

Comments:

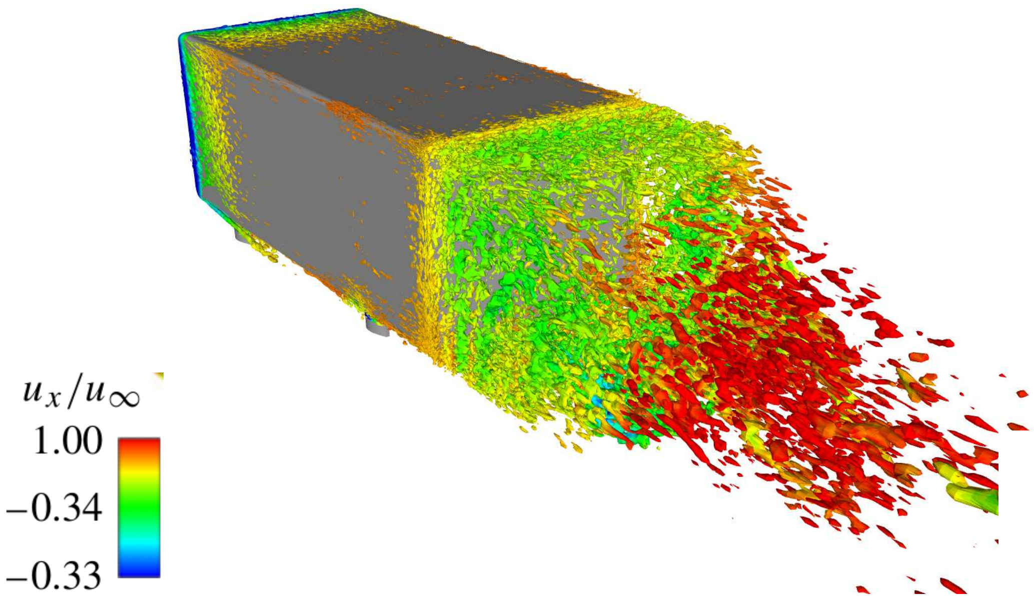
(1) Fixing the translation degree of freedom: Require $\sum_m \gamma^m = 0$—like in POD.

(2) Fixing the rotational degree of freedom: Maximize the variance in γ_1, then in γ_2, etc.—like in POD.

(3) Living with the mirror-symmetry: There is no cure—like with POD.

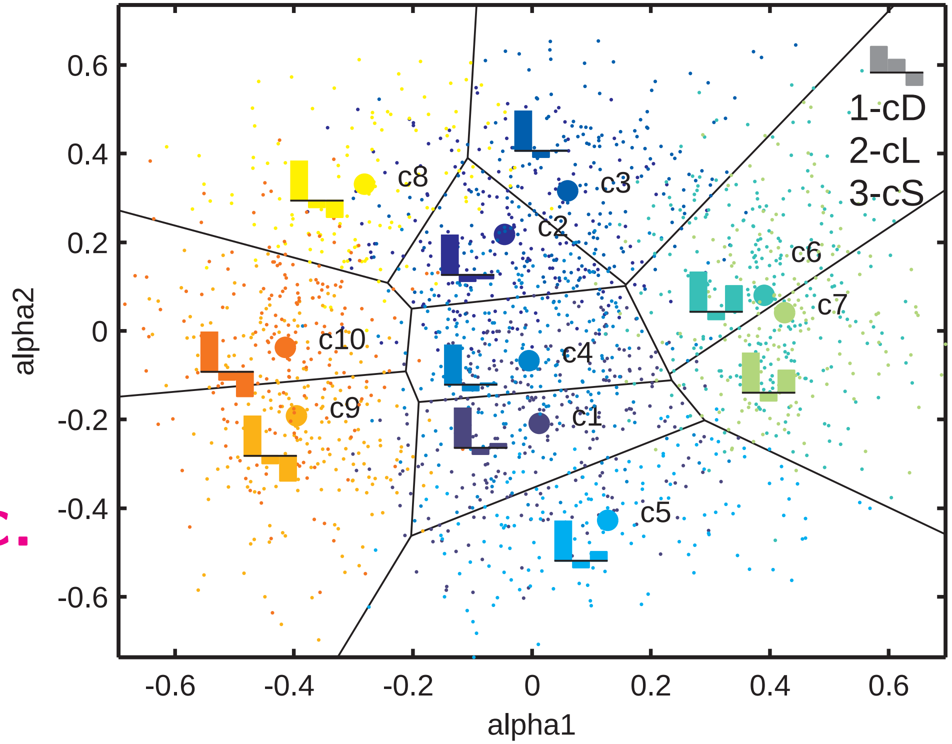
LES of Ahmed body — Proximity map

Kaiser+ 2014 JFM



Data space: $\boldsymbol{u}^m(\boldsymbol{x})$

Feature space: R^2

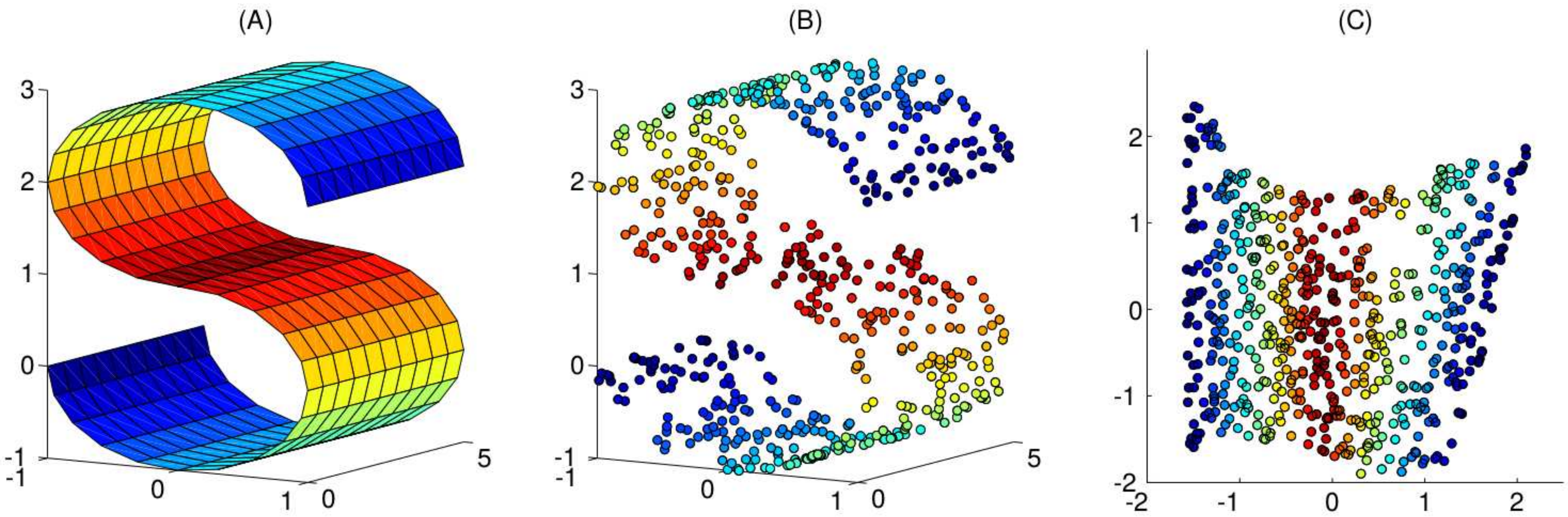


Proximity maps can be used for data, functions, etc.

Overview

Locally linear embedding — Idea

Roweis & Saul 2000 Science; Saul & Roweis 2000 Report



"Figure 1: The problem of nonlinear dimensionality reduction, as illustrated for three dimensional data (B) sampled from a two dimensional manifold (A). An unsupervised learning algorithm must discover the global internal coordinates of the manifold without signals that explicitly indicate how the data should be embed- ded in two dimensions. The shading in (C) illustrates the neighborhood-preserving mapping discovered by LLE."

Locally linear embedding — Algorithm

Roweis & Saul 2000 Science; Saul & Roweis 2000 Report

Algorithm: (1) Compute the K neighbors of each data point $\boldsymbol{x}_i \in R^D$, $i = 1, \ldots, M$.

(2) Compute the weights $\boldsymbol{W}_{ij}$ that best reconstruct each data point from its neighbors, minimizing the cost (A) by constrained linear fits.

(3) Compute the local coordinates $\boldsymbol{y}_i \in R^d$ (associated with $\boldsymbol{x}_i \in R^D$) best reconstructed by the weights $\boldsymbol{W}_{ij}$, minimizing the quadratic form (B) by its bottom nonzero eigenvectors.

Cost = Reconstruction error

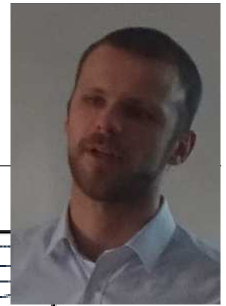
(A) $E(\boldsymbol{W}) = \sum_i \|\boldsymbol{x}_i - \sum_j \boldsymbol{W}_{ij}\boldsymbol{x}_j\|^2, \quad \sum_j \boldsymbol{W}_{ij} = 1$

Quadratic functional for local coordinates

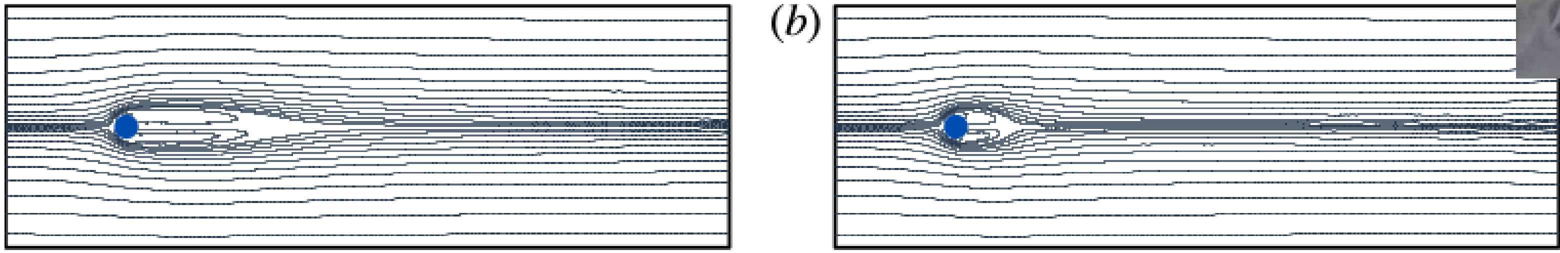
(B) $\Phi(\boldsymbol{Y}) = \sum_i \|\boldsymbol{y}_i - \sum_j \boldsymbol{W}_{ij}\boldsymbol{y}_j\|^2$

LLE—Data of wake transient

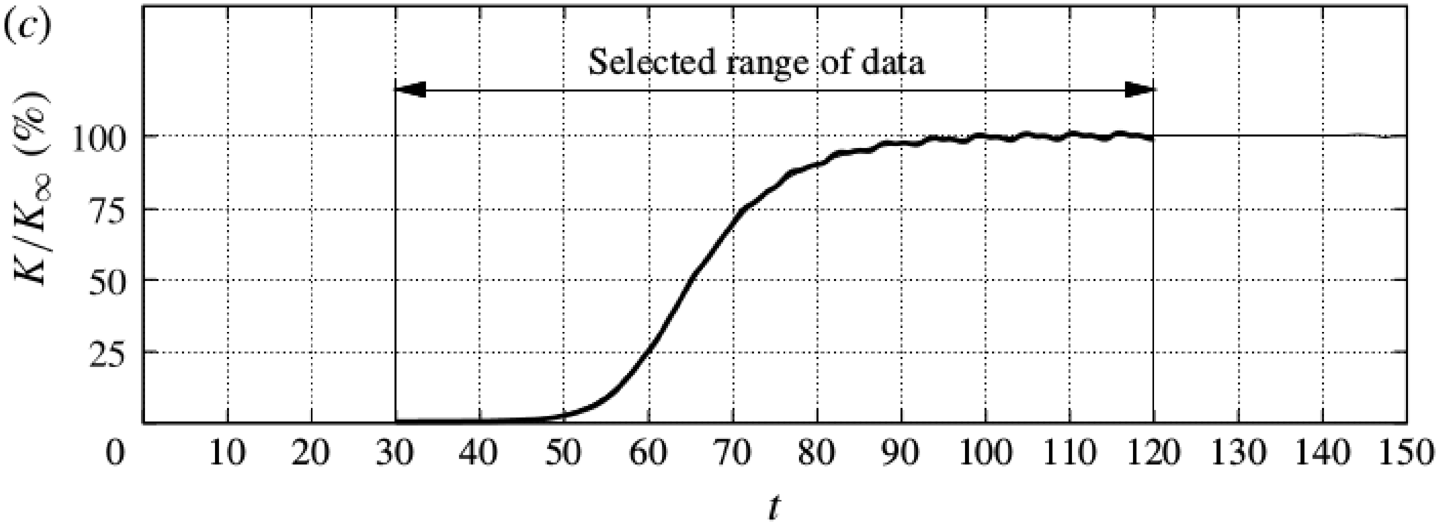
Ehlert, Noack, Morzyński & Nayeri (2020) JFM (subm)



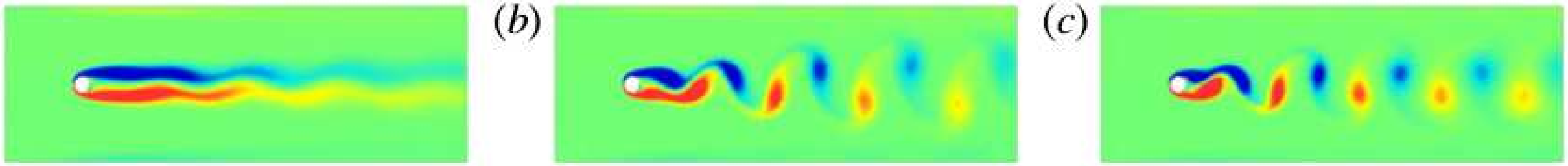
(a) (b)



(c)



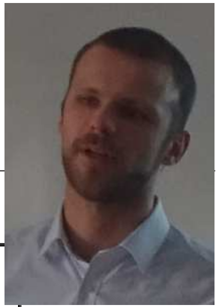
(a) (b) (c)

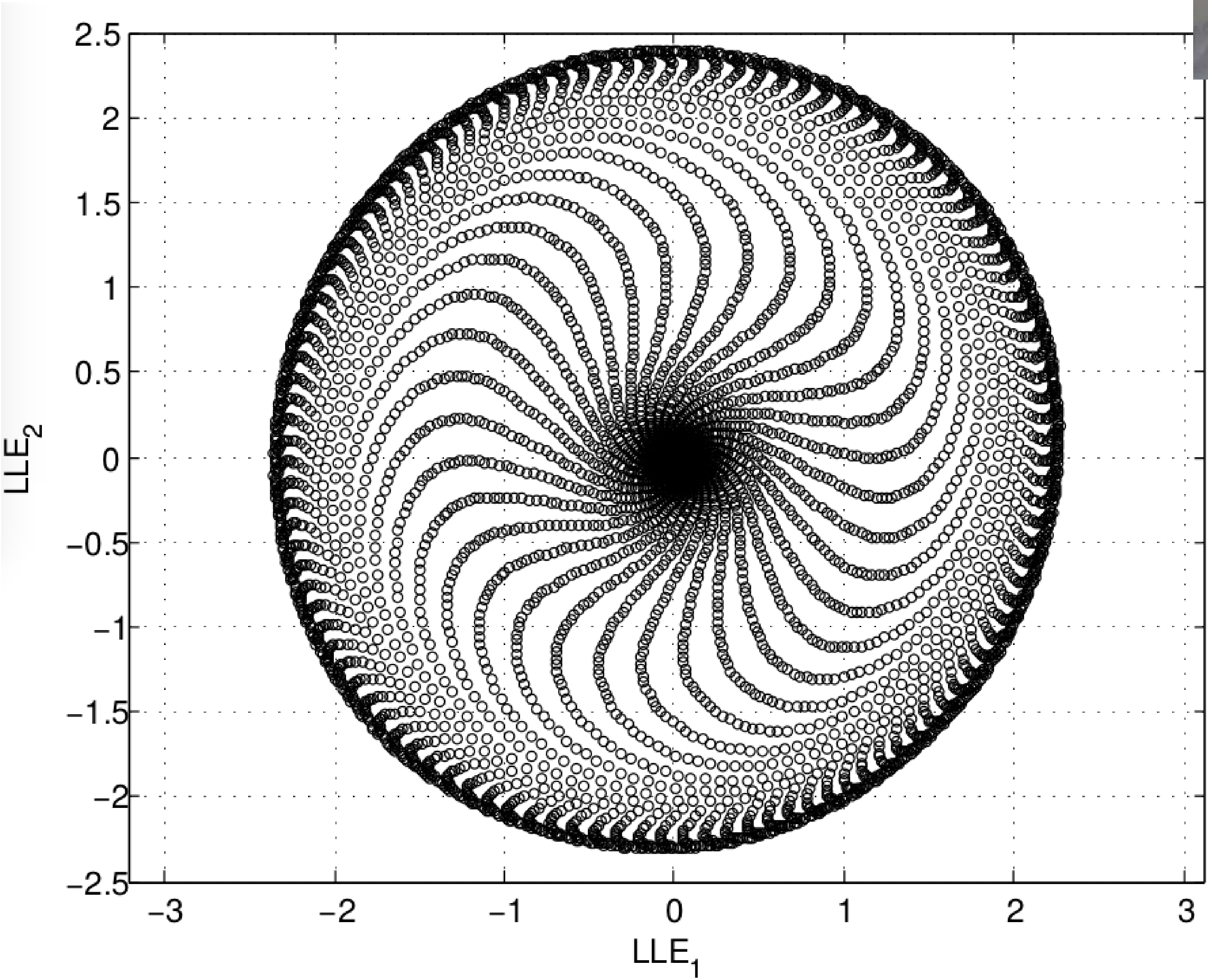


Data = 16 transients with 1000 snapshots each

LLE—Data in feature plane

Ehlert, Noack, Morzyński & Nayeri (2020) JFM (subm)

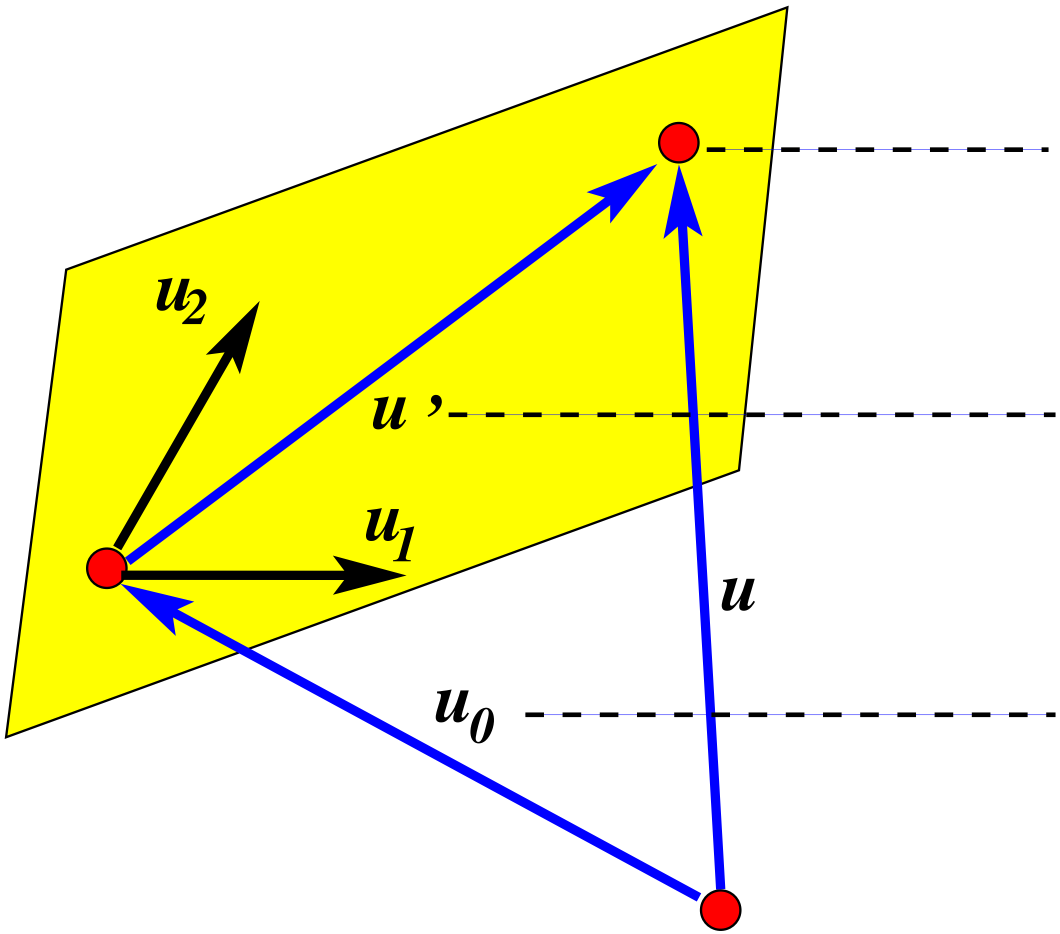




Overview

Low-dimensional Galerkin approximation

$$\mathbf{u}(\mathbf{x},t) = \mathbf{u}_0(\mathbf{x}) + \underbrace{\sum_{i=1}^{N} a_i(t)\,\mathbf{u}_i(\mathbf{x})}_{\mathbf{u}'(\mathbf{x},t)}$$



Galerkin approximation

fulfills BC

fluctuation $\mathbf{u}'$

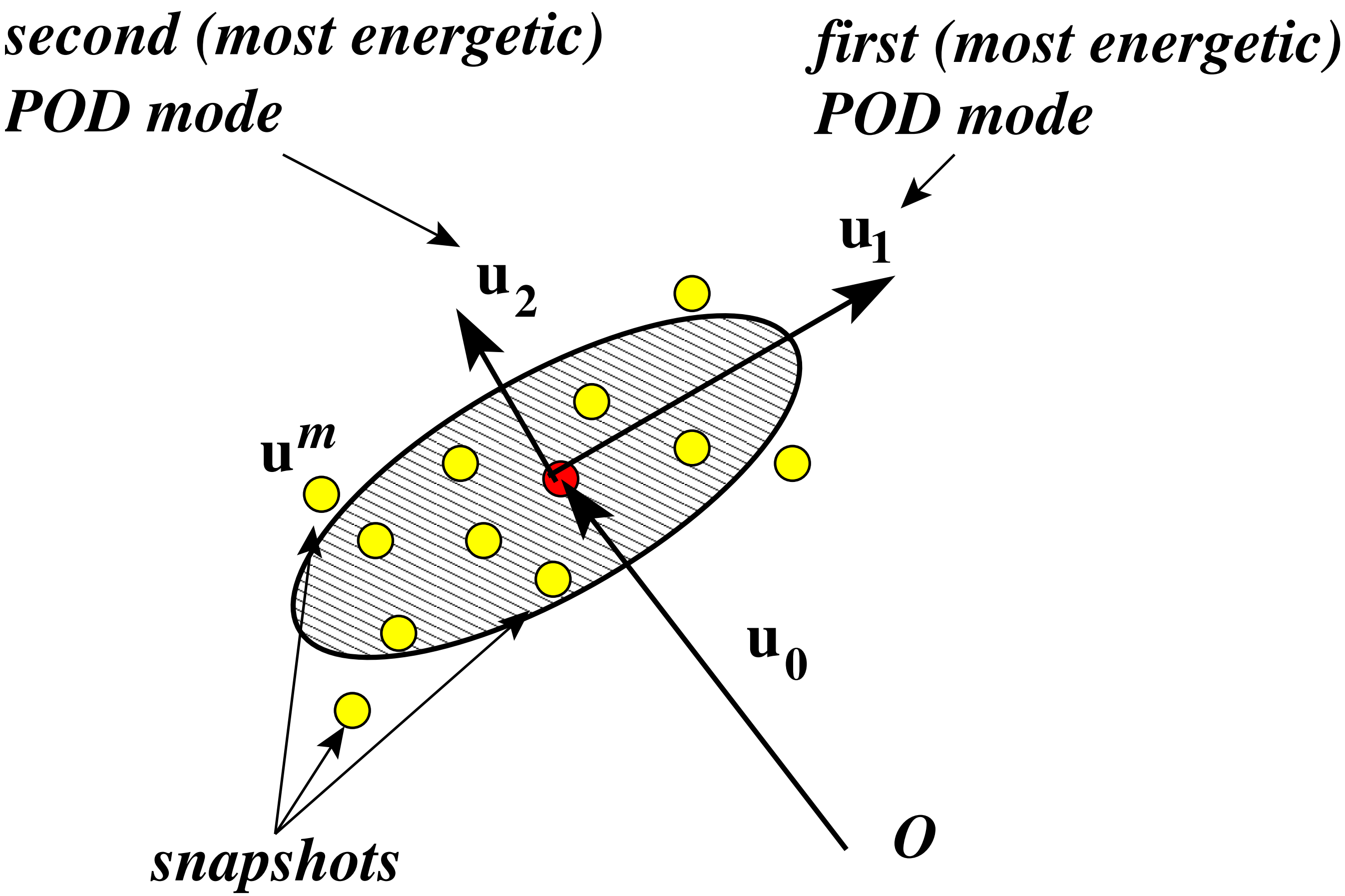
fulfills homogenized BC

basic mode

fulfills inhom. BC

'Least-dim.' (POD) Galerkin approximation

$$\mathbf{u}(\mathbf{x},t) = \mathbf{u}_0(\mathbf{x}) + \sum_{i=1}^{N} a_i(t)\,\mathbf{u}_i(\mathbf{x})$$



POD — mean flow

$$\mathbf{u}_0 = \frac{1}{M} \sum_{m=1}^{M} \mathbf{u}^m$$

Note that POD is a refined 2-points statistics up to second moments.

$$\langle \mathbf{u}'(\mathbf{x}, t) \otimes \mathbf{u}'(\mathbf{y}, t) \rangle = \sum_{i=1}^{\infty} \lambda_i \, \mathbf{u}_i(\mathbf{x}) \otimes \mathbf{u}_i(\mathbf{y})$$

Hence, a mimimum requirement to the snapshot ensemble is that the single points statistics of

- the first moments (mean values, $\overline{u(\boldsymbol{x}, t)}$, ...) and
- the second centered moments (variances, $\overline{[u'(\boldsymbol{x}, t)]^2}$, ...)

are accurate.

POD — correlation matrix

$$C^{mn} = \frac{1}{M} (\mathbf{u}^m - \mathbf{u}_0, \mathbf{u}^n - \mathbf{u}_0)_\Omega$$

Note that the mean value is subtracted. Otherwise:

- The first POD mode is approximately the mean flow.
- The fluctuation has to be orthogonal to the mean flow.
- Convergence of the POD with $N \to \infty$ is not guaranteed.
- λ_i cannot be interpretated as variances.
- The POD Galerkin approximation $\mathbf{u} = \sum a_i \mathbf{u}_i$ does not fulfill the boundary conditions for arbitrary a_i.
- The POD Galerkin model is non physical, allows for varying oncoming velocities, etc.
- The beauty of POD modelling is lost.

POD — Eigenproblem

Fredholm equation (discretized in time domain):

$$\mathbf{C}\mathbf{e}_i = \lambda_i\, \mathbf{e}_i.$$

Here, $\mathbf{C} := (C^{mn}) \in \mathcal{R}^{M\times M}$, $\mathbf{e}_i := (e_1^i, e_2^i, \ldots, e_M^i)$.

- $\mathbf{C}$ is symmetric $\Rightarrow \lambda_i \in \mathcal{R}$ and

$$\mathbf{e}_i \cdot \mathbf{e}_j = \delta_{ij}$$

- $\mathbf{C}$ is positiv semi-definite $\Rightarrow \forall i\colon \lambda_i \geq 0$

POD — eigenmodes

$$\mathbf{u}_i := \frac{1}{\sqrt{M\,\lambda_i}} \sum_{m=1}^{M} e_m^i \, (\mathbf{u}^m - \mathbf{u}_0)$$

Validation:

- Check $\left(\mathbf{u}_i, \mathbf{u}_j\right)_\Omega = \delta_{ij}$
- Check $\mathcal{K} = \frac{1}{2}\left\langle \|\mathbf{u}'\|_\Omega^2 \right\rangle = \frac{1}{2}\,\mathrm{trace}\, C = \frac{1}{2}\sum_{m=1}^{M} \lambda_m$

POD — mode amplitudes

$$a_i(t_m) = a_i^m := \sqrt{\lambda_i\, M}\, e_m^i$$

Validation:

- Check $a_i^m = (\mathbf{u}^m - \mathbf{u}_0, \mathbf{u}_i)_\Omega$
- Check $\langle a_i \rangle = \frac{1}{M} \sum_{m=1}^{M} a_i^m = 0$
- Check $\langle a_i\, a_j \rangle = \frac{1}{M} \sum_{m=1}^{M} a_i^m a_j^m = \lambda_i\, \delta_{ij}$

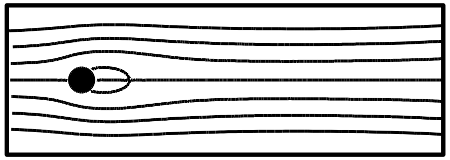
POD expansion of cylinder wake

Noack, Afanasiev, Morzyński, Thiele & Tadmor (2003) JFM

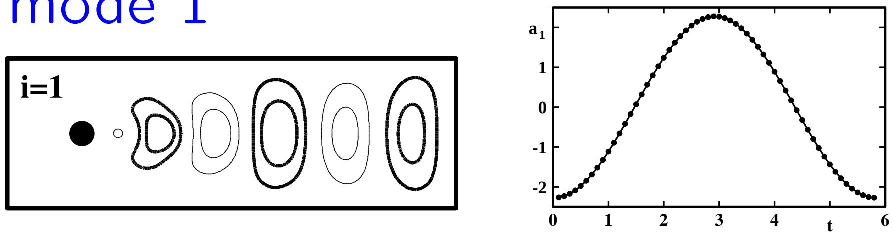
POD expansion

$$\boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{u}_0(\boldsymbol{x}) + \sum_{i=1}^{8} a_i(t)\, \boldsymbol{u}_i(\boldsymbol{x})$$

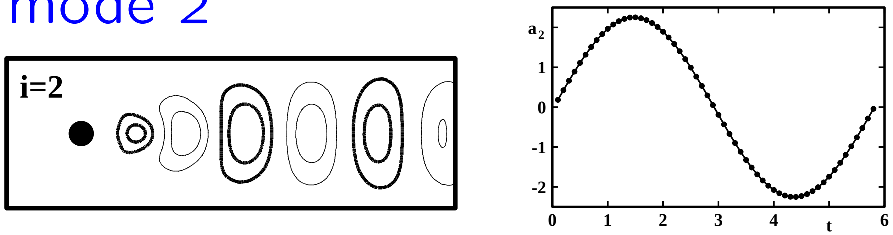
mode 0



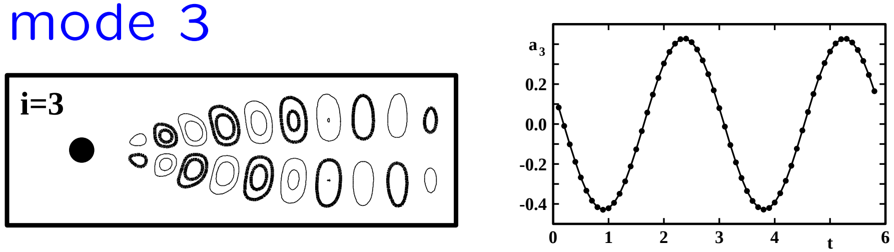
mode 1



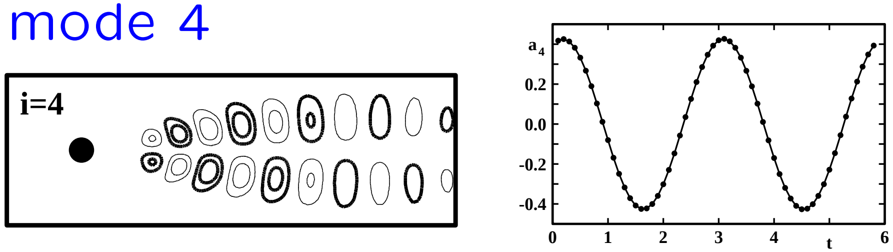
mode 2



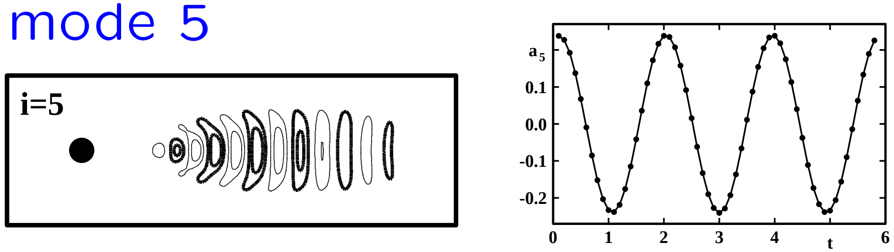
mode 3



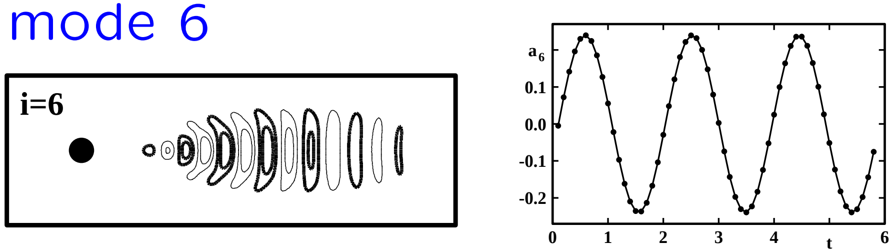
mode 4



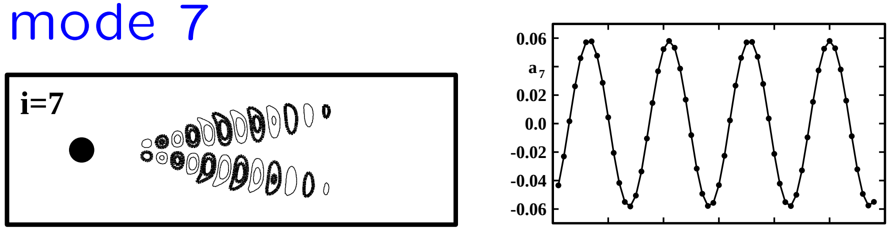
mode 5



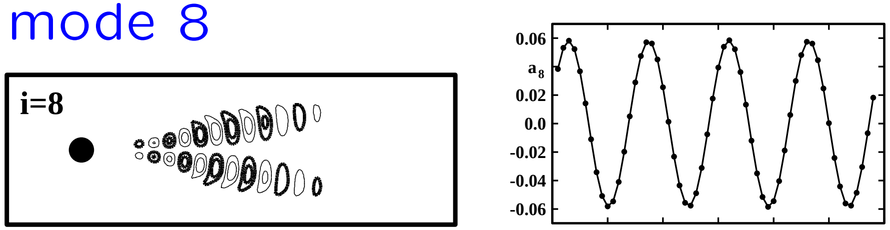
mode 6



mode 7



mode 8



POD of Kelvin-Helmholtz vortices

— ≡ *Noack, Papas & Monkewitz (2005) JFM* —

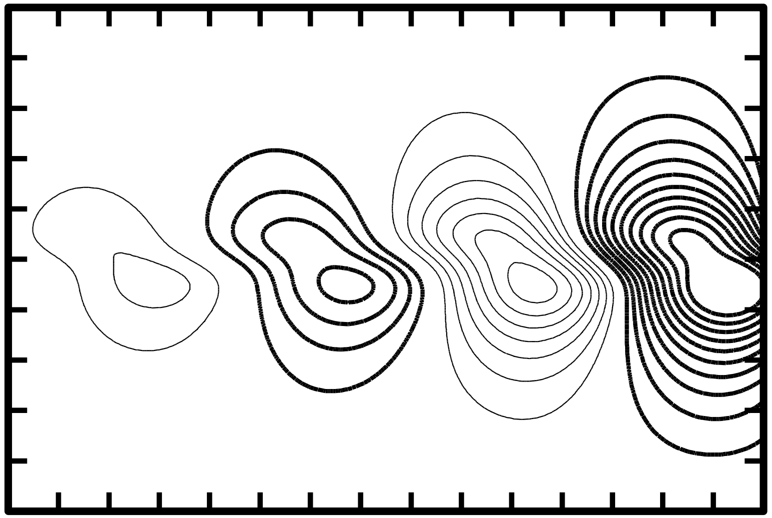
DNS

$Re_c = 100$

$u = \frac{2}{3} + \frac{1}{3} \tanh y$

- velocity ratio 3:1
- Dirichlet inflow condition

  $(+0.01 \times$ eigenmode$)$
- conv. outflow condition



POD

$\mathbf{u} = \sum_{i=0}^{4} a_i \mathbf{u}_i$

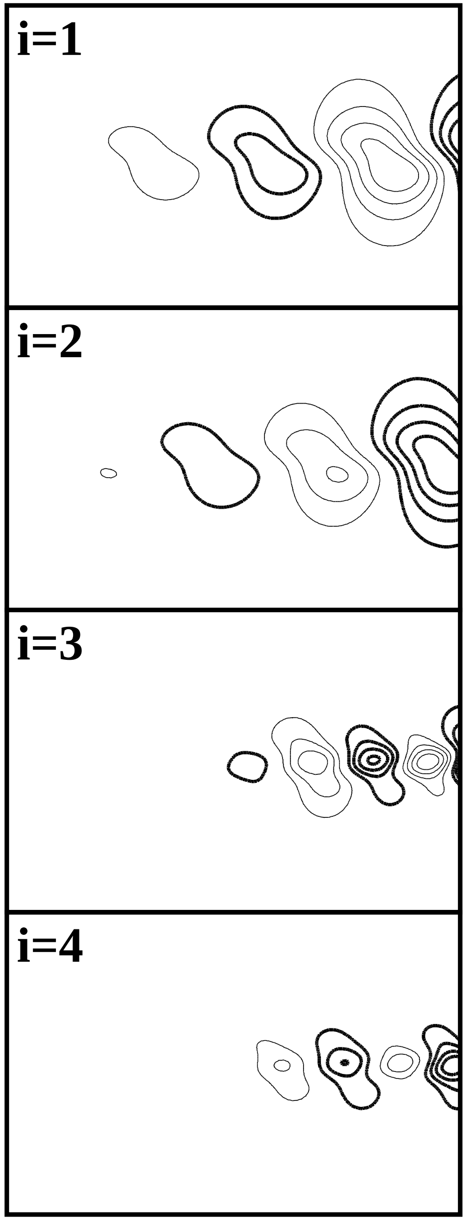
- mode 1

  $\sim \sin x$
- mode 2

  $\sim \cos x$
- mode 3

  $\sim \sin 2x$
- mode 4

  $\sim \cos 2x$



LES of turbulent mixing layer

— ≡ *Comte, Sivestrini & Bégou (1998) EJMB*—

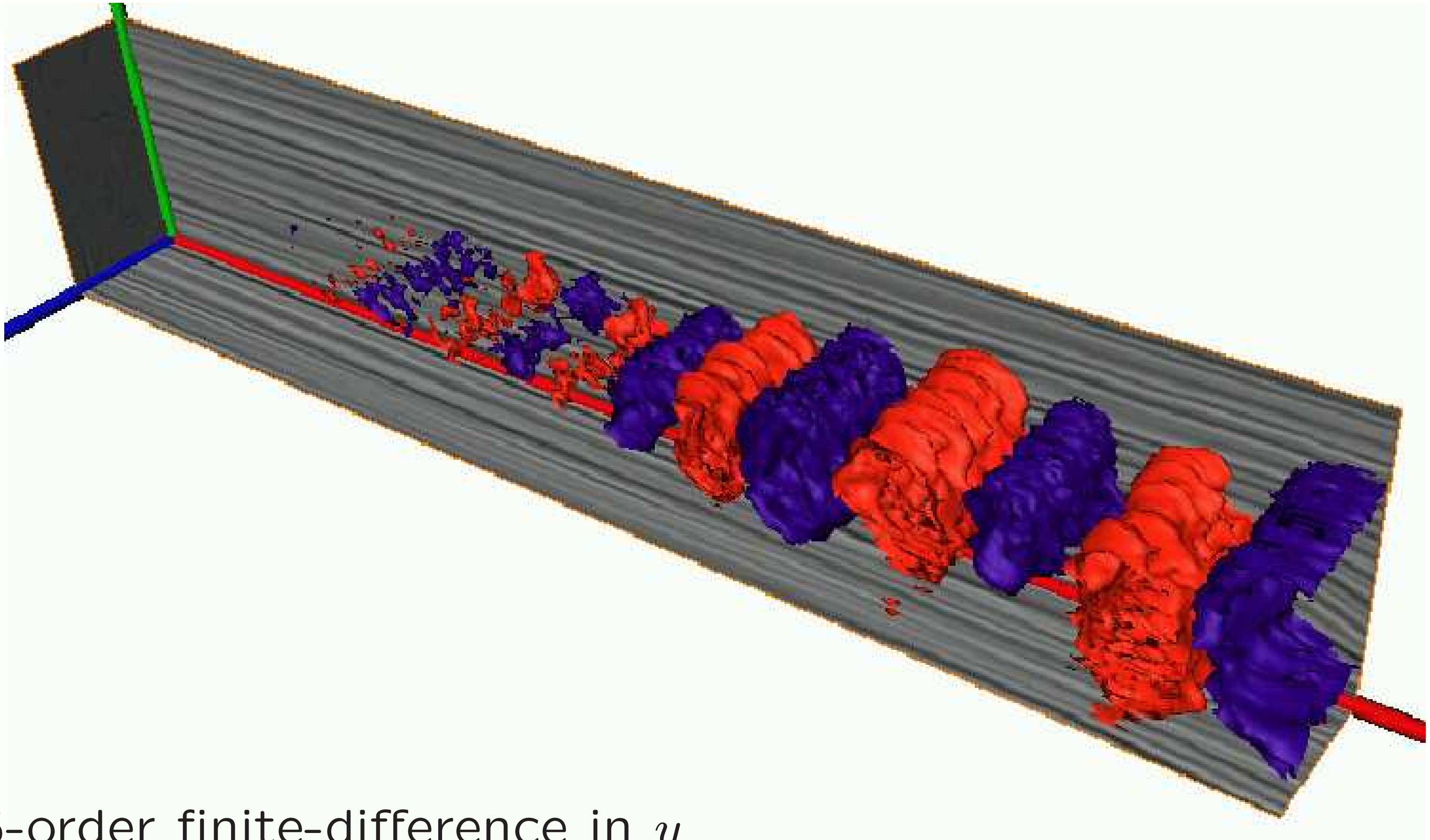
LES of mixing layer

at $Re \to \infty$

$U_1/U_2 = 3$

Visualization:

$v = \pm 0.04$



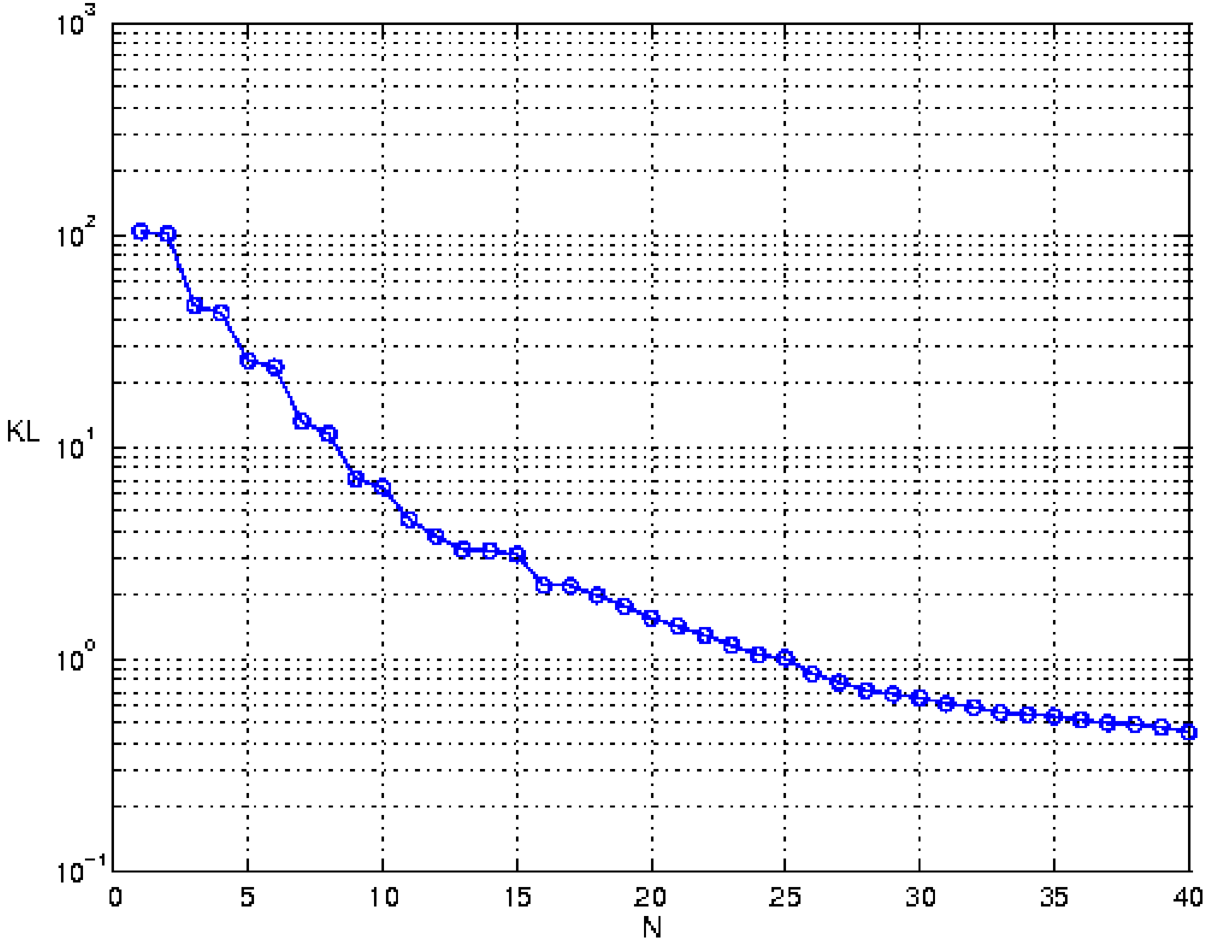
spectral in x,z, 6-order finite-difference in y

$0 \leq x/\delta_{sl} \leq 140$; $\quad -14 \leq y/\delta_{sl} \leq 14$; $\quad 0 \leq z/\delta_{sl} \leq 15$

POD of turbulent mixing layer

— ≡ *Noack, Pelivan, Comte, Morzyński & Tadmor (2004)* —

POD spectrum



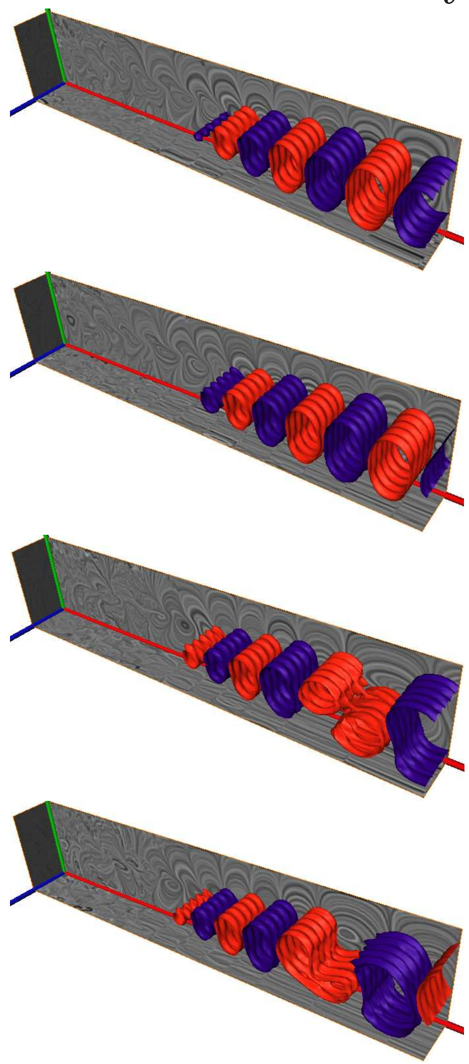
POD modes $\mathbf{u}_i$

i=1 *[22%]*

i=2 *[21%]*

i=3 *[10%]*

i=4 *[9%]*

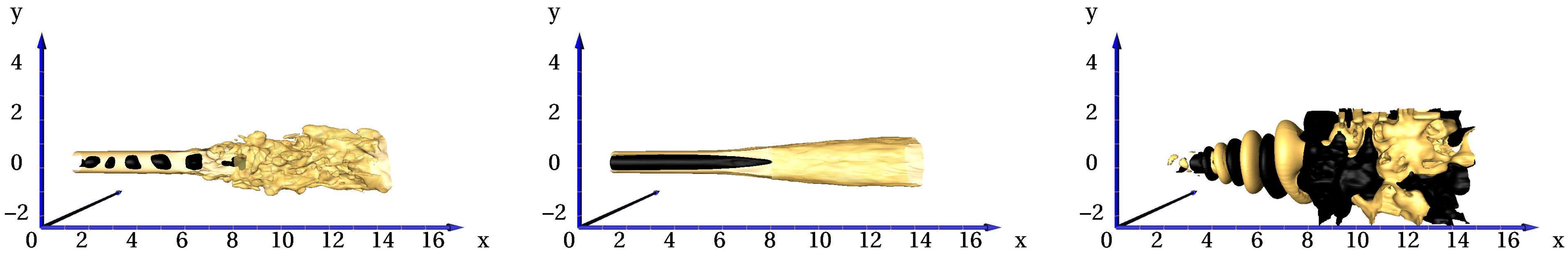


Jet noise LES (Re=3600, Ma=0.9)

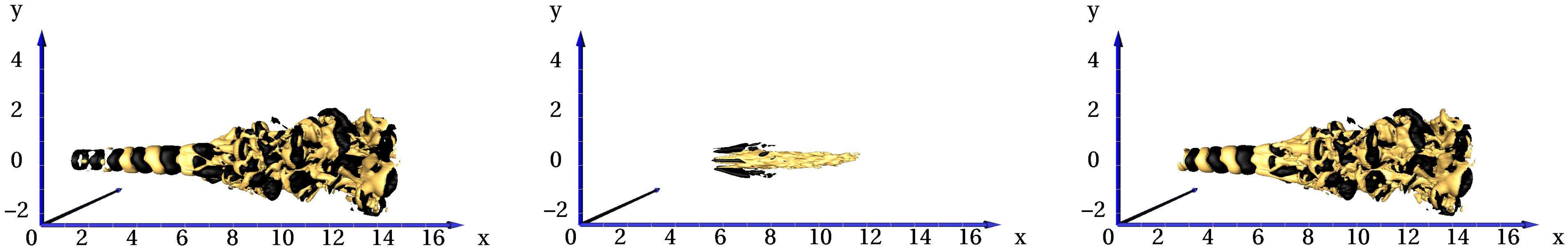
Gröschel, Schröder, Schlegel, Scouten, Noack & Comte (2006) CEMRACS

snapshot = average +fluctuation

Velocity field $\mathbf{u}$



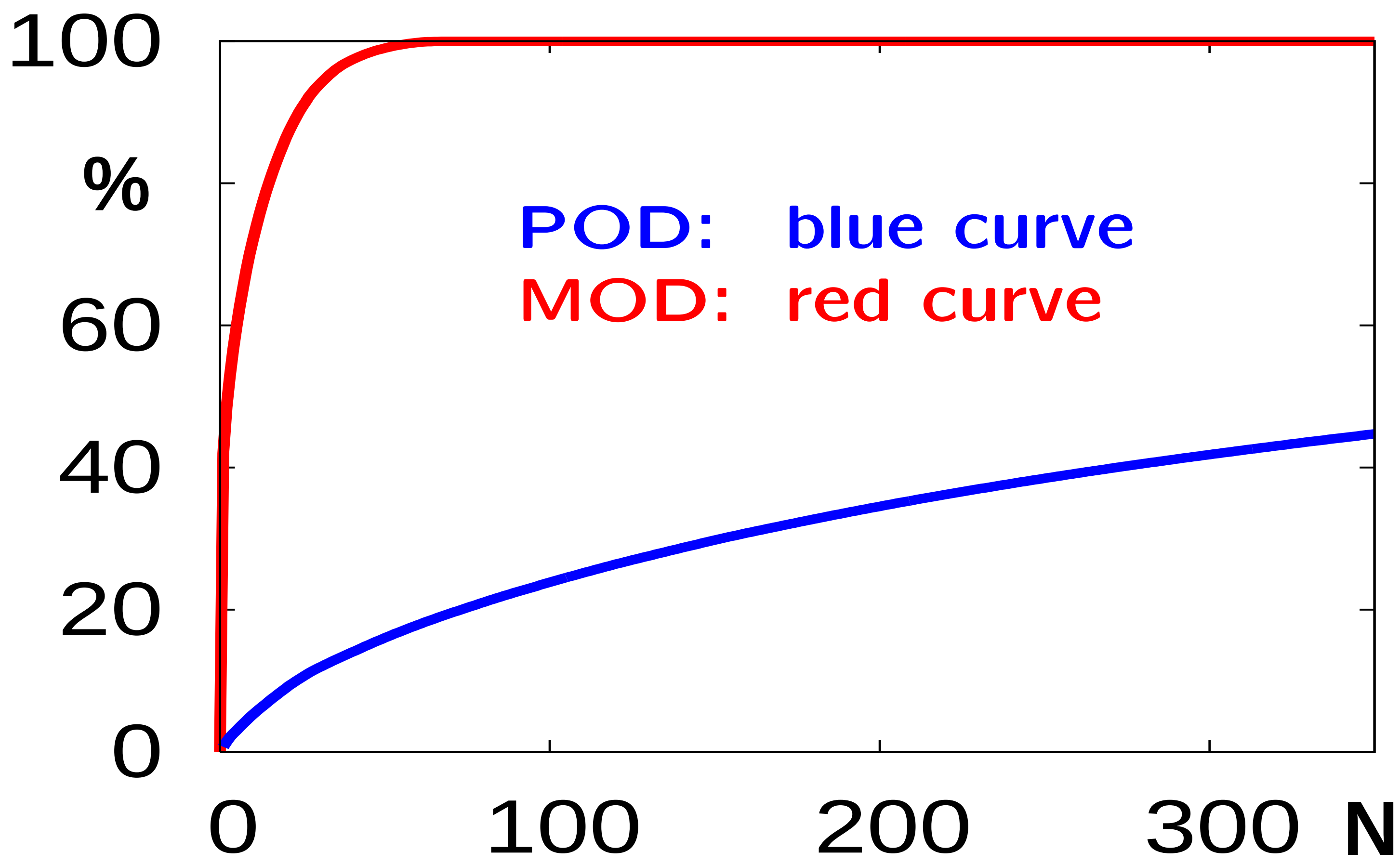
Acoustic source term of APE $\mathbf{L} = (\mathbf{u} \times \omega)'$



$M = 725$ snapshots

Noise resolution with POD and MOD

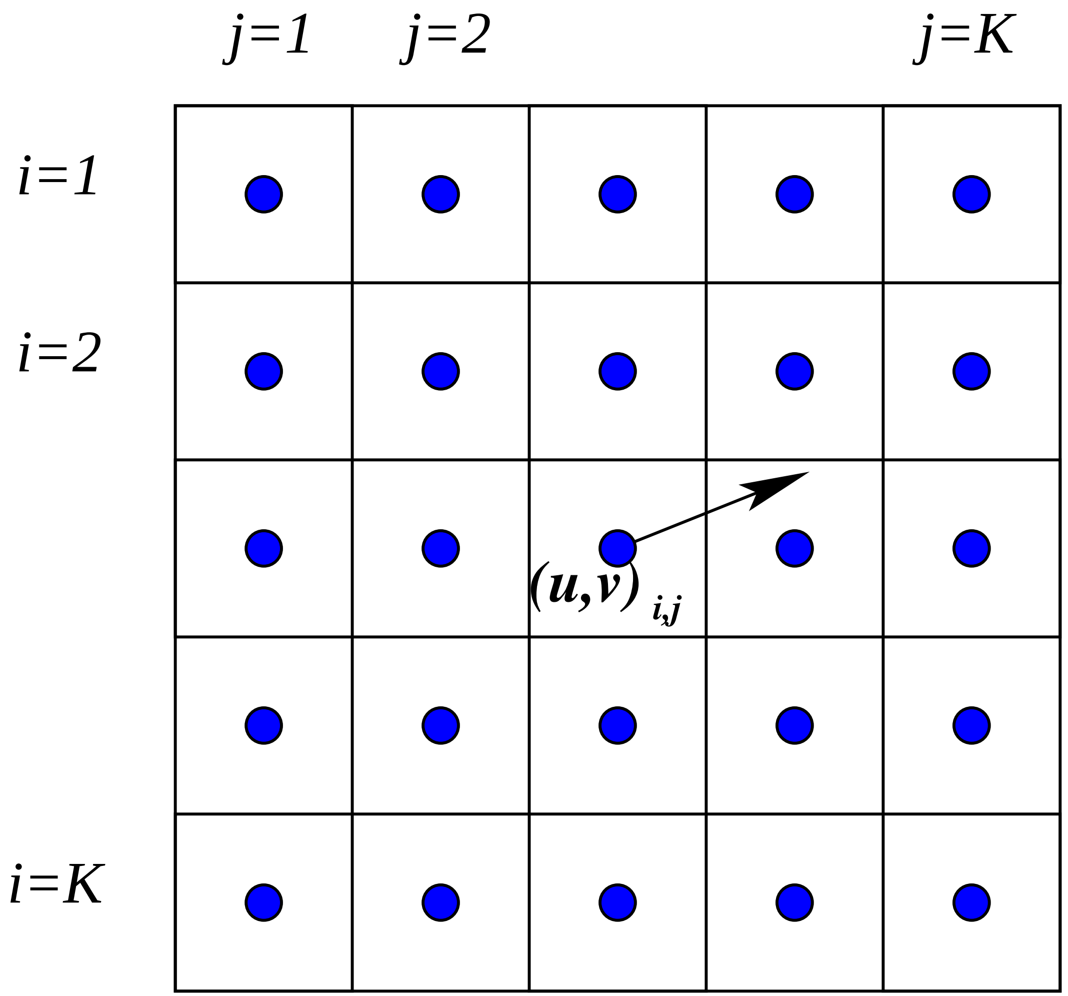
— Jordan, Schlegel, Stalnov, Noack & Tinney 2007 AIAA —



24 MOD modes $\mapsto$ 90% of far-field noise level

From flows to phase spaces

Discretized 2D flow



Phase space $\mathcal{R}^P$

$$
\begin{aligned}
u_1 &:= u_{11} \\
u_2 &:= u_{21} \\
u_3 &:= u_{31} \\
u_{K^2} &:= u_{KK} \\
u_{K^2+1} &:= v_{11} \\
\vdots & \quad \vdots \\
u_P &:= v_{KK} \\
P &= 2 \times K^2
\end{aligned}
$$

Autonomous SODE

Phase space $\mathcal{R}^P$

$$\mathbf{u} := (u_1, u_2, \ldots, u_P)^t$$

Dynamics

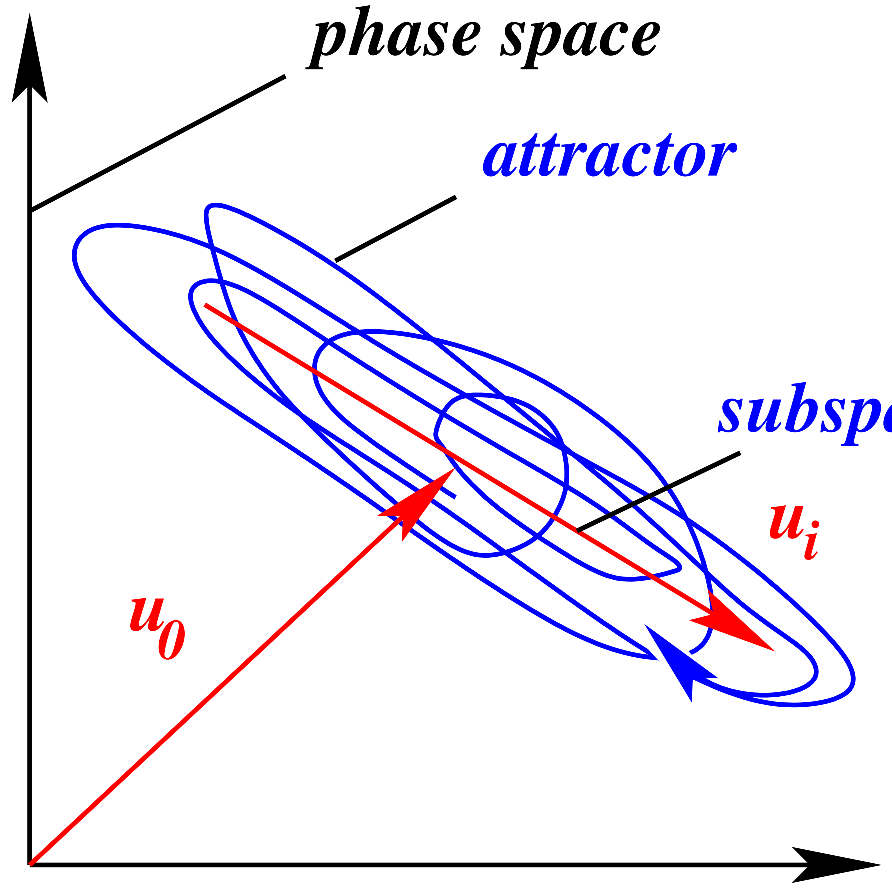
$$\frac{d\mathbf{u}}{dt} = \mathbf{F}(\mathbf{u})$$

Attractor $\mathcal{A} \subset \mathcal{R}^P$

$$\mathbf{u} \to \mathcal{A} \quad \text{as} \quad t \to \infty$$

characterized by ergodic measure $p_{\mathcal{A}}(\mathbf{u})$.

Idea of system reduction



Gaussian distribution of attractor

Attractor represented by trajectory $t \in [0, T] \mapsto \mathbf{u} \subset \mathcal{R}^P$

Gaussian probability distribution shall approximate attractor $p_{\mathcal{A}}(\mathbf{u})$

$$p_2(\mathbf{u}) = \frac{\sqrt{|Q|}}{(2\pi)^{P/2}} \exp\left[-\frac{1}{2}(\mathbf{u}-\mathbf{u}_0)^t\, Q\,(\mathbf{u}-\mathbf{u}_0)\right]$$

Matching the first statistical moments:

$$\mathbf{u}_0 = \overline{\mathbf{u}} = \int d\mathbf{u}\, p_2(\mathbf{u})\, \mathbf{u} = \frac{1}{T}\int_0^T dt\, \mathbf{u}$$

Matching the 2nd statistical moments of the fluctuation: $\mathbf{u}' := \mathbf{u} - \mathbf{u}_0$

$\boxed{Q = R^{-1}}$

$$R := \overline{\mathbf{u}' \otimes \mathbf{u}'} = \int d\mathbf{u}'\, p_2(\mathbf{u}_0 + \mathbf{u}')\, \mathbf{u}' \otimes \mathbf{u}' = \begin{pmatrix} \overline{u'_1 u'_1} & \overline{u'_1 u'_2} & \dots & \overline{u'_1 u'_P} \\ \overline{u'_2 u'_1} & \overline{u'_2 u'_2} & \dots & \overline{u'_2 u'_P} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{u'_P u'_1} & \overline{u'_P u'_2} & \dots & \overline{u'_P u'_P} \end{pmatrix}.$$

POD of attractor

Gaussian probability distribution of attractor $\mathcal{A} \subset \mathcal{R}^P$

$$p_2(\mathbf{u}) = \frac{\sqrt{|Q|}}{(2\pi)^{P/2}}\, e^{-\frac{1}{2}(\mathbf{u}-\mathbf{u}_0)^t\, Q\, (\mathbf{u}-\mathbf{u}_0)}, \qquad \mathbf{u}_0 = \overline{\mathbf{u}}, \qquad Q^{-1} = R = \overline{\mathbf{u}' \otimes \mathbf{u}'}$$

Principal axes

$$R\mathbf{u}_i = \lambda_i \mathbf{u}_i$$

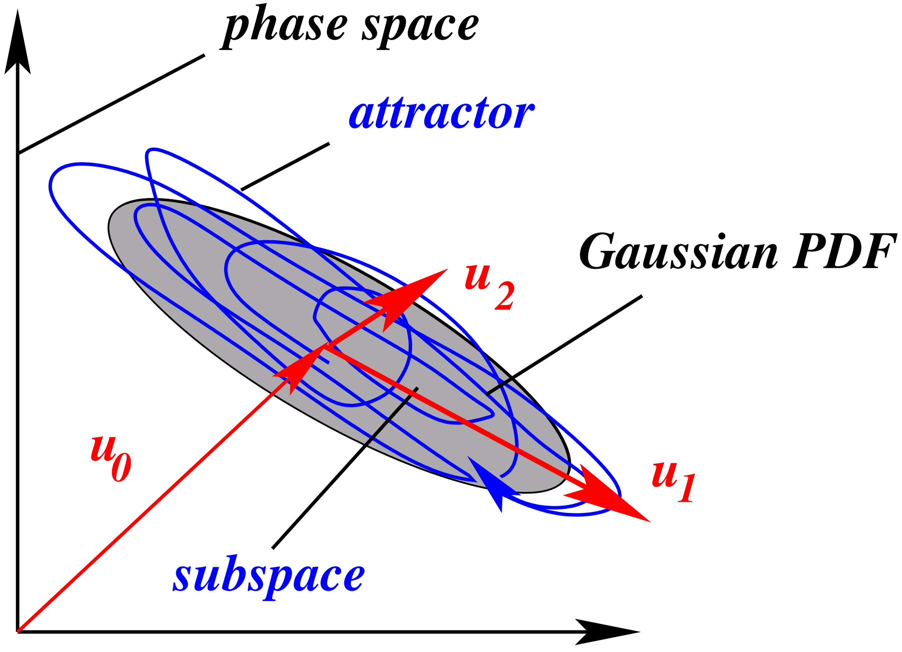
$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_P \geq 0.$

POD decomposition

$$\mathbf{u} = \mathbf{u}_0 + \sum_{i=1}^{P} a_i \mathbf{u}_i$$

Gaussian distribution

$$p_2(\mathbf{a}) = \frac{e^{-(\sum \frac{a_i^2}{\lambda_i})/2}}{\sqrt{(2\pi)^P \lambda_1 \ldots \lambda_P}}$$



$\mathbf{u}_i$: POD modes,

λ_i: POD eigenvalues

Properties of attractor POD

Trajectory

$$t \in [0, T] \mapsto \mathbf{u} \in \mathcal{R}^P$$

POD expansion $N \leq P$

$$\mathbf{u}^{[N]} = \mathbf{u}_0 + \sum_{i=1}^{N} a_i \mathbf{u}_i$$

Properties

- POD modes are orthogonal

$$\mathbf{u}_i \cdot \mathbf{u}_j = \delta_{ij}$$

- Mode amplitudes

$$a_i = \mathbf{u}' \cdot \mathbf{u}_i$$

- Statistics of mode amplitudes

$$\overline{a_i} = 0, \quad \overline{a_i a_j} = \delta_{ij} \lambda_i$$

Properties cont'd

- Correlation matrix

$$R := \overline{\mathbf{u}' \otimes \mathbf{u}'} = \sum_{i=1}^{P} \lambda_i \mathbf{u}_i \otimes \mathbf{u}_i$$

- Fluctuation energy (trace of $R/2$)

$$K = \frac{1}{2} \overline{\mathbf{u}' \cdot \mathbf{u}'} = \frac{1}{2} \sum_{i=1}^{P} \lambda_i$$

- **Optimality property:** Let

$$\mathbf{v}^{[N]} = \mathbf{v}_0 + \sum_{i=1}^{N} b_i \mathbf{v}_i$$

be any other expansion then

$$\overline{\|\mathbf{u} - \mathbf{u}^{[N]}\|^2} \leq \overline{\|\mathbf{u} - \mathbf{v}^{[N]}\|^2}$$

POD analysis — Nomenclature

$\langle \mathbf{F} \rangle := \frac{1}{M} \sum\limits_{m=1}^{M} \mathbf{F}^m$ .... ensemble average

$\langle \mathbf{F} \rangle_T := \frac{1}{T} \int\limits_0^T dt\, \mathbf{F}$ .... time average

$(\mathbf{F})_\Omega := \int\limits_\Omega dV\, \mathbf{F}$ .... volume integral

$[\mathbf{F}]_{\partial\Omega} := \int\limits_\Omega d\mathbf{A} \cdot \mathbf{F}$ .... surface integral

POD in continuum flow limit

Limit $K \to \infty$ for $\mathcal{R}^P$ = 2D flow on equidistant $K \times K$ grid

Quantity	$\mathcal{R}^P$	**flow**
state space	$\mathbf{u} = (u_1, \ldots, u_P)$	$\mathbf{u}(\mathbf{x})$
inner prod.	$\mathbf{u} \cdot \mathbf{v} = \sum u_i v_i \Delta x^2$	$(\mathbf{u}, \mathbf{v})_\Omega = \int dV_\Omega \mathbf{u} \cdot \mathbf{v}$
correlation	$\mathbf{R} = \overline{\mathbf{u}' \otimes \mathbf{u}'}$	$\mathbf{R}(\mathbf{x}, \mathbf{y}) = \overline{\mathbf{u}'(\mathbf{x}, t) \otimes \mathbf{u}'(\mathbf{y}, t)}$
Fredholm equation	$\mathbf{R}\mathbf{u}_i = \lambda_i \mathbf{u}_i$	$\int d\mathbf{y}\, \mathbf{R}(\mathbf{x}, \mathbf{y}) \cdot \mathbf{u}_i(\mathbf{y})$ $= \lambda_i\, \mathbf{u}_i(\mathbf{x})$
Exp.of $\mathbf{R}$	$\mathbf{R} = \sum \lambda_i \mathbf{u}_i \otimes \mathbf{u}_i$	$\mathbf{R}(\mathbf{x}, \mathbf{y}) = \sum \lambda_i \mathbf{u}_i(\mathbf{x}) \otimes \mathbf{u}_i(\mathbf{y})$
Exp. of K	$K = \frac{1}{2}\overline{\mathbf{u}' \cdot \mathbf{u}'} = \frac{1}{2}\sum \lambda_i$	$K = \frac{1}{2}\overline{(\mathbf{u}', \mathbf{u}')_\Omega} = \frac{1}{2}\sum \lambda_i$
GA	$\mathbf{u} = \mathbf{u}_0 + \sum a_i(t)\, \mathbf{u}_i$ $a_i = (\mathbf{u} - \mathbf{u}_0) \cdot \mathbf{u}_i$	$\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_0(\mathbf{x}) + \sum a_i(t)\, \mathbf{u}_i(\mathbf{x})$ $a_i = (\mathbf{u} - \mathbf{u}_0, \mathbf{u}_i)_\Omega$

POD (spatial vs. temporal formulation)

	spatial POD	temporal POD
$f(x,t) =$	$= \sum_{i=1}^{N} a_i(t)\, u_i(x)$	$= \sum_{i=1}^{N} a_i^\star(t)\, u_i^\star(x)$
eigenmodes	$u_i(x)$	$a_i^\star(t)$
normalisation	$(u_i, u_j)_\Omega = \delta_{ij}$	$\langle a_i^\star\, a_j^\star \rangle_T = \delta_{ij}$
bi-orthog.	$\langle a_i\, a_j \rangle_T = \lambda_i\, \delta_{ij}$	$(u_i^\star, u_j^\star)_\Omega = \lambda_i\, \delta_{ij}$
coefficients	$a_i = (f, u_i)$	$u_i^\star = \langle f\, a_i^\star \rangle$
correlation tensor	$R(x,y)$ $= \langle f(x,t) f(y,t) \rangle_T$	$R^\star(t,s)$ $= (f(x,t), f(x,s))_\Omega$
Fredholm equation	$\int dy\, R(x,y)\, u_i(y)$ $= \lambda_i u_i(x)$	$\frac{1}{T} \int ds\, R(t,s)\, a_i^\star(s)$ $= \lambda_i a_i^\star(t)$
	Note: $a_i^\star = a_i/\sqrt{\lambda_i},$	$u_i^\star = \sqrt{\lambda_i}\, u_i$
application	experimental data $P \ll M$	simulation data $P \gg M$

P: spatial dimension, M: number of snapshots

Overview

Alternatives to POD

(1) **Dynamic Mode Decomposition (DMD)**
Rowley+ 2009 JFM, Schmid 2010 JFM
ID of stability modes near steady solution;
Fourier modes on attractor

(2) **Recursive DMD (rDMD)** Noack+ 2016 JFM
Fourier-like modes with low residual

(3) **Extended POD (EPOD)** Hoarau 2006 PF
Linearly links flow to sensor data

(4) **Spectral POD (SPOD)** Sieber 2016 JFM
Interpolation between POD and Fourier modes

(5) **Spectral POD (SPOD)** *II* Towne+ 2018 JFM
Space and time-dependent modes

(6) **Convective POD (CPOD)** 2018 AFCC Schulze
POD-like modes for convection

... **Many more for many more purposes**