Predicting the temporal dynamics of turbulent channels through deep learning

G. Borrelli

The nature of turbulence

What is a turbulent flow? A flow characterized by a **chaotic** and **irregular** motion

Examples of turbulent flows are:

- Smoke of a cigarette
- Flow around aircraft wing-tips / wake of a boat
- Planetary atmospheres and ocean currents
- Flow in pipelines and ducts
- Geophysical flows
- Blood flow

Complex research: hard to define a model able to describe the structures and the mechanisms behind this phenomenon

1)

2)

1) https://www.alphr.com/science/1006713/what-is-turbulence-forecast/
2) https://www.jpl.nasa.gov/images/pia22944-dramatic-jupiter

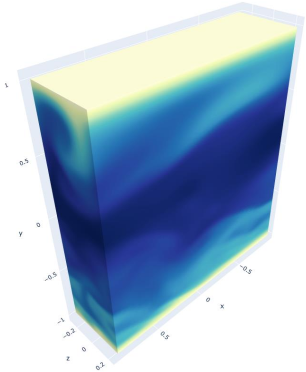
Research methods for turbulence

How can we deal with the complex nature of turbulence?

Experiments
- Expensive and time-consuming
- Limitation given by facilities and instruments

1) 

Numerical simulations
- Huge computational costs for direct numerical simulations (DNS)
- Limited to moderate Reynolds number and simple geometries (modelling necessary: LES, RANS)



Machine learning offers a third option to enrich the knowledge we have about turbulence

1) https://magazine.unibo.it/archivio/2021/03/10/turbolenza-e-sostenibilita-si-rafforza-il-laboratorio-ciclope

Machine learning and turbulence

When did machine learning and turbulence cross paths?

[1] In the early 1940s Kolmogorov considered turbulence as one of the main application of ML

Renewed interest in the 1980s due to advancement in CFD and development of backpropagation algorithm

Improved performance of deep neural networks (DNNs)

Turbulence is becoming always more a data-rich field

Fast-paced development of machine learning strategies applied to turbulence during last years

1) S. L. Brunton *et al.* “Machine learning for fluid mechanics “, 2020

Problems in machine learning

Nowadays machine learning is being employed in many different applications.

In turbulence some applications are:

- Flow modeling: improved results for RANS or LES.
- Identification of turbulent features
- Flow control strategies
- Generation of inflow conditions and extraction of flow patterns
- Prediction of temporal dynamics
- Machine learning-based reduced order models (ROMs)

Generation of turbulent data-sets employing limited initial information - avoiding experiments/simulations

Allowing to **extract governing physics** from large data-sets and **handle turbulent flows** efficiently

1) H. Eivazi *et al.*, "Recurrent neural networks and Koopman-based frameworks for temporal predictions in turbulence," 2020
2) T. Nakamura et al., "Convolutional neural network and long short-term memory based reduced order surrogate for minimal turbulent channel flow", 2021

Temporal predictions: low-order model

Previous studies from the same research group showed good predictions of **long-term statistics** and reproduction of **temporal dynamics** for a **9 equations low-dimensional model** (Moehlis *et al.* 2004)

- H. Eivazi *et al.*, "Recurrent neural networks and Koopman-based frameworks for temporal predictions in turbulence", 2020
- P.A. Srinivasan, *et al.* "Predictions of turbulent shear flows using deep neural networks", 2019

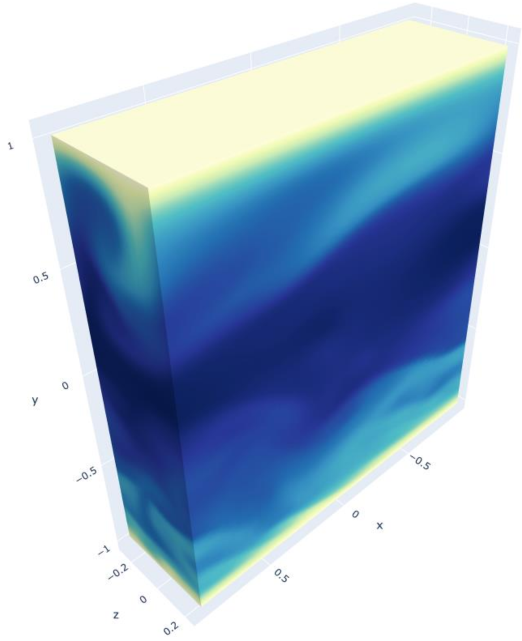
Limitation: low-order of complexity and governing equations are known **a priori**

How to improve?
Employ a data-driven framework, where the model is generated by the turbulent flow

Employed architectures in this thesis

LSTM: long-short term memory
KNF: Koopman with nonlinear forcing

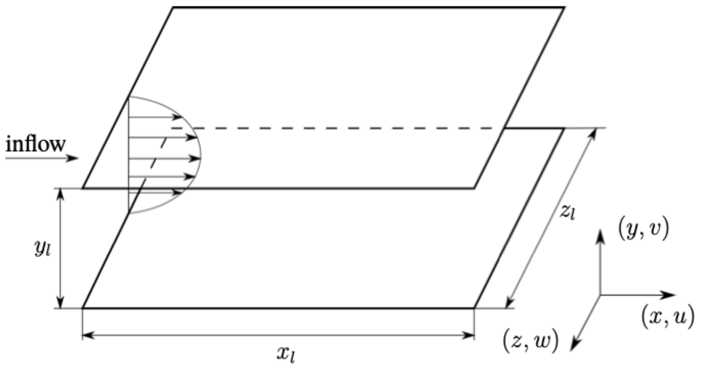
Question of the thesis



Is it possible to **predict the temporal evolution** of an **high-order** and **data-driven** representation of a **turbulent channel flow**?

- **High-order**: increased complexity
- **Data-driven**: model originated by turbulence itself

Why channel flows?



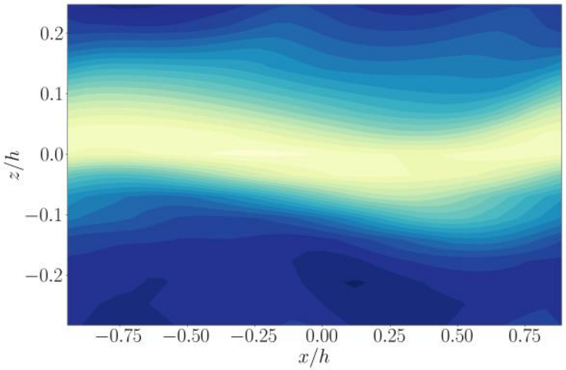
Wall-bounded flow

Turbulence generated by walls/flow interactions

- Investigation of flow's structures close to the wall
- Analysis of mechanisms of turbulence **production** and **transport**
- Formulation of hypotheses for **control** and **optimization**

Minimal channel flow

Why focus on minimal channel?



Good reproduction of turbulence and handy 'laboratory' [1)]

- Accurate low-order statistics close to the wall
- Single **low-velocity streak** in the **stream-wise** direction
- **Homogeneous** in the x- and z- directions

1) J. Jiménez and P. Moin, "The minimal flow unit in near-wall turbulence", 1991

Overview

Data-driven model

- Physical POD
- Fourier POD
- Energy-based truncation

Data-set analysis

- Frequency content
- Multi-step model

Results

LSTM predictions

- Temporal predictions
- Chaotic behaviour

Reduced order models

- ROMs-a: 100 modes
- ROMs-b: alternative

Koopman with nonlinear forcing

Data-driven model

Data-driven model
- Physical POD
- Fourier POD
- Energy-based truncation

Data-set analysis
- Frequency content
- Multi-step model

Results

LSTM predictions
- Temporal predictions
- Chaotic behaviour

Reduced order models
- ROMs-a: 100 modes
- ROMs-b: alternative

Koopman with nonlinear forcing

Data-driven model

How are we able to generate a data-driven model?

Proper orthogonal decomposition (POD)

Achieve **modal decomposition** between spatial and temporal description for a turbulent velocity field:

$$\boldsymbol{u}(\boldsymbol{x},t) = \sum_{j=1}^{m} a_j(t)\boldsymbol{\Phi}_j(\boldsymbol{x})$$

Data is re-arranged in a **snapshot matrix**

- N_p = number of grid points ($N_x \times N_y \times N_z$)
- N_t = number of time instants collected

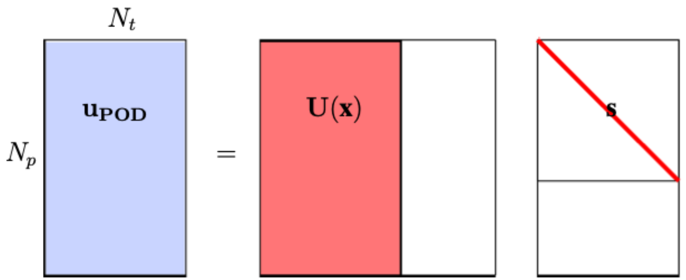
$$\mathbf{u}_{\mathbf{POD}} = \begin{bmatrix} u_{x_1}^{t_1} & \cdots & u_{x_1}^{t_{N_t}} \\ \vdots & \ddots & \vdots \\ u_{x_{N_p}}^{t_1} & \cdots & u_{x_{N_p}}^{t_{N_t}} \end{bmatrix}$$

Energy-based truncation to define the model

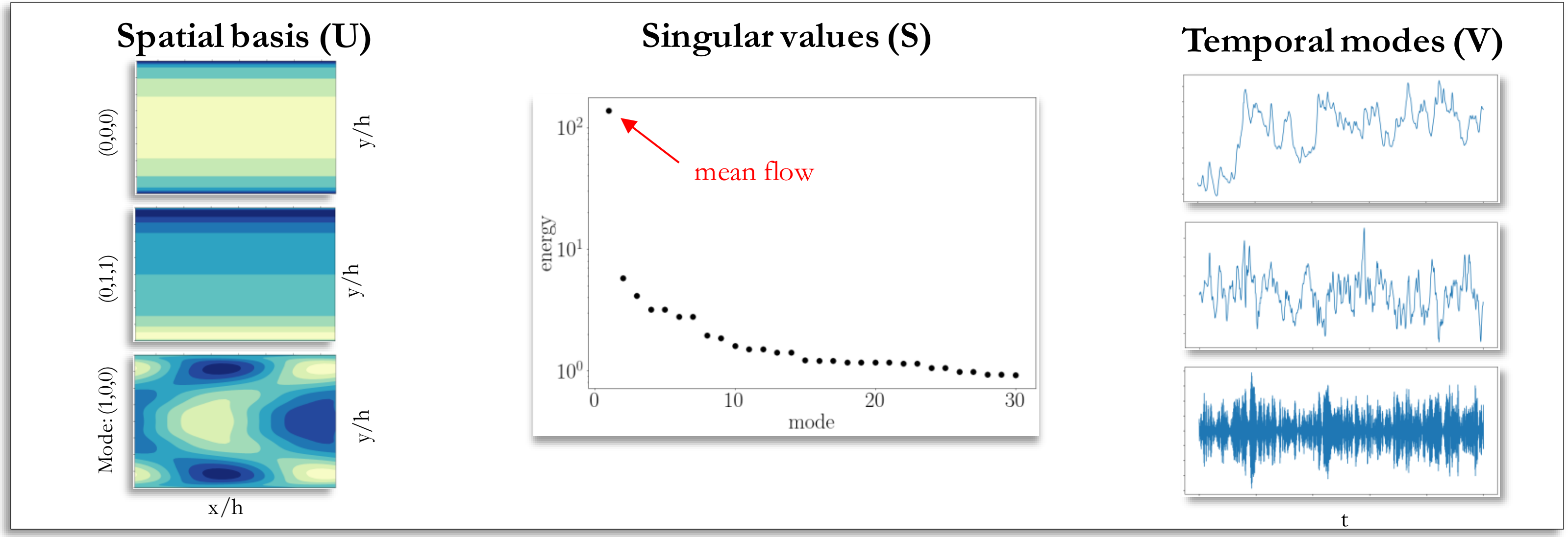
Optimal low-order reconstruction of the flow from the **energetic perspective**

Data-driven model: Physical POD

Proper orthogonal decompostion (POD) achieved by applying the **singular value decomposition** (**SVD**)



SVD economy-size is highlighted



Data-driven model: Fourier POD

Minimal channel is **homogeneous** in the x- and z-directions, thus the POD analysis can be performed directly on the **Fourier transform** of the fields in these directions[1)]

Benefits:

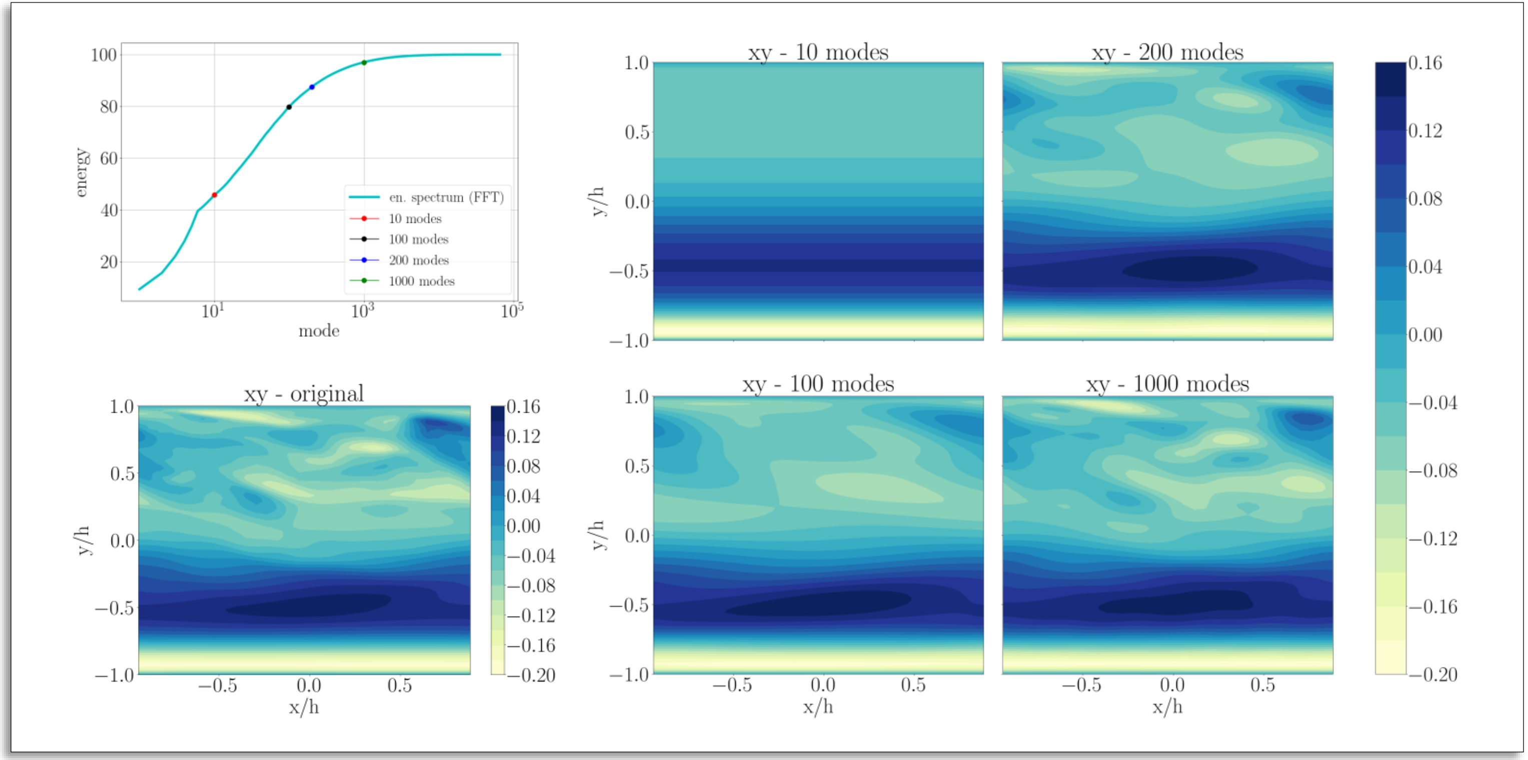
- SVD performed on a snapshot matrix of dimesion $(N_y \times N_t)$
- Lower computational cost and time of execution

Drawbacks:

- more articulated procedure
- complex coefficients
- SVD needs to be performed $(N_x \times N_z)$ times

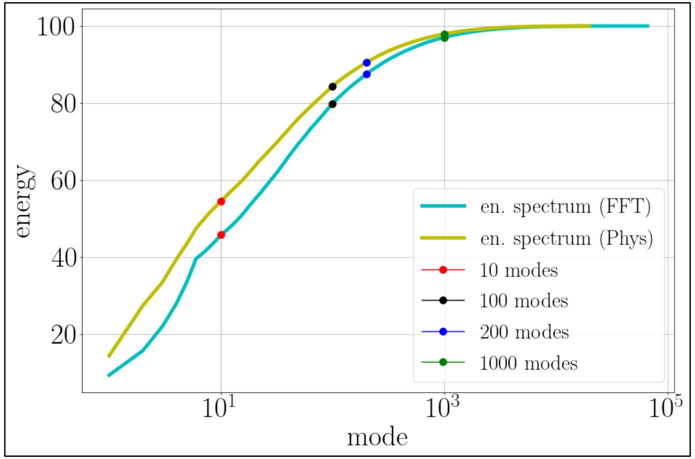
1) G. A. Webber *et al.*, "The Karhunen–Loéve decomposition of minimal channel flow", 1997

Data-driven model: Energy-based truncation



(Visualization after FFT POD **not** accounting for the mean)

Data-driven model: Energy-based truncation



Good agreement for the cumulative energy* calculated with both methods

Physical POD
- 10 modes = 54.48 %
- 100 modes = 84.34 %
- 200 modes = 90.48 %
- 1000 modes = 97.93 %

FFT POD
- 10 modes = 45.72 %
- 100 modes = 79.8 %
- 200 modes = 87.44 %
- 1000 modes = 97.04 %

And for errors in reconstruction: $\frac{||u_{orig} - u_{recon}||_2}{||u_{orig}||_2}$

Physical POD
- 10 modes = 37.96 %
- 100 modes = 22.55 %
- 200 modes = 17.82 %
- 1000 modes = 8.90 %

FFT POD
- 10 modes = 36.70 %
- 100 modes = 22.22 %
- 200 modes = 17.96 %
- 1000 modes = 8.78 %

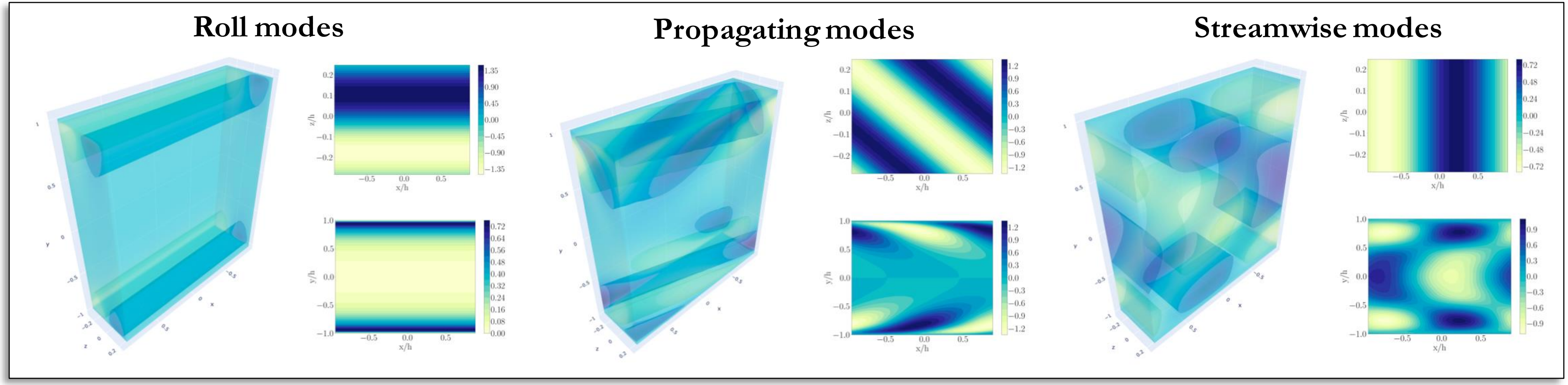
From now on the discussion is based on the **Fourier-POD** outputs. Analysis on the **x-component of the velocity fluctuations** only.

Example on a dataset of 10000 snapshots - *cumulative energy: $S^i = \sum_{j=1}^{i} \lambda_j \ / \sum_{j=1}^{N} \lambda_j, \ \lambda_j = s_j^2$

Data-driven model: Energy-based truncation

Data-driven model employs the first **100 most energetic modes** → **Is this model representative for the minimal channel turbulence?**

- **78.77 % of the energy** related to fluctuations is represented
- Turbulent structure typical of channel flows are recognized in the definition of the **spatial basis**



3D visualization of some of the coherent structures typically encountered in channel flows

Data-set analysis

Data-driven model
- Physical POD
- Fourier POD
- Energy-based truncation

Data-set analysis
- Frequency content
- Multi-step model

Results

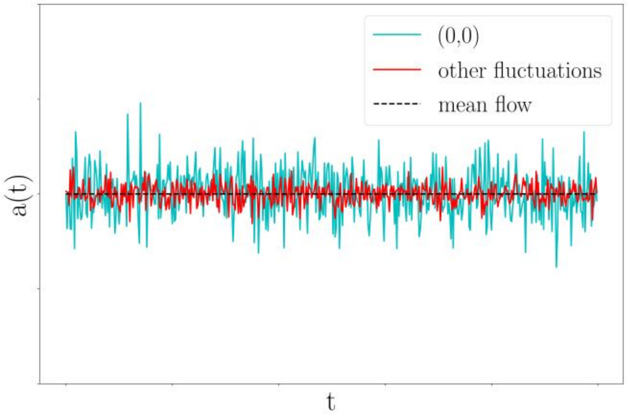
LSTM predictions
- Temporal predictions
- Chaotic behaviour

Reduced order models
- ROMs-a: 100 modes
- ROMs-b: alternative

Koopman with nonlinear forcing

Data-set analysis

Once we **fix** the **spatial basis**, we **focus on** the analysis of the **temporal modes**
These modes define the **data-set** employed for training and predictions



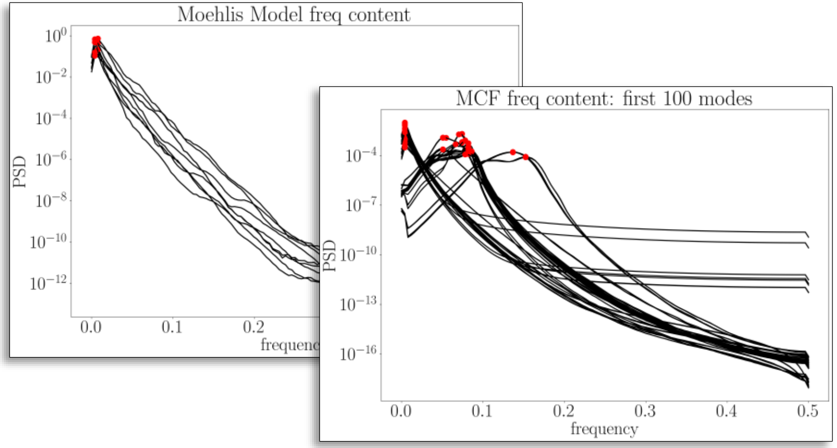
Decomposing temporal modes as:

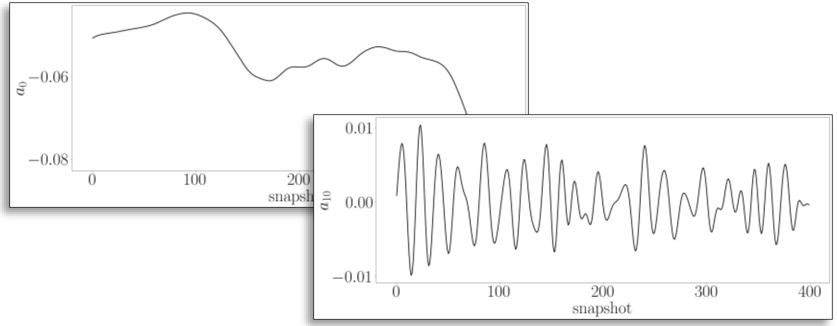
$$a(t) = \bar{a} + a'_{(0,0)}(t) + a'(t)$$

- Separating fluctuations related to wavenumber pair (0,0) - $a'_{(0,0)}$ - from the ones associated to the other wavenumbers - $a'(t)$
- **Predictions only on** $a'(t)$

100 modes account for **97.03 %** of the total energy related to fluctuations $a'(t)$
Data-set is scaled to have a better training

Data-set analysis: Frequency content





Moehlis: 9 modes with **same** frequency content
Minimal channel: group of modes with **different** frequency content

How do we deal with this aspect?

Multi-step concept

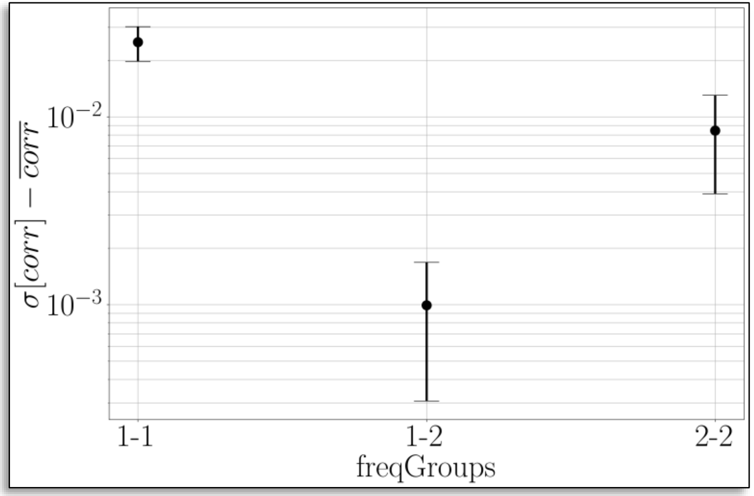
Dynamics is captured correctly for each mode

Different networks trained **separately** and responsible for the prediction of a reduced set of modes

Data-set analysis: Multi-step model

Can we apply a multi-step concept to our case?

Yes, by looking at the correlation of signals from different groups



Group [1]: low-frequency signals
Group [2]: mid/high frequency signals

If **correlation** between signals of the two groups is **low** we can train **different networks separately**

Long-short term memory predictions

Data-driven model
- Physical POD
- Fourier POD
- Energy-based truncation

Data-set analysis
- Frequency content
- Multi-step model

Results

LSTM predictions
- Temporal predictions
- Chaotic behaviour

Reduced order models
- Physical POD
- Fourier POD
- Energy-based truncation

Koopman with nonlinear forcing

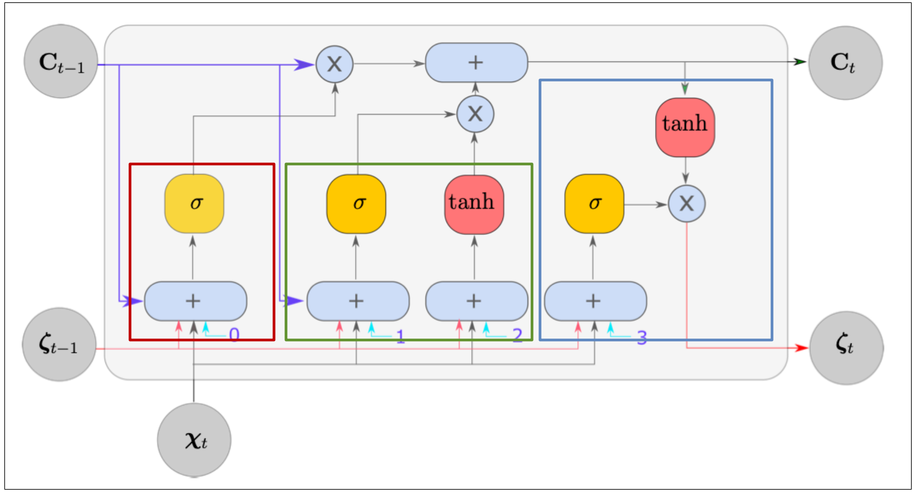
Long-short term memory

The success of **recurrent neural networks** (RNNs) to achieve the training on **sequential data** has been demonstrated

Long-short term memory networks (LSTMs) are ideal for **long-term** dependencies as the ones in **turbulent flows**

Gate mechanism:

- **Forget gate:** current input (χ_t) and previous output (ζ_{t-1}) to **define fraction of cell state** (C_{t-1}) **to keep**
- **Input gate:** compute **candidates** for the update of the **new cell state** ($\widetilde{C}_t$)
- **Output gate:** compute the **output** values (C_t) with the **new cell-state** (ζ_t)

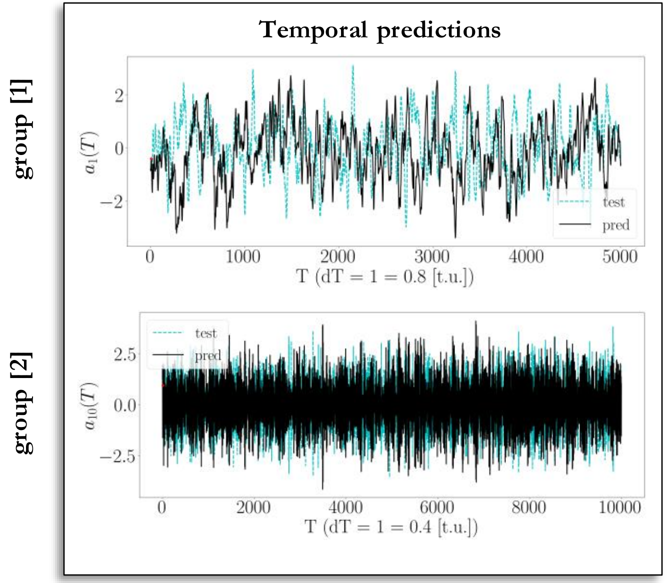


1) http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Long-short term memory predictions

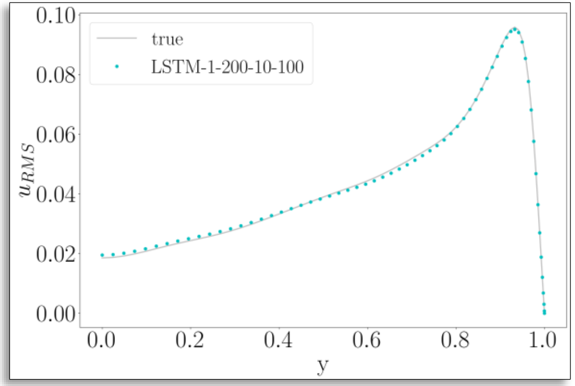
Idea: select an initial sequence of length *p* to predict the value at *p+1*

LSTM-1-200-10-100*



- Instantaneous predictions are in the correct range and frequency content is well reproduced
- Excellent agreement for the statistics

Best model:
$E_{u_{RMS}} = 1.3\ \%$



*LSTM-ly-cells-p-N: ly – number of layers, cells - number of neurons, p – initial sequence length, N – dataset dimension (x1000)

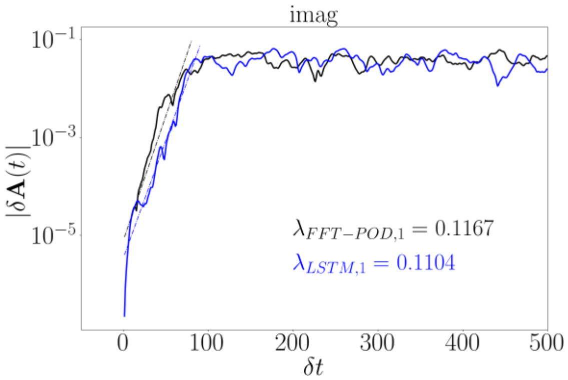
LSTM predictions: Chaotic behaviour

Is long-short term memory able to reproduce the chaotic physics of the minimal channel?

Analysis of the **Lyapunov exponent** (λ)

$\boldsymbol{\lambda > 0}$ is an indication that the system is **chaotic**

- **Infinitesimal perturbation** is introduced to generate a **perturbed data-set** ($\lambda_{FFT-POD}$)
- Effect of perturbation is analysed also for the predictions (λ_{LSTM})



real

	λ_1	λ_2
Data-driven	0.1258	0.1387
LSTM	0.1015	0.0852

imaginary

	λ_1	λ_2
Data-driven	0.1167	0.1763
LSTM	0.1104	0.1873

Example is reported for the imaginary part of the coefficients (evolution of the difference between trajectories)

Reduced order models

Data-driven model
- Physical POD
- Fourier POD
- Energy-based truncation

Data-set analysis
- Frequency content
- Multi-step model

Results

LSTM predictions
- Temporal predictions
- Chaotic behaviour

Reduced order models
- ROMs-a: 100 modes
- ROMs-b: alternative

Koopman with nonlinear forcing

Reduced order models

Why are reduced order models important?

- Represent most relevant structure of turbulence
- Extract the governing physics from large data-sets
- Less time needed to train a network

Energy-based **data-driven model** derived for this thesis is a reduced order model itself

We investigate the possibility of training a network employing **less modes** which are **representative for a group of turbulent structure**

ROMs-a: predicting* all the **100 modes** of the data-driven model

ROMs-b: predicting* only the modes employed for the training

*Predictions performed with LSTM network

Reduced order models

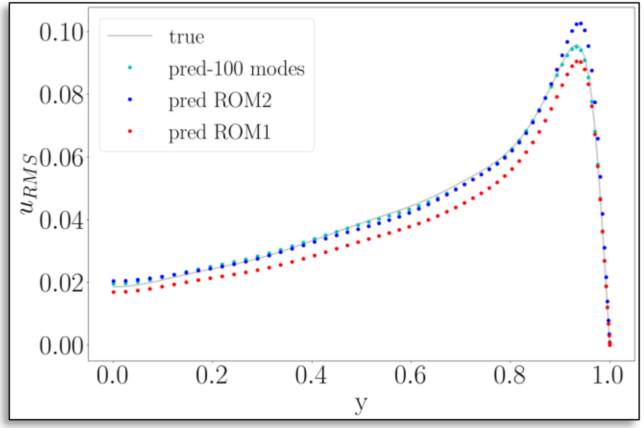
How can we choose representative modes for a turbulent feature?

6 sub-groups in the first 100 modes sharing some features in the homogeneous directions – excluding group of modes related to (0,0)

↓

If **high correlation** between signals of the **same sub-group** then less modes can be employed to describe the others

Reduced order models: ROMs-a



Example: **ROM2 –** 2 modes for each wv-pair

Low Freq.

- **(0,1):** 10 modes **1, 4,** 7, 9, 24, 26, 42, 60, 64, 94
- **(0,2):** 2 modes **44, 46**

Mid/high Freq.

- **(1,0):** 9 modes **19, 21**, 37, 39, 47, 50, 63, 74, 95
- **(1,1):** 16 modes **10, 12**, 14, 16, 29, 30, 32, 34, 51, 54, 56, 58, 77, 85, 90, 91
- **(1,2):** 4 modes **78, 80**, 82, 86
- **(2,1):** 5 modes **67, 69**, 70, 72, 98

(**in red** modes employed for the **training** of the specific ROM)

architecture	$E_{\overline{u'}}$ [%]	$E_{u_{RMS}}$ [%]
Data-driven	0.36	2.97±1.17
ROM2	0.61	8.02±2.19
ROM1	0.69	12.98±3.46

Increased errors as less modes are employed for training

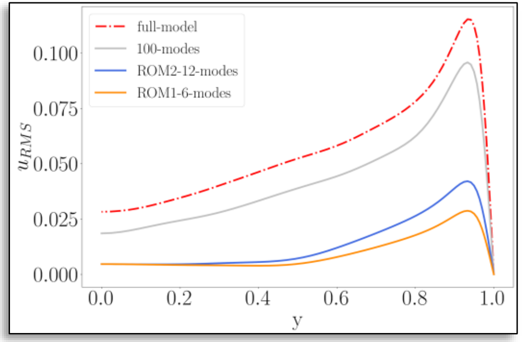
Only best models are reported in the figure
Averaged results over 3 models is reported in the table (stochastic nature of the training)

Architecture: LSTM-1-200-10-100

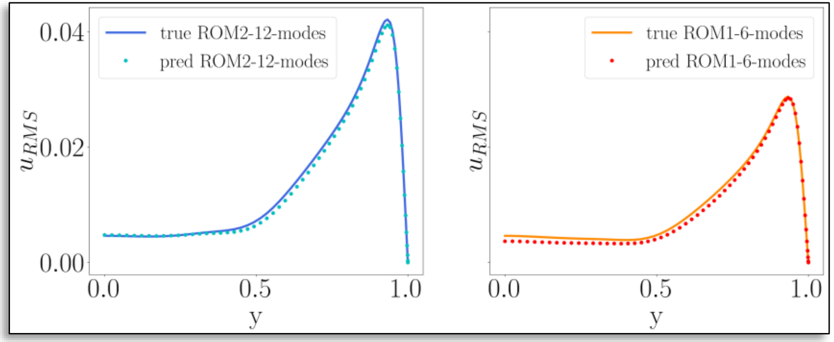
Reduced order models: ROMs-b

Only predicting the modes used for the training

Velocity fields reconstructed using only 6 or 12 modes



(Stream-wise fluctuations are reduced when less modes are considered)



architecture	Energy %*	$E_{u_{RMS}}$ [%]
Data-driven	97.05	2.97±1.17
ROM2-12modes	20.71	6.69±3.52
ROM1-6modes	10.74	8.51±5.06

*Cumulative energy for $a'(t)$ fluctuations only

Increased errors and less energy as less modes are employed for training

Koopman with nonlinear forcing

Data-driven model
- Physical POD
- Fourier POD
- Energy-based truncation

Data-set analysis
- Frequency content
- Multi-step model

Results

LSTM predictions
- Temporal predictions
- Chaotic behaviour

Reduced order models
- ROMs-a: 100 modes
- ROMs-b: alternative

Koopman with nonlinear forcing

Koopman with non-linear forcing

- **Idea:** Employ a **linear operator** on an **infinite-dimensional** space to describe a non-linear behaviour in a finite domain
- **Non-linearities** are modelled through an **external forcing** [1)]

How can we predict the state at the next time instant?

Given a dynamical system

$$\boldsymbol{x}_{t+1} = \boldsymbol{F}(x_t)$$

Koopman based framework:

$$\boldsymbol{x}^{m+1} = \boldsymbol{A}\boldsymbol{x}^m + \boldsymbol{B}\boldsymbol{f}^m$$

By knowing the maps $\boldsymbol{A}$ and $\boldsymbol{B}$ it is possible to advance the solution in time

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{x}^1 & \cdots & \boldsymbol{x}^{N-q} \\ \vdots & \ddots & \vdots \\ \boldsymbol{x}^q & \cdots & \boldsymbol{x}^{N-1} \end{bmatrix}$$

Hankel representation of state $\boldsymbol{x}$ and forcing term $\boldsymbol{f}$

Time-embedded form

$$\boldsymbol{H}_x^{m+1} = \boldsymbol{A}\boldsymbol{H}_x{}^m + \boldsymbol{B}\boldsymbol{H}_f{}^m$$

DMDc to find maps [2)]

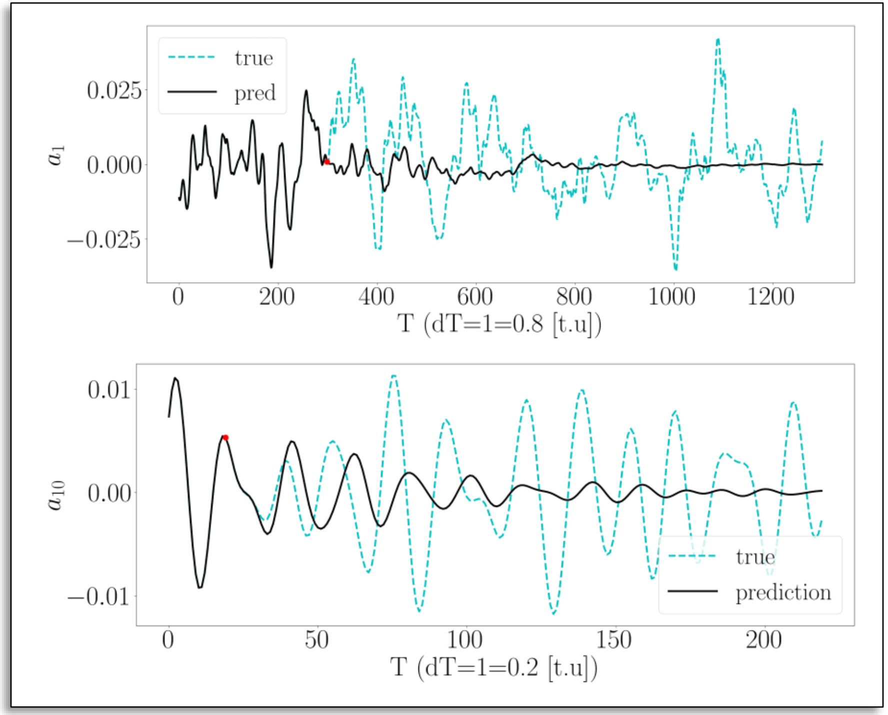
1) https://www.sciencedirect.com/science/article/pii/S0021999121003077?via%3Dihub
2) Proctor, J.L *et al.*, "Dynamic mode decomposition with control", 2016

KNF predictions

Data-driven model generates a dynamics which might be too complex for KNF

Low freq

Mid/high freq



(We performed several tests by changing the configuration without achieving significant improvements)

Conclusion and future work

Discussion

- The benefits of the **Fourier-POD** has been demonstrated from the computational standpoint.
- A **data-driven** with **100 modes** has been derived through an energy-based truncation.
- A **multi-step model** has been implemented based on the frequency content.
- **LSTM** network has lead to **excellent** predictions of the **statistics** and to a **good reproduction** of the **chaotic** physics for the data-driven model.
- Reduced order models have been investigated, accounting for the **turbulent structures**
- We have not been able to capture minimal channel dynamics with **KNF**

Future developments?

- **Extend POD** over the 3 components of the velocity to asses the feasibility of the neural networks to predict quantities as the Reynolds shear stress.
- Include predictions for the fluctuations related to **wavenumber pair (0,0).**
- Elaborate a **more articulated KNF** algorithm to deal with more complex dynamics.
- **Improve** the accuracy of the **ROMs**.
- Implement more sofisticated networks to improve short-term predicions.
- Apply data-driven approach to **other canonical cases.**

Thanks for your attention!

Questions?