Modal Decomposition of Turbulent Fluid Flow using Proper Orthogonal Decomposition and Autoencoders

Mikael Häglund
mikaelhagl@gmail.com

under the direction of
Prof. Christophe Duwig
Department of Chemical Engineering
KTH Royal Institute of Technology

Research Academy for Young Scientists
July 13, 2022

Abstract

The study of turbulent fluid flow often relies on computer simulations, because of the chaotic nature of fluid dynamics. Modal decomposition aims to extract dominant turbulent structures, called modes, from a simulation of a fluid flow. Modes can be analysed by themselves, and by combining the most dominant modes, a reduced-order model of the flow can be reconstructed, giving a less accurate but less complex model of the flow. Proper Orthogonal Decomposition (POD) is widely used for the properties of the modes it extracts. Autoencoders are artificial neural networks with architecture suitable for feature extraction. These two methods were applied to velocity data obtained from direct numerical simulation of a gas mixture containing N_2O_4 and NO_2 through a channel with a heated bottom wall. The POD modes were used to identify dominant turbulent structures of the flow, and the reconstruction quality from using POD and the autoencoder was compared. Due to the lack of clear turbulent structure in the flow, applying POD was not very effective in capturing turbulent features, losing 52.2% of the turbulent x-velocity and 75.2% of the turbulent y-velocity when using 20 modes. However, the autoencoder model was able to capture more of the turbulent velocity, losing 26.9% of x-velocity and 48.8% of y-velocity when using 2 modes, and only 17.1% and 32.5%, respectively, when using 20 modes.

Acknowledgements

I would like to thank Prof. Christophe Duwig for being my mentor for the project, providing support and guidance. I would also like to thank Dr. Kai Zhang and Dr. Marc Rovira for introducing me to the the theory behind my project, for providing insightful ideas, help, and for answering all the questions that I asked when I was confused. I would like to thank Antoni Kowalik for his invaluable help in making my programs run, going out of his way to help me, and for keeping spirits up when things were going slow and the office was too hot. Finally, I would like to thank the organisers of Rays — for excellence, and their partners Beijerstiftelsen and AstraZeneca for making this project possible.

Contents

1 Introduction **1**

1.1 Modal Decomposition 1

1.2 Proper Orthogonal Decomposition 2

1.3 Modal Decomposition using Autoencoders 6

1.4 Aim of Study 7

2 Method **7**

3 Results **9**

3.1 Modes obtained using POD 9

3.2 Reconstructions using POD and the Autoencoder 11

4 Discussion **11**

4.1 Structures in POD modes 12

4.2 Comparison between POD and AE reconstructions 13

4.3 Conclusion 14

References **15**

A POD modes 3-5 **17**

B Reconstructions of the flow using the Autoencoder and POD **18**

1 Introduction

The movement of fluids appears everywhere in nature and at all scales. The analysis of fluid flow is applicable to a wide range of engineering fields, including developing efficient combustion [1], studying heat recovery [2], and studying biotechnological processes [3], as well as advancing the theory of fluid dynamics. The Navier Stokes equations that govern turbulent fluid flow cannot generally be solved analytically, implying a large reliance on numerical and experimental methods for describing the complex flows that occur, even under elementary flow patterns. Since it is often difficult and expensive to perform large-scale experiments with fluids, computer simulations are commonly used to model fluid flow. The modelling of a fluid flow by computers is done by solving the governing equations of the system numerically. Various algorithms can be employed to perform this task. One such algorithm is Direct Numerical Simulation (DNS), which solves the Navier-Stokes equations directly on all scales, without using any turbulence modelling. In order to create an accurate model with DNS, a fine grid of points are required, making DNS a very computationally expensive task [4]. In contrast, Large Eddy Simulation (LES) only computes larger energy-containing scales directly, and smaller scales are modelled by turbulence models that combine experimental observations with theoretical knowledge [5]. The idea of LES is that the flow on small scales is not affected much by larger features of the flow, so little information is lost from not solving for them explicitly. While LES saves on computation compared to DNS, it is not as accurate [4].

1.1 Modal Decomposition

To analyse fluid flow, a commonly applied method has been to extract dominant features from the flow, called modes, to then obtain information about the original flow using the modes. The extraction of dominant features is motivated by observations of common flow features that appear over a large variety of flows and parameters like the Reynolds number, a measure of internal forces compared to frictional forces, as well as different

geometries of the space of the flow [6]. Additionally, as mentioned above, performing simulations of flows with DNS or similar methods is computationally expensive, so working with simpler models that still capture the most important features of the flow is essential for the applicability of computational fluid dynamics. Through modal decomposition, a reconstruction of the original flow can be obtained by only keeping the most contributing modes. This is called a reduced order model. The goal of any reduced order modelling is to reduce the available degrees of freedom while still retaining the information required to model the system with acceptable accuracy [7].

1.2 Proper Orthogonal Decomposition

Proper Orthogonal Decomposition (POD) is a modal decomposition technique developed by Lumley [8], that has been widely used for its properties beneficial in analysing fluid flows. POD also serves as a basis for many other modal decomposition techniques [6]. POD extracts spatial modes with time coefficients to express the original flow $\boldsymbol{q}(\boldsymbol{\xi}, t)$ as

$$\boldsymbol{q}(\boldsymbol{\xi}, t) = \overline{\boldsymbol{q}}(\boldsymbol{\xi}) + \sum_i a_i(t)\boldsymbol{\phi}_i(\boldsymbol{\xi}). \tag{1}$$

Here, $\boldsymbol{\xi}$ is a vector used to represent the space of the fluid flow, t is time, and $\boldsymbol{q}$ is the value of some parameter at a certain point and time. Parameters that are often studied include velocity, pressure, and temperature. The term $\overline{\boldsymbol{q}}(\boldsymbol{\xi})$ is the temporal mean of the studied parameter: the mean value over all measured time instances, computed for all points. When this quantity is subtracted from the original flow, the unsteady or turbulent component of the flow remains, which is then decomposed into modes using POD. The modes $\boldsymbol{\phi}_i$ depend only on the position in the space and are thus often referred to as spatial modes. Multiplying the modes by their respective temporal (time) coefficient a_i and summing over all such modes gives the decomposition of the original flow.

The way that the spatial modes are constructed with POD gives it two important properties: optimality and orthogonality. The optimality of POD means that among all

linear decompositions, POD is able to extract the most features from the flow with the least amount of modes. If the measured quantity $\boldsymbol{q}$ is velocity, this has the concrete implication that POD modes, on average, capture the most amount of turbulent kinetic energy in the flow among all linear decompositions. Orthogonality implies that POD modes are orthogonal, and consequently that the time coefficients of the modes are linearly uncorrelated, which is a useful property when constructing reduced order models [9].

We now motivate the two useful properties of POD stated above. The input of POD is a collection of snapshots, carrying information about the flow at a number of distinct time instances. Since we wish to decompose the time fluctuating part of the flow as in Eq. (1), the mean flow is first subtracted from each snapshot to obtain the data

$$\boldsymbol{x}(t_j) = \boldsymbol{q}(\boldsymbol{\xi}, t_j) - \overline{\boldsymbol{q}}(\boldsymbol{\xi}), \quad j = 1, 2, \ldots, m, \tag{2}$$

where m is the total number of snapshots. Let n be the number of points in the space of the flow, then the column vector $\boldsymbol{x}(t)$ has n components. The data from Eq. (2) are arranged into a matrix,

$$\boldsymbol{X} = [\boldsymbol{x}(t_1)\ \boldsymbol{x}(t_2)\ \cdots\ \boldsymbol{x}(t_m)] \in \mathbb{R}^{n \times m}, \tag{3}$$

where each column in $\boldsymbol{X}$ has the spatial data from one snapshot. The covariance matrix[1] $\boldsymbol{R}$ of $\boldsymbol{x}(t)$ for the m snapshots is constructed as

$$\boldsymbol{R} = \boldsymbol{X}\boldsymbol{X}^T. \tag{4}$$

The POD modes are the *eigenvectors* of $\boldsymbol{R}$, in other words, they are solutions to

$$\boldsymbol{R}\boldsymbol{\phi}_i = \lambda_i \boldsymbol{\phi}_i \tag{5}$$

[1]The actual covariance matrix has the factor $\frac{1}{m}$ or $\frac{1}{m-1}$ in front of $\boldsymbol{X}\boldsymbol{X}^T$, but this factor just scales the eigenvalues in Eq. (5) and therefore carries no significance.

where $\lambda_i \geqslant 0$ are the corresponding *eigenvalues* [10]. The eigenvalues are all real and non-negative since the covariance matrix is symmetric positive semidefinite. The time coefficients $a_i(t)$ are given by

$$a_i(t_j) = \frac{1}{n} \sum_{l=1}^{n} \boldsymbol{x}(\xi_l, t_j) \boldsymbol{\phi}_i(\xi_l). \tag{6}$$

Since the eigenvectors of $\boldsymbol{R}$ are orthogonal ($\boldsymbol{R}$ is symmetric), the spatial modes $\boldsymbol{\phi}_i$ are orthogonal. Moreover, if the modes are indexed such that the eigenvalues are ordered as

$$\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_n \geqslant 0, \tag{7}$$

the modes become ordered in terms of how well they fit the data [6]. The optimality of the POD modes also comes from the modes being solutions to Eq. (5), since the k:th solution minimises the mean-square error of the reconstruction using k modes, for all k. This comes from the way the matrix $\boldsymbol{R}$ was constructed. If a reconstruction with k modes $\boldsymbol{q}_k$ is defined at the point ξ_i and time t_j as

$$\boldsymbol{q}_k(\xi_i, t_j) = \overline{\boldsymbol{q}}(\xi_i) + \sum_{l=1}^{k} a_l(t_j) \boldsymbol{\phi}_l(\xi_i), \tag{8}$$

then the mean-square error of the reconstruction is

$$\text{MSE} = \frac{1}{mn} \sum_{i,j} (\boldsymbol{q}(\xi_i, t_j) - \boldsymbol{q}_k(\xi_i, t_j))^2 = \frac{1}{mn} \sum_{j} \|\boldsymbol{q}(\boldsymbol{\xi}, t_j) - \boldsymbol{q}_k(\boldsymbol{\xi}, t_j)\|^2, \tag{9}$$

which can be normalised as

$$\frac{1}{m} \sum_{i=1}^{m} \frac{\|\boldsymbol{q}(t_i) - \boldsymbol{q}_k(t_i)\|^2}{\|\boldsymbol{q}(t_i)\|^2}. \tag{10}$$

This is suitable for POD since it offers a physical interpretation if the measured quantity $\boldsymbol{q}$ is velocity: it is then proportional to the loss of turbulent kinetic energy in the system. This is because the time-constant part $\overline{\boldsymbol{q}}$ cancels out of the mean square error, leaving a quantity proportional to the difference in turbulent velocity squared, which is also

proportional to the difference in turbulent kinetic energy, or the loss of turbulent kinetic energy.

Finding the eigenvectors of $\boldsymbol{R} \in \mathbb{R}^{n \times n}$ is often computationally infeasible, since the number of points n usually is very large. To overcome this, a method known as Snapshot POD, developed by Sirovich [11], is used. Instead of solving Eq. (5), the smaller problem

$$\boldsymbol{X}^T \boldsymbol{X} \boldsymbol{\psi}_i = \lambda_i \boldsymbol{\psi}_i \tag{11}$$

is solved. The problem is smaller since $\boldsymbol{X^T X} \in \mathbb{R}^{m \times m}$, and the number of snapshots m is usually much smaller than the number of points. After solving Eq. (11), the original POD modes are recovered [6] as

$$\boldsymbol{\phi}_i = \boldsymbol{X} \boldsymbol{\psi}_i \frac{1}{\sqrt{\lambda_i}}, \quad i \leqslant m. \tag{12}$$

and the rest of the reconstruction is completed using Eq. (6). Note that this method is only able to extract at most m modes.

It is not generally the case that the POD modes correspond to physical patterns that can be observed in the flow. Nevertheless, since POD modes are the eigenvectors of the covariance matrix of the data, they measure how correlated values at points are and capture zones of correlated points, meaning that they could represent a coherent structure of the flow. Two POD modes can sometimes form a pair, where they together describe one dynamic motion in the flow [12]. Since those two modes are orthogonal, the spatial structure and the time coefficient of one of the modes is a shifted version of the other by a quarter of a period. A reconstruction with two modes that form a pair can capture a translating and alternating character of a flow pattern [13]. Pair modes will create a circle when their time coefficients are plotted in a phase portrait, where the value of one coefficient is on the x-axis and the other coefficient on the y-axis. This is because a circle can be parameterised as $(\cos t, \sin t)$ where t covers an interval of length at least 2π. Therefore, if the two time coefficients plot a circle for a given time interval, they are

sine and cosine waves in that interval and are thus offset from each other by a quarter period.

1.3 Modal Decomposition using Autoencoders

Deep learning is a technique within the field of artificial intelligence that uses large amounts of data to learn, as opposed to using human-designed rules. In a deep neural network (DNN), several layers of artificial neural networks are used, each of which interprets its input data differently [14]. An autoencoder (AE) is a DNN that has a structure useful for feature extraction [9]. An AE consists of an encoder, which maps the input data to a low-dimensional space called the latent space, and a decoder, which receives the data in the latent space as its input and maps it back into the original space. The purpose of the AE is to reconstruct the input data while limiting the reconstruction loss. By limiting the dimension of the latent space to some number d, the AE produces a reconstruction that is as good as possible, using only d "features" from the input data. This is the same idea that other techniques for reduced order modelling like POD use: to model the flow using a limited number of features. The elements of the latent space are therefore also modes, although they may not share the same properties as POD modes. AE-based modes are not necessarily orthogonal, nor are they ranked by energy content like POD modes. However, the modes obtained through the AE do not have to form a reconstruction as a linear sum like POD modes, a feature that may increase reconstruction accuracy [9].

An efficient model for the autoencoder is obtained through training the network on a part of the input data, called the training data. A snapshot from the training data is passed through the layers of the encoder and decoder, each layer taking the output of the previous layer as its input. The nodes of a given layer take inputs from the outputs of the nodes of the previous layer, and with each connection between two nodes is associated a weight that may amplify or reduce the strength of that connection. Additionally, there is a bias associated with each node, changing the influence that the node has on the next layer.

When an output reconstruction is produced, the loss is evaluated using some pre-specified loss function, and the weights and biases are adjusted to try to reduce the loss. The algorithm that does the adjustments is called the optimisation algorithm. One of the most widely used optimisation algorithms is Adam [15], used for its computational efficiency, low memory requirement, and straightforward implementation, among other properties [16]. In addition to weights and biases influencing the output of a layer in the network, activation functions can be used to further encourage behaviours in the performance of the network. By using non-linear activation functions, networks are aided in learning higher-order relations between variables [9, 17]. When the network has completed its training, the training data, along with the part that was excluded during the training called the validation data, are passed through the network, and the final performance of the network is measured by the loss in the reconstructions of the data.

1.4 Aim of Study

The present study aims to perform and compare modal decomposition using POD and autoencoders in a turbulent fluid flow by analysing the extracted features and quality of reconstructions of the methods, as well as finding physical interpretations of low-number modes generated by POD.

2 Method

The input data were obtained through Direct Numerical Simulation of the flow of a gas mixture containing N_2O_4 and NO_2 through a channel with a heated bottom wall, used in a study by Zhang et al. [2]. The gas mixture is at chemical equilibrium, with the mass fractions of the species being approximately 73% N_2O_4 and 22% NO_2. There is also 5% non-reacting N_2 present. The gas mixture enters the chamber at a temperature of 303 K, and the bottom heated wall is at the temperature 404 K. The temperature difference makes the gas mixture react in the reversible endothermic-exothermic reaction $N_2O_4 \rightleftharpoons$

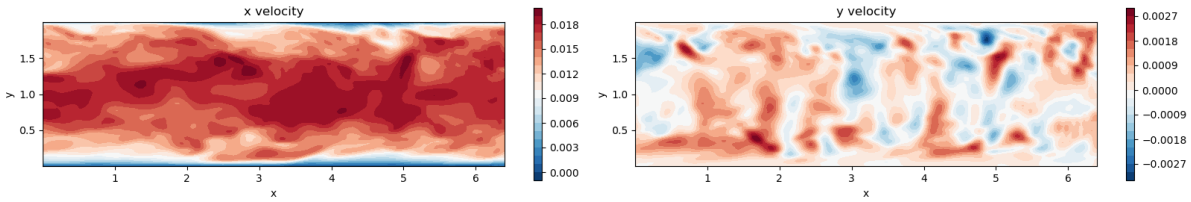


Figure 1: The x and y velocity components for all points at one time instance.

$2\,NO_2$. The heated wall along with the chemical reactions contribute to a turbulent flow. To reduce complexity in computations, a 2D slice of the channel was chosen as the input data, so that the heated wall was along the x-axis. The data were formatted so that it consisted of 886 snapshots, with 65536 points in a 256×256 mesh in each snapshot. The x and y velocities from one snapshot are shown in Figure 1. Snapshot POD was applied on the x and y velocity data to extract modes with time coefficients. The original flow was reconstructed using Eq. (1) for 2, 6, 12, and 20 modes. The autoencoder was run with the dimensions of the latent space $d = 2, 6, 12$, and 20, and reconstructions of the flow were obtained. Relevant hyperparameters for the AE are shown in Table 1, and its structure is shown in Figure 2. In the first and third dense layers, the activation function Rectified Linear Unit (ReLU), defined as

$$R(x) = \begin{cases} x & \text{if } x \geqslant 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{13}$$

is used. The autoencoder was trained to minimise the mean-square error in the reconstruction, and the Adam optimisation algorithm was used to train the network. Values for all three velocity components were obtained from both POD and the autoencoder, but only the x and y components were chosen for analysis since they offered clearer visualisation in the xy-plane used for the input.

Table 1: Hyperparameters used for the autoencoder.

Activation function	Training rate	Training snapshots	Validation snapshots
ReLU	10^{-2}	708	178

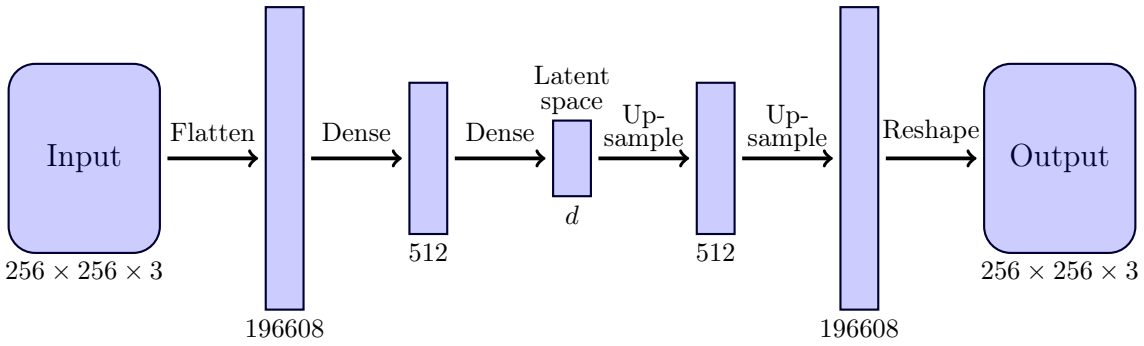


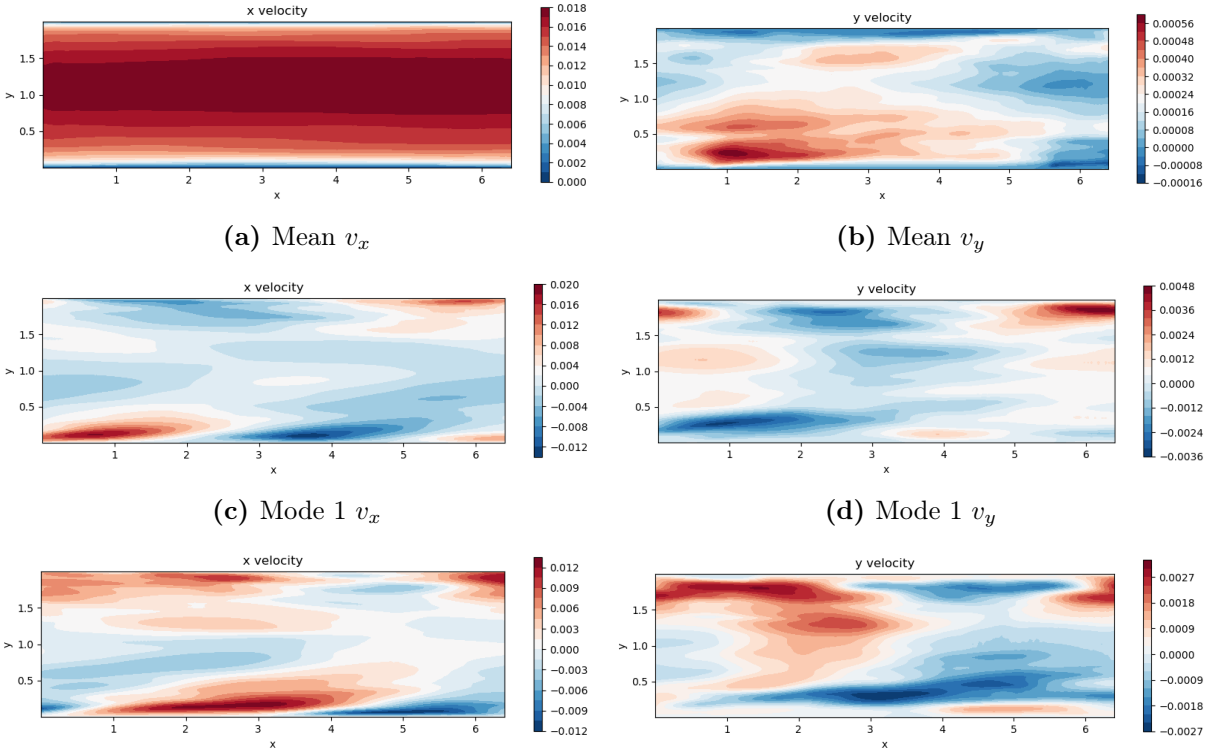
Figure 2: Structure of the autoencoder. First, the encoder flattens the input before it is put through a dense layer reducing the size to 512 using ReLU, followed by another dense layer reducing the size to d: the size of the latent space. The decoder then scales up the data in two analogous layers, using ReLU in the first one, before it is reshaped into the output reconstruction of the flow.

3 Results

The results are split split into two parts. First, the structure of the POD modes with time coefficients are presented. Then, the reconstructions using POD and the autoencoder are displayed and compared.

3.1 Modes obtained using POD

The mean flow and the first two spatial modes for the x and y velocity components are shown in Figure 3. Time coefficients for the first five modes are shown in Figure 4. Observing that the time coefficients for modes 4 and 5 look offset by a quarter period for some time intervals, it could be suggested that these modes form a pair. The phase portrait of the two time coefficients for relevant time intervals are shown in Figure 5. Intervals where the phase portrait formed a circle-like shape were chosen. In addition to the two displayed intervals, the snapshot intervals 20-180 and 708-872 also displayed circular shapes.

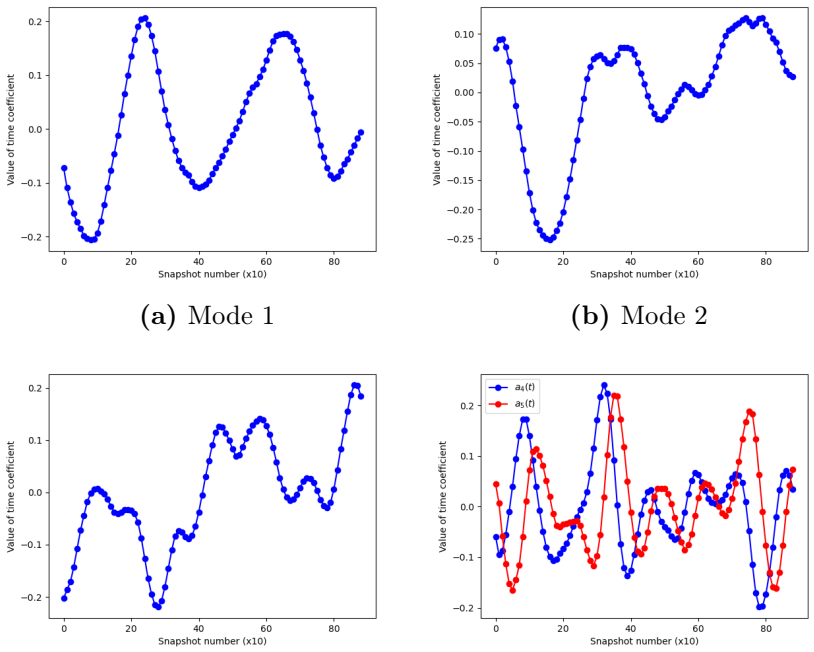


(a) Mean v_x **(b)** Mean v_y

(c) Mode 1 v_x **(d)** Mode 1 v_y

(e) Mode 2 v_x **(f)** Mode 2 v_y

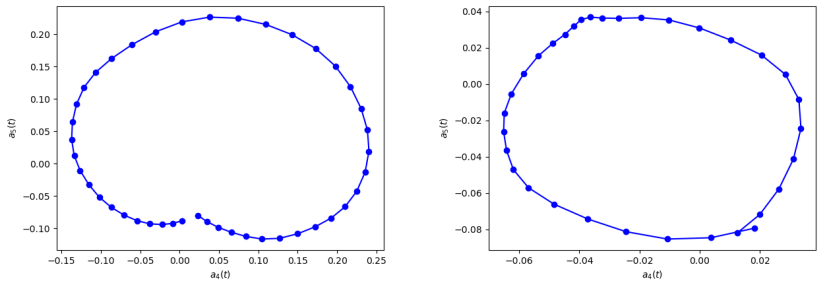
Figure 3: The mean flow and the first two spatial modes extracted from POD for the x and y velocity components.



(a) Mode 1 **(b)** Mode 2

(c) Mode 3 **(d)** Modes 4 and 5

Figure 4: Time coefficients for modes 1 - 5 extracted using POD.



(a) Snapshots 268-440 **(b)** Snapshots 440-572.

Figure 5: Phase portraits of the time coefficients for modes 4 and 5, in the snapshot intervals 268-440 and 440-572. The snapshot intervals 20-180 and 708-872 also displayed circular shapes.

Table 2: Percentage loss of x and y velocity components of the reconstructions from POD and the Autoencoder using 2, 6, 12, and 20 modes.

	2 modes	6 modes	12 modes	20 modes
POD v_x	90.0%	75.3%	63.2%	52.2%
POD v_y	97.5%	90.0%	83.0%	75.2%
AE v_x	26.9%	21.1%	17.6%	17.1%
AE v_y	48.8%	39.8%	33.6%	32.5%

3.2 Reconstructions using POD and the Autoencoder

After reconstructing the flow using both POD and the autoencoder, the loss was calculated using Eq. (10), with results in Table 2. In Figure 7, shown in Appendix B, the mean flow has been removed from the reconstructions to more clearly see to what extent the turbulent features of the flow are captured.

4 Discussion

Generally, a low number of POD modes were not able to capture significant portions of the turbulent kinetic energy of the system. The flow was not structured enough for common patterns seen in many other flows to appear. However, low-number POD modes still contain some information about the structure of the flow close to the heated wall, the main area of interest since it causes much of the turbulent flow. The autoencoder was quite successful at recreating the flow, capturing many of the relevant structures already

when using only 2 modes. POD did not recreate the flow as well, only capturing the larger-scale structures even when using 20 modes.

4.1 Structures in POD modes

There is quite a symmetric mean flow for the x velocity component, whereas the mean flow for the y-component is less structured (Figures 3a and 3b) as there is less physical motivation for the gas moving in the y-direction in a structured manner. There is one clear structure in the mean y-velocity: in the beginning of the channel, the gas that is close to the bottom wall rises. This behaviour makes physical sense as the gas will rise when it is exposed to the hot bottom wall. Additionally, the increased temperature of the gas will favour the endothermic reaction, decomposing more N_2O_4 into $2\,NO_2$, increasing the number of molecules, thus forcing the gas away from the wall.

The first two POD modes both show alternating structures of high and low x-velocity near the heated wall. However, these two modes do not form a pair, since only mode 1 has a clear sinusoidal time coefficient, whereas mode 2 has a more complex and less predictable time coefficient. Mode 1 captures a structure near the heated wall of an area with higher x-velocity followed by an area of lower x-velocity, with the two areas oscillating in value over time. The seemingly steady and repeatedly oscillating nature of this mode suggests that it could be a pattern of the flow with physical interpretability.

No structures of interest were found for mode 3. However, Modes 4 and 5 have quite complementary function, both displaying clear rectangular structures going from the bottom wall to the top wall in y-velocity, and some smaller regions near the wall for x-velocity, see Figure 6 in Appendix A. These structures are offset from each other, suggesting that these modes could form a pair, capturing the same structure of the flow together. This is further supported by the results of Figure 4d, showing that $a_4(t)$ in many time intervals looks like $a_5(t)$, but shifted a quarter period. Further, the phase portrait of these two modes form circular shapes for several time intervals as seen in Figure 5.

It must be noted that the structures observed in low-number modes are not necessarily

representative of the entire structure of the flow since they capture so little of the total turbulent kinetic energy of the system. Due to the weak structure of the flow, POD is not very effective, and many modes are required for an accurate reconstruction. The next section evaluates these reconstructions in comparison to those generated by the autoencoder.

4.2 Comparison between POD and AE reconstructions

The autoencoder performed well in reconstructing the flow. Referring to Table 2, 73.1% of the x-component of turbulent velocity was captured with $d = 2$, a good result in comparison to the reconstruction with 2 POD modes that only captured 10.0% of the turbulent x-velocity. Generally, the y component of the turbulent velocity was not captured by the reconstructions to the same extent as the x component. This could be due to the presence of a stronger mean flow in the x-direction, leaving less turbulent structures than for the y velocity. The AE still outperformed POD in capturing this energy.

However, as the dimension of the latent space increased, the AE model did not improve greatly, especially between using $d = 12$ and $d = 20$. This could be a consequence of the network not training for long enough, thus not finding a way to use the additional dimensions to create a more optimal model. Further increase in training efficiency could lie in optimising the hyperparameters for the network, or to change the schematic structure of the network.

Another flaw in this method lies in the individual AE modes not being isolated and ranked by energy content like the POD modes were. Previous work has extracted modes using AE-based models with success in making them orthogonal and interpretable. Still, there is an ongoing task of constructing AE-based models that extract physically interpretable modes from fluid flows, sharing some of the benefits of POD modes while being able to improve on them and maintaining reconstruction efficiency.

4.3 Conclusion

The data set used for this study did not display clear coherent structures like many other turbulent fluid flows. Because of this, performing modal decomposition with Proper Orthogonal Decomposition was not very effective, as even reconstructions with large amounts of modes were not able to capture the smaller-scale turbulent structures of the flow. However, some weak structures of the flow were identified using POD. Using an autoencoder for feature extraction from the flow yielded more accurate reconstructions, capturing some of the turbulent structures at smaller scales. Even though the autoencoder was very successful in comparison to POD for a low number of modes used for the reconstruction, it had some difficulty improving as more modes were added. This could be improved by increasing training efficiency through optimisation of hyperparameters and the structure of the neural network.

References

[1] Shen Y, Zhang K, Duwig C. Investigation of wet ammonia combustion characteristics using LES with finite-rate chemistry. Fuel. 2022;311:122422. Available from: https://www.sciencedirect.com/science/article/pii/S0016236121022948.

[2] Zhang K, Shen Y, Duwig C. Identification of heat transfer intensification mechanism by reversible N2O4 decomposition using direct numerical simulation. International Journal of Heat and Mass Transfer. 2022;182:121946. Available from: https://www.sciencedirect.com/science/article/pii/S0017931021010516.

[3] Sharma C, Malhotra D, Rathore A. Review of computational fluid dynamics applications in biotechnology processes. Biotechnology progress. 2011;27(6):1497-510.

[4] Anderson JD, Wendt J. Computational fluid dynamics. vol. 206. Springer; 1995.

[5] Moin P, Mahesh K. Direct Numerical Simulation: A Tool in Turbulence Research. Annual Review of Fluid Mechanics. 1998;30(1):539-78. Available from: https://doi.org/10.1146/annurev.fluid.30.1.539.

[6] Taira K, Brunton SL, Dawson STM, Rowley CW, Colonius T, McKeon BJ, et al. Modal Analysis of Fluid Flows: An Overview. AIAA Journal. 2017;55(12):4013-41. Available from: https://doi.org/10.2514/1.J056060.

[7] Walton S, Hassan O, Morgan K. Reduced order modelling for unsteady fluid flow using proper orthogonal decomposition and radial basis functions. Applied Mathematical Modelling. 2013;37(20):8930-45. Available from: https://www.sciencedirect.com/science/article/pii/S0307904X13002771.

[8] Lumley JL. The structure of inhomogeneous turbulent flows. Proceedings of the International Colloquium on the Finite Scale Structure of the Atmosphere and its Influence on Radio Wave Propagation. 1967.

[9] Eivazi H, Le Clainche S, Hoyas S, Vinuesa R. Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. Expert Systems with Applications. 2022;202:117038.

[10] Cordier L, Bergmann M. Proper Orthogonal Decomposition: an overview. In: Lecture series 2002-04, 2003-03 and 2008-01 on post-processing of experimental and numerical data, Von Karman Institute for Fluid Dynamics, 2008. VKI; 2008. p. 46 pages. Available from: https://hal.archives-ouvertes.fr/hal-00417819.

[11] Sirovich L. Turbulence and the dynamics of coherent structures, Parts I-III. Quarterly of applied mathematics. 1987;45(3):561-71.

[12] Rovira M. In pursuit of clean air through numerical simulations of no-waste pollutant removal. Stockholm, Sweden: KTH Royal Institute of Technology; 2022.

[13] Weiss J. A Tutorial on the Proper Orthogonal Decomposition. American Institute of Aeronautics and Astronautics; 2019. p. AIAA 2019-3333.

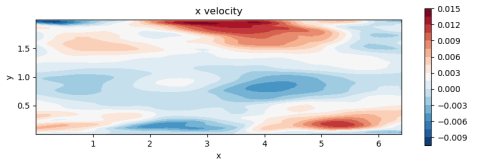
[14] Alzubaidi L, Zhang J, Humaidi AJ, Al-Dujaili A, Duan Y, Al-Shamma O, et al. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of big Data. 2021;8(1):1-74.

[15] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:14126980. 2014.

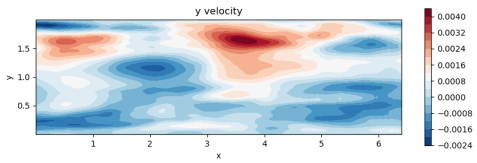
[16] Soydaner D. A Comparison of Optimization Algorithms for Deep Learning. CoRR. 2020;abs/2007.14166.

[17] Nwankpa C, Ijomah W, Gachagan A, Marshall S. Activation functions: Comparison of trends in practice and research for deep learning. arXiv preprint arXiv:181103378. 2018.

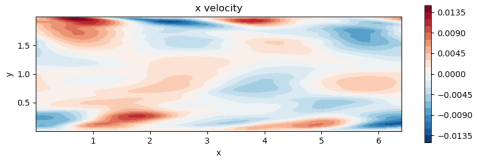
A POD modes 3-5



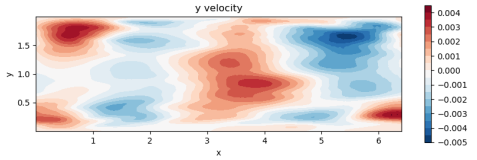
(a) Mode 3 v_x



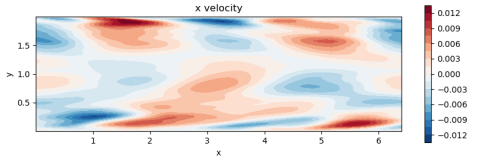
(b) Mode 3 v_y



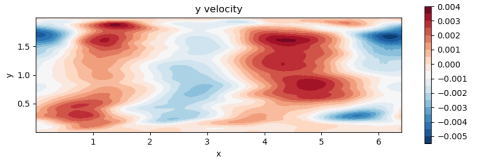
(c) Mode 4 v_x



(d) Mode 4 v_y



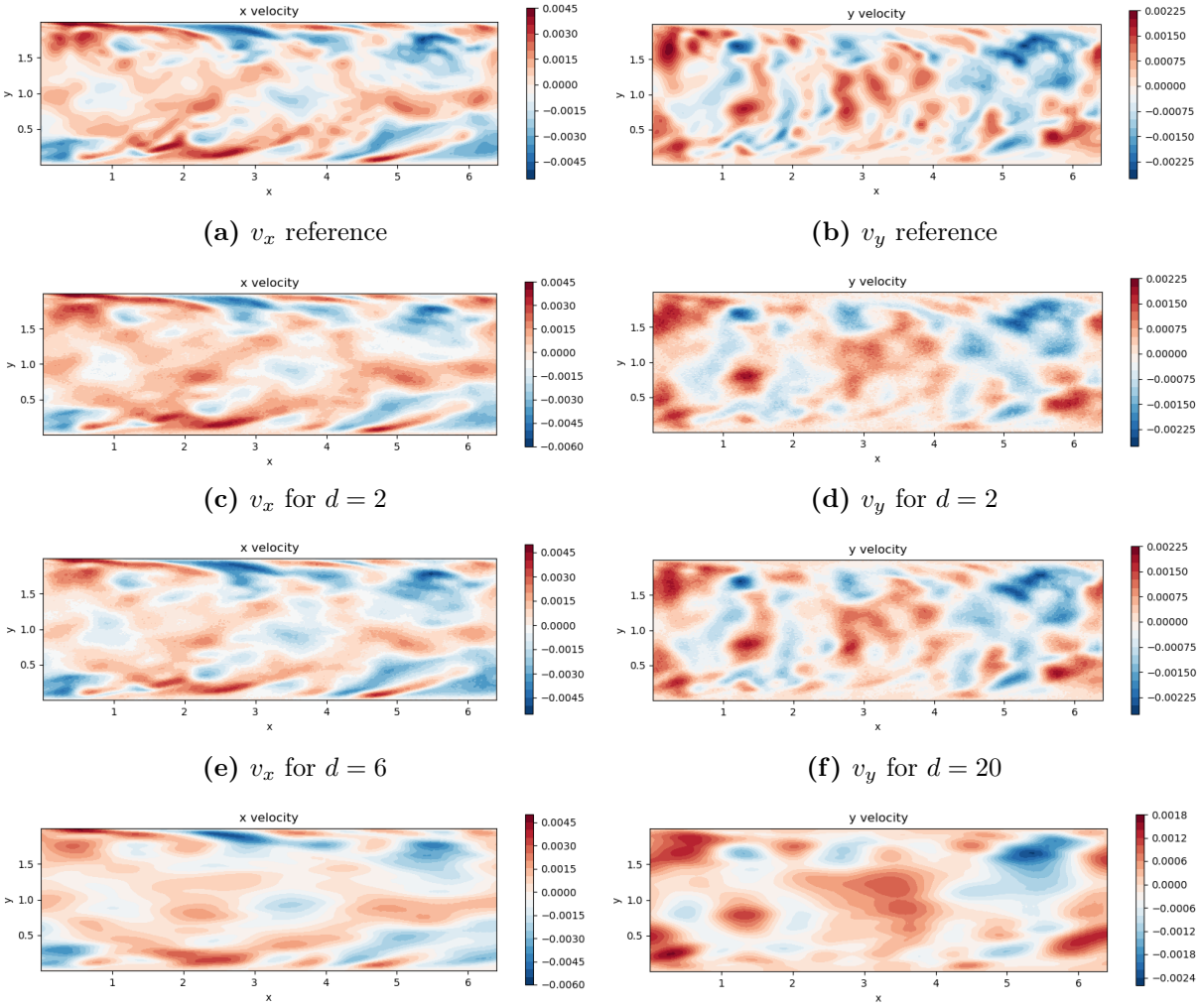
(e) Mode 5 v_x



(f) Mode 5 v_y

Figure 6: x and y velocity components for POD modes 3-5.

B Reconstructions of the flow using the Autoencoder and POD



(a) v_x reference

(b) v_y reference

(c) v_x for $d = 2$

(d) v_y for $d = 2$

(e) v_x for $d = 6$

(f) v_y for $d = 20$

(g) v_x with 20 POD modes

(h) v_y with 20 POD modes

Figure 7: Plots of reconstructions of the turbulent part of the flow using the autoencoder and POD. Figures (a) and (b) show the input data for the x and y velocity components, respectively. Figures (c)-(f) show reconstructions using different values of d, the dimension of the latent space. Figures (g) and (h) show reconstructions using 20 POD modes.