Edinburgh Research Explorer

A novel coupling of noise reduction algorithms for particle flow simulations

OPEN ACCESS

CrossMark
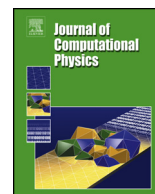
A novel coupling of noise reduction algorithms for particle flow simulations

M.J. Zimoń [a,b,*], J.M. Reese [c], D.R. Emerson [d]

[a] School of Mechanical, Aerospace and Civil Engineering, The University of Manchester, Manchester M13 9PL, UK
[b] James Weir Fluids Lab, Mechanical and Aerospace Engineering Department, The University of Strathclyde, Glasgow G1 1XJ, UK
[c] School of Engineering, The University of Edinburgh, Edinburgh EH9 3JL, UK
[d] Scientific Computing Department, STFC Daresbury Laboratory, Warrington WA4 4AD, UK

ARTICLE INFO

ABSTRACT

Proper orthogonal decomposition (POD) and its extension based on time-windows have been shown to greatly improve the effectiveness of recovering smooth ensemble solutions from noisy particle data. However, to successfully de-noise any molecular system, a large number of measurements still need to be provided. In order to achieve a better efficiency in processing time-dependent fields, we have combined POD with a well-established signal processing technique, wavelet-based thresholding. In this novel hybrid procedure, the wavelet filtering is applied within the POD domain and referred to as WAVinPOD. The algorithm exhibits promising results when applied to both synthetically generated signals and particle data. In this work, the simulations compare the performance of our new approach with standard POD or wavelet analysis in extracting smooth profiles from noisy velocity and density fields. Numerical examples include molecular dynamics and dissipative particle dynamics simulations of unsteady force- and shear-driven liquid flows, as well as phase separation phenomenon. Simulation results confirm that WAVinPOD preserves the dimensionality reduction obtained using POD, while improving its filtering properties through the sparse representation of data in wavelet basis. This paper shows that WAVinPOD outperforms the other estimators for both synthetically generated signals and particle-based measurements, achieving a higher signal-to-noise ratio from a smaller number of samples. The new filtering methodology offers significant computational savings, particularly for multi-scale applications seeking to couple continuum informations with atomistic models. It is the first time that a rigorous analysis has compared de-noising techniques for particle-based fluid simulations.

1. Introduction

Particle-based simulations, e.g. molecular dynamics, are well suited to investigate the effects of fluid–solid interactions and are widely used to study a broad range of complex physical phenomena [1]. For example, molecular dynamics (MD) can be performed to solve classical many-body problems from various fields, including rheology, tribology, and biological systems at the molecular scale. Dissipative particle dynamics (DPD) is another particle-based method that is gaining popularity and

---

* Corresponding author at: School of Mechanical, Aerospace and Civil Engineering, The University of Manchester, Manchester M13 9PL, UK.
  E-mail address: malgorzata.zimon@stfc.ac.uk (M.J. Zimoń).

can be viewed as a coarse-graining of molecular dynamics (a DPD particle is a collection of MD molecules), allowing for mesoscale modelling of complex fluids, e.g. surfactant solutions.

In particle simulations, the system evolves from some initial condition to a final state, preserving constraints of motion. The output of the calculation is a system variable, X, as a function of time. According to the quasi-ergodic hypothesis, the mean value of a sampled quantity, $X(t)$, for an equilibrium system is equal to its ensemble average, $\langle X \rangle$. In general, the mean of a property is often subject to fluctuations making any statistical interpretation of the data challenging. Uncertainty in the measurements is increased through thermal fluctuations introduced by thermostats and sampling with a finite number of particles. These effects are generally referred to as *noise*, and a major challenge in particle simulations is to filter, or *de-noise*, the fluctuations to obtain an accurate ensemble prediction. To produce a *good* approximation, cumulative averaging over large time intervals can be performed, but the computational intensity of the simulations is then substantially increased. Additionally, in unsteady flow simulations, it is difficult to establish the number of time-steps over which the averaging should performed. The development of an efficient filtering technique that provides clean particle distribution functions and smooth gradients, particularly when coupling across different length and time-scales (multi-scale modelling), is highly desirable.

Proper orthogonal decomposition (POD) is a statistical method that identifies a low-dimensional space by separating independent variations from linearly dependent aspects of data. In other words, it extracts correlations from the measurements through a low-rank approximation. The use of POD in interpreting coherent structures in turbulence is now well established [2] and the technique has recently been applied as a noise reduction tool for MD and DPD simulations [3]. Wavelet thresholding, on the other hand, is a non-linear procedure pioneered by Donoho and Johnstone [4] that was proposed from several optimality criteria, such as asymptotic minimax. It allows the analysis of signals at different resolutions and smooths out unwanted variations at a chosen level of detail. Both techniques have had some success but retain certain weaknesses. Classical orthogonal methods require large matrices to extract de-noised information, whereas wavelet-based thresholding is sensitive to the choice of filters used for the wavelet transform (WT). Wavelet thresholding can also over-smooth analysed signals or introduce *Gibbs*-like oscillations [5], and does not reduce the dimensionality as well as POD.

The aim of this work is to develop a method with the capability to improve the efficiency of estimating the unknown structures from particle-based simulation by solving the statistical inverse problem. We will briefly outline the theoretical basis of POD and wavelet transforms, along with their application to noise filtering. For particle-based simulations, we discuss an extension to POD based on *time windows* (WPOD) [3]. By considering their strengths and weaknesses, we propose a new approach, WAVinPOD, which shows good potential for improving the analysis of simulation data. Our method allows for a more efficient noise reduction, obtaining higher average signal-to-noise ratios and smaller errors in L_2 and Frobenius norm than either POD or wavelet thresholding alone. Our paper is organised as follows: the basic theory for the methods is briefly described in Sec. 2, followed by the results of applying WAVinPOD, POD, and wavelet thresholding to synthetic signals. A comparison of the performance of each technique in de-noising particle-based simulations is presented in Sec. 4, followed by some concluding remarks on the new approach.

2. Theoretical background

This section provides a brief mathematical description of POD and wavelet thresholding. Our novel procedure, WAVinPOD, is also introduced. A more extensive discussion on proper orthogonal decomposition can be found in Refs. [2,6,7], and a more detailed review of wavelet theory is given in Refs. [8–10].

2.1. Proper orthogonal decomposition

Proper orthogonal decomposition is often used for finding a low-dimensional approximate description of high-dimensional data that contains a large number of interrelated variables. In addition to order reduction, POD is also applied for feature extraction by revealing coherent structures within the data. The method was introduced to the turbulence community by Lumley [11]. However, the same procedure was developed independently by several groups and is known under different names, including *Principal Component Analysis* and the *Karhunen–Loéve Decomposition*, depending on the area of application.

The basis of POD is to describe a function, $f(t, x)$, as a finite sum of its variables:

$$f(t, x) \approx \sum_{n=1}^{r} \alpha_n(t)\phi_n(x), \tag{1}$$

where t and x represent the temporal and spatial components of the data, respectively. When r (a total number of elements) approaches infinity, the estimate becomes exact. Applying POD establishes a set of orthonormal basis functions (modes), $\phi_n(x)$, such that the first $k < r$ terms provide the best approximation of the function $f(t, x)$. Define an element $A(\tau^s, x)$ of the real $N \times M$ matrix as a measurement from the x-th probe taken at the τ^s-th time instant. An orthogonal decomposition

will determine the optimal approximation of the matrix, A_k ($A_k \approx A$), by first performing singular value decomposition (SVD) of the original real matrix A[1]:

$$A(\tau^s, x) = U \Sigma V^T, \tag{2}$$

where, in case of full SVD, $U \in \mathbb{R}^{N \times N}$, $V \in \mathbb{R}^{M \times M}$ are orthogonal matrices, the superscript T indicates a matrix transpose since $A \in \mathbb{R}$, and Σ is an $N \times M$ diagonal matrix. In Eq. (2), columns U and V are left and right singular vectors, respectively. Singular vectors can be considered as rotations and reflections, and Σ as a stretching matrix. Diagonal entries of Σ, called singular values of A, are non-negative numbers arranged in descending order, $s_1 \geq s_2 \geq \ldots \geq s_r \geq 0$. The number of non-zero singular values defines the rank of the original matrix A ($r = rank(A) = min(N, M)$). Equation (2) can also be expressed in the form

$$A(\tau^s, x) = Q V^T = \sum_{n=1}^{r} q_n v_n^T, \tag{3}$$

where $Q = U \Sigma$, A represents the function $f(t, x)$ from Eq. (1), q_n is a column matrix corresponding to $\alpha_n(t)$, and v_n^T is a transpose of a vector matrix representing $\phi_n(x)$. The description in Eq. (3) is an accurate approximation as the data-set has a finite size. To construct an optimal lower-rank estimate of A, for a determined value $k < r$, the diagonal matrix Σ_k can be obtained by setting $s_{k+1} = s_{k+2} = \ldots = s_r = 0$, resulting in

$$A_k(\tau^s, x) = U \Sigma_k V^T = \sum_{n=1}^{k} s_n u_n v_n. \tag{4}$$

Throughout this work, the rank of the best matrix approximation, k, will be referred to as the number of *dominant* modes.

2.1.1. Noise filtration with POD/WPOD

Proper orthogonal decomposition may also be considered as an *energy decomposition* which has the capability to filter out low energy spectra (i.e. noise) from raw data. If the previously considered matrix, A, is now a collection of M noisy measurements at N instants in time, it can be represented as a composition of the form $A = A_{\text{true}} + B$, where A_{true} is an $N \times M$ matrix that contains the *true* signal, and B is a matrix that denotes the unwanted noise. In general, we only know the original matrix A and we need to remove the noise to extract the information contained in A_{true}. For the special case of a synthetically generated matrix A, we will know A_{true} and corrupt the signal with artificially generated noise, represented by matrix B. Using POD, we can remove the noise from matrix A by creating a corresponding approximation matrix of rank k, A_k, that contains all the correlations from the original data, i.e. $A_k \approx A_{\text{true}}$. The approximation is obtained by truncating the singular values as described in Eq. (4); such a procedure is referred to as truncated SVD. The key property of POD is that A_k is optimal in the sense that $min\{\|A_{\text{true}} - A_k\|_F^2\} = \sum_{n=k+1}^{min(N,M)} s_n^2$, where $\|.\|_F$ is the Frobenius norm.

In the case of simulation data, Grinberg [3] developed an extension to POD for particle measurements that is based on time-windows and generally referred to as WPOD; in the approach, the data is defined as $A = N_{\text{POD}} N_{ts} \Delta t$, where N_{POD} is the number of time averages used, N_{ts} defines how many observations are in one average, and Δt is the simulation time-step. In this work, WPOD is utilised to filter out noise in molecular simulations based on the approach presented by Grinberg (note, in Grinberg's paper the matrix A is defined as T_{POD}). For a set of noisy observations (snapshots) $A(\tau^s, x)$, defined as a field at positions in space $x \in \mathbb{R}^d$, $d = 1, 2, 3$ and at discrete times τ^s, $s = 1, 2, \ldots, N_{\text{POD}}$, WPOD calculates a set of orthogonal basis modes by applying SVD to the POD window, SVD(A). Throughout this work, the singular vectors are referred to as temporal or spatial POD modes depending on whether they hold information either about the shape of the signal or its time nature. In the case of the matrix $A(\tau^s, x)$, a temporal POD mode (related to temporal coefficient $\alpha_k(t)$ from Eq. (1)) corresponding to mode number k is a left singular vector, u_k (column k of matrix U), and a spatial POD mode is a right singular vector, v_k.

To obtain the best estimate of A_{true}, the number of singular values k needs to be carefully determined. The main rationale of using SVD (or EVD) to filter out noise is based on the assumption that, unlike unwanted fluctuations, important (coherent) structures are energetic. One natural criterion for choosing k is to select the cumulative percentage of total energy (e.g. 90%) that the selected modes contribute. The energy threshold is often chosen arbitrarily, and it can depend on some practical details of a considered data-set. In the case of particle simulations, defining a cut-off based on energy poses difficulties, as it often happens that structures of interest contain very little energy when compared to dominant features. For that reason, some additional criteria need to be met in order to establish an appropriate subset of significant modes. In the present work, apart from studying the energy content of eigenvalues, most of the data is analysed with the log-eigenvalue diagram (LEV) [12] and by investigating the smoothness of temporal modes [3]. In some cases, the choice of k is verified with a recently proposed singular value hard threshold (SVHT) [13].

---

[1] Proper orthogonal decomposition can also be done by eigenvalue decomposition (EVD) of the symmetric matrix $C = A A^\dagger$ ($A^\dagger A$ if $N > M$).
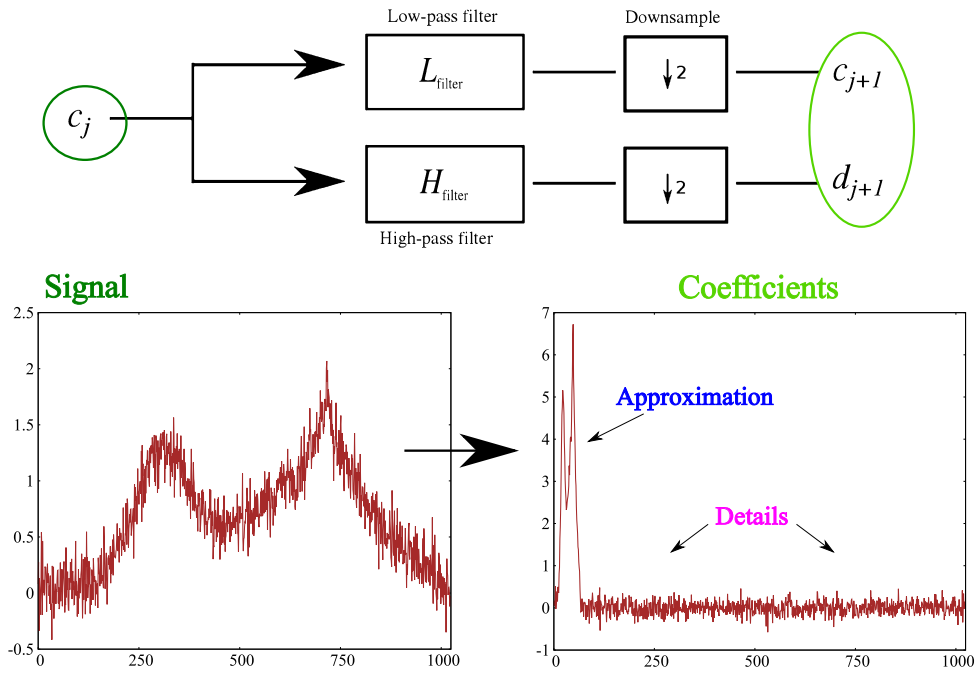
Fig. 1. One step of DWT with filter banks.

2.2. Multiresolution and fast wavelet transform

Multiresolution theory developed by Mallat [14] and Meyer [15] introduces the orthogonal discrete wavelet transform (DWT), where a signal is analysed at scales varying by a factor of 2. The transform utilises a shifted and scaled *mother wavelet*, ψ, forming a basis (*children*) defined as

$$\psi_{j,m}(t) = \frac{1}{\sqrt{2^j}} \psi \left(\frac{t - m2^j}{2^j} \right), \tag{5}$$

where 2^j is a discrete dilation parameter, an integer $j \in \mathbb{Z}$ represents a scale resolution, and $m2^j$ is a discrete shift ($m \in \mathbb{Z}$). The mother wavelet satisfies the *admissibility condition*, $C_\psi = \int_0^{+\infty} \frac{|\widehat{\psi}(\omega)|^2}{\omega} d\omega < +\infty$, where $\widehat{\psi}(\omega) = \int_{-\infty}^{+\infty} \psi(t)e^{-i\omega t} dt$. Wavelet function, together with a dilated and translated scaling function ϕ_s (*father*), $\phi_{s_{j,m}}(t) = \frac{1}{\sqrt{2^j}} \phi_s \left(\frac{t-m2^j}{2^j} \right)$, decomposes the signal into coefficients given as follows:

$$f(t) = \sum_{m \in \mathbb{Z}} c_{j_0,m} \phi_{s_{j_0,m}}(t) + \sum_{j,m \in \mathbb{Z}} d_{j,m} \psi_{j,m}(t). \tag{6}$$

The coefficients in Eq. (6) are obtained by integrating the product of the functions with the signal; $c_{j_0,m} = \left\langle f, \phi_{s_{j_0,m}} \right\rangle$ is a smoothed (coarse) approximation, $d_{j,m} = \left\langle f, \psi_{j,m} \right\rangle$ are the fine scale details at different resolutions $-J \le j < 0$ and $0 \le m < 2^{-j}$, where J is the maximum number of decompositions and $j \le j_0$. In other words, the details, $d_{j,m}$, and the approximation, $c_{j_0,m}$, are projections of the signal onto certain complimenting subspaces.

Mallat [9] and Daubechies [16] established a link between filter banks in signal processing and wavelets, allowing for a fast decomposition. The fast wavelet transform algorithm does not make use of the wavelet and scaling function, but of the quadrature mirror filters (QMFs)[2] that describe their interaction. In the fast transform, the signal is convolved with both a high-pass filter (H_{filter}, determining the wavelet function), which produces the details of the decomposition, and a low-pass filter, (L_{filter}, associated with the scaling function) which gives the approximation of the signal. The process is shown in Fig. 1. Given a signal $f \in \mathbb{R}^M$, the fast wavelet transform can consist of maximum $J = \log_2 M$ levels (e.g. for $M = 512$ only $J = 9$ stages of decomposition are possible). At each stage, the two sets of coefficients are produced by convolving the signal with the low and high-pass filters followed by dyadic decimation. The signal now has half of the number of samples which means that the scale is doubled. Details, d_{j+1}, are stored while the smoothed image (approximation, c_{j+1}) of the signal

---

[2] QMFs in digital signal processing are odd index alternated reversed versions of each other.

is again convolved with the QMFs resulting in a further set of coefficients. The process is repeated until a desired level of decomposition is reached. This algorithm progressively drains the signal of its information. The approach can be perceived as a mathematical microscope allowing us to *see* the signal at different dyadic magnifications, offering a powerful way of decomposing data into its elementary constituents across scales. Inverting the decomposition with an inverse wavelet transform consists of inserting zeros (upsampling) between samples and convolving the results with the reconstruction filters. If the signal length is not a power of 2, then extending the samples is needed. In the present work, the symmetric boundary value replication is applied [17]. The algorithm can be naturally extended to encode two-dimensional signals (e.g. images) [9]. This kind of two-dimensional transform leads to a decomposition of the signal into four components: approximation and details in three orientations (horizontal, vertical, and diagonal).

2.2.1. Signal de-noising with wavelets

There exist many variants of wavelet-based de-noising [9]. The procedures consist of wavelet decomposition, thresholding of the detail coefficients, and applying the inverse transform to reconstruct the signal. Donoho and Johnstone [4] showed that de-noising can be performed with hard thresholding, defined as $T_H(d_{j,m}) = d_{j,m} \mathbb{1}_{\{|d_{j,m}| > T_u\}}$, as well as with soft thresholding (wavelet shrinkage), $T_S(d_{j,m}) = \max\left(0, 1 - \frac{T_u}{|d_{j,m}|}\right) d_{j,m}$, where $\mathbb{1}$ is an indicator function and T_u is a threshold value. Here, we focus only on the second approach with a universal threshold:

$$T_u = \sigma_n \sqrt{2 \log(M)}. \tag{7}$$

The white noise level estimate is defined as $\sigma_n = MAD/0.6745$, with *MAD* being the median absolute deviation[3] of the fine scale wavelet coefficients [18]. Soft thresholding has the ability to efficiently smooth the signal but with loss of some characteristics, e.g. peak heights, over-smoothing the edges. The hard threshold method generally reproduces sharpness and discontinuities of the signal better, but at the cost of visual smoothness (can generate *Gibbs*-like oscillations) [18]. As most of the simulation results are corrupted by correlated noise, a level-dependent variance estimation introduced by Johnstone and Silverman [19] can be employed.

In the present work, for simplicity, only filters associated with Daubechies family of orthogonal wavelets were utilised [16], particularly *db*3 and *db*4 as they provided a good compromise in terms of signal-to-noise ratio and smoothness of data reconstruction; the numbers, 3 and 4, define how many vanishing moments are used. Most applications, including noise reduction, require that a function is approximated with few non-zero wavelet coefficients, i.e. has sparse representation. This depends mostly on the regularity of the signal, the number of vanishing moments of ψ and the size of its support [9]. Daubechies wavelet family is the most popular due to its orthogonal and compact support abilities. However, other wavelets, such as the Coiflets, Symmlets of Daubechies can be used for efficient data processing offering different beneficial properties. For example, the linear-phase biorthogonal filters can perform symmetric transforms solving the problem of edge discontinuities.

2.3. Wavelet thresholding within POD

In order to achieve a better efficiency of POD in processing non-stationary fields, we combined the method with wavelet-based filtering with fixed threshold (see Eq. (7)). In this new procedure, wavelet thresholding is applied within the POD domain as shown in Fig. 2. A significant benefit of this approach is that applying wavelet filtering in the SVD domain preserves the dimensionality reduction. Wavelet de-noising is used to eliminate remaining high frequencies from the dominant modes which would require larger amounts of data for POD or WPOD alone. In the following sections, it is shown that WAVinPOD outperforms the other estimators in de-noising synthetic and particle measurements.

Combining the wavelet transform and SVD (or EVD) has already been proposed in literature. Most of the procedures involve using SVD (or EVD) for noise level estimation, transforming a signal to the wavelet domain, and performing SVD (or EVD) on the chosen coefficients, or the whole matrix, as in Bakshi's Multiscale PCA [20] or its extension – multivariate wavelet de-noising developed by Aminghafari et al. [21]. However, these methods appear to be computationally expensive, considering the number of operations performed, making them unsuitable for de-noising particle-based simulations.

In this paper, unless stated, the left singular vectors arising in the WAVinPOD analysis are left unchanged. Applying the same filtering to u_n and v_n in cases where $N \ll M$ may not offer much improvement in noise reduction. This is due to the fact that processing short functions weakens their orthogonality property, resulting in unwanted aliases. However, when $N \sim M$, processing both sets of vectors can add to the performance as discussed in the following section.

More precisely, the general procedure for WAVinPOD de-noising is as follows:

- *Step 1*: Perform SVD on matrix data *A*.
- *Step 2*: Define the number k and set $s_n = 0$ for $n > k$.
- *Step 3*: Perform a wavelet transform of the k modes corresponding to the most energetic singular values.

---

[3] Median of absolute value can also be applied.

- *Step 4*: Apply wavelet de-noising (soft or hard) to detail coefficients and reconstruct the modes with the inverse wavelet transform.
- *Step 5*: Multiply the matrices according to Eq. (4) to obtain data approximation.

When the method is applied during a simulation run, the moving window is used in the same manner as in WPOD (see Sec. 2.1.1).

3. Analysis of synthetic data corrupted with noise

This section presents the results of applying POD, wavelet thresholding and WAVinPOD to synthetically generated data. These studies allow a straightforward comparison of the effectiveness of the methods in analysing various signals. All data processing was performed using the commercial software package MATLAB R2014b (the MathWorks Inc., Natick, MA, 2014). Three objective measures, averaged signal-to-noise ratio (SNR), relative error in the L_2 matrix norm, δ_2, and error in the matrix Frobenius norm, δ_F, were applied to a range of accepted test problems. For each observation, SNR was calculated as a ratio of the summed squared magnitude of the true signal to that of the noise, and expressed in the logarithmic decibel scale. The relative errors in the L_2 and Frobenius norms are given as

$$\delta_2 = \frac{\|A_{\text{true}} - A_k\|_2}{\|A_{\text{true}}\|_2}, \quad \delta_F = \frac{\|A_{\text{true}} - A_k\|_F}{\|A_{\text{true}}\|_F}, \tag{8}$$

where A_k is a de-noised matrix obtained with one of the methods. The L_2 norm of matrix A is defined as the maximum singular value of A, $\|A\|_2 = s_1$. The error in L_2, i.e. δ_2, compares the energy content of A_k with the original matrix. The Frobenius norm takes all entries of the difference, $A_{\text{true}} - A_k$, as a single vector and measures its length. It then indicates which output has the shortest *length* of errors [22].

Two different oscillating signals have been proposed that imitate spatially variable functions arising in various scientific problems. The first initially smooth $N \times M$ data matrix was generated as follows:

$$A^1_{\text{true}}(t, x) = \sin\left(\frac{\pi x}{M}\right) \cos\left(\frac{\pi t}{N}\right) + 0.8 e^{(-3\pi x/M)} \sin\left(\frac{9\pi x}{M}\right), \tag{9}$$

looping over $t = 1 : N$ and $x = 1 : M$. The second set of signals, A^2_{true}, was constructed with the following MATLAB code:

```
s=0:0.01:(M*0.01-0.01);
y(1,s/0.01)=3*sin(s) +sin(0.5*s + 40) + 2*sin(3*s - 60);
    for t=1:N
        for x=1:M
            A_true(t,x)=y(1,x)*cos(pi*t/N)+
            +sin(0.5*t+40)*cos(pi*x/M)+
            +(0.01*t+2+sin(t)).^2;
        end
    end
```

In both cases the signals were of length $M = 1000$ and initially only $N = 20$ observations were used. The number of time-samples required to de-noise a signal is of significance as it defines for how long the simulation has to run to provide enough statistics. The ranks of smooth matrices, equal to $k_1 = 2$ and $k_2 = 3$ for A^1_{true} and A^2_{true}, respectively, were increased by corrupting each signal with added white noise.[4] The resulting noisy data-sets, A_1 and A_2, were full-rank because of the partial de-correlation of the disturbed data points. In a real situation we will not know the original signal, but only corrupted measurements, and often we are not sure of their nature. For the analysis with POD and WAVinPOD we have to rely solely on examination of the eigenspectrum in order to establish an adequate number of k for the approximation. The previously listed criteria were utilised here to find the number of significant modes.

The main drawback of using wavelet transform and multiresolution analysis is the amount of parameters that need to be considered *a priori*, e.g. mother wavelet, number of vanishing moments and levels of decomposition. The choice of an appropriate model is often problematic and may lead to data misinterpretation if any deviations from it appear. In this case, we know that the signals to be recovered are smooth, therefore wavelet shrinkage should be used instead of hard thresholding. The filters associated with Daubechies wavelet *db*3 appeared here to give a good representation of polynomial behaviour within the signals. When applying wavelet thresholding to spatial modes obtained with SVD, the choice of wavelet basis is less critical than applying the transform directly to the noisy data. It is due to the fact, that WAVinPOD filters only components of the signal that have already been partially de-noised. For clear comparison, the same wavelet parameters are used for wavelet thresholding applied to raw data (WAV) and WAVinPOD.

At first the original matrix A^1_{true} was corrupted with noise of standard deviation $\sigma_n = 0.1$, producing A_1 with SNR ≈ 12.4 dB (see Fig. 4(a)). After applying SVD to the whole matrix, the criteria for determining k were utilised. Tests mentioned

---

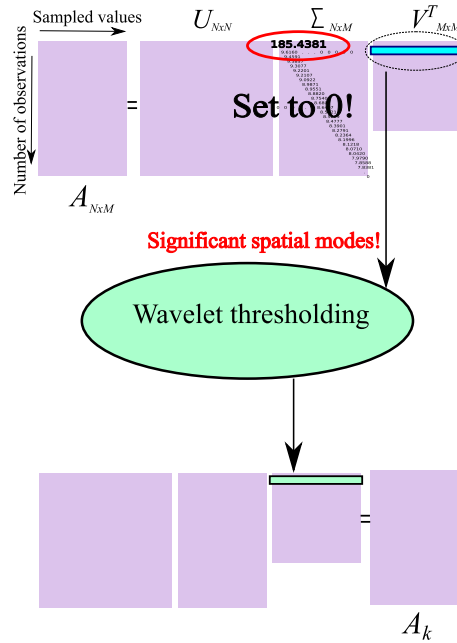[4] Zero-mean Gaussian noise was simply generated using a pseudo-random number generator.

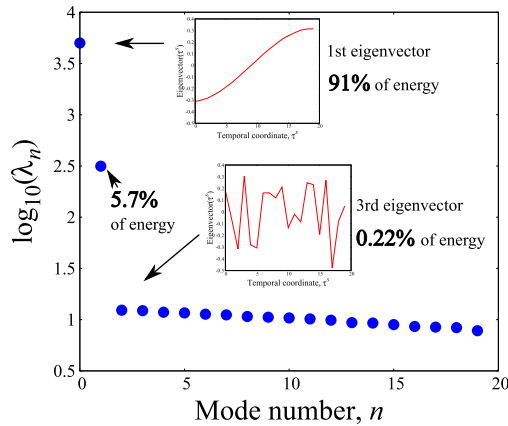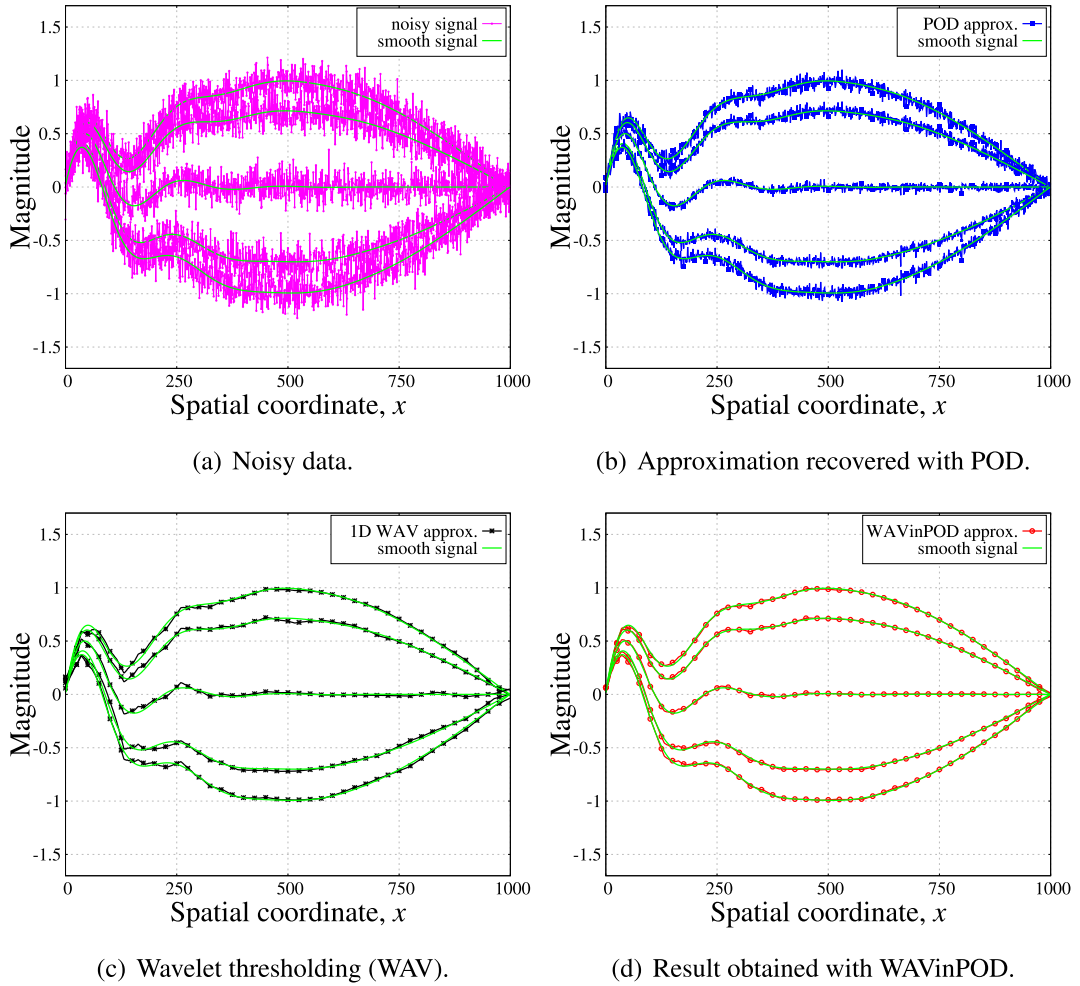**Fig. 2.** Graphical representation of WAVinPOD algorithm.

Fig. 3. Investigating criteria for the choice of significant k; LEV diagram with the 1st and 3rd eigenvector and relative energy of the first three eigenvalues.

in Sec. 2.1.1 managed to recover an accurate number of significant modes. The first and second eigenvalue, $\lambda_{n=1} = s_1^2$ and $\lambda_{n=2} = s_2^2$, were the most energetic containing together 96.7% of total variance ($E_\lambda^1 = 91\%$ and $E_\lambda^2 = 5.7\%$). In practice, it is common to select levels of energy threshold between 70% to 95% [23]. It is evident that the first two modes retained most of the significant information; the third eigenvalue corresponded to only 0.22% of the total energy. When the LEV diagram, $\log_{10}(\lambda_n)$, was plotted, the first two eigenvalues also appeared to be fast-decaying (see Fig. 3), while the other points formed almost a straight line. Choosing the truncation based on LEV is motivated by the idea that, if higher modes represent uncorrelated noise, then the corresponding eigenvalues should decay exponentially with increasing mode number [24]. The eigenvalues that are distinguishable from almost a straight line on the LEV diagram represent valuable information. Corresponding eigenvectors were smoother than other temporal modes which clearly contained high-frequency oscillations as shown in Fig. 3. The choice of k was further confirmed by applying the SVHT [13], which suggested a threshold of $th = 4.5982$ for known noise, and similar $th = 4.7231$ for unknown variance and $\beta = N/M = 20/1000 = 0.02$; the third singular value was smaller than the threshold, $s_3 = 3.5179 < th$ resulting in only two modes being recovered. We also propose using the σ_n estimated from the finest wavelet coefficients (see Eq. (7)) in cases where noise level is not known *a priori*. In this approach, a slightly smaller threshold was obtained, $th = 4.4097$, but sufficient to retain the same number $k = 2$. It is important to consider all the criteria because any single test on its own may not provide enough information to capture the significant phenomena. To achieve higher confidence in selecting an appropriate k it is best to analyse POD (or WPOD for simulation data) results with at least two tests. For wavelet transform, 6 levels of decomposition were performed.

(a) Noisy data.

(b) Approximation recovered with POD.

(c) Wavelet thresholding (WAV).

(d) Result obtained with WAVinPOD.

Fig. 4. Reconstruction of synthetic signals, initially corrupted with Gaussian noise (SNR = 12.4134 dB), with POD, 1D WAV and WAVinPOD.

**Table 1**
Comparison of de-noising efficiency.

| Method	SNR [dB]	δ_2	δ_F	t [s]
Signal with Gaussian noise SNR = 12.4134 dB; [$N = 20$, $M = 1000$]				
POD	22.7478	0.0453	0.0617	0.02
1D WAV	25.2223	0.0388	0.0452	0.13
WAVinPOD	33.3264	0.0147	0.0178	0.04
Signal with Gaussian noise SNR = 12.3998 dB; [$N = 240$, $M = 1000$]				
POD	32.5555	0.0130	0.0198	0.08
2D WAV	34.2217	0.0120	0.0167	0.13
WAVinPOD	37.7215	0.0077	0.0114	0.1

The enhanced signals obtained with each technique are plotted every 5th observation in Figs. 4(b)–4(d). Table 1 summarises values of averaged SNRs (in dB) and errors in L_2 and Frobenius norm for each reconstruction along with the time (in seconds) it took to perform every operation. For both WAVinPOD and POD, the time required to determine which EOFs were significant was not included in the final estimation of computational cost. In other words, the processing times presented here depict how long it took to perform the whole algorithm with pre-defined k. The number of time samples required to de-noise the signal is of significance as it defines for how long the simulation has to run to provide enough data. As expected, POD did not recover the signal well for a small number of observations; the reconstructed matrix had SNR = 22.7478 dB, $\delta_2 = 0.0453$ and $\delta_F = 0.0617$. Better result was obtained with 1D wavelet shrinkage, over 10% higher SNR than POD, 14% and 27% lower δ_2 and δ_F, respectively. Applying WAVinPOD was the most successful in extracting the smooth signal with the highest SNR = 33.3264 dB (about 168% higher than the original noisy data; over 46% and 32% bet-
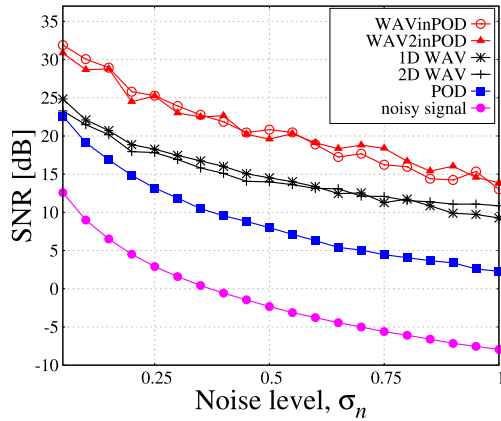
(a) Averaged signal-to-noise ratios for each reconstruction.
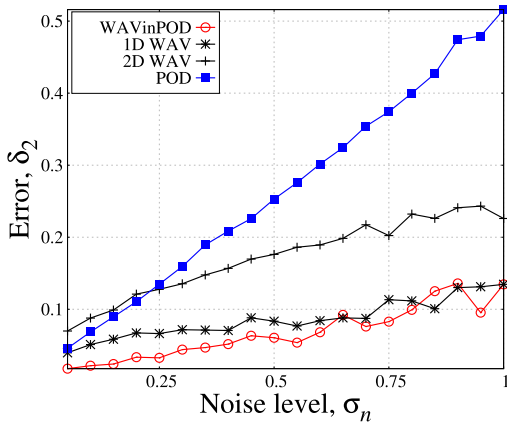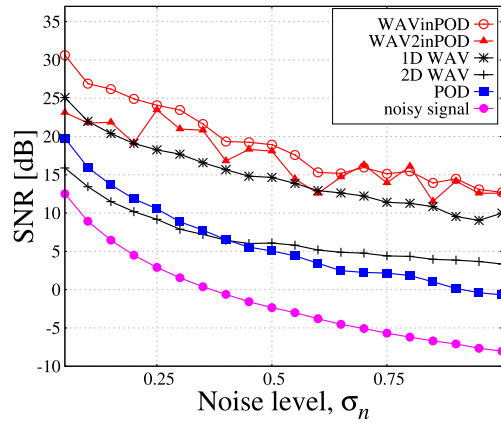

(b) Errors in L_2 norm.
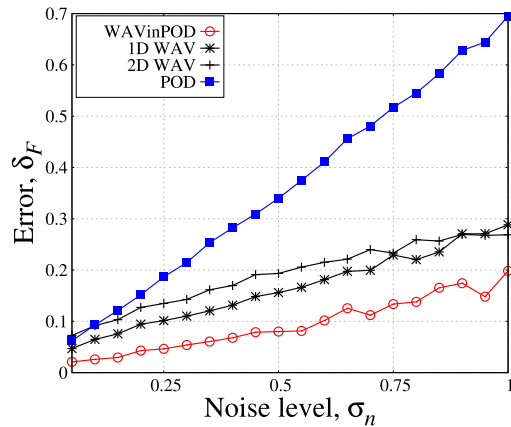
(c) Errors in Frobenius norm.

Fig. 5. Performance of WAVinPOD, compared with WAV2inPOD, POD and 1D wavelet thresholding for increasing noise variance measured in SNR (a). In (b) and (c) only WAVinPOD, POD and 1D WAV are considered for clarity. Analysed matrix A_1 is a set of $N = 20$ oscillating signals of length $M = 1000$. For WAVinPOD and WAV2inPOD, 2 modes were subject to soft wavelet thresholding with the *db*3 filter and 6 levels of decomposition.

ter than POD and wavelet shrinkage, respectively) and the smallest $\delta_2 = 0.0147$ and $\delta_F = 0.0178$, representing a reduction in errors of 60% for WAV and 70% for POD. The one-dimensional wavelet transform method appeared to be the slowest, about $t = 0.13$ s. Performing the 2D transform is faster (in this case, only $t = 0.03$ s), but for such a small N the noise reduction would be less effective: SNR $= 21.6070$ dB, $\delta_2 = 0.0893$ and $\delta_F = 0.0906$. In general, if only a small number of long data arrays is available, $N \ll M$, it is more beneficial to de-noise each signal separate, as there are insufficient observations to look for correlations between them. Proper orthogonal decomposition was faster than WAVinPOD but it produced the worst approximation. A similar discrepancy was observed with the smaller number of time samples; for $N = 10$ the signal-to-noise ratio for WAVinPOD was SNR $= 30.5314$ dB, while wavelet shrinkage with 1D transform and POD provided SNR $= 24.7175$ dB and SNR $= 19.6530$ dB, respectively.

Another analysis was performed in order to determine how many time-samples of the signal with similar noise level, SNR $= 12.4566$ dB, would be required for POD to achieve a comparable de-noising performance as WAVinPOD with $N = 20$. In this case, around 240 samples (12 times more than for WAVinPOD) allowed POD reach SNR $= 32.5555$ dB. The two-dimensional de-noising with wavelet shrinkage alone recovered better SNR $= 34.2217$ dB, while WAVinPOD again outperformed both techniques achieving the highest SNR $= 37.7215$ dB and lowest errors in norms, $\delta_2 = 0.0077$ and $\delta_F = 0.0114$. When a larger number of observations are available, i.e. the temporal modes are longer, applying wavelet thresholding to dominant left and right singular vectors (referred to as WAV2inPOD) may be beneficial, because the additional transformation does not strongly affect the orthogonality of modes. For $N = 240$, WAV2inPOD recovered the best approximation with SNR $= 39.8888$ dB. However, as our goal is to decrease the number of measurements required for data extraction, WAVinPOD is still the preferred de-noising approach.

Fig. 5 illustrates the performance of each method, measured with SNR (see Fig. 5(a)) and errors (see Figs. 5(b) and 5(c)) of the approximations, for increasing noise level added to the matrix A_{true}^1 with $N = 20$. The main observation is that WAVinPOD filters the signals much better than POD and wavelet thresholding for any give noise variance. In addition, we

(a) Averaged signal-to-noise ratios for each reconstruction.
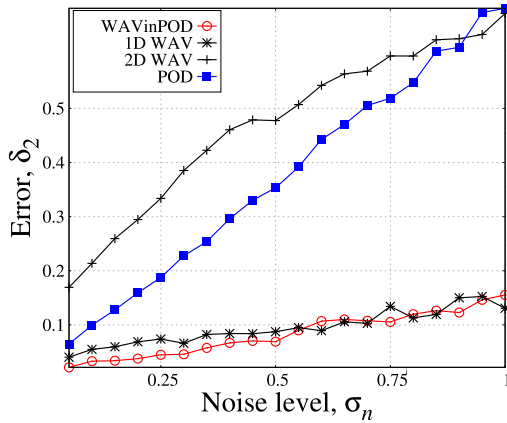

(b) Errors in L_2 norm.
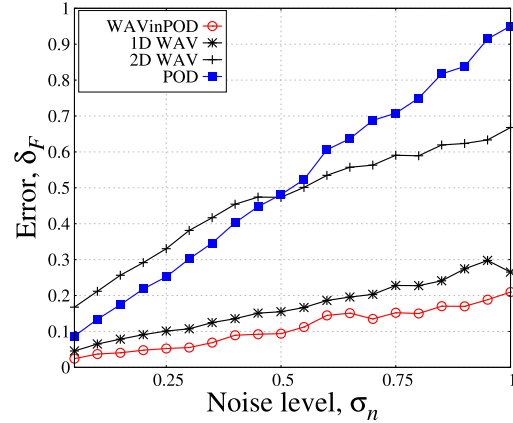
(c) Errors in Frobenius norm.

Fig. 6. Comparison of all the methods applied to the matrix A_1 with $N = 10$ oscillating signals of length $M = 1000$. For WAVinPOD and WAV2inPOD, 2 modes were subject to soft wavelet thresholding with the $db3$ filter and 6 levels of decomposition.

show that WAV2inPOD does not provide a higher SNR gain than the original procedure for *thin* data-sets, i.e. $N << M$. It is therefore preferable to use the less computationally expensive option, WAVinPOD. The same conclusions were drawn for a smaller number of signals, $N = 10$, where all the methods performed noticeably worse than WAVinPOD, as shown in Figs. 6(a)–6(c). For clarity, the errors in norms for WAV2inPOD are not included in the plots. When the number of time-samples is increased, WAVinPOD and POD results converge; this is presented in Fig. 7, where the error in L_2 norm of de-noised approximations is plotted against the increasing size of A_1. In contrast, it can be observed that processing both sets of dominant modes, i.e. performing WAV2inPOD, can offer further improvement in data quality for larger N.

As mentioned before, POD has the potential to successfully extract a clean signal and outperforms the ensemble averaging that is widely applied in particle simulations. Nevertheless, the method still requires a relatively large number of samples in order to de-noise the data. Combining POD with wavelet thresholding improves the computational efficiency of the noise reduction process. For a small number of observations, POD manages to retain the mean shape of the signal but dominant modes still contain some energy corresponding to unwanted fluctuations. In WAVinPOD, wavelet threshold-ing filters the dominant spatial modes from all the disturbances without modifying the signal's profile (the most energetic structure). This can be observed by performing the wavelet transform on the original clean signal and its reconstruction. Fig. 8(a) shows a sparse representation of the smooth signal at 6 levels of resolution in the wavelet domain. The peaks at the beginning of the plot (on the left) are the approximation coefficients, $c_{j_0,m}$, which contain the coarse information. Moving from left-to-right shows decomposition of the signal from coarse-to-fine resolution. The fine detail coefficients, $d_{j,m}$, are mostly zero as expected for an undisturbed smooth data-set. In Fig. 8(b) we show the POD approximation of A^1_{true} transformed into wavelet coefficients. The coarse coefficients are similar to the ones observed in Fig. 8(a). However, at fine resolution it can be seen that the signal still contains some noise. The line in Fig. 8(c) represents the WT of the same signal de-noised with WAVinPOD. For the problem considered, it was concluded that the WAVinPOD performed better than POD in removing noise from the data.
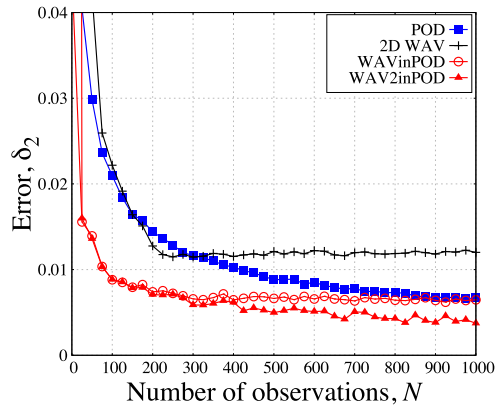
Fig. 7. Comparison of all the methods applied to the matrix A_1 with increasing number of oscillating signals. For WAVinPOD and WAV2inPOD, 2 modes were subject to soft wavelet thresholding with the $db3$ filter and 6 levels of decomposition. Note the convergence of WAVinPOD and POD for $N = M = 1000$; WAv2inPOD produced the lowest δ_2 for $N > 300$.
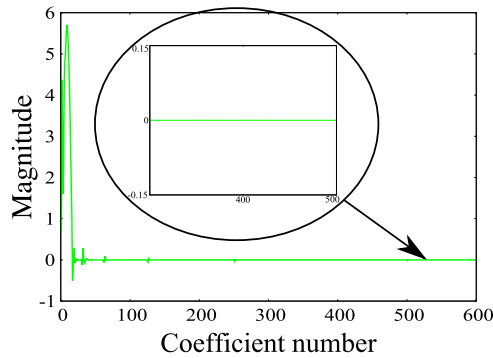
(a) Wavelet transform of clean signal.
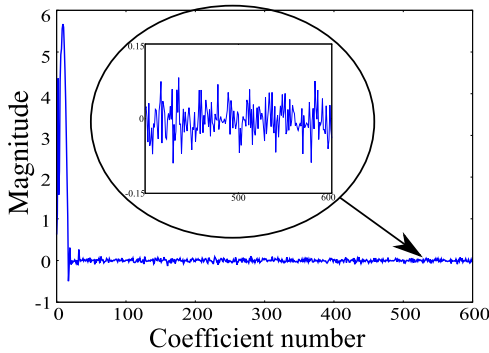
(b) Coefficients of $A_k \approx A_{\mathrm{true}}^1$ obtained with POD.
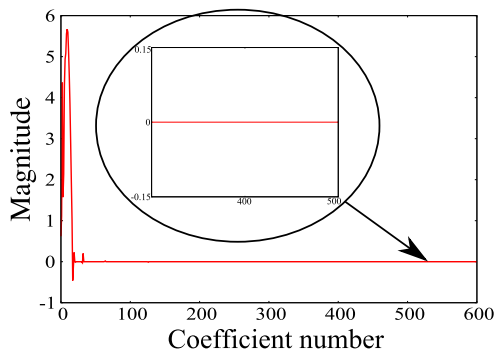
(c) WAVinPOD approximation in wavelet domain.

Fig. 8. Wavelet transform coefficients; 6 levels of decomposition performed with $db3$.

Additional analysis was performed on matrix A_2 (see noisy data in Fig. 9(a)). Again, all the criteria for selecting the number of significant modes recovered a rank equivalent to the dimensionality of true data-set, $k = 3$, with $N = 20$. Approximations of A_{true}^2 (shown in Fig. 9(a)), initially corrupted with noise of standard deviation $\sigma_n = 0.5$, obtained with POD, 1D wavelet thresholding and WAVinPOD are plotted every 5th observation in Figs. 9(b)–9(d). Fig. 10 compares the performance of each methodology for increasing noise level with the number of observations set to $N = 20$. The two-dimensional wavelet thresholding and WAV2inPOD provided the lowest quality de-noising and are not depicted in the figures for clarity. Our method, WAVinPOD, again outperformed the other techniques, producing the highest signal-to-noise ratios and data-sets with the smallest errors in Frobenius and L_2 matrix norm.
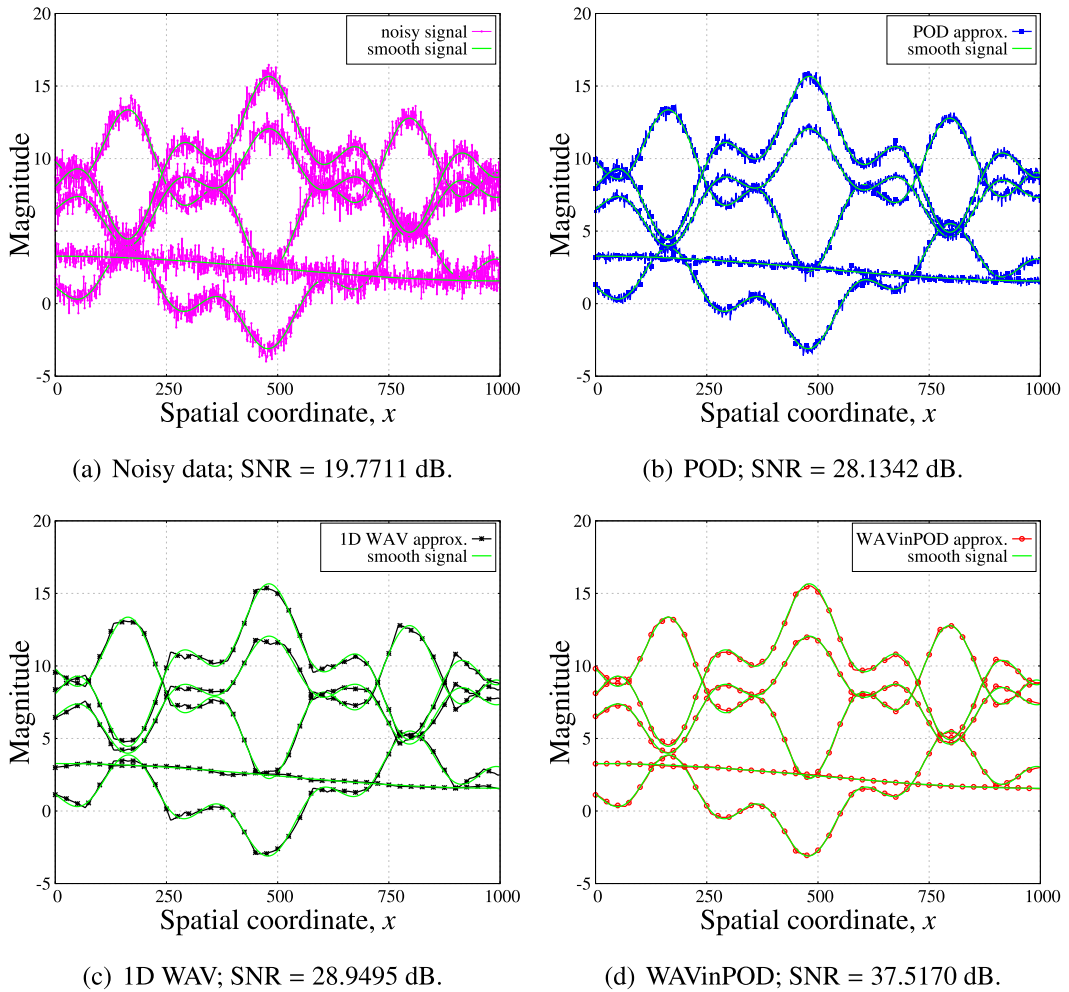
(a) Noisy data; SNR = 19.7711 dB.

(b) POD; SNR = 28.1342 dB.

(c) 1D WAV; SNR = 28.9495 dB.

(d) WAVinPOD; SNR = 37.5170 dB.

Fig. 9. Reconstruction of synthetic signals, corrupted with Gaussian noise (SNR = 19.7711 dB).

It is important to mention that wavelet thresholding alone does not reduce the dimensionality of a matrix as well as POD or WAVinPOD. Both methods based on SVD produce approximations with rank equal to the number of dominant modes. For the case of $N = 240$ observations (see Table 1), 2D wavelet analysis produced a matrix of rank = 20 (for 1D WAV rank = 80), whereas POD and WAVinPOD resulted in a matrix of rank = 2. Singular value decomposition gives the best rank reduction in all norms that are invariant under rotation [25]. The low rank of the matrix is important if compression of the data is of interest. Transferring the original matrix with rank 240 requires sending $240 \times 1000 = 240000$ samples of information. When the rank is lowered to 20, the matrix can be represented by 3 components that together contain $240 \times 20 + 20 \times 20 + 1000 \times 20 = 25200$ elements, which is almost an order of magnitude lower than for the original set. Having an approximation of rank = 2 further reduces the size of the matrices resulting in only 2484 samples of the data (about 96 times less than the original!). It should be stressed that using WAVinPOD allows not only successful extraction of information from disturbed data, but also reduces the computational cost of further processing or storage.

4. Removing noise from simulation results

Grinberg [3] showed how non-stationary particle data can be successfully de-noised using only POD with time windows. Recently, his method has been applied to study the filament dynamics [26]; these simulations will be analysed with WAVinPOD as part of the future work. In this paper, we report the results of applying WPOD, 2D wavelet thresholding and WAVinPOD (adapted to a moving window) to velocity measurements from time-dependent channel flows and density profiles from a simulation of phase separation phenomenon, performed with either MD or DPD. The aim is to investigate the benefits of applying WAVinPOD to real simulation data and compare its performance with other procedures in time-dependent modelling. In all the problems considered, a spatial distribution of the observables was calculated using the binning method (see Allen and Tildesley [27]). In this approach, the system's domain is partitioned into a number of bins and the mean velocity and number of particles in each bin is computed based on their position.
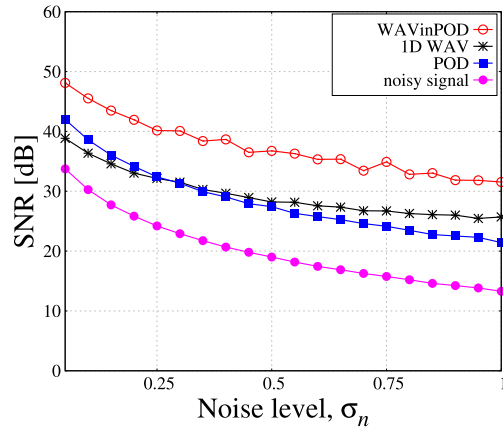
(a) Averaged signal-to-noise ratios for each reconstruction.
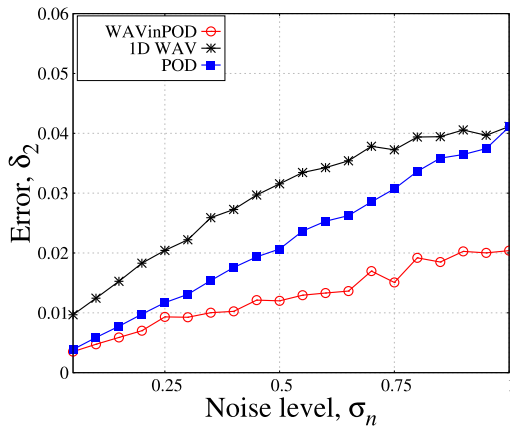
(b) Errors in L_2 norm.
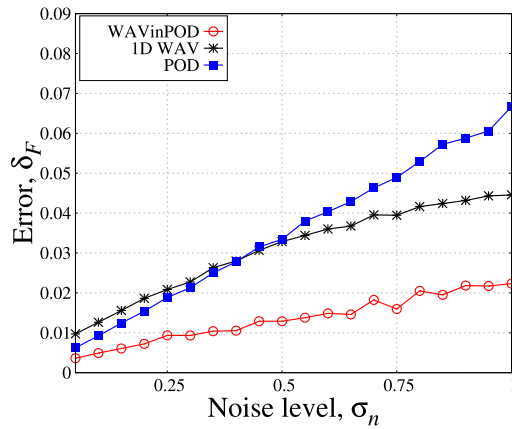

(c) Errors in Frobenius norm.

Fig. 10. Result of de-noising matrix A_2 ($N = 20$ oscillating signals of length $M = 1000$) with POD, 1D wavelet thresholding, and WAVinPOD. For WAVinPOD and WAV, soft wavelet thresholding with the $db3$ filter and 6 levels of decomposition was used.

The molecular dynamics simulations were carried out using the open-source mdFOAM solver which is implemented within the open source fluid dynamics toolbox, OpenFOAM, and developed and maintained by researchers at the University of Strathclyde and the University of Edinburgh.[5] The model fluids were either liquid argon in a krypton channel or water flowing between two silicon walls. All of the parameters from the MD simulations are presented in reduced units; the reference values are linked to the Lennard-Jones (L-J) potentials. For water, the quantities for length, energy and mass, respectively, were: $\sigma_{r,H_2O} = 3.1589 \cdot 10^{-10}$ m, $\epsilon_{r,H_2O} = 1.2868 \cdot 10^{-21}$ J, $m_{r,H_2O} = 2.987 \cdot 10^{-26}$ kg, and for argon: $\sigma_{r,Ar} = 3.405 \cdot 10^{-10}$ m, $\epsilon_{r,Ar} = 1.6568 \cdot 10^{-21}$ J, $m_{r,Ar} = 6.6904 \cdot 10^{-26}$ kg. We used the rigid TIP4P/2005 water model as described by Abascal and Vega [28]. For the DPD modelling, this was performed using DL MESO[6] and the dimensionless parameters were converted to physical units with the reference values of the cut-off radius, $r_{cut_r} = 6.46 \cdot 10^{-10}$ m. This was calculated with one DPD particle representing 3 water molecules, based on the relation described by Ghoufi et al. [29]. The DPD energy was $k_B T_r = 4.114 \cdot 10^{-21}$ J (where k_B is Boltzmann's constant and $T_r = 298$ K), and the mass of one water molecule was $m_r = 2.987 \cdot 10^{-26}$ kg. Parameters for DPD that enforce proper water compressibility for the system were taken from Groot and Warren [30].

The first simulations were modelled with MD and considered different types of flow involving liquid argon in a krypton nanochannel: flow driven by a periodically oscillating force with and without roughness and flow generated by an oscillating upper wall. For each case a periodic domain with dimensions: $25 \times 50 \times 10$ ($x \times y \times z$), with the wall thickness set to 5 in reduced units. The Lennard-Jones parameters describing argon–krypton interaction were: $\sigma_{Ar-Kr} = 1.02\sigma_{r,Ar}$ and $\epsilon_{Ar-Kr} = 1.18\epsilon_{r,Ar}$, taken from Sofos et al. [31] and Gotoh [32]. The motion of fluid particles was weakly coupled to a thermal
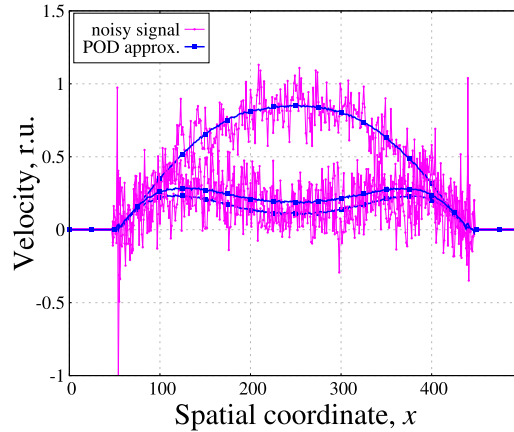
(a) De-noised velocity profiles with WPOD.
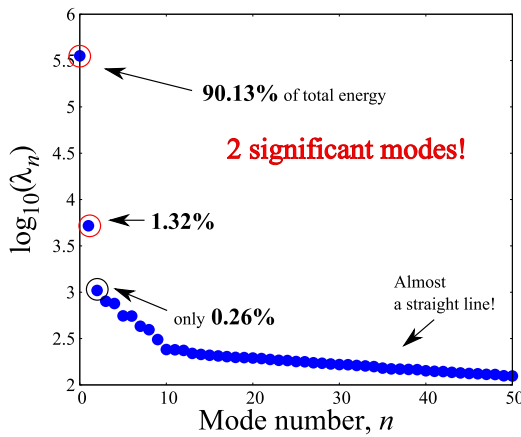
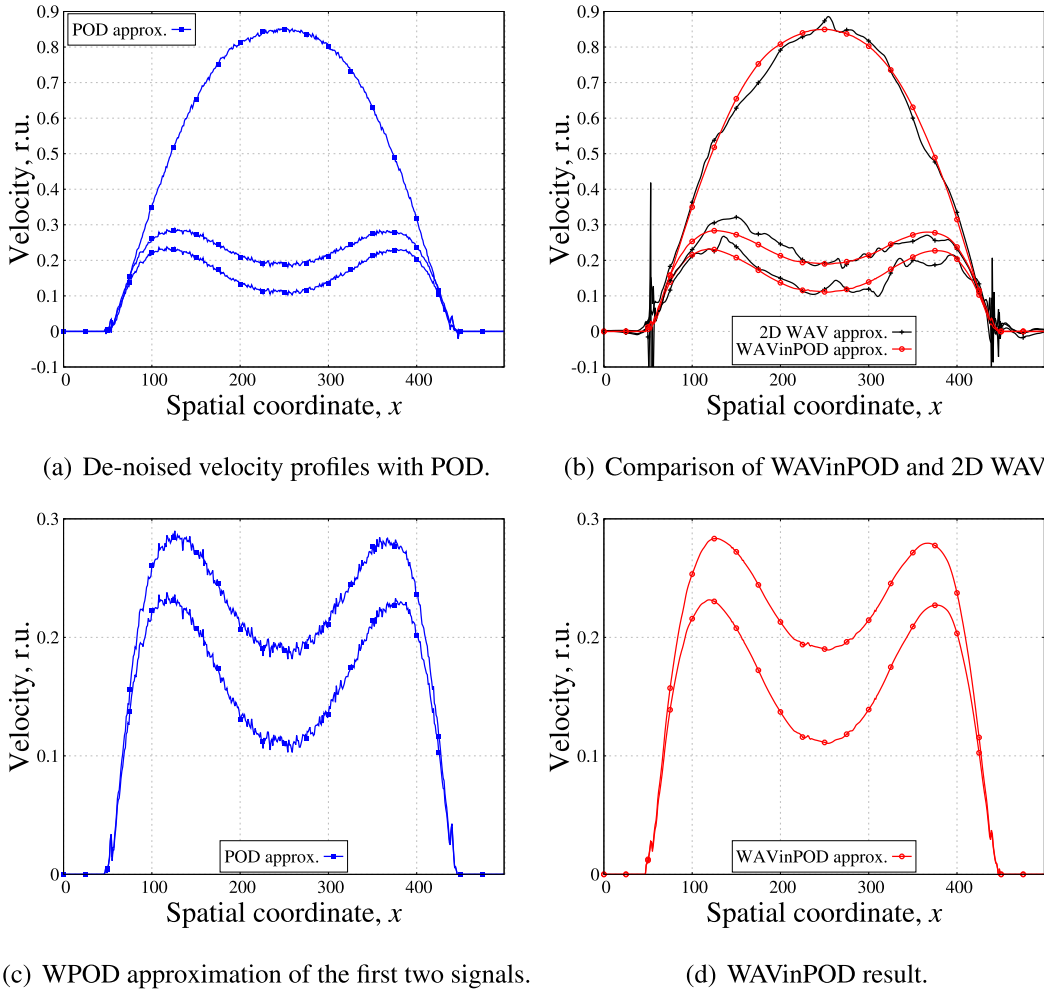(b) LEV diagram of eigenspectrum for the first 50 values.
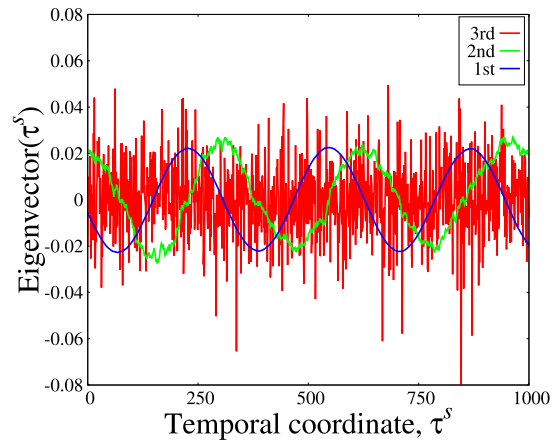
(c) The eigenvectors for $\tau^s = 1000$.

Fig. 11. Result of applying WPOD to the developed velocity field from an MD simulation of a periodically-pulsating flow in a smooth channel; $N_{ts} = 1$ and $N_{POD} = 4000$.

reservoir via the Berendsen thermostat. In order to not affect the calculations, the thermostat was applied only to the velocity component perpendicular to the flow direction. Based on the equipartition theorem, such a configuration ensures that each degree of freedom of the system is close to the right temperature [33]. For all simulations, the computational domain was divided into $M = 500$ horizontal bins, where the sampling of observables took place. After the target values of steady-state had been reached (temperature $T = 1$ and density $\rho = 0.8187$), the density controller was switched off and either an oscillating force of $F_x = 0.6\sin(0.0125 \cdot 2\pi t)$ was applied to every argon particle in the fill region, or the upper wall was set to move with a velocity $V = 0.5\sin(0.0125 \cdot 2\pi t)$. The time-step for each simulation was set to $\Delta t = 0.0025$ in reduced units, but the data was sampled every 100th time-step, $\Delta t_w = 100\Delta t = 0.25$ in order to ensure statistical independence.

The moving window was utilised in the manner as described by Grinberg [3]. While the WPOD window moved throughout the simulation, the matrix was being updated every N_{ts} of t_w samples. To allow a straightforward comparison of all the de-noising methods, in most of the cases no averaging was applied prior to data filtering, i.e. $N_{ts} = 1$. We stress that POD and WPOD essentially differ only in the implementation. Therefore, the results obtained from POD with a moving window are labelled in figures as POD approximations. Wavelet thresholding and WAVinPOD were applied to the same window, T_{POD}. The wavelet filter was again chosen from Daubechies family, but in this case the $db4$ was utilised with 8 levels of decomposition, in order to recover more smoothness.

Fig. 11(a) shows the WPOD approximation of instantaneous velocity measurements and original noisy profiles obtained from the oscillating force-driven flow in a smooth channel. For this problem the initial matrix contained $N = 4000$ observations and $M = 500$ velocity measurements at each time-step and a substantial reduction in noise level was achieved. Initial matrix contained. In this particular case, only two modes were used for velocity field extraction as a result of performing
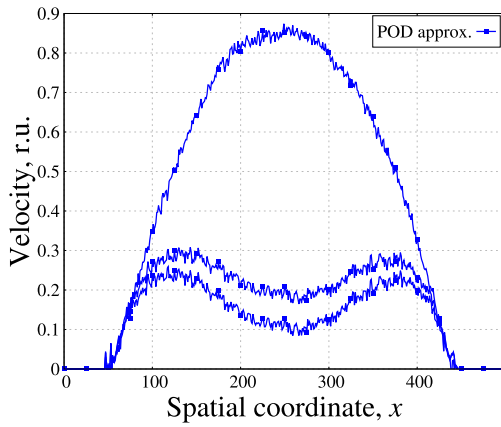
(a) De-noised velocity profiles with POD.

(b) Comparison of WAVinPOD and 2D WAV.

(c) WPOD approximation of the first two signals.

(d) WAVinPOD result.

Fig. 12. Analysis of de-noising performance; $N_{ts} = 1$ and $N_{POD} = 4000$; for WAVinPOD and WAV soft wavelet thresholding with the $db4$ filter and 8 levels of decomposition was used.

the following analysis: examination of the energy levels of the eigenspectrum, the rate of decay of the eigenvalues, and studying the smoothness of the eigenvectors. The first two eigenvalues were the most energetic. The LEV diagram is illustrated in Fig. 11(b); it can be seen that $\lambda_{n=1} = s_1^2$ and $\lambda_{n=2} = s_2^2$ are decaying much faster than the other eigenvalues, i.e. they correspond to the dominant modes. Eigenvalues with $k > 2$ represent features with a short correlation time (noise). As shown in Fig. 11(c), the 1st and 2nd eigenvector (or left singular vector) described the oscillating nature of the data whilst the remaining modes contained higher frequencies. In addition, it should be stressed that the first eigenvector was smoother than the second one, which was slightly disturbed. This observation suggests that noise had not been entirely filtered out from the second mode. We also utilised the SVHT for singular values, but the estimated threshold was too low, even when the noise level was determined from wavelet coefficients. This is due to the fact that simulation data often suffers from correlated noise for which the optimal threshold developed by Gavish and Donoho [13] has not been derived. However, we observed that if the square root of the standard deviation used in Eq. (7) is inserted into the formula for SVHT with known noise, the same number k is retained as the value established with other tests. For all the simulations discussed in this section we utilised this approach. This strategy is based solely on the experience and further investigation of the optimal universal threshold for de-noising particle data should be conducted.

Figs. 12(a)–12(d) compare WPOD approximation with the result obtained by applying WAVinPOD and wavelet thresholding to the same noisy matrix. It can be seen that WAVinPOD provides smoother profiles than WPOD for the same number of observations. The two-dimensional WAV approach produced the poorest result (see Fig. 12(b)), as the wavelets seemed to *follow* the noise. In addition, when only $N = 400$ observations (10 times less) were used for de-noising, WAVinPOD was still capable of extracting similar profiles, while applying the other techniques resulted in artifacts, and preserved unwanted frequencies. Figs. 13(a)–13(c) compare WPOD and WAVinPOD for $N = 400$, showing that the latter is more efficient in extracting information from the noisy data. Applying the criteria for defining k to the matrix with $N = 400$ successfully identified the number of significant modes, i.e. $k = 2$.
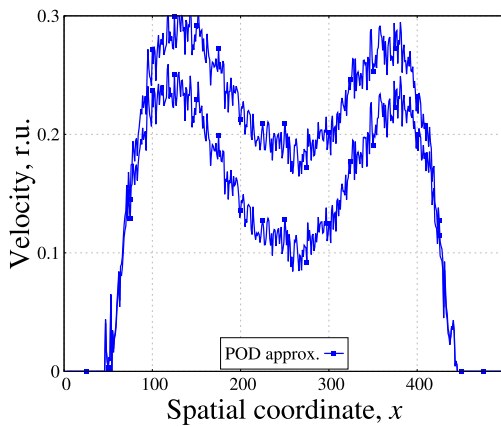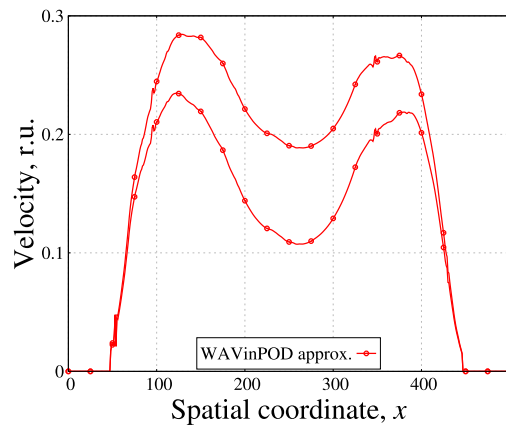
(a) De-noised velocity profiles with POD.
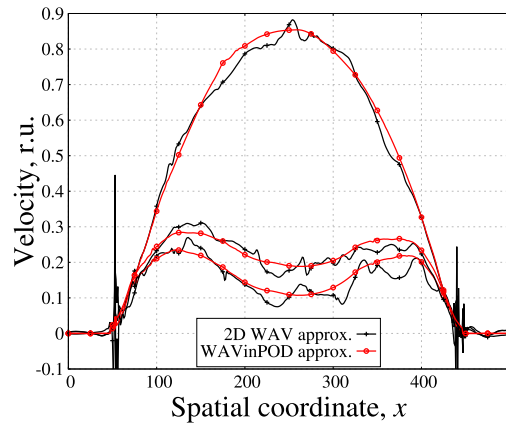
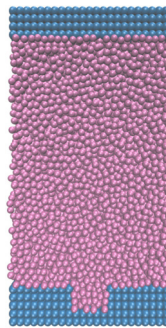(b) Comparison of WAVinPOD and 2D WAV.


(c) WPOD approximation of the first two signals.

(d) WAVinPOD result.

Fig. 13. Filtering velocity data with smaller number of observations: $N_{ts} = 1$ and $N_{POD} = 400$; for WAVinPOD and WAV soft wavelet thresholding with $db4$ filter and 8 levels of decomposition were used.

Fig. 14. Snapshot of the channel with introduced roughness.

Molecular dynamics simulations are often used to investigate the influence of atomistic scale surface roughness on the slip behaviour in liquid films [31,34]. In order to show how applying de-noising techniques can improve the study of slip phenomena, we introduced surface roughness to the previously described system. A periodic roughness was applied by placing a cavity with dimensions $5 \times 3 \times 10$ to the lower wall as presented in Fig. 14. All other simulation parameters were kept the same. During one simulation run, $N = 10000$ velocity profiles consisting of $M = 500$ points were collected. Fig. 15(a) illustrates how poor were the original instantaneous measurements in comparison to the velocity profiles obtained with WPOD; refinement of the grid present in the plot indicates the position of the cavity. If statistical averaging is to be
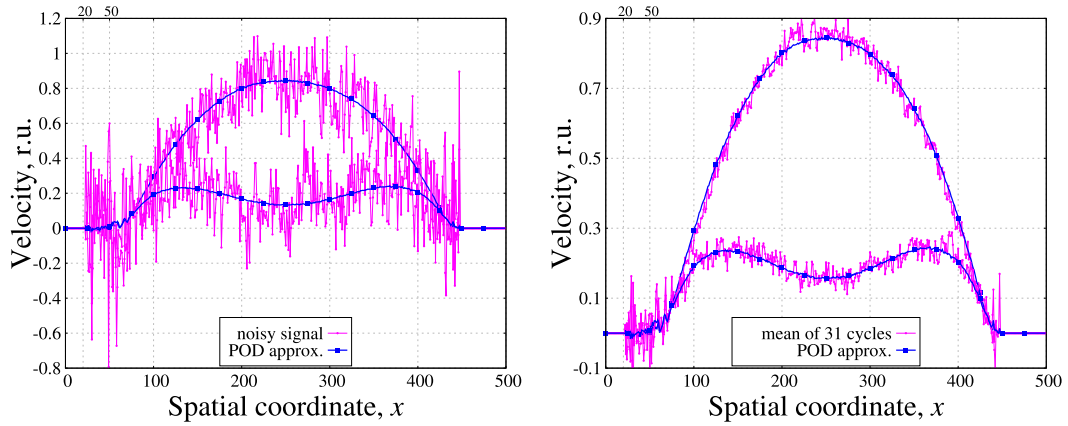
(a) WPOD; $N_{POD} = 10000$.

(b) Comparison of WPOD for $N_{POD} = 9920$ and averaging over 31 full periods.

Fig. 15. Filtering velocity data with WPOD with $N_{ts} = 1$ and statistical averaging. Refinement of the grid indicates the position of the cavity in the lower wall.
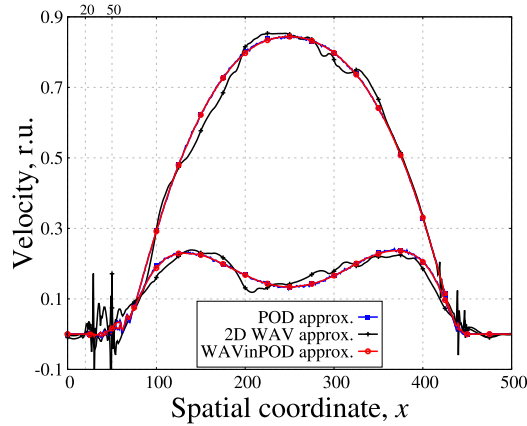
Fig. 16. Comparison of velocity profiles obtained with de-noising techniques; $N_{ts} = 1$ and $N_{POD} = 10000$. Note that WAVinPOD and WPOD produced almost identical distributions.
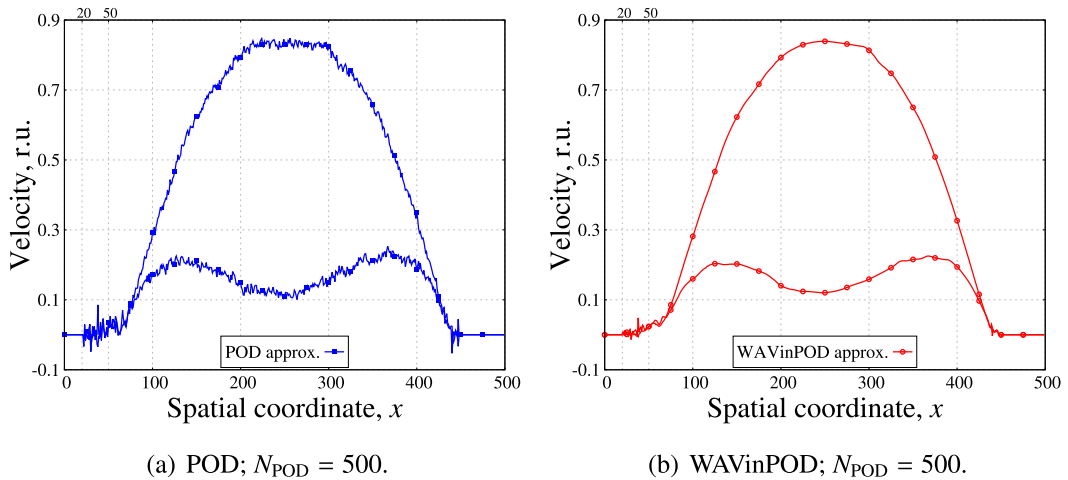
(a) POD; $N_{POD} = 500$.

(b) WAVinPOD; $N_{POD} = 500$.

Fig. 17. Comparison of WPOD and WAVinPOD (with the $db4$ filter and 8 resolutions) for decreasing number of velocity measurements. Both methods allowed for more visual study of slip-phenomena. However, more data would be needed for WPOD to match the performance of WAVinPOD.

(a) $N_{ts} = 1$, $N_{POD} = 6000$.

(b) WPOD, WAVinPOD and WAV; $N_{ts} = 1$ and $N_{POD} = 6000$. Note that WAVinPOD and WPOD produced comparable results.

(c) Velocity distributions obtained with WPOD for $N = 600$.

(d) WAVinPOD for $N = 600$.
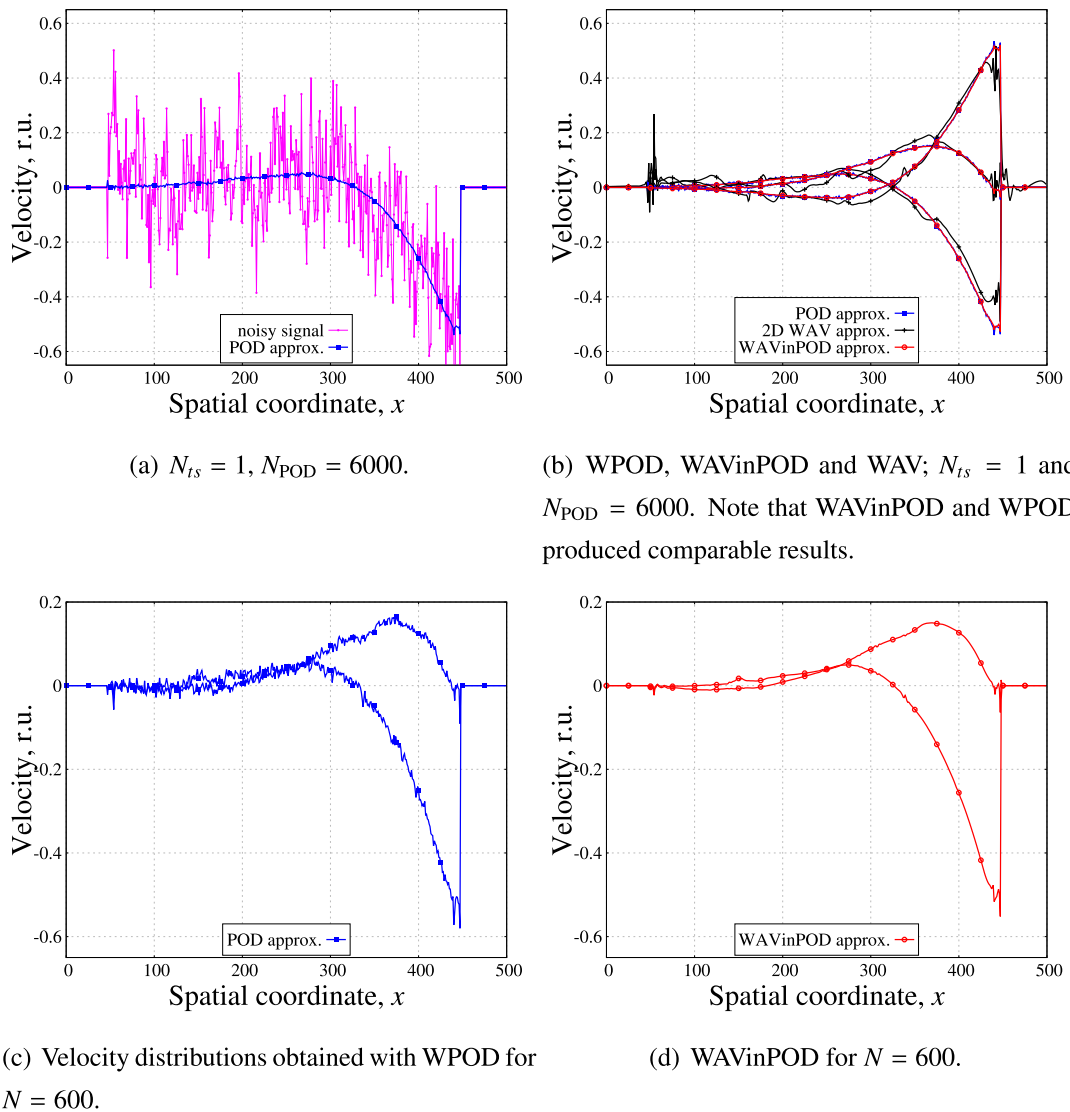
Fig. 18. Filtering velocity data from the oscillatory Couette flow simulated with MD; comparison of WPOD, WAVinPOD and WAV de-noising technique.

employed in order to improve the quality of the results, a simulation has to be either performed several times, or, in the case of periodic oscillations, left to run for long enough to gather a sufficient number of samples. In the simulation, a full period translated to 320 velocity measurements as $80/\Delta t_w = 320$, where the data was output at every $\Delta t_w = 0.25$ interval. In other words, in the collection of $N = 10000$ samples, there were 31 full oscillations, which were used to obtain an ensemble solution. Fig. 15(b) compares the quality of cumulative mean (average of 31 full periods) and the WPOD approximation. The latter clearly extracted smoother velocity profiles for the same number of measurements ($N = 31 \times 320 = 9920$). To obtain a comparable level of de-noising with statistical averaging, much more data would have to be collected, increasing the computational cost. Furthermore, WPOD (or WAVinPOD) do not require any *a priori* information regarding the nature of the oscillations, which is clearly beneficial, e.g. when the frequency of fluctuations changes over time. Applying 2D wavelet thresholding alone did not produce satisfactory results as shown in Fig. 16. Further improvement of the de-noising efficiency can be obtained by utilising WAVinPOD as illustrated in Figs. 17(a)–17(b). If the velocity approximation produced with WPOD for $N = 10000$ measurements (see Figs. 15(a) and 15(b)) is considered as a desired solution, the SNR of velocity profiles recovered with WAVinPOD for $N_{POD} = 500$ had SNR = 32.2875 dB, while WPOD provided SNR = 28.8545 dB (about 12% lower). In general, for a decreasing number of time samples, WPOD retains more noise, while WAVinPOD removes additional disturbances enhancing the analysis of slip velocity at the rough surface.

The last example with liquid argon was a non-equilibrium steady-state and time-periodic molecular dynamics simulation of Couette flow. The influence of oscillatory shear on the boundary slip is often studied with particle-based simulations, as experimental analysis is challenging [35]. We applied all the techniques to a collection of $N = N_{POD} = 6000$ noisy velocity
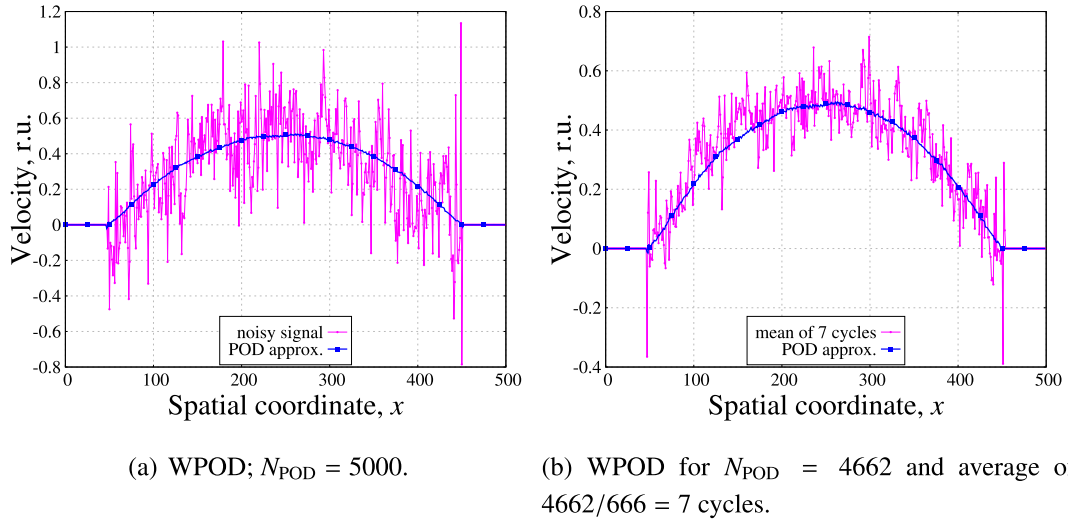
(a) WPOD; $N_{POD} = 5000$.

(b) WPOD for $N_{POD} = 4662$ and average of $4662/666 = 7$ cycles.

Fig. 19. Statistical averaging and WPOD low-rank approximations of the velocity measurements from a water flow simulation performed with MD.

(a) WPOD and 2D WAV for $N_{POD} = 5000$.

(b) WAVinPOD for $N_{POD} = 5000$.

(c) WPOD for $N_{POD} = 1000$.
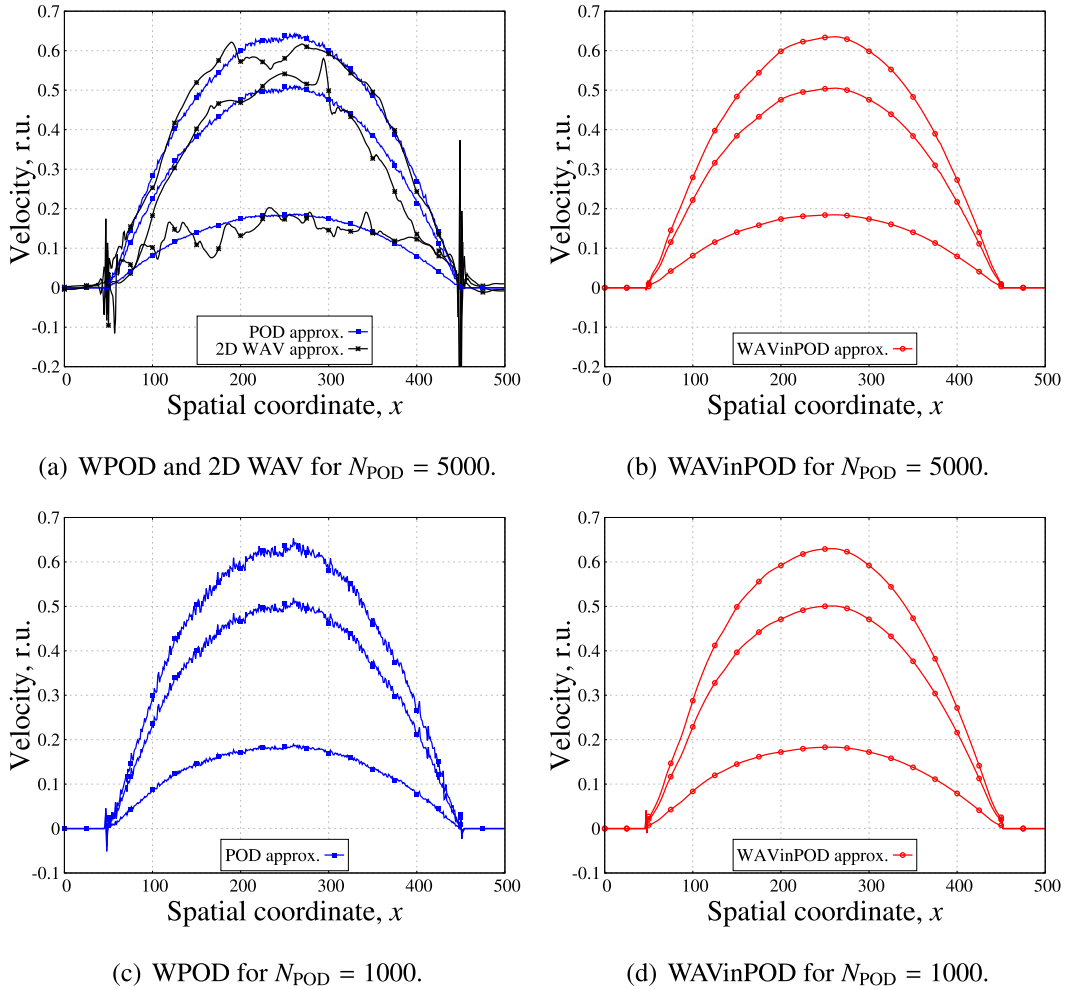
(d) WAVinPOD for $N_{POD} = 1000$.

Fig. 20. Comparison of WPOD, WAVinPOD and wavelet shrinkage (with filter $db4$, 8 resolutions and $k = 1$) in de-noising velocity data from simulation of unsteady water flow. WAVinPOD outperformed WPOD and statistical averaging, extracting smooth profiles even for 5 times smaller $N = N_{POD} = 1000$ and $N_{ts} = 1$.
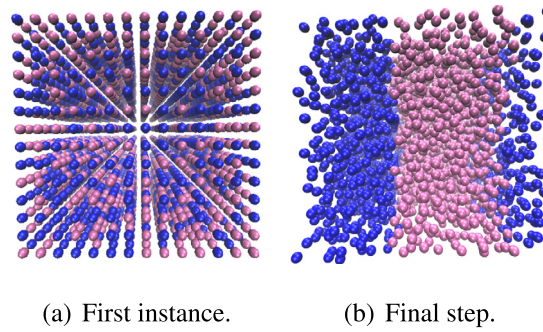
(a) First instance.　　　　　(b) Final step.

Fig. 21. Snapshots of the first and last time-step of the DPD simulation.

(a) Instantaneous averages at the final step.　　　　(b) Evolution of phase separation.
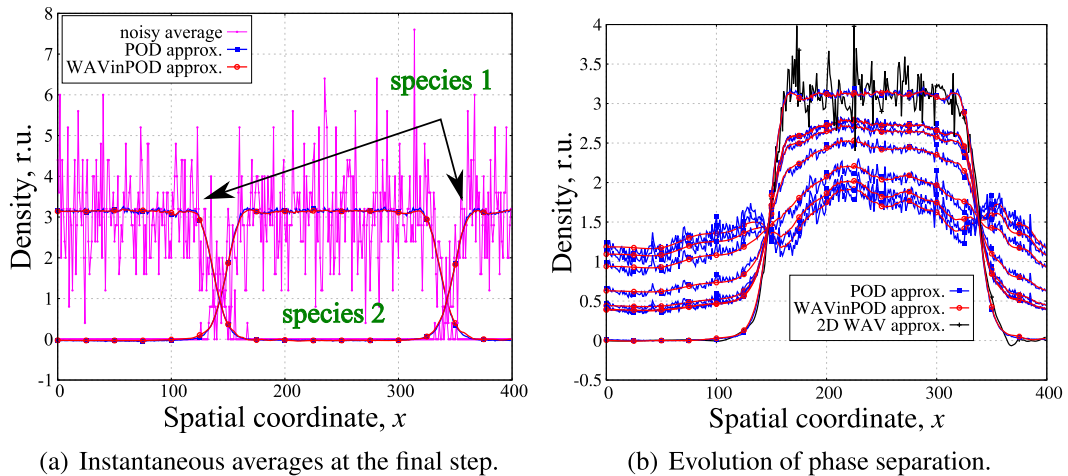
Fig. 22. Comparison of WAVinPOD, WPOD and 2D WAV applied to the density field from a DPD simulation of a phase separation phenomenon; $N_{ts} = 10$ and $N_{POD} = 2000$.

profiles with $M = 500$ samples obtained from the shear-driven flow induced by an oscillating upper wall (see Figs. 18(a) and 18(b)). The results were compared in Figs. 18(c)–18(d) with a smaller data-set containing only the last $N = 600$ measurements in order to establish whether any of the methods can reduce the computational time required to extract smooth data that is easy to analyse and process. Wavelet thresholding within POD extracted similar quality velocity profiles but WPOD required 10× more snapshots; WAVinPOD produced an ensemble with 18% higher SNR than WPOD, if we assume that the *true* solution is a set of profiles obtained with WPOD for $N_{POD} = 5000$. Two-dimensional wavelet thresholding with $db4$ and 8 resolutions was again the least successful. Our criteria for determining the number of significant modes recovered $k = 2$ for each data-set.

Molecular dynamics can be applied to many *real-life* problems, e.g. for studying water flow in nanostructured membranes. However, such modelling is quite challenging; the complexity of the water system requires small time-steps and the simulations often contain substantial statistical noise that is computationally demanding to reduce. In order to assess how de-noising techniques can improve the analysis of such data, we applied WPOD, WAVinPOD and wavelet shrinkage to an oscillating Poiseuille flow of water between two rigid planar silicon walls. The proportions of the computational domain were kept the same as in the previous MD simulations; the periodic channel had dimensions: $25 \times 50 \times 10$ ($x \times y \times z$), with thickness of the walls set to 5, expressed in reduced units for water. Water molecules were driven by a force of $F_x = 0.6 \sin(0.0125 \cdot 2\pi t)$ (again with different reduced units than for argon). A smaller time-step, $\Delta t = 0.0012$, was used in order to capture all the important dynamics with write-interval $\Delta t_w = 0.12$ (data was output every $\tau^s = 100\Delta t$). The system's temperature was set to $T = 3.816$, and water density was $\rho = 1.047$. At first, all the filtering methods were applied to $N = 5000$ and $M = 500$, resulting, apart from wavelet thresholding, in much improved instantaneous data quality.

Figs. 19(a) and 19(b) show how WPOD with $k = 1$ performed in comparison with statistical averaging over the full oscillatory period (one period every $80/\Delta t_w \approx 666$ observations) for the same data-set; it can be readily observed that the ensemble mean was still very noisy, and more measurements would be required to extract the same velocity profiles as with WPOD. Out of all the filtering techniques considered, WAVinPOD obtained the best results for $N = 5000$ (see Fig. 20(b)), and for a smaller system with $N = 1000$, as illustrated in Figs. 20(c)–20(d). Wavelet thresholding alone did not provide a good approximation, which shows how difficult it is to choose an adequate wavelet model to successfully de-noise the data.

In order to show the versatility of WAVinPOD, we applied it to density fields from a simulation of phase separation phenomenon performed with DPD. The studied system consisted of a periodic box filled with 3000 particles, divided into 2 species (1500 each). The particles from both species had the same size, and each bead had a mass $= 1$ in DPD units, but were set to repel each other in order to form two layers, as shown in Figs. 21(a)–21(b). We applied WAVinPOD together with WPOD to density profiles obtained from $M = 400$ bins spanning the x-direction. The simulation run for $\tau^s = 20000$, and an ensemble of $N_{POD} = 2000$ averages of $N_{ts} = 10$ samples each was formed. In Fig. 22(a), the original profiles for both species are plotted against WPOD and WAVinPOD approximations. The profiles were extracted by retaining only $k = 2$ modes and using the $db4$ filter with 8 decompositions. Both de-noising techniques produced similar results. However, when the density profiles were plotted at different instances (see Fig. 22(b)), it was clear that WAVinPOD provided smoother estimates. Again, applying wavelet thresholding alone resulted in poor filtering quality. It should be stressed that no additional averaging could have been performed without losing information on how the system was evolving.

5. Conclusions and remarks

We have introduced a novel de-noising technique for unsteady signals that couples wavelet thresholding with proper orthogonal decomposition, WAVinPOD. Our procedure combines the optimal dimensionality reduction of POD with the time-frequency localisation properties of wavelets. The new method was applied to a number of oscillating synthetic signals and unsteady particle-based data to investigate its capabilities to reduce noise. It was shown that the methodology outperforms the other techniques in extracting significant information from corrupted measurements. We demonstrated that WAVinPOD is beneficial in reducing noise in oscillatory flows, and can successfully extract smooth profiles of the properties. Relative to standard statistical averaging, WPOD, and wavelet shrinkage, WAVinPOD achieves higher signal-to-noise ratios and lower errors in Frobenius and L_2 norms at reduced computational cost. Ensemble solutions obtained with WAVinPOD allow a more accurate analysis of results obtained with particle-based simulations, including molecular dynamics and dissipative particle dynamics. This work has therefore provided a contribution to the challenge of improving the communication in multi-scale modelling that couples a molecular domain to continuum-based domain, such as fluid dynamics.

Performing WAVinPOD requires an estimate of k, which denotes the number of significant structures contained in the data. Numerous criteria for determining which orthogonal modes are dominant have been listed in this work. Selection of POD modes for separating the ensemble solution from fluctuating components can be done adaptively. It can also be performed by the user at the end of the simulation run as part of post-processing. In addition, we proposed applying the singular value hard threshold to automate the filtering process of WAVinPOD. However, as the method was specifically designed for systems corrupted with white noise, SVHT does require some modification. Further study will be performed in order to optimise the threshold estimate, and provide a sound mathematical explanation.

Although WAVinPOD is shown to be a valuable tool for removing noise from particle-based simulations, there is a clear scope to extend its applicability. One possible extension is its application to direct simulation Monte Carlo data to analyse problems in rarefied gas dynamics. This will be the subject of further research.

Acknowledgements

References

[1] G. Karniadakis, A. Beskok, N. Aluru, Microflows and Nanoflows: Fundamentals and Simulation, Springer, 2005.
[2] P. Berkooz, G. Holmes, J.L. Lumley, The proper orthogonal decomposition in the analysis of turbulent flows, Annu. Rev. Fluid Mech. 25 (1) (1993) 539–575.
[3] L. Grinberg, Proper orthogonal decomposition of atomistic flow simulations, J. Comput. Phys. 231 (16) (2012) 5542–5556.
[4] D.L. Donoho, J.M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, Biometrika 81 (3) (1994) 425–455.
[5] R.R. Coifman, D.L. Donoho, Translation-Invariant De-Noising, Springer, 1995.
[6] P. Holmes, J.L. Lumley, G. Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, Cambridge University Press, 1998.
[7] A. Chatterjee, An introduction to the proper orthogonal decomposition, Curr. Sci. 78 (7) (2000) 808–817.
[8] M. Farge, K. Schneider, Analysing and computing turbulent flows using wavelets, in: M. Lesieur, A. Yaglom, F. David (Eds.), New Trends in Turbulence, vol. 74, Springer, 2002, pp. 449–503.
[9] S.G. Mallat, A Wavelet Tour of Signal Processing, 2 edn., Academic Press, 1999.
[10] M. Vetterli, J. Kovačević, Wavelets and Subband Coding, vol. 87, Prentice–Hall, 1995.
[11] J.L. Lumley, The structure of inhomogeneous turbulent flows, in: Atmospheric Turbulence and Radio Wave Propagation, 1967, pp. 166–178.
[12] J.M. Craddock, C.R. Flood, Eigenvectors for representing the 500 mb geopotential surface over the Northern Hemisphere, Q. J. R. Meteorol. Soc. 95 (405) (1969) 576–593.
[13] M. Gavish, D.L. Donoho, The optimal hard threshold for singular values is 4/sqrt (3), IEEE Trans. Inf. Theory 60 (8) (2014) 5040–5053.
[14] S.G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.
[15] Y. Meyer, Orthonormal wavelets, in: Wavelets, Springer, 1989, pp. 21–37.

[16] I. Daubechies, Ten Lectures on Wavelets, vol. 61, SIAM, 1992.

[17] G. Strang, T. Nguyen, Wavelets and Filter Banks, SIAM, 1996.

[18] D.L. Donoho, De-noising by soft-thresholding, IEEE Trans. Inf. Theory 41 (3) (1995) 613–627.

[19] I.M. Johnstone, B.W. Silverman, Wavelet threshold estimators for data with correlated noise, J. R. Stat. Soc., Ser. B, Stat. Methodol. 59 (2) (1997) 319–351.

[20] B.R. Bakshi, Multiscale PCA with application to multivariate statistical process monitoring, AIChE J. 44 (7) (1998) 1596–1610.

[21] M. Aminghafari, N. Cheze, J.-M. Poggi, Multivariate denoising using wavelets and principal component analysis, Comput. Stat. Data Anal. 50 (9) (2006) 2381–2398.

[22] T.J. Ross, Fuzzy Logic with Engineering Applications, John Wiley & Sons, 2009.

[23] I.T. Jolliffe, Discarding variables in a principal component analysis. I: Artificial data, Appl. Stat. (1972) 160–173.

[24] D.S. Wilks, Statistical Methods in the Atmospheric Sciences, vol. 100, Academic Press, 2011.

[25] D. Achlioptas, F. Mcsherry, Fast computation of low-rank matrix approximations, J. ACM 54 (2) (2007) 9.

[26] M. Deng, L. Grinberg, B. Caswell, G.E. Karniadakis, Effects of thermal noise on the transitional dynamics of an inextensible elastic filament in stagnation flow, Soft Matter 11 (24) (2015) 4962–4972.

[27] M.P. Allen, D.J. Tildesley, Computer Simulation of Liquids, Clarendon Press, Oxford, 1987.

[28] J.L.F. Abascal, C. Vega, A general purpose model for the condensed phases of water: TIP4P/2005, J. Chem. Phys. 123 (23) (2005) 234505.

[29] A. Ghoufi, J. Emile, P. Malfreyt, Recent advances in Many Body Dissipative Particles Dynamics simulations of liquid–vapor interfaces, Eur. Phys. J. E 36 (1) (2013) 1–12.

[30] R.D. Groot, P.B. Warren, Dissipative particle dynamics: bridging the gap between atomistic and mesoscopic simulation, J. Chem. Phys. 107 (11) (1997) 4423–4435.

[31] F.D. Sofos, T.E. Karakasidis, A. Liakopoulos, Effects of wall roughness on flow in nanochannels, Phys. Rev. E 79 (2009) 026305.

[32] K. Gotoh, Liquid structure and Lennard-Jones (6, 12) pair potential parameters, Nature 239 (96) (1972) 154–156.

[33] A.I. Khinchin, Mathematical Foundations of Statistical Mechanics, Courier Corporation, 1949.

[34] N.V. Priezjev, Effect of surface roughness on rate-dependent slip in simple fluids, J. Chem. Phys. 127 (14) (2007).

[35] N.V. Priezjev, Molecular dynamics simulations of oscillatory Couette flows with slip boundary conditions, Microfluid. Nanofluid. 14 (1–2) (2013) 225–233.