Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Journal of Computational and Applied Mathematics 170 (2004) 27–58
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

Modelling the dynamics of nonlinear partial differential equations using neural networks

Nejib Smaoui*, Suad Al-Enezi

Department of Mathematics and Computer Science, Kuwait University, P.O. Box 5969, Safat 13060, Kuwait

Received 29 October 2002; received in revised form 15 December 2003

Abstract

The dynamics of two nonlinear partial differential equations (PDEs) known as the Kuramoto–Sivashinsky (K–S) equation and the two-dimensional Navier–Stokes (N–S) equations are analyzed using Karhunen–Loéve (K–L) decomposition and artificial neural networks (ANN). For the K–S equation, numerical simulations using a pseudospectral Galerkin method is presented at a bifurcation parameter $\alpha = 17.75$, where a dynamical behavior represented by a heteroclinic connection is obtained. We apply K–L decomposition on the numerical simulation data with the task of reducing the data into a set of data coefficients. Then we use ANN to model, and predict the data coefficients at a future time. It is found that training the neural networks with only the first data coefficient is enough to capture the underlying dynamics, and to predict for the other remaining data coefficients. As for the two-dimensional N–S equation, a quasiperiodic behavior represented in phase space by a torus is analyzed at $Re = 14.0$. Applying the symmetry observed in the two-dimensional N–S equations on the quasiperiodic behavior, eight different tori were obtained. We show that by exploiting the symmetries of the equation and using K–L decomposition in conjunction with neural networks, a smart neural model is obtained.

MSC: 35k55; 37N10; 65M70

Keywords: One-dimensional Kuramoto–Sivashinsky equation; Two-dimensional Navier–Stokes equations; Karhunen–Loeve decomposition; Neural networks

1. Introduction

During the last thirty years, there has been a lot of interest in studying dynamical systems that arise from solving the initial value problem for nonlinear partial differential equations (PDEs)

* Corresponding author. Tel.: +965-4811188x5604; fax: +965-4817201.
E-mail address: smaoui@mcs.sci.kuniv.edu.kw (N. Smaoui).

0377-0427/$ - see front matter
doi:10.1016/j.cam.2003.12.045

[8,16,17,20,21,28,41,42]. In this paper, two nonlinear parabolic PDEs known as the one-dimensional Kuramoto–Sivashinsky (K–S) equation and the two-dimensional Navier–Stokes (N–S) equations are studied. The one-dimensional K–S equation has been first derived in 1976 by Kuramoto and Tsuzuki [38] as a model for the interfacial instabilities in the context of angular phase turbulence for a system of a reaction diffusion equation that models the Belouzov–Zabotinskii reaction in three space dimensions. Also, in 1977, Sivashinsky [56] derived it independently to model thermal diffusion instabilities observed in laminar flame fronts in two space dimensions. In the last two decades, many theoretical and numerical studies were devoted to the K–S equation [2,31,34,35]. On the other hand, the two-dimensional Kolmogorov flow which is the solution to the two-dimensional N–S equations that exhibit turbulent behavior is studied because of the rich of symmetries and the lack of boundary layer associated with it as compared to the three-dimensional N–S equations.

Understanding the notion of turbulence is one of the major outstanding problem in nonlinear dynamics. In this regard, few attempts were made to serve as forerunners for the understanding of turbulence. One such scenario involves the understanding of the intermittent or bursting behavior of certain measured quantities. In 1958, Kolmogorov introduced a two-dimensional flow as an example on which to study transition to turbulence. This two-dimensional Kolmogorov flow is the solution of the two-dimensional N–S equations with force $\vec{f} = ((1/Re)k^3 \cos ky, 0)$ assumed stationary and spatially biperiodic. In the forcing $\vec{f}$, Re represents the Reynolds number, and k represents the wavenumber. The case where $k = 1$ in $\vec{f}$ has been studied in [6,45,43]. In that case, the Kolmogorov flow is stable for any value of Re. However, if $k > 1$, then the flow becomes unstable. In the last decade, bifurcation of the Kolmogorov flow have been investigated in [3–5,33,46–48,51,53,54,62,59,60].

Nicolaenko and She [46–48] investigated the bursting regimes of Kolmogorov flow when $k = 8$. They linked the observed dynamics of the bursting regimes with symmetry breaking heteroclinic connections where small-scale turbulent dynamics prevail, and while large-scale dynamics are associated to hyperbolic tori. Platt et al. [51] investigated Kolmogorov flow with a different forcing of the form $\vec{f} = ((k/Re)\sin ky, 0)$ and with $k = 4$. In 1992, Armbruster et al. [3] studied quasiperiodic and intermittent regimes when $k = 8$, and in 1994, Armbruster et al. [5] analyzed most bifurcation that occur at Reynolds number from 1 to 30. In [5], it has been shown that the low-dimensional attractor for the Kolmogorov flow has a crucial component that lies in the stable eigenspace of the trivial solution. Also, a new type of bifurcation known as a gluing bifurcation has been discovered. Later on, Smaoui and Armbruster [62] have constructed a system of ODEs that mimics the dynamics of Kolmogorov flow for a given Reynold number. In 2001, an artificial neural network model has been constructed in [59] to model the unstable manifold of the bursting behavior in Kolmogorov flow, and in 2002 an autoassociative neural network was also used by the same author for dimensionality reduction of the Kolmogorov flow [60].

The paper is organized as follow: In Section 2, we present two computational methods known as the Karhunen–Loéve decomposition (K–L) and artificial neural networks (ANN). In Section 3, we apply the two computational methods mentioned above on the one-dimensional K–S equation, and present a neural network model for the dynamical behavior at the bifurcation parameter $\alpha = 17.75$. We show that the model obtained captures the heteroclinic connection observed in numerical simulations. In Section 4, we show via K–L decomposition and group symmetry that the dynamics of the two-dimensional Kolmogorov flow, consisting of eight modulated travelling waves represented in phase space by eight tori, can be modelled by artificial neural networks, and we summarize in Section 5.

2. Computational methods

2.1. The K–L decomposition

The K–L decomposition has been used in various field of applications with different names, depending on the area of applications. In image processing, it is known as the Hotelling transform [26,30], and in pattern recognition the name of principle component analysis has been used [32]. The K–L decomposition was also used under a variety of other names such as factor analysis [27], singular value decomposition (SVD) or proper orthogonal decomposition (POD) [40], empirical orthogonal functions [39], and quasiharmonic modes [12]. Lumley [40] developed the POD technique to identify coherent structures in turbulent flow. Armbruster et al. [3,5] and Smaoui [59,62] have used K–L decomposition in the analysis of the two-dimensional N–S equations. Smaoui et al. [57,61,63,64] have also used K–L decomposition in fluid flow in porous media, in the analysis of sand stones, and in the study of flames. Regardless of the different names and applications, the method which is based on second order statistical properties is essentially the same. Since the K–L decomposition is heavily used in this paper, a detailed description of the method is now presented.

First, we consider the data to be a set of (real) random vectors, which depend on space and time

$$\{X_i\}_{i=1}^{M}, \tag{1}$$

where $X = [x_1, x_2, \ldots, x_N]^{\mathrm{T}}$. Next, the mean is computed as

$$\bar{X} = \frac{1}{M}\sum_{i=1}^{M} X_i. \tag{2}$$

For convenience, an additional sequence of vectors called caricature vectors which have zero mean are computed as:

$$\hat{X}_i = X_i - \bar{X}, \quad i = 1, \ldots, M. \tag{3}$$

Two approaches are known in the literature for computing the covariance matrix. The first approach is known by the direct method, and the second approach is known by the snapshot method. In the direct method, the covariance matrix is approximated by

$$C = \frac{1}{M}\sum_{i=1}^{M} \hat{X}_i \hat{X}_i^{\mathrm{T}}. \tag{4}$$

Notice that in Eq. (4), the covariance matrix C is an $N \times N$ matrix. If N is big, this matrix can become too large for practical computation. However, using the snapshot method which was first introduced by Sirovich in 1987 (for a more detailed description of the method see [55]) makes the computation more tractable. In the snapshot method, the covariance matrix is given by

$$C_{ij} = \frac{1}{M}\langle \hat{X}_i, \hat{X}_j \rangle, \quad i, j = 1, \ldots, M. \tag{5}$$

Here $\langle ., . \rangle$ denotes the usual Euclidean inner product. Now, the matrix is $M \times M$ instead of $N \times N$, and assuming that $M \ll N$, its eigenvalues and eigenvectors can be easily computed than the one defined by the direct method.

The covariance matrix C is symmetric and positive definite, then its eigenvalues $\{\lambda_i; i=1,\ldots,M\}$ are real, and its eigenvectors $\{\phi_i; i=1,\ldots,M\}$ form an orthogonal set [66]. The orthogonal eigenfunctions of the data are defined as

$$\Psi^k = \sum_{i=1}^{M} \phi_i^{[k]} \hat{X}_i, \quad k=1,\ldots,M, \tag{6}$$

where $\phi_i^{[k]}$ is the ith component of the kth eigenvector. A characterization of the eigenfunctions is that they form an optimal basis for the expansion of a spatiotemporal data set. This basis has a smaller least square error than a representation by any other basis, i.e.,

$$u(x,t) = \hat{X} \approx \sum_{i=1}^{M} a_i(t)\Psi_i(x), \tag{7}$$

where the a_i's are the data coefficients, computed from the projection of a sample vector onto an eigenfunction

$$a_i = \left(\frac{\hat{X} \cdot \Psi^{[i]}}{\Psi^{[i]} \cdot \Psi^{[i]}} \right). \tag{8}$$

The sum of the eigenvalues of the covariance matrix is defined as the "Energy" of the data and is given by

$$E = \sum_{i=1}^{M} \lambda_i. \tag{9}$$

Each eigenfunction has an energy percentage which depends on the eigenfunction's associated eigenvalues:

$$E_k = \frac{\lambda_k}{E}. \tag{10}$$

Assuming that the eigenvalues are sorted in a descending order, then the eigenfunctions are ordered from most to least energetic. Finally, any sample vector can be reconstructed using the eigenfunctions

$$X = \bar{X} + \sum_{i=1}^{M} a_i \Psi^{[i]}. \tag{11}$$

Using only the first most energetic K eigenfunctions, an approximation of the data is constructed as

$$u(x,t) = X \approx \bar{X} + \sum_{i=1}^{K} a_i \Psi^{[i]}. \tag{12}$$

2.2. Artificial neural networks

ANN have been used in many different disciplines such as speech recognition [9,11,19], image processing, control system, artificial intelligence [1,29,49,52], human face recognition [65], fluid flow in porous media [63,64], time series [37,50,58], and dynamical systems [13,25,36,44,57,60,62]. The

greatest advantage of a neural network is its ability to model complex nonlinear relationships without any assumptions about the nature of the relationships.

The way ANN analyze data is by passing it through several simulated processors which are interconnected with synaptic-like "weight". Once several records of the data to be analyzed are collected, then the network will run through them and learn how the inputs of each record may be related to the results. Then, it continually refines itself until it can produce an accurate response when given those particular inputs. If there is any relationship between the input and the result of each record, the network should be able to create an internal mapping of weights that can accurately reproduce the expected output.

During the training procedure, the network compares its actual response with the target response and adjusts its weights in such a way to minimize the sum of the square of the error E, defined as

$$E = \frac{1}{2} \sum_{p} \sum_{k} (z_k - y_k)_p^2, \tag{13}$$

where z_k is the desired vector of the kth output node, y_k is the actual output vector of the kth output node, and the p subscript refers to the specific input vector pattern used. The weights leading into an output node k are adjusted in proportion to the difference between the actual node output and its desired output.

Among the many algorithms available for adjusting the weight matrix, back-propagation is the most popular algorithm of all neural network paradigms [9,15,22–24]. It was initially developed and announced by Paul werbors in the early 1970s in his Harvard university doctoral thesis. Back-propagation algorithms is a gradient descent or steepest descent algorithm which is based on a multilayered, feedforward topology, with supervised learning.

In the beginning of the training stage, the network uses a random generator to initialize the weight matrices and bias vector. If the sum square error E is greater than a fixed value, then the new weight adjustments are computed for all hidden layer one by one, until the synaptic weights of the input layer are reached. Since the goal of the back-propagation algorithm is to locate the global minimum in the error surface, then new training algorithms were processed to speed up training and to make sure the search process does not get stuck in local minima. Among those algorithms, Levenberg –Marquardt algorithm has proven to be efficient and reliable algorithm for computing the weight matrix. It is given by

$$\Delta w_{hk} = (J^{\mathrm{T}} J + \mu I)^{-1} J^{\mathrm{T}} e, \tag{14}$$

where μ is scalar, J is the Jacobian matrix of derivatives of each error to each weight, and e is an error vector.

After the iterative optimization procedure has converged, an explicit nonlinear map to be used for future prediction is obtained:

$$x(t+P) = f(x(t), x(t-1)), \tag{15}$$

where $x(t)$ and $x(t-1)$ are the values of the data coefficients at time t and $t-1$, respectively; $x(t+P)$ is the predicted value of the data coefficient at the next P sampling instant, and f is a set of nonlinear function representing the ANN-model.

3. The K–S equation

The K–S equation

$$\frac{\partial u}{\partial t} + v\frac{\partial^4 u}{\partial x^4} + \frac{\partial^2 u}{\partial x^2} + \frac{1}{2}\left(\frac{\partial u}{\partial x}\right)^2 = 0, \quad (x,t) \in \mathbb{R}^1 \times \mathbb{R}^+, \tag{16}$$

$$u(x,t) = u(x+L,t), \tag{17}$$

$$u(x,0) = u_0(x), \tag{18}$$

where L is a period, has first been derived in 1976 by Kuramoto and Tsuzuki [38] as a model equation for interfacial instabilities in the context of angular phase turbulence for a system of a reaction–diffusion equation that model the Belouzov–Zabotinskii reaction in three space dimensions, and independently, in 1977, by Sivashinsky [56] to model thermal diffusion instabilities observed in laminar flame fronts in two space dimensions. In the last two decades, many theoretical and numerical studies were devoted to the K–S equation [2,31,34,35].

By setting $\tilde{t} = vt/4$ and $L = 2\pi$, Eq. (16) can therefore be transformed to

$$\frac{\partial u}{\partial t} + 4\frac{\partial^4 u}{\partial x^4} + \alpha\left[\frac{\partial^2 u}{\partial x^2} + \frac{1}{2}\left(\frac{\partial u}{\partial x}\right)^2\right] = 0, \quad 0 \leqslant x \leqslant 2\pi, \tag{19}$$

$$u(x,t) = u(x+2\pi,t), \tag{20}$$

$$u(x,0) = u_0(x), \tag{21}$$

where $\alpha = 4/v$ is a bifurcation parameter, and $u_0(x)$ is 2π-periodic. Eq. (16) is equivalent to Eq. (19), but with a different time scaling. In [31], extensive numerical simulations have been analyzed for values of bifurcation parameter α between 0 and 320. There, it has been shown that as α increases, the dynamics exhibit a variety of interesting behaviors, including fixed points, travelling waves, beating waves, homoclinic and heteroclinic orbits, and chaos. For the bifurcation parameter $\alpha = 17.75$, the dynamics of the K–S equation is shown to exhibit a homoclinic cycle. Armbruster et al. [2] and Kevrekidis et al. [35] explained these dynamics in an analytical frame work.

Since the goal of this work is to obtain an artificial neural network model for the K–S equation at $\alpha = 17.75$, we numerically compute the time series solution of Eq. (19) with $u_0(x) = \sin 2x$ by decomposing $u(x,t)$ via the expansion

$$u(x,t) = \sum_{k=-\infty}^{\infty} a_k(t)\mathrm{e}^{\mathrm{i}kx}. \tag{22}$$

Using the above expansion, Eq. (19) becomes

$$\sum_{k=-\infty}^{\infty} [\dot{a}_k(t) + (4k^4 - \alpha k^2)a_k(t)]\mathrm{e}^{\mathrm{i}kx} = \frac{\alpha}{2}\left(\sum_{k=-\infty}^{\infty} k a_k(t)\mathrm{e}^{\mathrm{i}kx}\right)^2. \tag{23}$$

The Fourier coefficients a_k are found via the orthogonality relationship:

$$\int_0^{2\pi} e^{ikx} e^{-ilx}\, dx = 2\pi\delta_{kl}. \tag{24}$$

Applying the orthogonality condition and truncating the expansion results in

$$\dot{a}_l(t) = (\alpha l^2 - 4l^4)a_l(t) + \frac{\alpha}{2}\sum_{n=-N+l}^{N} n(l-n)a_{l-n}a_n, \tag{25}$$

where $-N \leqslant l \leqslant N$. Eq. (25) is solved using a pseudospectral Galerkin method where the nonlinear term is treated using a "de-aliasing" technique known as aliasing removal by truncation [14].

Fig. 1 presents two numerical simulation results obtained at $\alpha = 17.75$. Looking carefully at Fig. 1(a), we see that it consists of two laminar states: one between two bursts and the other is on the other sides of the two bursts. In phase space concept, the behavior is similar to that of a heteroclinic orbit where two fixed points, one stable and the other unstable, are connected through their stable and unstable manifolds.

The K–L procedure has been used on the numerical results obtained in Fig. 1(a) with the task of extracting the coherent structures or the most energetic eigenfunctions of the numerical data. Table 1 depicts the eigenvalue spectrum showing that 99.88% of the energy is captured by the first five eigenfunctions. The first eigenfunction captures 55.69% of the total energy; the second eigenfunction captures 31.46% of the total energy, and the remaining three accounts for 12.73% of the total energy. Fig. 2 plots these five eigenfunctions. Since the first five eigenfunctions captured 99.88% of the total energy, this suggests that the dynamical behavior presented by the heteroclinic orbit lives in a low-dimensional linear space, and that the K–L expansion provides a set of vectors that span this space. Of course, the dimension of this space may not be the intrinsic dimension where the attractors reside (see Smaoui [60]). Projections of the numerical results onto the first five eigenfunctions are given in Fig. 3. It is evident from Fig. 3 that the two bursts appear in the same position in all five data coefficients.

To obtain a neural network model for the numerical simulation results, a feedforward neural network architecture with one input layer, two hidden layers both with nonlinear sigmoidal activation function $g(x) = \tanh x$, and an output layer with linear transfer function is used. The input layer consists of five data coefficients a_i, $i = 1,\ldots,5$ at time t_n and at time t_{n-1}. The output is the predicted value of the five data coefficients at the next P sampling instant $a_i(t_n + P)$, i.e., we have the following mapping:

$$a_i(t_n + P) = f(a_i(t_n), a_i(t_{n-1})), \quad i = 1,\ldots,5, \tag{26}$$

where f is a set of nonlinear function representing the ANN-model. For $P = 1$, a sensitivity study was conducted to fine tune both the sum square error E as defined in Eq. (13) and the number of nodes in the two hidden layers. The best architecture obtained consists of 10 nodes in the two hidden layers (see Fig. 4). The network was then trained using the data coefficients given in Fig. 3. Upon convergence, that is, when the E reaches a preset bound, the weights connecting all nodes were saved, and the network is ready for testing a new set of data coefficients not included during the training stage. This new set of data coefficients shown in Fig. 5 was derived by applying K–L decomposition on another simulation result which is presented in Fig. 1(b).

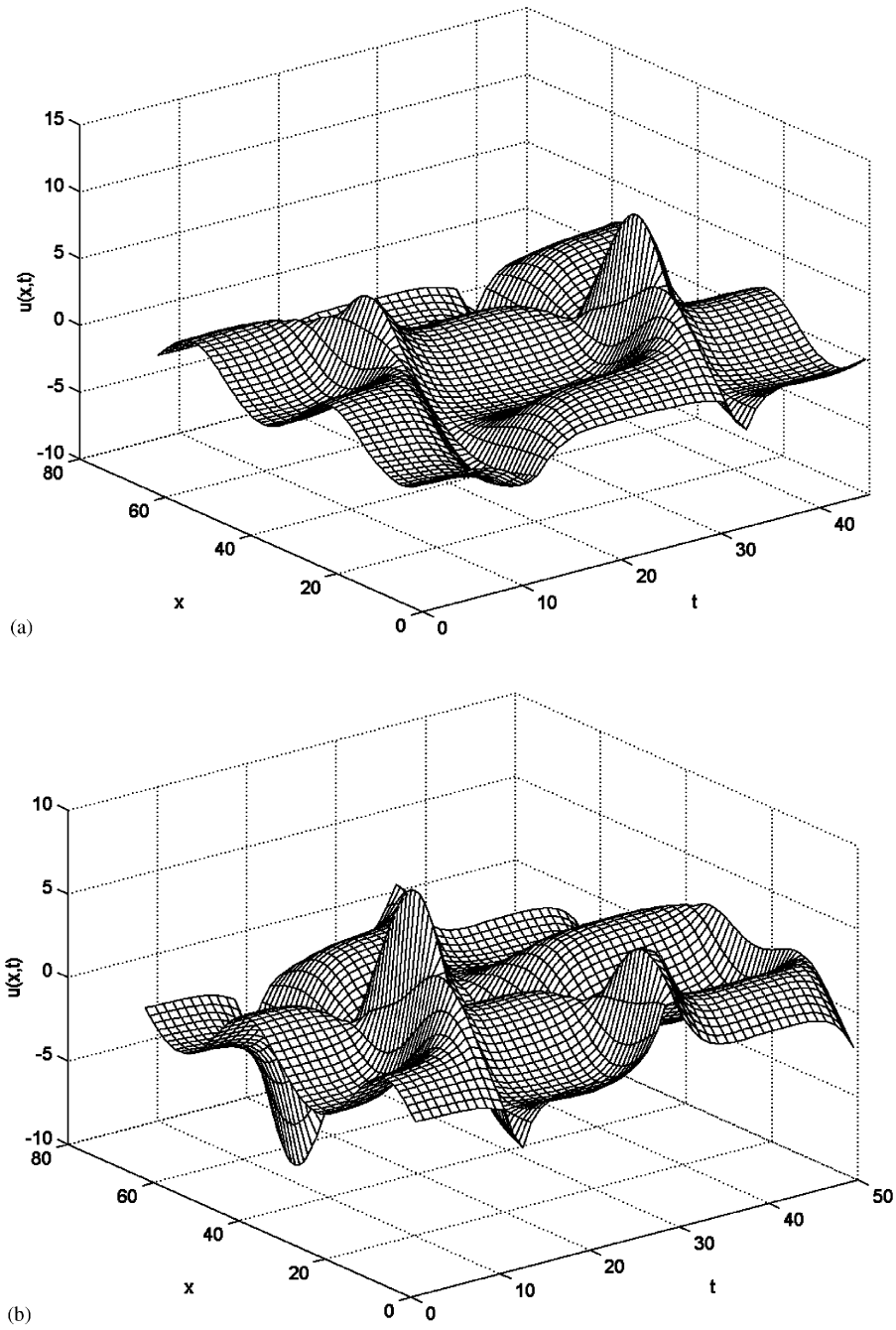

Fig. 1. (a) Numerical simulation results of the K–S equation at $\alpha = 17.75$ and with initial conditions $u_0(x) = \sin 2x$. (b) Another numerical simulation results showing different bursting events.

Table 1
Energy of the first five eigenfunctions of the K–S numerical simulation data at $\alpha = 17.75$ capturing 99.88% of the total energy

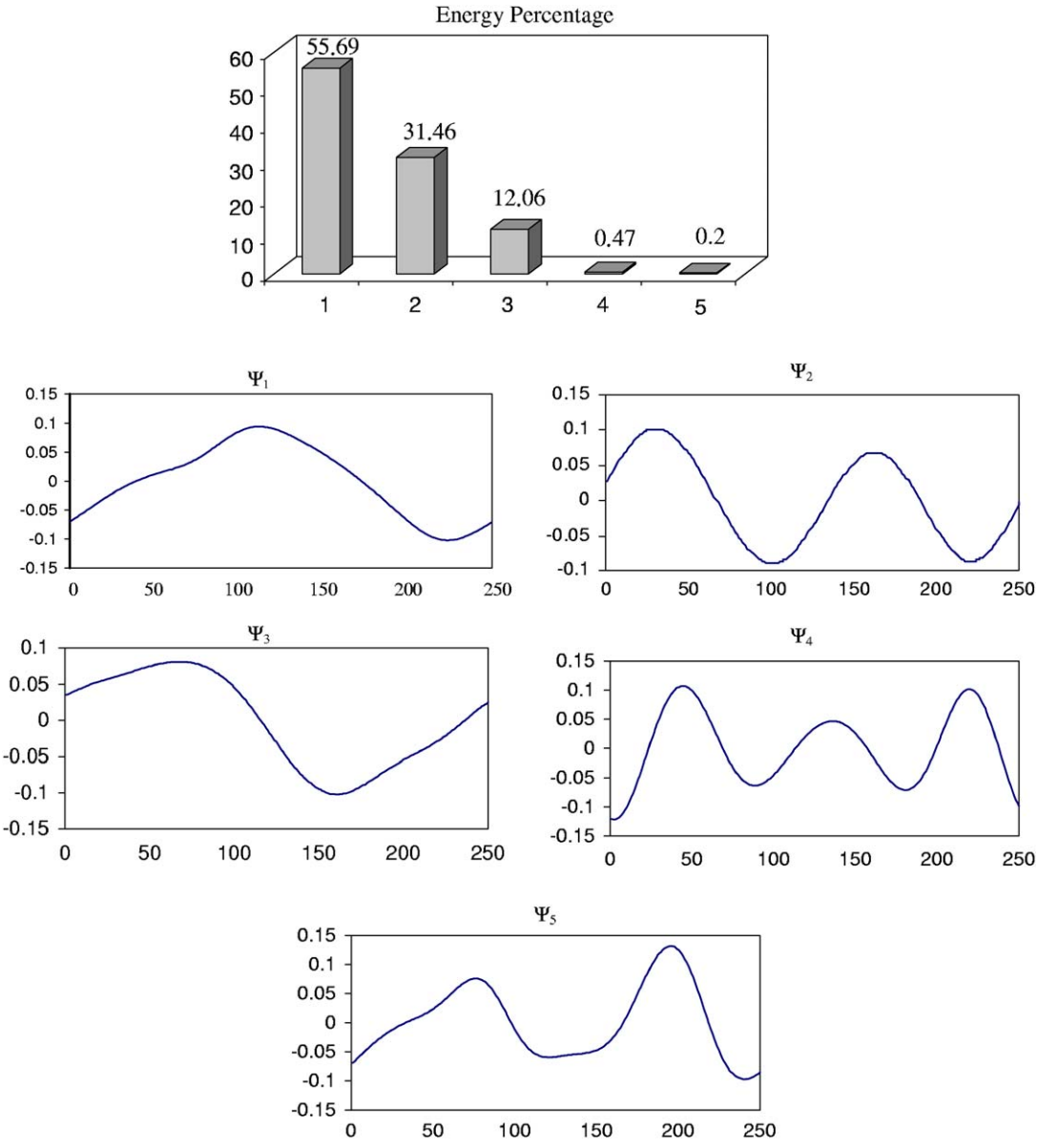

Fig. 2. The first five eigenfunctions of the K–S numerical simulation data at $\alpha = 17.75$.

Fig. 6 presents both the testing data coefficients and their neural network predictions. Comparing the original data coefficients with their neural network counterparts, one can say that the neural network model was able to capture the heteroclinic connection when $P = 1$. Since we are interested

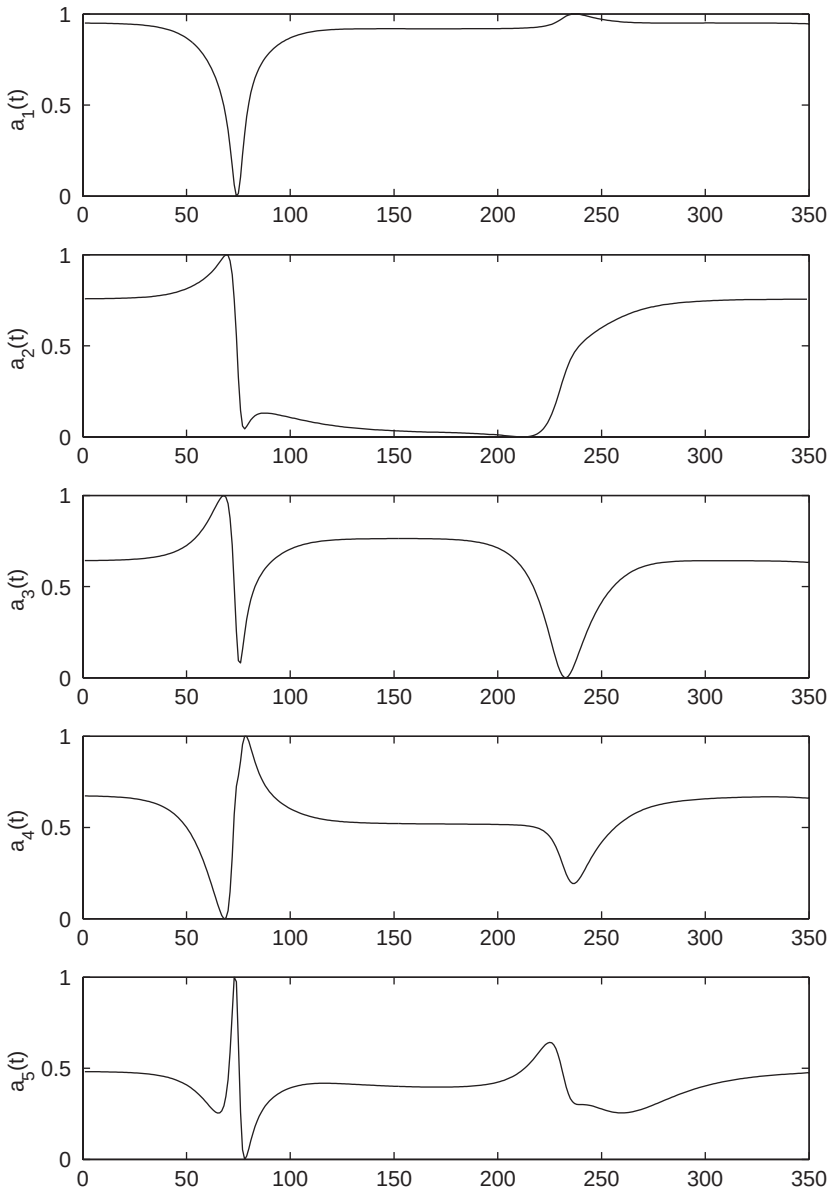

Fig. 3. The first five data coefficients of the numerical simulation data shown in Fig. 1(a) generated by K–L decomposition.

to predict the dynamical behavior for different values of P (i.e., $P=3,6$, and 9), and since for $P=1$ the neural network did not behave extremely well, then to improve the results one might include more time delays or old observations in the input layer. This approach will not be practical since it leads to a large network with a considerable number of weights to be estimated. To determine a better neural network model, we train the network using only the first data coefficient as a training data set. We realized that since the heteroclinic connection is captured by the first data coefficient,

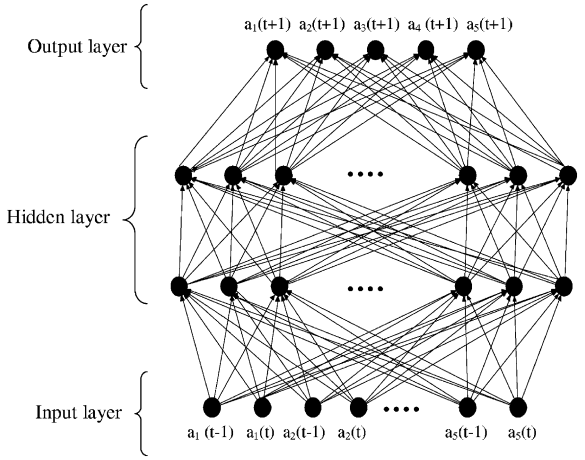

Fig. 4. The first artificial neural network architecture used for the K–S numerical simulation data.

and since the topological dimension of a heteroclinic connections is one, then one might not need the other data coefficients during the training stage. In this case, the best neural network architecture was found to consist of a 2-node input layer, one 5-node hidden layer and one-node output layer (see Fig. 7). Upon convergence, The resulting network constitutes a dynamical model that can be used to predict the other four data coefficients and those data coefficient resulted from the new numerical simulation. the model is represented by

$$a(t_{n+1}) = f(a(t_n), a(t_{n-1})) \tag{27}$$

or in terms of the saved weights and biases

$$a(t_{n+1}) = \mathbf{w}_2 g(\mathbf{w}_1 \mathbf{a}(t_n, t_{n-1}) - \boldsymbol{\theta}_1) - \boldsymbol{\theta}_2, \tag{28}$$

where $\mathbf{w}_1$ is the weight matrix for synapsis connecting the input nodes with nodes of the first hidden layer, $\mathbf{w}_2$ is the weight matrix for synapsis connecting the first hidden layer with the node at the output layer. These weights are given by

$$\mathbf{w}_1 = \begin{bmatrix} 2.3573 & -2.2035 \\ 1.9718 & 1.2895 \\ 0.317 & -0.8678 \\ 1.2571 & 0.8814 \\ -4.0324 & 0.9365 \end{bmatrix}, \tag{29}$$

$$\mathbf{w}_2 = [-0.5297 \;\; 0.0516 \;\; -1.5002 \;\; 0.1719 \;\; -0.0418]. \tag{30}$$

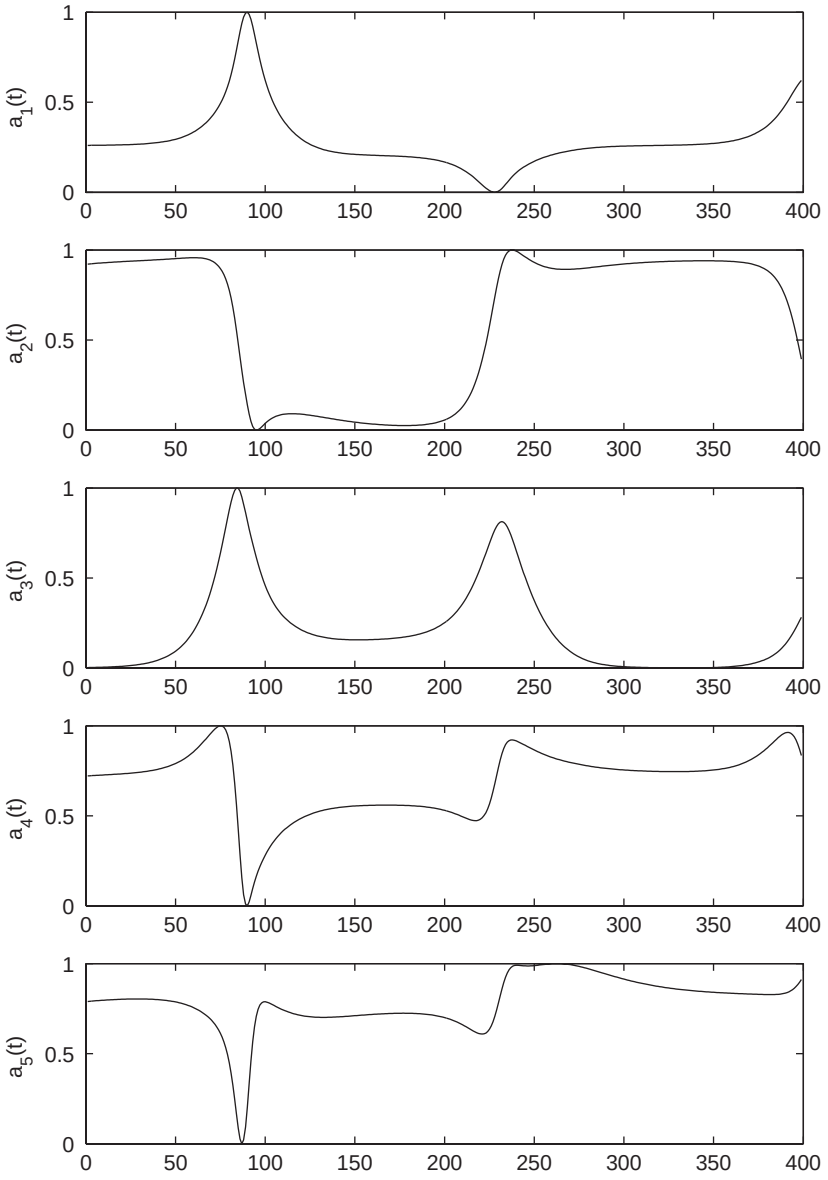

Fig. 5. The first five data coefficients of the numerical simulation data shown in Fig. 1(b) generated by K–L decomposition.

The bias vectors used for each layer $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are given by

$$\boldsymbol{\theta}_1 = \begin{bmatrix} -0.8089 \\ 0.216 \\ 0.0398 \\ -2.0482 \\ 1.0052 \end{bmatrix}, \quad \boldsymbol{\theta}_2 = [-0.1189]. \tag{31}$$

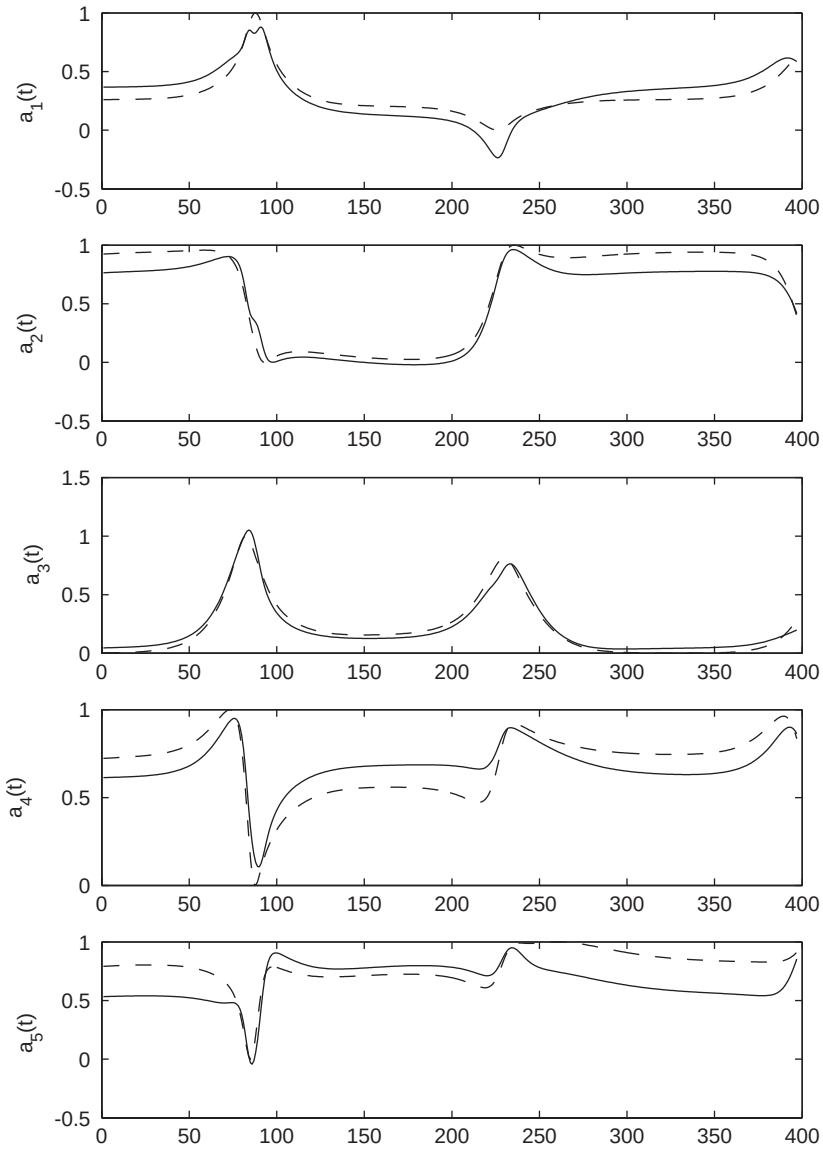

Fig. 6. Short-term prediction of the ANN model presented by Fig. 4 (solid) versus the original time series data coefficients (dashed).

$$\mathbf{a}(t_n, t_{n-1}) = \begin{pmatrix} a_1(t_n) \\ a_1(t_{n-1}) \end{pmatrix}$$

and $a(t_{n+1}) = a_1(t_{n+1})$, are the input and output vectors that consist of the values of the data coefficients at t_{n-1}, t_n and t_{n+1}, respectively.

Fig. 8 is a plot of the five data coefficients used for testing the neural network model obtained for $P = 1$. It should be noted again that the network was trained using only the first data coefficient.

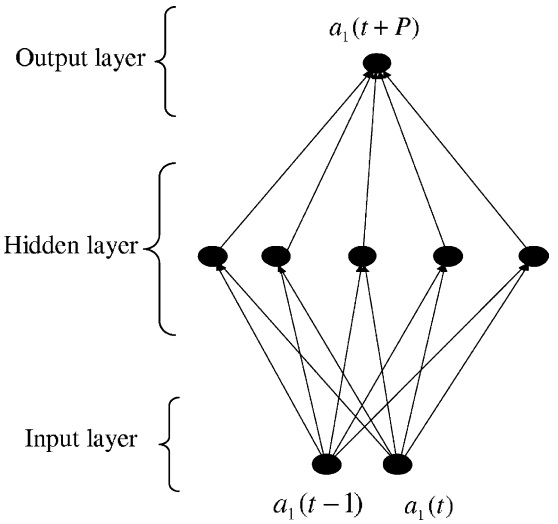

Fig. 7. The second artificial neural network architecture used for modelling the K–S equation for $P = 1, 3$, and 6.

Testing the model on the remaining four data coefficients shows an excellent agreement. To further test our model, we used the second set of data coefficient presented in Fig. 5. Fig. 9 shows that the original data coefficients agree very well with those predicted by the model.

The architecture given in Fig. 7 was also used to predict the data coefficients when $P = 3; 6$. However, for $P = 9$, the optimal neural network architecture was found to consist of a 2-node input layer, one 10-node hidden layer and a 1-node output layer. It should be noted that a prediction at P time steps into the future past the last observed point $a(t)$ will be made using observed data at times: $a(t)$ and $a(t-1)$. That is, a prediction at P time steps into the future is made by placing previously predicted values in the input layer. For $P = 3$ and 6, the model behaves very well in capturing the dynamics of the attractors; however, when $P = 9$, the model behavior degrades as P increase (see Fig. 9). Of course, this is because previously predicted values (made with some errors) are used to make subsequent prediction; therefore the errors get magnified upon iterations.

So far, a neural network model was presented to capture the dynamics of the one-dimensional K–S equation at $\alpha = 17.75$. The model was successful in predicting the dynamical behavior of the heteroclinic connection at different time steps. However, we showed that as the number of time steps increases, the model behavior degrades. In the next section, we analyze a more complicated dynamical behavior than the one presented in this section. The dynamical behavior is presented by eight different tori representing quasiperiodic behaviors of Kolmogorov flow at a bifurcation parameter $Re = 14.0$.

4. The two-dimensional N–S equation

The two-dimensional N–S equations with periodic boundary conditions in two directions $0 \leqslant x, y \leqslant 2\pi$ are given by

$$\vec{u_t} + (\vec{u}.\nabla)\vec{u} + \nabla p = v\nabla^2\vec{u} + \vec{f}, \tag{32}$$

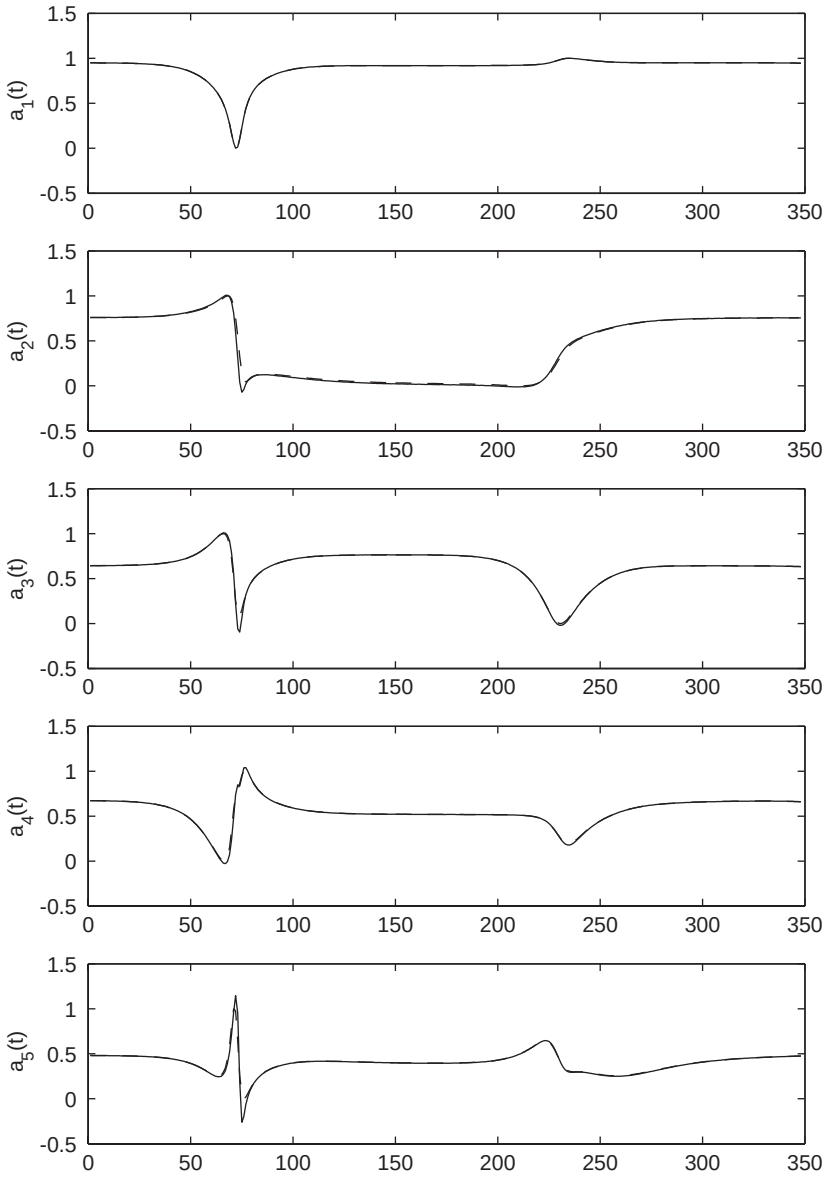

Fig. 8. Short-term ANN prediction of the first set of five data coefficients (solid) versus the original data coefficients (dashed).

$$\nabla \cdot \vec{u} = 0, \tag{33}$$

with a force $\vec{f}$ in the x, y coordinates, kinematic viscosity $v = 1/Re$ and pressure p. In 1958, Kolmogorov introduced a two-dimensional Kolmogorov flow as an example on which to study transition to turbulence with $\vec{u} = (4\cos 4y, 0)$ usually called the "the basic Kolmogorov flow". This flow is the solution to the N–S equations with force $\vec{f} = (k^3 v \cos 4y, 0)$, which is assumed

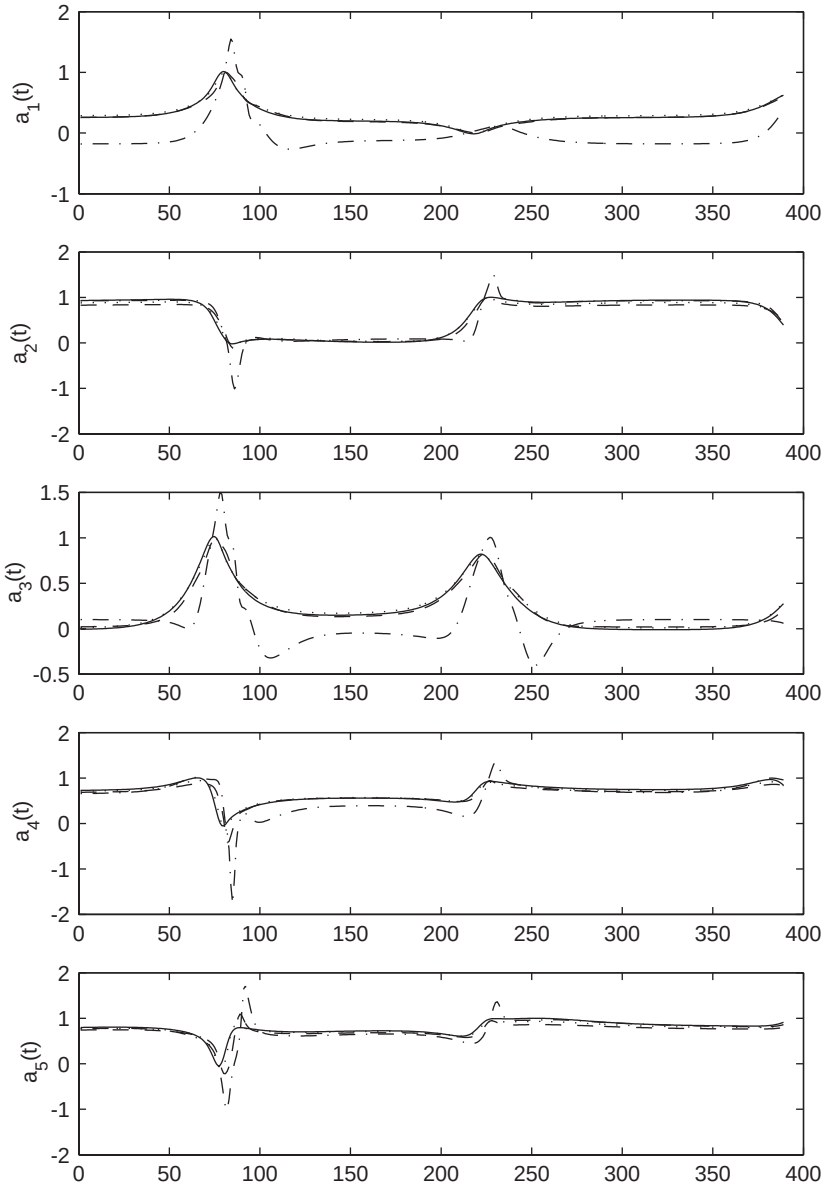

Fig. 9. Neural network prediction of the second set of data coefficients when $P = 1$ (solid line); $P = 3$ (point line); $P = 6$ (dashed line); and $P = 9$ (dashed-point line).

stationary and spatially biperiodic. Recently, large numerical simulations of this flow have revealed a wealth of bifurcation to many different attractors including: steady states, periodic, quasiperiodic and strange or chaotic attractors [4–6,33,46–48,51,53,54,58,59,67]. In this section, we are interested in the quasiperiodic behaviors at $Re = 1/v = 14.0$ and with a wavenumber $k = 4$.

If we let $\vec{u} = (u_1, u_2)$ and $\vec{f} = (f_1, f_2)$ in Eq. (32), the x and y components of $\vec{u}$ become

$$\frac{\partial u_1}{\partial t} + u_1 \frac{\partial u_1}{\partial x} + u_2 \frac{\partial u_1}{\partial y} + \frac{\partial p}{\partial x} = v\left(\frac{\partial^2 u_1}{\partial x^2} + \frac{\partial^2 u_1}{\partial y^2}\right) + f_1, \tag{34}$$

$$\frac{\partial u_2}{\partial t} + u_1 \frac{\partial u_2}{\partial x} + u_2 \frac{\partial u_2}{\partial y} + \frac{\partial p}{\partial y} = v\left(\frac{\partial^2 u_2}{\partial x^2} + \frac{\partial^2 u_2}{\partial y^2}\right) + f_2. \tag{35}$$

Also, by letting

$$u_1 = \frac{\partial \phi}{\partial y}, \tag{36}$$

$$u_2 = -\frac{\partial \phi}{\partial x} \tag{37}$$

and taking the partial derivative with respect to y of Eq. (34), the partial derivative with respect to x of Eq. (35), and subtracting the two equations, we get

$$\frac{\partial \Delta\phi}{\partial t} + \frac{\partial}{\partial x}\left(\Delta\phi \frac{\partial \phi}{\partial y}\right) - \frac{\partial}{\partial y}\left(\Delta\phi \frac{\partial \phi}{\partial x}\right) = v\Delta^2\phi + \frac{\partial f_1}{\partial y} - \frac{\partial f_2}{\partial x}. \tag{38}$$

To analyze the stability of the basic flow equation, we perturb it as

$$\phi = \phi' + \sin 4y. \tag{39}$$

Then Eq. (38) reduces to

$$v\Delta^2\phi' = \frac{\partial \Delta\phi'}{\partial t} + \frac{\partial}{\partial x}\left((\Delta\phi' - 16\sin 4y)\left(\frac{\partial \phi'}{\partial y} + 4\cos 4y\right)\right) - \frac{\partial}{\partial y}\left((\Delta\phi' - 16\sin 4y)\frac{\partial \phi'}{\partial x}\right) \tag{40}$$

or

$$\frac{\partial \Delta\phi'}{\partial t} = \Delta^2\phi' - Re\left[\frac{\partial}{\partial x}\left(\Delta\phi' \frac{\partial \phi'}{\partial y}\right) - \frac{\partial}{\partial y}\left(\Delta\phi' \frac{\partial \phi'}{\partial x}\right)\right] - 4Re\frac{\partial}{\partial x}[\Delta\phi' + 16\Delta\phi']\cos 4y, \tag{41}$$

where $Re = 1/v$ is the Reynolds number and $\tilde{t} = vt$. As the Reynolds number is increased further, the Kolmogorov flow, in general, seems to acquire more and more small scale structures without gaining much additional temporal structure [5]. Applying the inverse Laplacian on Eq. (41), we obtain

$$\frac{\partial \phi'}{\partial t} = \nabla^{-1}\left[\Delta^2\phi' - Re\left[\frac{\partial}{\partial x}\left(\Delta\phi' \frac{\partial \phi'}{\partial y}\right) - \frac{\partial}{\partial y}\left(\Delta\phi' \frac{\partial \phi'}{\partial x}\right)\right] - 4Re\frac{\partial}{\partial x}[\Delta\phi' + 16\Delta\phi']\cos 4y\right]. \tag{42}$$

Eq. (42) is called the stream function equation. It can also be written in terms of vorticity $\omega = -\Delta\phi$.

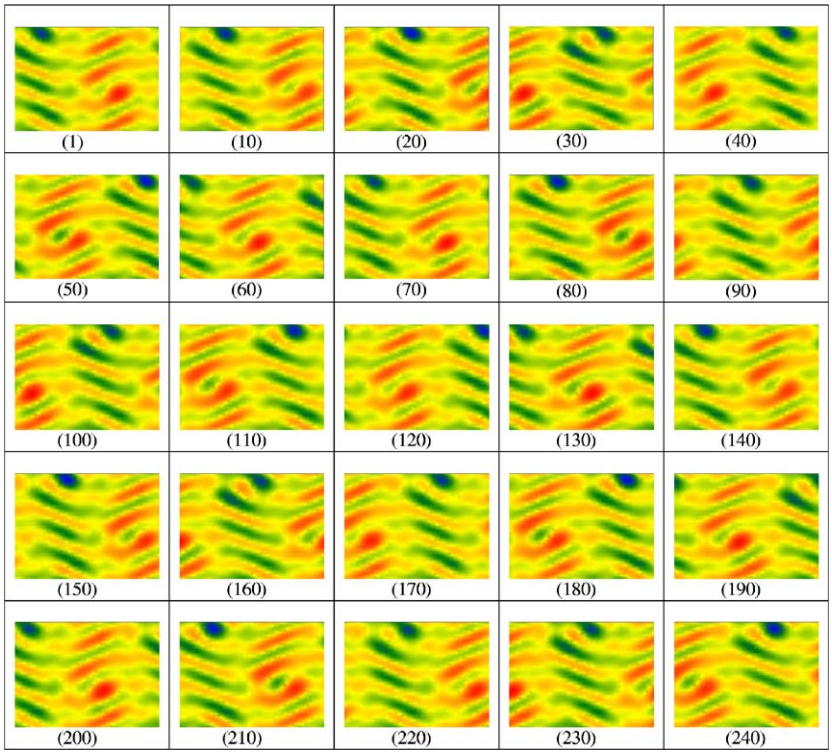

Fig. 10. Twenty-five snapshots of the vorticity at different time when $Re = 14$.

Numerical studies have shown that Kolmogorov flow exhibits a remarkable sequence of symmetry breaking bifurcation leading to a chaotic state. While numerical studies have revealed remarkable behaviors [3,5,59], the aim of this study is to construct an artificial neural network model for the quasiperiodic behavior shown in Fig. 10, by exploiting as much as possible the symmetries involved in the problem. Fig. 10 shows the numerical simulation of Eq. (42) presented in terms of the vorticity formulation.

4.1. The symmetries of the two-dimensional N–S equations

The idea of symmetry was first introduced by Sirovich [55] who suggested the use of symmetry operation to enlarge the available data set in order to get better averaging behavior. Later on, Aubry et al. [7] have shown that symmetrizing a data set is a necessary condition to achieve a representation of the global phase space of the resulting equivariant Galerkin system. Dellnitz et al. [18] determined the symmetry properties of the K–L operation with respect to the symmetry of an attractor in phase space. Berkooz and Titi [10] and Smaoui and Armburster [62] showed that applying a symmetry group on the eigenfunctions will be equivalent to applying the symmetry group on an actual realization.

Let us first describe the symmetries of the equation. The symmetries of the Kolmogorov flow are generated by the following transformations:

$$T_\xi : x \to x + \xi,$$
$$r : (x, y) \to \left(-x, -y + \frac{\pi}{4}\right),$$
$$s : (y, \phi) \to (-y, -\phi),$$
$$t : y \to y + \frac{\pi}{2}.$$

The transformations $(rs)^2$ is equivalent to the transformation t. This is shown below

$$rs : (x, y, \phi) \xrightarrow{s} (x, -y, -\phi) \xrightarrow{r} \left(-x, y + \frac{\pi}{4}, -\phi\right), \tag{43}$$

$$(rs)^2 : (x, y, \phi) \xrightarrow{rs} \left(-x, y + \frac{\pi}{4}, -\phi\right) \xrightarrow{rs} \left(x, y + \frac{\pi}{2}, \phi\right). \tag{44}$$

Thus, $(rs)^2 = t$, and $(rs)^8 = I_d$. Also, notice that r is a rotation by $\pi/4$ in the x, y-plane. The complete symmetry group of the Kolmogorov flow with periodic boundary conditions is then the semidirect product group $D_8 \dot{+} \mathrm{SO}(2)$, where

$$D_8 = \{I_d, r, rt, rt^2, rt^3, t, t^2, t^3\}. \tag{45}$$

In this section, we apply the D_8 symmetry group on both the eigenfunctions and the data set. First, applying the K–L decomposition on the vorticity numerical simulation data presented in Fig. 10 yields a set of 50 energetic eigenfunctions capturing 99.99% of the energy (see Fig. 11). Those eigenfunctions correspond to one laminar state described by a modulated travelling wave at $Re = 14.0$. Next, we act with the whole symmetry group D_8 on this set of eigenfunctions and obtain a set of 400 eigenfunctions. Since the eigenfunctions in this new set are not orthogonal, we use the theory presented in [62] to orthogonalize them (see Fig. 12). Second, instead of numerically integrating Eq. (42) to look for the eight modulated travelling waves obtained each at different initial conditions, we can act with the D_8 group on the numerical simulation data of one of the modulated travelling wave presented in Fig. 10. Then, the symmetrized numerical simulation data is projected onto the set of symmetrized eigenfunctions using Eq. (8) to obtain a set of data coefficients. Fig. 13 shows a plot of the first four data coefficients and Fig. 14 presents the fourth data coefficient a_4 versus the third data coefficient a_3 showing the eight modulated travelling waves earlier described. Inspection shows that the third and fourth data coefficients of the first modulated travelling wave MTW_1 is related with certain symmetry with the sixth, seventh and eighth modulated travelling waves $\mathrm{MTW}_{6,7,8}$. Also, the second modulated travelling wave MTW_2 is related by a different symmetry with the third, fourth and fifth modulated travelling waves $\mathrm{MTW}_{3,4,5}$ (see Table 2). These findings will help us reduce the data set needed for training a neural network.

4.2. A smart neural network model for the quasiperiodic regime

The first approach used to construct a neural network model consists of a combination of two neural networks in series, where one is used for classification purposes, and the other one is used for

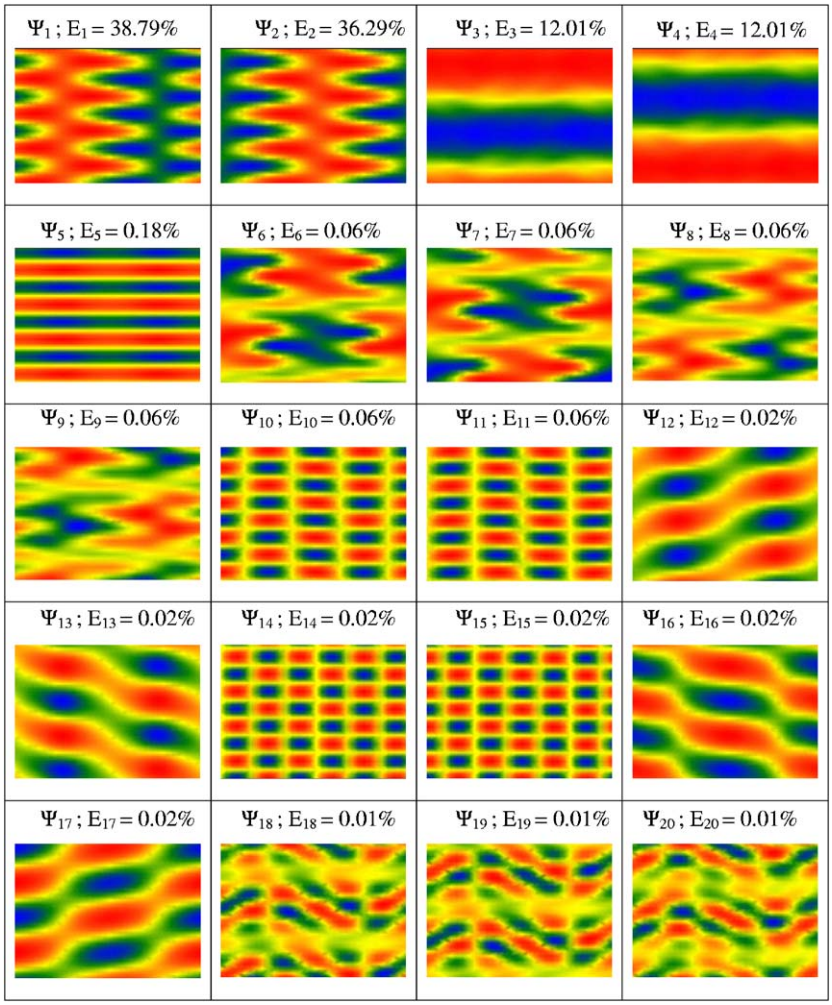

Fig. 11. The first 20 most energetic eigenfunctions and their associated energies for the vorticity data.

prediction. Fig. 15 is a neural network architecture used for classification with a 4-node input layer consisting of the four data coefficients, a 10-nodes hidden layer, and a one-node output layer that consists of the class rank of the MTW. The classification decision is based on a linear combination of the output of the activation function for each hidden layer node. The network is trained and tested using the four data coefficients of the eight modulated travelling waves. For a given set of data coefficients, the network is able to classify the set to one of the eight modulated travelling waves. Once the classification is done, then a separate network is constructed for predicting the four data coefficients at the next instant of time (i.e., $a_i(t+1),\ i=1,\ldots,4$ are predicted using actual data coefficients: $a_i(t); a_i(t-1),\ i=1,\ldots,4$) (see Fig. 16). In this case, eight different models were constructed each corresponding to one of the eight modulated travelling waves.

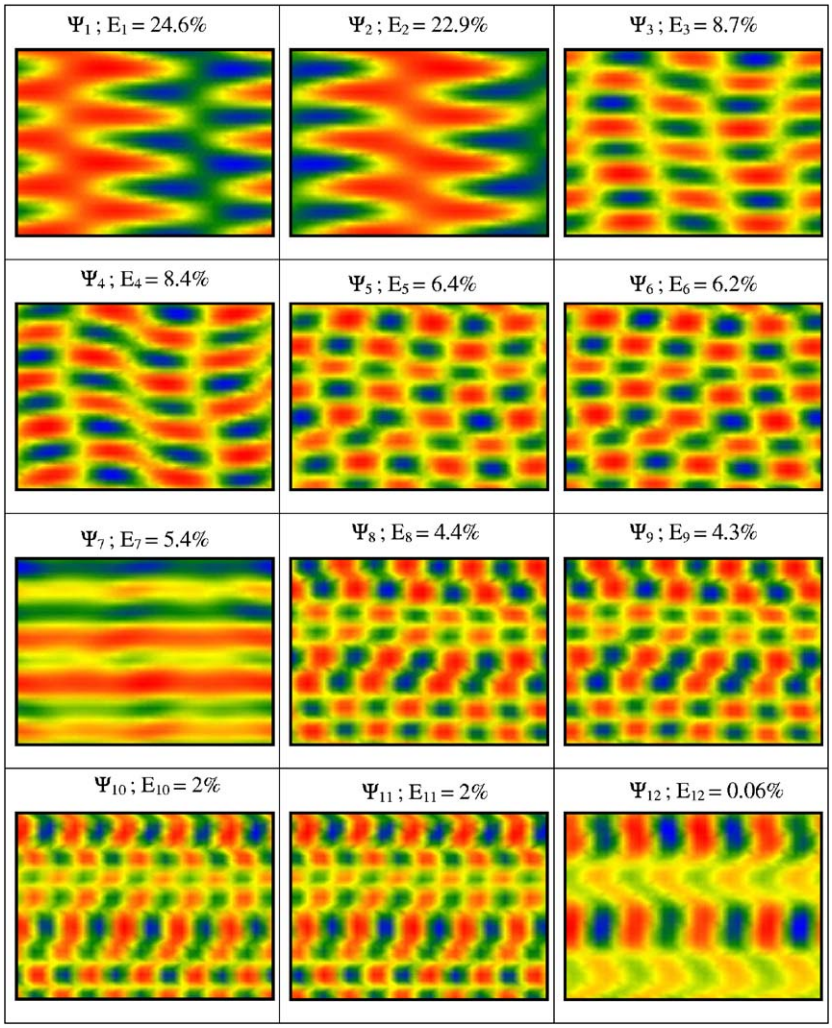

Fig. 12. The first 12 most energetic eigenfunctions and their associated energies for the symmetrized set of eigenfunctions presented in Fig. 11.

The second approach used to construct a neural network model for the quasiperiodic behavior is to exploit the symmetry observed in the data coefficients. Since the data coefficients of the modulated travelling waves $MTW_{1,6,7,8}$ and $MTW_{2,3,4,5}$ are each related with certain symmetries (see Table 2), then in principle, one can construct two neural network architectures similar to the one in Fig. 16. Training each network separately by making use of the symmetries will reduce the number of networks needed from 8 to 2. Therefore, instead of using eight different models for the eight modulated travelling waves as stated above, we construct only two. It should be noted that the network used for classifications will still be used here before applying the two networks for predictions.

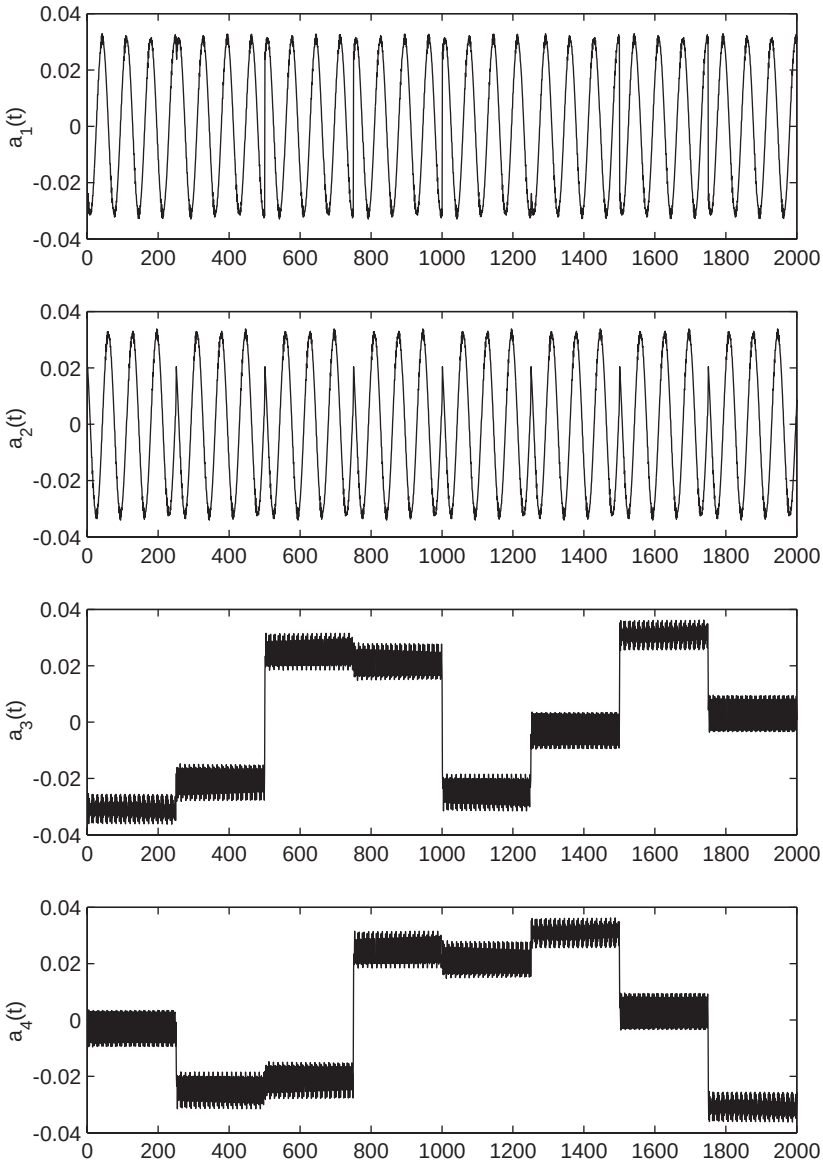

Fig. 13. The first four K–L data coefficients of the symmetrized data obtained by projecting the data onto 20 symmetrized eigenfunctions.

The third smart approach consists of applying both the symmetries of the two-dimensional N–S equations, and the symmetries observed in the data coefficients. That is, training only one network with both sets of data coefficients associated with the modulated travelling waves MTW_1 and MTW_2. Of course, the classification network is still needed to determine the right symmetry before making any prediction. A neural network with an architecture of eight nodes in the input layers, 10 nodes in

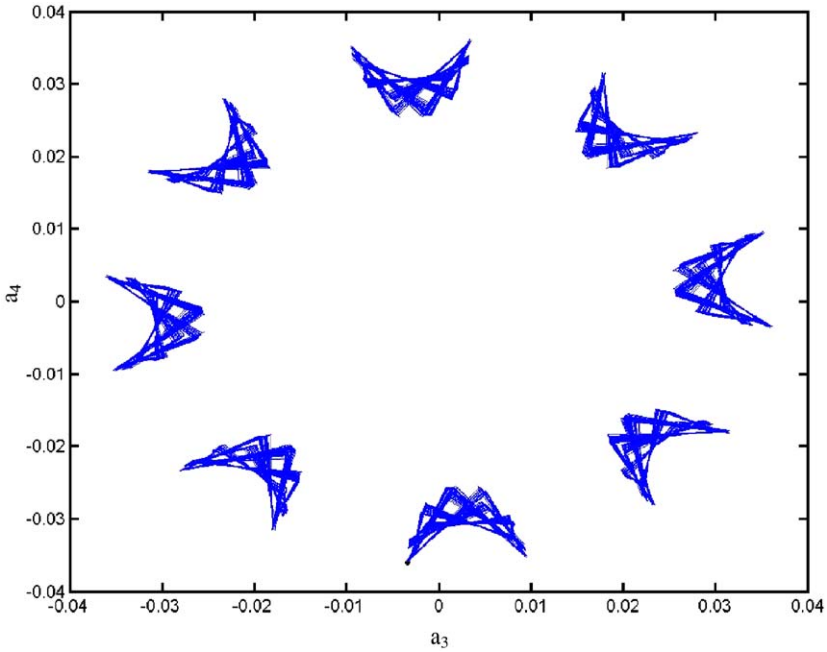

Fig. 14. The fourth K–L data coefficients a_4 versus the third K–L data coefficients a_3 of the symmetrized vorticity simulation data.

Table 2
The symmetries of modulated travelling waves (MTW)

Modulated travelling waves (MTW)		Symmetrized data coefficients
From	To	
MTW_3	MTW_2	$a_{3-MTW_3} = -a_{4-MTW_2}$ $a_{4-MTW_3} = a_{3-MTW_2}$
MTW_4	MTW_2	$a_{3-MTW_4} = -a_{3-MTW_2}$ $a_{4-MTW_4} = -a_{4-MTW_2}$
MTW_5	MTW_2	$a_{3-MTW_5} = a_{4-MTW_2}$ $a_{4-MTW_5} = a_{3-MTW_2}$
MTW_6	MTW_1	$a_{3-MTW_6} = a_{4-MTW_1}$ $a_{4-MTW_6} = -a_{3-MTW_1}$
MTW_7	MTW_1	$a_{3-MTW_7} = -a_{3-MTW_1}$ $a_{4-MTW_7} = -a_{4-MTW_1}$
MTW_8	MTW_1	$a_{3-MTW_8} = -a_{4-MTW_1}$ $a_{4-MTW_8} = a_{3-MTW_1}$

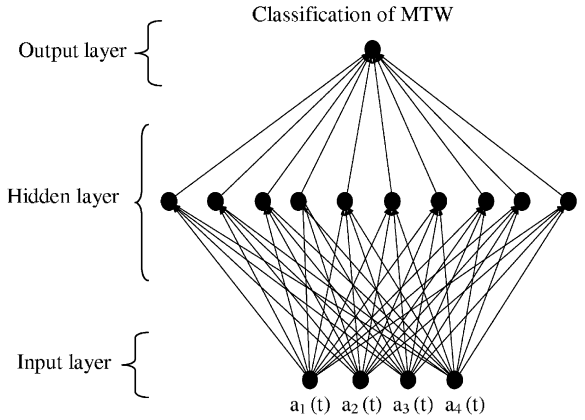

Fig. 15. A neural network architecture used for classification of the modulated travelling wave (MTW).

the hidden layers, and 4 in the output layers is used. Fig. 17 is the combined set of data coefficients for MTW_1 and MTW_2 used during the training stage. Figs. 18 and 19 show the excellent agreement between the original time series of the testing data coefficients of MTW_3 and MTW_8, respectively. The data coefficients corresponding to all eight modulated travelling waves were also tested by the model and the results are successful (see Fig. 20). The figure plots the fourth coefficient versus the third coefficient for both the original data and the predicted ones. The model obtained is now described as

$$\mathbf{a}(t_{n+1}) = f(\mathbf{a}(t_n), \mathbf{a}(t_{n-1})) \tag{46}$$

or in terms of the saved weights and biases as

$$\mathbf{a}(t_{n+1}) = \mathbf{w}_2 g(\mathbf{w}_1 \mathbf{a}(t_n, t_{n-1}) - \boldsymbol{\theta}_1) - \boldsymbol{\theta}_2, \tag{47}$$

where $\mathbf{w}_1$ is the weight matrix for synapsis connecting the input nodes with nodes of the first hidden layer, $\mathbf{w}_2$ is the weight matrix for synapsis connecting the first hidden layer with the node at the output layer. These weights are given by

$$\mathbf{w}_1 = \begin{bmatrix} 6.7131 & -4.6492 & -1.0658 & 1.8492 & -2.3588 & -34.834 & -31.9507 & 4.7008 \\ -5.0953 & -1.2234 & 4.4367 & 3.8324 & -38.0005 & 9.7642 & -9.1005 & -3.2881 \\ -6.5099 & 5.2293 & -2.6291 & 1.5278 & 39.3737 & 49.6235 & 4.1375 & -31.8184 \\ -6.192 & 3.9236 & -12.5531 & -0.7569 & -36.9474 & -22.3905 & 20.8317 & 10.745 \\ 16.4098 & -6.2891 & 6.4674 & 5.7389 & 19.3025 & 9.8594 & -12.0814 & -18.4667 \\ -3.3652 & 2.0109 & -4.3116 & 0.8604 & 49.0396 & 23.2479 & -30.1706 & 4.0732 \\ 4.9522 & 5.473 & 3.4735 & 1.2519 & -9.7755 & -8.6596 & 4.9247 & 17.056 \\ -1.7921 & 1.2112 & 3.28 & -2.9905 & 15.4324 & -10.1373 & 47.4068 & 42.7243 \\ 10.2125 & 9.8841 & -2.802 & 1.092 & 23.3546 & 29.8509 & -12.5327 & 8.4018 \\ -1.0126 & -3.3594 & 4.0767 & 0.2776 & 40.5062 & -27.4617 & -7.2383 & 13.7769 \end{bmatrix}, \tag{48}$$

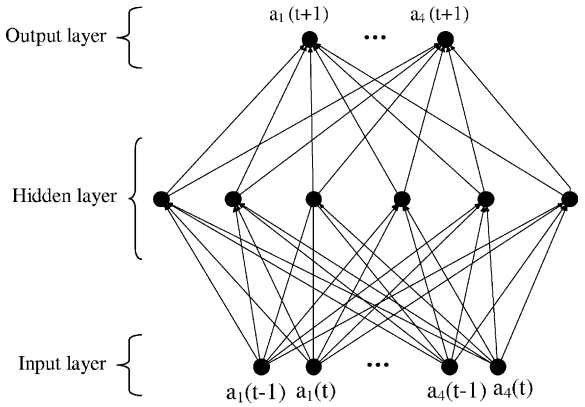

Fig. 16. A neural network architecture used for short-term prediction of the four data coefficients of the symmetrized system.

$$\mathbf{w}_2 = \begin{bmatrix} 0.0212 & -0.0433 & -0.0122 & -0.042 & 0.0082 & -0.0233 & 0.0287 & 0.0055 & 0.0107 & -0.0409 \\ -0.0201 & 0.0344 & -0.005 & -0.0302 & 0.0334 & -0.0129 & 0.0318 & 0.0034 & -0.0083 & 0.0281 \\ -0.0083 & -0.0119 & 0.0024 & 0.0053 & -0.0004 & -0.0311 & 0.0049 & -0.0355 & -0.0031 & 0.016 \\ 0.0501 & 0.0111 & 0.0469 & -0.0027 & -0.0009 & 0.0054 & -0.0128 & 0.0497 & 0.0017 & -0.0174 \end{bmatrix}. \tag{49}$$

The bias vectors used for each layer $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are given by

$$\boldsymbol{\theta}_1 = \begin{bmatrix} -1.2078 \\ -0.7394 \\ 0.4681 \\ -0.2382 \\ 1.0726 \\ 0.7741 \\ -0.7335 \\ 1.8313 \\ 0.9471 \\ 0.4497 \end{bmatrix}, \quad \boldsymbol{\theta}_2 = \begin{bmatrix} 0.0171 \\ -0.0031 \\ -0.0288 \\ 0.0032 \end{bmatrix}, \tag{50}$$

$$\mathbf{a}(t_{n-1}) = \begin{pmatrix} a_1(t_{n-1}) \\ a_2(t_{n-1}) \\ a_3(t_{n-1}) \\ a_4(t_{n-1}) \end{pmatrix}, \quad \mathbf{a}(t_n) = \begin{pmatrix} a_1(t_n) \\ a_2(t_n) \\ a_3(t_n) \\ a_4(t_n) \end{pmatrix} \quad \text{and} \quad \mathbf{a}(t_{n+1}) = \begin{pmatrix} a_1(t_{n+1}) \\ a_2(t_{n+1}) \\ a_3(t_{n+1}) \\ a_4(t_{n+1}) \end{pmatrix}$$

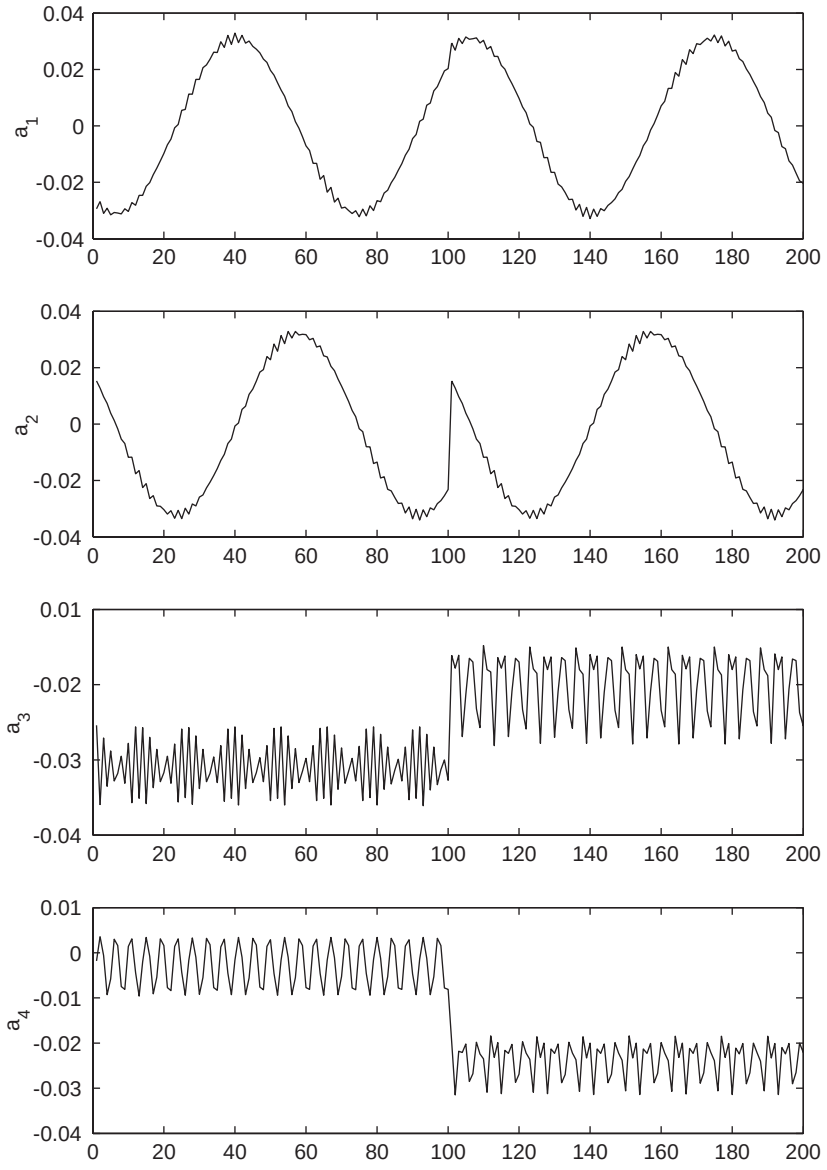

Fig. 17. The four data coefficients of two of the eight modulated travelling waves MTW_1 and MTW_2 used in training the neural network.

are the input and output vectors that consist of the values of the data coefficients at t_{n-1}, t_n and t_{n+1}, respectively.

5. Conclusions

A combination of K–L decomposition and artificial neural networks was used to model the K–S equation and the two-dimensional N–S equation. For the (K–S) equation, numerical solutions using

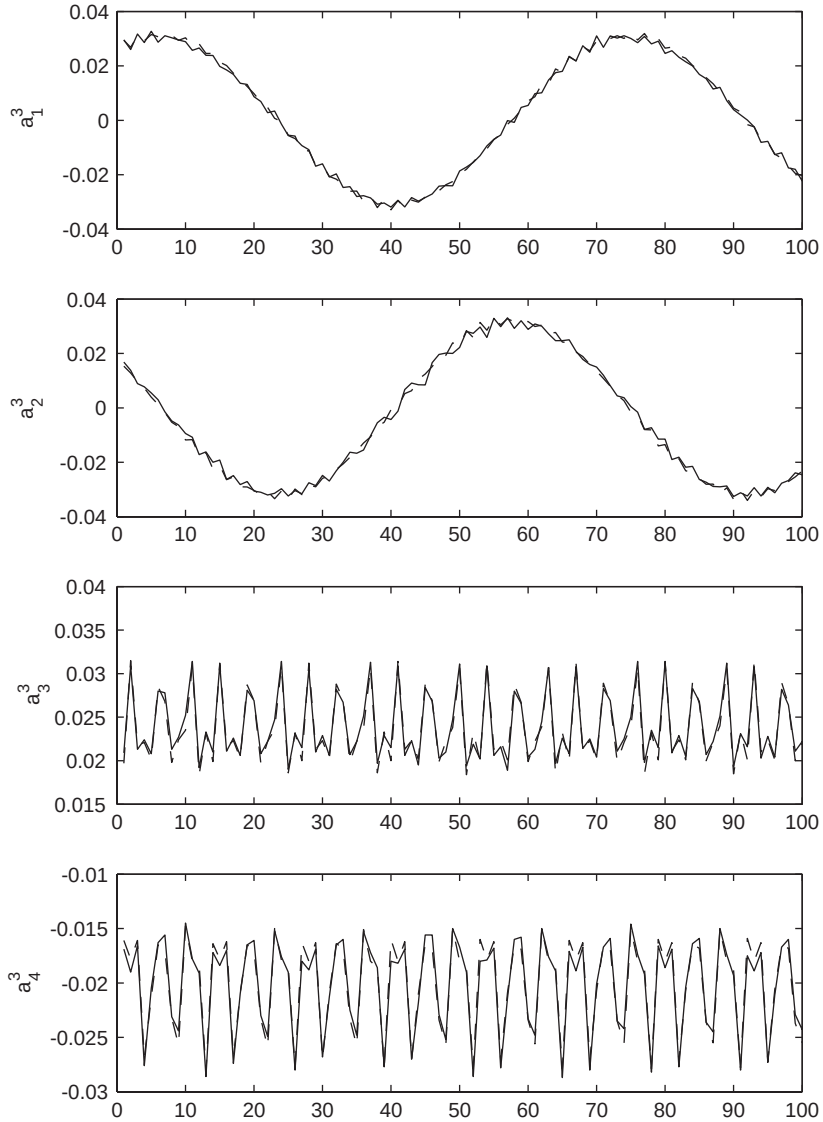

Fig. 18. Short-term neural network prediction of the four data coefficients of MTW_3 by using the smart neural network model.

pseudospectral techniques were obtained at a bifurcation parameter $\alpha = 17.75$. K–L decomposition were thus applied on the numerical simulation data to extract coherent structures of the dynamical behavior represented by a heteroclinic connection. Once the coherent structures were derived, data coefficients were obtained by projecting the numerical data onto the most energetic coherent structures. ANN was then used to model and predict P time steps into the future the dynamical behavior at $\alpha = 17.75$ for different values of P (i.e., $P = 3, 6$, and 9). We found that the neural network model

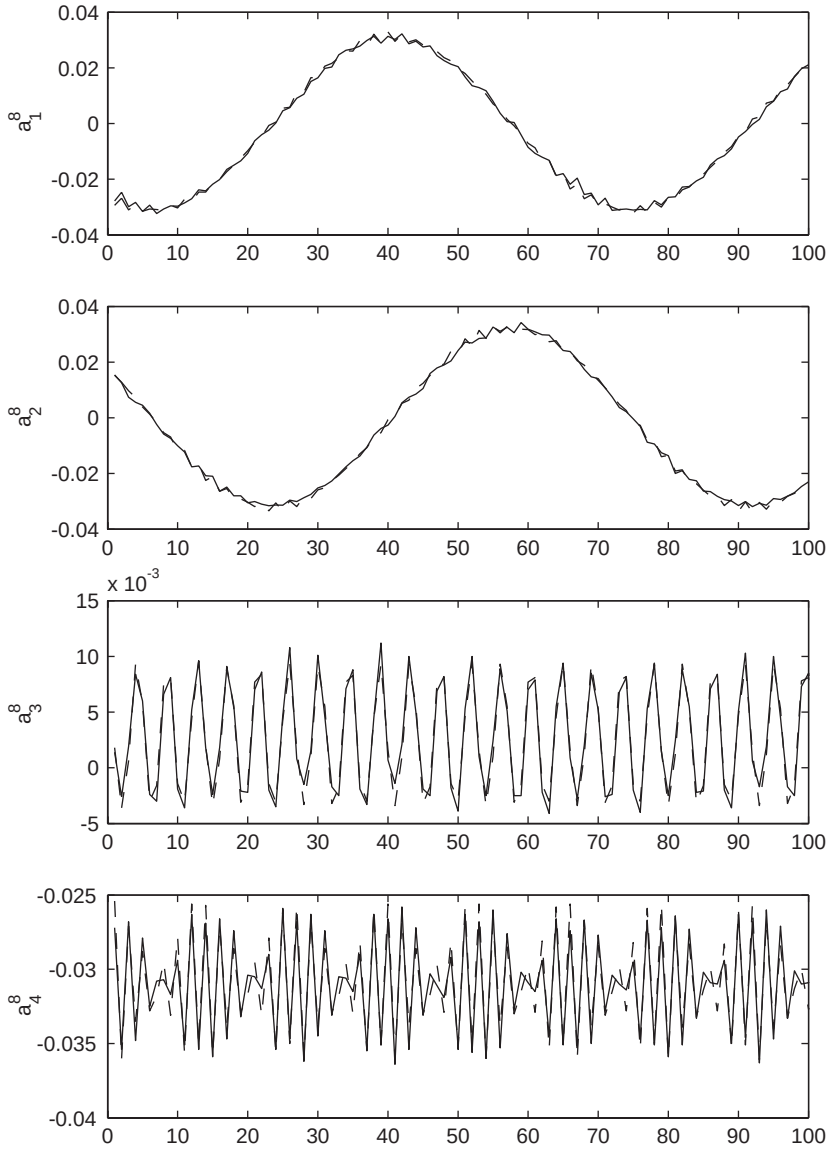

Fig. 19. Short-term neural prediction of the four data coefficients of MTW_8 by using the smart neural network model.

was able to capture the underlying dynamics, and observe that as P increases, the model behavior degrades.

As for the two-dimensional N–S equation, we analyzed a quasiperiodic behavior at $Re = 14.0$ and for wave number $k = 4$ represented in phase space by a torus. We used the D_8 group symmetries observed in the two-dimensional N–S equations to obtain eight different tori. K–L decomposition was then used on the numerical simulation data of one of the torus to extract the most energetic eigenfunctions. Those eigenfunctions were symmetrized to capture the coherent structures that span

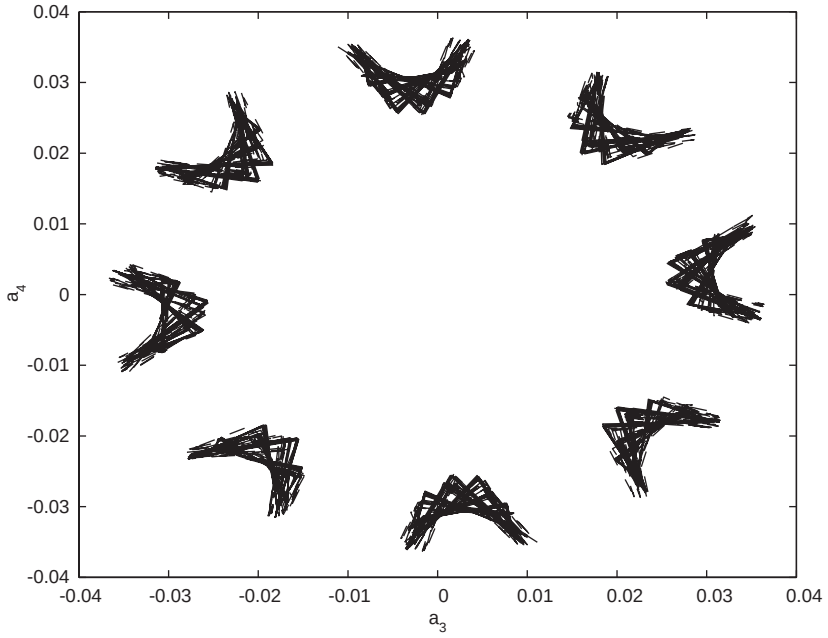

Fig. 20. The four K–L data coefficients a_4 versus the third K–L data coefficient a_3 of the symmetrized vorticity simulation data (solid) versus the neural network prediction (dashed).

the whole phase space. We applied the D_8 symmetry on the numerical data without numerically integrating the two-dimensional N–S equations. The obtained symmetrized data was projected onto the symmetrized eigenfunctions and a set of data coefficients was obtained representing the eight tori earlier observed. Three different neural network models were constructed to model and predict the data coefficients. It was found that by applying a certain symmetry observed in the data coefficient, one can obtain a reduced neural network model capable of producing the same dynamical behavior as in the other two models. This work establishes a foundation for modelling the two-dimensional N–S equations using neural networks for all values of Reynolds numbers and for different values of wave numbers which will be the subject of future studies.

Acknowledgements

We would like to thank Dr. Mansour Al-Zanaidi for making some instructive comments on the paper.

References

[1] P. Antsaklis, Neural networks in control system, IEEE Control Systems Mag. 10 (3) (1990) 3–5.
[2] D. Armbruster, J. Guckenheimer, P.J. Holmes, Kuramoto–Sivashinsky dynamics on the center-unstable manifold, SIAM J. Appl. Math. 49 (3) (1989) 676–691.

[3] D. Armbruster, R. Heiland, E. Kostelich, B. Nicolaenko, Phase-space analysis of bursting behavior in Kolmogorov flow, Physica D 58 (1992) 392.
[4] D. Armbruster, B. Nicolaenko, N. Smaoui, P. Chossat, Analyzing bifurcations in the Kolmogorov flow equations, in: P. Chossat (Ed.), Dynamics, Bifurcation and Symmetry, Kluwer, Dordrecht, 1994, pp. 11–33.
[5] D. Armbruster, B. Nicolaenko, N. Smaoui, P. Chossat, Symmetries and dynamics for 2-D Navier–Stokes flow, Physica D 95 (1) (1996) 81–93.
[6] V.I. Arnold, L.D. Meshalkin, Kolmogorov's seminar on selected problems of analysis, Usprkhi Mat. Nauk. 15 (1960) 247–250.
[7] N. Aubry, W-Y. Lian, E.S. Titi, Preserving symmetries in the proper orthogonal decomposition, SIAM J. Sci. Comput. 14 (1993) 483–505.
[8] A.V. Babin, M.I. Vishik, Attractors of partial differential equations and estimate of their dimension, Uspeki Mat. Nauk. 38 (1983) 133–187 (in Russian); Russian Math. Surveys 38 (1983) 151–213 (in English)..
[9] Y. Bengio, Neural Network for Speech and Sequence Recognition, International Thompson Computer Press, London, UK, 1996.
[10] G. Berkooz, E.S. Titi, Galerkin projections and the proper orthogonal decomposition for equivariant equations, Phys. Lett. A 174 (1993) 94–102.
[11] H. Bourlard, C.J. Wellekens, Speech dynamics and recurrent neural networks, Proceedings of ICASSP, Glasgow, UK, 1989, pp. 33–36.
[12] C.L. Brooks, M. Karplus, B.M. Pettitt, A Theoretical Prespective of Dynamics, Structure and Thermodynamics, Wiley, New York, NY, 1988.
[13] D.S. Broomhead, M. Kirby, The Whitney reduction network: a method for computing autoassociative graphs, Neural Comput. 30 (2001) 2595–2616.
[14] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer Series in Computational Physics, Springer, Berlin, 1988.
[15] M. Chester, Neural Networks: A Tutorial, Prentice-Hall, Englewood Cliffs, NJ, 1993.
[16] P. Constantin, C. Foias, B. Nicolaenko, R. Teman, Integral manifold and inertial manifolds for dissipative partial differential equations, Applied Mathematical Sciences, Springer, New York, 1989.
[17] P. Constantin, C. Foias, R. Teman, On the dimension of the attractors in two-dimensional turbulence, Physica D 30 (3) (1988) 284–296.
[18] M. Dellnitz, M. Golubitsky, M. Nicol, Symmetry of attractors and the Karhunen–Loéve decomposition, in: L. Sirovich (Ed.), Trends and Perspectives in Applied Mathematics, Springer, Berlin, New York, 1994.
[19] Y. Ephrain, L. Rabiner, On the relations between modelling approaches for speech recognition, IEEE Trans. Acoust. Speech Signal Process. 36 (1990) 372–379.
[20] C. Foias, G.R. Sell, R. Teman, Inertial manifolds for nonlinear evolutionary equations, J. Differential Equations. IMA Preprint Series (234) (1986).
[21] C. Foias, R. Teman, On the Hausdorff dimension of an attractor for the two-dimensional Navier–Stokes equations, Phys. Lett. 93 (9) (1983) 451–454.
[22] J.A. Freeman, Simulating Neural Networks, Addison-Wesley Publishing Company, Reading, MA, 1994.
[23] J.A. Freeman, D.M. Skapura, Neural Networks Algorithms, Applications, and Programming Techniques, Addison-Wesley Publishing Company, Reading, MA, 1992.
[24] A.A. Garrouch, N. Smaoui, Application of artificial neural network for estimating tight gas sand intrinsic permeability, Energ. Fuel. 10 (1996) 1053–1059.
[25] R. Gonzalez-Garcia, R. Rico-Martinez, W. Wolf, M. Lubke, M. Eiswirth, J.S. Anderson, I.G. Kevrekidis, Characterization of a two-parameter mixed-mode behavior regime using neural networks, Physica D 151 (1) (2001) 27–43.
[26] A.C. Gonzalez, P. Winiz, Digital Image Processing, 2nd Edition, Addison-Wesley, Reading, MA, 1987, pp. 122–130.
[27] H. Harman, Modern Factor Analysis, University of Chicago Press, Chicago, 1960.
[28] D. Henry, Geometric theory of semilinear parabolic equations, in: Lecture Notes in Mathematics, Vol. 840, Springer, New York, 1981.
[29] J.J. Hopfield, D.W. Tank, Neural computation of decisions optimization problems, Biol. Cybernet. 52 (1985) 141–144.

[30] H. Hotelling, Analysis of a complex statistical variables into principal components, J. Ed. Psychol. 24 (1933) 498–520.
[31] J.M. Hyman, B. Nicolaenko, S. Zaleski, Order and complexity in the Kuramoto–Sivashinsky model of weakly turbulent interface, Physica D 23 (1986) 265–292.
[32] L.T. Jollife, Principal Component Analysis, Springer, New York, 1986.
[33] M.S. Jolly, Bifurcation computations on an approximate inertial manifold for the 2-D Navier–Stokes equations, Physica D 36 (1993) 8–20.
[34] M.S. Jolly, I.G. Kevrekidis, E.S. Titi, Approximate inertial manifolds for the Kuramoto–Sivashinsky equation: analysis and computations, Physica D 44 (1990) 38–60.
[35] I.G. Kevrekidis, B. Nicolaenko, C. Scovel, Back in the saddle again: a computer assisted study of the Kuramoto–Sivashinsky equation, SIAM J. Appl. Math. 50 (3) (1990) 760–790.
[36] M. Kirby, R. Miranda, The nonlinear reduction of high-dimensional dynamical systems via neural networks, Phys. Rev. Lett. 72 (12) (1994) 1822.
[37] D. Kleifeld, H. Sompolinsky, Associative neural network model for the generation of temporal patterns, Biophysics 54 (1988) 1039–1051.
[38] Y. Kuramoto, T. Tsuzuki, Persistent propagation of concentration waves in dissipative media far from thermal equilibrium, Prog. Theoret. Phys. 55 (1976) 356–369.
[39] E.N. Lorenz, Deterministic nonperiodic flow, J. Atmos. Sci. 20 (1963) 130–141.
[40] J.L. Lumley, in: A.M. Yaglom, V.I. Tatarski (Eds.), Atmospheric Turbulence and Radio Wave Propagation, Nauka, Moscow, 1967.
[41] J. Mallet-Paret, Negatively invariant sets of compact maps and an extension of a theorem of Cartwright, J. Differential Equations 22 (1976) 331.
[42] R. Mañé, On the dimension of the compact invariant sets of certain nonlinear maps, in: Lecture Notes in Mathematics, Vol. 898, Springer, New York, 1981.
[43] C. Marchioro, An example of absence of turbulence for any Reynolds number, Comm. Math. Phys. 105 (1986) 99–106.
[44] R. Martinez, R.K. Krisher, I.G. Kevrekidis, M. Kube, J.L. Hudson, Discrete vs. continuous-time nonlinear signal processing of Cu electrodissolution data, Chem. Eng. Comm. 118 (1992) 25–48.
[45] L.D. Meshalkin, A.G. Sinal, Investigation of the stability of a stationary solution of a system of equations for the plane movement of an incompressible viscous fluid, J. Appl. Math. Mech. 25 (1961) 1700–1705.
[46] B. Nicolaenko, Z.S. She, Symmetry breaking homoclinic chaos in the Kolmogorov flows, in nonlinear world, in: V.G. Baryakhtar, et al., (Eds.), International workshop on Nonlinear and Turbulent Processes in Physics, Kiev, (1989), World Scientific, Singapore, 1990, pp. 602–618.
[47] B. Nicolaenko, Z.S. She, Temporal intermittency and turbulence production in the Kolmogorov flow, in: H.K. Moffatt (Ed.), Topological Fluid Mechanics, Cambridge University press, Cambridge, UK, 1990, pp. 256–277.
[48] B. Nicolaenko, Z.S. She, Symmetry breaking homoclinic chaos and vorticity bursts in periodic Navier–Stokes flows, European J. Mech. B 10 (1991) 67–74.
[49] J.H. Pao, Adaptive Pattern Recognition and Neural Networks, Addison-Wesley, Reading, MA, 1989.
[50] B.A. Pearlmutter, Learning state space trajectories in recurrent neural networks, Neural Comput. 1 (1989) 263–269.
[51] N. Platt, L. Sirovich, N. Fitzmaurice, An investigation of chaotic Kolmogorov flow, Phys. Fluids A 3 (1991) 681–696.
[52] M.B. Reid, L. Spirkovska, E. Ochoa, Rapid training of higher-order neural networks for invariant pattern recognition, Proceedings of the Joint International Conference on Neural Networks, Washington DC, 1989, pp. 689–692.
[53] Z.S. She, Metastability and vortex pairing in the Kolmogorov flow, Phys. Lett. A 124 (1987).
[54] Z.S. She, Large-scale dynamical and transition to turbulence in the two-dimensional Kolmogorov flow, in: H. Branover, M. Mond, Y. Unger (Eds.), Current Trends in Turbulence Research, AIAA, Washington, DC, 1988, pp. 374–396.
[55] L. Sirovich, Turbulence and dynamics of coherent structures. I. coherent structures, Quart. Appl. Math. 45 (1987) 561–571.
[56] G. Sivashinsky, Nonlinear analysis of hydrodynamic instability in laminar flames. I. derivation of basic equations, Acta Astronaut. 4 (1977) 1177–1206.

[57] N. Smaoui, Artificial neural network-based low-dimensional model for spatio-temporally varying cellular flames, Appl. Math. Modeling 21 (1997) 739–748.
[58] N. Smaoui, An artificial neural network noise reduction method for chaotic attractors, Internat. J. Comput. Math. 73 (2000) 417–431.
[59] N. Smaoui, A model for the unstable manifold of the bursting behavior in the 2d Navier–Stokes flow, SIAM J. Sci. Comput. 23 (2001) 824–840.
[60] N. Smaoui, Linear vs. nonlinear dimensionality reduction of high dimensional dynamical system, SIAM J. Sci. Comput. (2004), to appear.
[61] N. Smaoui, S. Al-Yakoob, Analyzing the dynamics of cellular flames using Karhunen–Loéve decomposition and autoassociative neural networks, SIAM J. Sci. Comput. 24 (5) (2003) 1790–1808.
[62] N. Smaoui, D. Armbruster, Symmetry and the Karhunen–Loéve analysis, SIAM J. Sci. Comput. 18 (1997) 1526–1532.
[63] N. Smaoui, A.A. Garrouch, A new approach combining Karhunen–Loéve analysis and artificial neural network for estimating gas sand permeability, J. Petrol. Sci. Eng. 18 (1997) 101–112.
[64] N. Smaoui, R. Gharbi, Using Karhunen–Loéve decommission and artificial neural network to model fluid flow in porous media, Appl. Math. Modeling 24 (2000) 657–675.
[65] N. Smaoui, I.M. Matar, Classifications of human faces using K–L decomposition and radial basis function neural networks, Internat. J. Comput. Math. 80 (3) (2003) 325–345.
[66] G. Strang, Linear Algebra and its Application, Academic Press, New York, 1976.
[67] R. Teman, Navier–Stokes Equations: Theory and Numerical Analysis, Revised Edition, North-Holland, Amsterdam, 1979.