Parameter Identification by Eigenfeature Analysis: Application to 2D Kuramoto-Sivashinsky Surface Models

D. Reiser[1*], M. Brenzke[1], S. Wiesen[1,2]

[1] *Forschungszentrum Jülich GmbH, Institute of Fusion Energy & Nuclear Waste Management – Plasma Physics, Partner of the Trilateral Euregio Cluster (TEC), 52425 Jülich, Germany*
[2] *DIFFER - Dutch Institute for Fundamental Energy Research, De Zaale 20, 5612 AJ Eindhoven, Netherlands*

We have developed a system that makes it possible to derive parameters of a Kuramoto-Sivashinsky (KS) model from a single given two-dimensional profile of surface structures, such as those produced by ion and plasma irradiation. The numerical method is inspired by well-known approaches to facial recognition. Starting from a scaled version of a KS Model to describe surface erosion, a training set of surface profiles is created. Each profile is assigned an appropriate feature in Fourier space and a Singular Value Decomposition is used to determine an orthogonal set of eigenfeatures that allow each profile to be assigned a point in the space of this basis and to determine the distances between them. It turns out that the profiles belonging to different model parameters are clearly separated from each other in this feature space, which enables very good identification. We explain the basic relationships using a synthetic data set and discuss the possibilities for applications to experimental results.

Keywords: Pattern Recognition, Data Driven Model Discovery, Ion Beam Analysis, Surface Morphology, Continuum Models

I. INTRODUCTION

By bombarding surfaces with high-energy ions, it is now possible to obtain extensive information on surface structures with a very high degree of accuracy. One example of this is atomic force microscopy, which is used intensively to investigate erosion-induced surface changes. These experimental findings can be compared with numerous theoretical and numerical studies, which have shown that a large number of morphological changes can be described by relatively simple continuum models. Since the pioneering work of Bradley and Harper [1], various linear and non-linear extensions of continuum models have been successfully used to describe the temporal evolution of a variety of surface structures. In all these continuum models, a single evolution equation is formulated for a height function $h(x, y, t)$, describing the height of a surface with respect to a reference x-y-plane and time t. Contributions to such an equation reflect various physical phenomena resulting from interactions between ions and solid surfaces. Discussions of derivations of such models, their fundamental properties and examples of applications can be found, for example, in Refs. [2–9]. Of course the properties of such model equations depend on certain parameters representing different physical processes in the system uinder consideration. Therefore, the derivation of relationships between observed surface profiles and the corresponding model parameters would be desirable. A semi-empirical approach to cope with this problem has been discussed by Muñoz-García et al. in Ref. [7], where they derived model coefficients from analytical estimates and experimental data. Another example is discussed in Ref. [8] where the authors identified the anti-diffusion coefficient of the one dimensional linearized version of a prototypical Kuramoto-Sivashinsky (KS) model [10; 11] for ion-beam irradiated surface profiles. Reiser has shown that, in principle, it is possible to derive the model coefficients of the two-dimensional nonlinear version of such continuum models by analyzing two or more consecutive snapshots of the surface structures during irradiation [12]. In further studies it became clear that the restriction of the time step between the snapshots, as required in this approach, is not compatible with experimental constraints. The data currently available are still too uncertain with respect to possibilities of tracking fine details to arrive at clear results. The basic problem lies in the necessary mounting and demounting of probes needed for the experimental analysis, which does not allow to follow individual points in time. For this reason, we decided to study, or rather develop, methods for which only a single snapshot is sufficient to enable model discovery. This parameter identification process is based on generating features from the spectra of surface profiles with known model parameters to construct a basis of an eigenspace in which any new profile snapshot can be expanded and be considered as a point with respect to this basis. Thus, the Fourier spectrum of a single newly acquired snapshot can then be compared with known snapshots in this eigenspace to then extract corresponding model parameters. A brief sketch of the basic ideas and preliminary results have been published already in Ref. [13]. In this work the approach is explained in detail and numerical results are presented to show that (1) all points in the orthogonal eigenspace

*Corresponding Author: d.reiser@fz-juelich.de

are well separated for different model parameters, (2) an identification of model parameters via nearest neighbor search is possible, (3) only a few eigenfeatures are needed for identification, (4) simple re-scaling allows the use of a short catalogue of training profiles to classify even profiles of very different ranges of model parameters and to consider realistic surface patterns from experiments. We have structured the manuscript as follows: In Section II we present the continuum model for surface structures to be discussed. An appropriate scaling is introduced to restrict the parameter range in the production of training data for the data analysis. In Sec. III we introduce the concept of features in Fourier space to characterize particular surface patterns. The numerical approach based on a Singular Value Decomposition (SVD) of the feature matrix is elucidated. Those concepts are applied to our synthetic data in Sec. IV. It is shown that the training with our set of training profiles gives a satisfactory quality in model identification. Particular aspects in model discovery of certain model parameters are discussed and methods to extend our approach to experimental data are sketched in Sec. V.

II. CONTINUUM MODELS FOR SURFACE MORPHOLOGY

The major goal in our analysis of surface patterns is to extract a continuum model of the KS type. In this work the particular model for the time evolution of the surface height $h(x,y,t)$ reads as

$$\begin{aligned}\frac{\partial h}{\partial t} &= \kappa_x \frac{\partial^2 h}{\partial x^2} + \kappa_y \frac{\partial^2 h}{\partial y^2} + \frac{\lambda_x}{2}\left(\frac{\partial h}{\partial x}\right)^2 \\ &+ \frac{\lambda_y}{2}\left(\frac{\partial h}{\partial y}\right)^2 - K\boldsymbol{\nabla}^2\boldsymbol{\nabla}^2 h - b\,\bar{h}\end{aligned} \tag{1}$$

Here $\bar{h}$ denotes the spatial average of h over the x-y-domain considered. The coefficient b represents a numerical damping effect to push $\bar{h}$ to zero. The coefficients λ_x and λ_y describe the basic process of slope-dependent erosion and (re-) deposition. The terms with κ_x and κ_y describe the effect of surface tension in the erosion and K represents surface diffusion. Of course, this model does not cover all possible morphological patterns observed in experiments. Rather, the way the model equations are formulated here, they represent a relatively simple variant of a KS model. Various extensions of the model have been discussed in the literature (for an overview see e.g. Ref. [6]), but in this paper we restrict ourselves to the form in Eq. 1 in order to have a manageable number of model parameters. The methods we present can also be applied directly to model extensions, which would, however, entail a larger numerical effort.

Now, to consider the equation in a more general context, a scaling is helpful. Therefore, scaling factors for time, spatial length and height are introduced via

$$x \to l_0\,x \quad , \quad y \to l_0\,y \quad , \quad t \to t_0\,t \quad , \quad h \to h_0\,h \tag{2}$$

This leads to the dimensionless form

$$\begin{aligned}\frac{\partial h}{\partial t} &= \kappa_x' \frac{\partial^2 h}{\partial x^2} + \kappa_y' \frac{\partial^2 h}{\partial y^2} + \frac{\lambda_x'}{2}\left(\frac{\partial h}{\partial x}\right)^2 \\ &+ \frac{\lambda_y'}{2}\left(\frac{\partial h}{\partial y}\right)^2 - K'\,\boldsymbol{\nabla}^2\boldsymbol{\nabla}^2 h - b'\,\bar{h}\end{aligned} \tag{3}$$

with scaled model parameters

$$\kappa_x' = \frac{\kappa_x\,t_0}{l_0^2} \quad , \quad \kappa_y' = \frac{\kappa_y\,t_0}{l_0^2} \tag{4}$$

$$\lambda_x' = \frac{\lambda_x\,h_0\,t_0}{l_0^2} \quad , \quad \lambda_y' = \frac{\lambda_y\,h_0\,t_0}{l_0^2} \tag{5}$$

$$K' = \frac{K\,t_0}{l_0^4} \quad , \quad b' = b\,t_0 \tag{6}$$

Introducing the constant reference values κ_0, λ_0 and K_0 as

$$\kappa_0 = \frac{l_0^2}{t_0} \quad , \quad \lambda_0 = \frac{l_0^2}{h_0\,t_0} \quad , \quad K_0 = \frac{l_0^4}{t_0} \tag{7}$$

leads to the equivalent expressions

$$\kappa_x' = \frac{\kappa_x}{\kappa_0} \quad , \quad \kappa_y' = \frac{\kappa_y}{\kappa_0} \tag{8}$$

$$\lambda_x' = \frac{\lambda_x}{\lambda_0} \quad , \quad \lambda_y' = \frac{\lambda_y}{\lambda_0} \tag{9}$$

$$K' = \frac{K}{K_0} \quad , \quad b' = \frac{K_0}{\kappa_0^2}\,b \tag{10}$$

Choosing $\kappa_0 = -\min(\kappa_x, \kappa_y)$ and $\lambda_0 = \max(|\lambda_x|, |\lambda_y|)$, prescribes λ_x' or λ_y' as ± 1. Parameter sets with $\min(\kappa_x, \kappa_y) \geq 0$ do not provide a linear instability driving the formation of surface structures, so such combinations are not considered. Consequently, this constraint always sets one of the two coefficients, κ_x' or κ_y', to -1. As mentioned, the parameter b' is needed just for numerical purposes, i. e. keeping $\bar{h} \sim 0$. Experience shows that $b' = 100$ is appropriate for our simulations. A rotation of the coordinate system, which transforms x into y, does not yield any new structures, so that one can restrict oneself to the following cases with $\kappa_x' = -1$ to conduct a comprehensive parameter scan for synthetic profiles.

$$\kappa_x' = -1 \ , \ \lambda_x' = \pm 1 \ , \ \kappa_y' \in [-1, \infty] \ , \ \lambda_y' \in [-1, 1]$$

$$\kappa_x' = -1 \ , \ \lambda_y' = \pm 1 \ , \ \kappa_y' \in [-1, \infty] \ , \ \lambda_x' \in [-1, 1]$$

(11)

The parameter K' is always set to 1 in these scans. The usual way to read it is that the scaled results can be converted to physical units as long as the parameters κ_0, λ_0, K_0 are known. Now our goal is to generate a catalog of training profiles in order to enable the identification of unknown profiles by comparing them with the catalog. For this purpose, the parameter ranges as described in the relations 11 are to be scanned. For the numerical calculations, a surface of size $L \times L$ is considered, with $L = 200$ in dimensional units. The reference plane is discretized using 256×256 grid points and the time integration of Eq. 3 is done using the Ralston's method (Eq. 3.5 in [14]) with dimensionless time step $\Delta t = 10^{-3}$. Here, however, we encounter a problem, because for a real profile, i.e. an experimentally observed pattern, the model parameters are at the center of interest and completely unknown. In general, one way out would be to abandon scaling the model equation and perform a scan in a suitable part of the physical parameter space. The consequence here would be that far more simulations would have to be provided. However, the method for evaluating the profiles, which is presented in the next sections, would not change. There is therefore no problem in principle and our selection of training data only serves to reduce the computational effort in order to demonstrate the usefulness of the method. In our particular case, however, it is even possible to find a re-scaling that allows the evaluation of real profiles. This point is discussed briefly in Sec. V.

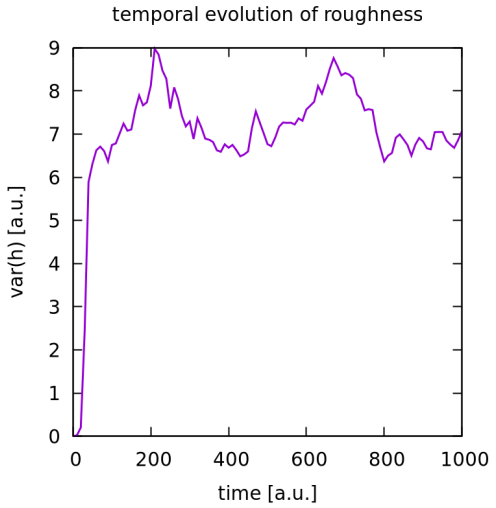

FIG. 1 Example of the time evolution of surface roughness $W^2 = \overline{h^2} - \overline{h}^2$ for model parameters $\kappa_x' = -1$, $\kappa_y' = -1$, $\lambda_x' = 1$, $\lambda_y' = 1/2$, $K' = 1$ and $b' = 100$. The simulation was started at $t = 0$ with small amplitude white noise. Up to $t \approx 40$ an exponential growth is observed, then a non-linear saturation sets in. For the subsequent analysis we focus on snapshots of the saturated phase.

III. EIGENFEATURE APPROACH FOR SINGLE SNAPSHOT DATA

The ideas presented in this work are inspired by known procedures from face recognition, the so-called "eigenface" analysis, [15]. We adapt it as follows: Based on a number of height profile snapshots in the saturated phase for different model parameters, certain features are defined to characterize the corresponding two-dimensional Fourier spectra. Using these features as characteristic patterns the algorithm constructs a basis of "eigenfeatures" in feature space, which is used to expand the feature vector of a particular height profile. If these base vectors are sufficiently representative of the particular model parameters and their impact on the features of the height profiles, it might be possible to identify unknown values of model parameters via inspection of the location in feature space.

A. Fourier Decomposition and Feature Space for Surface Structures

The identification of the model coefficients κ_x', κ_y', λ_x' and λ_y' of Eq. 3 using only a single image of the height profile of the surface $h(x, y, t)$ at a given time t is in the focus of our approach. To accomplish this, certain significant features of particular profiles have to be found, to find a clear correspondence between model parameters and structural details of the profile. The non-linear dynamics lead to strong temporal fluctuations in the surface structures and Fig. 1 illustrates this by means of the temporal changes of the roughness W, defined by $W^2 = \overline{h^2} - \overline{h}^2$. At the beginning an exponential growth appears and later on a non-linear saturation is reached. When comparing the snapshots point by point in the x-y-plane, a single image at one time t is usually completely different from an image at another time $t + \Delta t$, even if they are based on the same set of model parameters and both come from the saturated state. This is illustrated by Fig. 2, where snapshots for three different sets of model parameters at three different time points are shown. Thus, the consideration of individual pixels is not meaningful enough to infer the underlying parameters. This is confirmed by the observation that typical statistical measures such as autocorrelation, mutual information or Fisher information between successive snapshots decrease rapidly over time. From this point of view, the individual snapshots appear very different and unrelated after a certain time.

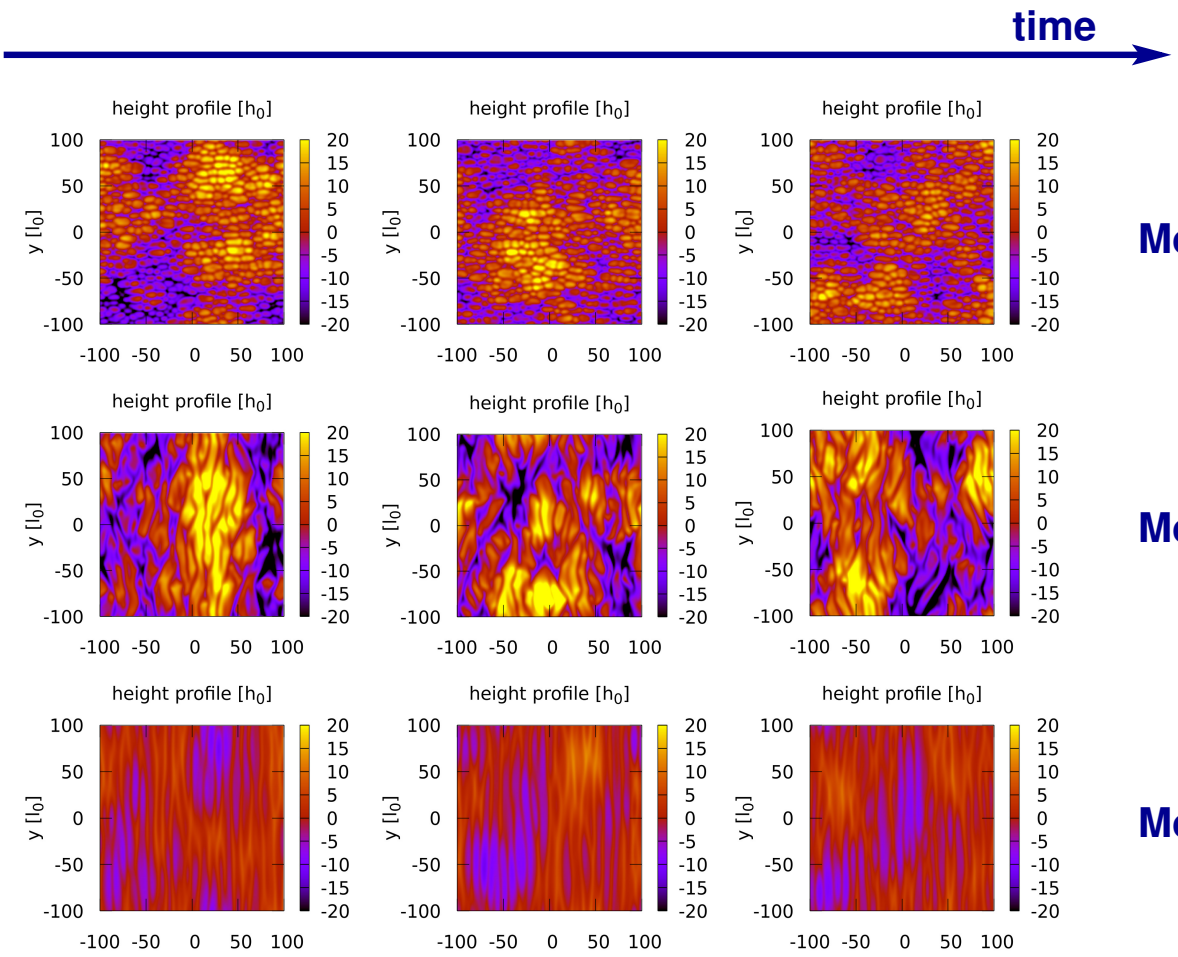

FIG. 2 Examples of time series of snapshots for different sets of model parameters. Even if the details are different, the snapshots of a particular model are similar, in the sense that they have similar structures for the bare human eye. The x- and y-coordinates are in units of l_0 (see section II) and the colorbar shows the height in units of h_0.

On the other hand, even with the bare eye, one finds that the profiles in Fig. 2 indeed do have typical features and profiles of the saturated phase if they belong to the same set of model parameters. They look "similar" for the human eye. The reason is that the typical spatial length scales and symmetries are very significant for a particular set of model parameters. Such a situation suggests to focus on the Fourier spectra. In fact, the Fourier spectra for different time points of the temporal evolution of the surface structures are very similar for a given set of parameters in the non-linear saturation and fluctuate only relatively little around a mean value. This is illustrated by Fig. 3, where certain features (functions of Fourier coefficients) corresponding to the real space profiles of Fig. 2 are shown. Therefore, we now proceed with the Fourier decomposition of the height profile

$$h(x,y,t) = \sum_{m=-N_x}^{N_x} \sum_{n=-N_y}^{N_y} h_{m,n}(t)\, e^{i\,mk_x x + i\,nk_y y} \qquad (12)$$

and introduce feature functions F of the form

$$F(m,n,t) = \begin{cases} \log_{10} f_{m,n} & : \; f_{m,n} \geq f_{m,n}^{\min} \\ \log_{10} f_{m,n}^{\min} & : \; \text{otherwise} \end{cases} \qquad (13)$$

where $f_{m,n}$ is a real valued function of the Fourier components $h_{m,n}(t)$. To avoid $f_{m,n} = 0$ in the logarithm a lower bound is introduced $f_{m,n}^{\min} = \rho\, f_{m,n}^{\max}$ with $0 < \rho < 1$. A particular profile $h(x,y,t)$ at time t is therefore described by a vector of the $N_F = (2N_x + 1) \times (2N_y + 1)$ points $F(m,n,t)$. In our numerical application a time series of N profiles is used to analyze the saturated phase of the surface dynamics.

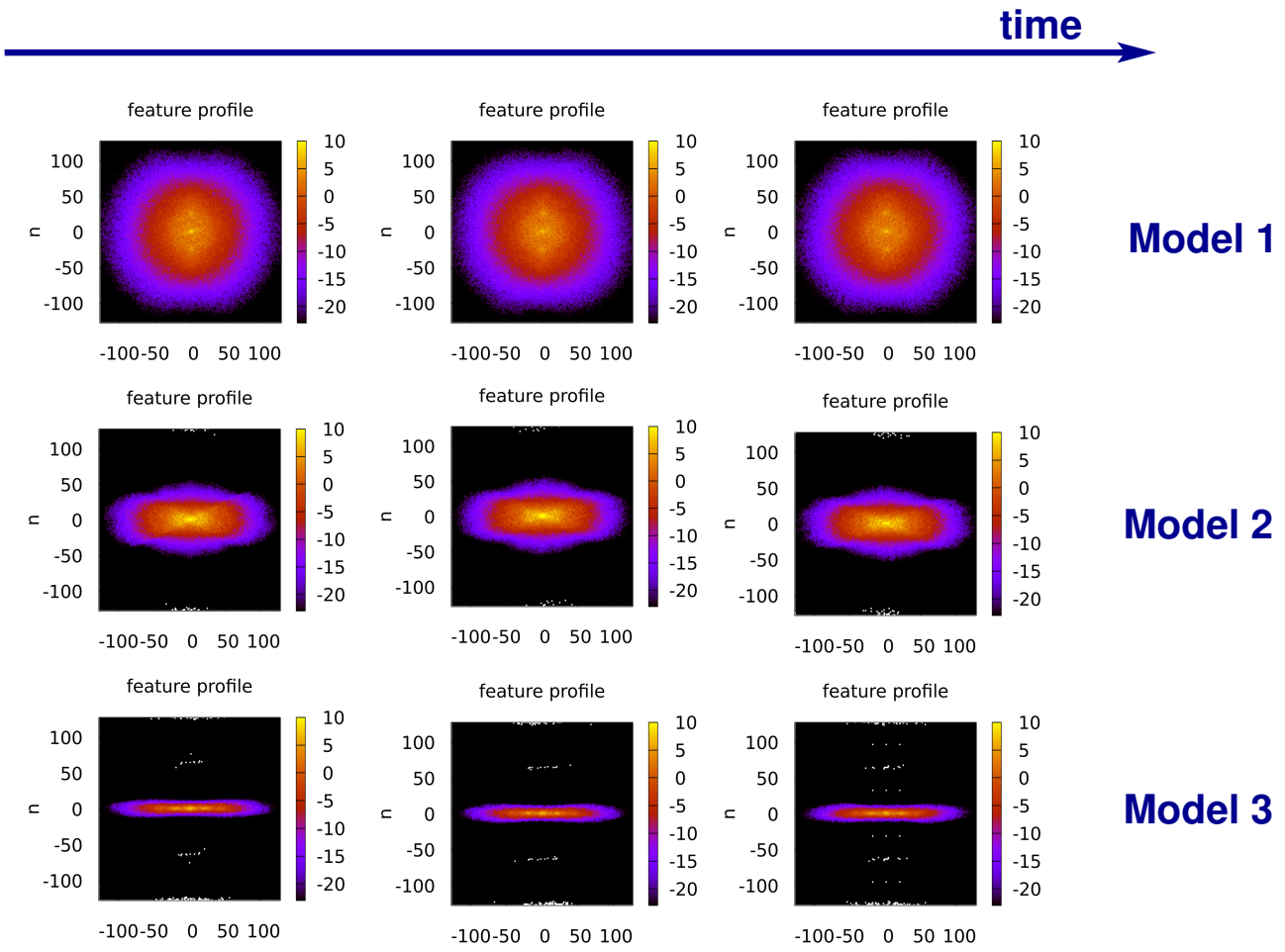

FIG. 3 The features corresponding to the real space profiles of Fig. 2 show certain similarities for a particular set of model parameters. All feature profiles are always very close to temporal average and a distinction between different model parameters is very clear. For the figures shown here the feature function $F^{(1)}$ is used, defined by Eqs. 27 and 28.

For this purpose, the statistically stationary state of the system will be characterized by the average feature elements

$$\bar{F}(m,n) = \frac{1}{N}\sum_{t=1}^{N} F(m,n,t) \tag{14}$$

B. Eigenvectors in Feature Space

To train the algorithm for pattern recognition in surface structures several forward simulations of model Eq. 3 are conducted. This provides typical patterns as defined in Eq. 13 for a variety of model parameter combinations. Each parameter set will be characterized by a temporal average as defined in Eq. 14 and these averaged features (characteristics of the Fourier spectra) form the components Ω_{ij}, of a feature matrix $\mathbf{\Omega} \in \mathbb{R}^{N_F \times N_P}$.

$$\Omega_{ij} = \bar{F}^{(j)}(m_i, n_i), \tag{15}$$

where N_F stands for the number of feature points and N_P for the number of parameter sets used for training. In this notation, the index i serves as a multi-index for all combinations of the indices m and n, and j labels a particular parameter set. The columns of the matrix $\mathbf{\Omega}$ are eventually feature vectors $\mathbf{\Phi}_j$, $j = 1, \ldots, N_P$. They form the training vectors of the recognition procedure and contain the averaged feature profiles for each combination of model parameters considered in the forward simulations. Usually $N_F \gg N_P$, which means the number N_F of feature points is much larger than the number N_P of training profiles. It would be advantageous to find an orthogonal set of base vectors in $\mathbb{R}^{N_F}$ to expand the profiles in this base feature vector space and it would be even better if this space was rather low dimensional,

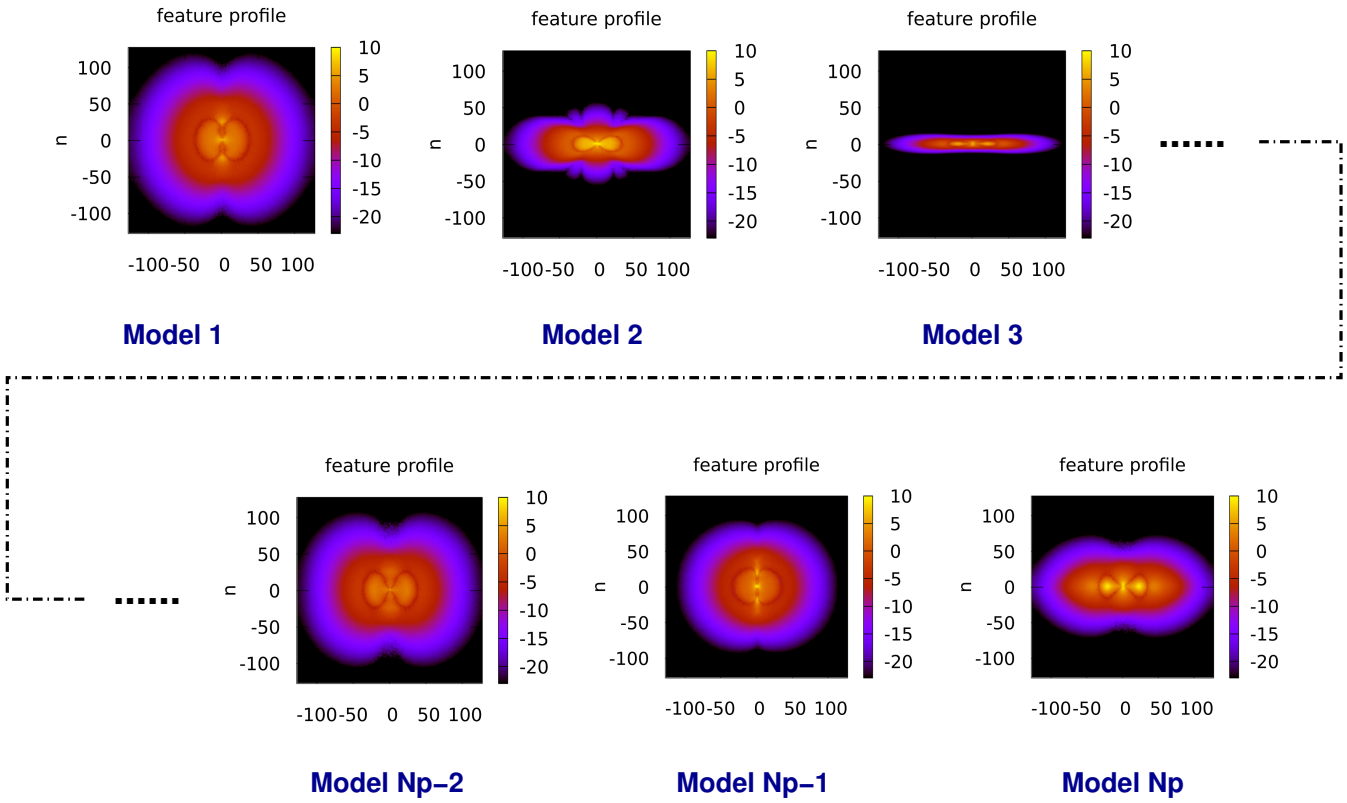

FIG. 4 Combining the averaged features of N_P sets of model parameters constitutes the feature matrix defined by Eq. 15.

i. e. only a few base vectors (much less than N_P) are needed to give an accurate representation of each feature vector for a given profile. It is well known that this can be accomplished by computing the SVD of the matrix $\mathbf{\Omega}$.

$$\mathbf{\Omega} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^{\mathrm{T}} \tag{16}$$

where $\mathbf{U} \in \mathbb{R}^{N_F \times N_F}$, $\mathbf{S} \in \mathbb{R}^{N_F \times N_P}$ and $\mathbf{V} \in \mathbb{R}^{N_P \times N_P}$. The matrices $\mathbf{U}$ and $\mathbf{V}$ are orthonormal and the matrix $\mathbf{S}$ has r non-zero components on the diagonal only, the singular values σ_1, σ_2, ..., σ_r, $r \leq \min(N_F, N_P)$. The other diagonal elements are zero, $\sigma_i = 0$ for $i > r$. The columns of $\mathbf{V}$ (denoted by $\mathbf{v}_i \in \mathbb{R}^{N_P}$, $i = 1, \ldots, N_P$) are normalized eigenvectors of the product $\mathbf{\Omega}^{\mathrm{T}} \cdot \mathbf{\Omega}$ and the columns of $\mathbf{U}$ (denoted by $\mathbf{u}_i \in \mathbb{R}^{N_F}$, $i = 1, \ldots, N_F$) are normalized eigenvectors of the product $\mathbf{\Omega} \cdot \mathbf{\Omega}^{\mathrm{T}}$. Thus, the matrix $\mathbf{\Omega}$ can be represented by dyadic products as

$$\mathbf{\Omega} = \sum_{i=1}^{r} \sigma_i \, \mathbf{u}_i \mathbf{v}_i \quad , \quad \mathbf{\Omega}^{\mathrm{T}} = \sum_{i=1}^{r} \sigma_i \, \mathbf{v}_i \mathbf{u}_i \tag{17}$$

and it follows that the $\mathbf{u}_i$ are the left-singular vectors of $\mathbf{\Omega}$ and the vectors $\mathbf{v}_i$ the right-singular vectors of $\mathbf{\Omega}$, meaning

$$\mathbf{u}_i \cdot \mathbf{\Omega} = \sigma_i \, \mathbf{v}_i \quad , \quad \mathbf{\Omega} \cdot \mathbf{v}_i = \sigma_i \, \mathbf{u}_i \tag{18}$$

Consequently, the vectors $\mathbf{u}_i$ corresponding to a zero singular value form a basis for the left nullspace of $\mathbf{\Omega}$, and the vectors $\mathbf{v}_i$ corresponding to a zero singular value form a basis for the right nullspace of $\mathbf{\Omega}$. Moreover, one obtains

$$\mathbf{\Omega}^{\mathrm{T}} \cdot \mathbf{\Omega} = \sum_{i=1}^{r} \sigma_i^2 \, \mathbf{v}_i \mathbf{v}_i \quad , \quad \mathbf{\Omega} \cdot \mathbf{\Omega}^{\mathrm{T}} = \sum_{i=1}^{r} \sigma_i^2 \, \mathbf{u}_i \mathbf{u}_i \tag{19}$$

and, therefore

$$\mathbf{\Omega}^{\mathrm{T}} \cdot \mathbf{\Omega} \cdot \mathbf{v}_i = \sigma_i^2 \, \mathbf{v}_i \quad , \quad \mathbf{\Omega} \cdot \mathbf{\Omega}^{\mathrm{T}} \cdot \mathbf{u}_i = \sigma_i^2 \, \mathbf{u}_i \tag{20}$$

Finally, the training profiles $\mathbf{\Phi}_j$ (feature vectors) can be represented as

$$\mathbf{\Phi}_j = \mathbf{\Omega} \cdot \hat{\mathbf{q}}_j = \sum_{i=1}^{r} \sigma_i \mathbf{v}_i \cdot \hat{\mathbf{q}}_j \, \mathbf{u}_i = \sum_{i=1}^{r} \sigma_i V_{ji} \, \mathbf{u}_i \tag{21}$$

where $\hat{\mathbf{q}}_j$ is the j-th unit Cartesian base vector in $\mathbb{R}^{N_P}$. Therefore,

$$\mathbf{\Phi}_j \cdot \mathbf{u}_i = \sigma_i V_{ji} \tag{22}$$

Similarly, the feature vector $\boldsymbol{\phi}$ for any other profile (not belonging to the training set) will be expanded in the

same manner using the base vectors $\mathbf{u}_i$

$$\boldsymbol{\phi} = \sum_{i=1}^{N_F} \varphi_i \, \mathbf{u}_i \tag{23}$$

where the expansion coefficients are

$$\boldsymbol{\phi} \cdot \mathbf{u}_i = \varphi_i \tag{24}$$

In practical applications the computation of the base vectors $\mathbf{u}_i$ is done via the computation of the eigenvectors $\mathbf{v}_i$ of $\mathbf{\Omega}^{\mathrm{T}} \cdot \mathbf{\Omega}$ and subsequent use of Eq. 18 in the form $\mathbf{u}_i = \mathbf{\Omega} \cdot \mathbf{v}_i / \sigma_i$ for the non-zero eigenvalues σ_i. Note, that the number r of non-zero eigenvalues is usually much smaller than N_F. Therefore, the base vectors $\mathbf{u}_i$, $i = 1, \ldots, r$ are sufficient to provide an expansion for the feature vector $\mathbf{\Phi}_j$, but might not be sufficient for the expansion of Eq. 23. This would be the case if $\boldsymbol{\phi}$ was very different from the training set and a check of the quality of the expansion of $\boldsymbol{\phi}$ when using the first r base vectors $\mathbf{u}_i$ is only needed as a prerequisite for the analysis described in the next section. This can be done, e. g. by checking if $|\boldsymbol{\phi}|$ is unchanged using the reduced expansion with r base vectors.

C. From Patterns to Model Parameters

The preparatory work of sections III.A and III.B now allows to represent any feature vectors with respect to an orthogonal basis consisting of the "eigenfeatures" $\mathbf{u}_i$. In particular, it is now possible to define a reasonable Euclidian distance vector

$$\boldsymbol{\delta}_j = \mathbf{\Phi}_j - \boldsymbol{\phi} = \sum_{i=1}^{r} (\sigma_i V_{ji} - \varphi_i) \, \mathbf{u}_i \tag{25}$$

such that the distance δ_j between any test profile $\boldsymbol{\phi}$ and the training profile $\mathbf{\Phi}_j$ can be obtained via

$$\delta_j^2 = \sum_{i=1}^{r} (\sigma_i V_{ji} - \varphi_i)^2 \tag{26}$$

This quantity describes how close a test profile is to the respective training profiles when measured against the specific feature function $F(m, n, t)$ as defined by Eq. 13. We usually have more than one model parameter to determine and it is generally not known beforehand which feature represents which parameter particularly well. Therefore, it makes sense to work with several feature functions $F^{(k)}(m, n, t)$ in order to improve the sensitivity of the algorithm with respect to different parameter combinations. This can be done by simply increasing the size of the feature matrix $\mathbf{\Omega}$.

IV. APPLICATION OF EIGENFEATURE APPROACH TO SYNTHETIC DATA

A. Example Data

To apply the approach described in section III synthetic data is generated to calculate the eigenfeatures of a set of $N_P = 146$ time series each consisting of 80 snapshots for grid sizes of 256×256 points. Each time series represents a different set of model parameters according to the scaled model defined by Eqs. 3, 8 – 10 and using the parameter range prescribed in Eqs. 11. The following feature functions have been employed

$$F^{(k)}(m, n, t) = \begin{cases} \log_{10} f_{m,n}^{(k)} & : \; f_{m,n}^{(k)} \geq f_{m,n}^{(k)\min} \\ \log_{10} f_{m,n}^{(k)\min} & : \; \text{otherwise} \end{cases} \tag{27}$$

with $\rho = 10^{-10}$ and $f_{m,n}^{(k)} = |h_{m,n}|^2 \, |g_{m,n}^{(k)}|^2$ where

$$\begin{aligned} g^{(1)} &= \left(\frac{\partial h}{\partial x} \right)^2 \quad , \quad g^{(2)} = \left(\frac{\partial h}{\partial y} \right)^2 \\ g^{(3)} &= \frac{\partial^2 h}{\partial x^2} \quad , \quad g^{(4)} = \frac{\partial^2 h}{\partial y^2} \end{aligned} \tag{28}$$

These feature functions have been evaluated for all 146×80 snapshots available from our forward simulations and the respective averages $\bar{F}^{(k)}(m, n, t)$ were taken over the 80 time steps of the non-linear saturation. The resulting 146 feature functions $\bar{F}^{(1)}, \ldots, \bar{F}^{(4)}$ form the training vectors $\mathbf{\Phi}_j$, $j = 1, \ldots, 146$ with dimension $N_F = 4 \times (2N_x + 1) \times (2N_y + 1)$, where $N_x = N_y = 128$.
As mentioned above, figure 2 shows exemplary snapshots from the dataset of surface height profiles from three time series with different model parameters. In these images and all following depictions of surface height profiles the abscissa and ordinate indicate the position in units of l_0 (see section II) and the colorbar shows the height in units of h_0. The corresponding feature images are shown in Fig. 3 and are obtained from the whole time series (more precisely: the part of the time series which belongs to the statistically stationary phase). For clarity only the features using $g^{(1)}$ are shown. The abscissa and ordinate are in arbitrary units and the colorbar indicates the value of the feature function $\bar{F}^{(1)}(m, n)$.
These examples show clear differences in the structures of the feature images indicative of the different structures observed in the actual surface profiles. For example, the elongated shape visible in the feature images of model 3 in figure 3 is indicative of structures with a stronger elongations along the y-axis in the actual profile than is the case for model 1 and model 2. The subsequent construction of the feature matrix Ω is illustrated by Fig. 4. After averaging over time for all N_P model parameter sets, the resulting feature profiles are combined in the feature matrix which contains the entire training data.

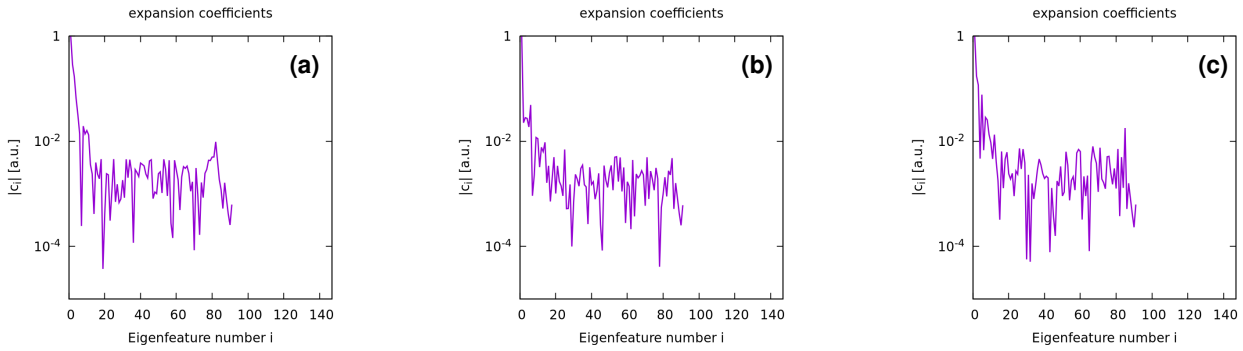

FIG. 5 Squared expansion coefficients for the training vectors $\mathbf{\Phi}_{20}$ (a), $\mathbf{\Phi}_{50}$ (b) and $\mathbf{\Phi}_{98}$ (c). On the x-axis the index (eigenfeature number) of the base vectors $\mathbf{u}_i$, $i = 1, \ldots, 146$ is given. The expansion coefficients are normalized by the value of the first component, which is therefore prescribed as 1. One finds that for $\mathbf{\Phi}_{20}$, $\mathbf{\Phi}_{50}$, $\mathbf{\Phi}_{98}$ and all other training vectors only the first few expansion coefficients are significant. Expansion coefficients not appearing for higher indices i in this half-logarithmic plot are zero.

We are now interested in the following questions

(1) Does the basis of eigenfeatures $\mathbf{u}_i$ provide a low dimensional space where only a small number of expansion coefficients is sufficient to get a reasonable representation of all $\mathbf{\Phi}_j$ according to Eq. 21?

(2) Is the basis of eigenfeatures sufficient to get a reasonable representation of a test vector $\boldsymbol{\phi}$ too? Of course, this is hard to check in general, but a simple check is to test if the length $\boldsymbol{\phi}^2$ is kept unchanged in the reduced basis with r base vectors.

(3) Is it possible to find a clear correspondence between the expansion coefficients φ_i, i. e. the location in feature space of the height profile to be investigated, and the model parameters related to this particular profile?

The third point is of course in the focus of this work. It is clear that the task to identify model parameters from a single snapshot can only be accomplished if the choice of feature functions $f_{m,n}^{(k)}$ is appropriate for the model used. This can not be predicted when studying an unknown situation and even for the model considered here, it is not obvious what might be the most informative function to classify the Fourier-decomposed profiles of the surface height. Therefore, the results presented here must be regarded as preliminary and no final answer can be given at the moment. However, we will show that, despite of these ambiguities, some clear correlations can be observed even for simple choices of features.

B. Expansion Coefficients

To illustrate the answer to item (1) we show in Fig. 5 the expansion coefficients of the averaged feature vectors $\mathbf{\Phi}_{20}$, $\mathbf{\Phi}_{50}$ and $\mathbf{\Phi}_{98}$ as representatives for the entire set of training vectors (the labeling is arbitrary and just according to our specific ordering of the set). Shown are the squared expansion coefficients scaled to the first component, which is therefore set to 1. It is obvious that only a few components contribute significantly to the expansion of the training vectors with respect to the eigenfeatures. Actually a number of less than 20 components is sufficient to get a good representation. Components beyond eigenfeature number 20 do not contribute significantly, because they are all of order 10^{-3}, compared to the first component. The same result is found also for all other training vectors. It can be concluded, that the features defined are useful to reduce the dimension of the feature space to a convenient number.

C. Nearest Neighbor Search

The next point concerns the accuracy in identifying model parameters from a test profile not belonging to the set of training data. As explained earlier, the training data consists of a set of 146 time-averaged profiles of surface height profile features, each obtained by averaging 80 snapshots. The notable point is that all these snapshots look "similar" but are indeed very different if one compares the 2D structures in real space point by point. Consequently the averaged picture is also quite different from a certain snapshot, but should be "similar" with respect to the features corresponding to the underlying model parameters.

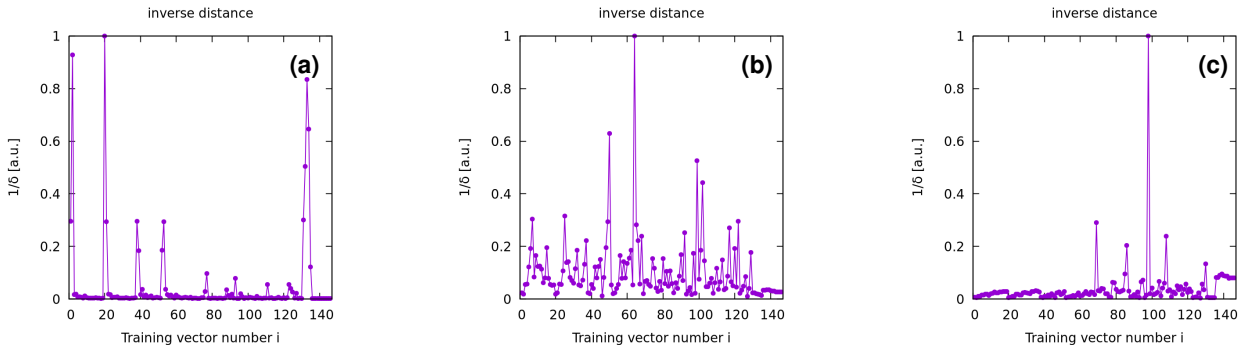

FIG. 6 Nearest neighbor analysis of a test vector (single snapshot) from the time series corresponding to training vectors (averages over time series) $\mathbf{\Phi}_{20}$ (a), $\mathbf{\Phi}_{50}$ (b) and $\mathbf{\Phi}_{98}$ (c), respectively. The strongest peaks correspond to the closest training vectors. They are located at $i = 20$ for $\mathbf{\Phi}_{20}$, at $i = 64$ for $\mathbf{\Phi}_{50}$ and at $i = 98$ for $\mathbf{\Phi}_{98}$. Note that the ideal result would be a dominant peak at $i = K$ for any profile of the time trace belonging to the parameter group of $\mathbf{\Phi}_K$. The deviation for $\mathbf{\Phi}_{50}$ is explained in the text.

We now test whether the group, i. e. the specific set of model parameters, of a single snapshot can be identified by a simple nearest neighbor check, for which we search for the training vector $\mathbf{\Phi}_j$ that has the shortest distance δ_j as defined by Eq. 26. A result for a snapshot belonging to the respective group of $\mathbf{\Phi}_{20}$, $\mathbf{\Phi}_{50}$ and $\mathbf{\Phi}_{98}$ is shown in Fig. 6. The figure shows the inverse distance of the test profile with respect to all training profiles. A good outcome would be to identify $\mathbf{\Phi}_{20}$, $\mathbf{\Phi}_{50}$ or $\mathbf{\Phi}_{98}$ as nearest neighbors, which would mean that the training set has sufficiently separation in the feature space. Such an "ideal" result is indeed obtained in the examples of Fig. 6 for $\mathbf{\Phi}_{20}$ and $\mathbf{\Phi}_{98}$. There the test profiles of the corresponding time traces have the shortest distance to the correct training vector. The result for a test vector of the $\mathbf{\Phi}_{50}$ group is a bit surprising ($i = 64$ has been identified as closest training vector, whereas $i = 50$ would be the optimal result). But, if one takes a closer look at the parameter sets underlying the groups of $\mathbf{\Phi}_{50}$ and $\mathbf{\Phi}_{64}$, respectively, one finds that they are identical except for the parameters λ'_x and λ'_y. But they differ just by their respective sign. This means that the model Eq. 3 is just changed by $\lambda'_x \to -\lambda'_x$ and $\lambda'_y \to -\lambda'_y$ when switching between set number 50 and set number 64. And this is the same as if one would change $h \to -h$, i. e. , the solutions are the same just with sign changed in the height profile. This gives the explanation for the "error": our particular choice of features to characterize the height profiles can not distinguish between surface patterns where only the sign is changed. At the moment we can not offer a simple solution for this, but to remove this mismatch in the identification of parameter groups, it would be needed to refine the feature definition to separate "mirrored" images. We extended this kind of analysis by considering the entire set of 80 snapshots of the respective time traces and counted which training vector has been identified how many times. Again, an ideal result would be that $i = K$ is identified for all snapshots belonging to $\mathbf{\Phi}_K$, which would lead to a count of 80 for the right training vector. The results are shown in Fig. 7. For $\mathbf{\Phi}_{98}$ (subfigure (c)) the result is very satisfactory, all 80 snapshots were identified as member of the group with number 98. For $\mathbf{\Phi}_{50}$ (subfigure (b)) the same mismatch that has been explained above appears for all snapshots. Therefore, the "error" is a systematic mismatch due to the oversimplified feature definition. The results for $\mathbf{\Phi}_{20}$ (subfigure (a)) show that most of the snapshots point to the correct result, but a few are tending to $\mathbf{\Phi}_{133}$. In this case there are almost identical parameters, but again a change in sign in the parameters λ'_x and λ'_y and also the value of $|\lambda'_y|$ is changed from 0.5 to 0.6 when going from model set 20 to set 133. Actually, the model, Eq. 3, considered in this work most often shows only a slight change with λ'_y and it seems that the precise value of the parameters λ'_x and λ'_y does not play a significant role. Therefore, the accurate reconstruction of those parameters might be difficult, but also not too important in our example. On the other hand we observed that the value of κ'_y is most of the times much more precisely found. The similarity between the models set numbered with 20 and the one with number 133 is visualized in Fig. 8. It is obvious that the difference in the parameter λ'_y is not causing a significant difference in the overall structure. From this point of view the pattern recognition technique is working well for our examples. All in all, it can be concluded, that, at least for the test profiles taken from the particular time series in the forward simulations, the comparison of locations in feature space leads to fairly good identification of model parameters. To cope with certain symmetries of the model a refined feature needs to be introduced.

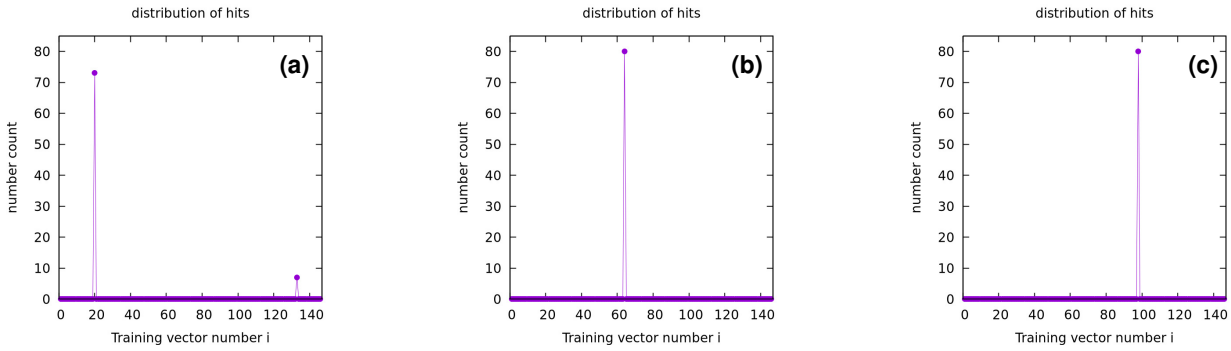

FIG. 7 Shown are the counts of identified training vectors when checked for the entire set of snapshots in the corresponding time trace. Ideally, a count of 80 (all snapshots) are assigned to the right training vector (here: $i = 20, 50, 98$). Deviations are explained in the text.

D. Identification of Arbitrary Synthetic Profiles

Now the next question arises immediately: is it possible to obtain the model parameters for a given single snapshot just by its location in feature space without comparing it with the training data via nearest neighbor distance? This would mean that a certain mapping between model parameters and coordinates in feature space is available. For this purpose we conducted a test with three surface patterns not being part of the model parameters used for preparing the training data. The parameters chosen for this test are listed in Tab. I. For each set we conducted a simulation and prepared a time trace of 80 snapshots in the saturated state as done for the training profiles. Then a nearest neighbor search was performed as before leading to the results shown in Fig. 9. One can see, that there might be more than a single pronounced match in the closest distance and several training vectors have been picked out as best match for the different parameter sets. For comparison with the

	λ_x'	λ_y'	κ_x'	κ_y'
test 1	1.0	0.32	−1.0	2.0
test 2	1.0	−0.34	−1.0	6.0
test 3	1.0	0.60	−1.0	4.0

TABLE I Model parameters for the three test cases. The other parameters used are $K' = 1$ and $b' = 100$.

(known) parameters of the test cases the model parameters for the respective best match training vectors are listed in Tab. II. The first entry for each test problem indicates the training vector which has been identified most often by the algorithm when counting among all 80 snapshots considered. Again the "symmetry error" appears, i. e. the parameters λ_x' and λ_y' both change sign when compared to the true parameter values. As mentioned

	λ_x'	λ_y'	κ_x'	κ_y'
test 1 - 106	−1.0	−0.49	−1.0	2.01
test 1 - 73	−1.0	−0.34	−1.0	1.94
test 1 - 115	−1.0	−0.12	−1.0	1.75
test 2 - 130	−1.0	0.64	−1.0	7.69
test 2 - 146	1.0	1.0	−1.0	5.0
test 3 - 94	−1.0	0.32	−1.0	4.09
test 3 - 146	1.0	1.0	−1.0	5.0
test 3 - 136	−1.0	1.0	−1.0	5.0

TABLE II Identified model parameters for the three test cases. Listed are the most pronounced hits counted among a time trace of 80 snapshots. For example: "test 2 - 130" means that model set number 130 has been identified for snapshots belonging to test model set 2.

above, this is caused by the inability of our symmetric feature choice which can not distinguish between h and $-h$ in the profiles. But more important is the result that the parameter κ_y' is identified quite well. Even though we are dealing with a relatively small training set, the location of the test profiles in eigenfeature space gives already a good estimate for the underlying model parameters. In order to increase the accuracy of interpolation in such a search algorithm a sufficient amount of training data would be essential. We also want to note that we also made some efforts to identify a simple relationship between the expansion components in eigenfeature space and the corresponding model parameters. This was not entirely successful and might be considered in detail in future work. It seems that the length of the eigenspace vector is strongly correlated to parameter κ_y', but other mappings could not be found up to now.

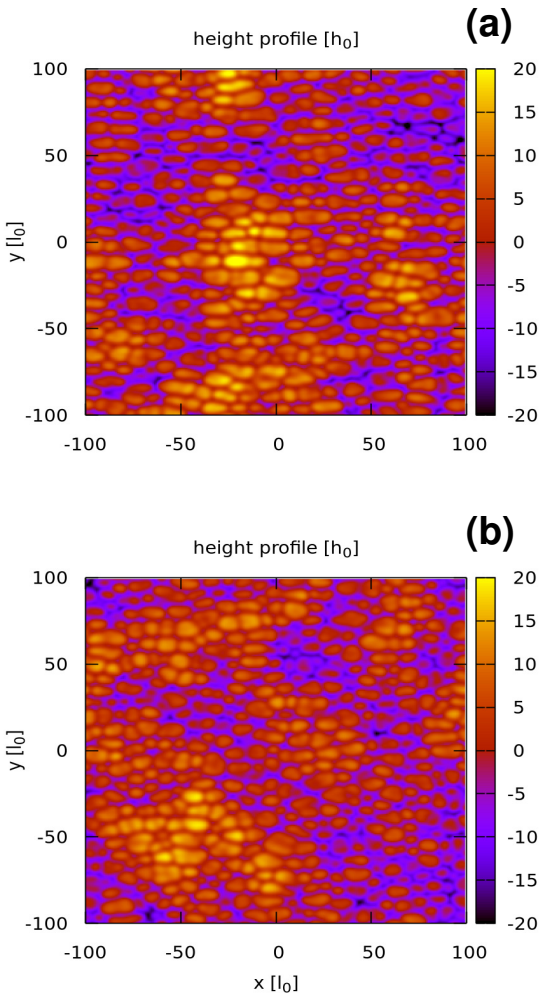

FIG. 8 Comparison of snapshots belonging to model parameter set number 20 (subfigure (a), $\kappa_x' = -1$, $\kappa_y' = -1$, $\lambda_x' = 1$, $\lambda_y' = 0.5$, $K' = 1$ and $b' = 100$) and number 133 (subfigure (b), $\kappa_x' = -1$, $\kappa_y' = -1$, $\lambda_x' = -1$, $\lambda_y' = -0.6$, $K' = 1$ and $b' = 100$). Because λ_x' and λ_y' have both changed sign, the profile of subfigure (b) has been multiplied by -1 to underline the symmetry. More details on this are discussed in the text.

V. APPLICATION TO EXPERIMENTAL DATA

In this section we will give some thoughts on how to apply the method described above in the case where only a single experimental result is available, i. e. only a snapshot, no time series and no indication of the magnitude of the (unscaled) model parameters. This represents a typical regression problem in model discovery, where one tries to reproduce a given profile with the help of a model idea like Eq. 1 and wants to find a set of best-fit parameters. The procedure presented here is not based on an optimization procedure to find best-fit parameters, rather it is a straight-forward comparison with a catalog of known results in an eigenfeature space that has to be set up appropriately (by the proper choice of a feature function). It is necessary to provide a sufficient set of training data to cover the relevant space of possible results. For our KS model studied this has been done by some scaling recipe and this allowed to reduce the amount of training data needed. The question is now what to do, when neither scaling parameters like in Sec. II are known, nor some hint on order of magnitude of model parameters is available. One way to deal with unknown scaling is to go through possible ranges of h_0 and l_0, which have been introduced in Sec. II. Given an experimental profile, a change of h_0 would just result in a change of magnitude of the surface height h and a change in l_0 would be a zoom in or out of the original profile, while keeping the dimensions and scaled heights in the same range as used for the training sets. In our case the ranges were $-100 \leq x, y \leq 100$ and typical ranges of h were of the order of 10. Because each re-scaling via h_0 and l_0 immediately gives new values for the features (e. g. feature $F^{(1)}$, as shown in Fig. 3, would be multiplied by $h_0^2\, l_0^2$) the resulting feature vector representing the experimental profile would be located at another point in eigenfeature space. Therefore, this new location can be used to perform the nearest neighbor search. Scanning through appropriate values of h_0 and l_0 is expected to give a minimum distance for some training vector for a certain combination of h_0 and l_0. Having found these optimal values, the model parameters of the training vector can be re-scaled according to Eqs. 4-6. Actually this identifies all parameters up to a constant t_0 which represents the time scale of the model. But, this parameter is special anyway, because we lost the information on the temporal behaviour of the model when considering just a single snapshot. In other words, all model parameters of KS models can be re-scaled by a time scale t_0 and a single snapshot is unchanged, just appears at a different time. Consequently, the model parameters can only be determined down to a multiplicative constant t_0 but the ratios between the parameters can be determined accurately. Determining the time constant would need to take into account some information on the temporal evolution of the system under consideration.

VI. CONCLUSION AND OUTLOOK

The complex problem of identifying model parameters from the height profile of plasma-irradiated surfaces was investigated along a Kuramoto-Sivashinsky type continuum model. Forward simulations for different combinations of model parameters were carried out to establish a database for pattern recognition analysis. Time averaged features constructed from the Fourier decomposition of the height profiles showed significant differences for different model parameter combinations and could be used to identify model parameters from single snapshots of the temporal evolution of profiles.

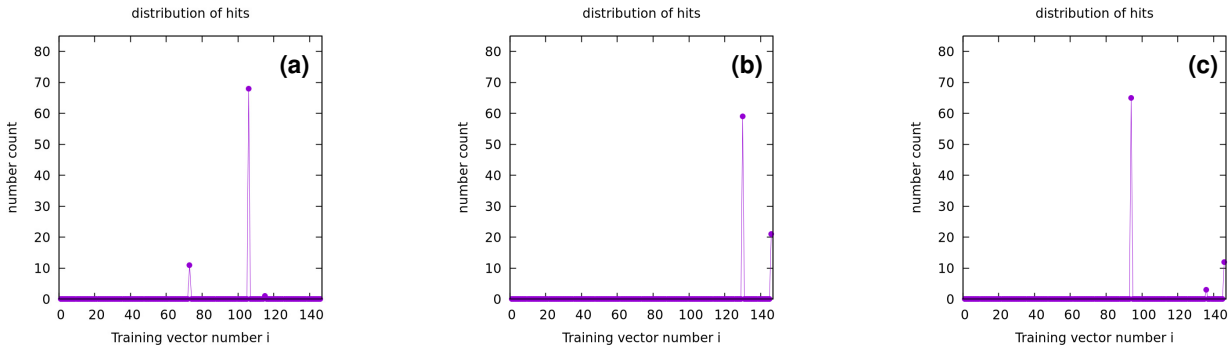

FIG. 9 Shown are the counts of identified training vectors when checked for three test profiles 1, 2 and 3 (labeled by (a), (b) and (c) in the figure). The model parameters corrsponding to the largest number of hits are listed in Tab. II.

Expanding a given profile in the eigenbasis of the feature space of a set of training data of known model parameters showed promising first results for parameter identification via a nearest-neighbor approach. To extract unknown parameters from new profiles which have not been part of the construction of the eigenbasis of the feature space, interpolation in the feature-model parameter space would be needed, but at present no clear mapping could be found. However, a sufficient catalog of simulation results together with an optimization procedure to adjust the scaling parameters for the surface height and the structural length scale can be used to obtain at least an estimate of the order of magnitude of the model parameters. The results we present here were actually found by trial and error. However, the quality of the method depends on a suitable choice of feature functions and other feature definitions yielded poorer results. Nevertheless, the selection presented here seems to be a good starting point for further studies. At the moment, we cannot say how to systematically improve the method by a specific choice of features and how many are needed for optimal results. We therefore consider this manuscript as a first step towards improving the method presented and explained in this paper.

Acknowledgments

This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200 — EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them.

This work was partly funded by the EUROfusion Enabling Research Project ENR-MOD.01.FZJ "Development of machine learning methods and integration of surrogate model predictor schemes for plasma-exhaust and PWI in fusion."

Ethical Approval

Not applicable

Availability of data and materials

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

[1] R. M. Bradley, and J. M. E. Harper. Theory of ripple topography induced by ion bombardment. *J. Vac. Sci. Technol. A*, 6:2390-2395, Jul 1988. doi: 10.1116/1.575561.

[2] J. Muñoz-García, R. Cuerno, and M. Castro. Coupling of morphology to surface transport in ion-beam irradiated surfaces: Oblique incidence. *Phys. Rev. B*, 78:205408, Nov 2008. doi: 10.1116/1.575561.

[3] D. Erb, R. de Schultz, A. Ilinov, K. Nordlund, R. M. Bradley, and S. Facsko. Nanopatterning of the (001) surface of crystalline Ge by ion irradiation at off-normal incidence: Experiment and simulation. *Phys. Rev. B*, 102: 165422, Oct 2020. doi: 10.1103/PhysRevB.102.165422.

[4] S. Facsko, T. Bobek, A. Stahl, H. Kurz, and T. Dekorsy. Dissipative continuum model for self-organized pattern formation during ion-beam erosion. *Phys. Rev. B*, 69: 153412, Apr 2004. doi: 10.1103/PhysRevB.69.153412.

[5] R. Cuerno and A. L. Barabási. Dynamic scaling of ion-sputtered surfaces. *Phys. Rev. Lett.*, 74:4746–4749, Jun 1995. doi: 10.1103/PhysRevLett.74.4746.

[6] R. Cuerno and J.-S. Kim. A perspective on nanoscale pattern formation at surfaces by ion-beam irradiation. *J. Appl. Phys.*, 128(18):180902, 2020. doi: 10.1063/5.0021308.

[7] J. Muñoz-García, R. Gago, L. Vázquez, J. A. Sánchez-García, and R. Cuerno. Observation and modeling of interrupted pattern coarsening: Surface nanostructuring by ion erosion. *Phys. Rev. Lett.*, 104:026101, Jan 2010. doi: 10.1103/PhysRevLett.104.026101.

[8] D. Gajardo, A. Mercado, and J. C. Muñoz. Identification of the anti-diffusion coefficient for the linear Kuramoto-Sivashinsky equation. *J. Math. Anal. Appl.*, 495(2):124747, 2021. doi: 10.1016/j.jmaa.2020.124747.

[9] M. Rost and J. Krug. Anisotropic Kuramoto-Sivashinsky Equation for Surface Growth and Erosion. *Phys. Rev. Lett.*, 75(21):3894–3897, 1995. doi: 10.1103/PhysRevLett.75.3894

[10] Y. Kuramoto and T. Tsuzuki. On the formation of dissipative structures in reaction-diffusion systems. *Progr. Theoret. Phys.*, 54(3), 1975. doi: 10.1143/PTP.54.687.

[11] G. I. Sivashinsky. Nonlinear analysis of hydrodynamic instability in laminar flame. I. derivation of basic equations. *Acta Astron.*, 4(11–12):1177–1206, 1977. doi: 10.1016/0094-5765(77)90096-0.

[12] D. Reiser. Model discovery for studies of surface morphological modifications based on Kuramoto-Sivashinsky dynamics. *Phys. Rev. E*, 100:033312, Sep 2019. doi: 10.1103/PhysRevE.100.033312

[13] S. Wiesen, S. Dasbach, A. Kit, A. E. Jaervinen, A. Gillgren, A. Ho, A. Panera, D. Reiser, M. Brenzke, Y. Poels, E. Westerhof, V. Menkovski, G. F. Derks, and P. Strand. Data-driven models in fusion exhaust: AI methods and perspectives. *Nucl. Fusion*, 64:086046, Jul 2024. doi: 10.1088/1741-4326/ad5a1d

[14] A. Ralston. Runge-Kutta methods with minimum error bounds. *Math. Comp.*, 16:431-437, 1962. doi: 10.1090/S0025-5718-1962-0150954-0.

[15] M. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 01 1991. doi: 10.1162/jocn.1991.3.1.71.