Monograph:

Guo, L.Z., Billings, S.A. and Wei, H.L. (2005) Estimation of spatial derivatives and identification of continuous spatio-temporal dynamical systems. Research Report. ACSE Research Report no. 898 . Automatic Control and Systems Engineering, University of Sheffield

Estimation of Spatial Derivatives and Identification of Continuous Spatio-Temporal Dynamical Systems

Guo, L. Z., Billings, S. A. and Wei, H. L.

Department of Automatic Control and Systems Engineering
University of Sheffield
Sheffield, S1 3JD
UK

Research Report No. 898
June 2005

Estimation of Spatial Derivatives and Identification of Continuous Spatio-Temporal Dynamical Systems

Guo, L. Z., Billings, S. A. and Wei, H. L.

Department of Automatic Control and Systems Engineering
University of Sheffield
Sheffield S1 3JD, UK

Abstract

A new approach for the estimation of spatial derivatives and the identification of a class of continuous spatio-temporal dynamical systems from experimental data is presented in this study. The proposed identification approach is a combination of implicit Adams integration and an orthogonal forward regression algorithm (OFR), in which the operators are expanded using polynomials as basis functions. The noisy experimental data are denoised by using biorthogonal spline wavelet filters and the spatial derivatives are estimated using a multiresolution analysis method. Finally, a bootstrap method is applied to refine the identified parameters from the OFR algorithm. The resulting identified models of the spatio-temporal evolution form a system of partial differential equations. Examples are provided to demonstrate the efficiency of the proposed method.

1 Introduction

Complex spatio-temporal patterns have been widely observed and explored in many diverse fields including physical, chemical, biological, and ecological systems. To better understand these spatio-temporal phenomena and to bring together experiment and theory, much has been achieved to model these systems with partial differential equations (PDE). However, in most of these studies, PDE models arise from mainly theoretical consideration and are derived by analytic modelling methods where often a large number of assumptions have to be made in order to obtain such models. It should be stressed that although some information about the physical properties for many of these systems might be available, normally not all the dynamical structures and parameters are known, therefore, the resulting theoretical PDE models are often over-simplified, or in error. These problems can result in large discrepancies between simulated and observed patterns both qualitatively and quantitatively. Therefore, there is a need to use identification methods to refine, update, validate, or even replace these theoretical PDE models.

The identification of PDE models of continuous spatio-temporal systems has been studied by several authors (Coca and Billings 2000, Fioretti and Jetto 1989, Voss, Bunner, and Abel 1998, Travis and White 1985, Phillipson 1971, Niedzwecki and Liagre 2003). However most of these studies assume that the form of the PDE equations is known up to a set of constant parameters. In this paper a novel approach is used to reconstruct the system of PDEs for unknown continuous spatio-temporal dynamical systems. This new approach represents one of the first algorithms to determine the PDE model terms, and estimate the unknown parameters, from a given noisy spatio-temporal data set. The approach can be regarded as the inverse of the classical Adams method for the numerical solution of differential equations, that is, the operator of the evolution is estimated from the observed values of the system variables. By using Adams integration, a system of algebraic equations can be obtained for the underlying continuous spatio-temporal system that is discrete in time. The advantages of the Adams method over Euler integration is that the former should provide a better fit for less data than the latter, and the latter works well only when the sampling interval is small which might amplify any possible noise. When applying the Adams formula, the higher-order spatial derivatives have to be included in the identification data. A major difficulty is that these derivatives can not generally be measured directly. In these circumstances the most common approach is to use finite difference approximations. However it is well known that numerically differentiating discrete noisy data can prove to be very difficult because such a differentiation process tends to amplify the effects of the noise, particularly in the case of higher-order derivative estimation. This may cause extra difficulties when the spatial domain of interest is of more than one-dimension. To overcome this problem, a B-spline wavelet-based strategy is introduced. The idea behind the proposed strategy is: the orginal noisy signal is approximately expressed as a series expansion of B-spline functions based on a multiresolution analysis, the resulting signal is then passed through a biorthogonal B-spline wavelet filter with a prespecified threshold value and a denoised signal is then obtained as a new series expansion of B-spline functions. Thanks to the properties of B-spline functions, the higher-derivatives of the signal can be computed easily and effectively by using the retrieved series of B-spline functions. In the case when the signal of interest is deterministic, the proposed strategy is still applicable for calculating the spatial derivatives of the signal. By adapting system identification techniques, the continuous operator can then be estimated from these denoised and estimated data. This is achieved by using a polynomial estimation of the operator and the OFR algorithm (Chen, Billings, and Luo 1989). Note that the estimation to the parameters is generally biased because of the presence of the noise and error. To overcome this problem, in this paper, a bootstrap method is applied to refine these parameters after obtaining the correct and significant terms from the OFR algorithm.

The paper is organised as follows. Section 2 introduces the basic idea of the proposed approach and presents the derivation of the system of algebraic equations using Adams formula. A bias analysis from standard least squares algorithm is given as well. The method for denoising the signals and estimating the spatial derivatives is presented in section 3. The OFR algorithm for detecting the significant terms and the bootstrap method for estimating the corresponding parameters are given in sections 4 and 5, respectively. Section 6 illustrates the proposed approach, and finally conclusions are given in section 7.

2 Problem description

2.1 System modelling

Assume that the evolution of a continuous spatio-temporal dynamical system is governed by a system of partial differential equations as follows

$$\dot{y} = f(y, y', y'', \cdots, y^{(l)}), x \in \Omega, t \in T \tag{1}$$

where $y(x,t) \in R^n$ is the independent variable of the system, The dot $\cdot$ denotes the time derivative of y and the prime $'$ denotes the spatial derivatives of y. $t \in T$ denotes time and $x \in \Omega \subset R^m$ denotes the spatial variable. $f(\cdot)$ is a unknown nonlinear function. Note that the time and spatial variables t and x do not appear in f directly. This indicates that the system under consideration is time and spatial invariant. The initial and boundary conditions for eqn.(1) are assumed as

$$g(y(0,t)) = y_i(x) \tag{2}$$

and

$$h(y(x,t)) = y_b(x,t), x \in \partial\Omega \tag{3}$$

For such a continuous spatio-temporal system, experimental measurements are often available in the form of a series of snapshots $y(x, n\Delta t)$, $n = 0, 1, 2, \cdots$, $x \in \Omega$, where Δt is the time sampling interval. In this paper, it is assumed that all the components of the vector $y(x,t) \in R^n$ at one location x are measurable (subject to some measurement noise) otherwise some state space reconstruction techniques may be needed (Packard, Crutchfield, and Farmer 1980, Takens 1981, Sauer, Yorke, and Casdagli 1991). The objective is to determine the nonlinear function f in eqn. (1) from these discrete (noisy) measured snapshots. To this end, the Adams-Bashforth formula (Press, Flannery, Teukolsky, and Vetterling 1992) is used to obtain a discrete representation of eqn. (1). Consider a point x in the spatial domain Ω, let $y_n(x) = y(x, n\Delta t)$, then it follows

$$y_{n+1}(x) = y_n(x) + \int_{n\Delta t}^{(n+1)\Delta t} \dot{y}(x,t)dt = y_n(x) + \int_{n\Delta t}^{(n+1)\Delta t} f(y(x,t), y'(x,t), y''(x,t), \cdots, y^{(l)}(x,t))dt \tag{4}$$

The Adams-Bashforth formula of order p is obtained by integrating a polynomial that interpolates $f_{n+1-j}(x), j = 1, \cdots, p$, that is

$$y_{n+1}(x) = y_n(x) + \Delta t \sum_{j=1}^{p} \alpha_j f_{n+1-j}(x) + e_{n+1}(x) \tag{5}$$

where $f_{n+1-j}(x) = f(y_{n+1-j}(x), y'_{n+1-j}(x), \cdots, y^{(l)}_{n+1-j}(x))$ and $e_{n+1}(x)$ is the approximation error of the Adams-Bashforth formula. Note that eqn. (5) reduces to Euler integration when $p = 1$. The advantages of Adams-Bashforth integration over Euler integration is the former should provide a better fit for less data than the latter and the latter works well only when the sampling interval Δt is small which might amplify any possible noise.

Unlike the numerical problem, in our case the (noisy) experimental data $y_n(x), n = 1, 2, \cdots$, is given, and the task is to determine the nonlinear function f in eqn. (5). Here it is assumed that the unknown nonlinear function f takes a form of polynomial and the polynomial form of eqn (5) is given by the model

$$y_{n+1}(x) = y_n(x) + \Delta t \sum_{j=1}^{p} \alpha_j \sum_{i=1}^{M} \beta_i \phi_i(y_{n+1-j}(x), \cdots, y^{(l)}_{n+1-j}(x)) + e_{n+1}(x) \tag{6}$$

where M denotes the order of the polynomial, β_i is the coefficient of the ith polynomial term, and $\phi_i(y_{n+1-j}(x), \cdots, y^{(l)}_{n+1-j}(x))$ is the corresponding monomial which is the product of different spatial derivatives of $y_{n+1-j}(x)$ at x.

Discretising the spatial domain Ω with sampling interval Δx, re-arranging and rewriting eqn. (6) in the form of a linear-in-the- parameters β_i yields

$$y(k, n+1) = y(k, n) + \sum_{i=1}^{M} \beta_i (\sum_{j=1}^{p} \Delta t \alpha_j \phi_i(y(k, n+1-j), \cdots, y^{(l)}(k, n+1-j))) + e(k, n+1) \tag{7}$$

where $k = x_0 + k\Delta x$ and $n = n\Delta t$ are the discrete spatial location and time instant, respectively, In principle, both the parameters α_j and β_i should be calculated during identification. For the sake of simplicity, the values of the α_j are the ones originally dictated by the Adams-Bashforth formula. Therefore β_i are the only parameters that need to be determined.

2.2 The bias analysis

Eqn. (7) is a discrete representation of the original continuous PDE (1). It is reasonable to assume that the modelling error $e(k, n)$ is an independent noise sequence with zero mean and finite variance. In this case, a least squares-like algorithm will provide an unbiased estimation to the parameter β_i if the data are deterministic and all spatial derivatives are available. However, in practice, this is clearly not realistic. In what follows it is shown that a least squares type algorithm will produce a biased estimation in the presence of noise, even if the noise is white.

Let

$$z(k,n) = y(k,n) + w(k,n) \tag{8}$$

be a noisy measurement of $y(k,n)$, where $w(k,n)$ is an independent, corrupting noise sequence with zero mean and finite variance. Then the noisy model, which consists of eqns. (7) and (8), can be expressed in a vector form as

$$\begin{aligned} z(k,n+1) &= y(k,n+1) + w(k,n+1) &(9)\\ &= y(k,n) + \sum_{i=1}^{M} \beta_i (\sum_{j=1}^{p} \Delta t \alpha_j \phi_i(k,n+1-j)) + e(k,n+1) + w(k,n+1) &(10)\\ &= P_z\beta + \varepsilon(k,n+1) \end{aligned}$$

where $\phi_i(k,n+1-j) = \phi_i(y(k,n+1-j),\cdots,y^{(l)}(k,n+1-j))$, $\beta = (1,\beta_1,\cdots,\beta_M)^T$, $\varepsilon(k,n+1) = e(k,n+1) + w(k,n+1) + (P_y - P_z)\beta$, and

$$\begin{aligned} P_y &= (y(k,n), \sum_{j=1}^{p} \Delta t \alpha_j \phi_1(y(k,n+1-j),\cdots,y^{(l)}(k,n+1-j)),\cdots, &(11)\\ &\qquad \sum_{j=1}^{p} \Delta t \alpha_j \phi_M(y(k,n+1-j),\cdots,y^{(l)}(k,n+1-j))) \end{aligned}$$

and

$$\begin{aligned} P_z &= (z(k,n), \sum_{j=1}^{p} \Delta t \alpha_j \phi_1(z(k,n+1-j),\cdots,z^{(l)}(k,n+1-j)),\cdots, &(12)\\ &\qquad \sum_{j=1}^{p} \Delta t \alpha_j \phi_M(z(k,n+1-j),\cdots,z^{(l)}(k,n+1-j))) \end{aligned}$$

The least squares parameter estimation is based on minimising the mean squared error criterion

$$J = E[\varepsilon(k,n+1)^2] \tag{13}$$

where $E[\cdot]$ denotes the expectation value, which gives rise to the estimate of β as

$$\tilde{\beta} = E[P_z^T P_z]^{-1} E[P_z^T z(k,n+1)] \tag{14}$$

It follows that if $E[P_z^T \varepsilon(k,n+1)] = 0$ the estimation will be unbiased. Under the assumptions that $e(k,n+1)$ and $w(k,n+1)$ are mutually uncorrelated white noise sequences, then

$$
\begin{aligned}
E[P_z^T \varepsilon(k,n+1)] &= E[P_z^T e(k,n+1)] + E[P_z^T w(k,n+1)] + E[P_z^T (P_y - P_z)]\beta \qquad (15)\\
&= E[P_z^T (P_y - P_z)]\beta
\end{aligned}
$$

which is generally not zero and immediately yields a biased estimation to the parameter $\tilde{\beta}$, even if the noise sequences are white, as follows

$$
\tilde{\beta} = \beta + E[P_z^T P_z]^{-1} E[P_z^T (P_y - P_z)]\beta \qquad (16)
$$

and the bias is closely dependent on the observation noise w and the estimation errors of the spatial derivatives.

The above analysis shows that in order to obtain an accurate identified model in the presence of noise, three problems need to be solved:

1. How to estimate the spatial derivatives. In practice, it is very difficult to measure these spatial derivatives directly. Therefore, some estimation methods have to be employed to obtain an estimated version of these derivatives. The most common used method is finite difference approximations. However, a finite difference approximation method, in particularly, for estimating the higher derivatives, is very sensitive to noise. The higher order derivatives of a signal with a modest SNR (Signal-to-Noise Ratio) could induce extremely low SNR's.

2. How to detect the model structure and select the significant terms. Because the order of polynomial form of f and the correct terms are generally unknown, there is a need to detect the correct order and these terms, which involves a combinational explosion if all possible model structures are tested in a brute force manner.

3. How to determine the parameter β_i. As mentioned above, a general least squares type algorithm will produce a biased estimation to the parameters even if the model structure is correct and the noise sequences are white.

In the following sections, these three problems will be investigated in more detail.

3 Spatial derivative estimation and wavelet filtering

Numerically differentiating discrete noisy data can prove to be very difficult because such a differentiation process tends to amplify the effects of noise, particularly in the case of higher-order derivative estimation. Even for the noise-free case it can be hard to obtain the desired accuracy using an ordinary difference approximation method because of the strong dependence on the length of the sampling interval and the SNR. The new method presented here involves estimating the spatial derivatives directly from a series of snapshots using a multiresolution analysis of the spatio-temporal signal based on B-spline functions. The advantage of using a B-spline wavelet expansion of a function is that the differentiation can be easily performed on the individual smooth wavelet bases and takes a simple form.

3.1 Multiresolution analysis

A multiresolution analysis of $L^2(R^n)$ is defined as a set of closed subspaces V_j with $j \in Z$ that exhibit the following properties (Chui, 1992)

1. $V_j \subset V_{j+1}$,
2. $f(x) \in V_j \Longleftrightarrow f(2x) \in V_{j+1}, j \in Z$,
3. $\bigcup_{j \in Z} V_j$ is dense in $L^2(R^n)$ and $\bigcap_{j \in Z} V_j = \emptyset$,
4. A scaling function $\phi(x) \in V_0$ exists such that the set $\{\phi(x-k) | k \in Z^n\}$ forms a Riesz basis of V_0.

Following the definition of the multiresolution analysis, the set of functions $\{\phi_{j,k} = 2^{j/2}\phi(2^j x - k)\}$ is a Riesz basis of V_j. Let W_j be a complementary space of V_j in V_{j+1}, such that $V_{j+1} = V_j \bigoplus W_j$. Consequently

$$\bigoplus_{j \in Z} W_j = L^2(R^n) \tag{17}$$

W_j is called a wavelet subspace. A function $\psi(x)$ is a wavelet if the set of functions $\{\psi(x-k) | k \in Z^n\}$ is a Riesz basis of W_0. It follows that the set of wavelet functions $\{\psi_{j,k} = 2^{j/2}\psi(2^j x - k)\}$ is a Riesz basis of $L^2(R^n)$.

At resolution j the projection P_j (resp. Q_j) of a function f onto V_j (resp. W_j) that corresponds to the above splitting of $L^2(R^n)$ can be written with the use of a dual scaling function $\tilde{\phi}$ (resp. dual wavelet function $\tilde{\psi}$) as follows

$$P_j f(x) = \sum_k < f, \tilde{\phi}_{j,k} > \phi_{j,k}(x) \tag{18}$$

$$Q_j f(x) = \sum_k < f, \tilde{\psi}_{j,k} > \psi_{j,k}(x) \tag{19}$$

where $< \cdot >$ denotes the inner product. Such wavelets are called biorthogonal wavelets. Generally, $\tilde{\phi} \neq \phi$ and $\tilde{\psi} \neq \psi$ except when orthogonality holds. Note that B-spline wavelets are only biorthogonal. The definition of a multiresolution analysis implies that for any $f(x) \in L^2(R^n)$,

$$\lim_{j\to\infty} P_j f(x) = f(x) \tag{20}$$

$$f(x) = \sum_j Q_j f(x) \tag{21}$$

Since W_j is the complementary subspace of V_j in V_{j+1}, that is $V_{j+1} = V_j \bigoplus W_j$, it follows that $P_{j+1}f(x) = P_j f(x) + Q_j f(x)$. This gives an alternative representation of the projection of a function $f \in L^2(R^n)$ at resolution $j+1$ using both the scaling and wavelet functions as

$$P_{j+1}f(x) = \sum_k < f, \tilde{\phi}_{j,k} > \phi_{j,k}(x) + \sum_k < f, \tilde{\psi}_{j,k} > \psi_{j,k}(x) \tag{22}$$

In this way, it is understood that the projection P_j provides an approximation of the function f at some resolution j and the details left by this approximation are contained in Q_j. By iteration, a wavelet decomposition can be obtained as follows

$$P_{j+1}f(x) = \sum_k < f, \tilde{\phi}_{j-l,k} > \phi_{j-l,k}(x) + \sum_{i=j-l}^{j} \sum_k < f, \tilde{\psi}_{i,k} > \psi_{i,k}(x) \tag{23}$$

Considering the time-frequency characteristics of the wavelet transform, the decomposition and reconstruction (23) are widely used as a filter for noisy signals. Such a decomposition and reconstruction can be performed in a fast way using Mallat's pyramid algorithm (Mallat, 1989). In this paper, the snapshots of (noisy) spatio-temporal signals are initially projected onto the V_j subspace at some designated resolution level j. Then the resulting signals are passed through a wavelet filter (decomposing and thresolding) and finally the filtered signals are used to reconstruct the original signals and their higher-order spatial derivatives.

3.2 Spatial derivative estimation using B-spline functions

3.2.1 One-dimensional case

The one-dimensional cardinal B-spline function N_m of m-th order is given by the following recursive relation

$$N_m(x) = \frac{x}{m-1} N_{m-1}(x) + \frac{m-x}{m-1} N_{m-1}(x-1) \tag{24}$$

where $N_1(x)$ is the indicator function

$$N_1(x) = \{ \begin{array}{ll} 1 & \text{if } x \in (0,1) \\ 0 & \text{otherwise} \end{array} \tag{25}$$

Let the m-th B-spline function be the scaling function, that is

$$\phi^m(x) = N_m(x) \tag{26}$$

Then the wavelets can be expressed in terms of the scaling function

$$\psi^m(x) = \sum_{l=0}^{3m-2} q_l \phi^m(2x - l) \tag{27}$$

with the coefficients given by

$$q_l = \frac{(-1)^l}{2^{m-1}} \sum_{k=0}^{m} \left(\begin{array}{c} m \\ k \end{array} \right) \phi^{2m}(l - k + 1), \; l = 0, \cdots .3m - 2 \tag{28}$$

Note that the B-spline function $\phi^m(x)$ has the property

$$\phi^{m\prime}(x) = N_m'(x) = N_{m-1}(x) - N_{m-1}(x-1) = \phi^{m-1}(x) - \phi^{m-1}(x-1) \tag{29}$$

From (20) any $f(x) \in L^2(R)$ can be represented as

$$f(x) = \lim_{j\to\infty} P_j f(x) = \lim_{j\to\infty} \sum_k < f, \tilde{\phi}_{j,k}^m > \phi_{j,k}^m(x) = \lim_{j\to\infty} \sum_k c_{j,k} \phi_{j,k}^m(x) \tag{30}$$

It follows that if $f(x)$ is smooth enough, the derivative function of $f(x)$ is of the following form

$$f'(x) = \lim_{j\to\infty} P_j f'(x) = \lim_{j\to\infty} \sum_k < f, \tilde{\phi}_{j,k}^m > \phi_{j,k}^{m\prime}(x) = \lim_{j\to\infty} \sum_k b_{j,k} \phi_{j,k}^{m-1}(x) \tag{31}$$

where $b_{j,k} = 2^j(c_{j,k} - c_{j,k-1})$ because of the property (29).

According to eqns. (30) and (31), the jth resolution approximation of a function $f \in L^2(R)$ and its derivative function can be obtained as

$$\begin{aligned} f(x) &\approx \sum_k c_{j,k} \phi_{j,k}^m(x) = \sum_k c_{j,k} \phi_{j,k}^m(x) \\ f'(x) &\approx \sum_k b_{j,k} \phi_{j,k}^{m\prime}(x) = \sum_k 2^j (c_{j,k} - c_{j,k-1}) \phi_{j,k}^{m-1}(x) \end{aligned} \tag{32}$$

The higher-order derivatives of f can be approximated iteratively using formula (32).

3.2.2 n-dimensional case

The B-spline function series representation of a multivariate function f defined on $\mathbf{R}^n$ can be described as follows. Let Φ be a bounded function defined on $\mathbf{R}^n$. For all $p \in \mathbf{Z}$ and $\mathbf{k} \in \mathbf{Z}^n$, a family of functions defined on $\mathbf{R}^n$ can be derived in terms of the translates and dyadic dilates of Φ: $\Phi_{p,\mathbf{k}}(\mathbf{x}) = \Phi(2^p\mathbf{x} - \mathbf{k})$. Then if these functions $\Phi_{p,\mathbf{k}}, p \in \mathbf{Z}, \mathbf{k} \in \mathbf{Z}^n$ form a Riesz basis, function f has a unique decomposition in terms of functions $\Phi_{p,\mathbf{k}}$

$$f(\mathbf{x}) = \sum_{p,\mathbf{k}} \alpha_{p,\mathbf{k}} \Phi_{p,\mathbf{k}}(\mathbf{x}) \tag{33}$$

Such a Riesz basis in space $L^2(R^n)$ can be constructed from some univariate scaling functions and the associated wavelet functions by using the tensor product method. For the sake of simplicity, consider the two-dimensional case. Let $\phi^{m_1}(x) = N_{m_1}(x)$ and $\phi^{m_2}(x) = N_{m_2}(x)$ be the two univariate B-spline scaling fucntions of order m_1 and m_2. Note that here m_1 and m_2 could be different. Then a two-dimensional scaling function can be introduced as follows

$$\Phi^{m_1,m_2}(x_1, x_2) = \phi^{m_1}(x_1)\phi^{m_2}(x_2) \tag{34}$$

and the $2^n - 1$ mother wavelets $\Psi^{m_1,m_2,l}, l = 1, 2, \cdots, 2^n - 1$, are obtained by substituting some $\phi^{m_i}(x_i)$s by $\psi^{m_i}(x_i)$ in (34).

Similar to the one-dimensional case, the jth resolution approximation of a function $f \in L^2(R^2)$ onto V_j with two dimensional B-spline function $\Phi^{m_1,m_2} = N_{m_1}N_{m_2}$ as scaling function can be obtained as

$$f(x_1, x_2) \approx P_j f(x_1, x_2) = \sum_k < f, \tilde{\Phi}^{m_1,m_2}_{j,k_1,k_2} > \Phi^{m_1,m_2}_{j,k_1,k_2}(x_1, x_2) = \sum_k \mathbf{c}_{j,k_1,k_2} \phi^{m_1}_{j,k_1}(x_1)\phi^{m_2}_{j,k_2}(x_2) \tag{35}$$

and its partial derivative functions are approximated as

$$\frac{\partial f(x_1, x_2)}{\partial x_1} \approx \sum_k \mathbf{c}_{j,k_1,k_2} \frac{\partial \phi^{m_1}_{j,k_1}(x_1)\phi^{m_2}_{j,k_2}(x_2)}{\partial x_1} = \sum_k 2^j (c_{j,k_1,k_2} - c_{j,k_1-1,k_2}) \phi^{m_1-1}_{j,k_1}(x_1)\phi^{m_2}_{j,k_2}(x_2) \tag{36}$$

and

$$\frac{\partial f(x_1, x_2)}{\partial x_2} \approx \sum_k \mathbf{c}_{j,k_1,k_2} \frac{\partial \phi^{m_1}_{j,k_1}(x_1)\phi^{m_2}_{j,k_2}(x_2)}{\partial x_2} = \sum_k 2^j (c_{j,k_1,k_2} - c_{j,k_1,k_2-1}) \phi^{m_1}_{j,k_1}(x_1)\phi^{m_2-1}_{j,k_2}(x_2) \tag{37}$$

The higher-order partial derivatives of f can be approximated iteratively using formulae (36) and (37).

The new procedure of denoising a signal and obtaining the spatial derivatives from noisy experimental data can be summarised as follows

- Approximating the original signal with its projection onto V_j with B-spline functions as scaling functions (eqns.(30) and (35)). The selection of the resolution level j depends on the desired approximation accuracy. Generally, the higher the resolution, the better the approximation. The order of B-spline scaling function should be chosen in a way that the signals resulting from subsequent differentiations are at least C^2;
- Performing a wavelet decomposition. The scales of the decomposition generally depend on the length of the data sequence;
- Conducting a threshold filtering method for the obtained wavelet coefficients. The selection of the threshold depends on the SNR of the noisy data;
- Reconstructing the original signal from the denoised wavelet coefficients;
- Computing the spatial derivatives using eqns. (32), (36), and (37).

4 Term detection using OFR algorithm

The noisy discrete model (9) of the partial differential equation (1) obtained is of the form

$$z(k,n+1) = P_z\beta + e(k,n+1) + w(k,n+1) + (P_y - P_z)\beta \tag{38}$$

where $\beta = (1, \beta_1, \cdots, \beta_M)^T$, and P_y and P_z are defined as in (11) and (12).

Substituting $y(k,n) = z(k,n) - w(k,n), y'(k,n) = z'(k,n) + w'(k,n), \cdots, y^{(l)}(k,n) = z^{(l)}(k,n) + w^{(l)}(k,n)$ into P_y yields

$$\begin{aligned}
P_y - P_z \quad = \quad & (y(k,n) - z(k,n), \sum_{j=1}^{p} \Delta t\alpha_j(\phi_1(y(k,n+1-j), \cdots, y^{(l)}(k,n+1-j)) \\
& -(\phi_1(z(k,n+1-j), \cdots, z^{(l)}(k,n+1-j))), \\
& \cdots, \sum_{j=1}^{p} \Delta t\alpha_j(\phi_M(y(k,n+1-j), \cdots, y^{(l)}(k,n+1-j)) \\
& -\phi_M(z(k,n+1-j), \cdots, z^{(l)}(k,n+1-j))) \\
= \quad & (-w(k,n), \\
& \psi_1(z(k,n), \cdots, z(k,n+1-p), w(k,n), \cdots, w(k,n+1-p), \cdots, w^{(l)}(k,n), \cdots, w^{(l)}(k,n+1-p)), \\
& \cdots, \\
& \psi_M(z(k,n), \cdots z(k,n+1-p), w(k,n), \cdots, w(k,n+1-p), \cdots, w^{(l)}(k,n), \cdots, w^{(l)}(k,n+1-p)))
\end{aligned} \tag{39}$$

where ψ_i is a sum of the products of its variables, which are some of the $z(k,n), \cdots z(k,n+1-p), w(k,n), \cdots, w(k,n+1-p), \cdots, w^{(l)}(k,n), \cdots, w^{(l)}(k,n+1-p))$. It is stressed that $w^{(i)}(k,n)$ should be understood as the estimate error of $y^{(i)}(k,n)$ rather than the i-th order derivative of $w(k,n)$. Then eqn. (38) can be augmented as

$$z(k,n+1) = P_z\beta + Q_z\theta + w(k,n+1) \tag{40}$$

where Q_z includes the terms associated with error variables, that is, it is a vector composed of all possible monomials formed by the products of the lagged signals of $z(k,n), w(k,n), \cdots, w^{(l)}(k,n)$ from $P_y - P_z$, and θ represents the corresponding coefficients. Note that in eqn (40), $e(k,n+1)$ has been absorbed into $w(k,n+1)$. This can be done because both $e(k,n+1)$ and $w(k,n+1)$ are white. In this way, the bias associated with the noisy model (9) can often be reduced or eliminated by using the well known Orthogonal Forward Regression algorithm (Chen, Billings, and Luo 1989). The OFR algorithm involves a stepwise orthogonalisation of the regressors and a forward selection of the relevant terms based on the Error Reduction Ratio criterion (Billings, Chen, and Kronenberg 1988). It should be emphasised that $w^{(1)}(k,n), \cdots, w^{(l)}(k,n)$ are actually the estimation errors of the spatial derivatives of $y(k,n)$, the OFR algorithm can not eliminate these errors so that the final estimation of the parameters will generally be biased. However, one of the advantages of the OFR algorithm is that it can be used to separate the deterministic part ($P_z\beta$) and uncertain part ($Q_z\theta$) of the model and therefore determine the final deterministic model structure by removing the uncertain part. In general, the OFR algorithm can produce the correct terms as long as the presence of the noise and/or the filter does not change the correlations between the variables. With this deterministic model structure, the parameters can be estimated using some other method. In this paper, a bootstrap algorithm is applied to tackle this problem, this will be discussed in the next section.

Let N be the length of the measured data from a specific spatial location k, then a linear-in-the parameter form of the system at that location can be expressed as

$$Z = Z(k) = P_z(k)\beta + Q_z(k)\theta + E(k) \tag{41}$$

where $Z(k) = (z(k,1), \cdots, z(k,N))^T$, $E(k) = (w(k,1), \cdots, w(k,N))^T$, and $P_z(k)$ and $Q_z(k)$ are corresponding regression matrices. For a given candidate regressor set $G = \{\varphi_i\}_{i=1}^M$, where $\varphi_i = (\sum_{j=0}^{p-1} \Delta t \alpha_j \phi_i(k, 2-j), \cdots, \sum_{j=0}^{p-1} \Delta t \alpha_j \phi_i(k, N+1-j))^T$, the OFR algorithm can be outlined as follows

Step 1

$$I_1 = I_M = \{1, \cdots, M\}$$

$$W_i = \varphi_i, \hat{b}_i = \frac{W_i^T Z}{W_i^T W_i} \tag{42}$$

$$l_1 = arg\max_{i\in I_1}(\hat{b}_i^2\frac{W_i^TZ}{Z^TZ}) = arg\max_{i\in I_1}(ERR_i) \tag{43}$$

$$W_1^0 = W_{l_1}, c_1^0 = \frac{W_1^{0T}Z}{W_1^{0T}W_1^0} \tag{44}$$

$$a_{1,1} = 1 \tag{45}$$

Step $j, j > 1$

$$I_j = I_{j-1}\backslash l_j - 1 \tag{46}$$

$$W_i = \varphi_i - \sum_{k=1}^{j-1}\frac{W_k^{0T}Z}{W_k^{0T}W_k^0}W_k^0, \hat{b}_i = \frac{W_i^TZ}{W_i^TW_i} \tag{47}$$

$$l_j = arg\max_{i\in I_j}(\hat{b}_i^2\frac{W_i^TZ}{Z^TZ}) = arg\max_{i\in I_j}(ERR_i) \tag{48}$$

$$W_j^0 = W_{l_j}, c_j^0 = \frac{W_j^{0T}Z}{W_j^{0T}W_j^0} \tag{49}$$

$$a_{k,j} = \frac{W_k^{0T}\varphi_{l_j}}{W_k^{0T}W_k^0}, k = 1, \cdots, j-1. \tag{50}$$

The procedure is terminated at the M_s-th step when the termination criterion

$$1 - \sum_{i=1}^{M_s} ERR_i < \rho \tag{51}$$

is met, where ρ is a designated error tolerance, or when a given number of terms in the final model is reached.

The estimated coefficients are calculated from the following equation

$$\tilde{\beta} = \begin{pmatrix} \tilde{\beta}_{l_1} \\ \tilde{\beta}_{l_2} \\ \vdots \\ \tilde{\beta}_{l_{M_s}} \end{pmatrix} = \begin{pmatrix} 1 & a_{1,2} & \cdots & a_{1,M_s} \\ 0 & 1 & \vdots & a_{2,M_s} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}^{-1} \begin{pmatrix} c_1^0 \\ c_2^0 \\ \vdots \\ c_{M_s}^0 \end{pmatrix} \tag{52}$$

and the selected terms are $\varphi_{l_1}, \cdots, \varphi_{l_{M_s}}$.

Note that the prediction error $w(k,n)$ are not measurable *a priori* and therefore must be estimated as follows

$$\tilde{w}(k,n+1) = z(k,n+1)P_z(k)\tilde{\beta} \tag{53}$$

Using this estimated noise sequence and the model variables, repeat the above algorithm to identify the terms and parameters for the deterministic part $P_z\beta$ and uncertain part $Q_z(k)\theta$ until the final residuals become unpredictable and the parameters do not change anymore. Then the final model comprises of a deterministic part (without errors as variables) and an uncertain part (including error variables). By removing the uncertain part, a final deterministic model can be obtained. As mentioned earlier, the estimation to the parameters is generally biased so that these estimated parameters must be refined. In the following section, a bootstrap algorithm is introduced as one solution to this problem.

5 The Bootstrap algorithm for parameter estimation

The bootstrap method, which is similar to the jackknife method, is a Monte Carlo simulation based statistical technique for estimating standard errors and bias, and requires a minimum number of mild assumptions and does not require a large number of samples.

Assume that a random sample $\xi = (\xi_1, \cdots, \xi_n)$ from an unknown probability distribution F has been observed, and an estimate for a parameter $\theta = \pi(F)$ is of interest on the basis of the observation ξ. A bootstrap method is associated with the notion of a bootstrap sample. Let $\tilde{F}$ be an empirical distribution function of F, which is defined to be

$$\xi \prec F \overset{e.d.}{\to} \xi^* \prec \tilde{F} \tag{54}$$

where $\xi^* = (\xi_1^*, \cdots, \xi_n^*)$, the symbol $\xi \prec F$ implies that the random sample ξ comes from the distribution F, the symbol $\overset{e.d.}{\to}$ implies that $\tilde{F}$ assigns to a set $\Omega = \Omega(F,\xi)$ in the sample space of ξ with an empirical probability $p = 1/n$ in the sense that for any $i, j \in \{1, 2, \cdots, n\}$

$$Pr\{\xi_i^* = \xi_j\} = \frac{1}{n} \tag{55}$$

The star notation indicates that ξ^*, which is referred to as a bootstrap sample of size n, is not the actual data set ξ but rather a randomized, or resampled, version of ξ. From the plug-in principle, the plug-in estimate of $\theta = \pi(F)$ can be calculated by

$$\tilde{\theta} = \pi(\tilde{F}) \tag{56}$$

This means that the function $\theta = \pi(F)$ of the unknown distribution F can be estimated using the same function of the empirical distribution $\tilde{F}$ with the form of (56). In practice the function π is unknown, and the unknown parameter θ is often approximated using a statistical estimator $\tilde{\theta} = s(\xi)$ based on the observed data. The bootstrap replication of $\tilde{\theta}$ is therefore given by $\tilde{\theta}^* = s(\xi^*)$. The function $s(\cdot)$ here can be any statistic of interest, for example, the mean, the standard derivation, or the second moment of the sampled data.

To obtain an efficient bootstrap estimate, it is often required to generate a number of independent bootstrap samples, say resampling the original observed data ξ L times, where the number L is ordinarily chosen in the range 25-200 and $L = 50$ is often enough to give a good estimate (Efron and Tibshirani 1993). Assume that the bootstrap replication at step l is $\tilde{\theta}^*(l) = s(\xi^*(l))$ for $l = 1, 2, \cdots, L$. The bootstrap estimate $\tilde{\theta}^*$ can then be obtained by synthesizing all the individual bootstrap replications $\tilde{\theta}^*(l) = s(\xi^*(l))$ via a given function g with the form $\tilde{\theta}^* = g(\theta^*(1), \cdots, \theta^*(L))$. Readers are referred to Efron and Tibshirani (1993) for more details of the bootstrap methods.

The basic idea of applying bootstrap methods for refining the identified parameters for continuous spatio-temporal systems is outlined below.

After the significant variables and model terms have been correctly selected, and that the system output can be expressed using a linear-in-the-parameters model form given by

$$z(k, n+1) = z(k, n) + \sum_{i=1}^{M} \beta_i \left(\sum_{j=1}^{p} \Delta t \alpha_j \phi_i(z(k, n+1-j), \cdots, z^{(l)}(k, n+1-j))\right) = z(k, n+1) - P_z(k)\beta \tag{57}$$

For given a set of data of length N at some spatial location, the estimates for the unknown parameter β can then be calculated using a standard least-squares algorithm. The model residuals (one-step-ahead prediction errors) for any spatial location k can be estimated as

$$\varepsilon(k) = Z(k) - \tilde{Z}(k) = Z(k) - P_z(k)\tilde{\beta} \tag{58}$$

where $\varepsilon(k) = (\varepsilon(k, 1), \cdots, \varepsilon(k, N))^T$ and $Z(k) = (z(k, 1), \cdots, z(k, N))^T$. By performing the bootstrap resampling theory, a bootstrap sample $\{\varepsilon^*(k, 1), \cdots, \varepsilon^*(k, N)\}$ will be obtained. The bootstrap response (output) of the system is defined to be

$$Z^*(k) = Z(k) + \mu\varepsilon^*(k) \tag{59}$$

where μ is a weighting coefficient satisfying $0 < \mu \leq 1$. Note that the regression matrix $P(k)$ contains the spatial derivatives of $z(k, n)$, to apply the bootstrap method they must be updated too. This is achieved according to the following formula

$$Z^{(i)*}(k) = Z^{(i)}(k) + \delta \frac{\varepsilon^*(k)}{\Delta x^i} \tag{60}$$

where $Z^{(i)}$ are the estimated or smoothed i-th spatial derivatives and Δx is the spatial sampling interval. Using (59) and (60), the updated $P_z^*(k)$ can be obtained. Corresponding to the above bootstrap observations, a bootstrap estimate of β can be obtained by applying standard least square algorithm again with the bootstrap output vector Z^* and the corresponding bootstrap regression matrix $P_z^*(k)$. Denote this bootstrap estimate by $\tilde{\beta}^{(1)}$. This procedure will be repeated L times. Let $\tilde{\beta}^{(l)}$ be the bootstrap estimate at step l for $l = 1, 2, \cdots, L$. Define

$$\tilde{\beta}^* = (\tilde{\beta}_1^*, \cdots, \tilde{\beta}_M^*)^T \tag{61}$$

where $\tilde{\beta}_i^* = 1/L \sum_{l=1}^{L} \tilde{\beta}^{(l)}$. The bootstrap estimate (61) will be used to replace the original estimate $\tilde{\beta}$.

The obtained final model and parameters need to be assessed. A commonly used approach to check the validity of the identified model is to use higher order statistical correlation analysis (Billings and Voon 1986, Billings and Zhu 1994). An alternative is to check both the short and the long term predictive ability of the model or some quantitative invariants such as Lyapunov exponents and correlation dimensions etc. If the performance of the obtained model is not satisfied, the whole identification procedure needs to be repeated to improve the performance.

6 Numerical simulation and analysis

6.1 Example 1: A reaction-diffusion system

Consider the following nonlinear reaction-diffusion system

$$\begin{aligned} \frac{\partial y_1(x,t)}{\partial t} &= d_1 \frac{\partial^2 y_1(x,t)}{\partial x^2} + y_1(x,t)^2 - y_1(x,t)^3 - y_2(x,t) \\ \frac{\partial y_2(x,t)}{\partial t} &= d_2 \frac{\partial^2 y_2(x,t)}{\partial x^2} + \delta y_1(x,t) - \gamma y_2(x,t) \end{aligned} \tag{62}$$

with $x \in \Omega = [0, 1]$, and initial conditions

$$y_1(x, 0) = y_2(x, 0) = sin(\pi x) \tag{63}$$

and Dirichlet boundary conditions, that is, $y_1(0, t) = y_1(1, t) = y_2(0, t) = y_2(1, t) = 0$.

Variables	Terms	Estimates	ERR	STD
$y_1(k,n+1) - y_1(k,n)$	$y_2(k,n)$	-9.9965e-01	9.8306e-01	3.5675e-03
	$y_1(k,n)^2$	9.9914e-01	1.0310e-02	2.6301e-03
	$y_1(k,n)^3$	-9.9925e-01	6.6341e-03	1.8192e-05
	$y_1''(k,n)$	5.8700e-04	2.8424e-07	6.2970e-06
	$y_2(k,n)^2$	-3.3997e-05	2.6646e-08	4.4351e-06
	$y_2(k,n)y_2'(k,n)$	7.2338e-05	7.7779e-09	3.3983e-06
	$y_1(k,n)y_2(k,n)^2$	1.8113e-05	3.0238e-09	2.9070e-06
$y_2(k,n+1) - y_2(k,n)$	$y_1(k,n)$	3.9987e+01	9.9913e-01	6.1911e-03
	$y_2(k,n)$	2.0002e-01	9.7398e-04	6.1450e-05
	$y_2''(k,n)$	6.1831e-04	8.8796e-08	1.3261e-05

Table 1: The terms and parameters of the final model for Example 1 from data without noise

For the purpose of identification using the proposed approach, the PDEs (62) with parameters $d_1 = d_2 = 0.0006188, \delta = 40$ and $\gamma = -0.2$, were numerically solved by linearised θ-methods (Ramos 1997) with the time step $\Delta t = 0.01$, space step $\Delta x = 0.02$, and $\theta = 1/2$. To test the proposed derivative estimation and signal denoising methods, white noise then was added to the output signals so that the SNR of the corrupted data is 37.7437DB for y_1 and 33.7121DB for y_2. The noisy data are plotted in Fig.(1) and Fig.(2).

For the purpose of comparison, the identification algorithm was applied for both data with and without noise. A set of 100 spatio-temporal observations randomly selected among the data set was used for the identification. In the simulation, the highest order of the derivatives with respect to the spatial variables was set to be 2. To estimate these spatial derivatives from the noisy data, the proposed denoising and derivative estimation methods were used using the following parameters: the resolution level for the B-spline expansion of the data was 3, the order of the B-spline function was set to be 6, and the hard thresolding value of the wavelet filter was 0.01. Typical denoised data and estimated derivatives are shown in Figs.(5-10).

The 3rd Adams-Bashforth integration formula was used and a polynomial expansion of order 3 for the nonlinear function f was used. The identified terms and parameters from noise-free data using the OFR are listed in Tables (1), where ERR denotes the Error Reduction Ratio and STD denotes the standard deviations.

It can be seen that the Estimates and ERR in Tables (1) suggests that the terms $y_2(k,n)^2$, $y_2(k,n)y_2'(k,n)$, $y_1(k,n)y_2(k,n)^2$ make insignificant contributions to the reduction of the total errors (note that the reduction ratios can be regarded as an equivalent representation of correlation coefficients) and therefore can be removed. After removing those three terms, the parameters are recalculated using the OFR algorithm again, which results in the following identified continuous spatio-temporal dynamical model for the noise-free case

Variables	Terms	Estimates	ERR	STD
$y_1(k,n+1)-y_1(k,n)$	$y_2(k,n)$	-9.9957e-01	9.7671e-01	4.0821e-03
	$y_1(k,n)^2$	1.0075e+00	1.7012e-02	2.6256e-03
	$y_1(k,n)^3$	-1.0132e+00	6.2687e-03	1.1782e-04
	$y_1''(k,n)$	2.9715e-03	7.7078e-07	1.1422e-04
$y_2(k,n+1)-y_2(k,n)$	$y_1(k,n)$	4.0149e+01	9.9902e-01	6.5227e-03
	$y_2(k,n)$	1.9398e-01	9.7732e-04	2.9484e-04
	$y_2''(k,n)$	9.5253e-05	1.8226e-07	2.8099e-04

Table 2: The terms and parameters of the final model for Example 1 from noisy data after removing the uncertain and insignificant terms

$$\begin{aligned}
\frac{\partial y_1(x,t)}{\partial t} &= 0.0005806\frac{\partial^2 y_1(x,t)}{\partial x^2} + 0.99867y_1(x,t)^2 - 0.99868y_1(x,t)^3 - 0.99966y_2(x,t) \quad (64)\\
\frac{\partial y_2(x,t)}{\partial t} &= 0.00061305\frac{\partial^2 y_2(x,t)}{\partial x^2} + 39.987y_1(x,t) + 0.19988y_2(x,t)
\end{aligned}$$

which indicates an excellent identified result.

In the case that the data contains noise, the terms and parameters of the final model from noisy data after removing the uncertain and insignificant terms are listed in Table (2). Comparing the results in Table (2) with the true model eqn. (62) it can be observed that some of the parameters are biased, in particular, the bias of the parameters corresponding to the second order spatial derivatives are quite large, although the terms selected are correct. These results show that the proposed approach did not change the correlations between variables but did introduce some corruption of the original signals through the filtering procedure. These parameters were then tuned using the bootstrap method with $L = 20$ and the obtained model is as follows

$$\begin{aligned}
\frac{\partial y_1(x,t)}{\partial t} &= 0.00031\frac{\partial^2 y_1(x,t)}{\partial x^2} + 1.0032y_1(x,t)^2 - 1.0194y_1(x,t)^3 - 0.9996y_2(x,t) \qquad (65)\\
\frac{\partial y_2(x,t)}{\partial t} &= 0.00048\frac{\partial^2 y_2(x,t)}{\partial x^2} + 40.0246y_1(x,t) + 0.2086y_2(x,t)
\end{aligned}$$

It can be observed that the bias of the parameters has now been significantly reduced.

6.2 Example 2: Two-dimensional Swift-Hohenberg equation

In this section, the two dimensional Swift-Hohenberg equation is considered (Swift and Hohenberg 1977). The model is of the

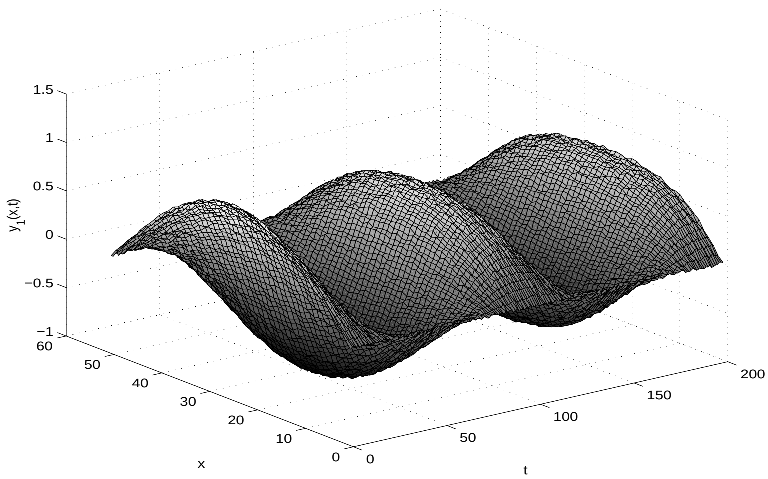

Figure 1: Example 1: Data $y_1(x,t)$

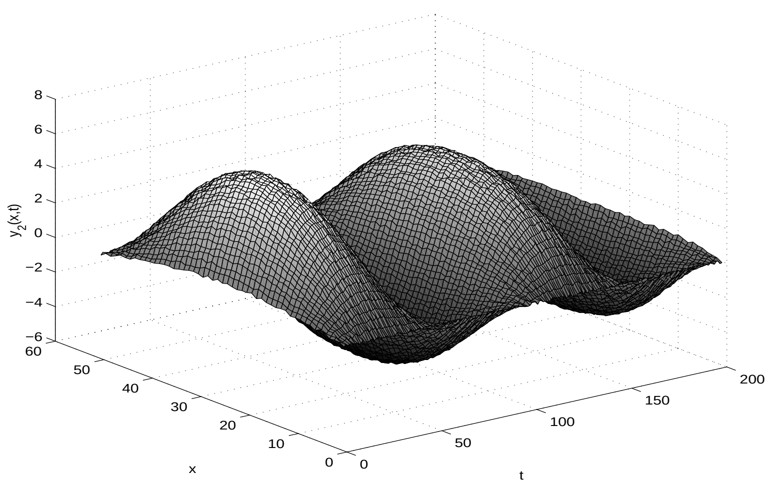

Figure 2: Example 1:Data $y_2(x,t)$

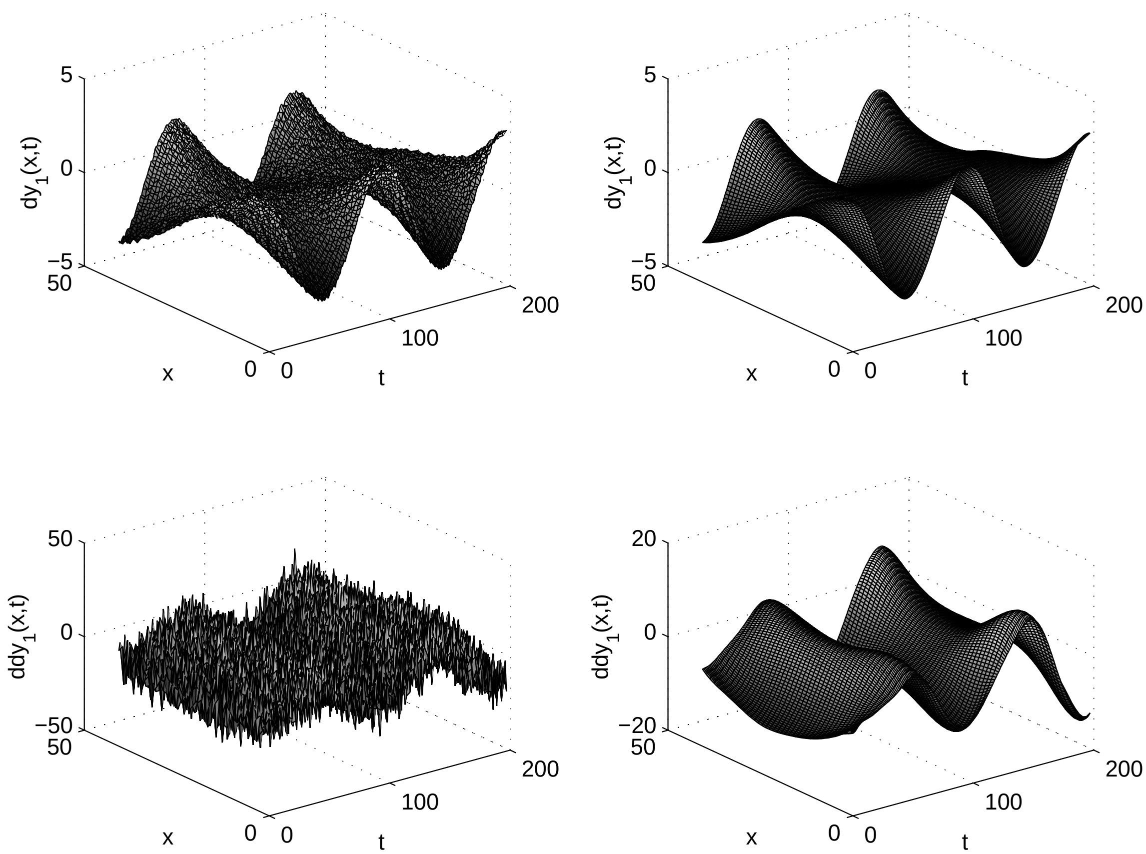

Figure 3: Example 1: The estimated spatial derivatives $y_1(x,t)$ (left: from finite difference method; right: estimated from the proposed method)

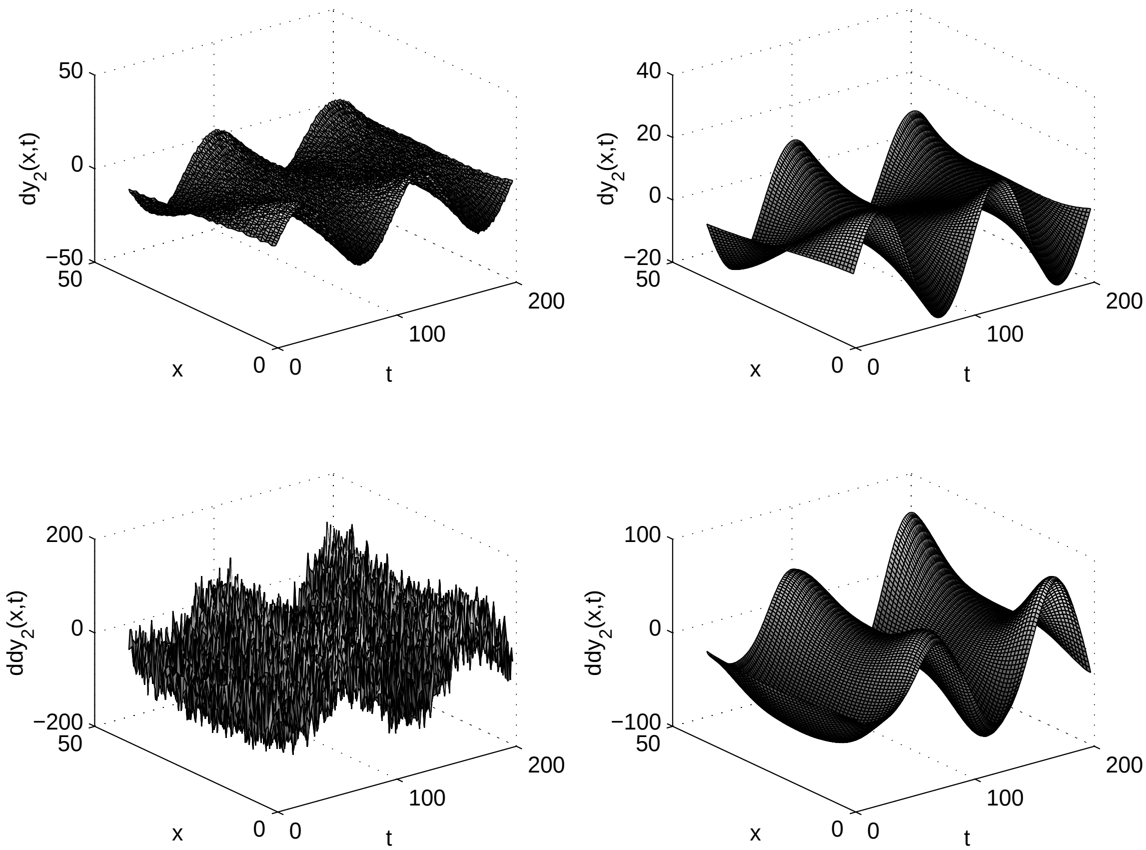

Figure 4: Example 1: The estimated spatial derivatives $y_2(x,t)$ (left: from finite difference method; right: estimated from the proposed method)

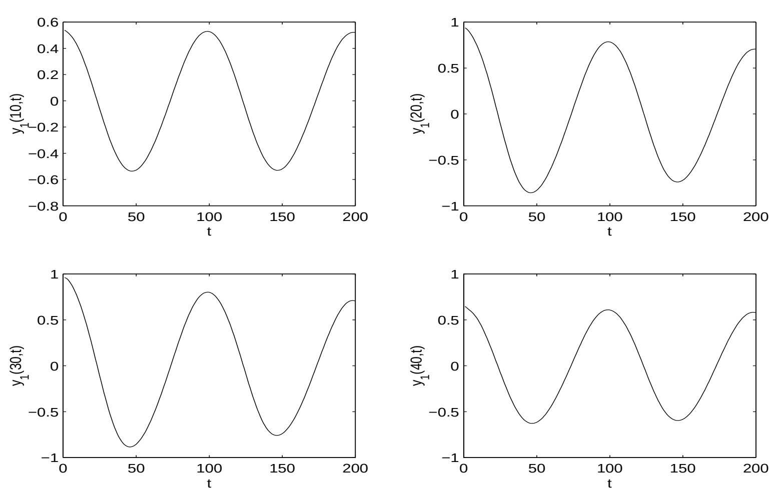

Figure 5: Example 1:Denoised $y_1(x,t)$

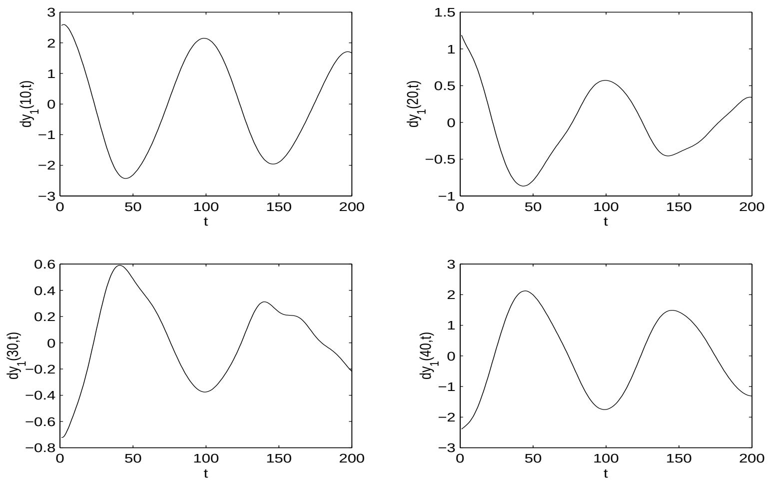

Figure 6: Example 1:Estimated first order spatial derivatives of $y_1(x,t)$

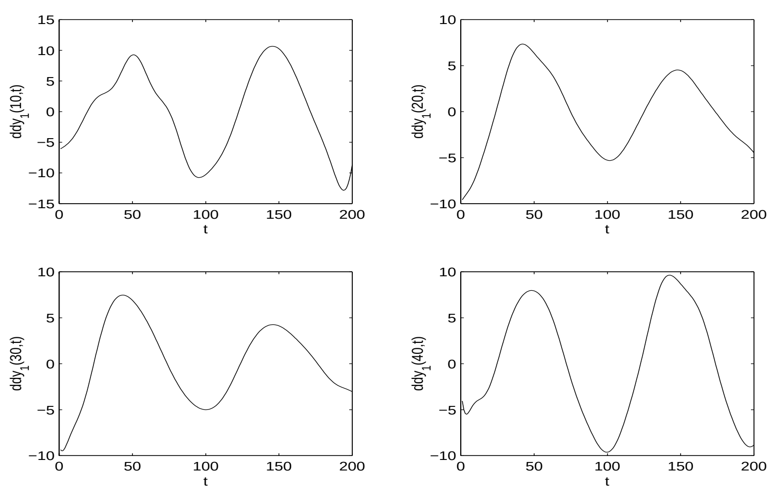

Figure 7: Example 1:Estimated second order spatial derivatives of $y_1(x,t)$

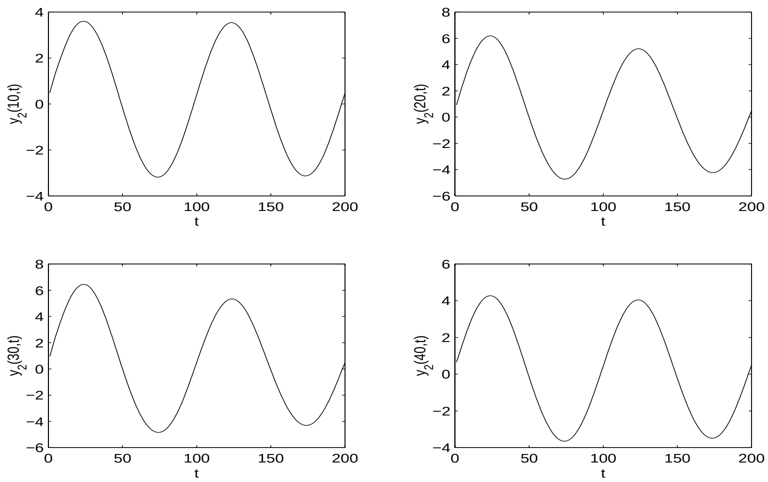

Figure 8: Example 1:Denoised $y_2(x,t)$

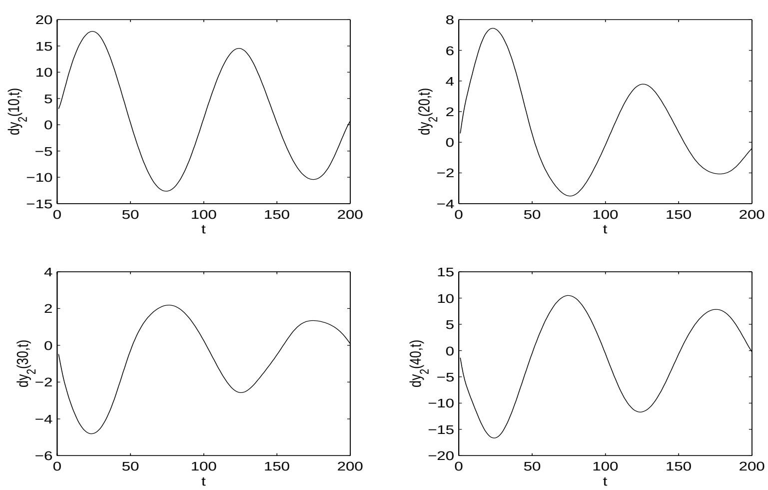

Figure 9: Example 1:Estimated first order spatial derivatives of $y_2(x,t)$

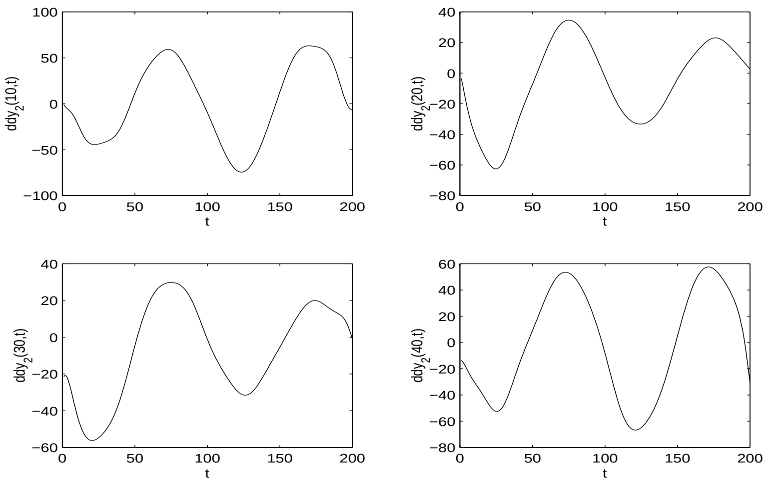

Figure 10: Example 1:Estimated second order spatial derivatives of $y_2(x,t)$

following form

$$\partial_t u = (r - (\nabla^2 + k^2)^2)u - u^3 = (r - k^4)u - u^3 - 2k^2(\partial_{xx} + \partial_{yy})u - (\partial_{xxxx} + \partial_{yyyy} + 2\partial_{xxyy})u \quad (66)$$

For the purpose of identification using the proposed approach, the PDEs (66) with parameters $r = 0.1, k = 1$, were numerically solved using a Runge-Kutta integration method with space step $\Delta x = \Delta y = 0.15$ and periodic boundary conditions. The initial conditions were chosen as uniformly distributed independent random numbers within the interval $[-10, 10]$. The obtained solution was sampled at a time interval$\Delta t = 0.01$. Some snapshots are plotted in Fig.(11).

A set of 100 spatio-temporal observations randomly selected among the data set with additive white noise of SNR $= 62.7437$DB was used for the identification. In the simulation, the highest order of the derivatives with respect to the spatial variables was set to be 4. To calculate these spatial derivatives using the proposed denoising and derivative estimation methods, the following parameters were used: the resolution level for the B-spline expansion of the data was 4, the orders of the B-spline functions were set to be 6 and 7, and the hard thresolding value of the wavelet filter was 0.001. The 4th Adams-Bashforth integration formula was used and the polynomial expansion of order 3 of the nonlinear function f was used. The identified terms and parameters using estimated derivatives and the orthogonal least squares algorithm are listed in Table (3), where ERR denotes the Error Reduction Ratio and STD denotes the standard deviations. After removing the insignificant terms according to the values of the ERRs and the estimates and applying the bootstrap algorithm, the final PDE model was obtained as follows

$$\partial_t u = -0.91269u - 0.99852u^3 - 1.9461\partial_{xx}u - 1.9517\partial_{yy}u - 0.98807\partial_{xxxx}u - 0.988\partial_{yyyy}u - 1.9761\partial_{xxyy}u \quad (67)$$

Terms	Estimates	ERR	STD
$u_{xx}(x,y,t)$	-1.9537e+00	4.8455e-01	5.1899e-02
$u_{yy}(x,y,t)$	-1.9209e+00	2.3845e-01	3.8081e-02
$u_(x,y,t)$	-7.0723e-01	2.4130e-02	3.6330e-02
$u_{xxyy}(x,y,t)$	-1.9960e+00	1.0319e-02	3.5585e-02
$u_{yyyy}(x,y,t)$	-9.9479e-01	2.8556e-02	3.3428e-02
$u_{xxxx}(x,y,t)$	-9.9787e-01	9.1663e-02	2.5307e-02
$u_(x,y,t)^3$	-1.0221e+00	8.3810e-03	2.4423e-02
$u_{yy}(x,y,t)^2u_{yyyy}(x,y,t)$	-4.0302e-03	8.9614e-05	2.4413e-02
$u_{yy}(x,y,t)u_{yyyy}(x,y,t)u_{xxyy}(x,y,t)$	-1.9533e-03	4.4761e-05	2.4408e-02
$u_{yy}(x,y,t)u_{yyyy}(x,y,t)^2$	-9.5217e-04	1.7198e-04	2.4390e-02
$u_{xxxx}(x,y,t)u_{yy}(x,y,t)u_{yyyy}(x,y,t)$	-1.0148e-03	5.1941e-05	2.4384e-02
$u_{xxxx}(x,y,t)u_{yy}(x,y,t)u_{yyyy}(x,y,t)$	-6.4276e-03	4.7234e-05	2.4379e-02
constant	3.9068e-02	2.7782e-05	2.4378e-02

Table 3: The terms and parameters of the final model

From the simulation results of Examples 1 and 2, the following observations can be made

- The proposed approach works better for data with a higher SNR. This is because the filtering procedure can introduce some error on the data and this error is generally larger for data with a lower SNR, and this is therefore more likely to change the correlations between the variables. The OFR algorithm basically works according to the correlation coefficients of the data and may not produce the correct structure of the model if the correlations have been changed. In this case, the structure and parameters of the obtained PDE model can be quite different from the original although the model itself may work well for predictions.
- It has been found that there is an advantage of the proposed derivative estimation approach over a finite difference approximation method. That is that the proposed approach works better with a high sampling frequency because the estimation can be performed in a higher resolution level which will produce more accurate approximation when noise is present.
- Experience on simulation studies shows that the data for identification have to have a sufficient variability in both the spatial and time domains. In this way the identification results are much better than that just using the data from one specific spatial location. This is why the data for identification in our simulations were generated randomly among space and time. This is to be expected and is equivalent to persistently exciting data concepts in temporal model estimation.

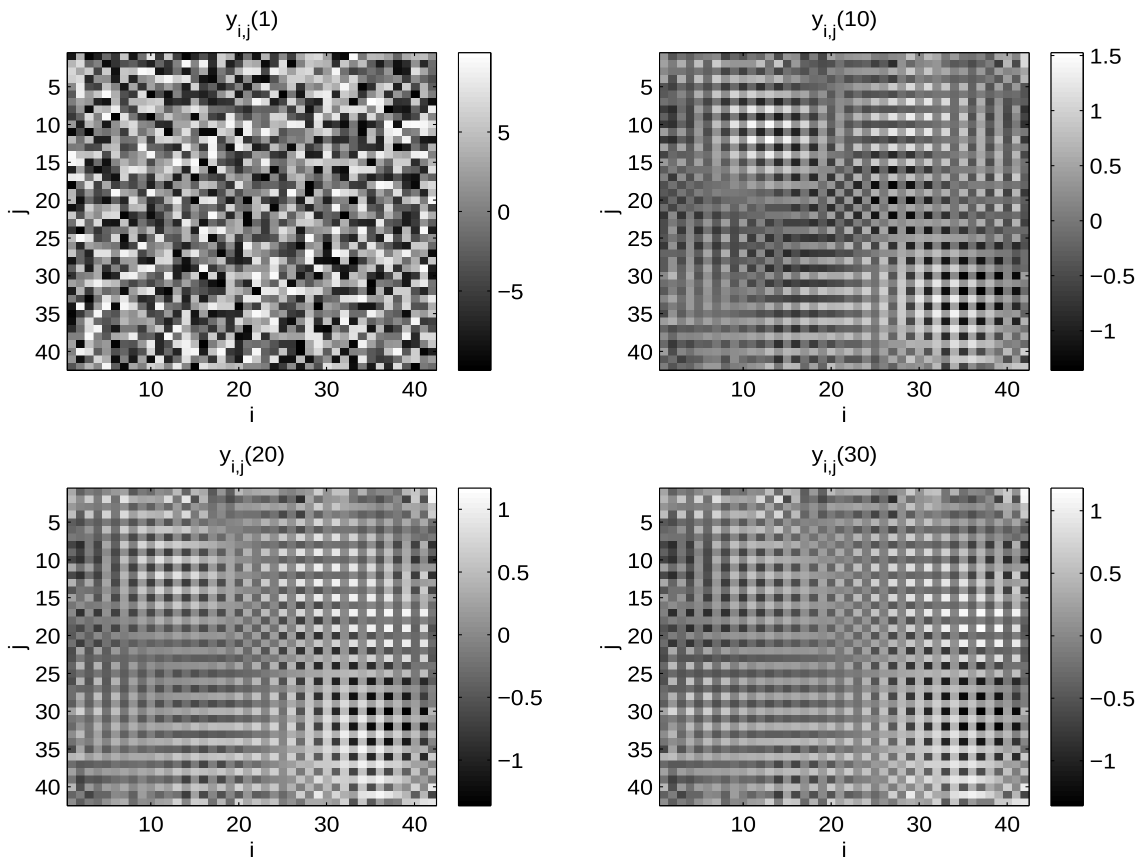

Figure 11: Example 2: Some snapshots

7 Conclusions

A new approach for the identification of both the model terms or structure and the unknown parameters in PDE models of continuous spatio-temporal dynamical systems has been introduced. It has been shown that by combining the Adams integration and the OFR algorithm, a system of PDEs for the underlying continuous spatio-temporal system can be obtained. By using B-spline wavelet multi-resolution analysis and a B-spline biorthogonal wavelet filtering technique, the measured signals are denoised and represented as a series of B-spline functions. Then the higher derivatives can be calculated from this series expansion directly. The proposed method was tested on simulated Data and was shown to perform very well.

8 Acknowledgement

The authors gratefully acknowledge financial support from EPSRC (UK).

References

[1] Billings, S. A., Chen, S., and Kronenberg, M. J., (1988) Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator, *Int. J. Contr.*, Vol. 49, pp. 2157-2189.

[2] Billings, S. A., and Voon, W. S. F. (1986) Correlation based model validity tests for nonlinear models, *International Journal of Control*, Vol. 40, No. 1, pp. 235-244.

[3] Billings, S. A. and Zhu, Q. M. (1994) Nonlinear model validation using correlation tests, *International Journal of Control*, Vol. 60, No. 6, pp. 1107-1120.

[4] Chen, S., Billings, S. A., and Luo, W., (1989) Orthogonal least squares methods and their application to non-linear system identification, *International Journal of Control*, Vol. 50, No. 5, pp. 1873-1896.

[5] Chui, C. K., (1992) *An introduction to wavelets*, Academic Press, New York.

[6] Coca, D. and Billings, S. A., (2000) Direct parameter identification of distributed parameter systems, *Int. J. Systems Sci.*, Vol. 31, No.1, pp. 11-17.

[7] Efron, B. and Tibshirani, R. J., (1993) *An introduction to the Bootstrap*, Chapman & Hall, New York.

[8] Fioretti, S. and Jetto, L., (1989) Accurate derivative estimation from noisy data: a state-space approach, *Int. J. Systems Sci.*, Vol. 20, No. 1, pp.33-53.

[9] Mallat, S. G., (1989) A theory of multiresolution signal decomposition: the wavelet representation, *IEEE Pattern Anal. & Machine Intelligence*, Vol. 11, pp.674-693.

[10] Niedzwecki, J. M. and Liagre, P. -Y. F., (2003) System identification of distributed-parameter marine riser models, *Ocean Engineering*, Vol. 30, No. 11, pp. 1387-1415.

[11] Packard, N., Crutchfield, J., and Farmer, D., (1980) Geometry from a time series, *Phys. Rev. Lett.* , Vol. 45, pp.712-715.

[12] Phillipson, G. A., (1971) *Identification of distributed systems*, American Elsevier, New York.

[13] Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., (1992) *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed. Cambridge, England: Cambridge University Press.

[14] Ramos, J. L., (1997) Linearisation methods for reaction-diffusion equations: 1-D problems, *Applied Mathematics and Computation*, Vol.88, pp. 199-224.

[15] Sauer, T., Yorke, J. A., and Casdagli, M., (1991) Embedology, *Journal of Statistical Physics*, Vol. 65, pp. 579-616.

[16] Swift and Hohenberg, (1977) Hydrodynamic fluctuations at convective instability, *Phys. Rev. A*, Vol.15, No.1, pp. 319-328.

[17] Takens, F., (1981) Detecting strange attractors in turbulence, in *Lecture Notes in Mathematics*, No. 898, pp. 366-381, Springer-Verlag.

[18] Travis, C. C. and White, H. H., (1985) Parameter identification of distributed parameter systems, *Mathematical Bioscience*, 77, pp. 341-352.

[19] Voss, H., Bunner, M. J., and Abel, M., (1998) Identification of continuous, spatiotemporal systems, *Physical Review E*, Vol. 57, No.3, pp. 2820-2823.