

Iowa State University

From the Selected Works of Simon Laflamme

October, 2010

Self-Organizing Inputs for Adaptive Neurocontrol of Large-Scale Systems

Simon Laflamme, *Massachusetts Institute of Technology*

Jerome J. Connor, *Massachusetts Institute of Technology*



Available at: https://works.bepress.com/simon_laflamme/14/

Self-Organizing Inputs for Adaptive Neurocontrol of Large-Scale Systems

Simon Laflamme* and Jerome J. Connor

Department of Civil and Environmental Engineering
Massachusetts Institute of Technology
77 Massachusetts Avenue, Cambridge, MA 02139

ABSTRACT

Black-box modeling for system identification and control can be constrained by the geometric scale of a plant. Those constraints arise from a technical or economical issue in obtaining input-output data sets for training purposes. A solution is sequential adaptation, which must be achieved in real time in the case of control systems, because a prescribed level of performance is required while external excitations are occurring. A major challenge is the proper selection of inputs that would lead to a quick and optimal adaptation.

This paper proposes a new method for selection of inputs in black-box modeling. The method, termed the Self-Organizing Inputs (SOI) algorithm, consists of organizing the input space sequentially and in real-time. This self-organization leads to an enhanced representation of the system dynamics, which improves adaptation convergence and performance. The algorithm is designed based on time series analysis, and on the theory that dynamic systems can be topologically reconstructed using a single observation delayed in time, provided the choice of time delay and embedding dimension is appropriate. The SOI feature sequentially identifies the proper time delay and dimension, and adapts the required dynamic inputs for the black-box model. That allows a representation to self-organize, self-adapt, and identify or control by utilizing the essential dynamics of the plant captured via limited state measurement.

The SOI algorithm is applied to an adaptive neurocontroller to simulate control of an existing structure located in Boston, Massachusetts. The performance of the SOI neurocontroller is assessed for acceleration mitigation of harmonic and wind excitations, using a semi-active control strategy. Results show that the proposed neurocontroller performs well at vibration mitigation for both strategies, using limited acceleration measurements and limited knowledge about the dynamic properties of the large-scale structure. Its performance is similar to a fixed inputs strategy in the case of the harmonic excitation, and better in the case of a wind mitigation. In both cases, the SOI algorithm results in a significantly smaller network. It is also demonstrated that the proposed method can be used for nonstationary processes.

1 INTRODUCTION

The level of uncertainty relative to the fundamental dynamic properties increases with the geometric scale of the controlled plant. Large members, static loads, and connections can be modeled, but the construction process of large civil and mechanical systems results in dynamic uncertainties of significant ranges. Moreover, there exist sources of nonlinearity and measurement noises that might not be modeled, such as the actuator dynamics, sensor inaccuracies, and time delays induced in the control loop

* Corresponding author (laflamme@mit.edu)

[1, 2, 3]. The scale of such systems results in technically challenging and economically nonviable system identification schemes. Adaptive control strategies have been proposed to cope with large uncertainties [4, 5, 6]. Intelligent controllers, such as fuzzy logic systems and neural networks, are a special class of adaptive controllers. They are characterized by their capacity to create a functional representation based on training from input-output data sets. The main challenge is to collect those sets, and to use an appropriate training algorithm to ensure a quick convergence of the controlled system for a required level of performance. When such data sets are not available and a control performance is necessary, which is often the case for civil structures exposed to random wind loads and earthquake excitations, a sequential and real-time learning strategy is required.

During sequential training for a multi degree-of-freedom (MDOF) system, the available inputs can be quite large, and the challenge is to select the appropriate inputs that will be fed in the black-box model. The problem of input selection for black-box modeling has been overlooked for a long time [7]. Intuitively, the selected inputs need to contain the essential dynamics of a system by selecting the appropriate states, time delay, and embedding dimension [8]. It follows that the selection of inputs for an uncertain system is achieved arbitrarily, which may lead to a sub-optimal control system. A strategy has to be developed in order to obtain a better representation of the system dynamics from the inputs.

This paper presents a strategy for sequential and real-time input selection for black-box systems. The algorithm, termed the self-organizing inputs (SOI) algorithm, studies a sliding window of the time series of a single input and output, decides on the optimal dynamic inputs, and adapts the black-box input space. It results that the black-box model itself is optimized, as the system essential dynamics are optimally represented in real-time. Moreover, such strategy allows the model to be more responsive to local changes in the system dynamics. It is based on Takens embedding theorem [9], which states that the delayed time measurements of a single observation can define an entire autonomous dynamic system.

The delay embedding theorem has been applied in many fields for model prediction, system identification, control, and more specifically, for input selection of neural networks. Principe *et al.* [10] used local nonlinear embedding maps with a self-organizing mapping neural network, and applied it to system identification and control. Plagianakos & Tzanaki [11] used a neural network to predict an earthquake excitation selecting inputs based on the embedding theorem. Walker *et al.* [12] utilized the same strategy to design a radial-basis model for modeling of an electronic circuit with dynamic effect. Leung *et al.* [13] used a radial basis function network whose topology was optimized using the cross-validated subspace method. Széliga *et al.* [14] used a neural network to model a slowly varying nonstationary system. Shen *et al.* [15] proposed a predictive control scheme for chaotic systems using a neural network to make predictions. daSilva *et al.* [7] forecasted short term electricity loads, and noted the paramount importance of a proper input selection for short-term forecast. Zolock & Greif [16] used a time-delay neural network to predict wheel/rail responses of rail vehicles.

In this paper, it is demonstrated that the SOI algorithm can be applied online, and also to nonstationary systems. The SOI algorithm is applied in the context of an adaptive neurocontroller. The proposed method has the powerful advantage of taking the input selection out of the black-box design process, and it can be applied sequentially and in real time.

The paper is organized as follows. Section 2 describes the SOI algorithm. In that section, the embedding theorem is reviewed, and extended to nonstationary and MDOF systems shown. Section 3 describes the adaptive neurocontroller to be used along with the SOI algorithm in the simulations. A simple tracking example of a nonlinear function is shown. Section 4 simulates the algorithm on an existing structure located in Boston, MA. Simulations include harmonic and wind excitations controlled with a semi-active device. It is also demonstrated that the assumption of local stationarity holds. Section 5 concludes the paper.

2 SELF-ORGANIZING INPUTS ALGORITHM

Consider the following dynamic system:

$$\begin{aligned}x(k+1) &= f(x(k), u(k)) \\u(k+1) &= g(x(k), u(k)) \\y(k) &= h(x(k), u(k))\end{aligned}\tag{1}$$

where x is a state, u is an input, y is an observation, k is a discrete time step, and f, g, h are functions. A black-box model aims at finding an approximate representation of a function using measurable observations. This functional representation can be achieved using numerous system identification tools, such as fuzzy logic, neural networks (including radial basis function and wavelet networks), polynomial models, and rational models [17, 18]. For instance, the representation of the state dynamics f in (1) can be approximated in term of a single-layer wavelet neural network [19]:

$$\Psi(x(k), u(k)) = \sum_{i=1}^h \alpha_i \phi_i(x(k), u(k))\tag{2}$$

where h is the number of nodes in the network, α are the nodal weights, and ϕ is a wavelet function. This representation is designed or trained based on input-output data sets, and is therefore dependant on the selection of those data. Some systems do not have sets that are available a priori. In that case, training can be achieved sequentially. When sequential training is utilized, the choice of outputs is often trivial, as it corresponds to a precise identification or control objective. However, the choice of inputs can be more complex. Some dynamic systems are characterized by a significant number of degree-of-freedoms or measurable states, and the system designer must decide, prior to the training period, which available inputs should be used in the input-output sets for the training process. This paper proposes a solution for the input selection, the SOI algorithm. The algorithm has been designed to sequentially select the inputs of adaptive black-box controllers, where the representation $\tilde{g}(y(k), u(k))$ is adaptive and maps the required force output in function of the observations of the dynamic system:

$$u(k+1) = \tilde{g}(y(k), u(k), t)\tag{3}$$

The SOI algorithm is inspired by Takens Embedding Theorem [9]. The Embedding Theorem states that the phase space of autonomous dynamic systems can be diffeomorphically mapped to another phase space constructed from the time delayed measurements of a single observation. This theorem is applicable offline, because a study of the output time series is necessary to select the proper time delay of the observation, along with its appropriate embedding dimension. It what follows, the Embedding theorem is reviewed and its extension to nonautonomous stationary systems explained. It will be demonstrated that the SOI algorithm is applicable for nonstationary multi degree-of-freedom (MDOF) systems.

2.1 The Embedding Theorem

Takens [9] showed in 1980 that the phase-space of an autonomous dynamic system $x(k+1) = f(x(k))$ in a manifold \mathcal{M} can be diffeomorphically mapped to another phase-space constructed from a single measurement. In other terms, a one-to-one map exists between both phase-space, and all of the coordinate-independent properties are preserved. Fig. 1 illustrates the concept. In the figure, the phase-space reconstructed from the measurements of the unknown system is geometrically equivalent to the phase-space of the unknown system.

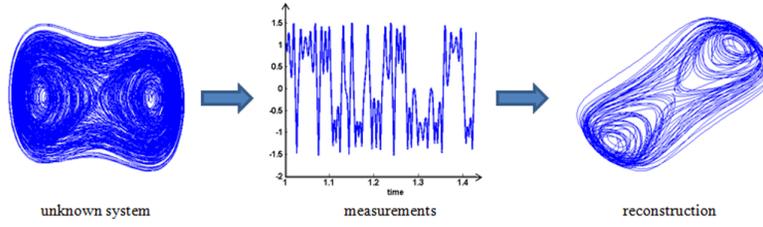


Figure 1: Illustration of the phase-space reconstruction.

The challenge is to form a vector ν , termed the *delay vector*, that will form the new phase-space. The delay vector is constructed using a time delay τ and embedding dimension d of a single measurement y :

$$\begin{aligned} \nu(k) &= [y(k) \quad y(k - \tau) \quad y(k - 2\tau) \quad \dots \quad y(k - (d - 1)\tau)] \\ &= \Phi(x(k)) \end{aligned} \quad (4)$$

where $\Phi : \mathcal{M} \rightarrow \mathbb{R}^d$. Several techniques exist to select an appropriate time delay τ and embedding dimension d . Intuitively, τ has to be large enough such that $y(k - \tau)$ gives additional information on the dynamics. For instant, if $\tau = 1$, the phase-space is collapsed on a 45° line, and $y(k) \approx y(k - 1)$. Thus, τ has to be large enough to unfold the phase-space, but not too large as the phase-space will fold back. The embedding d has to encompass all of the essential dynamics. However, a large d may contain too much redundant information, and might result in a degradation of precision. Techniques selected for the SOI algorithm will be discussed later.

Using (4), the functional f can be written:

$$\tilde{f} = \Phi \circ f \circ \Phi^{-1} \quad (5)$$

where \circ denotes function composition. Applying (4) to (5):

$$\begin{aligned} \tilde{f}(\nu(k)) &= \Phi \circ f \circ \Phi^{-1}(\nu(k)) \\ &= \Phi \circ f \circ \Phi^{-1}(\Phi(x(k))) \\ &= \Phi \circ f(x(k)) \\ &= \Phi(x(k + 1)) \\ &= \nu(k + 1) \end{aligned} \quad (6)$$

shows the equivalence between f and \tilde{f} , and that the coordinate-independent properties are preserved. In essence, the theorem states that, given the observation $y(k)$, it is possible to predict future observations using a map constructed from $\nu(k)$.

In the last decade, the embedding theorem has been extended to a general class of nonautonomous systems with deterministic forcing [20], state-dependant forcing [21], and stochastic forcing [22]. Those theorems address nonautonomous systems by expanding the phase-space sufficiently until the system becomes autonomous. The forcing u is allowed to evolve on its own space $u \in \Sigma$, where Σ is a shift space associated with a shift map $\sigma : \Sigma \rightarrow \Sigma$ such that:

$$\sigma(u(k)) = u(k + 1) \quad (7)$$

Thus, the new system evolves in the skew product $\mathcal{T} : \mathcal{M} \times \Sigma \rightarrow \mathcal{M} \times \Sigma$:

$$\mathcal{T}(x(k), u(k)) = (f(x(k), u(0)), \sigma(u(k))) \quad (8)$$

The main implication of such transformation is that the phase-space can become infinite dimensional if ν depends solely on \mathcal{M} . Therefore, a restriction on the map Φ is imposed. This restriction is the inclusion of the inputs u along with the observations y , where ν becomes:

$$\begin{aligned}\nu &= [y(k) \quad y(k - \tau) \quad y(k - 2\tau) \quad \dots \quad y(k - (d - 1)\tau) \\ &\quad u(k) \quad u(k - \tau) \quad u(k - 2\tau) \quad \dots \quad u(k - (d - 1)\tau)] \\ &= \Phi(x(k), u(k))\end{aligned}\tag{9}$$

with $\Phi : \mathcal{T} \rightarrow \mathbb{R}^{2d}$. Monroig *et al.* [23] showed that multivariate observations could be used if inputs were to be unknown.

The embedding theorems previously described for general systems of the form (1) do not apply to nonstationary systems. The next subsections discuss how the SOI algorithm is used for nonstationary systems, and its application to MDOF systems.

2.2 Extension to Nonstationary Systems

The use of an adaptive controller inherently transforms systems of the form (1) into nonstationary systems:

$$\begin{aligned}x(k + 1) &= f(x(k), u(k), t) \\ u(k + 1) &= g(x(k), u(k), t) \\ y(k) &= h(x(k), u(k), t)\end{aligned}\tag{10}$$

This subsection discusses how the embedding theorem is adapted to use Φ as inputs for nonstationary representations.

Nonstationarity systems in the context of the embedding theorem have been studied for the last decade. Strategies include overembedding the delay vector [24, 25], using nonuniform embedding for multivariate time series [26], and using local maps for locally varying dynamics [27]. Here, the technique of local maps is used, with the assumption that each of those local maps are stationary or quasi-stationary. A map f can be written as the union of several local maps f_q for $q = 1, \dots, Q$ [10]:

$$f = \bigcup_{q=1, \dots, Q} f_q\tag{11}$$

It is assumed that an adaptive representation $\tilde{f}(x(k), u(k), t)$ can be divided into local stationary maps:

$$\tilde{f} = \bigcup_{q=1, \dots, Q} \tilde{f}_q(x(k), u(k))\tag{12}$$

The output can then be represented by another shift map ξ modified from (7) where $\xi : \Sigma \times \Xi \rightarrow \Sigma \times \Xi$ with:

$$\xi(u(k), t) = (u(k + 1), t)\tag{13}$$

Let the manifold \mathcal{M} be discretized locally following (11) such that $\mathcal{M} = \bigcup_{q=1, \dots, Q} \mathcal{M}_q$. The map ξ is assumed to be smooth and compact, as adaptive representations are also assumed to be smooth and

compact. The system is now allowed to evolve over the manifold \mathcal{W} where \mathcal{W} is globally defined as the skew product $\mathcal{W} : \mathcal{M} \times \Sigma \times \Xi \rightarrow \mathcal{M} \times \Sigma \times \Xi$, or locally defined as:

$$\mathcal{W} : \bigcup_{q=1, \dots, Q} \{\mathcal{M} \times \Sigma\}_q \quad (14)$$

Note that as $Q \rightarrow \infty$, $\Sigma \times \Xi \rightarrow \Xi$. Hence, for a large number of local maps Q , the same delay vector as (9) can be used to embed \mathcal{W} , given that the map Φ is allowed to smoothly evolve with time to follow the translation in Ξ :

$$\boldsymbol{\nu}(k) = \Phi(x(k), u(k), t) \quad (15)$$

with $\Phi : \mathcal{W} \rightarrow \mathbb{R}^{2d}$.

The assumption of local stationarity or quasi-stationarity will be assessed in Section 4 in the context of a nonstationary wind excitation. There exist many methods that aim at determining stationarity of a time series, but in the case of a finite time series, most methods can be unclear and ambiguous [28]. The responses of a MDOF system subjected to a wind excitation can be assumed to be mutually independent Gaussian stationary processes [29]. Since an adaptive controller is used, if one can show that within the last n observations that are used to create a local map q the variation of the controller is small such that $\tilde{f}_{t=0} \approx \tilde{f}_{t=n}$ so that (3) can be written:

$$u(k+1) \approx \tilde{f}(x(k), u(k)) \quad (16)$$

within a map q , then (10) can be assumed to be stationary. Hence, to determine the level of stationarity of local maps q , a map will be declared stationary if the variation of the representation \tilde{f} within a map q is below a certain threshold. The percentage of stationary maps in function of the threshold will be the stationarity index.

2.3 MDOF systems

This subsection reviews design considerations for MDOF systems, and shows that the embedding theorem holds given the condition that the observation is not taken at the null point of a dominant node. MDOF systems have several measurable nodes and states. Multivariate observations can be included in the delay vector to represent a dynamic system. This is achieved by scaling inputs that represent different states. Scaling prevents some states from having more importance in the neural network than others, and also leads to a numerically more robust network with improved convergence [8].

Nevertheless, implementing several output observations may result in a high dimensional space, and some black-box systems such as wavelet networks are not suited for such problems [30]. In the case of a system where the state being identified or controlled, here termed the *objective state* x_k , is different than the *observed state* x_j , the available measurements must be topologically equivalent to the objective state. Modal decomposition can be used to show that a map exists between two different degree-of-freedom of a system, provided that they are not null point nodes. The response of a MIMO system can be decomposed into modes:

$$\dot{\boldsymbol{x}} = \sum_{i=1}^n \dot{\tilde{q}}_i(t) \tilde{\boldsymbol{\psi}}_i \quad (17)$$

where n is the number of degree-of-freedom, \dot{q} is the modal coordinate, and $\tilde{\psi}_i$ is the i^{th} modal shape vector. Thus, the response of a single observation can written:

$$\dot{x}_j = \sum_{i=1}^n \dot{q}_i(t) \tilde{\psi}_{j,i} \quad (18)$$

where $\tilde{\psi}_{j,\bullet}$ is the vector of the j^{th} components of the modal shape vector. Assuming that the degree-of-freedom being mapped are not stationary nodes, a map g exists such that $\tilde{\psi}_{k,\bullet} = g(\tilde{\psi}_{j,\bullet})$ where k is another component of the modal shapes. Hence, the response of another state can be obtained:

$$\begin{aligned} \dot{x}_k &= \sum_{i=1}^n \dot{q}_i(t) \tilde{\psi}_{k,i} \\ &= \sum_{i=1}^n \dot{q}_i(t) g(\tilde{\psi}_{j,i}) \\ &= \sum_{i=1}^n g_i \left(\dot{q}_i(t) \tilde{\psi}_{j,i} \right) \\ &= g(f(x_j)) \end{aligned} \quad (19)$$

Therefore, if a map exists such that a state x_k can be mapped from the state x_j , then there exist a map from the delayed vector $\Phi(x_j)$ to the state x_k . In other words, provided that the observed state x_j and the objective state x_k are not null point nodes, then the objective state x_k can be represented using a delayed vector constructed from the observations on x_j .

Consequently, one needs to ensure that the measurement is not taken at the null point of a dominant mode. In practice, that measurement should be taken at the maximum value of a node node if possible. For instance, in the case of cantilever-like civil structures, where masses are installed in series and separated by stiffness and dashpot elements, it is known that the response is dominated by the first few modes. Typically, an observation taken at the top floor would satisfy the requirements mentioned above.

2.4 SOI algorithm

The SOI algorithm sequentially studies the last time series composed of the last n measurements of a single observation, and determines the delay vector (15) that would encompass the essential dynamics of the system. In other words, the inputs for the black-box representation are composed of the delayed measurements of the observations of an input and an output, with the time delay τ and embedding dimension d selected in real-time, based on the study of the time series of a measurement. This sequential and real-time selection is done smoothly, over a small map q , to agree with the above conditions for constructing (15).

The time delay τ is selected using the *mutual dependence* method, theory presented by [31] based on Shannon's information theory. Given a measurement $y(k)$, a second measurement $y(k - \tau)$ is taken, and the information gained from the new measurement is evaluated. The value of τ that leads to the optimal information is kept as the optimal τ . The embedding dimension d is selected using the false nearest neighbor (FNN) method. In the FNN method, a point that is a nearest neighbor in Euclidian term must remain a nearest neighbor if the embedding dimension is increased. If it is not the case, then the point is declared to be a false neighbor. The FNN strategy is thus to increase the embedding dimension until the percentage of FNN is below a satisfactory threshold. The algorithm presented by [32] is used for the selection of d .

Fig. 2 illustrates the SOI algorithm for the case of an adaptive controller. The time window memory is a sliding window using the last n observation of a state output, the signal, and the force output. The value for n is kept small to ensure quick computation time. The SOI algorithm takes those last n observations, and select τ and d as mentioned above. The delay vector ν is adapted accordingly, at each time step, in the neighborhoods $\tau \in [\tau - 1, \tau + 1]$ and $d \in [d - 1, d + 1]$ to ensure robustness of the black-box representation.

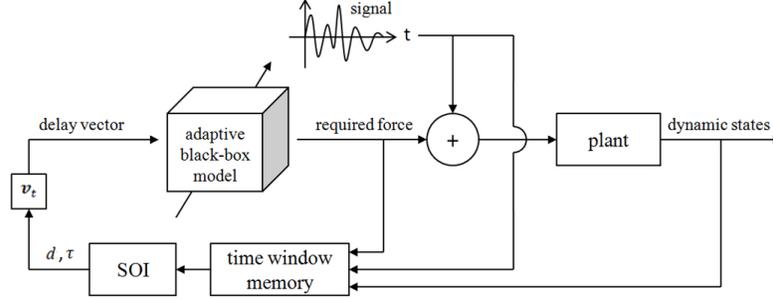


Figure 2: Schematic of the SOI algorithm.

3 ADAPTIVE WAVELET NEUROCONTROLLER

This section describes the black-box system to be used in the simulations, an adaptive wavelet neurocontroller. It consists of an adaptive single layer feedforward wavelet neural network of the form (2) where Phi returns the required force output. The wavelet functions ϕ are *mexican hat* wavelets:

$$\phi(x, t) = \left(1 - \frac{\|\nu(t) - \mu\|^2}{\sigma^2}\right) e^{-\frac{\|\nu(t) - \mu\|^2}{\sigma^2}} \quad (20)$$

where $\|\cdot\|$ denotes the Euclidean norm of the vector, and μ and σ are the center and bandwidth of the function respectively. The hidden layer is self-organizing. Following Kohonen's self-organizing input (SOM) theory [33], it has the capacity of adding and pruning nodes. Nodes are added if the Euclidean distance of the new inputs to the closest node center is beyond a threshold. When a node is added, its center is at the new input, the weight at zero, and the bandwidth in function of the distance to the closest node and predetermined resolution coefficient. Fig. 3a illustrates a *mexican hat* wavelet. Fig. 3b shows a single-layer feed-forward wavelet neural network.

The controller has been proposed for control of civil structures, and is therefore specialized for semi-active control. Three regions of force reachability are defined: the entire actuation force set C , the reachable force set C_d (in function of the semi-active device dynamics), and the transition layer C_t that acts as a smooth transition outside C_d . Thus, $C \supset C_t \supset C_d$. The wavelet nodes are allowed to be adapted when the control force $u \in C_d$, the adaption is smoothed using a function m_c for $u \in C_t$, and the adaptation is stopped for $u \in C$. Moreover, a sliding mode controller is included to account for the error between the required and the applied force. In this case, the objective state is used for the back-propagating error (regulatory control problem). The nodal parameters (centers, bandwidths, weights) are sequentially adapted at each time step following a back-propagation algorithm:

$$\begin{aligned} \dot{\alpha} &= -(1 - m_c)(\Gamma_\alpha \phi) B^T P^T s \\ \dot{\phi} &= -(1 - m_c)(\Gamma_\phi \alpha) B^T P^T s \end{aligned} \quad (21)$$

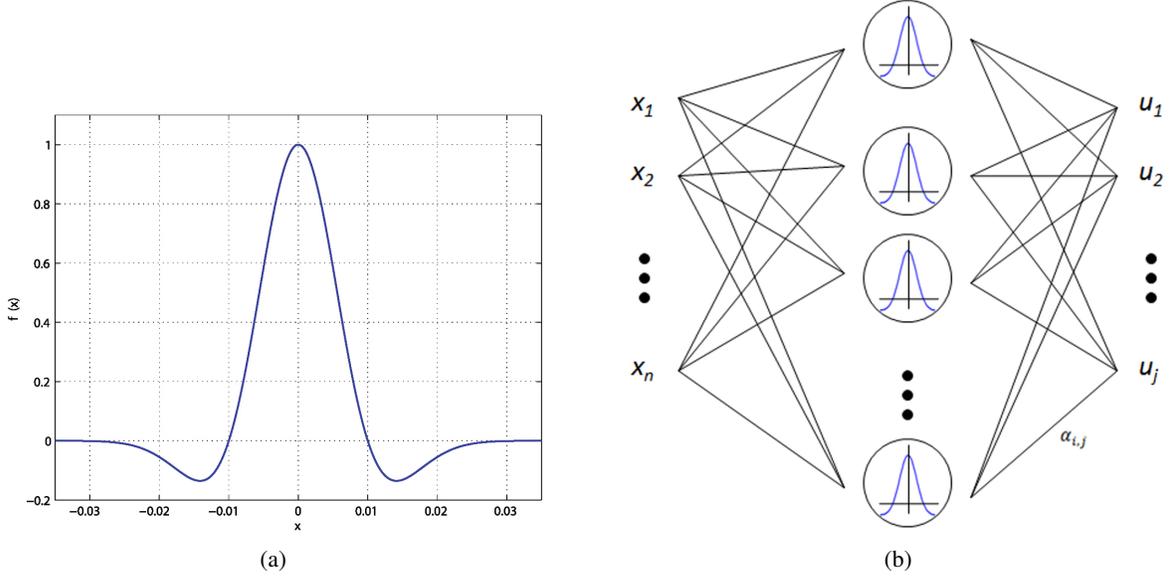


Figure 3: Illustration of: a) a *mexican hat* wavelet with $\mu = 0$ and $\sigma = 0.01$; and b) a wavelet neural network.

where $m_c \in [0, 1]$, $\mathbf{\Gamma}$ is the adaptation weights matrix, \mathbf{B} is the control weight matrix conform to the state-space formulation to be defined later for completeness, \mathbf{P} is the error weights on the sliding surface s , with the target sliding surface being $s = 0$. In other terms, $\mathbf{P}^T s$ is the back-propagation error. In the case of limited measurements, the error weights will take values of 0 where the states are unavailable. The dimension of the nodal parameters $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are adapted when the dimension of the input $\boldsymbol{\nu}$ is updated based on the SOI algorithm. The addition and suppression of nodal dimensions is done smoothly.

3.1 Example

A first simple example is showed here to demonstrate the performance of the proposed SOI adaptive controller. In this example, the neurocontroller is used for an active tracking problem of the following nonlinear function:

$$f(x, \dot{x}, u) = \left[1 - \left(\frac{(x - 0.1)^2}{0.1^2} + \frac{(\dot{x} - 0.05)^2}{0.05^2} \right) \right] e^{\left(-\frac{(x-0.1)^2}{0.1^2} + \frac{(\dot{x}-0.05)^2}{0.05^2} \right)} + u \quad (22)$$

with $x = 0.2 \sin 0.05t$, where the tracking objective is:

$$f^* = 0.02 \sin 0.01t \quad (23)$$

The control force u is provided by an ideal actuator, and the function is sampled at 100 Hz. The ideal actuator is herein defined as a device that can instantaneously apply a required force. Fig. 4a shows a quick convergence of the response to the tracking objective. It is compared with the performance of the same network with fixed values for τ and d that has been selected based on the study of the complete time series. It demonstrates that the SOI algorithm results in a significantly better tracking. Fig. 4b demonstrates that the control error stays within the prescribed control bounds. Fig. 4c is a plot of the embedding dimension d and delay τ against time, showing the sequential adaptation of the input space topology.

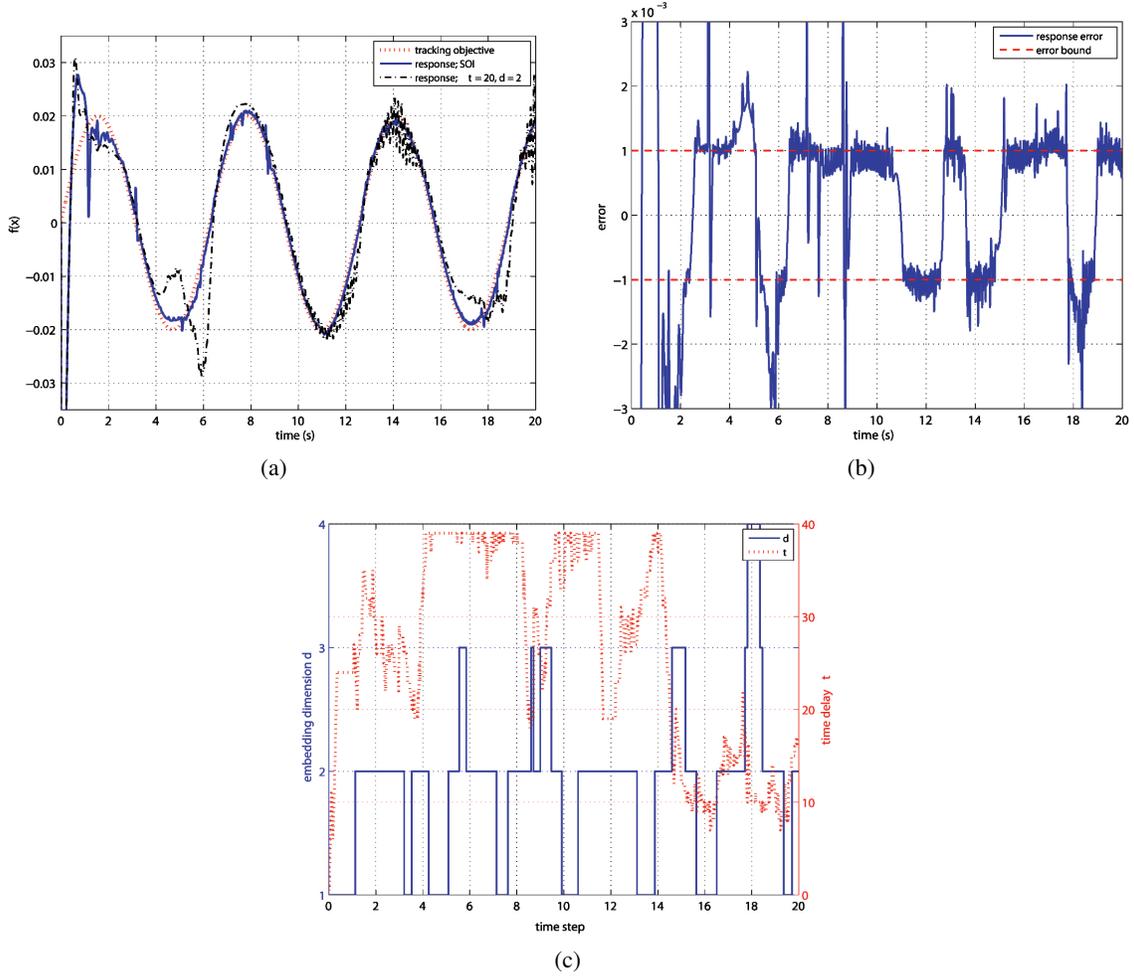


Figure 4: Performance of the SOI adaptive controller. a) time series performance against fixed inputs ($\tau = 20$; $d = 2$); b) error metric; and c) evolution of τ and d .

4 SIMULATIONS

A 39-story office tower located in downtown Boston, Massachusetts, as described in [34] and shown in Fig. 5, is loaded in the X-direction in order to assess the performance of the proposed SOI adaptive controller. The building was equipped in the 1990's with fluid dampers in order to mitigate excessive acceleration levels produced by the proximity of an existing 52-story tower. Dampers are installed every other story from the 5th floor up to the 34th floor. The viscous dampers in the X-direction have a capacity of 1350 kN (300 kips) with a damping coefficient of 52550 kN·s/m (300 kips·s/in) below the 26th floor, and a capacity of 900 kN (200 kips) with a damping coefficient of 35000 kN·s/m (200 kips·s/in) from the 26th floor and above. In the simulation, viscous dampers have been replaced by variable friction devices of similar capacities installed at the same locations. The friction devices consist of a variable Coulomb friction element, a viscous element, and a stiffness element installed in parallel. The equation of motion of the system can be written in the form of:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} + \mathbf{B}_u\mathbf{u} + \mathbf{B}_w\mathbf{w} \quad (24)$$

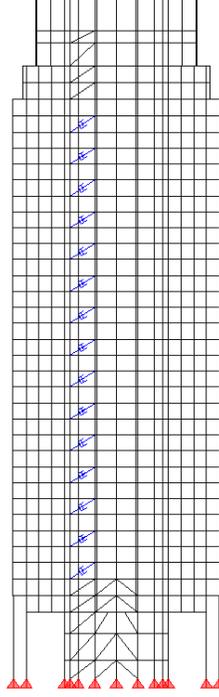


Figure 5: Simulated structure in the X-direction.

where \mathbf{A} is the state-space matrix, \mathbf{B} is the control force incidence vector with subscripts \mathbf{u} and \mathbf{w} referring to actuation and wind inputs respectively, and \mathbf{w} is the wind excitation input, with:

$$\mathbf{A} = \begin{bmatrix} \mathbf{O} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C} \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} \mathbf{x} \\ \dot{\mathbf{x}} \end{bmatrix}$$

$$\mathbf{B}_u = \begin{bmatrix} \mathbf{O} \\ -\mathbf{M}^{-1}\mathbf{F} \end{bmatrix} \quad \mathbf{B}_w = \begin{bmatrix} \mathbf{O} \\ -\mathbf{M}^{-1}\mathbf{E} \end{bmatrix}$$

where $\mathbf{M} \in \mathbb{R}^{n \times n}$, $\mathbf{C} \in \mathbb{R}^{n \times n}$, and $\mathbf{K} \in \mathbb{R}^{n \times n}$ are the mass, damping, and stiffness matrices respectively, $\mathbf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix, $\mathbf{F} \in \mathbb{R}^{n \times a}$ is the damper location matrix, $\mathbf{E} \in \mathbb{R}^{n \times 1}$ is a vector of ones, n is the number of states, a is the number of actuators, and \mathbf{O} are compatible zero matrices such that $\mathbf{A} \in \mathbb{R}^{2n \times 2n}$, $\mathbf{B}_u \in \mathbb{R}^{2n \times a}$, $\mathbf{B}_w \in \mathbb{R}^{2n \times 1}$, $\mathbf{X} \in \mathbb{R}^{2n \times 1}$, $\mathbf{u} \in \mathbb{R}^{a \times 1}$, and $\mathbf{w} \in \mathbb{R}^{n \times 1}$.

Two simulations are conducted. The first one consists of an harmonic excitation to show convergence of the SOI algorithm with respect to a classical approach of fixed inputs. An analysis is also conducted on the sensitivity with respect to the predefined network resolution. The second simulation looks at the performance of the controller for wind mitigation to compare with the fixed inputs case, along with results from [34] where viscous dampers are utilized.

For all simulations, decentralized control is used for each semi-active device installed between the $(i + 1)^{\text{th}}$ and the i^{th} floor. The assumed available measurements are inter-story displacements $x_{i+1} - x_i$, inter-story velocity $\dot{x}_{i+1} - \dot{x}_i$, acceleration \ddot{x}_i and force input u_i at the device locations. The sliding surface \mathbf{P} is constructed accordingly. The force inputs for all simulations are scaled by a factor of 10^{-2} with respect to acceleration inputs.

4.1 Harmonic Excitation

A first set simulation is conducted using a harmonic excitation at the fundamental frequency acting on every floor: $W(t) = 5 \sin 1.19t$. Results from the SOI algorithm are compared with classical fixed parameter strategies. Several fixed parameters have been attempted, and results show the optimal result using $\tau = 20$, and $d = 3$. The controller non-adaptive parameters are kept constant for both the SOI and fixed parameter cases. Fig. 6a shows the convergence of the acceleration of the 37th floor for both cases. The convergence speed and final accelerations are similar. Fig. 6b shows the force-velocity plot of the 10th control device for the last 10 seconds of simulation. Both cases converged to a similar actuation dynamics. Fig. 6c shows the size of the hidden layer for the 10th control device, for both SOI and fixed inputs cases. The network size corresponds to the number of nodes. The SOI clearly achieved a representation using about a third of the nodes.

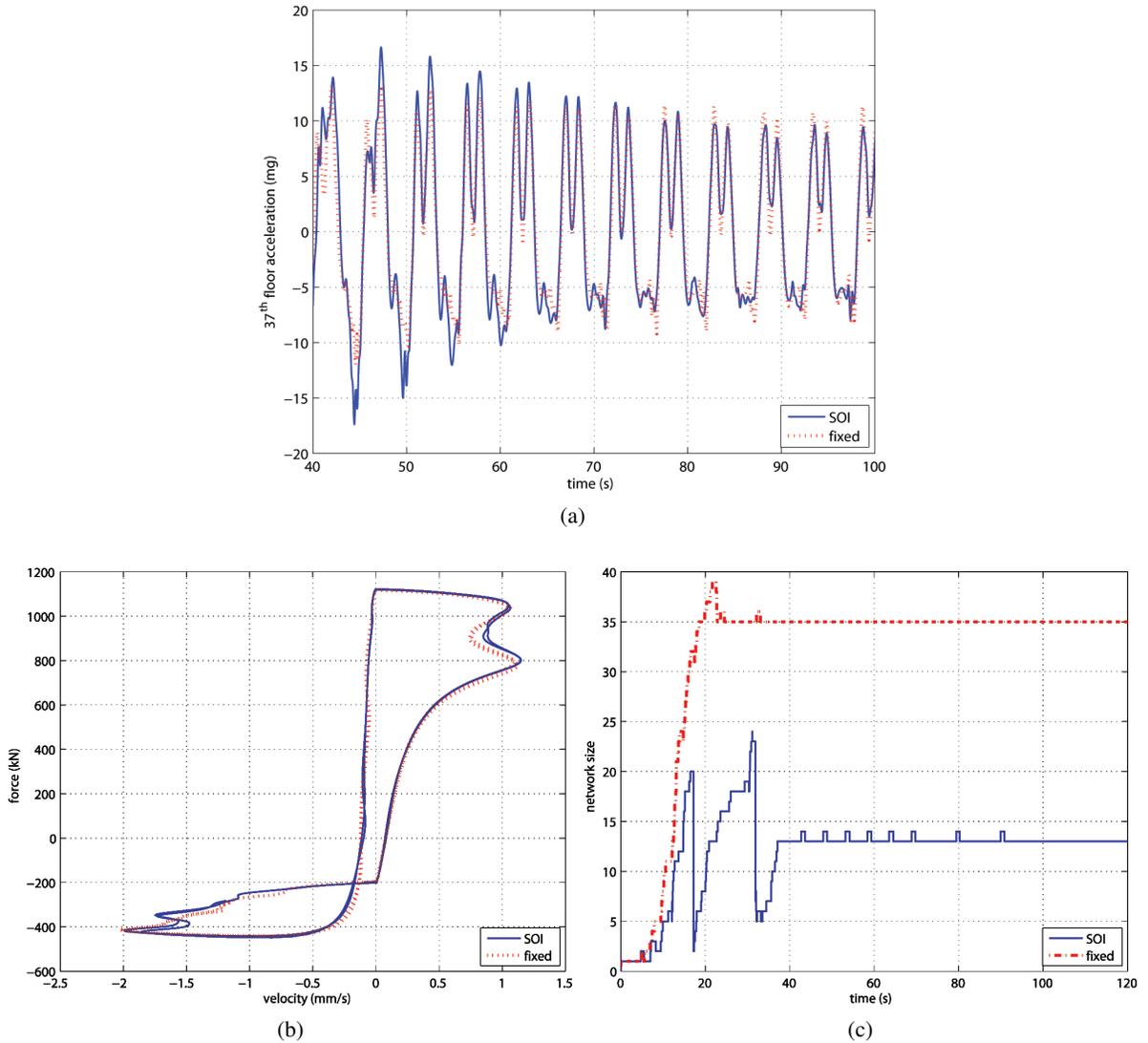


Figure 6: Performance of the SOI adaptive controller. a) time series performance against fixed inputs; b) force-velocity plot; and c) evolution of network sizes.

4.2 Wind Excitation

This simulation is conducted using a wind load such that the building responses under the uncontrolled case and the passive viscous cases compare with the responses reported in [34]. The wind load is simulated over 200 seconds. It is repeated 5 consecutive times to verify the performance of the algorithm over training periods. Results from the SOI algorithm are compared against the fixed inputs case. The fixed input vector case is constructed using a time delay $\tau = 10$ and embedding dimension $d = 3$. The control objective is acceleration mitigation of the 37th floor, which coincides with the largest acceleration among the occupied floors. Results are also compared against mitigation from the full voltage and the linear-quadratic regulator (LQR) cases. The full voltage case refers to the semi-active device utilized with full damping capacity, while the LQR case is a LQR controller designed with full knowledge of the structure.

Fig. 7a shows the performance of the SOI algorithm against various control strategies. Results show that the controller outperforms the fixed input case, and that the general performance of the neurocontroller improves with training periods. It is also better than the full voltage strategy, but does not outperform an LQR controller with full parametric knowledge, as one would expect. Fig. 7b shows the evolution of the network sizes for the 10th semi-active device. It is shown that the SOI algorithm can outperform the fixed inputs case with significantly less network nodes, and that the network size keeps evolving with respect to time. The result is representative of other control devices. Table 1 summarizes performance of the controller after one training period against various control strategies.

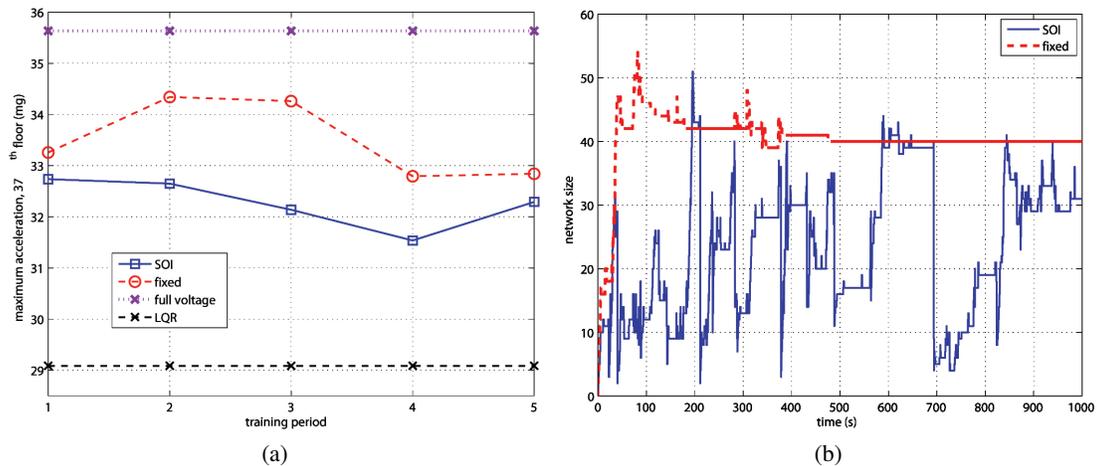


Figure 7: Performance of the SOI adaptive controller against the fixed inputs case. a) acceleration mitigation; and b) network size.

Table 1: Maximum acceleration at the 37th floor after one training period, wind excitation.

control strategy	maximum acceleration \ddot{x}_{37} (mg)	reduction (%)
uncontrolled	70.1	
viscous dampers	45.2	35.5
full voltage	35.6	49.2
no voltage	67.5	3.71
SOI	32.7	53.4
fixed inputs	33.3	52.5
LQR	29.1	58.5

4.2.1 Stationarity of Local Maps

The level of stationarity of local maps is evaluated following the technique described in Section 2.2. Fig. 8a shows the wind excitation on the 5th floor over the period 100-200 sec. The level of stationarity is evaluated after the initial adaptation (100 sec) assumed to be already achieved. The excitation is representative of the excitations on the other floors. Fig. 8b shows the level of stationary maps in function of the percentage variation of \tilde{f} . The figure illustrates that 46.5% of maps showed 0% variation in \tilde{f} . This level quickly raise to 64.3% for 1% variation, and then to 69.9% at 5% variation. The degree at which the assumption of local stationary or quasi-stationary maps holds depends on the acceptable level of quasi-stationarity. In this case, a level of 5% is taken as acceptable, and a level close to 70% stationary maps is interpreted as sufficient for the assumption to hold.

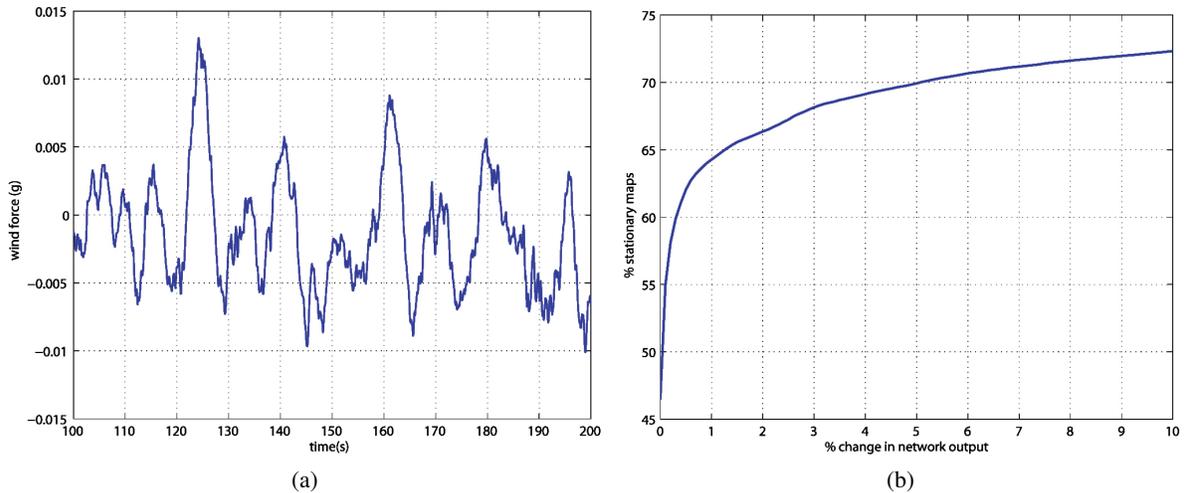


Figure 8: a) wind excitation; and b) stationarity index

5 CONCLUSION

A method for sequentially organizing the input space of a black-box representation, the SOI algorithm, has been proposed. The algorithm analyzes a sliding window of the last observations, and determines which inputs would optimally represent the system dynamics. It is based on the embedding theorem. The SOI algorithm has the powerful advantage to take the input selection out of the black-box

design process, and can be applied sequentially and in real time to nonautonomous nonstationary MDOF systems.

The algorithm has been applied in the context of an adaptive neurocontroller. Simulations on an existing 39-story building show that the SOI strategy can perform as well as a fixed input strategy in the case of an harmonic excitation, with only a third of the network size. Moreover, it is capable of outperforming a fixed strategy under a wind excitation, with a significantly leaner network. It was also demonstrated that the algorithm can be used with nonstationary systems.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge Professor Jean-Jacques E. Slotine and Dr. Anuradha M. Annaswamy for their guidance in the research.

REFERENCES

- [1] J. Ghaboussi and A. Joghataie, "Active control of structures using neural networks," *Journal of Engineering Mechanics*, vol. 121, no. 4, pp. 555–567, 1999.
- [2] R. Jiménez and L. Alvarez-Icaza, "A real-time estimation scheme for buildings with intelligent dissipation devices," *Mechanical Systems and Signal Processing*, vol. 21, no. 6, pp. 2427–2440, 2007.
- [3] S. Suresh, S. Narasimhan, and N. Sundararajan, "Adaptive control of nonlinear smart base-isolated buildings using Gaussian kernel functions," *Structural Control and Health Monitoring*, vol. 15, no. 4, pp. 585–603, 2008.
- [4] J. Slotine, W. Li, *et al.*, *Applied nonlinear control*. Prentice-Hall Englewood Cliffs, NJ, 1991.
- [5] K. Astrom and B. Wittenmark, *Adaptive control*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1994.
- [6] K. Narendra and A. Annaswamy, *Stable adaptive systems*. Dover Publications, NY, 2005.
- [7] A. da Silva, P. Alexandre, V. Ferreira, and R. Velasquez, "Input space to neural network based load forecasters," *International Journal of Forecasting*, vol. 24, no. 4, pp. 616–629, 2008.
- [8] M. Nørgaard, O. Ravn, N. Poulsen, and L. Hansen, *Neural networks for modelling and control of dynamic systems: A practitioner's handbook*. Springer Verlag, 2000.
- [9] F. Takens, "Detecting strange attractors in turbulence," *Dynamical systems and turbulence, Warwick 1980*, pp. 366–381, 1980.
- [10] J. Principe, L. Wang, and M. Motter, "Local dynamic modeling with self-organizing maps and applicationsto nonlinear system identification and control," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2240–2258, 1998.
- [11] V. Plagianakos and E. Tzanaki, "Chaotic analysis of seismic time series and short term forecasting using neural networks.," 2001.
- [12] D. Walker, N. Tuffiaro, and P. Gross, "Radial-basis models for feedback systems with fading memory," *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, vol. 48, no. 9, pp. 1147–1151, 2001.

- [13] H. Leung, T. Lo, and S. Wang, "Prediction of noisy chaotic time series using an optimal radial basis function neural network," *IEEE Transactions on Neural Networks*, vol. 12, no. 5, pp. 1163–1172, 2001.
- [14] M. Széliga, P. Verdes, P. Granitto, and H. Ceccatto, "Modelling nonstationary dynamics," *Physica A: Statistical Mechanics and its Applications*, vol. 327, no. 1-2, pp. 190–194, 2003.
- [15] L. Shen, M. Wang, W. Liu, and G. Sun, "Prediction based chaos control via a new neural network," *Physics Letters A*, vol. 372, no. 46, pp. 6916–6921, 2008.
- [16] J. Zolock and R. Greif, "A Methodology for the Modeling of Forced Dynamical Systems From Time Series Measurements Using Time-Delay Neural Networks," *Journal of Vibration and Acoustics*, vol. 131, p. 011003, 2009.
- [17] A. Ruano, *Intelligent control systems using computational intelligence techniques*. IET, 2005.
- [18] L. Aguirre and C. Letellier, "Modeling nonlinear dynamics and chaos: a review," *Mathematical Problems in Engineering*, 2009.
- [19] S. Laflamme, J.-J. Slotine, and J. Connor, "Wavelet network for semi-active control," *Journal of Engineering Mechanics*, (under review).
- [20] J. Stark, "Delay embeddings for forced systems. I. Deterministic forcing," *Journal of Nonlinear Science*, vol. 9, no. 3, pp. 255–332, 1999.
- [21] V. Caballero, "On an embedding theorem," *Acta Mathematica Hungarica*, vol. 88, no. 4, pp. 269–278, 2000.
- [22] J. Stark, D. Broomhead, M. Davies, and J. Huke, "Delay embeddings for forced systems. II. Stochastic forcing," *Journal of Nonlinear Science*, vol. 13, no. 6, pp. 519–577, 2003.
- [23] E. Monroig, K. Aihara, and Y. Fujino, "Modeling dynamics from only output data," *Physical Review E*, vol. 79, no. 5, p. 56208, 2009.
- [24] R. Hegger, H. Kantz, L. Matassini, and T. Schreiber, "Coping with nonstationarity by overembedding," *Physical review letters*, vol. 84, no. 18, pp. 4092–4095, 2000.
- [25] P. Verdes, P. Granitto, and H. Ceccatto, "Overembedding method for modeling nonstationary systems," *Physical review letters*, vol. 96, no. 11, p. 118701, 2006.
- [26] Y. Hirata, H. Suzuki, and K. Aihara, "Reconstructing state spaces from multivariate data using variable delays," *Physical Review E*, vol. 74, no. 2, p. 26202, 2006.
- [27] Q. Meng and Y. Peng, "A new local linear prediction model for chaotic time series," *Physics Letters A*, vol. 370, no. 5-6, pp. 465–470, 2007.
- [28] R. Manuca and R. Savit, "Stationarity and nonstationarity in time series analysis," *Physica D: Nonlinear Phenomena*, vol. 99, no. 2-3, pp. 134–161, 1996.
- [29] J. Chen, Y. Zeng, and H. Sun, "Dynamic reliability analysis of antenna reflector accuracy under wind excitation," *Computers & Structures*, vol. 59, no. 5, pp. 819–822, 1996.
- [30] S. Billings and H. Wei, "A new class of wavelet networks for nonlinear system identification," *IEEE Transactions on Neural Networks*, vol. 16, no. 4, pp. 862–874, 2005.

- [31] A. Fraser and H. Swinney, "Independent coordinates for strange attractors from mutual information," *Physical Review A*, vol. 33, no. 2, pp. 1134–1140, 1986.
- [32] M. Kennel, R. Brown, and H. Abarbanel, "Determining embedding dimension for phase-space reconstruction using a geometrical construction," *Physical Review A*, vol. 45, no. 6, pp. 3403–3411, 1992.
- [33] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.
- [34] R. McNamara and D. Taylor, "Fluid viscous dampers for high-rise buildings," *The Structural Design of Tall and Special Buildings*, vol. 12, no. 2, pp. 145–154, 2003.