Multiscale Asymmetric Orthogonal Wavelet Kernel for Linear Programming Support Vector Learning and Nonlinear Dynamic Systems Identification

Zhao Lu, *Member, IEEE,* Jing Sun, *Fellow, IEEE,* and Kenneth Butts, *Member, IEEE*

***Abstract*—Support vector regression for approximating nonlinear dynamic systems is more delicate than the approximation of indicator functions in support vector classification, particularly for systems that involve multitudes of time scales in their sampled data. The kernel used for support vector learning determines the class of functions from which a support vector machine can draw its solution, and the choice of kernel significantly influences the performance of a support vector machine. In this paper, to bridge the gap between wavelet multiresolution analysis and kernel learning, the closed-form orthogonal wavelet is exploited to construct new multiscale asymmetric orthogonal wavelet kernels for linear programming support vector learning. The closed-form multiscale orthogonal wavelet kernel provides a systematic framework to implement multiscale kernel learning via dyadic dilations and also enables us to represent complex nonlinear dynamics effectively. To demonstrate the superiority of the proposed multiscale wavelet kernel in identifying complex nonlinear dynamic systems, two case studies are presented that aim at building parallel models on benchmark datasets. The development of parallel models that address the long-term/mid-term prediction issue is more intricate and challenging than the identification of series-parallel models where only one-step ahead prediction is required. Simulation results illustrate the effectiveness of the proposed multiscale kernel learning.**

***Index Terms*—Linear programming support vector regression, model sparsity, multiscale orthogonal wavelet kernel, NARX model, parallel model, type-II raised cosine wavelet.**

I. Introduction

SINCE THE inception of support vector learning, the burgeoning of kernel learning has expanded the theory of computational learning to a new horizon [1]–[3]. As a universal approach for solving the problems of multidimensional function estimation, the support vector machine (SVM) was initially developed to solve pattern recognition problems. More recently, the notion of support vector learning has been successfully generalized to various fields such as nonlinear regression, linear operator equations, and signal processing by introducing the ε-insensitive loss function [4]–[8]. When SVM is employed to tackle the problems of function approximation and estimation, the approaches are often referred to as the support vector regression (SVR).

As a typical nonparametric kernel learning approach, the advent of SVR also provides a promising avenue to nonlinear dynamical system modeling. With the aid of duality, the conventional SVR formulates the modeling task as a quadratic programming problem, through which a kernel expansion representation for the underlying system can be calculated with generalization capability [9]–[11]. However, since all data points not inside the ε-tube are selected as support vectors, substantial redundant terms may be included in the nonparametric kernel expansion representation derived from conventional quadratic programming support vector regression (QP-SVR) when used for identifying complex nonlinear dynamical systems [12]–[14]. This may lead to the loss of model succinctness and thereby the degradation of computational efficiency in evaluating the model, which has been a main stumbling block in applying SVR for nonlinear systems identification, where large quantities of sampled data are usually involved in training. Meanwhile, the required calculation for solving the quadratic programming problem can be computationally burdensome in practice.

To surmount these problems, the algorithm of linear programming support vector regression (LP-SVR) was proposed in [15] and [16]. Due to the different support vector selection mechanism, LP-SVR is exceptional in building parsimonious models, which makes it advantageous for nonlinear systems identification. Rather than capitalizing on duality to reformulate the optimization problem in terms of kernel function as is done in QP-SVR, LP-SVR uses the kernel expansion as a model representation in the hypothesis space, and the ℓ_1 norm of the kernel expansion coefficients vector serves as a regularizer for controlling the model complexity and structural risk, which inherently enforces sparseness of the solution. This leads to the superiority of LP-SVR in model sparsity, computational efficiency, and adaptability to more general non-Mercer kernel functions [14], [17].

On the other hand, the kernel expansion model in SVR for capturing the underlying system dynamics is more delicate than that used for approximating the indicator functions in support vector classification, and various systems modeling problems need various sets of kernel functions due to the

Manuscript received November 13, 2012; revised May 18, 2013; accepted August 16, 2013. Date of publication September 18, 2013; date of current version April 11, 2014. This paper was recommended by Associate Editor S. Hu.

Z. Lu is with the Department of Electrical Engineering, Tuskegee University, Tuskegee, AL 36088 USA (e-mail: zlu@ieee.org).

J. Sun is with the Department of Naval Architecture and Marine Engineering and the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: jingsun@umich.edu).

K. Butts is with Toyota Motor Engineering and Manufacturing North America, Ann Arbor, MI 48105 USA (e-mail: ken.butts@tema.toyota.com).

Digital Object Identifier 10.1109/TCYB.2013.2279834

complexity of the dependencies [6]. The kernel functions determine the class of functions from which an SVM can draw its solution, and the choice of kernel significantly affects the performance of an SVM [18]. Therefore, it is of practical and theoretical importance to construct specific kernels that reflect the specific properties of the underlying system dynamics. In the realm of nonlinear dynamic system identification, the nonlinear autoregression with exogenous input (NARX) model is used widely for representing discrete-time nonlinear systems, and the regressor for the NARX model consists of two parts, an autoregressive (AR) part and a moving-average (MA) part. The mathematical description of the NARX model is as follows:

$$\hat{y}_n = f(y_{n-1},\ y_{n-2},\ \cdots,\ y_{n-P},\ u_n,\ u_{n-1},\ \cdots,\ u_{n-Q+1}) \tag{1}$$

where u_n and y_n are the input and output to the system at time instant t_n, and the vectors $\boldsymbol{y}_{n-1} = [y_{n-1},\ y_{n-2},\ \cdots,\ y_{n-P}]^{\mathrm{T}}$ and $\boldsymbol{u}_n = [u_n,\ u_{n-1},\ \cdots,\ u_{n-Q+1}]^{\mathrm{T}}$ are the AR and MA parts, respectively. The AR part is a window of past system outputs with output order P, and the MA part is a window of past and current system inputs with input order Q. In the endeavor of applying kernel learning strategies for identifying NARX models, the idea of the composite kernel was conceptualized and developed for taking account of the different cause-effect relationships of the AR and MA parts to the NARX model output instead of assimilating them [19]–[21]. The model represented by a composite kernel expansion is in the form of

$$\hat{y}_n = \sum_{i=1}^{N} \beta_i (k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)) \tag{2}$$

where β_i is the expansion coefficient and N is the number of sampled data. The k_1 and k_2 are the kernel functions for the AR and MA parts, respectively, and $k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)$ is defined as the composite kernel. The composite kernel expansion model (2) enables us to use different kernel functions for the AR and MA parts of the regressor in (1).

Furthermore, how to design the apposite kernel functions k_1 and k_2 in (2) for identifying the underlying nonlinear dynamics is obviously crucial. Due to the ubiquity of transient characteristics and multiscale structures in nonlinear dynamics, refinable kernel functions capable of characterizing the dynamics of the underlying time series and taking account of local as well as global complexity in signals are highly desirable. Compared to other basis functions, the most significant property of wavelets lies in their ability in capturing localized temporal and frequency information of rapidly changing transient signals [22].

In this paper, we investigate one class of closed-form orthogonal wavelets, the type-II raised cosine wavelet, from which a new multiscale orthogonal wavelet kernel for nonlinear dynamic system modeling is educed. Although the wavelet kernels for support vector learning have been discussed in [23]–[25], most of them are based on a Morlet wavelet given by a sine (cosine) wave modulated by a Gaussian envelope [26], and they have an explicit closed form but do not lead to the interscale orthogonality required in the framework of multiresolution analysis. Hence, the use of nonorthogonal wavelets as kernel functions is unable to elicit a systematic implementation of multiscale learning in the manner of multiresolution analysis. On the other hand, the attraction of using an orthogonal wavelet kernel function mainly lies in the ease of crystallizing the capability of orthogonal wavelet for multiresolution analysis into the systematic implementation of multiscale kernel learning via dyadic dilations. However, the difficulties encountered in constructing an orthogonal wavelet kernel mainly lie in the fact that almost all the known orthonormal wavelets are not expressible in closed form, and can only be expressed in terms of integrals or recurrence formulas [27], [28]. Although it is known that Haar and Shannon orthonormal wavelets can be expressed in explicit formulae, the discontinuity in the Haar wavelet and the poor time localization of the Shannon wavelet limit their utility in the multiscale modeling context [28].

Recent research has confirmed the existence of nontrivial orthonormal wavelets (i.e., not the Haar and Shannon), such as the raised cosine wavelet and Young's function based wavelets [27]–[29], that have relatively simple analytic forms, and therefore new avenues for kernel-based multiscale learning and analysis are available. In our research, the use of the type-II raised cosine wavelet for constructing kernel functions enables a systematic implementation of the multiscale learning strategy via dyadic dilations under the framework of multiresolution analysis. The asymmetry of the proposed innovative multiscale kernel function is one salient feature that sets it apart from most of widely used kernel functions. Although the asymmetrical kernels have been found useful in nonlinear regression and object tracking [30]–[32], they were rarely studied and applied in the context of support vector learning [33]. It is well-known that only the Mercer kernel can be used for conventional quadratic programming support vector learning to ensure the positive definiteness of the Hessian matrix [1]–[6] for optimization. However, as isotropic similarity measures, the symmetric kernels have limited capability in representing an anisotropic object in a compact and sparse model. The asymmetric kernel, with the symmetric kernel as a special case, offers more flexibility in representing irregular complex dependencies. In this paper, on the strength of the adaptability of LP-SVR to non-Mercer kernels, a new trail to multiscale kernel learning is blazed for estimating complex dependencies by constructing the asymmetric closed-form orthogonal wavelet kernel. While new nonstandard kernels are attracting more and more interest in constructive approximation theory [34], [35], this is the first study of asymmetrical multiscale orthogonal wavelet kernels for support vector learning to our best knowledge.

Our simulation study focuses on the development of parallel models for a hydraulic robot arm and the Box and Jenkin's gas furnace system using published benchmark datasets, to validate the effectiveness and practicality of the developed multiscale orthogonal wavelet kernel for nonlinear dynamic system identification. The NARX model (1) is also called the series-parallel model because the system and model are parallel with respect to u_n but in series with respect to y_n. Contrary to the series-parallel model (1), where the past values

of the system input and the system output constitute the regressor, the regressor of the parallel model is composed of the past values of the system input and the model output, that is

$$\hat{y}_n = f(\hat{y}_{n-1}, \hat{y}_{n-2}, \cdots, \hat{y}_{n-P}, u_n, u_{n-1}, \cdots, u_{n-Q+1}). \tag{3}$$

Hence, the parallel models are recurrent per se due to the feedback involved. It is opined in the community of systems identification that building parallel models of nonlinear dynamic systems is one of the most formidable technical challenges [36], [37]. In the literature, most studies of nonlinear system identification concentrate on the series-parallel model, because of its ease in optimizing the model parameters. Much less has been reported on the study of effective approaches for identifying parallel models of the nonlinear dynamical systems. An early attempt to exert support vector learning for parallel model identification was reported in [38], where the least square support vector machine (LS-SVM) was employed for modeling autonomous nonlinear systems. Yet the issue of complete loss of model sparsity associated with LS-SVM has made it unsuitable for nonlinear system identification in practice [39].

This paper is organized as follows. In the next section, a brief review of linear programming support vector learning with composite kernel is given for completeness. The construction of a new multiscale asymmetric orthogonal wavelet kernel is developed in Section III. Section IV presents the simulation study for identifying the nonlinear parallel model on two benchmark datasets. Finally, Section V concludes this paper. The following generic notations will be used throughout this paper: lower case symbols such as $x, y, \alpha, \cdots$ refer to scalar valued objects, lower case boldface symbols such as $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\beta}, \cdots$ refer to vector valued objects, and finally capital boldface symbols such as $\mathbf{K}_1, \mathbf{K}_2, \cdots$, are used for matrices.

II. Linear Programming SVR With Composite Kernel

Conventionally, the problem of system identification consists of setting up a suitably parameterized identification model and adjusting the parameters of the model to optimize a performance function based on the error between the system and the identification model outputs. Contrary to that, a model identified through support vector regression is represented as the kernel expansion on the support vectors, which are the data points in a selected subset of the training data [4]–[6]. In other words, the model is represented in a data-dependent nonparametric form. The vector pairs $[(\boldsymbol{y}_{i-1})^T, (\boldsymbol{u}_i)^T]^T$ corresponding to the nonzero coefficients β_i in model representation (2) are the support vectors (SV).

Consequently, the model sparsity, which is defined as the ratio of the number of support vectors to the number of all training data points, plays a key role in controlling model complexity and alleviating model redundancy. A kernel expansion model with substantial redundant terms is against the parsimonious principle which ensures the simplest possible model that explains the data, and may deteriorate the generalization performance and increase the computational requirements substantially.

The number of nonzero components in the coefficients vector $\boldsymbol{\beta} = [\beta_1, \beta_2, \cdots, \beta_N]^{\mathrm{T}}$ largely determines the complexity of the kernel expansion model (2). In order to enforce the sparseness of the model, the linear programming support vector learning employs the ℓ_1 norm of the coefficients vector $\boldsymbol{\beta}$ in model (2) as a regularizer in the objective function to control the model complexity and structural risk. By introducing the ε-insensitive loss function, which is defined as

$$L(y_n - \hat{y}_n) = \begin{cases} 0, & \text{if } |y_n - \hat{y}_n| \le \varepsilon \\ |y_n - \hat{y}_n| - \varepsilon, & \text{otherwise} \end{cases} \tag{4}$$

the regularization problem to be solved for procuring the model becomes

$$\text{minimize} \quad R_{\text{reg}}[f] = \|\boldsymbol{\beta}\|_1 + C \sum_{n=1}^{N} L(y_n - \hat{y}_n) \tag{5}$$

where the parameter C controls the extent to which the regularization term influences the solution and ε is the error tolerance. Geometrically, the ε-insensitive loss function defines a ε-tube. The idea of using the ℓ_1 norm to secure a sparse representation in LP-SVR is also explored in the emerging theory of compressive sensing [40]–[42].

By introducing the slack variables $\xi_n, n = 1, 2, \ldots, N$, to accommodate otherwise infeasible constraints and to enhance robustness, the regularization problem (5) can be transformed into the following equivalent constrained optimization problem:

$$\begin{aligned} &\text{minimize } \|\boldsymbol{\beta}\|_1 + C \sum_{n=1}^{N} \xi_n \\ &\text{subject to} \begin{cases} \sum_{i=1}^{N} \beta_i (k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)) - y_n \le \varepsilon + \xi_n \\ y_n - \sum_{i=1}^{N} \beta_i (k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)) \le \varepsilon + \xi_n \\ \xi_n \ge 0 \qquad n = 1, 2, \ldots, N \end{cases} \end{aligned} \tag{6}$$

where the constant $C > 0$ determines the tradeoff between the sparsity of the model and the amount up to which deviations larger than ε can be tolerated. For the purpose of converting (6) into a linear programming problem, the components β_i of the coefficients vector $\boldsymbol{\beta}$ and their absolute values $|\beta_i|$ are decomposed as follows:

$$\beta_i = \alpha_i^+ - \alpha_i^- \qquad |\beta_i| = \alpha_i^+ + \alpha_i^- \tag{7}$$

where $\alpha_i^+, \alpha_i^- \ge 0$, and for a given β_i there is only one pair (α_i^+, α_i^-) fulfilling both equations in (7) because $\{[1\ 1]^{\mathrm{T}}, [-1\ 1]^{\mathrm{T}}\}$ is a nonstandard basis for R^2. It is underscored that both variables cannot be positive at the same

time, i.e., $\alpha_i^+ \cdot \alpha_i^- = 0$. In this way, the optimization problem (6) can be reformulated as

$$\begin{aligned} &\text{minimize } \sum_{i=1}^{N}(\alpha_i^+ + \alpha_i^-) + C\sum_{n=1}^{N}\xi_n \\ &\text{subject to } \begin{cases} \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-)(k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)) \\ \quad -\xi_n \le \varepsilon + y_n \\ -\sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-)(k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)) \\ \quad -\xi_n \le \varepsilon - y_n \\ \xi_n \ge 0, \qquad n = 1,\ 2,\ \cdots,\ N. \end{cases} \end{aligned} \tag{8}$$

Next, define the vector

$$\boldsymbol{c} = (\underbrace{1,\ 1,\ \cdots,\ 1}_{N},\ \underbrace{1,\ 1,\ \cdots,\ 1}_{N},\ \underbrace{C,\ C,\ \cdots,\ C}_{N})^T \tag{9}$$

and write the ℓ_1 norm of $\boldsymbol{\beta}$ as

$$\|\boldsymbol{\beta}\|_1 = (\underbrace{1,\ 1,\ \cdots,\ 1}_{N},\ \underbrace{1,\ 1,\ \cdots,\ 1}_{N}) \begin{pmatrix} \boldsymbol{\alpha}^+ \\ \boldsymbol{\alpha}^- \end{pmatrix} \tag{10}$$

with the N-dimensional column vectors $\boldsymbol{\alpha}^+$ and $\boldsymbol{\alpha}^-$ defined as $\boldsymbol{\alpha}^+ = \left(\alpha_1^+,\ \alpha_2^+,\ \cdots,\ \alpha_N^+\right)^T$ and $\boldsymbol{\alpha}^- = \left(\alpha_1^-,\ \alpha_2^-,\ \cdots,\ \alpha_N^-\right)^T$, the constrained optimization problem (8) can be cast as a linear programming problem in the following form:

$$\begin{aligned} &\text{minimize } \boldsymbol{c}^T \begin{pmatrix} \boldsymbol{\alpha}^+ \\ \boldsymbol{\alpha}^- \\ \boldsymbol{\xi} \end{pmatrix} \\ &\text{subject to } \begin{cases} \begin{pmatrix} \mathbf{K_1}+\mathbf{K_2} & -(\mathbf{K_1}+\mathbf{K_2})-\mathbf{I} \\ -(\mathbf{K_1}+\mathbf{K_2}) & \mathbf{K_1}+\mathbf{K_2} \quad -\mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\alpha}^+ \\ \boldsymbol{\alpha}^- \\ \boldsymbol{\xi} \end{pmatrix} \le \begin{pmatrix} \boldsymbol{y}+\varepsilon \\ \varepsilon-\boldsymbol{y} \end{pmatrix} \\ \qquad \boldsymbol{\alpha}^+,\ \boldsymbol{\alpha}^- \ge 0, \quad \boldsymbol{\xi} \ge 0 \end{cases} \end{aligned} \tag{11}$$

where $\boldsymbol{\xi} = (\xi_1, \xi_2, \cdots, \xi_N)^T$, $\boldsymbol{y} = (y_1, y_2, \cdots, y_N)^T$ and $\mathbf{I}$ is an $N \times N$ identity matrix. $\mathbf{K}_1$ and $\mathbf{K}_2$ are the kernel matrices with entries defined as $(\mathbf{K}_1)_{in} = k_1(\boldsymbol{y}_{i-1},\ \boldsymbol{y}_{n-1})$, $(\mathbf{K}_2)_{in} = k_2(\boldsymbol{u}_i,\ \boldsymbol{u}_n)$. The calculation of the vectors $\boldsymbol{\alpha}^+$, $\boldsymbol{\alpha}^-$ and the support vectors selection can be accomplished by solving the optimization problem (11) using the well-known simplex or primal-dual interior point algorithms. With the solution to linear programming problem (11), the coefficients of the composite kernel expansion model (2) can be calculated by using (7), and thereby the model (2) can be built as follows:

$$\hat{y}_n = \sum_{i \in SV} \beta_i (k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)). \tag{12}$$

This composite kernel expansion on selected support vectors is for representing the nonlinear dynamics underlying the time series $\{u_i,\ y_i\}$, $i = 1,\ 2,\ \cdots,\ N$. One disadvantage of LP-SVR is the lack of the theoretical understanding of the support vector selection mechanism [12].

Most of the preceding work applying support vector learning to nonlinear systems identification [9]–[11], [17], [25] treat system identification as a general regression problem, where the AR and MA parts are consolidated in the regressor. However, the chosen single kernel function might be ineffective in characterizing different cause-effect relationships of the AR and MA parts to the model output. Modeling the different dependencies by heterogeneous kernel functions is the main motivation for using the composite kernel, which provides new degrees of freedom in representing nonlinear dynamics and also makes the model more amenable to control law design.

III. Construction of the Multiscale Orthogonal Wavelet Kernel

The methodology of wavelet has been successfully applied for nonlinear dynamic systems identification because of its exceptional capability in rendering multiresolution decomposition and capturing localized temporal and frequency information. This capability will be further exploited through the orthogonal wavelet kernel developed in this section.

A multiresolution analysis is a decomposition of $L^2(R)$, into a chain of nested subspaces $\cdots \subset V_{-1} \subset V_0 \subset V_1 \subset \cdots V_{j-1} \subset V_j \subset V_{j+1} \cdots$ such that:

1) (separation) $\bigcap_{j \in Z} V_j = \{0\}$;
2) (density) $\overline{\bigcup_{j \in Z} V_j} = L^2(R)$;
3) (scaling) $f(x) \in V_0$ if and only if $f(2^j x) \in V_j$;
4) there exists a scaling function $\varphi \in V_0$ whose integer-translates span the space V_0, and for which the set $\{\varphi(x - m),\ m \in Z\}$ is an orthonormal basis;

where j is the dilation index for the resolution level, m denotes the translation index, and the subspaces V_j are simply scaled versions of V_0. The scaling function φ is also called the father wavelet, and its binary dilations and translations constitute orthonormal bases for all V_j subspaces. Let W_j be the orthogonal complement of V_j in V_{j+1}, i.e., $W_j = V_{j+1} \ominus V_j$. V_j is called the approximation space, and W_j is called the wavelet space or detail space. The wavelet function ψ is defined such that $\{\psi(x - m)\}_{m \in Z}$ is an orthonormal basis of W_0. The wavelet space W_j is a dilation of W_0 and the basis of W_j is $\psi_{j,\,m}(x) = 2^{j/2}\psi(2^j x - m)$, i.e., $W_j = span\,(\{\psi_{j,\,m}\}_{m \in Z})$. In other words, the wavelet functions span the orthogonal complement between approximation spaces at two subsequent scales. By successively decomposing the approximation space as $V_{j+1} = W_j \oplus V_j$, it arrives that $\overline{\bigoplus_{j \in Z} W_j} = L^2(R)$ according to the density property, i.e., $L^2(R)$ can be decomposed as a direct sum of the spaces W_j.

In the presence of irregular localized features, a multiresolution learning algorithm may be required to take care of local and global complexity of the input–output map. Multiresolution approximation can be defined as a mathematical process of hierarchically decomposing the input–output approximation to capture both macroscopic and microscopic features of the system behavior. The unknown function underlying any given measured input–output data can be considered consisting of high-frequency local input–output variation details superimposed on the comparatively low frequency smooth background. At each stage, finer details are added to the coarser description, providing a successively better approximation to the input–output data.

As the cornerstone of nonlinear support vector learning algorithm, the kernel functions play an essential role in providing a general framework to represent data. However, as the conexus between wavelet multiresolution analysis and nonparametric kernel learning, the construction of the orthogonal wavelet kernel is not a trivial task due to the fact that almost all known orthonormal wavelets, except for the Haar and the Shannon, cannot be expressed in a closed form in terms of elementary analytical functions, such as the trigonometric, exponential, or rational functions [27], [28]. Without a closed form, the use of the orthogonal wavelet as kernel functions for nonparametric learning will be limited because of the resulting memory requirement, computational burden, and loss of accuracy. Although the Morlet wavelet has been capitalized on to construct wavelet kernels in the literature [23]–[25], a systematic implementation of multiresolution modeling is not provided because of the non-orthogonality of the Morlet wavelet.

In an attempt to develop multiscale support vector learning strategies that provide spatially varying resolution, one newly discovered closed-form orthogonal wavelet, the type-II raised cosine wavelet, is explored herein to construct a novel multiscale orthogonal wavelet kernel. As in the harmonic analysis signal reconstruction technology, the raised-cosine scaling function is derived from its power spectrum (spectrogram). The power spectrum of the raised-cosine scaling function is defined as [27] and [29]

$$|\hat{\varphi}(\omega)|^2 = \begin{cases} 0, & |\omega| \geq \pi(1+b) \\ \frac{1}{2}\left(1+\cos\frac{|\omega|-\pi(1-b)}{2b}\right), & \pi(1-b) < |\omega| < \pi(1+b) \\ 1, & |\omega| \leq \pi(1-b) \end{cases} \tag{13}$$

where $\hat{\varphi}(\omega)$ is the Fourier transform of the scaling function $\varphi(x)$, that is

$$\hat{\varphi}(\omega) = \int_{-\infty}^{\infty} \varphi(t) e^{-i\omega t} dt. \tag{14}$$

The spectrum of the type-I raised cosine scaling function is the usual positive square root of the power spectrum (13), and details of the type-I raised cosine wavelet can be found in [27] and [29]. The spectrum of the type-II raised cosine scaling function $\hat{\varphi}(\omega)$ derived from (13) is complex, $\hat{\varphi}(-\omega) = \overline{\hat{\varphi}(\omega)}$, and given as follows:

$$\hat{\varphi}(\omega) = \begin{cases} 0, & \omega \geq \pi(1+b) \\ \frac{1}{2}\left[1+\exp i\left(\frac{\omega-\pi(1-b)}{2b}\right)\right], & \pi(1-b) \leq \omega \leq \pi(1+b) \\ 1, & 0 \leq \omega \leq \pi(1-b). \end{cases} \tag{15}$$

As a special case of Lemarié–Meyer wavelets, the peculiarity of raised cosine wavelets lies in that they are expressible in a simple closed form, which is eminently significant for the construction of the multiscale orthogonal wavelet kernel. By using the inverse Fourier transform, the type-II raised cosine scaling function can be calculated from (15) as follows:

$$\varphi(x) = \frac{\sin\pi(1-b)x + \sin\pi(1+b)x}{2\pi x(1+2bx)} \tag{16}$$

which is a real-value function and visualized in Fig. 1.

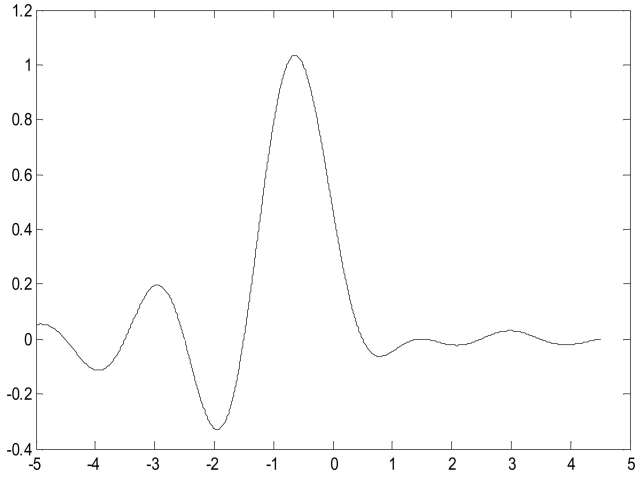

Fig. 1. Scaling function of the type-II raised cosine wavelet.

To derive the type-II raised cosine wavelet function from the explicit form of the scaling function (16) along the line of least resistance, one may apply the following theorem directly [29].

Theorem. Let $\wp$ be the set of all $g \in L^1(R)$ such that $g(x) \geq 0$ and

$$\text{supp } g \subset [-\pi/3,\ \pi/3]$$

$$g(x) \text{ is even, and } \int_{-\upsilon}^{\upsilon} g(x)dx = \pi \text{ for some } 0 < \upsilon \leq \pi/3$$

where supp $g = \overline{\{x \in R |\, g(x) \neq 0\}}$. For each $g \in \wp$, the function $\varphi(x)$ defined by its spectrum

$$\hat{\varphi}(\omega) = \frac{1}{2} + \frac{1}{2}\exp i\vartheta(\omega) \tag{17}$$

where $\vartheta(\omega) = \int_{-\omega-\pi}^{\omega-\pi} g(x)\,dx$ is a real band-limited orthonormal cardinal scaling function and the corresponding mother wavelet function $\psi(x)$ is given by

$$\psi(x) = 2\varphi(2x-1) - \varphi\left(\frac{1}{2} - x\right). \tag{18}$$

The rigorous proof of this theorem can be found in [29]. Apparently, the type-II raised cosine scaling function spectrum $\hat{\varphi}(\omega)$ given by (15) is in the form of (17). Hence, it follows from this theorem that the type-II raised cosine wavelet function $\psi(x)$ is in the form of

$$\psi\left(x+\frac{1}{2}\right) = \frac{1}{2\pi x\,[1+4bx]}\left[\sin 2\pi(1-b)x + \sin 2\pi(1+b)x\right] - \frac{1}{2\pi x\,[1-2bx]}\left[\sin\pi(1-b)x + \sin\pi(1+b)x\right]. \tag{19}$$

Parallel to the corresponding scaling function (16), the type-II raised cosine wavelet function (19) is also an asymmetric function, which is illustrated in Fig. 2.

Like most Lemarié–Meyer wavelets, raised cosine wavelets are orthonormal, band-limited, and fast-decaying in time. In particular, a distinctive and interesting property associated with type-II raised cosine wavelets is that the wavelet function

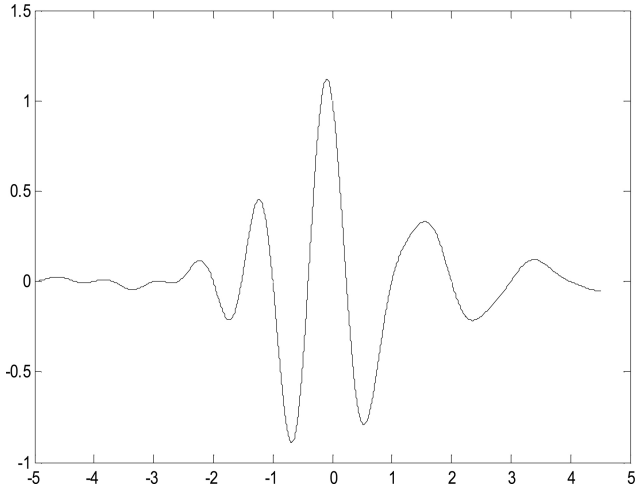

Fig. 2. Type-II raised cosine wavelet function.

is also a sampling function at the half-integers in the sense that $\psi(n+1/2) = \delta_{0n}$ where $\delta_{0n} = \begin{cases} 0, & n \neq 0 \\ 1, & n = 0 \end{cases}$ [27], [28]. The strong affinity between reproducing kernel expansion in reproducing kernel Hilbert space and sampling expansions was unveiled recently [43]–[47], which enlightens us to capitalize on the type-II raised cosine wavelet to construct an innovative kernel function. In the framework of multiresolution analysis, the 1-D multiscale wavelet kernel can be constructed in virtue of the type-II raised cosine wavelet function

$$k(x, y) = \sum_{j=\min}^{\max} \lambda_j \psi(2^j x - y) \tag{20}$$

where the resolution level index j corresponds to a ladder of subspaces for accommodating the characteristics in the different binary dilated scale levels. The kernel (20) can be easily generalized to the multidimensional version by using the tensor product

$$k(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{d} \sum_{j=\min_i}^{\max_i} \lambda_j \psi(2^j x_i - y_i) \tag{21}$$

where d stands for the dimension of the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$. Most commonly-used kernel functions can be classified into either translation-invariant kernels, such as the radial basis function (RBF) kernel and inverse multiquadric kernel, or rotation-invariant kernels, such as the sigmoid kernel and polynomial kernel. However, as a multiscale kernel function, the orthogonal wavelet kernel defined by (21) is neither translation-invariant nor rotation-invariant.

In particular, the asymmetry inherited in the type-II raised cosine wavelet function distinguishes the kernel function (21) from other kernel functions, including the Morlet wavelet kernel. The Morlet wavelet kernel defined by

$$k(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{d} \phi\left(\frac{x_i - y_i}{\delta}\right) \tag{22}$$

where $\phi(x) = cos(1.75x)exp(-\frac{x^2}{2})$, is apparently symmetric and translation-invariant. Albeit the possibility of using asymmetric kernels for support vector learning was mentioned and investigated in [33] and [48], the related research has been sparse. In this paper, the adaptability of LP-SVR to more general kernels makes it possible to represent and approximate unknown systems by asymmetric kernel expansion. The anisotropic representation capability inherited in the asymmetric kernel allows it to go beyond the limit of symmetric kernels in estimating complex dependencies and representing irregular features. It is also noteworthy that the cross-kernel used for solving the linear operator equation via SV learning is essentially an asymmetric kernel and the expansion on asymmetric kernels is used to represent the approximation solution to the linear operator equation [49].

IV. Application to Nonlinear Dynamic Modeling

The essence of modeling is prediction, and forecasting future behavior based on past observations has been a long standing topic in system identification and time series modeling [50]–[52]. According to the different regression vectors used, the identification model for dynamical systems can be categorized as the series-parallel model or NARX model and the parallel model or nonlinear output error model [36]. Without coupling to the real systems, the nonlinear parallel models are emancipated from relying on the outputs of the actual systems. In effect, the parallel model (3) is a recurrent NARX model, whose computational capability and equivalence to the Turing machine were emphasized in [53] and [54]. The identification of the series-parallel model amounts to building a one-step ahead predictor, while the identification of the parallel model is for long-term/mid-term prediction.

In the realm of nonlinear systems identification, there is general consensus that one of the most formidable technical challenges is how to build a model usable in parallel configuration, which is much more intractable than building the series-parallel model due to the feedback involved. However, a multitude of applications, e.g., fault detection and diagnosis, predictive control, simulation, etc. require a parallel model since a prediction of many steps into the future is needed.

In theory, long-term/mid-term predictions can be obtained from a short-term predictor, for example a one-step ahead predictor, simply by applying the short predictor many times (steps) in an iterative way. This is called iterative prediction [50]–[52]. The other way, called direct prediction, provides a once-completed predictor with a long-term prediction step, and the specified multistep prediction can then be obtained directly from the established predictor in a manner similar to computing one-step predictions [50], [51]. The main downside of the direct modeling approach is that it requires different models for different steps ahead prediction. It is generally believed that the iterative prediction approach is in most cases more efficient than the direct approach assuming that the dynamics underlying the time series are correctly specified by the model [50].

In this simulation study, to demonstrate the superiority and effectiveness of the proposed novel kernel function for nonlinear dynamical system modeling, the LP-SVR learning algorithm with multiscale asymmetric orthogonal wavelet kernel is

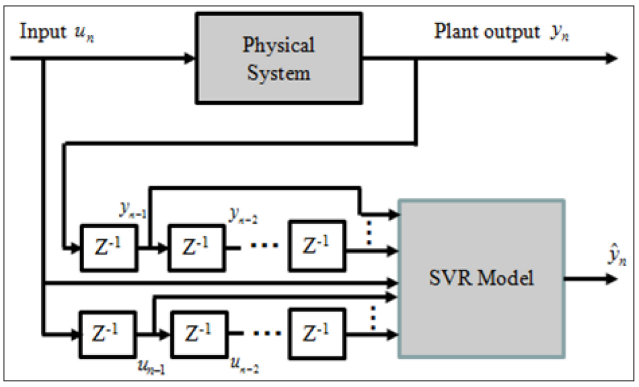

Fig. 3. Model in series-parallel configuration.

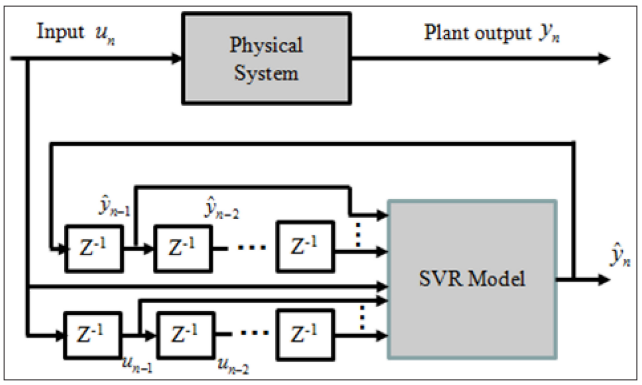

Fig. 4. Model in parallel configuration.

used to build the parallel models for a benchmark hydraulic robot arm dataset and the well-known Box and Jenkins' dataset in virtue of the iterative prediction approach. Although these two datasets have been used widely for performance evaluation of various system identification methods in the literature [11], [17], [25], [55]–[58], to our best knowledge, this is the first study on learning the parallel models for long-term/mid-term prediction on these benchmark datasets.

Partitioning the benchmark datasets into training and validation subsets, the identification procedure includes two phases. The one-step ahead predictor, i.e., the series-parallel model as shown in Fig. 3, is first learned on the training dataset, and then in the second phase the procured one-step ahead predictor is used in the parallel configuration for long-term/mid-term prediction on the validation dataset, as shown in Fig. 4.

For the sake of comparison, several commonly used kernel functions are employed for modeling on the same dataset as well, such as the Morlet wavelet kernel defined by (22), the Gaussian RBF kernel defined by

$$k(\boldsymbol{x},\ \boldsymbol{y}) = \exp\left(\frac{-\|\boldsymbol{x}-\boldsymbol{y}\|^2}{2\sigma^2}\right) \tag{23}$$

the polynomial kernel defined by

$$k(\boldsymbol{x},\ \boldsymbol{y}) = (1 + \langle \boldsymbol{x},\ \boldsymbol{y}\rangle)^q \tag{24}$$

the inverse multiquadric kernel defined by

$$k(\boldsymbol{x},\ \boldsymbol{y}) = \frac{1}{\sqrt{\|\boldsymbol{x}-\boldsymbol{y}\|^2 + c^2}} \tag{25}$$

and the B-spline kernel defined by

$$k(\boldsymbol{x},\ \boldsymbol{y}) = \prod_{i=1}^{d} B_{2J+1}\left(x_i - y_i\right) \tag{26}$$

where σ, q, c, J are the adjustable parameters of the above kernel functions. For the B-spline kernel, B-spline function $B_\ell(\cdot)$ represents a particular example of a convolutional basis and can be expressed explicitly as [59], [60]

$$B_\ell(x) = \frac{1}{\ell!}\sum_{r=0}^{\ell+1}\binom{\ell+1}{r}(-1)^r\left(x + \frac{\ell+1}{2} - r\right)_+^{\ell} \tag{27}$$

where the function $(\cdot)_+$ is defined as the truncated power function, that is

$$x_+ = \begin{cases} x, & \text{for } x > 0 \\ 0, & \text{otherwise.} \end{cases} \tag{28}$$

A. Hydraulic Robot Arm Dynamical System Identification

For the hydraulic robot arm dynamical system, the position of a robot arm is controlled by a hydraulic actuator. The control input u_n represents the size of the valve opening through which oil flow into the actuator, and the output y_n is a measure of the oil pressure which determines the robot arm position. In modeling this dynamical system, with the aim of achieving a fair comparison, the same regressor $[y_{n-1}, y_{n-2}, y_{n-3}, u_{n-1},\ u_{n-2}]$ and dataset partition scheme as those in the literature are adopted herein [11], [17], [25], [55]. The first half of the data set containing 511 training data pairs is used for training in series-parallel configuration, and the other half for validation data in parallel configuration.

In the training phase, the model (12) with $\boldsymbol{y}_{n-1} = [y_{n-1}, y_{n-2}, y_{n-3}]$ and $\boldsymbol{u}_n = [u_{n-1}, u_{n-2}]$ is learned by LP-SVR to attain the one-step ahead approximator. Upon training completion, our objective is to provide satisfactory multistep prediction without using the actual system output y_n, i.e., to validate the model in parallel configuration as follows:

$$\hat{y}_n = \sum_{i\in SV}\beta_i(k_1(\boldsymbol{y}_{i-1}, \hat{\boldsymbol{y}}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)) \tag{29}$$

where $\hat{\boldsymbol{y}}_{n-1} = [\hat{y}_{n-1}, \hat{y}_{n-2}, \hat{y}_{n-3}]$. The approximation accuracies on the training and validation datasets are evaluated by calculating the root mean square error (RMSE)

$$E_{RMS} = \sqrt{\frac{1}{M}\sum_{n=1}^{M}\left[\hat{y}_n - y_n\right]^2} \tag{30}$$

where $\hat{y}_n$ is the estimated output of the model and M is the number of data in the dataset for evaluation. The validation accuracy is crucial in evaluating generalization performance of the model. In applying SVR with kernel functions to train the model, manual tuning of the kernel parameters as well as ε and C for optimum results is required.

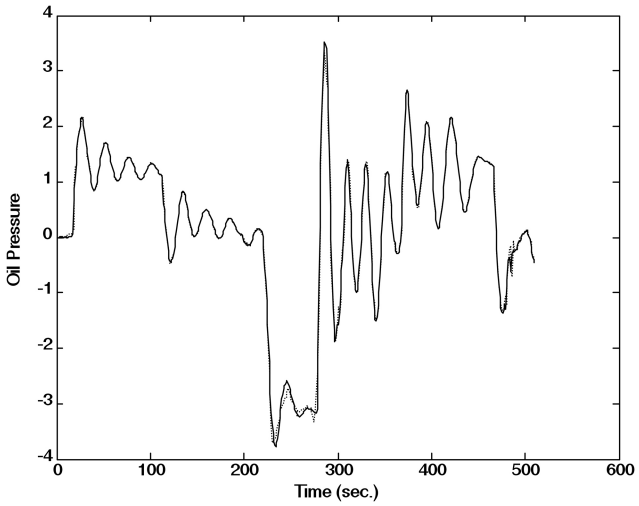

Fig. 5. Training in series-parallel configuration for the model (12) of robot arm by LP-SVR with multiscale asymmetric wavelet kernel (solid line: actual system output, dotted line: model output).

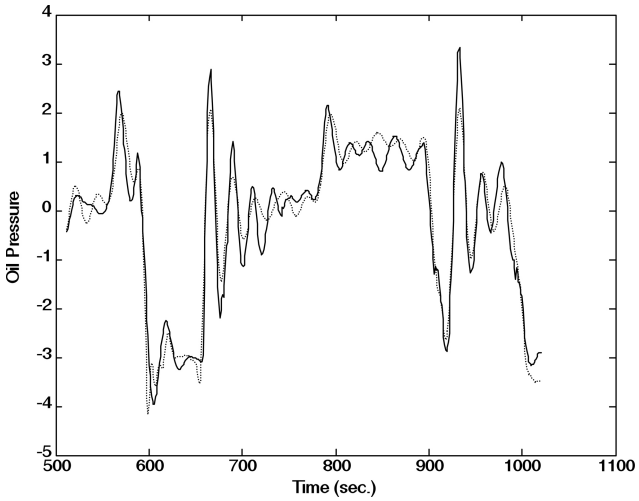

Fig. 6. Validation in parallel configuration for the model (29) of robot arm by LP-SVR with multiscale asymmetric wavelet kernel (solid line: actual system output, dotted line: model output).

By setting the parameters $\varepsilon = 0.02$, $C = 0.46$ and the kernel functions $k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1})$ and $k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)$ as

$$k_1(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{3} \sum_{j=-5}^{-4} \lambda_j \psi(2^j x_i - y_i) \tag{31}$$

$$k_2(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j=-4}^{0} \lambda_j \psi(2^j x_1 - y_1) \times \sum_{j=-9}^{-1} \lambda_j \psi(2^j x_2 - y_2) \tag{32}$$

where the kernel parameters $\lambda_j = 2^{(j/\mu)}$ and $\mu = 2.9$, the training result based on the multiscale asymmetric wavelet kernel (31) and (32) is illustrated in Fig. 5, and the training RMSE is 0.1027. The attained model is subsequently validated on the validation dataset in parallel configuration for long-term/mid-term prediction, and the validation result is shown in Fig. 6 and Table I. Following the same procedure, the other kernel functions are also used to train model (12) by LP-SVR and QP-SVR, respectively. After tuning the parameters for optimum results, the validation performances of models learned by LP-SVR in parallel configuration are depicted in Figs. 7–11. In conjunction with the model sparsity, the RMSEs on the training and validation datasets obtained by these comparative models are listed in Table I for LP-SVR and Table II for QP-SVR.

Measured by the support vector ratio, the sparsity of the model with multiscale asymmetric wavelet kernel is commensurate with the models adopting polynomial kernel and

TABLE I
ROBOT ARM PARALLEL MODEL IDENTIFICATION BY LP-SVR WITH DIFFERENT COMPOSITE KERNEL FUNCTIONS

Kernel functions k_1 and k_2	SV ratio	Training RMSE	Validation RMSE
Gaussian RBF kernel	11.0%	0.2005	0.8456
Polynomial kernel	2.0%	0.0913	0.7089
B-Spline kernel	16.4%	0.0717	0.4940
Inverse multi-quadric kernel	5.5%	0.1189	0.7167
Morlet wavelet kernel	3.1%	0.1450	0.6816
Multi-scale RC II wavelet kernel	3.7%	0.1027	0.4150

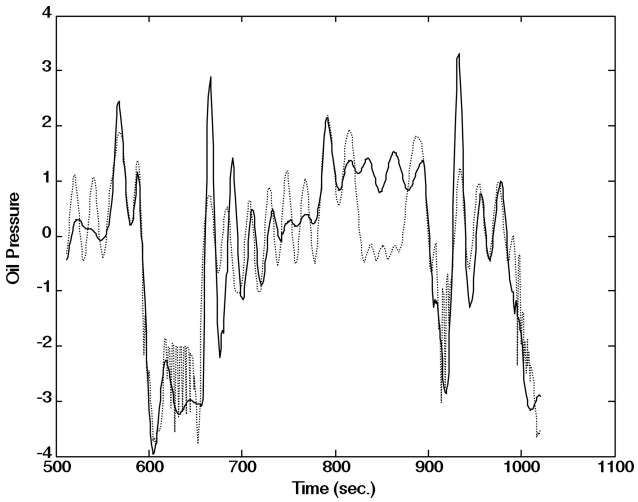

Fig. 7. Validation in parallel configuration for the model (29) of robot arm by LP-SVR with Gaussian RBF kernel (solid line: actual system output, dotted line: model output).

TABLE II
ROBOT ARM PARALLEL MODEL IDENTIFICATION BY QP-SVR WITH DIFFERENT COMPOSITE KERNEL FUNCTIONS

Kernel functions k_1 and k_2	SV ratio	Training RMSE	Validation RMSE
Gaussian RBF kernel	40.7%	0.3024	0.6973
Polynomial kernel	27.6%	0.0893	0.7386
B-Spline kernel	31.7%	0.0529	0.6296
Inverse multi-quadric kernel	50.9%	0.1483	0.6532
Morlet wavelet kernel	32.9%	0.1493	0.7365

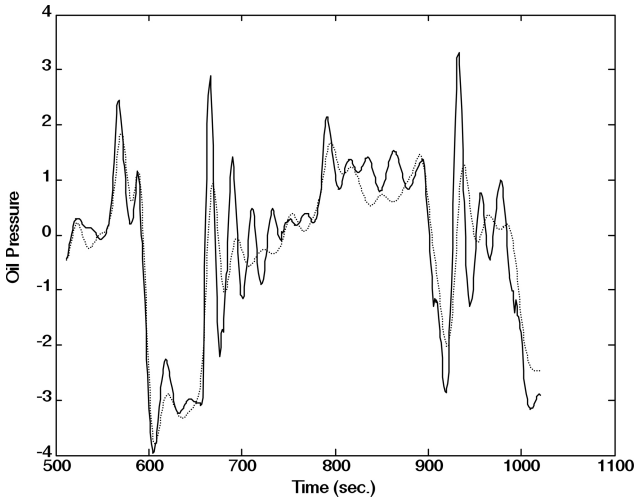

Fig. 8. Validation in parallel configuration for the model (29) of robot arm by LP-SVR with polynomial kernel (solid line: actual system output, dotted line: model output).

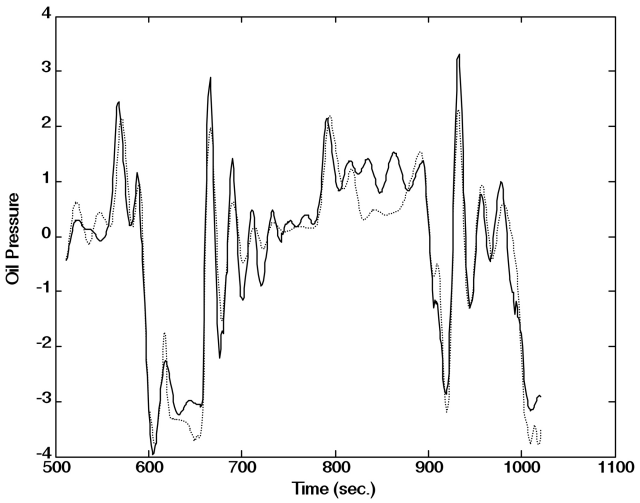

Fig. 9. Validation in parallel configuration for the model (29) of robot arm by LP-SVR with B-spline kernel (solid line: actual system output, dotted line: model output).

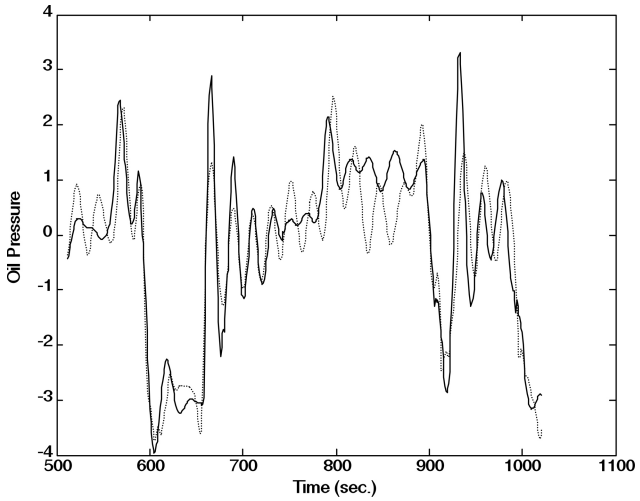

Fig. 10. Validation in parallel configuration for the model (29) of robot arm by LP-SVR with inverse multiquadric kernel (solid line: actual system output, dotted line: model output).

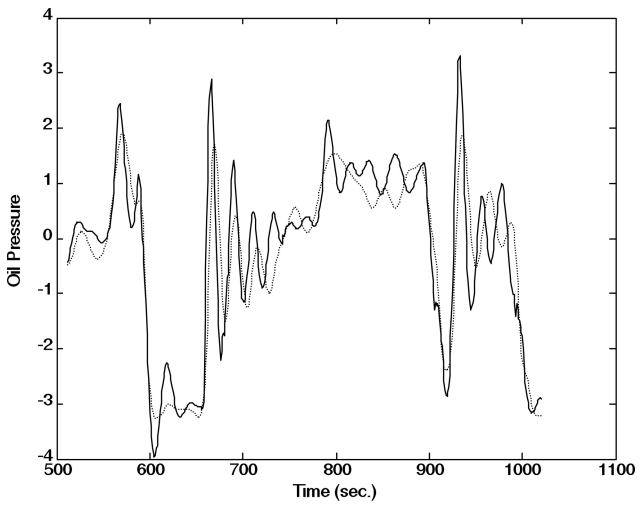

Fig. 11. Validation in parallel configuration for the model (29) of robot arm by LP-SVR with Morlet wavelet kernel (solid line: actual system output, dotted line: model output).

Morlet wavelet kernel in Table I. It is evident that the validation accuracy shown in parallel configuration is considerably improved by using the multiscale asymmetric wavelet kernel, which implies excellent generalization performance.

In parallel configuration, the errors for the s-step prediction are the accumulation of the errors of the previous $(s-1)$ steps. Generally, the longer the forecasting horizon, the larger the accumulated errors are and the less accurate the iterative method is. Hence, it is remarkable that, while using the identical regressor on the same training and validation datasets, this parallel model validation accuracy is even better than some of those obtained in series-parallel configuration by other popular learning strategies. For example, the RMSE was 0.467 for a one-hidden-layer sigmoid neural network case and 0.579 for a wavelet network case [55].

In terms of computing time for training, LP-SVR is around seven times faster than QP-SVR on this dataset (Intel Core i5 processor), and the computing resource required by QP-SVR might become prohibitively expensive when increasing the size of the training dataset. It is also obvious by comparing the model sparsities in Tables I and II that the LP-SVR substantially exceeds QP-SVR in producing succinct model representations.

B. Box and Jenkins' Identification Problem

The well-known Box and Jenkins' gas furnace dataset was recorded from a combustion process of a methane–air mixture. The original dataset consists of 296 input–output data pairs that were recorded at a sampling rate of 9 s. The gas combustion process has one input variable, gas flow rate u_n, and one output variable, the concentration of carbon dioxide (CO_2) in the outlet gas, y_n. The instantaneous value of the output y_n can be regarded as being influenced by ten variables y_{n-1}, y_{n-2}, $\cdots$, y_{n-4}, u_{n-1}, u_{n-2}, $\cdots$, u_{n-6} [56], [58]. In modeling this dynamical system, the regressor $[y_{n-1},\ y_{n-2},\ u_{n-2},\ u_{n-3},\ u_{n-4}]$ is employed herein. The

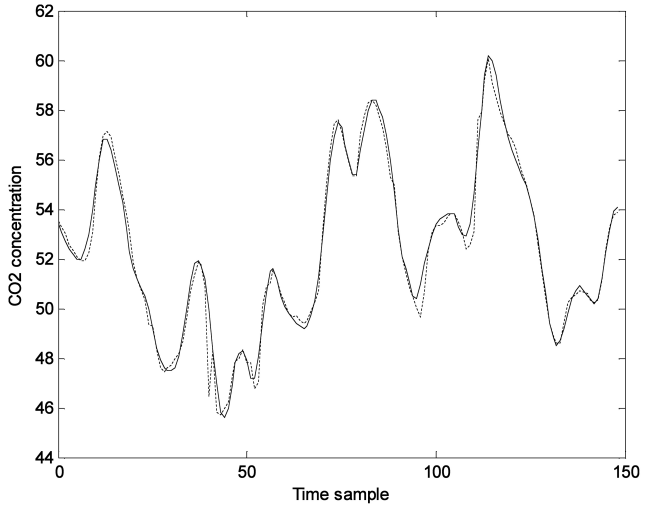

Fig. 12. Training in series-parallel configuration for the model (12) of a gas furnace by LP-SVR with multiscale asymmetric wavelet kernel (solid line: actual system output, dotted line: model output).

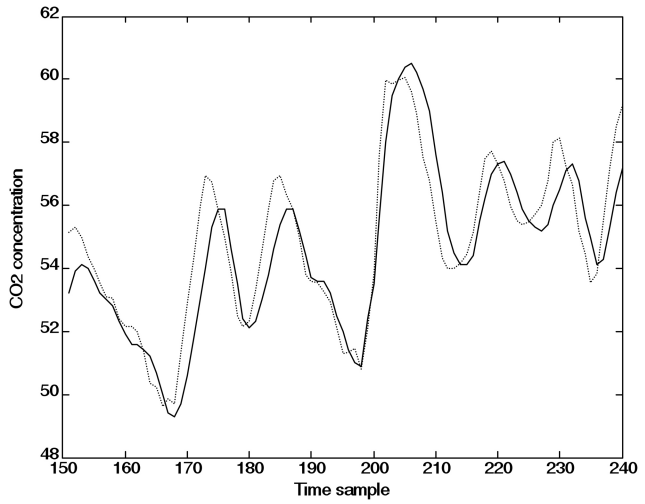

Fig. 13. Validation in parallel configuration for the model (35) of a gas furnace by LP-SVR with multiscale asymmetric wavelet kernel (solid line: actual system output, dotted line: model output).

first 150 data pairs are used for training in series-parallel configuration, and the subsequent 140 data pairs are used for validation in parallel configuration. Due to the different distribution and magnitude order of the measurements in this dataset, the data standardization and proper rescaling are necessary before the start of training [61].

In training the model (12) with $\boldsymbol{y}_{n-1} = [y_{n-1},\ y_{n-2}]$ and $\boldsymbol{u}_n = [u_{n-2},\ u_{n-3},\ u_{n-4}]$, the kernel functions $k_1(\boldsymbol{y}_{i-1}, \boldsymbol{y}_{n-1})$ and $k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)$ are set as

$$k_1(\boldsymbol{x}, \boldsymbol{y}) = \prod_{i=1}^{2} \sum_{j=1}^{2} \lambda_j \psi(2^j x_i - y_i) \tag{33}$$

$$k_2(\boldsymbol{x}, \boldsymbol{y}) = \sum_{j=-11}^{0} 2^j \psi(2^j x_1 - y_1) \times \prod_{i=2}^{3} \sum_{j=-4}^{1} 2^j \psi(2^j x_i - y_i) \tag{34}$$

where the kernel parameters are $\lambda_j = 2^{(j/\mu)}$ and $\mu = 2$. The training result based on the multiscale asymmetric wavelet

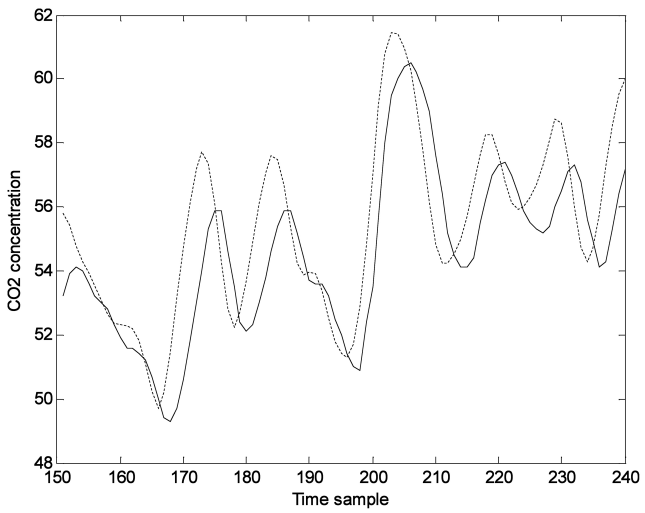

Fig. 14. Validation in parallel configuration for the model (35) of a gas furnace by LP-SVR with Gaussian RBF kernel (solid line: actual system output, dotted line: model output).

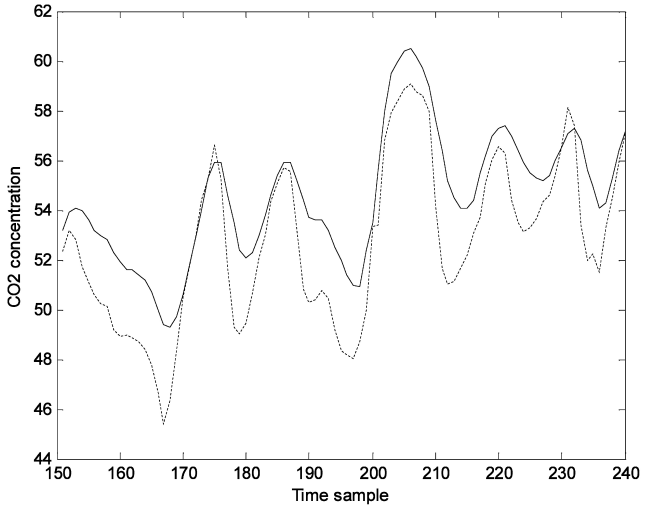

Fig. 15. Validation in parallel configuration for the model (35) of a gas furnace by LP-SVR with polynomial kernel (solid line: actual system output, dotted line: model output).

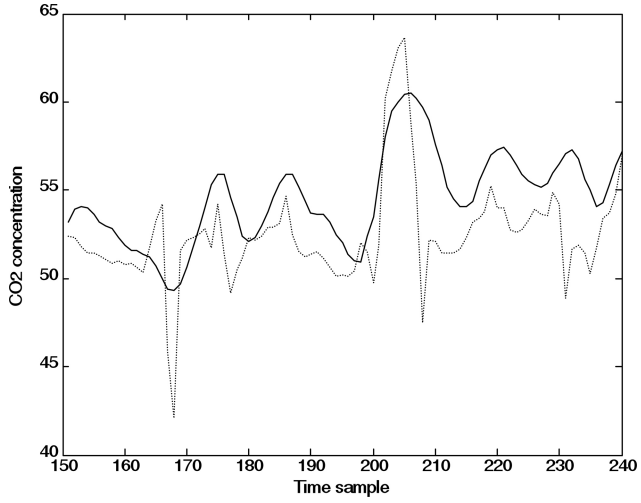

Fig. 16. Validation in parallel configuration for the model (35) of a gas furnace by LP-SVR with B-spline kernel (solid line: actual system output, dotted line: model output).

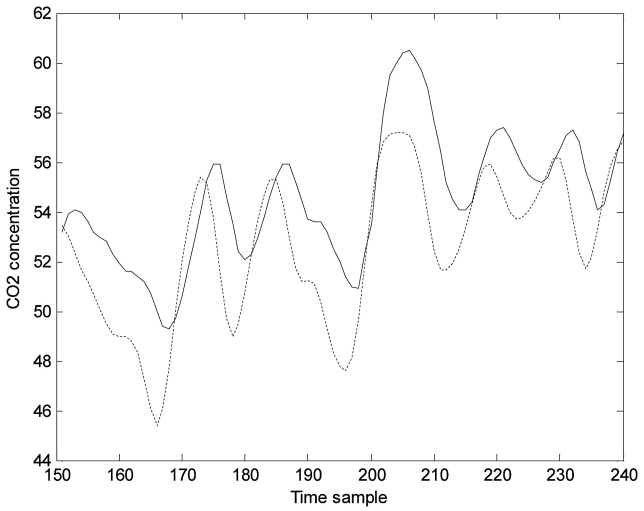

Fig. 17. Validation in parallel configuration for the model (35) of a gas furnace by LP-SVR with inverse multiquadric kernel (solid line: actual system output, dotted line: model output).

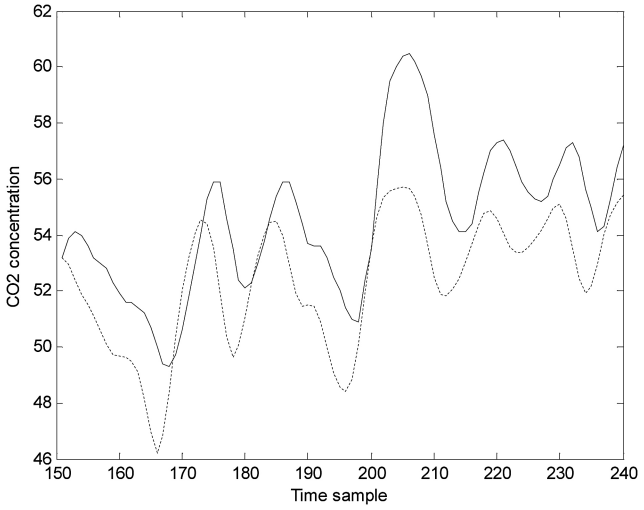

Fig. 18. Validation in parallel configuration for the model (35) of a gas furnace by LP-SVR with Morlet wavelet kernel (solid line: actual system output, dotted line: model output).

kernel (33) and (34) is illustrated in Fig. 12, and the corresponding RMSE is 0.5001. Subsequently, the model is validated in parallel configuration

$$\hat{y}_n = \sum_{i \in SV} \beta_i (k_1(\boldsymbol{y}_{i-1}, \hat{\boldsymbol{y}}_{n-1}) + k_2(\boldsymbol{u}_i, \boldsymbol{u}_n)) \tag{35}$$

where $\hat{\boldsymbol{y}}_{n-1} = [\hat{y}_{n-1}, \hat{y}_{n-2}]$. The validation results are plotted in Fig. 13, and the corresponding RMSE is 1.1956. The model is also trained with other kernel functions by LP-SVR and QP-SVR, respectively. The validation performances of models learned by LP-SVR in parallel configuration are depicted in Figs. 14–18. Together with the model sparsity, the training RMSE and validation RMSE are listed in Tables III and IV. Again, similar to that in the robot arm case study, the advantages of multiscale asymmetric wavelet kernel in modeling accuracy, generalization capability, and model sparsity with LP-SVR are clearly demonstrated through the simulation results.

TABLE III
GAS FURNACE PARALLEL MODEL IDENTIFICATION BY LP-SVR WITH DIFFERENT COMPOSITE KERNEL FUNCTIONS

Kernel functions k_1 and k_2	SV ratio	Training RMSE	Validation RMSE
Gaussian RBF kernel	1.3%	1.1853	1.9001
Polynomial kernel	12.8%	0.7204	2.2744
B-Spline kernel	73.8%	0.0988	3.2156
Inverse multi-quadric kernel	1.3%	1.3254	2.6664
Morlet wavelet kernel	68.5%	0.2094	2.6469
Multi-scale RC II wavelet kernel	8.7%	0.5001	1.1956

TABLE IV
GAS FURNACE PARALLEL MODEL IDENTIFICATION BY QP-SVR WITH DIFFERENT COMPOSITE KERNEL FUNCTIONS

Kernel functions k_1 and k_2	SV ratio	Training RMSE	Validation RMSE
Gaussian RBF kernel	80.5%	1.0421	2.6201
Polynomial kernel	77.9%	0.6667	2.4571
B-Spline kernel	62.4%	0.8426	1.9846
Inverse multi-quadric kernel	78.5%	0.6140	2.4761
Morlet wavelet kernel	88.6%	0.1109	2.7435

V. CONCLUSION

In spite of the prosperity of wavelet theory in the realms of signal processing and multiscale modeling, the utility of the wavelet multiresolution analysis in kernel learning has been quite limited owing to the lack of explicit closed-form expressions of almost all orthonormal wavelets. The main contribution of this paper is to bridge the gap between wavelet multiresolution analysis and support vector learning by constructing a closed-form multiscale orthogonal wavelet kernel and demonstrating its value in nonlinear dynamic modeling. Also, in the scenario of LP-SVR with composite kernel, the efficacy of the proposed multiscale orthogonal wavelet kernel function is evaluated and confirmed through simulation study.

Based on the illuminating discovery of nontrivial closed-form orthogonal wavelets, a new multiscale asymmetric orthogonal wavelet kernel, the type-II raised cosine wavelet kernel, is devised in this paper. With the capability to represent complex dependencies at different scales, the proposed orthogonal wavelet kernel function enables multiscale support vector learning under the framework of multiresolution analysis. On the other hand, the anisotropy of the asymmetric kernel confers more flexibility in representing irregular complex dependencies, and thereby this research significantly complements the dearth of the investigation on asymmetric kernel functions in the realm of computational learning theory. The advantages of the proposed kernel design are demonstrated on two challenging nonlinear dynamic system modeling problems, where

parallel models capable of long-term/mid-term prediction are developed and shown to have substantial benefits in both modeling accuracy and sparsity when compared to other widely used kernel functions.

REFERENCES

[1] B. Schölkopf and A. J. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2002.

[2] T. Hofmann, B. Schölkopf, and A. J. Smola, "Kernel methods in machine learning," *Ann. Statist.*, vol. 36, no. 3, pp. 1171–1220, 2008.

[3] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–202, Mar. 2001.

[4] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statist. Comput.*, vol. 14, no. 3, pp. 199–222, 2004.

[5] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, MA, USA: Cambridge Univ. Press, 2000.

[6] V. N. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed. Berlin, Germany: Springer-Verlag, 2000.

[7] J. Krebs, "Support vector regression for the solution of linear integral equations," *Inverse Problems*, vol. 27, no. 6, pp. 1–23, 2011.

[8] J. L. Rojo-Álvarez, G. Camps-Valls, M. Martinez-Ramon, E. Soria-Olivas, A. Navia-Vazquiz, and A. R. Figueiras-Vidal, "Support vector machines framework for linear signal processing," *Signal Process.*, vol. 85, no. 12, pp. 2316–2326, Dec. 2005.

[9] H. R. Zhang, X. D. Wang, C. J. Zhang, and X. S. Cai, "Robust identification of nonlinear dynamic using support vector machine," *IEE Proc. Sci. Meas. Technol.*, vol. 153, no. 3, pp. 125–129, 2006.

[10] W. C. Chan, C. W. Chan, K. C. Cheung, and C. J. Harris, "On the modeling of nonlinear dynamic systems using support vector neural networks," *Eng. Appl. Artif. Intell.*, vol. 14, no. 2, pp. 105–113, 2001.

[11] A. Gretton, A. Doucet, R. Herbrich, P. J. W. Rayner, and B. Schölkopf, "Support vector regression for black-box system identification," in *Proc. 11th IEEE Workshop Statist. Signal Process.*, 2001, pp. 341–344.

[12] V. Kecman, *Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models*. Cambridge, MA, USA: MIT Press, 2001.

[13] V. Kecman, T. Arthanari, and I. Hadžić, "LP and QP based learning from empirical data," in *Proc. Int. Joint Conf. Neural Netw.*, 2001, pp. 2451–2455.

[14] G. Bloch, F. Lauer, C. Guillaume, and C. Yann, "Support vector regression from simulation data and few experimental samples," *Inform. Sci.*, vol. 178, no. 20, pp. 3813–3827, Oct. 2008.

[15] A. Smola, B. Schölkopf, and G. Rätsch, "Linear programs for automatic accuracy control in regression," in *Proc. 9th Int. Conf. Artif. Neural Netw.*, 1999, pp. 575–580.

[16] V. Kecman and I. Hadžić, "Support vectors selection by linear programming," in *Proc. Int. Joint Conf. Neural Netw.*, 2000, pp. 193–198.

[17] Z. Lu and J. Sun, "Non-Mercer hybrid kernel for linear programming support vector regression in nonlinear systems identification," *Appl. Soft Comput.*, vol. 9, no. 1, pp. 94–99, 2009.

[18] V. N. Vapnik, *Statistical Learning Theory*. New York, NY, USA: Wiley, 1998.

[19] M. Martinez-Ramon, J. L. Rojo-Álvarez, G. Camps-Valls, J. Muñoz-Marí, A. Artes-Rodriguez, and A. R. Figueiras-Vidal, "Support vector machines for nonlinear kernel ARMA system identification," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1617–1622, Nov. 2006.

[20] H. Wang, F. Sun, Y. Cai, and Z. Zhao, "Online chaotic time series prediction using unbiased composite kernel machine via Cholesky factorization," *Soft Comput.*, vol. 14, no. 9, pp. 931–944, 2010.

[21] Z. Lu, J. Sun, and K. Butts, "Linear programming SVM-ARMA$_{2K}$ with application in engine system identification," *IEEE Trans. Autom. Sci. Eng.*, vol. 8, no. 4, pp. 846–854, Oct. 2011.

[22] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd ed. San Francisco, CA, USA: Academic, 2009.

[23] L. Zhang, W. Zhou, and L. Jiao, "Wavelet support vector machine," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 34, no. 1, pp. 34–39, Feb. 2004.

[24] Q. Wu, "The forecasting model based on wavelet ν-support vector machine," *Expert Syst. Appl.*, vol. 36, no. 4, pp. 7604–7610, 2009.

[25] Z. Lu, J. Sun, and K. Butts, "Linear programming support vector regression with wavelet kernel: A new approach to nonlinear dynamical systems identification," *Math. Comput. Simulation*, vol. 79, no. 7, pp. 2051–2063, 2009.

[26] A. Sharifi, M. A. Shoorehdeli, and M. Teshnehlab, "Design of a prediction model for cement rotary kiln using wavelet projection fuzzy inference system," *Cybern. Syst.*, vol. 43, no. 5, pp. 369–397, 2012.

[27] G. G. Walter and J. Zhang, "Orthonormal wavelets with simple closed-form expressions," *IEEE Trans. Signal Process.*, vol. 46, no. 8, pp. 2248–2251, 1998.

[28] A. I. Zayed and G. G. Walter, "Wavelets in closed forms," in *Wavelet Transforms and Time-Frequency Signal Analysis*, L. Debnath, Ed. Cambridge, MA, USA: Birkhäuser, 2000.

[29] G. G. Walter and X. P. Shen, *Wavelets and Other Orthogonal Systems*, 2nd ed. Boca Raton, FL, USA: Chapman and Hall/CRC, 2000.

[30] P. Michels, "Asymmetric kernel functions in non-parametric regression analysis and prediction," *The Statistician*, vol. 41, no. 4, pp. 439–454, 1992.

[31] M. Mackenzie and A. Kiet Tieu, "Asymmetric kernel regression," *IEEE Trans. Signal Process.*, vol. 15, no. 3, pp. 276–282, 2004.

[32] A. Yilmaz, "Kernel-based object tracking using asymmetric kernels with adaptive scale and orientation selection," *Mach. Vision Appl.*, vol. 22, no. 2, pp. 255–268, 2011.

[33] K. Tsuda, "Support vector classifier with asymmetric kernel functions," in *Proc. Eur. Symp. Artif. Neural Netw.*, 1999, pp. 183–188.

[34] S. D. Marchi and R. Schaback, "Nonstandard kernels and their applications," *Dolomites Res. Notes Approximation*, vol. 2, pp. 16–43, 2009.

[35] G. Fasshauer, "Positive definite kernels: Past, present and future," *Dolomites Res. Notes Approximation*, vol. 4, pp. 21–63, 2011.

[36] O. Nelles, *Nonlinear System Identification: From Classical Approaches to Neural Networks and Fuzzy Models*. Berlin, Germany: Springer, 2001.

[37] O. Nelles, "On the identification with neural networks as series-parallel and parallel models," in *Proc. Int. Conf. Artif. Neural Netw.*, Oct. 1995, pp. 255–260.

[38] J. A. K. Suykens and J. Vandewalle, "Recurrent least squares support vector machines," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 47, no. 7, pp. 1109–1114, Jul. 2000.

[39] J. A. K. Suykens, T. V. Gestel, J. DeBrabanter, B. DeMoor, and J. Vandewalle, *Least Squares Support Vector Machines*. Singapore: World Scientific, 2002.

[40] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inform. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[41] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution," *Commun. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, 2006.

[42] F. Chen, Q. Wang, S. Wang, W. Zhang, and W. Xu, "Object tracking via appearance modeling and sparse representation," *Image Vision Comput.*, vol. 29, no. 11, pp. 787–796, 2011.

[43] A. I. Zayed, *Advances in Shannon's Sampling Theory*. Boca Raton, FL, USA: CRC Press, 1993.

[44] C. V. M. van der Mee, M. Z. Nashed, and S. Seatzu, "Sampling expansions and interpolation in unitarily translation invariant reproducing kernel Hilbert spaces," *Advances Comput. Math.*, vol. 19, no. 4, pp. 355–372, 2003.

[45] D. Han, M. Z. Nashed, and Q. Sun, "Sampling expansions in reproducing kernel Hilbert and Banach spaces," *Numer. Functional Anal. Optimization*, vol. 30, no. 9, pp. 971–987, 2009.

[46] M. Z. Nashed, and G. G. Walter, "General sampling theorems for functions in reproducing kernel Hilbert spaces," *Math. Control Signals Syst.*, vol. 4, no. 4, pp. 363–390, 1991.

[47] M. Z. Nashed and G. G. Walter, "Reproducing kernel Hilbert space from sampling expansions," *Contemporary Math.*, vol. 190, pp. 221–226, 1995.

[48] B. Schölkopf, S. Mika, C. J. C. Burges, P. Knirsch, K. R. Müller, G. Ratsch, and A. J. Smola, "Input space versus feature space in kernel-based methods," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1000–1017, Sep. 1999.

[49] J. Weston, A. Gammerman, M. O. Stitson, V. Vapnik, V. Vovk, and C. Watkins, "Support vector density estimation," in *Advances in Kernel Methods*, B. Schölkopf, C. J. C. Burges, A. J. Smola, Eds. Cambridge, MA, USA: MIT Press, 1999.

[50] H. L. Wei and S. A. Billings, "Long term prediction of non-linear time series using multiresolution wavelet models," *Int. J. Control*, vol. 79, no. 6, pp. 569–580, 2006.

[51] K. Judd and M. Small, "Towards long-term prediction," *Physica D*, vol. 136, nos. 1–2, pp. 31–44, 2000.

[52] G. Bontempi and S. B. Taieb, "Conditionally dependent strategies for multi-step-ahead prediction in local learning," *Int. J. Forecasting*, vol. 27, no. 3, pp. 689–699, 2011.
[53] H. T. Siegelmann, B. G. Horne, and C. L. Giles, "Computational capabilities of recurrent NARX neural networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 27, no. 2, pp. 208–215, Apr. 1997.
[54] T. Lin, B. G. Horne, P. Tino, and C. L. Giles, "Learning long-term dependencies in NARX recurrent neural networks," *IEEE Trans. Neural Netw.*, vol. 7, no. 6, pp. 1329–1338, Nov. 1996.
[55] J. Sjöberg, Q. Zhang, L. Ljung, A. Berveniste, B. Delyon, P. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: A unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.
[56] D. Kukolj and E. Levi, "Identification of complex systems based on neural and Takagi–Sugeno fuzzy model," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 1, pp. 272–282, Jan. 2004.
[57] S. K. Oh and W. Pedrycz, "The design of self-organizing polynomial neural networks," *Inform. Sci.*, vol. 141, nos. 3–4, pp. 237–258, 2002.
[58] H. Du and N. Zhang, "Application of evolving Takagi–Sugeno fuzzy model to nonlinear system identification," *Appl. Soft Comput.*, vol. 8, no. 1, pp. 676–686, Jan. 2008.
[59] P. Wittek and C. L. Tan, "Compactly supported basis functions as support vector kernels for classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 2039–2050, Oct. 2011.
[60] O. Christensen, *Functions, Spaces, and Expansions: Mathematical Tools in Physics and Engineering*. Cambridge, MA, USA: Birkhäuser, 2010.
[61] S. Ghahramani, *Fundamentals of Probability With Stochastic Processes*, 3rd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2005.

Zhao Lu (M'08) received the M.S. degree in control theory and engineering from Nankai University, Tianjin, China, in 2000, and the Ph.D. degree in electrical engineering from the University of Houston, Houston, TX, USA, in 2004.

From 2004 to 2006, he was a Post-Doctoral Research Fellow with the Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI, USA, and with the Department of Naval Architecture and Marine Engineering, University of Michigan, Ann Arbor, MI, USA, respectively. Since 2007, he has been with the faculty of the College of Engineering, Tuskegee University, Tuskegee, AL, USA, where he is currently an Associate Professor with the Department of Electrical Engineering. His current research interests include machine learning, computational intelligence, and nonlinear control theory.

Jing Sun (M'89–SM'00–F'04) received the B.S. and M.S. degrees from the University of Science and Technology of China, Hefei, China, in 1982 and 1984, respectively, and the Ph.D. degree from the University of Southern California, Los Angeles, CA, USA, in 1989.

From 1989 to 1993, she was an Assistant Professor with the Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI, USA. She joined the Ford Research Laboratory, Dearborn, MI, USA, in 1993, where she was with the Powertrain Control Systems Department. After spending almost ten years in the industry, she returned to academia and joined the faculty of the College of Engineering, University of Michigan, Ann Arbor, MI, USA, in 2003, where she is currently a Professor with the Department of Naval Architecture and Marine Engineering and the Department of Electrical Engineering and Computer Science. She holds over 30 U.S. patents and has co-authored a textbook on *Robust Adaptive Control*. Her current research interests include system and control theory and its applications to marine and automotive propulsion systems.

Dr. Sun was a recipient of the 2003 IEEE Control System Technology Award.

Kenneth Butts (M'10) received the B.E. degree in electrical engineering from the General Motors Institute (now Kettering University), Flint, MI, USA, the M.S. degree in electrical engineering from the University of Illinois at Urbana-Champaign, Urbana, IL, USA, and the Ph.D. degree in electrical engineering systems from the University of Michigan, Ann Arbor, MI, USA.

He is currently an Executive Engineer with the Powertrain, Chassis, and Regulatory Division, Toyota Motor Engineering and Manufacturing North America, Ann Arbor, MI, USA. In this position, he is investigating methods to improve engine control development productivity. He has been focusing on the field of automotive electronics and control since 1982, almost exclusively on research and advanced development of powertrain controls.