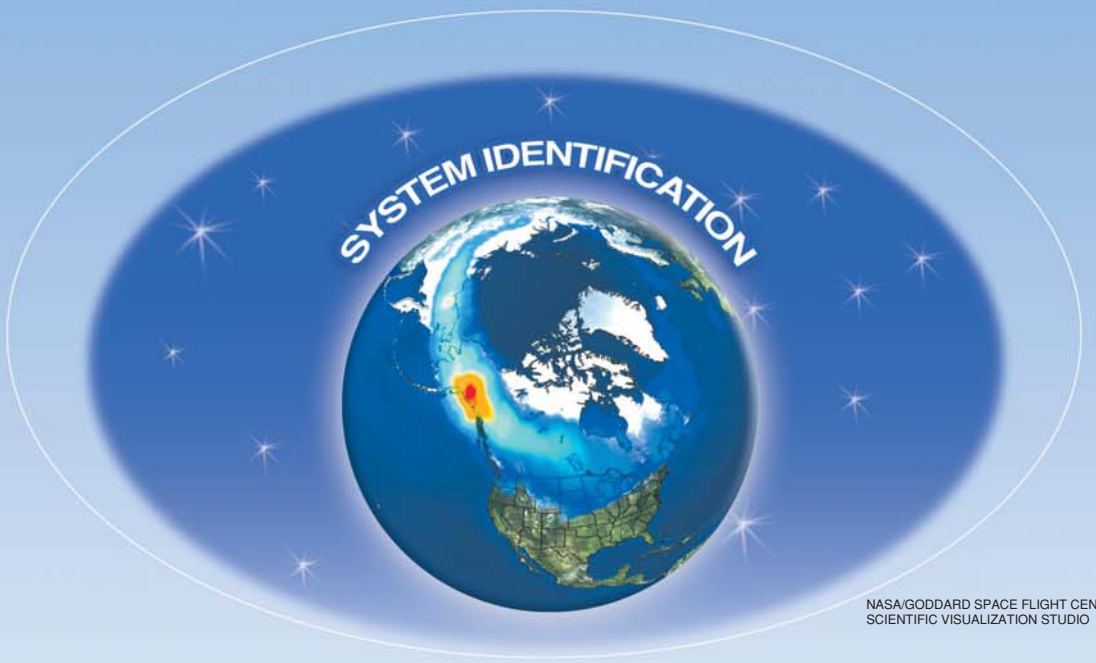

NASA/GODDARD SPACE FLIGHT CENTER
SCIENTIFIC VISUALIZATION STUDIO

Electric Load Forecasting

MARCELO ESPINOZA, JOHAN A.K. SUYKENS, RONNIE BELMANS, and BART DE MOOR

USING KERNEL-BASED MODELING FOR NONLINEAR SYSTEM IDENTIFICATION

Short-term load forecasting (STLF) concerns the prediction of power-system loads over an interval ranging from less than one hour to one week. Load forecasting has become a major field of research in electrical engineering. The power industry requires forecasts not only from the production side but also from a financial perspective. It is necessary to predict hourly loads as well as daily peak loads. Accurate tracking of the load by the system generation at all times is a basic requirement in the operation of power systems and must be accomplished for various time intervals. Since electricity cannot be stored efficiently in large quantities, the amount generated at any given time must cover all of the demand from consumers as well as grid losses.

Forecasts of the load are used to decide whether extra generation must be provided by increasing the output of online generators, by committing one or more extra units, or by the interchange of power with neighboring systems. Similarly, forecasts are used to decide whether the output of an already running generation unit should be decreased or switched off, which is determined by generation control functions, such as scheduling, unit-commitment, coordination, and interchange evaluation.

In addition, the liberalization of electric energy markets worldwide has led to the development of energy exchanges where consumers,

Digital Object Identifier 10.1109/MCS.2007.904656

1066-033X/07/$25.00©2007IEEE

FIGURE 1 Transformer in a substation. Substations are the last points in a transmission grid since they are the off-take points used by local distribution companies. The transformer converts the power from high voltage to the required level.

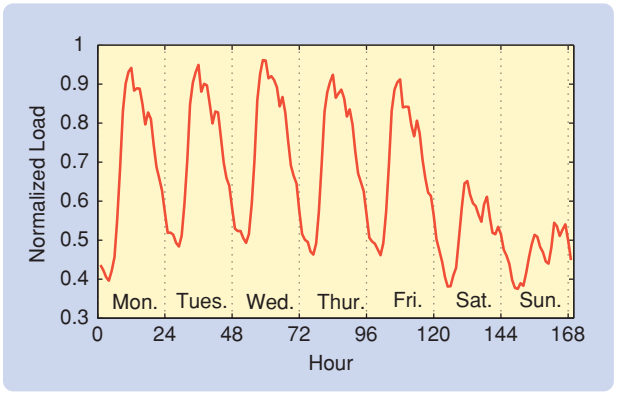

FIGURE 2 Example of a load series within a week. The daily pattern of consumption starts with low values of the load in the early morning, followed by the morning peak, and a decreasing consumption toward the end of the day. These daily patterns are also present during the weekends. The daily pattern is one of several seasonal patterns affecting the observed load.

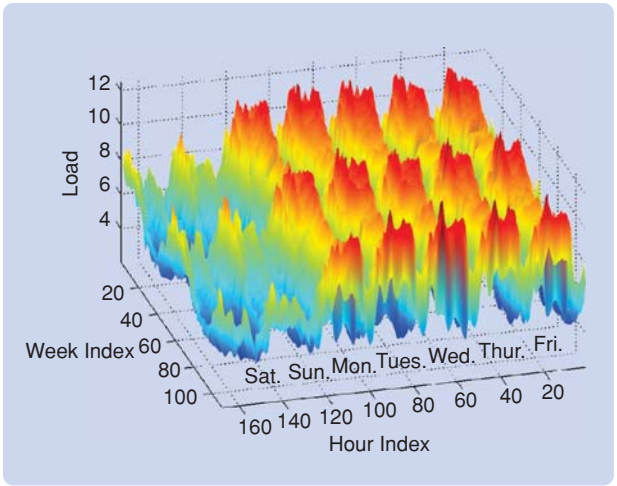

FIGURE 3 Seasonality of the load. Various seasonal profiles can be observed. The weekend is different from work days; every day has a peak in the morning and another in the evening.

generators, and traders can interact, leading to price settings, giving a new dimension to the problem of STLF [1]. Unexpected changes in the load can cause quick changes in energy price, while critical moments of high energy demand can lead, if not properly anticipated, to major problems both in availability and costs of energy.

Long time series, provided by the Belgian transmission system operator (TSO) ELIA, are used in this article to illustrate the use of nonlinear system identification techniques for STLF. The available time series contains hourly load values taken from multiple substations within the Belgian grid. Such substations correspond to the off-take points used by local distribution companies. The voltage is converted from high voltage, usually above 70 kV, to the required level on each substation (Figure 1). The observed load series can show various patterns because of the types of customers taking power from the substation. In this context, residential, business, and industrial customers are documented for some locations [2], [3] and can usually be recognized by their load pattern over a day.

For residential and commercial customers, load series show a strong seasonal behavior as well as dependence on local weather conditions. On the other hand, load series with an industrial profile are more irregular because the energy consumption is determined by operational decisions in a production or manufacturing facility. It is not unusual to have large industrial customers supplied by dedicated substations. To produce accurate forecasts for such industrial substations, it may be necessary to have information regarding operational decisions taken by plant managers. In practice, the load at a substation shows a mixture of these profiles, depending on the composition of customers downstream. In this article, load series with a high residential component are used. However, the exact composition of residential, commercial, and industrial customers is unknown.

For the type of load series under study, building a model for load forecasting must take into account seasonal patterns at multiple levels. A winter-summer pattern, weekly pattern, and intradaily pattern are shown in figures 2 and 3. These patterns also interact with external variables that affect the load, such as weather fluctuations. When the weather is cold, there is a requirement for heating, which translates into an increase in energy demand. Hot days in summer trigger the use of air conditioning equipment, also increasing the demand. On the other hand, the load on a Monday looks like the previous Monday, although a Monday in winter is different from a Monday in summer, as shown on Figure 4. The same observation can be made for weekends. However, special days, such as May 1, Easter, and Christmas, can show different behavior. All of these effects can combine with each other, and thus the influence of the weather on a winter Monday is different from the effect of the weather on a summer Friday. The effect of weather on the load is non-

linear, which is one of the main reasons for using nonlinear models for this problem. Particularly, the local temperature affects the load in a nonlinear manner, as shown on Figure 5. Actual weather observations are required to estimate a forecasting model. To produce load predictions, external weather forecasts [4] must be used instead of actual temperature values. For simulation purposes, the temperature values can be varied to assess the effect of temperature changes on the load.

System identification [5], [6] techniques for modeling and forecasting are used for STLF, where the main goal is to generate a model that captures the dynamics and interactions among possible explanatory variables for the load. For this purpose, a wide range of linear and nonlinear models have been developed incorporating the seasonal and cyclical properties of the load. The simplest approach is to assume deterministic seasonality, which can be represented by means of binary variables. More complex approaches include the assumption of stochastic seasonality, in the framework of Box-Jenkins seasonal ARIMA models [7], [8] in time-series analysis; the use of nonparametric models with seasonal components [9]; or the application of seasonally varying parameters in an autoregression [10], [11]. Additional examples related to traditional time-series analysis are considered in [12]–[14] while neural networks applications are given in [15]–[19]. The explanatory variables most widely used in the literature include past values of the load, weather information, calendar information, and error-correction terms [20], [21].

BUILDING A FORECASTING MODEL

We use a nonlinear autoregressive with exogenous inputs (NARX) model structure [6]. In this model structure, the load at a given hour is explained by the evolution of the load at previous hours as well as by the effect of exogenous variables that keep track of seasonal patterns and weather variations. To keep track of the day-to-day cycle, we use the binary-valued vector $\boldsymbol{W}_t \in \{0,1\}^7$, which is a vector of zeros with a 1 in the position of the day of the week of the load being observed at time t. For example, Monday corresponds to $\boldsymbol{W}_t = [1, 0, \cdots, 0]$. In the same way, the variable $\boldsymbol{M}_t \in \{0,1\}^{12}$ is defined as a vector of zeros with a 1 in the position of the month to which the load at time t belongs. A binary-valued vector $\boldsymbol{H}_t \in \{0,1\}^{24}$ is similarly defined to keep track of the hour of the day at which the load is observed. In addition, temperature variables are included to capture the effects of weather conditions. The hourly temperature variable T_t is the observed local temperature at hour t in Ukkel, Belgium, a reference meteorological station. From T_t, three variables are formed to capture the effect of cooling and heating requirements [22] on the load. The variable $\mathrm{CR}_t = \max(T_t - 20\,^\circ\mathrm{C}, 0)$ captures the cooling requirement when the ambient temperature rises above $20\,^\circ\mathrm{C}$. Heating and extra-heating variables are defined using $\mathrm{HR}_t = \max(16.5^\circ\mathrm{C} - T_t, 0)$ and $\mathrm{XHR}_t = \max(5.0^\circ\mathrm{C} - T_t, 0)$, respectively, where the temperature thresholds are based on standard techniques used in the energy industry. With these definitions, we formulate the vector $\boldsymbol{V}_t = [\mathrm{CR}_t, \mathrm{HR}_t, \mathrm{XHR}_t]$.

The NARX model formulation thus contains the following explanatory variables:

- an autoregressive part of 48 lagged load values (the previous two days)
- temperature-related variables measuring the effect of temperature on cooling and heating requirements (three variables).
- calendar information in the form of dummy variables for month of the year $\boldsymbol{M}_t$, day of the week $\boldsymbol{W}_t$, and hour of the day $\boldsymbol{H}_t$ (43 variables).

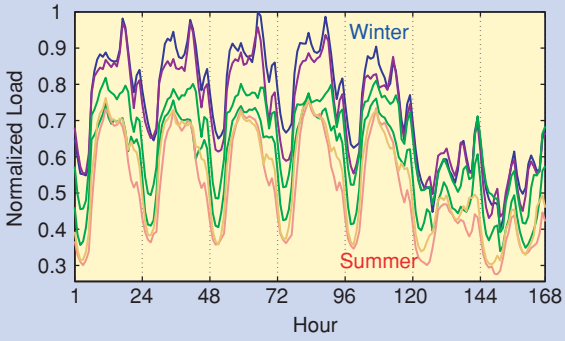

FIGURE 4 Comparison of a weekly profile over the year. The load in winter (blue) is different from the load in summer (red), both being different from profiles during spring or autumn (green). Notice the pronounced evening peaks that occur only in winter.

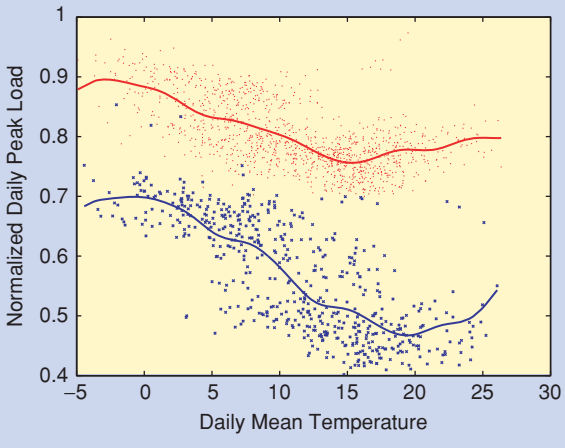

FIGURE 5 Nonlinear relation between temperature and load. One of the reasons for using nonlinear models is the relation between the ambient temperature and the observed load. Cold days trigger more energy consumption as do hot days. The daily peak load is plotted against the daily mean temperature for working days (red) and weekends (blue). The nonlinear relation is represented by a continuous line for each case, obtained with a nonlinear regression of load as a function of temperature. A forecasting model must be able to cope with this nonlinear effect.

This set gives $48 + 3 + 43 = 94$ explanatory variables to be included in the regression vector x_t of the NARX model

$$y_t = f(\boldsymbol{x}_t) + e_t, \tag{1}$$

where f is an unknown function, y_t denotes the load at hour t, e_t is assumed to be a white noise process, $\boldsymbol{x}_t \in \mathbb{R}^n$ is the regression vector

$$\boldsymbol{x}_t = [y_{t-1}, \ldots, y_{t-48}, \boldsymbol{V}_t, \boldsymbol{M}_t, \boldsymbol{W}_t, \boldsymbol{H}_t], \tag{2}$$

and the prediction at time t is given by $\hat{y}_t = f(\boldsymbol{x}_t) = y_t - e_t$.

In the context of linear system identification, the unknown function f in (1) is identified by parameterizing f as a function of a parameter vector $\boldsymbol{\theta} \in \mathbb{R}^n$, and then using a data sample $\mathcal{D} = \{\boldsymbol{x}_t, y_t\}_{t=1}^N$ to find an estimate $\hat{\boldsymbol{\theta}}$.

Model Structures and LS-SVM

Various structured elements can be incorporated into the model formulation, providing a practical way to include prior knowledge or existing information about the problem at hand.

Consider the regression vector $\boldsymbol{z}_t = [y_{t-1}; \ldots; y_{t-p}; \boldsymbol{u}_t; \boldsymbol{u}_{t-1}; \ldots; \boldsymbol{u}_{t-q}] \in \mathbb{R}^n$ containing p past values of the output $y_t \in \mathbb{R}$ and q past values of input vectors $\boldsymbol{u}_t \in \mathbb{R}^{N_u}$, and a nonlinear function $f : \mathbb{R}^n \to \mathbb{R}$ A NARX model structure [5], [6], [44] can be formulated as

$$y_t = f(\boldsymbol{z}_t) + e_t, \tag{S1}$$

where the error term e_t is assumed to be white noise.

The AR-NARX [5], [45], [24] model structure incorporates an autoregressive process for the error terms e_t given by

$$e_t = \sum_{i=1}^{q} \rho_i e_{t-i} + r_t, \tag{S2}$$

where the residuals e_t of (S1) follow an autoregressive process of order q given by (S2), and r_t is a white noise.

For each model structure, the function f can be parameterized in different ways. In the absence of prior information about its structure, the function f it can be parameterized using a black-box formulation in primal space based on LS-SVM, that is,

$$f(\boldsymbol{z}_t) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_t) + b, \tag{S3}$$

where $\boldsymbol{\varphi} : \mathbb{R}^n \to \mathbb{R}^{n_h}$ denotes the feature map. As explained in the article, the feature map is used in relation to a Mercer kernel in such a way that the feature map does not need to be computed explicitly. This parameterization can be used to estimate a NARX or AR-NARX structure.

On the other hand, the function f can be parameterized by using a partially linear (PL) structure. Some of the regressors in z_t can be included as linear terms, and others can be included under a nonlinear black-box term. Consider a partition of the regression vector $\boldsymbol{z}_t$ as follows. Consider the set $\mathcal{Z} = \{x : x$ is a component of the vector $\boldsymbol{z}_t\}$, and define an arbitrary partition $\mathcal{Z} = \mathcal{Z}_A \cup \mathcal{Z}_B$ with $\mathcal{Z}_A \cap \mathcal{Z}_B = \emptyset$. Define a vector $\boldsymbol{z}_{A,t} \in \mathbb{R}^{N_a}$ with regressors $\boldsymbol{x} \in \mathcal{Z}_A$, and a vector $\boldsymbol{z}_{B,t} \in \mathbb{R}^{N_b}$ with regressors $\boldsymbol{x} \in \mathcal{Z}_B$. The original regression vector is thus partitioned into $\boldsymbol{z}_t = [\boldsymbol{z}_{A,t}; \boldsymbol{z}_{B,t}]$. The subscripts A and B represent the subset of regressors entering linearly or nonlinearly into the model, respectively. The nonlinear component of this PL parameterization is expressed under a black-box formulation using LS-SVM. The nonlinear function f for a PL-NARX or a PL-AR-NARX is parameterized as

$$f(\boldsymbol{z}_t) = \beta^T \boldsymbol{z}_{A,t} + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_{B,t}) + b, \tag{S4}$$

for a given partition $\boldsymbol{z}_t = [\boldsymbol{z}_{A,t}; \boldsymbol{z}_{B,t}]$. The condition $\mathcal{Z}_A \cap \mathcal{Z}_B = \emptyset$ is imposed to ensure a unique representation of the parameter β [46].

The model formulations NARX, AR-NARX, PL-NARX, and PL-AR-NARX can be estimated using the framework of LS-SVM for regression [24], [47], [48]. An advantage is that the model can be solved and estimated for large-scale problems using the Nystrom method described in the article. A summary of the various model representations in primal and dual space is given in Table S1, where for simplicity the AR-NARX structure is described only for the AR(1)-NARX case.

TABLE S1 Summary of nonlinear model structures and representations using LS-SVM. A NARX model structure can be parameterized as a full black box or using a partially linear structure. The addition of an autoregressive process on the residuals leads to the AR(1)-NARX structure, which can also be parameterized by means of a partially linear structure. The resulting model structures NARX, AR(1)-NARX, PL-NARX, and PL-AR(1)-NARX can be written on primal and dual representations using LS-SVM.

NARX Model	
Primal	$\hat{y}_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_t) + b$
Dual	$\hat{y}_t = \sum_{i=1}^N \alpha_i K(\boldsymbol{z}_i, \boldsymbol{z}_t) + b$
AR(1)-NARX Model	
Primal	$\hat{y}_t = \rho y_{t-\tau} + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_t) - \rho \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_{t-\tau}) + (1-\rho) b$
Dual	$\hat{y}_t = \rho y_{t-\tau} + h(\boldsymbol{z}_t) - \rho h(\boldsymbol{z}_{t-\tau})$ with $h(\boldsymbol{z}_t) = \sum_{i=\tau+1}^N \alpha_{i-\tau} [K(\boldsymbol{z}_i, \boldsymbol{z}_t) - \rho K(\boldsymbol{z}_{i-\tau}, \boldsymbol{z}_t)] + b$
PL-NARX Mode	
Primal	$\hat{y}_t = \beta^T \boldsymbol{z}_{A,t} + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_{B,t}) + b$
Dual	$\hat{y}_t = \beta^T \boldsymbol{z}_{A,t} + \sum_{i=1}^N \alpha_i K(\boldsymbol{z}_{B,i}, \boldsymbol{z}_{B,t})$
PL-AR(1)-NARX Model	
Primal	$\hat{y}_t = \rho y_{t-\tau} + \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_{B,t}) - \rho \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{z}_{B,t-\tau}) + b(1-\rho) + \beta^T (\boldsymbol{z}_{A,t} - \rho \boldsymbol{z}_{A,t-\tau})$
Dual	$\hat{y}_t = \rho y_{t-\tau} + h(\boldsymbol{z}_{B,t}) - \rho h(\boldsymbol{z}_{B,t-\tau}) + \beta^T (\boldsymbol{z}_{A,i} - \rho \boldsymbol{z}_{A,i-\tau})$ with $h(\boldsymbol{z}_{B,t}) = \sum_{i=\tau+1}^N \alpha_{i-\tau} \left[K(\boldsymbol{z}_{B,i}, \boldsymbol{z}_{B,t}) - \rho K(\boldsymbol{z}_{B,i-\tau}, \boldsymbol{z}_{B,t})\right] + b$

The estimate $\hat{\boldsymbol{\theta}}$ is typically obtained as the solution of an optimization problem [5]. When f is parameterized as a linear function of the form

$$f(\boldsymbol{x}_t) = \boldsymbol{\theta}^T \boldsymbol{x}_t + b, \tag{3}$$

then [1] is an ARX model, and the estimates $\hat{\boldsymbol{\theta}}$, $\hat{b}$ are usually obtained as the solution of a least-squares optimization problem. Linear models assume a linear effect from the input variable $\boldsymbol{x}_t$ to the output y_t.

Nonlinear effects from $\boldsymbol{x}_t$ to y_t can be identified when the function f is parameterized as a nonlinear function. In this article, we illustrate the use of least squares support vector machines (LS-SVM) [23] as a tool for estimating the nonlinear function f in the NARX model (1). This model structure can be further extended to the case where the residuals e_t follow an autoregressive process of order q, leading to an AR-NARX model structure [24] as shown in "Model Structures and LS-SVM."

MODEL REPRESENTATION AND ESTIMATION

Least-squares support vector machines belong to the class of kernel methods that use positive-definite kernel functions to build a nonlinear representation of the original inputs in a high-dimensional feature space. We start by parameterizing the function f in (1) as

$$f(\boldsymbol{x}_t) = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b \tag{4}$$

where $x_t \in \mathbb{R}^n$ is the regression vector, $b \in \mathbb{R}$ is a bias term, $\boldsymbol{w} \in \mathbb{R}^{n_h}$ is an unknown coefficient vector, and $\boldsymbol{\varphi} : \mathbb{R}^n \to \mathbb{R}^{n_h}$ is a nonlinear feature map, which transforms the original input $\boldsymbol{x}_t \in \mathbb{R}^n$ to a high-dimensional vector $\boldsymbol{\varphi}(\boldsymbol{x}_t) \in \mathbb{R}^{n_h}$, which can be infinite dimensional. If the feature map $\boldsymbol{\varphi}$ is known in advance, then the model becomes a linear problem because the unknown $\boldsymbol{w}$ can be estimated by using linear regression techniques. For example, consider the regression

$$y_i = w x_i + e_i, \tag{5}$$

to be estimated with a data set $\{x_i, y_i\}_{i=1}^N$, with $w, y_i, x_i \in \mathbb{R}$, where e_i is assumed to be white noise. The linear regression (5) can be solved using least squares to obtain an estimate $\hat{w}$ of w. If the system from which the data are collected is linear, the regression (5) provides a good approximation of the system behavior. However, if the true system follows the process

$$y_i = w_1 x_i^2 + w_2 \sqrt{2} x_i + w_3 + e_i, \tag{6}$$

the regression (5) is not correctly specified because it does not contain the nonlinear effect x^2. To obtain the correct specification, the original input x can be mapped to a higher dimensional space by means of the feature map $\boldsymbol{\varphi} : \mathbb{R} \to \mathbb{R}^3$ defined by

$$\boldsymbol{\varphi}(x) = [x^2, \sqrt{2}x, 1]. \tag{7}$$

Solving the regression

$$y_i = \boldsymbol{w}^T \boldsymbol{\varphi}(x_i) + e_i \tag{8}$$

yields an estimate of $\boldsymbol{w} = [w_1, w_2, w_3]$.

In this example, the feature map $\boldsymbol{\varphi}$ is assumed to be known, and thus the coordinates in the high-dimensional space can be computed directly to arrive at the correct regression specification. However, in the context of LS-SVM, the feature map $\boldsymbol{\varphi}$ does not have to be known explicitly but rather is implicitly defined by using a kernel function. This fact is a key element of support vector machines.

POSITIVE-DEFINITE KERNEL AND FEATURE MAP

A kernel is a function from $X \times X$ to $\mathbb{R}$, where usually $X \subseteq \mathbb{R}^n$. A commonly used kernel function is the Gaussian radial basis function (RBF) kernel, shown in Figure 6, given by

$$K(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{||\boldsymbol{x} - \boldsymbol{z}||_2^2}{\sigma^2}\right), \tag{9}$$

where σ is a tuning parameter.

Kernel functions are used extensively. In statistics, kernel functions are used for nonparametric regression [25], [26] as well as inverse problems [27]. In probability theory, kernels arise as covariances of stochastic processes [28]. Since the 1990s, kernel functions have been used in the context of machine learning and pattern recognition following the developments in statistical learning theory and support vector machines [29].

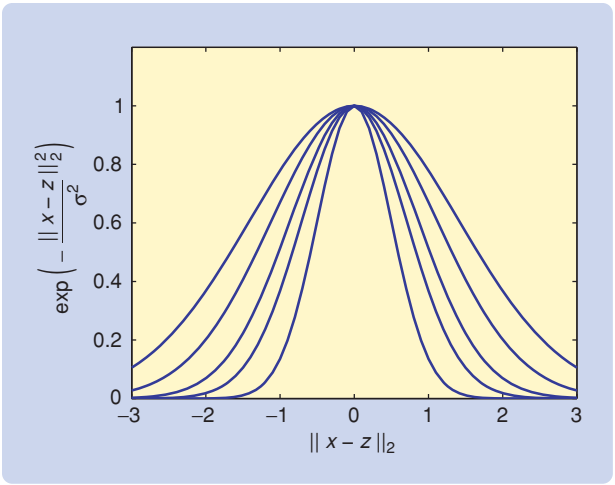

FIGURE 6 Example of a kernel function. The Gaussian radial basis function kernel is shown here for several values of the parameter σ. The Gaussian kernel function is frequently used in the context of kernel methods because of its flexibility. A model using Gaussian kernels can approximate a large class of smooth functions.

The relation between the feature map $\boldsymbol{\varphi}$ and kernel functions was provided by James Mercer in 1909 [30] working in the field of integral equations. Mercer's theorem shows that, for every positive-definite kernel function K, there exists a mapping $\boldsymbol{\varphi} : \mathbb{R}^n \to \mathbb{R}^{n_h}$ from the input space to the feature space such that

$$K(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{\varphi}(\boldsymbol{x})^T \boldsymbol{\varphi}(\boldsymbol{z}) = \sum_{i=1}^{n_h} \varphi_i(\boldsymbol{x}) \varphi_i(\boldsymbol{z}). \tag{10}$$

Mercer's theorem can be illustrated explicitly when a feature map $\boldsymbol{\varphi}$ is known, by showing that a dot product of two vectors in feature space gives the same result as a kernel function evaluated on the points in input space. For example, consider the feature map defined in (7). The dot product of two vectors in feature space is given by

$$\begin{aligned} \boldsymbol{\varphi}(x_1)^T \boldsymbol{\varphi}(x_2) &= \left[x_1^2, \sqrt{2}x_1, 1\right]^T \left[x_2^2, \sqrt{2}x_2, 1\right] \\ &= x_1^2 x_2^2 + 2x_1 x_2 + 1 \\ &= (x_1 x_2 + 1)^2, \end{aligned}$$

which is equivalent to the polynomial kernel of degree $d = 2$ given by $K(x_1, x_2) = (1 + x_1 x_2)^2$ evaluated at the points x_1, x_2.

However, when the dimension of the input vector n or the degree of the polynomial kernel d increases, or when using a different kernel function, the situation may become more complicated. In general, for the polynomial kernel of degree d given by $K_{\text{pol}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^T \boldsymbol{x}_j + c)^d$ [31], where $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^n$, the induced feature map is represented by all possible product monomials between the components of $\boldsymbol{x}_i$ up to degree d, leading to a feature map of dimension $n_h = \binom{n+d}{d}$. When $n = 1$ and $d = 2$, the dimension of the feature map is $n_h = 3$, as explicitly described in the above example. For $d = 4$ and $n = 94$, using the regression vector $\boldsymbol{x}$ defined in (2), the dimension of the feature map is $n_h = 3,612,280$. Such a large number of components of $\boldsymbol{\varphi}$ requires the estimation of $\boldsymbol{w}$ of the same size, which is impractical if attempted directly. Increasing the degree of the polynomial kernel to $d = 7$, leads to a feature map of dimension $n_h = 1.72 \times 10^{10}$, which is an intractable number of parameters to be estimated direct-

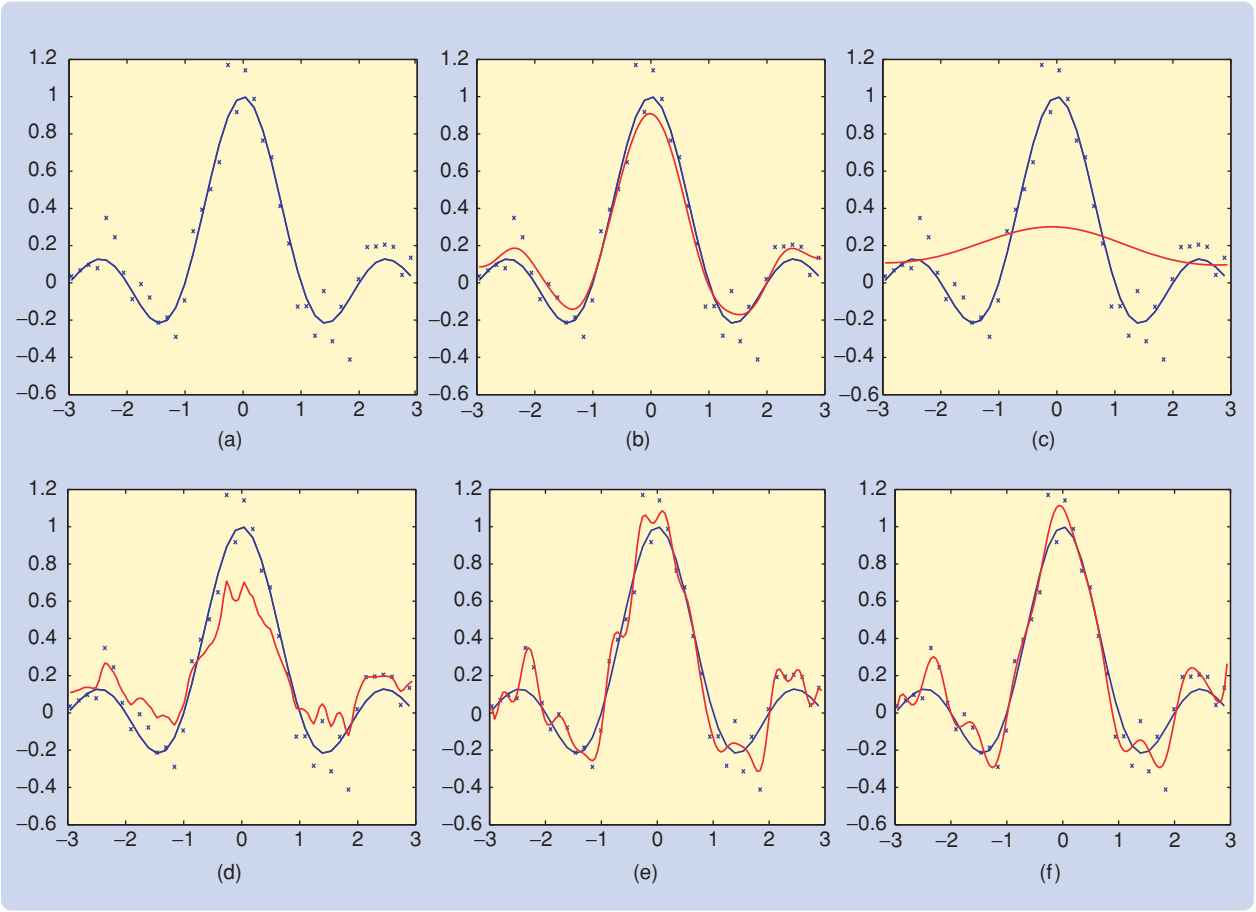

FIGURE 7 Effect of using various kernel parameters in the same LS-SVM regression. (a) The original function (blue) is approximated with a nonlinear regression with LS-SVM estimated on the available noisy data points. (b)–(f) Each of the five approximations (red) is obtained with a different value of σ for the Gaussian radial basis function kernel parameter. The parameter σ, as well as the regularization constant γ, must be determined by model selection procedures.

ly. For the Gaussian RBF kernel (9) the associated feature map $\boldsymbol{\varphi}$ is infinite dimensional [32]. Mercer's theorem guarantees that a feature map exists for every kernel function and provides a tool for expressing f in terms of the kernel function, as shown in the following section. In this way, kernel methods can work in the feature space without requiring explicit knowledge or computation of the feature map $\boldsymbol{\varphi}$, working directly and exclusively with the kernel function.

To apply (10), any positive-definite kernel function can be chosen. For particular applications, as in text mining and bioinformatics, ad hoc kernel functions can be designed. In practice, the Gaussian kernel function is frequently used since they can approximate a large class of smooth functions [33], [34].

PRIMAL AND DUAL MODEL REPRESENTATIONS

LS-SVM regression estimation involves primal and dual model representations. Given the training data set $\mathcal{D} = \{\boldsymbol{x}_t, y_t\}_{t=1}^N$ the goal is to estimate the model (1), where f is parameterized as in (4). The parameterization leads to the model

$$y_t = \boldsymbol{w}^T \boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \tag{11}$$

where $\boldsymbol{x}_t \in \mathbb{R}^n$, $y_t \in \mathbb{R}$, $\boldsymbol{\varphi} : \mathbb{R}^n \to \mathbb{R}^{n_h}$ is the unknown feature map, and e_t is a white noise process.

TABLE 1 Effect of the number of support vectors on model performance. The fixed-size LS-SVM requires the definition of a subset of size M to build a finite-dimensional approximation of the feature map. The use of a larger M decreases the cross validation mean-squared error of the model. While increasing from $M = 200$ to $M = 400$ provides substantial improvement, the benefit of moving from $M = 800$ to $M = 1000$ is much smaller.

Subset Size	Cross Validation MSE
M = 200	0.032
M = 400	0.022
M = 600	0.017
M = 800	0.016
M = 1000	0.015

Consider the LS-SVM constrained optimization problem (LCOP) of minimizing, with respect to W, b, and e_t, the objective function

$$J(\boldsymbol{w}, b, e_t) = \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \gamma\frac{1}{2}\sum_{t=1}^{N} e_t^2 \tag{12}$$

subject to

$$y_t = \boldsymbol{w}^T\boldsymbol{\varphi}(\boldsymbol{x}_t) + b + e_t, \quad t = 1, \ldots, N, \tag{13}$$

where the regularization constant $\gamma > 0$ is included to control the bias-variance tradeoff, to ease the effects of ill-

TABLE 2 Model performance on the test set for different forecasting modes. Linear ARX models are estimated using the entire training sample (N = 36,000) The NARX models are estimated with LS-SVM using only the last 1000 observations (N = 1,000), and with fixed-size LS-SVM (FS-LSSVM) on the entire sample. The performance is assessed using the mean-squared error (MSE) and mean absolute percentage error (MAPE) on a test set of 15 days out-of-sample. The FS-SLSSVM models for Series 1 and Series 5 are further re-estimated using an AR-NARX formulation to correct for detected autocorrelation, giving marginally better results.

Load Series	Forecasting Mode	Performance Evaluation	ARX Linear	NARX LS-SVM	NARX FS-LSSVM	AR-NARX FS-LSSVM
Series 1	1-h-ahead	MSE	1.4%	2.2%	0.6%	0.5%
		MAPE	2.5%	2.8%	1.5%	1.4%
	24-h-ahead	MSE	9.5%	5.0%	2.7%	2.7%
		MAPE	5.9%	4.3%	3.1%	3.1%
Series 2	1-h-ahead	MSE	3.0%	3.4%	2.3%	–
		MAPE	3.9%	4.3%	3.4%	–
	24-h-ahead	MSE	11.9%	20.2%	11.5%	–
		MAPE	7.9%	10.6%	7.4%	–
Series 3	1-h-ahead	MSE	10.2%	9.7%	6.7%	–
		MAPE	24.9%	29.4%	17.7%	–
	24-h-ahead	MSE	15.0%	15.1%	9.4%	–
		MAPE	29.7%	30.1%	23.1%	–
Series 4	1-h-ahead	MSE	7.4%	4.9%	4.0%	–
		MAPE	16.2%	12.6%	10.5%	–
	24-h-ahead	MSE	14.7%	10.1%	6.0%	–
		MAPE	22.3%	20.7%	14.5%	–
Series 5	1-h-ahead	MSE	1.7%	2.2%	0.9%	0.7%
		MAPE	2.2%	2.6%	1.7%	1.6%
	24-h-ahead	MSE	6.7%	9.0%	3.8%	3.6%
		MAPE	4.4%	5.5%	3.4%	3.2%

conditioning, and to provide an expression for w in terms of the dual variables.

The following development shows the surprising fact that f can be expressed in terms of a positive-definite kernel function K without having to compute, or even know, the feature map φ. The derivation is based on solving the constrained optimization problem by introducing Lagrange multipliers and using Mercer's theorem. Consider the Lagrangian of LCOP given by

$$\mathcal{L}(w, b, e_t; \alpha) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{t=1}^{N} e_t^2 - \sum_{t=1}^{N} \alpha_t (w^T \varphi(x_t) + b + e_t - y_t), \quad (14)$$

where $\alpha_t \in \mathbb{R}$ are Lagrange multipliers. It follows from the conditions for optimality $\partial \mathcal{L}/\partial w = 0$, $\partial \mathcal{L}/\partial b = 0$, $\partial \mathcal{L}/\partial e_t = 0$, and $\partial \mathcal{L}/\partial \alpha_t = 0$, that

$$w = \sum_{i=1}^{N} \alpha_i \varphi(x_i), \quad (15)$$

$$0 = \sum_{t=1}^{N} \alpha_t, \quad (16)$$

$$\alpha_t = \gamma e_t, \qquad t = 1, \ldots, N, \quad (17)$$

$$y_t = w^T \varphi(x_t) + b + e_t, \quad t = 1, \ldots, N. \quad (18)$$

Replacing the expression for w from (15) into (18) yields

$$y_t = \sum_{i=1}^{N} \alpha_i \varphi(x_i)^T \varphi(x_t) + b + e_t. \quad (19)$$

Now, using Mercer's theorem to replace the dot product $\varphi(x_i)^T \varphi(x_t)$ yields

$$y_t = \sum_{i=1}^{N} \alpha_i K(x_t, x_i) + b + e_t, \quad (20)$$

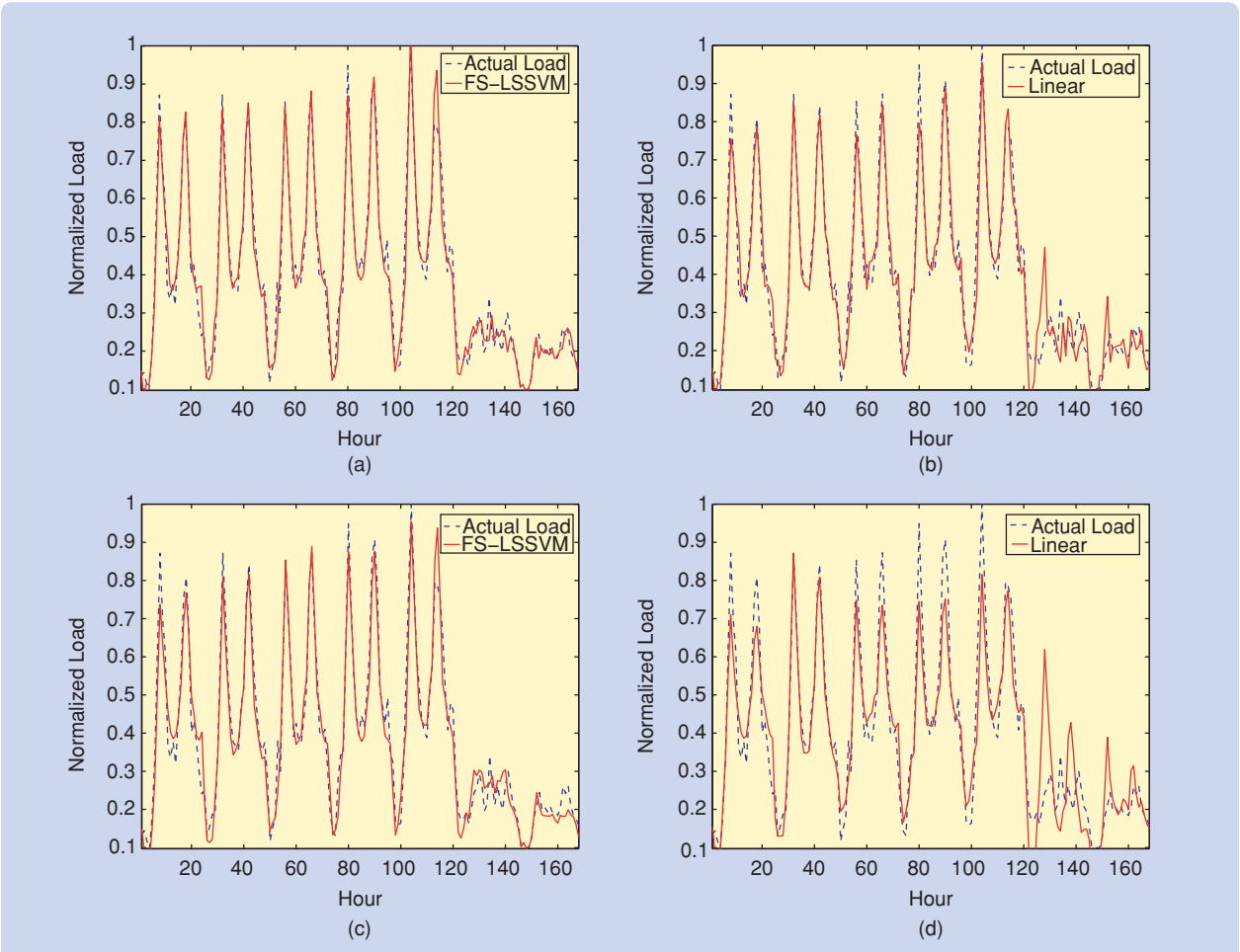

FIGURE 8 Forecasting comparison. The 1-h-ahead predictions obtained with (a) FS-LSSVM and (b) the linear model are compared with the actual load for a test period of seven days. The same comparison can be observed for the 24-h-ahead simulations obtained with (c) FS-LSSVM and (d) the linear model.

for a chosen positive-definite kernel function K. Replacing $e_t = \alpha_t/\gamma$ from (17) into (20) gives

$$y_t = \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_t, \boldsymbol{x}_i) + b + \frac{\alpha_t}{\gamma}, \qquad t = 1, \ldots, N, \quad (21)$$

$$0 = \sum_{t=1}^{N} \alpha_t, \quad (22)$$

with unknowns α_t, $t = 1, \ldots, N$, and b. Building the kernel matrix $\Omega_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and writing (21) and (22) in matrix notation gives the linear system

$$\begin{bmatrix} \Omega + \frac{1}{\gamma}\mathbf{I} & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ b \end{bmatrix} = \begin{bmatrix} \boldsymbol{y} \\ 0 \end{bmatrix}, \quad (23)$$

where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]^T$ are the dual variables and $\boldsymbol{y} = [y_1, \ldots, y_N]^T$. Solving for $\boldsymbol{\alpha}$ and b yields the estimate for f expressed in dual form

$$f(\boldsymbol{x}_t) = \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x}_t) + b. \quad (24)$$

Estimation in Primal Space

As shown above, for each positive-definite kernel, the feature map does not need to be known. Mercer's theorem allows us to solve the model in dual variables, where the final expression (24) is obtained in terms of kernel function evaluations. This primal-dual formulation can be exploited further. Consider the fact that the size of the system (23) is given by the number of training data points N but not by the dimension of the input vector $\boldsymbol{x}$. Although this fact provides a practical advantage for working with small samples of high-dimensional inputs, solving the system (23) is too time consuming and possibly not feasible when a large number of data points is available, as in the case of STLF. In this case, we seek an explicit expression for a finite-dimensional approximation $\tilde{\boldsymbol{\varphi}}$ of the feature map $\boldsymbol{\varphi}$ induced by the positive-definite kernel K.

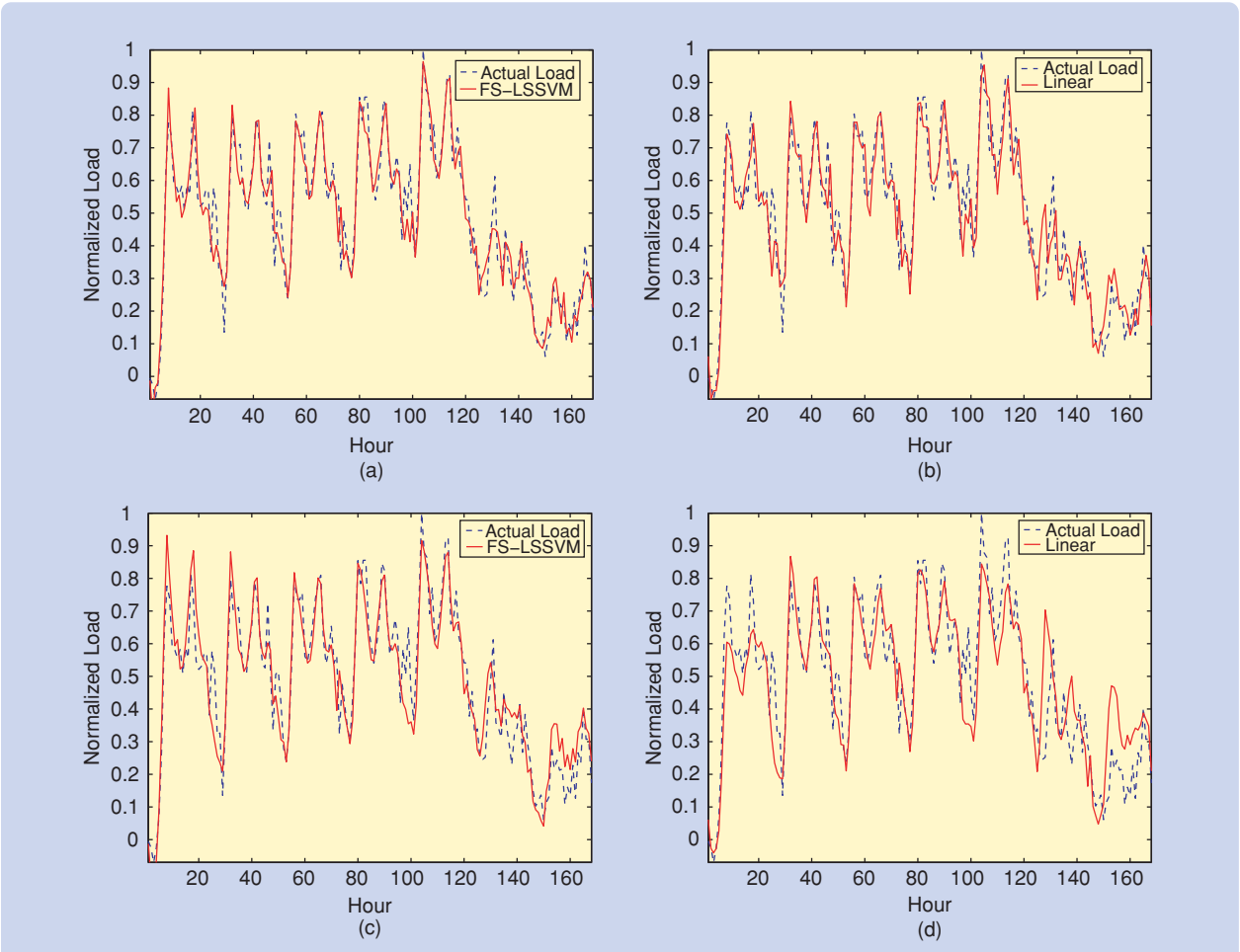

FIGURE 9 Forecasting comparisons for another substation. The 1-h-ahead predictions obtained with (a) FS-LSSVM and (b) the linear model are compared with the actual load for a test period of seven days. The same comparison can be observed for the 24-h-ahead simulations obtained with (c) FS-LSSVM and (d) the linear model.

We use hourly time-series data recorded over approximately four years, leading to a sample of 36,000 data points for model estimation. Solving the system (23) with this data set requires the inversion of a matrix of size 36,000 × 36,000, which is impractical. Under these circumstances we find $\tilde{\boldsymbol{\varphi}}$ by using an $M \times M$ kernel matrix Ω_M evaluated on a subset of M points, where $M \ll N$. The matrix Ω_M provides the starting point for computing $\tilde{\boldsymbol{\varphi}}$ for all N points in the complete sample. Given a subset $\mathcal{S}_M = \{\boldsymbol{x}_i, y_i\}_{i=1}^M$ of the original data set $\mathcal{D}$, where $M \ll N$, it is possible to compute the eigendecomposition of the kernel matrix Ω_M and use its eigenvalues λ_i and eigenvectors u_i, $i = 1, \ldots, M$, to compute the ith required component of an arbitrary point $\tilde{\boldsymbol{\varphi}}(\boldsymbol{x})$ (particularly for a point not included in the original small sample) by means of

$$\tilde{\varphi}_i(\boldsymbol{x}) = \frac{M}{\sqrt{\lambda_i}} \sum_{k=1}^{M} u_{ik} K(\boldsymbol{x}_k, \boldsymbol{x}), \quad i = 1, \ldots, M, \tag{25}$$

where u_{ik} is the kth component of the eigenvector u_i. This expression, known as the Nyström method [35], [36], leads to the $M-$dimensional approximation

$$\tilde{\boldsymbol{\varphi}}(\boldsymbol{x}) = [\tilde{\varphi}_1(\boldsymbol{x}), \tilde{\varphi}_2(\boldsymbol{x}), \ldots, \tilde{\varphi}_M(\boldsymbol{x})]^T. \tag{26}$$

The approximation $\tilde{\boldsymbol{\varphi}}$ provides an explicit finite-dimensional representation of the feature map. Therefore, LCOP can be solved directly in primal space to estimate $\boldsymbol{w}$ and b, leading to a sparse representation of the model [23]. The subset of size M used to evaluate the kernel matrix Ω_M must be chosen beforehand by the user, leading to the fixed-size LS-SVM method [23]. The subset of size M can be selected either randomly or by using an active selection procedure that focuses on the more important regions of the data set [37]. One such method is the maximization of the quadratic Renyi entropy [23], [35], which provides a link between a kernel matrix and the density of the underlying data.

The basic steps for the fixed-size LS-SVM method can be summarized as follows:

1) Select a subset $\mathcal{S}_M$ of $M \ll N$ samples from the training set $\mathcal{D} = \{\boldsymbol{x}_i, y_i\}_{i=1}^N$.
2) Build a kernel matrix Ω_M evaluated on $\mathcal{S}_M$ with a given kernel parameter.
3) Compute the eigendecomposition of Ω_M.
4) Use (25) to compute each component of the finite-dimensional approximation $\tilde{\boldsymbol{\varphi}}$ in (26).
5) Choose $\gamma > 0$ and solve LCOP in primal space using $\tilde{\boldsymbol{\varphi}}$.

The advantage of this estimation technique is that it provides a sparse representation of the nonlinear regression problem, where the dimension of the finite-dimensional approximation of the feature map is given by M. By choosing $M \ll N$, the sparseness is improved with respect to the case of solving (23) directly, where the solution is expressed in terms of the N values of α_i.

Tuning Parameter Selection

The system (23) gives the solution of the LS-SVM regression for a chosen kernel function K and a chosen regularization constant γ. Usually training of the LS-SVM model involves the optimal selection of kernel parameters (σ for the RBF kernel; c and d for a polynomial kernel) and the regularization constant γ, which are tuning parameters. The tuning parameters are selected in such a way that the model shows good generalization ability, maximizing the performance of the model over data not used for model estimation (out of sample or fresh data). Figure 7 shows an example of function approximation using LS-SVM with various kernel parameters. The original function is approximated with LS-SVM estimated on the available (noisy) points. By varying the kernel parameters, the quality of the approximation changes drastically.

The selection of the tuning parameters is usually done by cross-validation techniques, where parameters are chosen based on their performance on the data, or Bayesian inference, which involves the assumption of a probability density function for the parameters [38]–[41]. Cross validation, which provides a good assessment of the generalization ability of a model and is used extensively in machine learning and statistics [42], works by dividing the sample into m parts. The model is estimated using $m-1$ parts, and the predictive performance is assessed on the remaining part by using the mean-squared error (MSE). The process is repeated m times (each time leaving out one of the m parts) and the performance measurements are averaged, thus minimizing any data selection bias. The sequence of steps for performing cross validation is given as follows:

1) Partition the sample $\mathcal{D} = \{\boldsymbol{x}_i, y_i\}_{i=1}^N$ into m parts $\mathcal{D}_1, \ldots, \mathcal{D}_m$, such that $\mathcal{D} = \bigcup_{k=1}^m \mathcal{D}_k$.
2) Define a grid of possible values for σ and γ.
3) For every combination of σ and γ, repeat the following cycle:
4) **for** $k = 1$ to m.
5) Leave out the data from $\mathcal{D}_k$.
6) Define a training set $\mathcal{T}_k$ consisting of the remaining $m-1$ parts, $\mathcal{T}_k = \bigcup_{i=1, i\neq k}^m \mathcal{D}_i$.
7) Estimate a model using LS-SVM or fixed-size LS-SVM.
8) Evaluate the one-step-ahead MSE of the estimated model on the data from $\mathcal{D}_k$ that was left untouched. Call this MSE_k.
9) **end for loop**.
10) Average the obtained MSE_k over the m repetitions. This value is the cross validation MSE for a given σ and γ.
11) Find the combination of σ and γ with the lowest cross validation MSE.

IMPLEMENTATION AND RESULTS

We consider five load series from the Belgian grid, where each series shows a high component of residential customers. Each load series is modeled independently. A training set of 36,000 hourly observations is used to estimate each model. The assessment of the models is made on data that are not used for model estimation. This test set (or out-of-sample data) consists of the block of 15 days after the last training point. Given the large size of the training set, a fixed-size LS-SVM is used to estimate the model in primal space by building the approximation $\tilde{\boldsymbol{\varphi}}$ of the feature map $\boldsymbol{\varphi}$. The number of support vectors M must be chosen by the user. By means of the Nyström method (25), the approximation $\tilde{\boldsymbol{\varphi}}$ has dimension M, and the estimation of a fixed-size LS-SVM in primal space translates into a parametric estimation problem of finding an estimate $\hat{w}$ of dimension M. In principle, any M can be selected given the computational resources at hand. For our purposes, we test the methodology for $M = 200, 400, 600, 800$, and 1000 support vectors. For each value of M, the support vectors are selected using the quadratic Renyi entropy criterion [23], [43]. Between 0.5% and 3% of the available training set is used to build a sparse feature map approximation $\tilde{\boldsymbol{\varphi}}$ for the entire

Model Validation

One of the fundamental questions in system identification is to verify whether the selected model is a valid representation of the underlying system. An assessment of the quality of the model can be obtained by checking the properties of the residuals of the regression. If the regression is correctly specified, all correlations, in time and across input variables, are captured by the model. Therefore, the residuals of a correctly specified model are expected to show a small degree of autocorrelation, as well as little correlation with any of the input series. Significant autocorrelation suggests the presence of unmodeled dynamics, and the model structure can be modified to incorporate them.

Figure S1 shows an histogram of the residuals from the FS-LSSVM model for Series 4 and the empirical density function estimated using a Parzen window method. There are no outliers, and the residuals are centered around zero. Figure 11 shows the autocorrelation plot of the residual series from the FS-LSSVM models for Series 1 and 5. Statistically significant autocorrelation levels appear at particular lags considering a 95% confidence.

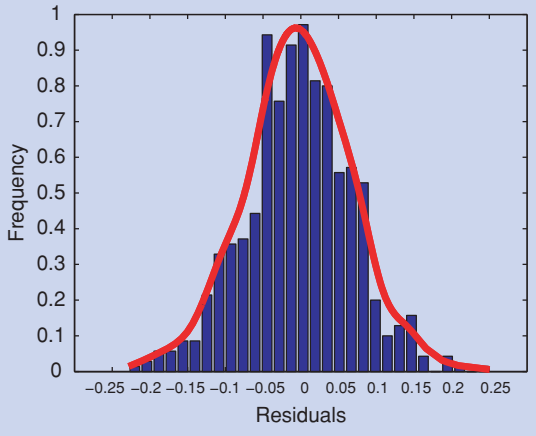

FIGURE S1 Empirical distribution of the residuals. The histogram (blue) and the empirical density function (red) for the residual on the training set of the NARX model estimated with Series 4. The residuals are symmetrically centered around 0, without outliers.

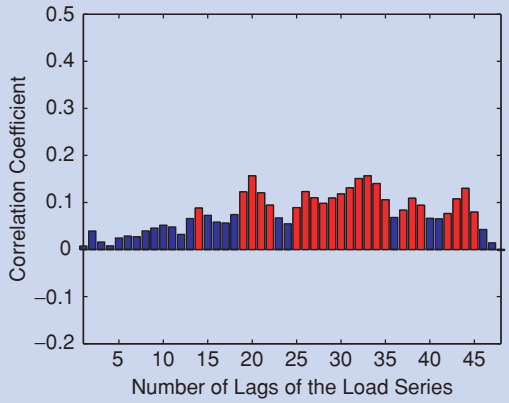

FIGURE S2 Correlation between the residuals and the regressors in the NARX model for Series 2. The bars show the correlation coefficient between the residual series and each of the delayed load series, which are lagged from 1 to 48 h. The bars in red represent those correlations that are statistically significant at a 95% level. The maximum significant correlation is 0.15 for the load series delayed 20 h. Although the correlation is statistically significant, the coefficients have small magnitude.

To examine the correlation between the residuals and the regressors in the model, consider regressors given by the past values of the load. Figure S2 shows the correlation between the residuals and each of the regressors. The blue bars show the magnitude of the correlation coefficient between the residuals and each of the 48 regressors. The red bars highlight those correlation coefficients that are statistically significant. The maximum correlation coefficient is 0.15, which denotes low correlation. The existence of autocorrelation in the residuals provides insight for further refinement of the model.

sample. Although increasing from $M = 200$ to $M = 400$ improves the performance significantly, the benefit of moving from $M = 800$ to $M = 1000$ is much smaller, as shown in Table 1. Taking a Gaussian RBF kernel, the tuning of the kernel parameter σ and the regularization constant γ is performed by ten-fold cross validation in the training sample.

Three models are estimated for each load series, the fixed-size LS-SVM (FS-LSSVM) estimated using the entire sample, the standard LS-SVM in dual version estimated with the last 1000 data points of the training sample, and a linear ARX model (3) estimated with the same regression vector x_t as the FS-LSSVM. In this way, it is possible to compare the difference in performance between two nonlinear models in the following two cases: when the full sample is taken into account (fixed-size LS-SVM) or only when the last 1000 h (last 42 days) are considered, as well as the performance of a traditional linear model. The models are compared based on their performance on the data not used during training.

The forecasting performance is assessed as follows. The simplest scheme is to forecast the first out-of-sample load value using all information available, then wait 1 h until the true value of this forecast is observed, and then forecast the next value using all available information (1-h-ahead prediction). However, planning engineers require forecasts with a longer time horizon, at least a full day. In this case, it is necessary to predict the first out-of-sample value using the full training sample, then predict the second value out-of-sample using this first prediction, and continue by iterative simulation. In practice, it is reasonable to stop this iterative process after 24 h and update the

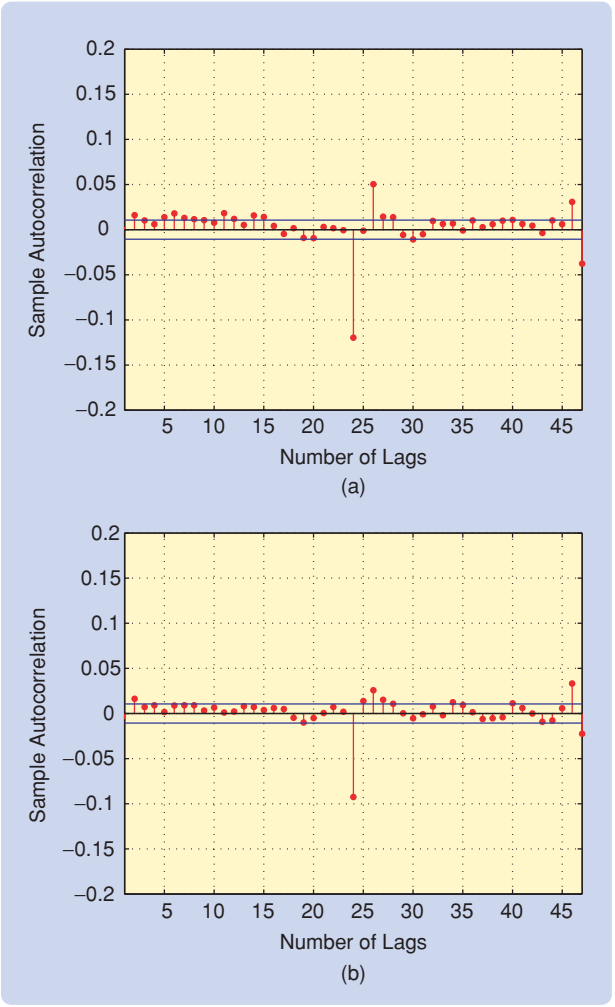

FIGURE 10 Autocorrelation of the NARX residuals in the training set. The horizontal bands provide the reference for statistical significance at a 95% level. There is significant autocorrelation on the residuals of the NARX models for (a) Series 1 at lags 6, 11, 24, 26, 46, and 47 and (b) for Series 5 at lags 24, 25, 26, and 46. Autocorrelation in the residuals of a model suggests unmodeled dynamics.

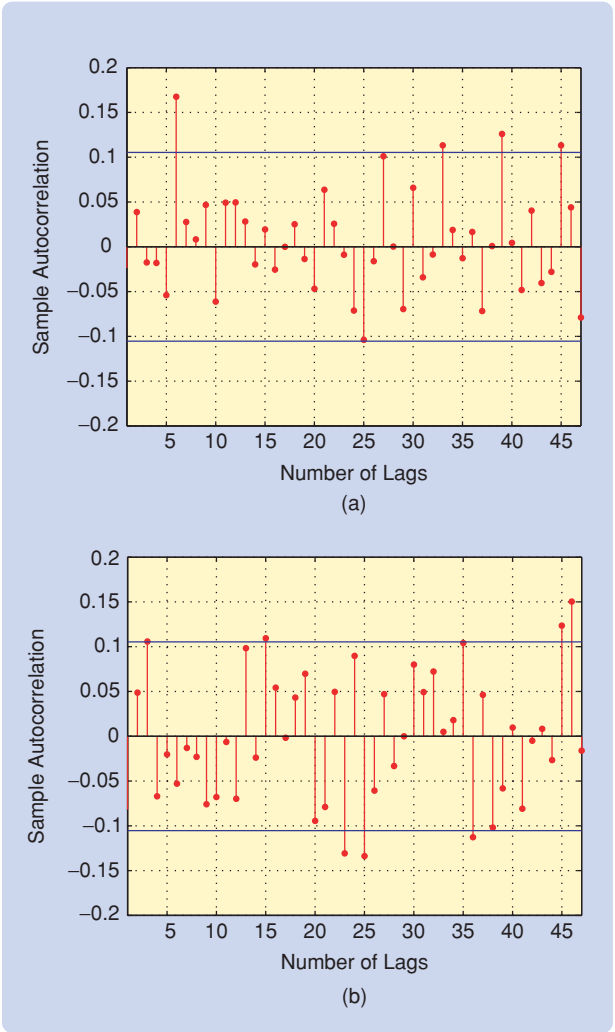

FIGURE 11 Autocorrelation of the NARX residuals on the test set. The horizontal band provides the reference for statistical significance at a 95% level. Significant autocorrelation on the residuals (a) for Series 1 is observed at lags 6, 34, 39, and 45 and (b) for Series 5 observed at lags 15, 23, 25, 36, 45, and 46.

available data with actual observations. The forecasting performance is quantified by using both the MSE and the mean absolute percentage error (MAPE)

$$\text{MAPE} = \sum_{i=1}^{N} \frac{|y_i - \hat{y}_i|}{y_i}. \tag{27}$$

Forecasts are obtained by computing the one-step-ahead prediction and the 24-h-ahead-simulation with updates at 00:00 h of each day.

Forecasting Performance

Table 2 compares the performance of the estimated models for the two forecasting modes over the five load series. The FS-LSSVM performs better than the traditional LS-SVM by using more information and including the entire available data, rather than using only the last 1000 data points. In the context of STLF, the existence of seasonal variations makes it useful to include as many data points as possible into the model. On the other hand, the linear model shows good performance in some series, but is always outperformed by the FS-LSSVM.

The FS-LSSVM and the linear model forecasts are compared in figures 8 and 9 for two substations. The top panels show the 1-h-ahead forecasts, while the bottom panels show the 24-h-ahead simulations. Each plot shows the first seven days of the test set, starting with 00:00 h on Monday. The FS-LSSVM models provide better forecasts, particularly for the case of 24-h-ahead simulation. The variation in performance observed for each load series also reflects the fact that the composition of the underlying customers is different.

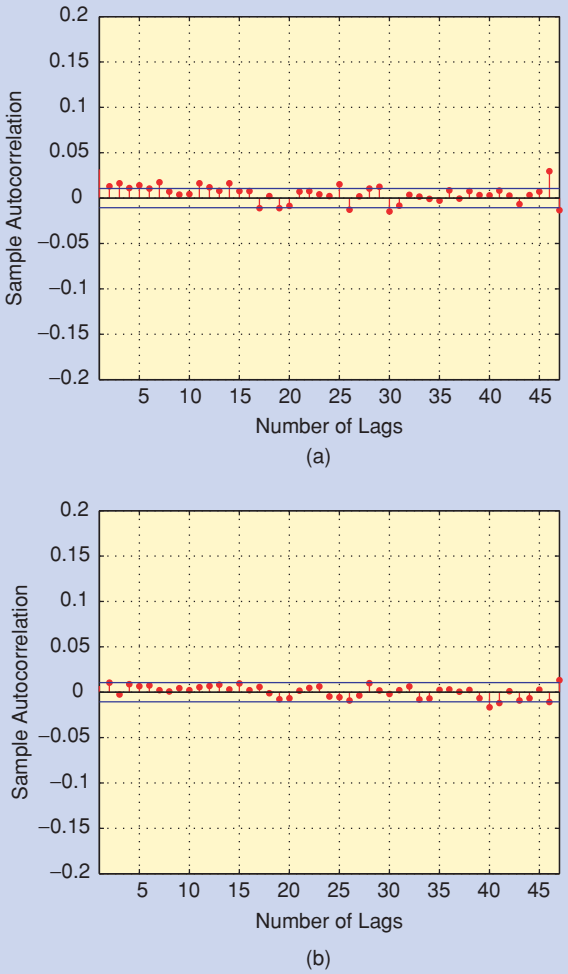

FIGURE 12 Autocorrelation of the AR-NARX residuals in the training set. The autocorrelation is largely reduced by using the AR-NARX model structure for both (a) Series 1 and (b) Series 5.

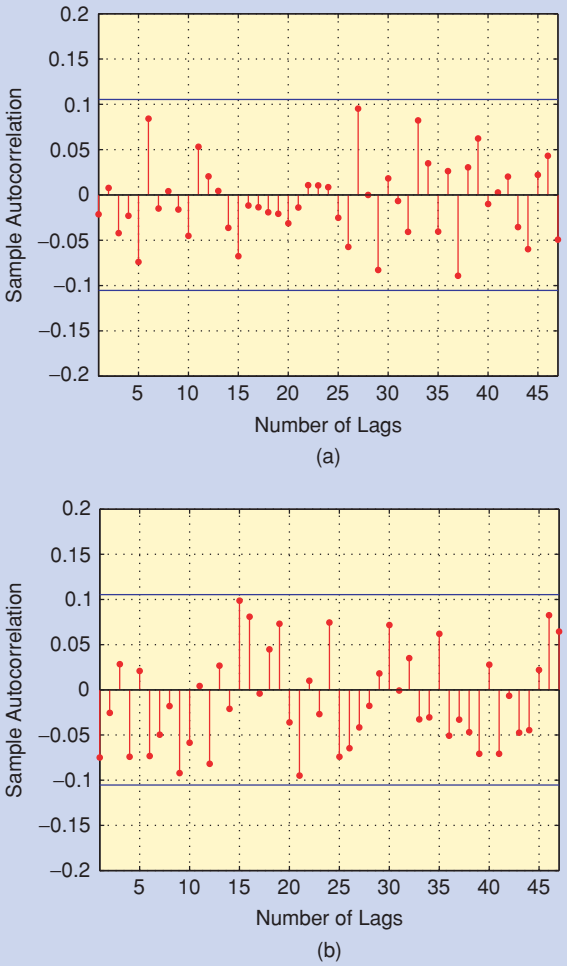

FIGURE 13 Autocorrelation of the AR-NARX residuals in the test set. Using the re-estimated models on the test set, we obtain no significant autocorrelation in the out-of-sample residuals for (a) Series 1 and (b) Series 5.

Model Validation and Improvements

Usually, model validation is the last stage of any modeling task in system identification. The examination of the residuals of a model provides insight into the validity of the assumptions behind the model structure and estimation technique used in the process of model building, as explained in "Model Validation."

Consider the five NARX models estimated with FS-LSSVM and their residuals in the training sample. Autocorrelation is detected in the residuals of the models for Series 1 and Series 5, with significant autocorrelation at lags 24 and elsewhere, as shown on Figure 10. The residuals of the remaining three models do not show significant autocorrelation. These findings suggest that the NARX model structure does not capture all of the temporal dynamics in the data from Series 2 and Series 5. When these models are used to produce the out-of-sample forecasts, the test set residuals also have a pattern of significant autocorrelation, as shown on Figure 11. One way to correct the autocorrelation is to use the AR-NARX model structure described in "Model Structures and LS-SVM." The AR-NARX incorporates a temporal dependence on the residuals of the nonlinear regression. The models for Series 1 and 5 are re-estimated with an AR-NARX structure, with nonzero coefficients for ρ_k in (S4) at the lags identified using the autocorrelation plots.

The estimated AR-NARX models for Series 1 and 5 give residuals in the training set for which the autocorrelation problem is reduced, as shown on Figure 12. The lack of autocorrelation confirms that the models have a better structure. The forecasts on the test set, therefore, are also improved, and significant autocorrelation is also removed in the test set residuals, as shown on Figure 13. The forecasting performance is marginally improved, as given in the last column of Table 2.

CONCLUSIONS

This article illustrates the application of a nonlinear system identification technique to the problem of STLF. Five NARX models are estimated using fixed-size LS-SVM, and two of the models are later modified into AR-NARX structures following the exploration of the residuals. The forecasting performance, assessed for different load series, is satisfactory. The MSE levels on the test data are below 3% in most cases. The models estimated with fixed-size LS-SVM give better results than a linear model estimated with the same variables and also better than a standard LS-SVM in dual space estimated using only the last 1000 data points. Furthermore, the good performance of the fixed-size LS-SVM is obtained based on a subset of $M = 1000$ initial support vectors, representing a small fraction of the available sample. Further research on a more dedicated definition of the initial input variables (for example, incorporation of external variables to reflect industrial activity, use of explicit seasonal information) might lead to further improvements and the extension toward other types of load series.

ACKNOWLEDGMENTS

This work was supported by grants from the following: GOA AMBioRICS, CoE EF/05/006 OPTEC, FWO (G.0499.04, G.0211.05, G.0302.07), and IUAP P6/04 (DYSCO).

REFERENCES

[1] D. Bunn, "Forecasting load and prices in competitive power markets," *Proc. IEEE*, vol. 2, no. 88, pp. 163–169, 2000.

[2] J. Jardini, C. Tahan, M. Gouvea, and S. Ahn, "Daily load profiles for residential, commercial and industrial low voltage consumers," *IEEE Trans. Power Delivery*, vol. 15, no. 1, pp. 375–380, 2000.

[3] E. Carpaneto, G. Chicco, R. Napoli, and M. Scutariu, "Customer classification by means of harmonic representation of distinguishing features," in *Proc. IEEE Power Tech. Conf.*, vol. 3, Bologna, June 2003.

[4] J. Taylor and R. Buizza, "Neural network load forecasting with weather ensemple predictions," *IEEE Trans. Power Syst.*, vol. 17, no. 2, pp. 626–632, 2002.

[5] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, NJ: Prentice Hall, 1987.

[6] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Deylon, P. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modelling in system identification: A unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[7] G. Box and G. Jenkins, *Time Series Analysis, Forecasting and Control*, 3rd ed. Englewood Cliffs, NJ: Prentice Hall, 1994.

[8] J. Nowicka–Zagrajek and R. Weron, "Modeling electricity loads in California: ARMA models with hyperbolic noise," *Signal Processing*, vol. 82, no. 12, pp. 1903–1915, 2002.

[9] L. Yang and R. Tschernig, "Non-and semiparametric identification of seasonal nonlinear autoregression models," *Econometric Theory*, vol. 18, no. 6, pp. 1408–1448, 2002.

[10] P. Franses and R. Paap, *Periodic Time Series Models*. London, U.K.: Oxford Univ. Press, 2003.

[11] M. Espinoza, C. Joye, R. Belmans, and B. De Moor, "Short term load forecasting, profile identification and customer segmentation: A methodology based on periodic time series," *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1622–1630, 2005.

[12] N. Amjady, "Short-term hourly load forecasting using time-series modeling with peak load estimation capability," *IEEE Trans. Power Syst.*, vol. 16, no. 4, pp. 798–805, 2001.

[13] R. Ramanathan, R. Engle, C. Granger, C. Vahid-Aragui, and F. Brace, "Short-run forecasts of electricity load and peaks," *Int. J. Forecasting*, vol. 13, no. 2, pp. 161–174, 1997.

[14] S.-J. Huang and K.-R. Shih, "Short term load forecasting via ARMA model identification including non-gaussian process considerations," *IEEE Trans. Power Syst.*, vol. 18, no. 2, pp. 673–679, 2003.

[15] A. Khotanzad, R. Afkhami-Rohani, and D. Maratukulam, "ANNSTLF-artificial neural network short-term load forecaster-generation three," *IEEE Trans. Power Syst.*, vol. 13, no. 4, pp. 1413–1422, 1998.

[16] K.-H. Kim, H.-S. Youn, and Y.-C. Kang, "Short-term load forecasting for special days in anomalous load conditions using neural networks and fuzzy inference method," *IEEE Trans. Power Syst.*, vol. 15, no. 2, pp. 559–565, 2000.

[17] L. Mohan Saini and M. Kumar Soni, "Artificial neural network-based peak load forecasting using conjugate gradient methods," *IEEE Trans. Power Syst.*, vol. 17, no. 3, pp. 907–912, 2002.

[18] D. Fay, J. Ringwood, M. Condon, and M. Kelly, "24-h electrical load data–A sequential or partitioned time series?" *Neurocomputing*, vol. 55, no. 3, pp. 469–498, 2003.

[19] H. Hippert, D. Bunn, and R. Souza, "Large neural networks for electricity load forecasting: Are they overfitted?" *Int. J. Forecasting*, vol. 21, no. 3, pp. 425–434, 2005.

[20] H. Steinherz, C. Pedreira, and R. Castro, "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Trans. Power Syst.*, vol. 16, no. 1, pp. 44–55, 2001.

[21] G. Gross and F. Galiana, "Short-term load forecasting," *Proc. IEEE*, vol. 75, no. 12, pp. 1558–1573, 1987.

[22] R. Engle, C. Granger, J. Rice, and A. Weiss, "Semiparametric estimates of the relation between weather and electricity sales," *J. Amer. Statist. Assoc.*, vol. 81, no. 394, pp. 310–320, 1986.
[23] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines*. Singapore: World Scientific, 2002.
[24] M. Espinoza, J.A.K. Suykens, and B. De Moor, "LS-SVM regression with autocorrelated errors," in *Proc. 14th IFAC Symp. Syst. Identification*, Mar. 2006, pp. 582–587.
[25] W. Hardle, *Applied Nonparametric Regression*. Cambridge, UK: Cambridge Univ. Press, 1990.
[26] E. Parzen, "On estimation of a probability density function and mode,"*Annals Math. Statistics*, vol. 33, no. 1065–1076, 1962.
[27] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA: SIAM, 1990.
[28] M. Loeve, *Probability Theory II*. New York: Springer-Verlag, 1978.
[29] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.
[30] J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Philos. Trans. R. Soc. London*, vol. 209, pp. 415–446, 1909.
[31] C.J.C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, June 1998.
[32] I. Steinwart, D. Hush, and C. Scove, "An explicit description of the reproducing kernel Hilbert spaces of Gaussian RBF kernels," *IEEE Trans. Inform. Theory*, vol. 52, pp. 4635–4643, 2006.
[33] C. Micchelli, Y. Xu, and H. Zhang, "Universal kernels," *J. Mach. Learning Res.*, vol. 6, pp. 2651–2667, Dec. 2006.
[34] M. Genton, "Classes of kernel for machine learning: A statistics perspective," *J. Mach. Learning Res.*, vol. 2, pp. 299–312, Dec. 2001.
[35] M. Girolami, "Orthogonal series density estimation and the kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 6, pp. 1455–1480, 1998.
[36] C. Williams and M. Seeger, "Using the Nystrom method to speed up kernel machines," in *Proc. NIPS 2000*, vol. 13., pp. 682–688.
[37] M. Espinoza, J.A.K. Suykens, and B. De Moor, "Least squares support vector machines and primal space estimation," in *Proc. 42nd IEEE Conf. Decision and Control*, 2003, pp. 5716–5721.
[38] D. MacKay, "Bayesian interpolation," *Neural Comput.*, vol. 4, no. 3, pp. 415–447, 1992.
[39] T. Pena Centeno and N. Lawrence, "Optimising kernel parameters and regularisation coefficients for non-linear discriminant analysis," *J. Mach. Learning Res.*, vol. 7, pp. 456–491, Feb. 2006.
[40] T. Van Gestel, M. Espinoza, J.A.K. Suykens, and B. De Moor, "Bayesian input selection for nonlinear regression with LS-SVMs," *in Proc. 13th Syst. Identification Symp.*, 2003, pp. 578–583.
[41] T. Van Gestel, M. Espinoza, B. Baesens, J.A.K. Suykens, C. Brasseur, and B. De Moor, "A bayesian nonlinear support vector machine error correction model," *J. Forecasting*, vol. 25, no. 2, pp. 77–100, March 2006.
[42] G. Golub, M. Heath, and G. Wahba, "Generalized cross-validation: A method for choosing a good ridge regression parameter," *Technometrics*, vol. 21, pp. 215–223, 1979.
[43] M. Espinoza, J.A.K. Suykens, and B. De Moor, "Fixed-size least squares support vector machines: A large scale application in electrical load forecasting," *Comput. Management Sci.*, vol. 3, no. 2, pp. 113–129, April 2006.
[44] A. Juditsky, H. Hjalmarsson, A. Benveniste, B. Deylon, L. Ljung, J. Sjoberg, and Q. Zhang, "Nonlinear black-box modelling in system identification: Mathematical foundations," *Automatica*, vol. 31, pp. 1725–1750, 1995.
[45] R. Guidorzi, *Multivariable System Identification: From Observations to Models*. Italy: Bononia Univ. Press, 2003.
[46] P. Speckman, "Kernel smoothing in partial linear models," *J. R. Statist. Soc. B*, vol. 50, pp. 413–136, 1988.
[47] M. Espinoza, J.A.K. Suykens, and B. De Moor, "Partially linear models and least squares support vector machines," in *Proc. 43rd IEEE Conf. Decision and Control*, 2004, pp. 3388–3393.
[48] M. Espinoza, J.A.K. Suykens, and B. De Moor, "Kernel based partially linear models and nonlinear identification,"*IEEE Trans. Automat. Contr.*, vol. 50, no. 10, pp. 1602–1606, 2005.

AUTHOR INFORMATION

Marcelo Espinoza (marcelo.espinoza@esat.kuleuven.be) received the civil industrial engineering degree and a M.Sc. in applied economics from the Universidad de Chile in 1998. He received a master degree in artificial intelligence in 2002 and his Ph.D. degree in 2006 from the Katholieke Universiteit Leuven, Belgium. Currently, he is a postdoctoral researcher with the Electrical Engineering Department of the K.U. Leuven. He can be contacted at Kasteelpark Arenberg 10, 3001 Heverlee, Belgium.

Johan A.K. Suykens received his master's degree in electromechanical engineering and the Ph.D. degree in applied sciences from the K.U. Leuven, Belgium, in 1989 and 1995, respectively. In 1996 he was a visiting postdoctoral researcher at the University of California, Berkeley. He is currently a professor with the K.U. Leuven. His research interests include the theory and application of neural networks and nonlinear systems. He is author of *Artificial Neural Networks for Modelling and Control of Nonlinear Systems* (Kluwer Academic Publishers) and *Least Squares Support Vector Machines* (World Scientific) and coauthor of *Cellular Neural Networks, Multi-Scroll Chaos and Synchronization* (World Scientific), and is associate editor for *IEEE Transactions on Neural Networks* and *IEEE Transactions on Circuits and Systems*. He received the International Neural Networks Society INNS 2000 Young Investigator Award for significant contributions in neural networks.

Ronnie Belmans received the M.S. degree in electrical engineering in 1979 and the Ph.D. degree in 1984, both from the K.U. Leuven, Belgium, the special doctorate in 1989 and the habilitierung in 1993, both from the RWTH, Aachen, Germany. Currently, he is a full professor with the K.U. Leuven. His research interests include techno-economic aspects of power systems, power quality, and distributed generation. He is also guest professor at Imperial College of Science, Medicine and Technology, London-UK. Since June 2002 he has been chairman of the board of directors of ELIA, the Belgian transmission grid operator. He is a Fellow of the IEEE.

Bart De Moor obtained his master's degree in electrical engineering in 1983 and a Ph.D. in engineering from the K.U. Leuven, Belgium, where he is a full professor. He was a visiting research associate at Stanford University from 1988 to 1990. His research interests are in numerical linear algebra and optimization, system theory, control and identification, quantum information theory, data mining, information retrieval and bioinformatics. He has published more than 250 journal papers, 350 conference proceedings publications, five books, and numerous science popularizing contributions. He was awarded the 1986 Leybold-Heraeus Prize, 1989 Leslie Fox Prize, 1990 Guillemin-Cauer best paper Award of *IEEE Transactions on Circuits and Systems*, 1992 Laureate of the Belgian Royal Academy of Sciences, 1994 Siemens Award, 1996 best paper award of *Automatica*, and the 1999 IEEE Signal Processing Society Best Paper Award. From 1991–1999 he was the chief advisor on science and technology for the Belgian Federal and the Flanders Regional Governments. At present, he is the chief-of-staff of the minister-president of Flanders. He is a Fellow of the IEEE.