# Short-term Load Forecasting based on Wavelet Approach

Ali Karami Ghanavati[1], Amir Afsharinejad[1], Navid Vafamand[1], Mohammad Mehdi Arefi[1]
Mohammad Sadegh Javadi[2], and João P. S. Catalão[2,3]

[1]School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran
[2]Institute for Systems and Computer Engineering, Technology and Science (INESC TEC), 4200-465 Porto, Portugal
[3]Faculty of Engineering, University of Porto (FEUP), 4200-465 Porto, Portugal

Emails: alikarami.shirazu@gmail.com (A.K.G.), amir.afshar.ctrleng@gmail.com (A.A.), n.vafamand@shirazu.ac.ir (N.V.),
arefi@shirazu.ac.ir (M.M.A.), msjavadi@gmail.com (M.S.J.), catalao@fe.up.pt (J.P.S.C.)

*Abstract*—**This paper develops a novel short-term load forecasting technique to predict the demanding power for the next hour. In this study, a linear equation-error Auto Regressive Auto Regressive Moving Average Exogenous (ARARMAX) model is trained to specify power consumption as a function of a few past hours. The parameters of the candidate mathematical model are estimated by using two least squares-based iterative algorithms. The main difference with these algorithms is the total number of past data involved in the modeling. Whereas practical data are always subject to noise and un-accurate measuring, a wavelet de-noising technique is utilized to reduce the effect of noise on forecasting which leads to more precise predictions. The superiority of the proposed approach is validated by utilizing practical data from a power utility in Canada in January 1995. The first three days' data are utilized to train the selected model and the fourth-day data are dedicated to test the prediction of the provided model. The $L_2$ and $L_\infty$ norms error and MAPE, MAE, and RMSE are selected as criteria to show the merits of the proposed approach.**

*Keywords— Load forecasting, System identification, Least Squares Algorithm, ARARMAX model, Wavelet.*

## NOMENCLATURE

### Least squares modeling parameters

| | |
|---|---|
| $n_a, n_b, n_c, n_p, n_q$ | Number of coefficients in $A(z)$, $B(z)$, $C(z)$, $P(z)$, $Q(z)$ |
| $\vartheta$ | Parameter vector |
| $\emptyset(t), \widehat{\emptyset}(t)$ | Information vector and its estimation |
| $w(t-i), \widehat{w}(t-i)$ | Unmeasurable noise and its estimation |
| $k, k_{max}$ | Iteration number and its maximum value |
| $L$ | Number of input/output data |

### De-noising Wavelet parameters

| | |
|---|---|
| $\psi(t)$ | Mother wavelet |
| $g, h$ | Low-pass and high-pass filters |
| $y_{noisy}, y_{denoised}$ | Noisy and noise-free signals |

## I. INTRODUCTION

By the continuous penetration of electrical energy to the industry and non-industry utilities, the request and consumption of electricity are increasing daily. To treat this daily demand and assure the economic and effective operation of power systems, daily predication of the power demand is essential. The load forecasting can be categorized as long-term and short-term. The long-term furcation deals with the general strategy and planning for establishing new power generator utilities and distribution networks [1]. Meanwhile, the short-term forecasting deals with the daily power demand and its prediction for the next 24 hours to a few days based on the historical statistics [2]. The short-term load forecasting precision can affect the operation reliability and cost-effectiveness of power systems and electricity marketing. On the other hand, the short-term historical statistics are noisy and may be inaccurate for some hours in a day. Furthermore, they have complex behavior. Thereby, predicting future power demand is a challenging issue. In this regard, using proper modeling algorithms and de-noising approaches are vital [2].

The problem of load forecasting is investigated in several references with different methods [3]. For instance, in [4], a chaos theory-based maximum overlap discrete wavelet transform neural network was developed. In [5], a probabilistic autoregressive moving average (ARMA) was developed for the day-ahead forecasting of a load. The model was trained based on the least-squares (LS)-based iterative methods. In [6], probabilistic multi-stage load forecasting was presented and four wavelet-based decompositions were used in order to address the effect of pre-processing. The short-term load forecasting model for energy management systems by the use of a neural network was studied in [7]. An accurate model to predict the power request in the residential portion for a month in advance was studied in [8]. Monthly power demand, air, and social variables were selected as key factors in the study. That approach was taken advantage of support vector regression and fuzzy rough feature selection with particle swarm optimization algorithms. Two novel Kalman-based learning algorithms for an online Takagi-Sugeno fuzzy ARMA model identification were proposed in [9]. However, the computational burden of those approaches is high and the trained model needs several factors as inputs.

In the presented paper, an equation-error ARARMAX model is trained to deal with the short-time load forecasting of power demand. A systematic LS-based approach is developed to compute the parameters of the equation-error ARARMAX model. In the proposed modeling approach, wavelet as the pre-processing de-noising technique is considered. First, by using the wavelet a pre-processing is done on the data and then by using a method of model identification, parameters of the model are obtained. The main contribution of this paper is to present a novel least-squares approach to estimate all parameters of the ARARMAX system adopted for the short-term load forecasting. Several simulations of practical data gathered from a power utility in Canada in January 1995 are carried out. The effect of pre-processing de-noising method and the different LS approaches on the system accuracy is illustrated.

The rest of the paper is organized as follows: In Section 2 the proposed algorithm that consists of model identification algorithms, wavelet, and load forecasting is presented. In Section 3 simulations and comparison results between two approaches of with and without pre-processing are provided. Finally, conclusions are drawn in Section 4.

## II. Proposed Approach

### A. Identification model for a signal and its parameter estimation

The specifications of a time-series signal can be described by a mathematical model comprising the last values of the time-series and a colored noise. However, it is important to estimate the parameters of the mathematical model based on the time-series data. In this section, an ARMA model is used to capture the behavior of the short-term load time-series, as follows

$$A(z)y(t) = \frac{D(z)}{C(z)}w(t), \quad (1)$$

where $y(t)$ is the time-series data and $w(t)$ is a white noise with zero mean and variance $\sigma^2$. Also, the polynomials $A(z)$, $C(z)$, $D(z)$ in the unit backward shift operator $z^{-1}$ are described as:

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_{n_a} z^{-n_a},$$
$$C(z) = 1 + c_1 z^{-1} + c_2 z^{-2} + \cdots + c_{n_c} z^{-n_c}, \quad (2)$$
$$D(z) = 1 + d_1 z^{-1} + d_2 z^{-2} + \cdots + d_{n_d} z^{-n_d}.$$

For the case study of the load prediction, the general structure of $A(z)$ given in (2) is chosen as [9]:

$$A(z) = 1 + a_1 z^{-1} + a_{24} z^{-24} + a_{25} z^{-25}. \quad (3)$$

Since the matrix $A(z)$ is not ordered sequentially and the input $u(t)$ does not appear in (1), a recursive technique is suggested in Algorithm 1. In Algorithm 1, the parameters $p_i$ and $d_i$ are identified. However, it is not feasible to compute the original parameters $a_i$ and $c_i$ in (1). Although the system (4) is accurate and can represent the behavior of the short-term load time-series, it needs more information of the past load values, because the order of $P(z)$ is higher than that of the $A(z)$. In the next algorithm, the drawback of Algorithm 1 is solved, adequate since our goal is forecasting and the entrance $u(t)$ is omitted.

---

**Algorithm 1 (Conventional least-square error) [10]**

1. Multiply both sides of (1) by $C(z)$, as
$$A(z)C(z)y(t) = D(z)w(t). \quad (4)$$

2. Let
$$P(z) = 1 + p_1 z^{-1} + p_2 z^{-2} + \cdots + p_{n_p} z^{-n_p}. \quad (5)$$
where $n_p = n_a + 2n_c$ where $n_a = 3$ here. Therefore, (4) results
$$P(z)y(t) = D(z)w(t). \quad (6)$$

3. Define the parameter and information vectors $\vartheta$ and $\emptyset(t)$ as
$$\vartheta = \begin{bmatrix} p_1 & \dots & p_{n_p} & d_1 & \dots & d_{n_d} \end{bmatrix}^T, \quad (7)$$
$$\emptyset(t) = \begin{bmatrix} -y(t-1) & \dots \\ -y(t-n_p) & w(t-1) & \dots & w(t \\ -n_d) \end{bmatrix}^T. \quad (8)$$

4. Therefore, the regressor equation is chosen as
$$y(t) = \emptyset^T(t)\vartheta + w(t). \quad (9)$$

5. Using all available data and ignoring the white noise $w(t)$ to find the best estimation of $\vartheta$, the regressor equation will be
$$Y = \Phi\vartheta \quad (10)$$
where
$$Y = [y(1) \ \dots \ y(L)]^T,$$
$$\Phi(L) = [\emptyset(1) \ \dots \ \emptyset(L)]^T,$$
$$\emptyset(t) = \begin{bmatrix} -y(t-1) & \dots \\ -y(t-n_p) & w(t-1) & \dots & w(t \\ -n_d) \end{bmatrix}^T, \quad (11)$$

6. The vector $\emptyset(t)$ contains unmeasurable noise terms $w(t-i)$ and it can be not constructed simply. So, to estimate the noise terms, an iterative procedure should be exploited. Perform the following iterative procedure of the index $k$ for $1 \le k \le k_{max}$
$$\widehat{\emptyset}_k(t) := \begin{bmatrix} -y(t-1) & \dots \\ -y(t-n_p) & \widehat{w}_{k-1}(t \\ -1) & \dots & \widehat{w}_{k-1}(t-n_d) \end{bmatrix}^T,$$
$$\widehat{\Phi}_k(L) = \begin{bmatrix} \widehat{\emptyset}_k(1) & \widehat{\emptyset}_k(2) & \dots & \widehat{\emptyset}_k(L) \end{bmatrix}^T, \quad (12)$$
$$\hat{\vartheta}_k = \begin{bmatrix} \widehat{\Phi}_k^T(L)\widehat{\Phi}_k(L) \end{bmatrix}^{-1} \widehat{\Phi}_k^T(L)Y(L),$$
$$\widehat{w}_k(t) = y(t) - \widehat{\emptyset}_k^T(t)\hat{\vartheta}_k, \quad t = 1,2,\dots,L,$$
where $\widehat{w}_{k-1}(t-i)$ is the estimate of the unknown noise term $w(t-i)$, $L$ is the length of a group of data and $\widehat{\emptyset}_k$ is the estimation of $\emptyset(t)$.

7. The output of the algorithm is $\hat{\vartheta}_{k_{max}}$.

---

To find the unknown parameters of the ARMA equation error model (1), the fictitious control input is added to the model (1). Based on the structure of $A(z)$ in (2), the following modified equation-error ARARMAX is considered:

$$\bar{A}(z)y(t) = B(z)u(t) + \frac{D(z)}{C(z)}w(t), \quad (13)$$

where
$$u(t) = -y(t)$$
$$\bar{A}(z) = 1 + a_1 z^{-1},$$
$$B(z) = b_1 z^{-24} + b_2 z^{-25}, \qquad (14)$$
with $C(z)$ and $D(z)$ given in (2). Note that the coefficients $b_1$ and $b_2$ stand for the original coefficients $a_{24}$ and $a_{25}$ in (2), respectively.

### Algorithm 2 (Modified ARARMAX model and least square)

1. Multiply both sides of (13) by $C(z)$, as
$$\bar{A}(z)C(z)y(t) = B(z)C(z)u(t) + D(z)w(t). \qquad (15)$$

2. Let
$$P(z) := C(z)\bar{A}(z)$$
$$= 1 + p_1 z^{-1} + p_2 z^{-2} + \cdots \qquad (16)$$
$$+ p_{n_p} z^{-n_p}.$$

$$Q(z) := C(z)B(z)$$
$$= q_1 z^{-24} + q_2 z^{-25} + \cdots \qquad (17)$$
$$+ q_{n_q} z^{-(25+nc)},$$

where $n_p = n_a + n_c$, and $n_q = n_b + n_c$ with $n_a = 1$ and $n_b = 2$ here. Therefore, (15) results
$$P(z)y(t) = Q(z)u(t) + D(z)w(t). \qquad (18)$$

3. Define the parameter and information vectors $\vartheta$ and $\emptyset(t)$ as
$$\vartheta = \begin{bmatrix} p_1 & \cdots & p_{n_p} & q_1 & \cdots & q_{nq} & d_1 & \cdots & d_{n_d} \end{bmatrix}^T, \qquad (19)$$

$$\emptyset(t) = \begin{bmatrix} -y(t-1) & \cdots \\ -y(t-n_p) & u(t-24) & \cdots & u(t \\ -25-n_c) & w(t-1) & \cdots & w(t \\ -n_d) \end{bmatrix}^T. \qquad (20)$$

Therefore, the regressor equation is chosen as
$$y(t) = \emptyset^T(t)\vartheta + w(t). \qquad (21)$$

4. Use steps 5-7 of Algorithm 1.

Although the output of Algorithm 2 is not still the parameters of the model (13), it is feasible to compute the parameters $a_i$, $b_i$, $c_i$, and $d_i$ [11]. From the definitions of $P(z)$ and $Q(z)$ in (16) and (17), one infers that
$$B(z)P(z) = A(z)Q(z). \qquad (22)$$

Now, replacing the estimates of polynomials, $B(z)$, $P(z)$, $A(z)$ and $Q(z)$ into (22), it follows that
$$(b_1 z^{-24} + b_2 z^{-25})\left(1 + p_1 z^{-1} + p_2 z^{-2} + \cdots \right.$$
$$\left. + p_{n_p} z^{-n_p}\right)$$
$$= (1 + a_1 z^{-1})(q_1 z^{-24} + q_2 z^{-25} + \qquad (23)$$
$$+ q_{nq} z^{-(25+nc)}).$$

Letting the coefficients of the same power of $z^{-1}$ on both sides of (23) equal to each other, $(2 + n_p)$ equations are obtained as
$$z^{-24}: \quad b_1 = q_1,$$
$$z^{-25}: \quad b_1 p_1 + b_2 = q_1 a_1 + q_2,$$
$$\vdots \qquad (24)$$
$$z^{-(25+n_p)+1}: \quad b_1 p_{n_p} + b_2 p_{n_p-1} = q_{n_q-1} a_1 + q_{n_q},$$

$$z^{-(25+n_p)}: \quad b_2 p_{n_p} = q_{n_q} a_1,$$

Or equivalently [10]
$$S_1 \begin{bmatrix} b_1 \\ b_2 \\ a_1 \end{bmatrix} = M_1, \qquad (25)$$

where
$$S_1 := \begin{bmatrix} 1 & 0 & -1 \\ p_1 & 1 & -q_1 \\ p_2 & p_1 & -q_2 \\ \vdots & \vdots & \vdots \\ p_{n_p} & p_{n_p-1} & -q_{n_q-1} \\ 0 & 0 & -q_{n_q} \end{bmatrix} \in \mathbb{R}^{(2+n_p)\times(3)}, \qquad (26)$$

$$M_1 := \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_{n_q} \\ 0 \end{bmatrix} \in \mathbb{R}^{2+n_p},$$

Finally, the estimates of the system parameters are computed from
$$\begin{bmatrix} b_1 \\ b_2 \\ a_1 \end{bmatrix} = [S_1^T S_1]^{-1} S_1^T M_1. \qquad (27)$$

From (16), one also has
$$(1 + c_1 z^{-1} + c_2 z^{-2} + \cdots + c_{n_c} z^{-n_c})(1 + a_1 z^{-1})$$
$$= 1 + p_1 z^{-1} + p_2 z^{-2} + \cdots \qquad (28)$$
$$+ p_{n_p} z^{-n_p}.$$

Considering the same reasoning as (24), the following matrix equation is given [10]:
$$S_2 \begin{bmatrix} c_1 \\ \vdots \\ c_{n_c} \end{bmatrix} = M_2, \qquad (29)$$

where
$$S_2 := \begin{bmatrix} 1 & 0 & \cdots & 0 \\ a_1 & 1 & \ddots & \vdots \\ 0 & a_1 & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 \\ 0 & 0 & 0 & a_1 \end{bmatrix} \in \mathbb{R}^{n_p \times n_c},$$
$$M_2 := \begin{bmatrix} p_1 - a_1 \\ p_2 \\ p_3 \\ \vdots \\ p_{n_p} \end{bmatrix} \in \mathbb{R}^{n_p}, \qquad (30)$$

Finally, one has [10]
$$\begin{bmatrix} c_1 \\ \vdots \\ c_{n_c} \end{bmatrix} = [S_2^T S_2]^{-1} S_2^T M_2. \qquad (31)$$

### Algorithm 3 (Extracting the model coefficients)

1. Get the output of Algorithm 2 (i.e. $\hat{\vartheta}_{k_{max}}$). So, the coefficients of $D(z)$ are available.
2. Construct matrices $S_1$ and $M_1$ in (26).
3. Compute the coefficients of $\bar{A}(z)$ and $B(z)$ by (27).
4. Calculate the coefficients of $A(z)$ from $\bar{A}(z)$ and $B(z)$ from (14).
5. Construct matrices $S_2$ and $M_2$ in (30).
6. Compute the coefficients of $C(z)$ by (31).

The summary of extracting the ARARMAX equation error model (13) is given in Algorithm 3.

*B. De-noising data using wavelet*

In Subsection A, the way of training a proper ARAMA model based on the given time-series is presented. However, in practice, the gathered data is subjected to noise. To have a smooth output signal it is needed to decompose the gather data from its noise.

In this regard, the wavelet transform is used as the preprocessing procedure to remove the high-frequency information of the data. Wavelet transform can be utilized to analyze the frequency characteristics of a signal [12]. The ability to evaluate data in different frequency bands and scales can help examine the data behavior in different small sections of frequency and eliminating the high-frequency sections. Thereby, it can be widely utilized in the de-noising procedure. To perform the de-noising based on the wavelet, the short-time Fourier transform (STFT) was introduced [8]. By the means of the STFT, each of the time-windowed blocks of a signal is analyzed.

The wavelet has two main parts, the mother wavelet which is characterized wavelet family, and the daughter wavelets. The mother wavelet is given by [13]:

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \quad a \in \mathbb{R}^+, b \in \mathbb{R}, \tag{32}$$

where $a$ is the scaling factor and $b$ is the shifting factor.

To calculate the discrete wavelet transform (DWT), the signal $x$ is passed through a low-pass and high-pass filter, which are named $g$ and $h$. The high-pass $y_{high}(n)$ and the low-pass filter outputs $y_{low}(n)$ are so-called detail and approximation coefficients.

$$y_{low}(n) = \sum_{k=-\infty}^{\infty} x[k]g[2n-k] \tag{33}$$

$$y_{high}(n) = \sum_{k=-\infty}^{\infty} x[k]h[2n-k] \tag{34}$$

The decomposition of the signal obtained from (33) and (34) should be repeated for high-frequency resolutions [14]. Fig. 1 provides the three-level decomposition using the DWT.
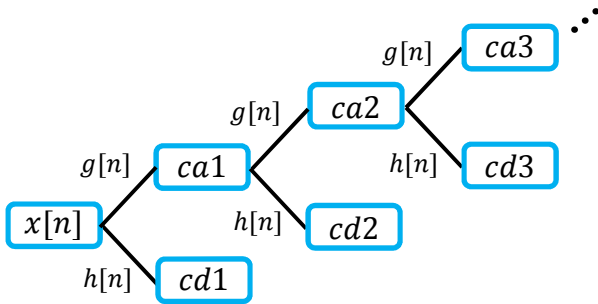


Fig. 1. Wavelet three-level decomposition of a time-domain signal [14].

In Fig. 1, $x[n]$ denotes the time-domain signal, $ca1$, $ca2$, and $ca3$ are the approximation coefficients of levels one, two, and three, respectively.

The detail coefficients of levels one, two, and three are respectively $cd1$, $cd2$, and $cd3$, and $g[n]$ and $h[n]$ are low-pass and high-pass filters, respectively. By substituting $a = 2^j$ and $b = k2^j$ into (32), the daughter wavelets shown in (35) are obtained.

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{2^j}}\psi(2^{-j}t - k) \tag{35}$$

in which both scale parameter, $j$, and shift parameter, k, are integers.

One of the techniques that helps wavelet transform to be used as a denoising tool, is the thresholding denoising technique [13]. The denoising problem is given by:

$$y_{noisy}(t) = y_{denoised}(t) + \Lambda e_i, \quad i = 1,2,\dots,n \tag{36}$$

where $y_{noisy}(t)$ and $y_{denoised}(t)$ denote noisy data and noise-free signal, $e_i$ is a normal distribution and $\Lambda$ is a particular noise level [15].

By using the thresholding technique, the detail coefficients are processed; then, the signal is reconstructed. The procedure of denoising the signals with the white noise can be done by the thresholding technique.

By using this method, the noise will be restrained, meanwhile, the characteristics of the original signal will be remained [16]. Wavelet denoising uses hard- or soft-thresholding techniques. In the hard-thresholding technique, a specific value is assigned to the coefficient magnitude as it exceeds a pre-specified threshold. But, the soft-thresholding technique shrinks the coefficients as they exceed the threshold. Using the soft-thresholding technique is more common in the literature.

The soft-thresholding is given as follows:

$$THR_s(Y,T) = \begin{cases} sgn(Y)(|Y| - T), & |Y| \geq T \\ 0, & |Y| < T \end{cases}, \tag{37}$$

where function $sgn(.)$ is defined as

$$sgn(Y) = \begin{cases} 1, & Y > 0 \\ 0, & Y = 0 \\ -1, & Y < 0 \end{cases}. \tag{38}$$

The wavelet-based denoising technique is summarized in Algorithm 4.

**Algorithm 4 (denoising steps using wavelet)**

1. Decomposition: In this stage, the Haar mother wavelet [14] is selected and the number of decomposition levels is specified. Then, the wavelet decomposition of the signal $y_{noisy}(t)$ at last level five is performed.

2. Soft-threshold coefficients: For each of the levels specified in step 1, suitable thresholds are selected based on (38) and the soft thresholding is applied to the detail coefficients given in (33).

3. Reconstruction: The signal $y_{denoised}(t)$ is reconstructed using the original coefficients of the last level and the modified detail coefficients of all levels.

## C. Load forecasting

Based on Algorithms 1-3, the short term one-hour load forecasting is developed. Initially, the available data is denoised by the wavelet approach summarized in Algorithm 4. Then, two approaches are considered to train an ARARMA model. In the first approach, Algorithm 1 is used to train the coefficients of the system (1). Meanwhile, the second approach deploys Algorithms 2 and 3 to train and compute the coefficients of the system (13). Since the system (13) is equivalent to (1), the system (1) is obtained. Finally, the system (1) can be re-written as current load power as its previous values and colored noise, as

$$y(t) = f(y(t - i), w(t - j)) \tag{39}$$

for some $i, j \geq 1$. After the system (39) is trained, its time argument $t$ is replaced by $t + 1$. Consequently,

$$y(t + 1) = f(y(t + 1 - i), w(t + 1 - j)) \tag{40}$$

for some $i, j \geq 0$. By using the model (40) and by having the current and past values of the load power, it is feasible to one-hour-ahead load power. Thereby, the short-time load forecasting is achieved.

## III. SIMULATION RESULTS

In order to show the merits and accuracy of the suggested modeling, practical data of a power utility in Canada in the year 1995, as shown in Fig. 2 is considered. The data of three days for each hour are used to estimate the system dynamics and parameters and train the model and then utilized to forecast the load power demands of the fourth day.

Two scenarios are considered in this section. In the first scenario, the applicability of training algorithms is evaluated. In the second scenario, the effect of the de-noising technique on the trained data is investigated. In the simulation, $w(t)$ is selected as an uncorrelated noise sequence with zero mean and unit variance. The polynomials of the ARMA equation error model (1) are chosen as

$$\begin{aligned} C(z) &= 1 + c_1 z^{-1}, \\ D(z) &= d_1 z^{-1} + d_2 z^{-2}, \end{aligned} \tag{41}$$

**Scenario 1:** In this scenario, Algorithms 1-3 are utilized to train a proper model and forecast the load based on the original data. By using Algorithm 1, the model (6) with the parameters given in Table I is calculated. Moreover, deploying Algorithms 2 and 3 results in the model (13) with the coefficients specified in Table I. The outputs of the obtained models and the trained and test data are shown in Fig. 3.
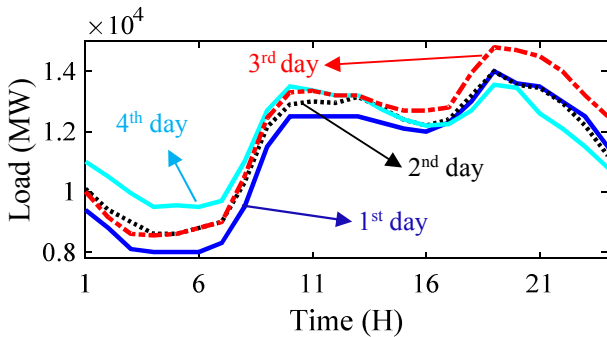


Fig. 2. Load of four days of January 95 [9].

TABLE I. ALGORITHMS 1-3 APPROACHES

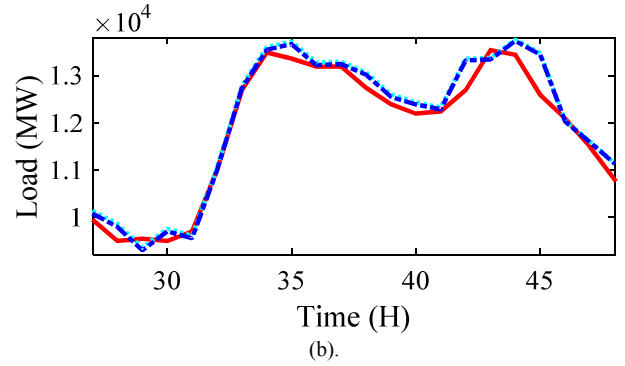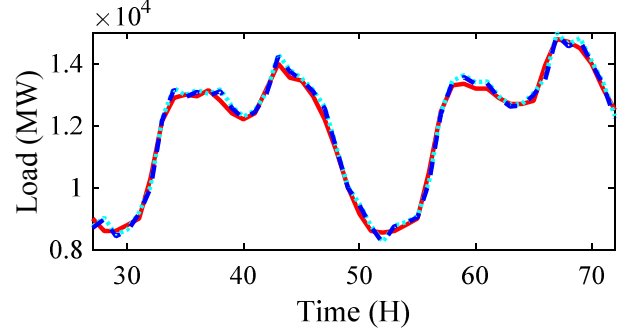| Coef. | Value | Coef. | Value | Coef. | Value |
|---|---|---|---|---|---|
| $p_1$ | −1.4621 | $d_1$ | 0.1966 | $c_1$ | −0.6091 |
| $p_2$ | 0.5386 | $d_2$ | −0.0552 | $d_1$ | 0.1966 |
| $p_{24}$ | −0.8479 | $a_1$ | −0.8638 | $d_2$ | −0.0552 |
| $p_{25}$ | 1.1939 | $a_{24}$ | −0.8480 | | |
| $p_{26}$ | −0.4285 | $a_{25}$ | 0.6862 | | |



(a).



(b).

Fig. 3. Real and output model (real data by solid red line, Algorithm 1 by dotted light blue line, and Algorithms 2 and 3 by dashed dark blue line) (a). Training data (2nd and 3rd days) and (b). Test data (4th day).

TABLE II. PREDICTION ERROR OF TWO APPROACHES FOR SCENARIO 1.

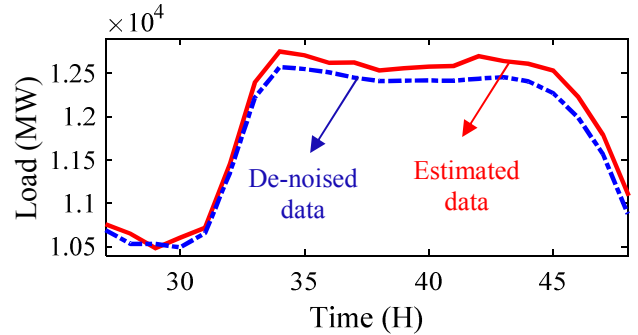| Approach | $L_2$ | $L_\infty$ | MAPE | MAE | RMSE |
|---|---|---|---|---|---|
| Algorithm 1 | 1358.8 | 853.9 | 0.0180 | 0.067 | 289.70 |
| Algorithms 2 and 3 | 1469.3 | 883.5 | 0.0204 | 0.070 | 313.24 |



Fig. 4. De-noised and predicted load using de-noised data.

To better compare the results of the predictions, Table II is provided. It should be pointed out that the $L_2$ and $L_\infty$ norms error, mean absolute percentage error (MAPE), root mean squared error (RMSE), and mean absolute error (MAE) between real and predicted load have been selected as a comparison criterion among the proposed algorithms.

As can be seen in Table II, the $L_2$ and MAPE performance criteria are improved based on the proposed Algorithms 2 and 3 over the conventional Algorithm 1 [10]. On the other hand, Algorithm 1 has a slightly more accurate prediction than Algorithms 2 and 3. Meanwhile, Algorithms 2 and 3 uses three current and past values (i.e. $y(t)$, $y(t-23)$, and $y(t-24)$), meanwhile Algorithm 1 needs 5 current and past values (i.e. $y(t)$, $y(t-1)$, $y(t-23)$, $y(t-24)$, and $y(t-25)$). Obviously, the approach 1 exploits more data and information with almost the same accuracy compared with the other approach.

**Scenario 2:** In this scenario, initially the wavelet technique is applied to the data to decompose noise. Then, Algorithms 2 and 3 are used to predict. Fig. 4. demonstrates de-noised data (de-noised measured load) and predicted load using these data with the parameters estimated by algorithm two. The results are reported in Table III, where it obviously shows the superiority of the proposed approach over using noisy data to predict the load on the fourth day. The calculation times of Algorithms 1-4 are computed via a 7-core computer with the Matlab 2019 and presented in Table IV. It is inferred from Table IV that the wavelet approach has a longer calculation time in comparison with the approaches of Scenario 1. Though, it results in more precise predictions.

TABLE III. PREDICTION ERROR OF ALGORITHMS 2 AND 3 WITH AND WITHOUT DE-NOISING DATA

| Approach | $L_2$ | $L_\infty$ | MAPE | MAE | RMSE |
|---|---|---|---|---|---|
| Without de-noising | 1469.3 | 883.5 | 0.0204 | 0.070 | 313.24 |
| With de-noising | 800.58 | 261.8 | 0.0134 | 0.021 | 170.64 |

TABLE IV. CALCULATION TIME OF FORECASTING METHODS

| Approach | Overall calculation time |
|---|---|
| Algorithm 1 | 0.026917 sec |
| Algorithm 2 and 3 | 0.015497 sec |
| Algorithm 2 and 3 + Wavelet | 0.048648 sec |

## IV. CONCLUSION

In this paper, an error-equation ARARMAX model was selected to identify the behavior of power consumption to forecast the electricity request in the next 24 hours. Two algorithms were presented for estimating the parameters of the model. It was showed that Algorithm 1 provided just a little more accurate load forecasting in comparison with Algorithm 2 whereas it utilized more data for prediction. This makes Algorithm 2 more justifiable to use. Additionally, the prediction error between the real load and the predicted load was minimized due to an accurate short-term load forecasting by utilizing wavelet approaches. The $L_2$ and $L_\infty$ norms error and MAPE, MAE, and RMSE were chosen as criteria for comparison between the represented algorithms. Each algorithm was stimulated by the use of practical data from a typical power utility. Where the data of the first three days were used to train the model and the data of the fourth day were used to test the model by prediction of the load. Simulation results validated that the proposed approach reduced the prediction error which leads to a more precise short-term load forecasting.

REFERENCES

[1] T. Hong, J. Wilson, J. Xie, "Long term probabilistic load forecasting and normalization with hourly information," IEEE Trans. Smart Grid, vol. 5, no. 1, pp. 456–462, 2014.

[2] A. Ashfaq, N. Javaid, A. Mateen, M. Awais, Z.A. Khan. "Short-term load forecasting in smart grids: an intelligent modular approach," Energies, vol. 12, no. 1, 2019.

[3] T. Hong, and S. Fan, "Probabilistic electric load forecasting: a tutorial review," Int. J. Forecast. vol. 32, no., 3, pp. 914–938, 2016.

[4] L.Rui, L. Ting, "Short-term load forecast using maximum overlap discrete wavelet transform and BP neural network based on chaos theory," 2019 Chinese Control And Decision Conference (CCDC), 2019.

[5] M. Imani, H. Ghassemian, "Residential load forecasting using wavelet and collaborative representation transforms," Applied Energy, vol. 253, 2019.

[6] L. Alfieri, and P. De Falco, "Wavelet-Based Decompositions in Probabilistic Load Forecasting," IEEE Trans. Smart Grid, vol. 11, no. 2, pp. 1367 – 1376, 2020.

[7] K. Junhong, M. Jihoon, H. Eenjun, K. Pilsung, "Recurrent inception convolution neural network for multi short-term load forecasting," Energy and buildings, vol. 194, vpp. 328-341, 2019.

[8] H. Son, and C. Kim, "Short-term forecasting of electricity demand for the residential sector using weather and social variables," Resources, conservation and recycling, vol. 123, pp. 200-207, 2017.

[9] N. Vafamand, M. M. Arefi, A. Khayatian, "Nonlinear system identification based on Takagi-Sugeno fuzzy modeling and unscented Kalman filter," ISA Transaction,vol. 74, pp. 134-143, 2018.

[10] F. Ding, D. Meng, J. Dai, Q. Li, A. Alsaedi, and T. Hayat, "Least squares based iterative parameter estimation algorithm for stochastic dynamical systems with ARMA noise using the model equivalence," International Journal of Control, Automation and Systems, vol. 16, no. 2, pp.630-639, 2018.

[11] D. Meng, and F. Ding, "Model equivalence-based identification algorithm for equation-error systems with colored noise," Algorithms, vol. 8, no. 2, pp. 280-291, 2015.

[12] K. Borisagar, R. Thanki, and B. Sedani. "Fourier Transform, Short-Time Fourier Transform, and Wavelet Transform," Speech Enhancement Techniques for Digital Hearing Aids, pp. 63-74. Springer, 2019.

[13] L. Debnath, F. Shah, "Wavelet transform and their applications," NY: Springer, 2015.

[14] M. Zadkarami, A. Karami ghanavati, A. A. Safavi, "Soft Sensor Design for Distillation Columns Using Wavelets and Gaussian Process Regression", 6th Int. Conf. Control, Instrumentation and Automation, 2019.

[15] M. Sadooghi, S. Khadem, "A new performance evaluation scheme for jet engine vibration signal denoising". Mechanical Systems & Signal Processing. vol. 76, pp. 201-212, 2016.

[16] C. Gao, S. Zhou, "Wavelet transform threshold noise reduction methods in the oil pipeline leakage monitoring and positioning system," J. Electronics, vol. 27, no. 3, pp. 405-411, 2010.