Research Article

A Thermal Load Forecasting Algorithm Based on Trajectory Tracking

Guo Ling,[1] Du Weiwei [ID],[1] Yang Jiancheng,[1] Wang Liping,[1] Wan Ping,[1] and Liu Ling[2]

[1]*Army Logistics University, Chong Qing 401311, China*
[2]*Chongqing Vocational Institute of Engineering, Chong Qing 402218, China*

Correspondence should be addressed to Du Weiwei; 317527432@qq.com

The heating load forecast provides a basis for saving heating energy. Considering the nonstationary, nonlinear, and large time-delay characteristics of thermal load, this paper introduces the trajectory tracking stability theory into the field of load forecasting and proposes a heuristic correction that can ensure the convergence of forecast errors and does not depend on the system prediction model algorithm. The Lyapunov method is used to derive an error convergence criterion that has nothing to do with the prediction model, and a heuristic correction algorithm is designed for the predicted value with error divergence trend to ensure the error convergence of the load forecast sequence.

1. Introduction

Thermal load forecasting is based on the historical data of the heating process and to analyze and study the inherent changes in the thermal load demand which can be combined with weather, temperature, or holiday types. The purpose of studying it is to estimate and speculate the load demand in advance. It is the basis of thermal scheduling and energy-saving control [1].

The heating process is a very complex dynamic system, and the relationship between energy and material transmission is highly nonlinear. It is difficult to establish a mathematical model to predict the heating load through a physical model, so most of the current forecasting methods are based on historical data. Based on statistical analysis [2–9], as early as 1984, Werner conducted tests on multiple district heating systems in Sweden and analyzed the main factors affecting the heat load, outdoor temperature, natural wind, and solar radiation. The results of the study showed that 60% of the total heat load of the heating network can be considered. It is affected by the outdoor temperature. The influence of natural wind will increase the heat load by 1% to 4%. The solar radiation heat can reduce the heat load by 1% to 5%. The heat consumption of domestic hot water will be affected by workdays or weekends. Different, the average domestic hot water load accounts for 30% of the total load, and the pipe network heat loss accounts for 5% to 8% of the total load. Wojdyga analyzed the relevant meteorological factors that affect the heat supply of the central heating system, including outdoor temperature, solar radiation intensity, and wind speed. Caldera et al. analyzed the relationship between the building's volume, shape factor, window-to-wall ratio, window transmittance, indoor and outdoor temperature, and other geometric parameters and thermal physical parameters and the building's heating energy consumption. Dotzauer introduced social factors into the heating load prediction and used the outdoor temperature–heating load curve to establish a heating load prediction model. By analyzing outdoor temperature, wind speed, solar radiation intensity, and other meteorological factors that affect heat load, Yetemen et al. established an autoregressive model for heat load forecasting, and Pedersen et al. established a linear regression model for heat load forecasting. Popescu and others further introduced factors such as indoor temperature, circulating water flow, and water supply temperature at the outlet of the heat exchange station into the calculation model of building heat consumption and obtained a multiple regression prediction

model of building heat consumption. Nielsen et al. linked heat supply with meteorological data and date information and established a gray box forecast model for heat supply based on statistics-related theories.

In addition to the abovementioned prediction models based on classical statistical theory, artificial intelligence methods represented by neural networks have also been widely used in heat load forecasting research [10–14]. Aydinalp et al. established a neural network-based building heating energy consumption and domestic hot water energy consumption model by inputting the characteristic parameters of the building, the performance parameters of the heating system, the average indoor temperature, the number of heating degree days, and the economic situation of the family. To predict the energy consumption of residential buildings in Canada, Kwok et al. used a combination of models and neural network predictors to predict building energy and conducted theoretical analysis. Kusiak et al. used humidity and temperature as inputs and used neural network methods to predict building steam load. Duanmu et al. proposed a novel HCLFM model to predict the cooling load of buildings. In addition, in order to meet the actual needs of small sample modeling, the support vector machine method is also used for building energy consumption prediction [15, 16]. Dong et al. used the support vector machine method to predict building energy consumption in tropical areas. Four commercial buildings in Singapore were randomly selected as the research object. The method selected outdoor monthly average temperature, relative humidity, and sunshine as input variables to predict tropical areas. The relative error of prediction of building energy consumption is less than 4%, which verifies the feasibility and adaptability of the support vector machine method to predict building energy consumption in tropical areas. Li et al., respectively, used support vector machines and neural networks to predict building energy consumption and compared the prediction results. These studies apply intelligent algorithms to heating load forecasting, but the research is conducted offline, staying at the laboratory simulation level, and no application examples of energy-saving control in heating engineering are given.

Throughout the research status, the current load forecasting methods are mostly independent or combined use of the following methods: regression analysis [17, 18], time series [19], Kalman filter [20], neural network [21, 22], gray model, wavelet analysis, support vector machine [23], combination method [24], fuzzy processing [25], etc. Among them, regression analysis and time series methods have high requirements for historical data and large forecast errors; neural network convergence speed is slow, generalization ability is poor, and training is easy to overfit; gray system model has better performance when the original data has an exponential growth trend. High prediction accuracy: obviously, if you can combine multiple methods for research and application based on the advantages and disadvantages of various methods, you will get higher prediction accuracy, which also makes the idea of combined prediction that have

developed in recent years. In addition, most of the existing forecasting methods are point forecasting methods, which cannot determine the range of possible fluctuations in the forecast results, and does not reflect the uncertain factors in the system. In these studies, the traditional heating load forecasting method is difficult to adapt to the heating load nonlinearity, resulting in low forecast accuracy, poor accuracy, and even failure, which cannot meet the needs of heating energy saving. The existing intelligent heating load forecasting method is researched for offline, staying at the laboratory simulation level, and no application examples of energy saving in heating engineering are given, and heating load forecasting software is not developed, making the theoretical results unable to be applied to actual projects. . However, with the continuous improvement of load forecasting requirements for heat dispatching and heating energy-saving control, it is necessary to further study the accuracy and accuracy of heating load forecasting, while ensuring the universality and stability of the algorithm.

According to the actual demand of the heating system for the prediction accuracy of the heat load prediction, this paper uses the traditional ARMA time series method [26] for linear load prediction and BP neural network method for nonlinear load prediction according to the characteristics of the heating load. In view of the shortcomings of existing prediction methods, the idea of trajectory tracking control is introduced into the field of load forecasting [27]. A heuristic load forecasting algorithm based on trajectory tracking is proposed, which can ensure the convergence of forecast errors and does not depend on the system prediction model. The final simulation results analyze the performance comparison of several algorithms.

2. Classic Thermal Load Forecasting Method

2.1. Time Series Forecasting Method. The historical data of the heating load is an ordered collection sampled and recorded at a certain time interval, so it is a time series. The time series method is a more mature algorithm developed in the current heating system load forecast. According to the historical data of load, establish a mathematical model describing the change of the heating load with time, establish the expression of load prediction on the basis of this model, and predict the future load.

As a typical representative of the time series model, the ARMA model is expressed as

$$
\begin{aligned}
Y_t = {} & \omega_1 Y_{t-1} + \omega_2 Y_{t-2} + \cdots + \omega_p Y_{t-p} + e_t - \varphi_1 e_{t-1} \\
& - \varphi_2 e_{t-2} - \cdots - \varphi_q e_{t-q},
\end{aligned}
\tag{1}
$$

where Y_t is the value of the stationary time series at time t, e_t is the random interference acting on the time series at time t, the actual parameters ω_1, ω_2, ..., ω_p are the autoregressive coefficients, and the actual parameters φ_1, φ_2, ..., φ_q are the movements average coefficients.

By performing statistical processing on the above model, the autocorrelation coefficient can be obtained as

$$
\begin{cases}
\rho_1 = \dfrac{(1 - \phi_1\theta_1)(\phi_1 - \theta_1)}{(1 + \theta_1^2 - 2\phi_1\theta_1)}, \\[2mm]
\rho_2 = \phi_1\rho_1, \\[2mm]
\ldots \\[2mm]
\rho_k = \phi_1\rho_{k-1}.
\end{cases} \tag{2}
$$

Therefore, the autocorrelation coefficient ρ_1 is a function of θ_1 and ϕ_1, and the autocorrelation function starts from ρ_1 and decays exponentially. If $\phi_1 > 0$, the exponential decay of the autocorrelation function is smooth; if $\phi_1 < 0$, the decay of the autocorrelation function is alternating and oscillates between positive and negative values. The sign of ρ_1 is determined by $\phi_1 - \theta_1$, which determines the direction in which the exponential decay tends to zero. The initial value of its partial autocorrelation function is $\phi_{11} = \rho_1$, which will decay exponentially thereafter. If $\phi_1 > 0$, ϕ_{kk} is smooth exponential decay; if $\phi_1 < 0$, ϕ_{kk} exponential decay is oscillating. The sign of ϕ_{11} is the same as ρ_1, which is also determined by $\phi_1 - \theta_1$.

The prediction of the ARMA model requires that the sequence must be stable and that the influencing factors on the heating load must be basically the same within the time frame studied. If the given sequence is not a stationary sequence, you must preprocess the given sequence to make it stable, and then use the ARMA model to model.

The heat load time series modeling process is shown in Figure 1. The main steps are as follows:

Step 1. Preprocess the sequence to determine whether the time series of heat load is a stationary impure random sequence. If it is a nonstationary sequence, process the sequence to meet the conditions of ARMA model modeling; that is, the processed sequence is a stationary non–white noise sequence.

Step 2. Calculate the values of the sample autocorrelation coefficient (ACF) and sample partial autocorrelation coefficient (PACF) of the observation sequence.

Step 3. According to the sample autocorrelation coefficient and partial autocorrelation coefficient, select the appropriate ARMA model to fit.

Step 4. Estimate the value of unknown parameters in the model.

Step 5. Test the validity of the model. If the fitting model fails the test, go to Step 3; reselect the model and then fit.

Step 6. Model optimization, if the fitting model passes the test, still turn to Step 2, fully consider various possibilities, establish multiple fitting models, and select the optimal model from all the models that pass the test.

Step 7. Use the fitted ARMA model to predict the future trend of the sequence, that is, the load value at the future moment.
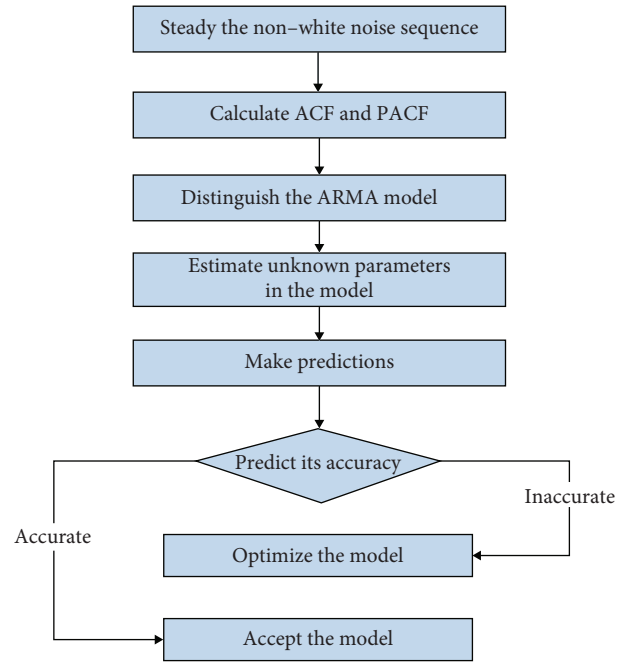

Figure 1: Flowchart of modeling based on time series method.

The advantage of the time series method is that it requires less data, less workload, and faster calculation speed, which can reflect the continuity of the recent changes in load. The disadvantage of the time series method is that the modeling process is more complicated and requires higher theoretical knowledge, and the model has higher requirements for the stability of the original time series. It is only suitable for the prediction of relatively uniform load changes. These factors often lead to the model prediction error being large.

2.2. Neural Network Prediction Method. The artificial neural network method selects the load of the past period as a training sample, constructs a suitable network structure, and uses a certain training algorithm to train the network to meet the accuracy requirements. This neural network is used as a load prediction model in neural network prediction. In general, an optimized BP neural network is selected to establish a prediction model. The main work of modeling is as follows.

2.2.1. Selection of Model Input and Output. The choice of predictive model input will have a great influence on the learning of the network. Regarding the load forecasting model of BP neural network, the difference mainly lies in the selection of input elements and the determination of the number of hidden layers. Since the current neural network's ability to promote prediction is mainly determined through experiments, the selection of input variables is also based on empirical judgment. Generally, the real load data of 24 or 48 points per day in the days before the prediction day is selected as the model input, and the number of nodes in the input layer of the neural network is determined accordingly.

In addition, in terms of the output of the model, the output of the model should be the load of the cycle corresponding to the predicted day.

2.2.2. Determination of the Number of Network Layers.

When each hidden node adopts the S-type compression function shown in the following formula, a hidden layer is enough to realize any decision classification problem, so a three-layer BP neural network is often used in heating load prediction:

$$y = \frac{1}{1 + e^{-x}}. \tag{3}$$

2.2.3. Determination of the Number of Hidden Layers.

To train a three-layer BP network with a single hidden layer, the number of hidden layer nodes can be selected according to the following empirical formula:

$$n_1 = \sqrt{n + m} + a. \tag{4}$$

In the formula, n is the number of input nodes, m is the number of output nodes, and a is a constant between 1 and 10. Because the selection of the number of nodes in each layer in the neural network based on the BP algorithm has a great influence on the performance of the network, the number of nodes in the hidden layer needs to be properly selected. After many times of training the data, and according to the training results, when the hidden layer node is about 15, the results obtained are more reasonable.

2.2.4. Normalization Processing of Data.

Before the neural network is trained and tested, in order to meet the network input and output data requirements and improve the convergence speed, in order to avoid the saturation of neurons, the network input data and actual output data need to be calculated according to equation (5) to [−1, 1]. In the interval, after using ANN prediction, the output layer is converted back to the load value according to equation (6):

$$y = 2 \times \frac{x - x_{\min}}{x_{\max} - x_{\min}} - 1, \tag{5}$$

$$x = 0.5 (y + 1) (x_{\max} - x_{\min}) + x_{\min}. \tag{6}$$

In the formula, x and y are the values before and after normalization of the input sample, and $x_{\min}$ and $x_{\max}$ are the minimum and maximum values of the load in the sample collection, respectively.

2.2.5. Weight Initial Value Setting.

The selection of the initial weight is one of many factors that affect the training of the network tending to have the smallest acceptable error. The initialization of the weight strongly affects the final solution. Generally, the initial weight of the network is selected as $w_{\max} = 1.0$.

The advantages of artificial neural networks are the ability to adapt to a large number of unstructured and nonprecision laws, the characteristics of information memory, autonomous learning, knowledge reasoning, and optimized calculation, as well as strong calculation ability, complex mapping ability, and fault tolerance. And various intelligent processing capabilities, especially its learning and adaptive functions, are not available in other algorithms. The shortcomings are that the number of neural network layers and the number of neurons are mostly determined based on subjective experience, it is difficult to determine the network structure scientifically, the learning speed is slow, and there are local minimum points.

3. Introduction of Trajectory Tracking Stability Theory

3.1. Trajectory Tracking Error Model.

Considering the existence of a moving target in two-dimensional space, its ideal trajectory is shown in curve C in Figure 2 in the inertial coordinate system.

Suppose there is a moving point $P(x, y)^T$ to track the trajectory C, and the angle between the velocity v of point P and the positive direction of the y-axis is θ; that is,

$$\theta = \arctan \frac{\dot{x}}{\dot{y}}. \tag{7}$$

Then, the position motion model of point P can be expressed as

$$\dot{P} = \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} v \sin \theta \\ v \cos \theta \end{bmatrix} = R(\theta) \begin{bmatrix} 0 \\ v \end{bmatrix}. \tag{8}$$

Among them,

$$R(\theta) = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}. \tag{9}$$

At any time, the trajectory coordinate system is defined as follows: taking the desired position of point P (that is, the true position of the target) $P_a = (x_a, y_a)^T$ as the origin, the y_a-axis is along the tangent of P_a, and the x_a-axis is perpendicular to y_a and pointing to the right, as shown in Figure 3.

Then, the rotation angle θ_a of the trajectory coordinate system relative to the inertial coordinate system is

$$\theta_a = \arctan \frac{\dot{x}_a}{\dot{y}_a}. \tag{10}$$

Referring to equation (8), the motion model of the desired position P_a can be expressed as

$$\dot{P}_a = \begin{bmatrix} \dot{x}_a \\ \dot{y}_a \end{bmatrix} = \begin{bmatrix} v_a \sin \theta_a \\ v_a \cos \theta_a \end{bmatrix} = R(\theta_a) \begin{bmatrix} 0 \\ v_a \end{bmatrix}. \tag{11}$$

Among them, v_a is the desired speed of point P (that is, the true speed of the target), and the direction is consistent with the y_a-axis, and $R(\theta_a)$ and $R(\theta)$ have the same form.

Therefore, the tracking error ε of P and P_a in the trajectory coordinate system is
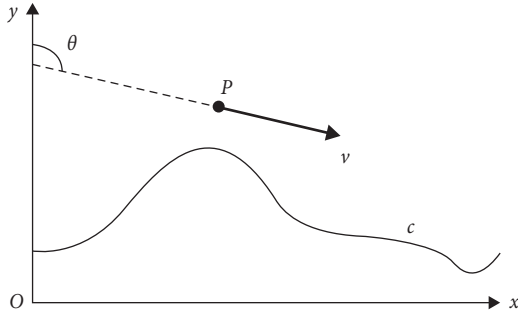
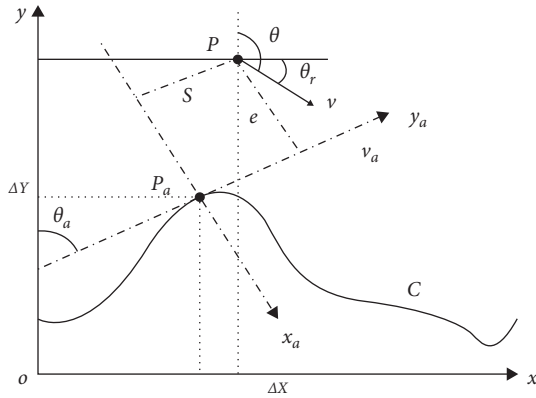FIGURE 2: Schematic diagram of target trajectory tracking.

FIGURE 3: Schematic diagram of coordinated system and tracking error.

$$\varepsilon = \begin{bmatrix} e \\ s \end{bmatrix} = \begin{bmatrix} \Delta X \cos \theta_a - \Delta \sin \theta_a \\ \Delta X \sin \theta_a + \Delta Y \cos \theta_a \end{bmatrix}$$
$$= \begin{bmatrix} \cos \theta_a & \sin \theta_a \\ -\sin \theta_a & \cos \theta_a \end{bmatrix}^T \begin{bmatrix} \Delta X \\ \Delta Y \end{bmatrix} = R^T (\theta_a)(P - P_a). \tag{12}$$

Among them, e and s are the lateral and forward tracking errors of P and P_a in the trajectory coordinate system, respectively, and ΔX and ΔY are the lateral and longitudinal tracking errors of P and P_a in the inertial coordinate system, respectively.

From equation (12), we can know that $\varepsilon^T \varepsilon = (P - P_a)^T RR^T (P - P_a) = \|P - P_a\|^2$. Obviously, there is only if $\varepsilon \longrightarrow 0$, so the fundamental goal of P to track the ideal trajectory is to make $\varepsilon \longrightarrow 0$.

3.2. Trajectory Tracking Stability Criterion. In control theory, the Lyapunov criterion [28] described in Lemma 1 is commonly used to judge the stability of the tracking system and to check the error convergence trend.

Lemma 1. *For continuous nonlinear time-varying systems $\dot{\chi} = f(\chi)$, $t \geq 0$, if a scalar function $V(\chi)$ with continuous first-order partial derivatives for χ can be constructed, $V(0) = 0$ satisfies the condition for all nonzero state points χ in the state space $\mathfrak{R}$:* ① *$V(\chi)$ is positive definite;* ②

$\dot{V}(\chi) = (dV(\chi)/dt)$ *is negative definite; and* ③*when $\|\chi\| \longrightarrow \infty$, then $V(\chi) \longrightarrow \infty$. Then, the system equilibrium state $\chi = 0$ is asymptotically stable in a large range; that is, χ asymptotically converges to zero.*

Based on the above lemma to analyze the tracking error convergence, the key is to construct the Lyapunov candidate function $V(\chi)$, and different Lyapunov candidate functions may obtain different stability criteria. Considering that the main purpose of this article is to verify the applicability of the trajectory tracking stability theory in load forecasting, no special research has been conducted on the selection of Lyapunov candidate functions, so the most commonly used Lyapunov function type is directly used:

$$V = \frac{1}{2}\varepsilon^T \varepsilon. \tag{13}$$

Then, V derivates t:

$$\begin{aligned} \dot{V} &= \varepsilon^T \dot{\varepsilon} = \varepsilon^T \left[\dot{R}^T (\theta_a)(P - P_a) + R^T (\theta_a)\left(\dot{P} - t\dot{P}_a\right) \right] \\ &= \varepsilon^T \left[\begin{bmatrix} 0 & -\dot{\theta}_a \\ \dot{\theta}_a & 0 \end{bmatrix} R^T (\theta_a)(P - P_a) + R^T (\theta_a)\left(\dot{P} - t\dot{P}_a\right) \right] \\ &= \varepsilon^T \begin{bmatrix} 0 & -\dot{\theta}_a \\ \dot{\theta}_a & 0 \end{bmatrix} \varepsilon + \varepsilon^T R^T (\theta_a)\left(\dot{P} - t\dot{P}_a\right) \\ &= \varepsilon^T R^T (\theta_a)\left(\dot{P} - t\dot{P}_a\right). \end{aligned} \tag{14}$$

Therefore, if equation (14) is negatively set at $\varepsilon \neq 0$, the tracking error of the moving point P to the target trajectory will converge to zero asymptotically, which can ensure the stability of trajectory tracking.

If the actual load sequence is regarded as a trajectory on the time axis, high-precision load prediction is to realize the high-precision tracking of the trajectory composed of predicted points and the actual load trajectory. Therefore, the convergence analysis of the load forecast error is essentially equivalent to the convergence analysis of the tracking error of the predicted point.

4. Process of Heat Load Prediction Algorithm Based on Trajectory Tracking

4.1. Condition of Error Convergence for Heating Load Forecast

Theorem 1. *At any time, if $\theta > \theta_a$ holds when $\Delta Y > 0$, or $\theta < \theta_a$ holds when $\Delta Y < 0$, the prediction error will converge to zero asymptotically.*

In a two-dimensional coordinate system with the x-axis as the time and the y-axis as the load value, the load time series can be regarded as a trajectory. If moving point $P = (x, y)^T$ represents the predicted load value y at time x and $P_a = (x_a, y_a)^T$ represents the true load value y_a at time x, since both have the same time axis, the lateral error of point P and its expected position P_a in the inertial coordinate system must meet $\Delta X = 0$, as shown in Figure 4.

Therefore, we can further expand equation (14) to obtain

$$\dot{V} = \begin{bmatrix} e & s \end{bmatrix} \begin{bmatrix} \cos\theta_a & -\sin\theta_a \\ \sin\theta_a & \cos\theta_a \end{bmatrix} \begin{bmatrix} v\sin\theta - v_a\sin\theta_a \\ v\cos\theta - v_a\cos\theta_a \end{bmatrix}$$
$$= s\left[v\cos(\theta - \theta_a) - v_a\right] + ev\sin(\theta - \theta_a) \quad (15)$$
$$= sv\cos\theta_r + ev\sin\theta_r - sv_a.$$

When $\Delta Y = 0$, it is known from $\Delta X = 0$ and equation (12) that $\varepsilon = 0$; that is, the prediction error converges.

When $\Delta Y \neq 0$, considering that P and P_a have the same time axis, there are only two possible geometric relationships; that is, point P is directly above or below point P_a, so we can solve the prediction error according to the following two cases. The sufficient condition for the convergence of the prediction error can be expressed as follows:

Case 1. When P is directly above P_a, that is, $\Delta Y > 0$.

From the trajectory coordinate system in Figure 4, the following geometric features can be obtained intuitively:

① If $\theta_a < (\pi/2)$, then $e < 0$, $s > 0$, as shown in Figure 4(a).

② If $\theta_a > (\pi/2)$, then $e < 0$, $s < 0$, as shown in Figure 4(b).

③ For any θ_a, there are

$$\cos\theta_a = \frac{s}{\sqrt{s^2 + e^2}},$$
$$\sin\theta_a = \frac{-e}{\sqrt{s^2 + e^2}}. \quad (16)$$

Then, by equation (15),

$$\dot{V} = sv\cos\theta_r + ev\sin\theta_r - sv_a$$
$$= v\sqrt{s^2 + e^2}\left[\frac{s\cos\theta_r}{\sqrt{s^2 + e^2}} - \frac{-e\sin\theta_r}{\sqrt{s^2 + e^2}} - \frac{sv_a}{v\sqrt{s^2 + e^2}}\right] \quad (17)$$
$$= v\sqrt{s^2 + e^2}\left[\cos\theta - \frac{sv_a}{v\sqrt{s^2 + e^2}}\right].$$

In the actual load curve, the real value sequence adds a data point every time ΔT, and the predicted value sequence also adds a data point every time ΔT, so v = van and $v > 0$; that is, in a two-dimensional coordinate system with the x-axis as the time axis, the actual load value and the predicted value extend toward the positive direction of the x-axis at the same speed. From equation (17) and Lemma 1, if and only if $\cos\theta < (sv_a/v\sqrt{s^2 + e^2}) = \cos\theta_a$, that is, when $\theta > \theta_a$, $\dot{V} < 0$, the prediction error converges to zero.

Case 2. When P is directly below P_a, that is, $\Delta Y < 0$.

From the track coordinate system of Figure 4, the following geometric features can also be obtained intuitively:

① If $\theta_a < \pi/2$, then $e > 0$, $s < 0$, as shown in Figure 4(c).

② If $\theta_a > \pi/2$, then $e > 0$, $s > 0$, as shown in Figure 4(d).

③ For any θ_a, there are

$$\cos\theta_a = \frac{-s}{\sqrt{s^2 + e^2}},$$
$$\sin\theta_a = \frac{e}{\sqrt{s^2 + e^2}}. \quad (18)$$

Then, by equation (15),

$$\dot{V} = sv\cos\theta_r + ev\sin\theta_r - sv_a$$
$$= v\sqrt{s^2 + e^2}\left[\frac{s\cos\theta_r}{\sqrt{s^2 + e^2}} + \frac{e\sin\theta_r}{\sqrt{s^2 + e^2}} - \frac{sv_a}{v\sqrt{s^2 + e^2}}\right] \quad (19)$$
$$= v\sqrt{s^2 + e^2}\left[\cos\theta + \frac{sv_a}{v\sqrt{s^2 + e^2}}\right].$$

An analysis similar to Case 1 can be made. Therefore, from equation (19) and Lemma 1, it can also be seen that, if and only if $\cos\theta > (-sv_a/v\sqrt{s^2 + e^2}) = \cos\theta_a$, that is, when $\theta < \theta_a$, $\dot{V} < 0$, the prediction error converges to zero.

Noticed that the above analysis process is completely derived from trajectory tracking and its stability theory and has nothing to do with the specific prediction model used. From this, we have obtained the conclusions about the error convergence conditions that have nothing to do with the prediction model in this paper.

4.2. Error Correction Strategy for Heating Load Forecast. At any sampling time x, the historical predicted value $y(x)$ and the real values $y_a(x)$ and $y_a(x - 1)$ are known, and the predicted value $y(x + 1)$ at the next time can be calculated by any existing prediction model (defined as the preliminary prediction model); then the value Δy, the angle θ between the ray from point $(x, y(x))$ to point $(x + 1, y(x + 1))$ and the positive direction of the y-axis, and the angle θ_a between the ray from point $(x - 1, y_a(x - 1))$ to point $(x, y_a(x))$ and the positive direction of the y-axis can be calculated. Then, according to Theorem 1, it can be judged whether the prediction error convergence condition is satisfied.

Obviously, if the error convergence condition is satisfied, it indicates that the predicted value $y(x + 1)$ solved by the preliminary prediction model has a tendency to converge to zero, and no correction is needed at this time; if the error convergence condition is not satisfied, it indicates that the predicted value $y(x + 1)$ solved by the preliminary prediction model is implicit. The trend of error divergence should be corrected at this time.

For the predicted value with error divergence trend, this paper further proposes an adaptive correction algorithm for the predicted value. The basic idea of the algorithm is to calculate ΔY and $\tan\theta_a$ from known conditions and then use the monotonic feature of the tangent function to adjust the value of $\tan\theta$ based on Theorem 1, so that the predicted value $y(x + 1)$ at any time meets the error convergence condition. The method of recalculating the predicted value $y(x + 1)$ is as follows:

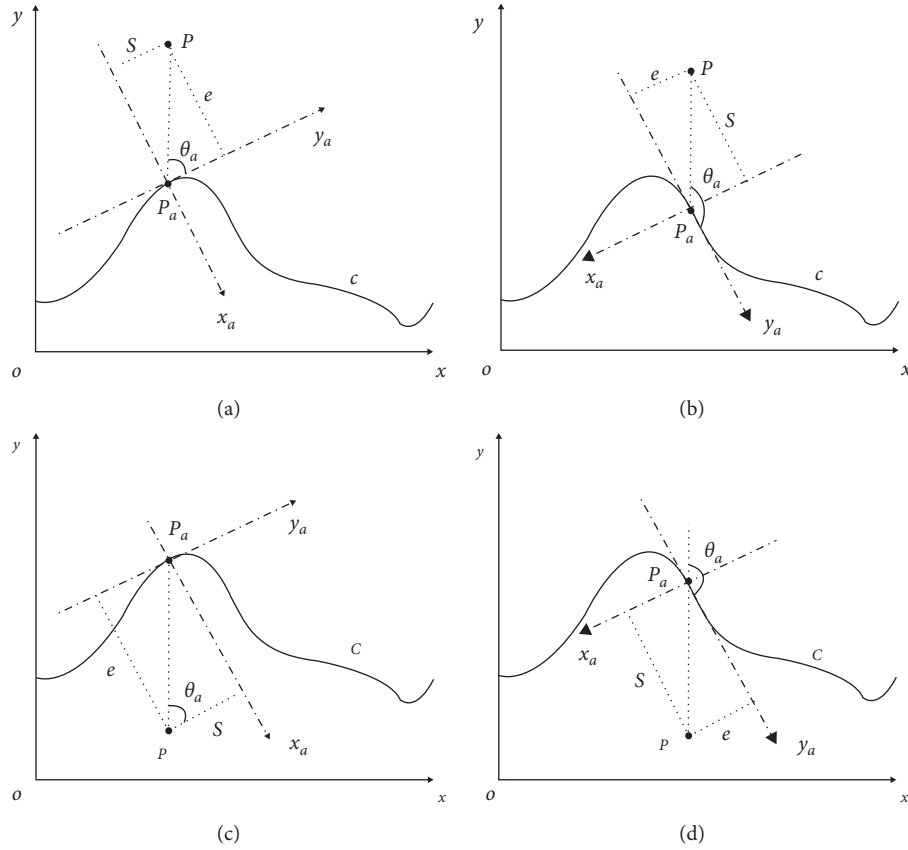Figure 4: Convergence analysis of forecasting error. (a) $\Delta Y > 0$, $\theta_a < \pi/2$; (b) $\Delta Y > 0$, $\theta_a > \pi/2$; (c) $\Delta Y < 0$, $\theta_a < \pi/2$; and (d) $\Delta Y < 0$, $\theta_a > \pi/2$.

① If $\Delta Y > 0$ and $y_a(x) - y_a(x-1) = 0$, then

$$y(x+1) = k_1 y(x), \quad 0 < k_1 < 1. \tag{20}$$

② If $\Delta Y > 0$ and $y_a(x) - y_a(x-1) > 0$, then

$$y(x+1) = y(x) + k_2[y_a(x) - y_a(x-1)], \quad 0 < k_2 < 1. \tag{21}$$

③ If $\Delta Y > 0$ and $y_a(x) - y_a(x-1) < 0$, then

$$y(x+1) = y(x) + k_3[y_a(x) - y_a(x-1)], \quad k_3 > 1. \tag{22}$$

④ If $\Delta Y < 0$ and $y_a(x) - y_a(x-1) = 0$, then

$$y(x+1) = k_4 y(x), \quad k_4 > 1. \tag{23}$$

⑤ If $\Delta Y < 0$ and $y_a(x) - y_a(x-1) > 0$, then

$$y(x+1) = y(x) + k_5[y_a(x) - y_a(x-1)], \quad k_5 > 1. \tag{24}$$

⑥ If $\Delta Y < 0$ and $y_a(x) - y_a(x-1) < 0$, then

$$y(x+1) = y(x) + k_6[y_a(x) - y_a(x-1)], \quad 0 < k_6 < 1. \tag{25}$$

The correction coefficients k_1–k_6 are selected by the user. The following theoretically proves the effectiveness of this revised algorithm.

Proof. Take $X > 0$ as an example.

When $y_a(x) - y_a(x-1)$, then $\theta_a = \pi/2$. From equations (7) and (20), we can get

$$\tan \theta = \frac{1}{y(x+1) - y(x)} = \frac{1}{(k_1 - 1)y(x)}. \tag{26}$$

Since $0 < k_1 < 1$ and $y(x) > 0$, then $\tan \theta < 0$ and $\theta > \pi/2$, so $\theta > \theta_a$ holds. When $y_a(x) - y_a(x-1) > 0$, from equations (7) and (10), we can find that equation (7) is equivalent to

$$\tan \theta_a = \frac{1}{y_a(x) - y_a(x-1)} = \frac{k_2}{y(x+1) - y(x)} = k_2 \tan \theta. \tag{27}$$

Since $0 < k_2 < 1$ and $\tan\theta_a > 0$, according to the monotonic characteristics of tangent function, $\theta > \theta_a$ was established.

When $y_a(x) - y_a(x-1) < 0$, from equations (7) and (10), we can find that equation (22) is equivalent to

$$\tan\theta_a = \frac{1}{y_a(x) - y_a(x-1)} = \frac{k_3}{y(x+1) - y(x)} = k_3 \tan\theta. \tag{28}$$

Since $k_3 > 1$ and $\tan\theta_a < 0$, so $\tan\theta_a < \tan\theta < 0$, according to the monotonic characteristics of tangent function, $\theta > \theta_a$ was established.

In summary, $\Delta Y > 0$ holds when $\theta > \theta_a$, and the revised predicted value $y(x+1)$ meets the error convergence condition of Theorem 1. Similarly, it can be proved that $\Delta Y < 0$, so the correction algorithm is theoretically effective. □

4.3. Full Description Based on Trajectory Tracking. According to the previous analysis and derivation process, the load forecasting algorithm proposed in this paper is described as follows:

Step 1. Choose an existing prediction model (such as ARMA or BPNN model) to establish a preliminary prediction model of load time series.

Step 2. At time x, use the preliminary prediction model to calculate the preliminary prediction value $y(x+1)$ at time.

Step 3. From the predicted values $y(x)$ and $y(x-1)$ at times x and $x+1$, use equation (7) to calculate the angle θ between the ray from point $(x, y(x))$ to point $(x+1, y(x+1))$ and the positive direction of the y-axis.

Step 4. From the predicted values $y_a(x)$ and $y_a(x-1)$ at times x and $x-1$, use equation (9) to calculate the angle θ_a between the ray from point $(x-1, y_a(x-1))$ to point $(x, y_a(x))$ and the positive direction of the y-axis.

Step 5. From the predicted value $y(x)$ and the real value $y_a(x)$ at time x, we can calculate $\Delta Y = y(x) - y_a(x)$.

Step 6. According to Theorem 1 of this paper, judge whether ΔY, θ, and θ_a meet the convergence condition of the prediction error. If it is satisfied, the predicted value $y(x+1)$ solved by the preliminary prediction model is considered valid; otherwise, go to Step 7.

Step 7. According to the prediction value correction algorithm proposed in this paper, solve the load prediction value $y(x+1)$ at time $x+1$ again.

From the previous analysis and algorithm description, this method is actually a heuristic correction algorithm for the existing prediction model. The heuristic rules of the algorithm are derived based on the theory of trajectory tracking stability and have nothing to do with the specific prediction model selected, so it is a heuristic correction algorithm that can ensure the convergence of prediction errors and does not depend on the system

model. In addition, during the iterative process along the sampling time, the algorithm feeds back the inherent laws contained in the historical prediction sequence to the future prediction, which improves the prediction accuracy.

5. Simulation and Result Analysis

It is of no practical significance to compare this method with other autonomous prediction algorithms alone. The effectiveness of this method is reflected in the revision and improvement of existing prediction models. In the experimental part, the classic autoregressive moving average (ARMA) model [29] and Back Propagation Neural Network (BPNN) model [30] are taken as examples to compare and analyze the correction effect of the method under different prediction models and prediction field of view conditions. Use the load sample data GEFCom2012 published by the Global Energy Forecasting Competition to simulate on the MATLAB platform. In all experiments, the parameters of the modified algorithm in this paper are selected as $k_1 = k_2 = k_6 = 0.9$, $k_3 = k_4 = k_5 = 1.1$.

5.1. Comparison Experiment with the ARMA Prediction Model. Based on the actual measurement of a certain residential area in Jinan on December 9, 2018, the load value after a single prediction field of view of 0.5 h was predicted and experimentally measured, using a separate ARMA model (ARMA method) and based on the ARMA model. The error-corrected method (ARMA + EC method) obtained the forecast and measured results at 48 o'clock on the day, and a comparative analysis was conducted.

Based on the historical load value of 120 points totaling 60 hours before the current day as the data source, an ARMA prediction model was established. The classic (p, q)-order ARMA prediction model can be expressed by equation (1).

The main work of modeling with ARMA lies in model order determination and coefficient estimation. Firstly, use Autocorrelation Function (ACF) to analyze and remove the fixed trend items in the original data source. Then, the Bayesian Information Criterion (BIC), Yule–Walker, and Newton–Raphson algorithms are used to determine the model order, autoregressive coefficient, and moving average coefficient, respectively. In the experiment, the BIC values corresponding to each order ARMA candidate model are shown in Table 1.

It can be seen that the ARMA (5, 5) model will get the minimum BIC value of 5.7496, so ARMA (5, 5) is used as the prediction model in the experiment. At the same time, the order coefficients of the ARMA (5, 5) prediction model are $\omega_1 = 0.9189$, $\omega_2 = -0.2617$, $\omega_3 = 0.3394$, $\omega_4 = 0.7065$, $\omega_5 = 0.2710$, $\varphi_1 = -0.1748$, $\varphi_2 = -0.3256$, $\varphi_3 = 0.0260$, $\varphi_4 = -0.5668$, and $\varphi_5 = -0.7124$.

Two methods are used to obtain the forecast result of 48 o'clock on the day, as shown in Figure 5. Analysis of the data in the figure shows the following:

① At 8:00, the predicted value is greater than the true value, that is, $\Delta Y > 0$; the ray connected by the true

Table 1: BIC values for ARMA model candidate.

Order	$q = 1$	$q = 2$	$q = 3$	$q = 4$	$q = 5$	$q = 6$
$p = 1$	5.95	5.98	5.97	5.99	5.93	5.95
$p = 2$	5.91	5.95	5.96	6.05	5.95	5.94
$p = 3$	5.97	5.95	5.99	6.02	5.87	6.00
$p = 4$	5.96	5.99	5.86	5.89	5.77	5.81
$p = 5$	5.99	6.04	6.01	5.96	**5.75**	5.77
$p = 6$	6.02	6.03	6.00	5.95	5.79	5.80

value from 7:30 to 8:00 forms an angle θ_a with the positive direction of the y-axis; from 8:00 to 8:30, the ray connected by the predicted value forms an angle θ with the positive direction of the y-axis. Obviously, at this time, $\Delta Y > 0$ and $\theta > \theta_a$. Theorem 1 shows that the load value predicted by the classic ARMA model at 8:30 already has the error convergence trend, and the method in this paper does not need to be corrected, so the prediction results of the two methods are completely equal.

② At 19:30, the predicted value is less than the true value; that is, $\Delta Y < 0$. The rays formed by the true value from 19:00 to 19:30 form an angle θ_a with the positive direction of the y-axis, from 19:30 to 20:00 The rays formed by the predicted values form an angle θ with the positive direction of the y-axis. Obviously, $\Delta Y < 0$ and $\theta > \theta_a$ at this time. Theorem 1 shows that the load value predicted by the classic ARMA model at 20:00 does not have the error convergence trend, and it needs to be recalculated using the revised algorithm in this paper, so as to obtain the load forecast value at 20:00 with less error.

Similar predictions can be made at other times. The results show that, in the stage where the error of the classic ARMA prediction model is gradually increasing, the method in this paper can effectively evaluate the convergence of the prediction error and make adaptive corrections to ensure the convergence of the prediction error and improve the prediction accuracy.

5.2. Comparison Experiment with the BPNN Prediction Model.
Taking December 21, 2018 as the forecasting comparison date, the load value after a single forecast horizon of 0.5 h is predicted, and the separate BPNN model (BPNN method) and the error correction method in this paper (BPNN + EC method) based on the BPNN model are, respectively, adopted. We can get the forecast result at 48 o'clock on the forecast day and make a comparative analysis.

In the past 60 days, a total of 2880 historical load values were used as data sources to establish a BPNN prediction model. As a very widely used neural network prediction model, the main job of BPNN modeling is to determine the number of nodes in each layer. In the experiment of this paper, the input layer is designed as 8 nodes, which correspond to the 4 load data closest to the current time and the
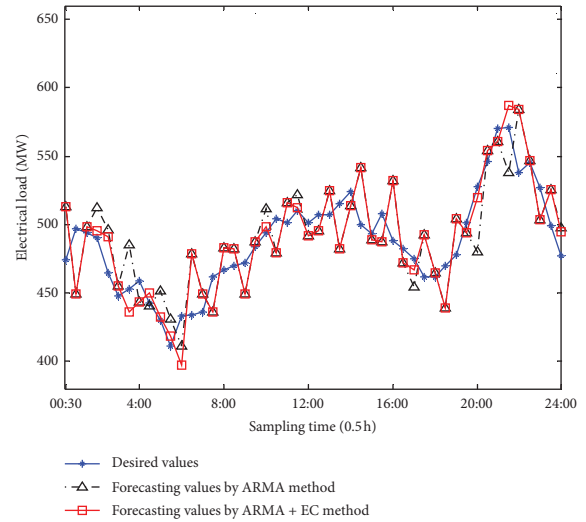

Figure 5: Forecasting result comparison between ARMA method and ARMA + EC method.

load data at the same time in the last 4 days; and according to Kolmogorov's theorem and empirical knowledge, the hidden layer is selected as 17 intermediate nodes; Bayesian regularization (BR) and Levenberg–Marquardt (LM) rules are used to learn and train the network model, and all input data during training are normalized to [0, 1].

Two methods are used to obtain the comparison result between the forecast and actual measurement at 48 o'clock on the day, as shown in Figure 6. The data in the figure is compared and analyzed using the same method as the comparison experiment of the ARMA prediction model. The results show that for the BPNN prediction model, the correction method can also obtain a similar improvement effect; that is, when the prediction value of the BPNN model shows an error divergence trend, this method can effectively evaluate the convergence of the prediction error and perform adaptive correction to ensure the error convergence and improve prediction accuracy.

5.3. Comparative Experiments in Different Prediction Fields.
Starting from November 2nd, 2018, 60 consecutive days as the comparison period for forecasting and measuring, the load values after different forecast nodes 0.5 h, 1 h, 2 h, 4 h, and 8 h are predicted, using separate ARMA method and ARMA + EC in this paper The EC method, as well as the separate BPNN method and the BPNN + EC method in this article, obtained the comparison results of 60 days of multipoint prediction and actual measurement for comparative analysis.

In order to verify the improvement effect of the algorithm in this paper, the most commonly used statistical indicators Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are selected to evaluate the prediction accuracy, and two new indicators the Reduction Rate of MAE (RRMAE) and the Reduction Rate of RMSE
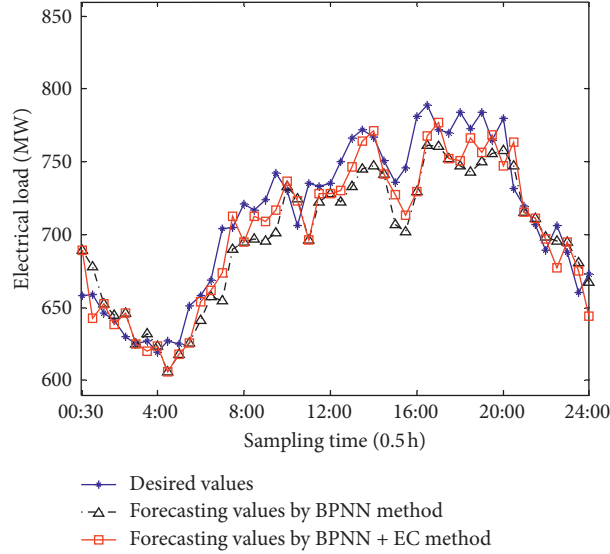
FIGURE 6: Forecasting result comparison between BPNN method and BPNN + EC method.

TABLE 2: Comparison between ARMA and ARMA + EC methods with multiple forecasting nodes.

Prediction nodes	MAE (MW)		RR$_{MAE}$ (%)	RMSE (MW)		RR$_{RMSE}$ (%)
	ARMA	ARMA + EC		ARMA	ARMA + EC	
0.5	22.46	18.72	**16.68**	25.33	21.48	**15.19**
1	23.39	19.56	16.35	26.55	22.59	14.89
2	25.40	21.41	15.73	28.81	24.71	14.23
4	29.22	24.98	14.50	33.48	29.12	13.00
8	36.99	32.40	12.39	43.16	38.58	10.60

TABLE 3: Comparison between BPNN and BPNN + EC method with multiple forecasting nodes.

Prediction nodes (h)	MAE (MW)		RR$_{MAE}$ (%)	RMSE (MW)		RR$_{RMSE}$ (%)
	BPNN	BPNN + EC		BPNN	ARMA + EC	
0.5	19.39	15.84	**18.31**	23.28	19.49	**16.30**
1	20.19	16.53	18.12	24.27	20.38	16.05
2	21.77	17.91	17.72	26.31	22.22	15.55
4	24.95	20.82	16.57	30.30	25.90	14.52
8	31.31	26.61	15.00	38.35	33.48	12.68

(RRRMSE) are defined to evaluate the degree of correction of the prediction error of this method. The mathematical expressions of the indicators are as follows:

$$
\begin{aligned}
\text{MAE} &= \frac{1}{N} \sum_{t=1}^{N} |y(t) - y_a(t)|, \\
\text{RMSE} &= \sqrt{\frac{\sum_{t=1}^{N} (y(t) - y_a(t))^2}{N}}, \\
\text{RR}_{\text{MAE}} &= \frac{\text{MAE}_{\text{typical}} - \text{MAE}_{\text{proposed}}}{\text{MAE}_{\text{typical}}} \times 100\%, \\
\text{RR}_{\text{RMSE}} &= \frac{\text{RMSE}_{\text{typical}} - \text{RMSE}_{\text{proposed}}}{\text{RMSE}_{\text{typical}}} \times 100\%.
\end{aligned}
\tag{29}
$$

N is the total prediction points, $y(t)$ and $y_a(t)$ are the predicted value and the true value at the time of t, MAE$_{\text{typical}}$, MAE$_{\text{typical}}$, and RMSE$_{\text{typical}}$ are the MAE and RMSE values of the existing prediction model, and MAE$_{\text{proposed}}$ and RMSE$_{\text{proposed}}$ are the MAE and RMSE values of the model modified by the method in this paper.

The above two experimental methods were used to establish ARMA and BPNN prediction models for different prediction nodes (0.5 h, 1 h, 2 h, 4 h, and 8 h). For different prediction nodes, the ARMA method and the ARMA + EC method are used to obtain the comparison value between the multipoint prediction and the actual measurement for 60 consecutive days from November 2, 2018 to February 1, 2019; BPNN method and BPNN + EC are used methods to obtain the comparison value between the multipoint prediction and the actual measurement for 60 consecutive days

in the same period. Calculating the statistical indicators, the results are shown in Tables 2 and 3.

Experimental results show the following:

① Compared with the single ARMA and BPNN prediction models, the prediction error of the method in this paper is significantly reduced. Compared with the ARMA model, the maximum reduction rates of the two indexes of MAE and RMSE reached 16.6759% and 15.1931%. Compared with the BPNN model, the maximum reduction rates of the two indexes of MAE and RMSE reached 18.31149% and 16.2956%.

② For different prediction models and prediction nodes, the method is independent of the selected system model, and the improvement of prediction performance is both effective and robust. In all cases, MAE and RMSE both have a performance improvement of at least 10%.

③ As the prediction node increases, the load value fluctuation between two adjacent samples increases, resulting in the corresponding increase in the MAE and RMSE indicators of the two algorithms; that is, the prediction accuracy decreases with the increase of the prediction field of view, so this method is more suitable for short-term load forecasting

6. Summary

Starting from the historical load sequence to study how to reduce the prediction error is an important means to improve the accuracy of heating load prediction. This paper proposes a correction method for load prediction error based on trajectory tracking. This method is completely based on the theory of trajectory tracking stability. There are no special requirements for the prediction model itself; that is, it has nothing to do with the system model, which can theoretically ensure error convergence, and simulation experiments also show its effectiveness.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

References

[1] J.-b. Zhang, L. Guo, Z.-x. Wang et al., "Thermal load forecasting model of heating system based on genetic algorithm optimization BP neural network," *Journal of Sichuan Ordnance*, vol. 35, no. 4, pp. 152–156, 2014.

[2] S. Werner, *The Heat Load in District Heating System*, Chalmers University of Technology, Swenden, Europe, 1984.

[3] K. Wojdyga, "An influence of weather conditions on heat demand in district heating systems," *Energy and Buildings*, vol. 40, no. 11, pp. 2009–2014, 2008.

[4] M. Caldera, S. P. Corgnati, and M. Filippi, "Energy demand for space heating through a statistical approach: application to residential buildings," *Energy and Buildings*, vol. 40, no. 10, pp. 1972–1983, 2008.

[5] E. Dotzauer, "Simple model for prediction of loads in district-heating systems," *Applied Energy*, vol. 73, no. 3-4, pp. 277–284, 2009.

[6] O. Yetemen and T. Yalcin, "Climatic parameters and evaluation of energy consumption of the Afyon geothermal district heating system, Afyon, Turkey," *Renewable Energy*, vol. 34, no. 3, pp. 706–710, 2009.

[7] L. Pedersen, J. Stang, and R. Ulseth, "Load prediction method for heat and electricity demand in buildings for the purpose of planning for mixed energy distribution systems," *Energy and Buildings*, vol. 40, no. 7, pp. 1124–1134, 2008.

[8] D. Popescu, F. Ungureanu, and A. Hernández-Guerrero, "Simulation models for the analysis of space heat consumption of buildings," *Energy*, vol. 34, no. 10, pp. 1447–1453, 2009.

[9] H. A. Nielsen and H. Madsen, "Modelling the heat consumption in district heating systems using a grey-box approach," *Energy and Buildings*, vol. 38, no. 1, pp. 63–71, 2006.

[10] M. Aydinalp, V. I. Ugursal, and A. S. Fung, "Modeling of the space and domestic hot-water heating energy-consumption in the residential sector using neural networks," *Applied Energy*, vol. 79, no. 2, pp. 159–178, 2004.

[11] S. S. K. Kwok, R. K. K. Yuen, and E. W. M. Lee, "An intelligent approach to assessing the effect of building occupancy on building cooling load prediction," *Building and Environment*, vol. 46, no. 8, pp. 1681–1690, 2011.

[12] A. Kusiak, M. Li, and Z. Zhang, "A data-driven approach for steam load prediction in buildings," *Applied Energy*, vol. 87, no. 3, pp. 925–933, 2010.

[13] L. Duanmu and Z. Wang, "A simplified method to predict hourly building cooling load for urban energy planning," *Energy Build*, vol. 58, pp. 281–291, 2012.

[14] M. P. Kody, S. Akshay, J. C. Wesley, and F. E. Thomas, "Heating, cooling, and electrical load forecasting for a large-scale district energy system," *Energy*, vol. 74, pp. 877–885, 2014.

[15] B. Dong, C. Cao, and S. E. Lee, "Applying support vector machines to predict building energy consumption in tropical region," *Energy and Buildings*, vol. 37, no. 5, pp. 545–553, 2005.

[16] Q. Li, Q. Meng, J. Cai, H. Yoshino, and A. Mochida, "Predicting hourly cooling load in the building: a comparison of support vector machine and different artificial neural networks," *Energy Conversion and Management*, vol. 50, no. 1, pp. 90–96, 2009.

[17] S. Li and Z. Jiang, "Establishment and Analysis of regression Model for heating load," *District Heating*, no. 6, pp. 122–129, 2018.

[18] E. Ceperic, V. Ceperic, and A. Baric, "A strategy for short-term load forecasting by support vector regression machines," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4356–4364, 2013.

[19] D. P. Patel, A. Tiwari, and V. Dubey, "An analysis of short term load forecasting by using time series analysis," *International Journal of Research in Computer and Communication Technology*, vol. 2, pp. 47–52, 2013.

[20] R. Shankar, K. Chatterjee, K. Chatterjee, and T. K. Chatterjee, "A very short - term load forecasting using Kalman filter for load Frequency control with economic load dispatch," *Journal*

of Engineering Science and Technology Review, vol. 5, no. 1, pp. 97–103, 2012.

[21] P. Liu, Li Rui, and Y. Wang, "Prediction of district heating load based on grey neural network model," *Heating Ventilating & Air Conditioning*, vol. 49, no. 5, pp. 124–128, 2019.

[22] P.-C. Chang, C.-Y. Fan, and J.-J. Lin, "Monthly electricity demand forecasting based on a weighted evolving fuzzy neural network approach," *International Journal of Electrical Power & Energy Systems*, vol. 33, no. 1, pp. 17–27, 2011.

[23] J. Zhang, Q. Tian, and M. Wang, "Heating load prediction based on support vector regression machine with parameters optimized by genetic algorithm," *Heating Ventilating & Air Conditioning*, vol. 47, no. 2, pp. 104–108, 2017.

[24] X. Yu, J. Gu, C. Qi et al., "Comparison of several centralized heating load forecasting models," *Heating Ventilating & Air Conditioning*, vol. 49, no. 2, pp. 96–99, 2019.

[25] A. Khosravi, S. Nahavandi, D. Creighton, and D. Srinivasan, "Interval type-2 fuzzy logic systems for load forecasting: a comparative study," *IEEE Transactions on Power Systems*, vol. 27, no. 3, pp. 1274–1282, 2012.

[26] H. Yang, Z. Pan, and W. Bai, "Review of time series prediction methods," *Computer Science*, vol. 46, no. 1, pp. 21–28, 2019.

[27] Y. D. Song, Z. X. Shen, D. L. Dai et al., "Short-term load forecasting in electrical power systems via trajectory tracking and error correction approach," *Renewable Sustainable Energy*, vol. 6, 2014.

[28] J. Y. Yi, K. M. Kim, J. Lee, and M. S. Oh, "Exergy analysis for utilizing latent energy of thermal energy storage system in district heating," *Energies*, vol. 12, no. 7, p. 1391, 2019.

[29] R. Liu and C. Zhao, "Study of the short-term forecasting of the temperature sensitive load based on data mining technology," *Advances of Power System & Hydroelectric Engineering*, vol. 33, no. 11, pp. 32–38, 2017.

[30] C. Chen, D. Chen, and L. He, "Short-term forecast of urban natural gas load based on BPNN-EMD-LSTM combined mode," *Safety and Environmental Engineering*, vol. 26, no. 1, pp. 149–154, 2019.