

# Research Article **Power Load Prediction Based on Fractal Theory**

# Liang Jian-Kai,<sup>1</sup> Carlo Cattani,<sup>2</sup> and Song Wan-Qing<sup>1</sup>

<sup>1</sup>College of Electronic & Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China <sup>2</sup>Department of Mathematics (DIPMAT), University of Salerno, 84084 Fisciano, Italy

Correspondence should be addressed to Song Wan-Qing; swqls@126.com

Received 1 August 2014; Accepted 7 September 2014

Academic Editor: Xiao-Jun Yang

Copyright © 2015 Liang Jian-Kai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The basic theories of load forecasting on the power system are summarized. Fractal theory, which is a new algorithm applied to load forecasting, is introduced. Based on the fractal dimension and fractal interpolation function theories, the correlation algorithms are applied to the model of short-term load forecasting. According to the process of load forecasting, the steps of every process are designed, including load data preprocessing, similar day selecting, short-term load forecasting, and load curve drawing. The attractor is obtained using an improved deterministic algorithm based on the fractal interpolation function, a day's load is predicted by three days' historical loads, the maximum relative error is within 3.7%, and the average relative error is within 1.6%. The experimental result shows the accuracy of this prediction method, which has a certain application reference value in the field of short-term load prediction.

#### 1. Introduction

Short-term load forecasting plays an important role in control and operation of the power system. People are the main consumers of electrical energy. The periodicity of production and life of the people decides the periodicity of power load. The cyclicity of power load is performed as the week and seasonal periodicity. Power load not only is cyclical but also has certain continuity, which generally does not occur in big jumps and the load curve is continuous between any two points, making it possible to predict load.

Although traditional forecasting methods, such as gray theory, expert systems theory, and fuzzy mathematics, are relatively mature, the forecast results are often unsatisfactory [1]. To further improve the prediction accuracy, we need to make some improvements on the traditional methods. In recent years, prediction experts have put forward a prediction method, which is based on particle swarm optimization extended memory and support vector regression (SVR) and a prediction method which combines support vector machines (SVM) and wavelet neural network optimization [2–4]. This improves the accuracy of prediction but has a complex computing process. Since Mandelbrot created fractal geometry, fractals have been described in a large number of mathematical models of natural phenomena and have increasingly attracted people's attention. Fractals as a branch of nonlinear theory have penetrated into many other branches, and fractal dimension has been widely used in image processing, data compression, fault diagnosis, voice recognition, pattern recognition, and so on [5, 6].

This study is based on the existing similar daily load forecasting method and a deep research on the fractal characteristics of the power load, which designs corresponding fractal characteristic value algorithms to achieve scientific data processing for power load.

### 2. Fractal Theory

2.1. The Fractal Dimension. Fractal theory gives a geometric definition about calculation dimension formula: a set A consists of m compositions which are similar and the similarity ratio r can be considered as a dimension D:

$$D(A) = \frac{\log m}{\log r}.$$
 (1)

Since this formula can only be used in measuring the strict self-similarity geometry, the power load can adopt a similar method to approximate calculation.

Box-counting dimension is one of the most widely used dimensions. To calculate this dimension for a fractal F, imagine this fractal lying on an evenly spaced grid, and count how many boxes are required to cover the set. The box-counting dimension is calculated by seeing how this number changes as we make the grid finer by applying a box-counting algorithm [7].

Suppose that  $N(\varepsilon)$  is the number of boxes of side length  $\varepsilon$  required to cover the set. Then the box-counting dimension is defined as

$$d_{c}(F) = \lim_{\varepsilon \to 0} \frac{\ln N(\varepsilon)}{\ln (1/\varepsilon)}.$$
(2)

2.2. Fractal Interpolation Function. Fractal interpolation function is a method of fractal curve construction which has the advantage that it can reflect adjacent local features. The fractal interpolation algorithm can construct an iterated function system over the entire range rather than a function, so it can maintain the most characteristic of the original sample curve and the sample interpolation points can be displayed with rich details. Because it can well reflect the characteristics of the load curve at each point, one can more accurately predict the load.

Fractal interpolation method is based on the theory of iterated function systems. If given a set of fractal interpolation function, iterated function systems (IFS) will make the attractor close to the curve of fractal interpolation function. Each  $\omega_i$  in IFS is the affine transform function, which is given by the following structure formula [8, 9]:

$$\omega_i \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_i & 0 \\ c_i & d_i \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix}, \quad (3)$$

where (x, y) is a coordinate,  $a_i$ ,  $c_i$  are elements of transformation matrix,  $d_i$  is the vertical scaling factor, and  $e_i$ ,  $f_i$  are constant components of the transformed (x, y).

When *d* was given, the other parameters can be expressed as

$$a_{i} = \frac{x_{i} - x_{i-1}}{x_{n} - x_{0}},$$

$$c_{i} = \frac{y_{i} - y_{i-1}}{x_{n} - x_{0}} - d_{i} \frac{y_{n} - y_{0}}{x_{n} - x_{0}},$$

$$e_{i} = \frac{x_{n} x_{i-1} - x_{0} x_{i}}{x_{n} - x_{0}},$$

$$f_{i} = \frac{x_{n} y_{i-1} - x_{0} y_{i}}{x_{n} - x_{0}} - d_{i} \frac{x_{n} y_{0} - y_{n} x_{0}}{x_{n} - x_{0}}.$$
(4)

After obtaining the parameters, the IFS attractor can be obtained by deterministic iterative algorithm. With the increased number of iterations, the fitting degree of curve obtained by interpolation continues to improve and form a stable constant interpolation curve. 2.2.1. The Vertical Scale Factor. When calculating the affine coefficients of IFS, we temporarily regard vertical scaling factor *d* as a free parameter that can be empirically selected. However, parameter *d* has an impact on interpolation result, which is closely related to the complexity of corresponding fractal interpolation function. When |d| < 1, IFS converges to the only attractor.

There are many methods of calculating the vertical scaling factor d. In this paper, we use the analytical method to obtain parameter d; its principle is by calculating the minimum mean square error of the original and mapping function. The process is as follows [10, 11].

There is the following data sequence:  $\{(x_i, y_i) : i = 0, 1, 2, 3, ..., n\}; (x_p, y_p) \text{ and } (x_q, y_q) \text{ are two successive interpolation points; and } p < q$ ; then formula (3) is rewritten as

$$\omega_i \begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} a & 0 \\ c & d \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} = \begin{bmatrix} m \\ W_m \end{bmatrix}, \quad (5)$$

where  $m = int(ax_i + e)$  and the values of *m* points correspond to the values in the original function

$$c = \frac{\left(y_q - y_p - d\left(y_n - y_0\right)\right)}{\left(x_n - x_0\right)},$$

$$f = \frac{\left(x_n \cdot y_p - x_0 \cdot y_q - d\left(x_n \cdot y_0 - x_0 y_n\right)\right)}{\left(x_n - x_0\right)}.$$
(6)

The mean square error of the mapping function and the original function is

$$E = \sum_{i=0}^{n} \left[ W_m - y_m \right]^2.$$
 (7)

So  $W_m$  is obtained by

$$W_{m} = \left(\frac{y_{q} - y_{p}}{x_{n} - x_{0}}\right) \cdot x_{i} + \frac{x_{n} \cdot y_{p} - x_{0} \cdot y_{q}}{x_{n} - x_{0}} - d\left(\frac{y_{n} - y_{0}}{x_{n} - x_{0}} \cdot x_{i} + \frac{x_{n} \cdot y_{0} - x_{0}y_{n}}{x_{n} - x_{0}} - y_{i}\right).$$
(8)

To make the minimum mean square error E, the partial derivative of d will be zero:

$$\frac{\partial E}{\partial d} = 2\sum_{i=0}^{n} (A_i d - B_i) A_i = 0,$$

$$d_j = \frac{\sum_{i=0}^{n} B_i A_i}{\sum_{i=0}^{n} A_i^2}, \quad (j = 1, 2, ..., n),$$
(9)

where

$$A_{i} = y_{i} - [\varepsilon_{i}y_{0} + y_{n}(1 - \varepsilon_{i})],$$
  

$$B_{i} = y_{m} - [\varepsilon_{i}y_{p} + y_{q}(1 - \varepsilon_{i})],$$
  

$$\varepsilon_{i} = \frac{(x_{n} - x_{i})}{(x_{n} - x_{0})}.$$
(10)

2.2.2. The Method of Seeking Attractor. Attractor of IFS has a complex structure fractal diagram; the basic idea is that global and local geometric objects have self-similar structures under affine transformation. According to this principle, ultimately getting the attractor has nothing to do with the initial generator, but it depends on a set of iterative codes by affine transformation (also known as IFS code). Currently, we can use two ways to construct attractor on a computer, a deterministic algorithm (recursive algorithm) and a stochastic algorithm (random iterative algorithm) [12, 13]. Here we take an improved deterministic algorithm.

The procedure of improved deterministic algorithm is summarized in Algorithm 1.

## 3. The Design of Overall Program

The process is mainly divided into the following steps: load data preprocessing, similar day selection, short-term load forecasting, and daily load curve drawing.

3.1. Load Data Preprocessing. Short-term load forecasting relies on large amounts of historical data. Conducting preparatory work of load forecasting, we select and collect the original data regardless of what methods forecasters adopt; the true and reliable degree of data will have great influence on the extent of these predictions and directly affect the accuracy of prediction. In the process of collecting the original data, some abnormal data must be effectively corrected or complemented [14]. In the paper, we compare the value of the load at this time to the average value of the load at a time on different days. If the difference is greater than a threshold, we think it is the burr interference, modified by formula (12):

$$\overline{y}(j) = \frac{1}{n} \sum_{t=-n/2}^{n/2} y(j+t) \quad (t \neq 0, n \text{ is even, } j = 1, 2, \dots,)$$
(11)

$$y(j) = \begin{cases} \sum_{t=-n/2}^{n/2} k_t \cdot y(j-t), & y(j) > K_h \cdot \overline{y}(j) \\ \sum_{t=-n/2}^{n/2} k_t \cdot y(j-t), & y(j) < K_l \cdot \overline{y}(j) \\ y(j), & K_l \cdot \overline{y}(j) < y(j) < K_h \cdot \overline{y}(j), \end{cases}$$
(12)

where *n* is the number of sample points,  $K_t$  is the weight coefficient,  $K_h$  is the upper limit threshold value, and  $K_l$  is the lower threshold value. Here we set  $K_h = 1.1$ ,  $K_l = 0.9$ .

3.2. Design of Load Forecasting. The basic steps of fractal interpolation algorithm used in load forecasting are as follows.

Step 1. Preprocess load data; remove or repair spike points.

*Step 2*. Obtain samples, according to the relevant factors of the forecast day and historical day; use a similar day algorithm to select the same or similar type of three to five days from the history days as similar days of the predicted days.

*Step 3.* Choose a more similar day from similar days as a reference date.

*Step 4.* Establish IFS of the benchmark daily load; calculate fractal interpolation parameters.

*Step 5*. Establish IFS of the other similar daily load; also calculate fractal interpolation parameters.

*Step 6*. Obtaining the weighting average fractal interpolation parameters of several similar days, get a statistical meaning of IFS.

3.3. Daily Load Curve Drawing. Daily load curve can be drawn by the attractor method. According to the short-term load forecasting model, we obtain affine transformation group  $\{X; \omega_1, \omega_2, \ldots, \omega_n\}$ , set the generator, then use the deterministic algorithm to iterate the initial start point, and eventually draw daily load curve.

#### 4. The Prediction Example

This section will predict the work day and holidays load and prove the accuracy and usefulness of short-term load forecasting by fractal theory. The load data are provided by Eastern Slovakian Electricity Corporation. The calculation process is as shown in Figure 1.

4.1. Workday Load Forecasting. According to a city grid, we predict July 22 load data by July historical load data and July 22 is Tuesday, which is a work day. In this example, we take the 48 points from daily load curve (every 30 minutes to take a point) and calculate box-counting dimension from July 1 to July 26. July 8, July 15, and July 22 are Tuesdays, which meet the weekly periodic characteristics which accord with the regularities of load changes that vary on a weekly cycle. In addition, July 21 is close to July 22, which may indirectly reflect their loads trend. Obtained box dimension of July 21 is 1.2078, and the box dimensions of July 8 and July 15 are 1.2067 and 1.2008; they are relatively close, so we select July 8, July 15, and July 21 as similar days of July 22. Box-counting dimensions of July are shown in Table 1.

According to load data of July 8, 15, and 21, fractal interpolation parameters of three similar days were calculated. Weighting and averaging them, we finally get a statistically iterated function system and IFS codes (affine transformation set { $\omega_n$ , n = 1, 2, 3, ..., N}, w contains five parameters of a, c, d, e, and f). Parts of them are shown in Table 2.

According to the corresponding parameters of IFS, we weight and average them to obtain IFS codes by the fractal interpolation. Then by benchmark interpolating points, we use the deterministic algorithm to get attractor, which can be considered as the prediction daily load curve fit by the **Input**: the initial set of points x, the maximum number of iterations  $N_{max}$ ; **Output**: the attractor;

- (i) .take the point  $x_i$ , i = 0, 1, 2, ..., k within the set  $x \{x_0, x_1, x_2, ..., x_k\}$  and use iterative sequence by IFS affine transformation  $\{X, \omega_1, \omega_2, ..., \omega_n\}$ ;
- (ii) then replace the  $x_i$ , i = 0, 1, 2, ..., k by each new generated point,
  - $N_{\rm max} = N_{\rm max} 1;$
- (iii) if  $N_{\text{max}} \neq 1$ , return to step (ii), or go to the next step;
- (iv) .take the end point  $x_i$ , i = 0, 1, 2, ..., k within the set  $x \{x_0, x_1, x_2, ..., x_k\}$  and use iterative sequence by IFS, replace the  $x_i$ , i = 0, 1, 2, ..., k by each new point until the end.

ALGORITHM 1: Deterministic algorithm of seeking attractor.



FIGURE 1: The calculation process.

TABLE 1: Box-counting dimension.

Data	Box-counting dimension		
July 8	1.2067		
July 15	1.2008		
July 21	1.2078		

es

$\omega_i$	а	С	d	е	f
$\omega_1$	0.0213	-0.5357	-0.0698	0.4894	474.0602
$\omega_2$	0.0213	-0.3673	-0.0534	0.9894	453.7342
$\omega_3$	0.0213	-0.1882	-0.0274	1.4894	432.0924
$\omega_4$	0.0213	-0.4455	-0.0646	1.9894	444.5385
$\omega_5$	0.0213	0.0029	-0.0003	2.4894	403.6593
$\omega_6$	0.0213	-0.7698	-0.1085	2.9894	453.3236
$\omega_7$	0.0213	0.2650	0.0375	3.4894	366.3361
$\omega_8$	0.0213	0.2286	0.0344	3.9894	376.0253
$\omega_9$	0.0213	0.5837	0.0832	4.4894	359.7458
$\omega_{10}$	0.0213	-0.1254	-0.0199	4.9894	420.9114
$\omega_{11}$	0.0213	0.9604	0.1403	5.4894	346.4575
$\omega_{12}$	0.0213	0.6629	0.0893	5.9894	394.8628
$\omega_{13}$	0.0213	1.3693	0.1894	6.4894	364.2898
$\omega_{14}$	0.0213	0.9708	0.1426	6.9894	419.1270
$\omega_{15}$	0.0213	0.4246	0.0648	7.4894	481.0946
$\omega_{16}$	0.0213	0.2279	0.0282	7.9894	508.1429
$\omega_{17}$	0.0213	0.6179	0.0860	8.4894	485.8020
$\omega_{18}$	0.0213	0.2819	0.0430	8.9894	521.3269
$\omega_{19}$	0.0213	0.2887	0.0437	9.4894	530.2681

historical data. Figure 2 shows the prediction load curve and the actual load curve for July 22.



FIGURE 2: Actual and predicted daily load curve.

Through Figure 3, we conclude that the maximum relative error is 3.6778% and the average relative error is 1.32%, showing that the predicted and actual values are close and the relative error is small.

According to Figure 2, we can see the following.

- (1) Predicted load point and the actual load point are basically consistent with high prediction accuracy.
- (2) When electrical load changes are great, the error is small; on the contrary, when electrical load changes are small, the error is relatively large, so that is exactly the characteristics of fractal interpolation function algorithm.

4.2. Holidays Load Forecasting. After completing workday load forecasting, we predict July 19 load data by July historical





FIGURE 4: Actual and predicted daily load curve.

TABLE 3: Box-counting dimension.

Data	Box-counting dimension		
July 5	1.2119		
July 12	1.2144		
July 18	1.2109		

load data of July 5, 12, and 18, being holidays. Box-counting dimensions are shown in Table 3.

From the different types of load days, the fractal theory is proven to be accurate for short-term load forecasting. By the same principle, we can draw the predicted and actual load curves for July 19 from Figure 4.

Through Figure 5, we conclude that the maximum relative error is 3.0818% and the average relative error is 1.57%, showing that the predicted and actual values are close and the relative error is small.

#### **5. Conclusions**

Fractal theory has a great advantage in the study of nonlinear systems. In the paper, combining the fractal characteristics with its own short-term power load curve, we can have a deep study on algorithm and forecasting steps. Three main conclusions of this approach are as follows.





- (i) Through the concept of fractal theory and conventional methods for solving fractal interpolation parameters, we introduce the method of the vertical scaling factor to better reflect the details of the load, improving the accuracy of prediction.
- (ii) By studying the characteristics of the load data, using a box-counting method to select similar days, the algorithm was simplified for the application to power load calculation and good foundation was laid.
- (iii) Using both fractal interpolation function and improved deterministic iterative attractor algorithms, we can accurately predict the electric power load.

# **Conflict of Interests**

The authors declare that there is no conflict of interests regarding the publication of this paper.

# Acknowledgments

This project is supported by Shanghai Nature Science Foundation of China (Grant no. 14ZR1418500) and Postgraduate Innovation Project of Shanghai University of Engineering Science (Grant no. E1-0903-14-01039). The authors are grateful to the editor and the reviewers for their constructive comments.

#### References

- H. S. Hippert, E. C. Pedreira, and C. R. Souza, "Neural networks for short-term load forecasting: a review and evaluation," *IEEE Transactions on Power Systems*, vol. 16, no. 1, pp. 44–55, 2001.
- [2] W.-C. Hong, "Chaotic particle swarm optimization algorithm in a support vector regression electric load forecasting model," *Energy Conversion and Management*, vol. 50, no. 1, pp. 105–117, 2009.
- [3] Q. Zhang, K. Lai, and D. Niu, "Optimization combination forecast method of SVM and WNN for power load forecasting," in *Proceedings of the 4th International Joint Conference on Computational Sciences and Optimization (CSO "11)*, pp. 249– 253, 2011.
- [4] Q. C. Duan, Y. Zeng, D. W. Huang, P. Duan, and D. Liu, "Shortterm power load forecast based on particle swarm optimization

with extended memory and support vector regression," *Power System Protection and Control*, vol. 40, no. 2, pp. 40–44, 2012.

- [5] H. Shen, B. Ma, and F. Liu, "Study on power system load characteristics based on fractal theory," *Zhejiang Electric Power*, vol. 2, no. 2, pp. 15–17, 2007.
- [6] M. Yamahara, T. Noguchi, M. Okawa, and N. Yamada, "The relationship between subjective sleep disturbance and complexity of 24-hour activity utilizing fractal theory in psychiatric inpatients," *Sleep and Biological Rhythms*, vol. 7, no. 1, pp. 11–16, 2009.
- [7] M. D. Prieto, A. G. Espinosa, J.-R. Riba Ruiz, J. C. Urresty, and J. A. Ortega, "Feature Extraction of demagnetization faults in permanent-magnet synchronous motors based on boxcounting fractal dimension," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 1594–1605, 2011.
- [8] M. A. Navascués and M. V. Sebastián, "Generalization of Hermite functions by fractal interpolation," *Journal of Approximation Theory*, vol. 131, no. 1, pp. 19–29, 2004.
- [9] S. Xue-Yang and X. Yu-Cheng, "Research on development character of middle and small size fault structure in DongPang mine field on fractal theory," in *Proceedings of the 1st International Conference on Computing Control and Industrial Engineering* (CCIE '10), pp. 170–174, Wuhan, China, June 2010.
- [10] T. Darmanto, I. S. Suwardi, and R. Munir, "Cyclical metamorphic animation of fractal images based on a family of multi-transitional IFS code approach," in *Proceedings of the 1st IEEE Conference on Control, Systems and Industrial Informatics* (ICCSII '12), pp. 231–234, Bandung, Indonesia, September 2012.
- [11] M.-Y. Zhai, "Signal recovery in power-line communications systems based on the fractals," *IEEE Transactions on Power Delivery*, vol. 26, no. 3, pp. 1864–1872, 2011.
- [12] J. Lü and G. Chen, "A new chaotic attractor coined," *International Journal of Bifurcation and Chaos*, vol. 12, no. 3, pp. 659– 661, 2002.
- [13] H. Zhang and M. Liu, "Tree growth simulation method based on improved IFS algorithm," in *Proceedings of the International Conference on Computational Intelligence and Software Engineering (CiSE '09)*, pp. 1–5, Wuhan, China, December 2009.
- [14] W.-L. Xue, Study on Fractal Method of Power Load Forecasting, Harbin Institute of Technology, 2004.



The Scientific World Journal





**Decision Sciences** 







Journal of Probability and Statistics



Hindawi Submit your manuscripts at http://www.hindawi.com



(0,1),

International Journal of Differential Equations





International Journal of Combinatorics





Mathematical Problems in Engineering



Abstract and Applied Analysis



Discrete Dynamics in Nature and Society







Function Spaces



International Journal of Stochastic Analysis



Journal of Optimization