# Short-Term Load Forecasting Method Based on Fractal Theory

HONGSHENG SU
School of Automation and Electrical Engineering
Lanzhou Jiaotong University
Lanzhou 730070
P.R.CHINA
shsen@163.com

*Abstract:* - In terms of the present short-term load forecasting(STLF) methods, whether the linear or the nonlinear, neither could meet the STLF requirements better with the rapid developments of electrical power systems and electrical power markets, and so a novel STLF method was proposed based on fractal theory in this paper. Firstly, the paper investigated the fractal characteristics of power system loads based on fractal theory, then gave out the calculating method of the correlative dimension and embedded dimension according to G-P algorithm. Next, the paper discussed the C-C algorithm and revised it, and then used it to work out the time-delay. Finally, the paper established the STLF model and put into practice. The simulation results indicate that the proposed method possesses higher precision, and is an ideal STLF predictor.

*Key-Words:* - Short-term load forecasting, Fractal theory, Embedded dimension, Reconstruction, G-P algorithm, C-C algorithm

## 1 Introduction

Load forecasting is a very important work for power systems scheduling. Accurate short-term load forecasting (STLF) makes the generators start and stop more logically, the utilization of the resources more economically, and checking-repair plan more soundly, thus normal activities of the society production and life are ensured, and the economical and social benefits of electrical power enterprises are dramatically improved, the safe operation of the grids as well as scheduling is optimized. Hence, it possesses very important practical significance to research on STLF of electrical power systems.

Investigations on STLF have already been a long history. The conventional STLF methods include regression analysis[1-3], trend extrapolation[4,5], and etc. The new methods, such as ANN[6-8], expert systems (ES)[9,10], gray prediction[11,12], fuzzy forecasting[13,14], and Support Vector Machines[15,16], and some other new methods[17-24], are reported in recent years. The characteristics of regression analysis and trend extrapolation are represented by structure simplicity, principle easiness, and forecasting speed rapidness. However, the methods encounter the larger difficulties in initializing the forecasting models, and moreover, the influences of the diverse factors have not been fully considered. ANN, ES, and some new methods possess many advantages, but still present the flaws. ANN exposes some flaws such as low learning speed, even so doesn't converge sometimes. Compared with other methods, ES requires doing more working to construct ES knowledge base. However, to construct ES knowledge base is a difficult task. To change the situation, many scholars do their best to achieve many good predicting models, but all these models do a bit improvement on original ones only. The values of their practical applications are not too large, but the complexities of the models increase on and on. Later, one proposes the STLF models based on chaos theory [25-28]. It reconstructs the phase space through embedding dimension in one-dim time series, then STLF is implemented in phase space, in this way the forecasting results of the model are more creditable and reliable. However, chaos theory describes the change of the system phenomenon alone, but the geometry shape of the system is not considered, and the applications are for that limited. In view of the fact that fractal theory can consider the geometry shape of the system, and has close relationship with chaos theory [29-31], the paper therefore applies fractal theory to construct STLF model. Compared with other forecasting models, the results show that the model possesses higher

forecasting accuracy and better predicting behavior, and is a very ideal predictor.

This paper is organized as follows: In section 2 we introduce fractal theory foundation. In section 3, we discuss fractal analysis method, and analysis significantly G-P algorithm and C-C algorithm. In section 4, we investigate fractal characteristics of power systems load, and include self-comparability of power system loads under same space state, and self-comparability of power system loads under same time scale, and self-comparability of power system loads under diverse-time diverse-area scale. In section 5 based on fractal theory we present the STLF model. In section 6 we put the STLF model into practice, and compare the result with other methods. Eventually, the conclusion is summarized in section 7.

# 2 Fractal Theory Foundation

Fractal and Chaos theory are the two significant theoretical discoveries during interdisciplinary science investigation in the Twentieth Century. The thing that fractal theory considers is the representing structure of the system, while the thing that chaos theory investigates is the representing phenomenon of the system, that is, the phenomenon's deepening. The two theories make people's understanding on social phenomenon and natural law increase to a new stage. In recent years, fractal theory has been widely applied in physical chemistry, engineering, and economics and other fields, and the great achievements are acquired due to its stronger application background and use value. Below we mainly discuss fractal discrimination method.

## 2.1 Fractal Dimension

The thing that is described by Fractal dimension is how complex nonlinear systems evolve in size, or fill interspace. For fractal the evolution is implemented in space, and for time-series, the change is based on statistic significance. For the time-series with fractal characteristic, the relativities between data make each point congregate together slowly. But in fully random system, due to no relativities between data, each point could not get together. Fractal dimension is a very significant parameter of chaos systems, and is a constant parameter when chaos arresting cell varies, and is main tool for chaos investigation. Fractal dimension is differ from general integer dimension, it may be fraction. The common used fractal dimension has conjunction dimension, Hausdorff dimension, resemblance dimension, box dimension, and information dimension.

## 2.2 Lyapunov Exponential

Lyapunov exponential is used to scale the regular degree of complex kinetics. When Lyapunov exponential $\lambda<0$, then corresponding system is stable, and is insensitive to initial conditions, and conversely if $\lambda>0$, the corresponding system may be locally unstable, but at one time, the system is stable in whole, under this condition, chaos arresting cell is possibly formed to generate fractal. Hence, $\lambda>0$ may be seen as criterion of chaos behaviour. It must be pointed out that Lyapunov exponential $\lambda>0$ can distinguish out chaos system and no-chaos system, but can not tell out the extent of fractal characteristic[32,33].

## 2.3 Kolmogorov Entropy

Kolmogorov entropy is an important parameter of fractal theory, and is a quantification index to weigh the fractal extent[34]. Kolmogorov entropy can depict out the degree of fractal to some extent. In other words, Kolmogorov entropy can determine out how large the confirmation of the system possesses in a system with confirmation and randomicity co-existing. i.e., when Kolmogorov entropy $K=0$, and system is in rule movement; when $K>0$, and system is in chaos state; when $K=\infty$, and system is in random pocomotion. Simultaneously, the predictability of the system is determined by the value of Kolmogorov entropy, i.e., the smaller the $K$ value is, the lower the prediction of the system is. Kolmogorov entropy is calculated below.

$$K_{2,m}(r)=\frac{1}{\tau}\ln\frac{C_m(r)}{C_{m+1}(r)} \tag{1}$$

where $m$ is the embedded dimension, $\tau$ is the delay time, $C_m(r)$ is the conjunction integral function.

# 3 Fractal Analysis Method

The properties of time-series of power system load, such as self-comparability, fractal dimension, and evolution behaviour, are investigated using G-P algorithm in the paper. Since G-P algorithm is proposed according to phase space reconstruction and embedded theory, below we firstly give a simple introduction to them.

### 3.1 State Space Reconstruction

State space reconstruction, i.e., the recurrence process of fractal characteristic, means that isomorphic state space is constructed from single variable, thus origin system model may be constructed using an observing quantification.

Let the *n*-dimensional autonomous dynamic system be expressed below.

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2, ..., x_n) \\ \dot{x}_2 &= f_2(x_1, x_2, ..., x_n) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, x_2, ..., x_n) \end{aligned} \tag{2}$$

where $(x_1, x_2, \ldots, x_n)$ is the coordinates of state space of the system. After differential and eliminating $(x_2, \ldots, x_n)$, we then have

$$x_1^{(n)} = f(x_1, \dot{x}_1, \ddot{x}_1, ..., x_1^{(n-1)}) \tag{3}$$

At the moment, the coordinates of state space are replaced by each order differential coefficient, and the original evolution information has not loss. Differential coefficient is calculated using time-series value in diverse time as follow.

$$x(t), x(t+\tau), x(t+2\tau), ..., x(t+(n-1)\tau) \tag{4}$$

where $\tau$ is the delay time, $x(t+\tau)=x(t)+\dot{x}(t)\ \tau$. Similarly, $\ddot{x} \approx \dfrac{x(t+2\tau)-2x(t+\tau)+x(t)}{2\tau}$, and suchlike.

According to Takens theorem, the relationship between the dimension $D_m$ of the arresting cell and the embedded dimension *n* should meet

$$n \geq 2D_m + 1 \tag{5}$$

### 3.2 G-P Algorithm Analysis

The above equation (5) gives out the lower limit of the embedded dimension *n*, but does not have upper limit, which makes it limited in applications. According to Whitehead's embedding theory and Packing's reconstructing phase space, P. Grassberger and I. Procaccis proposed the calculating way of fractal dimension from time-series of data in 1983, and defined as G-P algorithm[35]. As the algorithm can work out the fractal dimension from one-dimension time-series, it has been wide applications in each field.

For convenient analysis, let $\tau$ in (4) be 1, and $x(t+(n-1))$ be $x_n$, we then get 1-dimensional time-series as follows.

$$x_1, x_2, x_3, ..., x_n \tag{6}$$

where $x_i$ represents the observing value in the *i*th time. Now assume that these data are divided into several groups, and each group has *m* data, i.e., the embedded dimension is *m*. Then the data of the 1st group are $\boldsymbol{y}_1=(x_1, x_2, \ldots, x_m)$, and $\boldsymbol{y}_2=(x_2, x_3, \ldots, x_{m+1})$ for the 2nd group, and likewise, we have

$$\boldsymbol{Y}_i=(x_i, x_{i+1}, \cdots, x_{m+i-1}),\ i=1,2, \cdots, N_m \tag{7}$$

According to (7), one-dimension time-series data in (6) is divided into $N_m$ groups, and $N_m = n-m+1$, where each group data expresses a vector or a point in *m*-dimension space, and all the points constitute a subset $J(m)$ in m-dimension Euclidean space $E^m$. and the distance is defined between the points below.

$$r_{i,j} = d(y_i - y_j) = \{\sum (x_i - x_j)^2\}^{1/2} \tag{8}$$

where $r_{ij}$ expresses the distance between $y_i$ and $y_j$.

**Definition 1**. Let $C(r)$ be conjunction integral function described below.

$$C(r) = \frac{2}{N_m(N_m-1)} \sum_{\substack{i,j=1 \\ i \neq j}}^{N_m} \theta(r - r_{ij}) \tag{9}$$

where $\theta$ is Heaviside unit function described by

$$\theta(x) = \begin{cases} 0, x \leq 0 \\ 1, x > 0 \end{cases} \tag{10}$$

Let $r \to 0$, we then have

$$\ln C(r) = \ln C + D(m) \ln r \tag{11}$$

Then

$$D(m) = \lim_{r \to 0} (\ln C(r) / \ln r) \tag{12}$$

If $D(m)$ is constant as *m* increases, then

$$D_2 = \lim_{m \to \infty} D(m) \tag{13}$$

where $D_2$ is called as conjunction dimension.

### 3.3 Time-delay τ Selection

It is important to correctly select the time-delay $\tau$ and embedded dimension $m$ during phase-space reconstruction. Generally, there are two kinds of viewpoints present, one viewpoint thinks that the selection on $\tau$ and $m$ is unattached in theory each other, such as sequence correlation method, phase-space expansion method, and plural self-correlation method, and etc. Another one thinks that the selection on $\tau$ and $m$ is mutually attached, such as time-window method, C-C algorithm[36,37], they may work out $\tau$ and time window at the same time.

C-C algorithm was proposed by H .S. Kim, R. Eykholt, and J. D. Salas in1999, it adopts the conjunction integral to estimate $\tau$. Below let us make an introduction.

For analysis simplicity, let $t$ be time-delay, Eq.8 can be rewritten below.

$$C(m,N,r,t)=\frac{2}{N_m(N_m-1)}\sum_{\substack{i,j=1\\ i\neq j}}^{N_m}\theta(r-r_{ij}) \tag{14}$$

Now consider time-series $\boldsymbol{x}=(x_i|i=1,2,\cdots,N)$, let $t$=1, we get single tiem-series itself, and $t$=2, we get $\boldsymbol{x}=(x_1,x_3,\cdots,x_{N-1})$ and $\boldsymbol{x}=(x_2,x_4,\cdots,x_N)$, the sequence length is $N/2$. For any $t$, we have

$$\begin{aligned} x^1&=\{x_i\mid i=1,t+1,\cdots,N-t+1\}\\ x^2&=\{x_i\mid i=2,t+2,\cdots,N-t+2\}\\ &\cdots\ \cdots\ \cdots\\ x^t&=\{x_i\mid i=1,2t,\cdots,N\}\end{aligned} \tag{15}$$

where $N=tl$, $l=N/t$ is sequence length. Below we define checkout statistic by

$$S_1(m,N,r,t)=C(m,N,r,t)-C^m(1,N,r,t) \tag{16}$$

For each sub-sequence, we have

$$S_2(m,N,r,t)=\frac{1}{t}\sum_{s=1}^{t}\left[C_s(m,\frac{N}{t},r,t)-C_s^m(1,\frac{N}{t},r,t)\right] \tag{17}$$

Let $N\to\infty$, we have

$$S_2(m,r,t)=\frac{1}{t}\sum_{s=1}^{t}\left[C_s(m,r,t)-C_s^m(1,r,t)\right] \tag{18}$$

If the time-series are unattached and follow the same distribution, when $N\to\infty$, $S_2(m,r,t)=0$ for all $r$. But in practice, $N$ is usually not equal infinity, and so $S_2(m,r,t)\neq 0$. Hence, the local maximal time-interval may select the zero-point of $S_2(m,r,t)$, or the time-point which possess minimal discrimination for all $r$. To select the two $r$ which correspond to the maximum and the minimum, we define

$$\Delta S_2(m,t)=\max\{S_2(m,r_j,t)\}-\min\{S_2(m,r_j,t)\} \tag{19}$$

Clearly, $\Delta S_2(m, t)$ scales the deviation regarding $r$. Hence, the local maximal time should select the smaller value between zero-point of $S_2(m, r, t)$ and $\Delta S_2(m, t)$. Seen from the above analysis, the optimal time-delay should be the first local minimum of $S_2(m, r, t)$ and $\Delta S_2(m, t)$. The problem is that how large $N$ is required to get rational estimation on $m$, according to DBS statistic conclusion,

$$N_{\min}\approx 10^{2+0.4m} \tag{20}$$

and

$$r_i=i\times 0.5\sigma \tag{21}$$

where $\sigma$ is standard difference of time-series. Let $m$=2,3,4,5, and $i$=1,2,3,4, and then define

$$\overline{S_2}(t)=\frac{1}{16}\sum_{m=2}^{5}\sum_{i=1}^{4}S_2(m,r_i,t) \tag{22}$$

$$\Delta\overline{S_2}(t)=\frac{1}{4}\sum_{m=2}^{5}\Delta S_2(m,t) \tag{23}$$

Then the optimal time-delay $t_d$ is the zero-pointing of (22) or the first local minimum of (23). Consider the two conditions, synthetically, we define the global minimum of

$$S_{2cor}(t)=\Delta\overline{S_2}(t)+\left|\overline{S_2}(t)\right| \tag{24}$$

as the embedded window $t_w$.

According to the above method, we find it possesses flaws when calculating the $\tau$. As in practice, the zero-pointing of (22) is not equal to the first local minimum of (23). If time-series has cycle $T$, then $t=KT$ is zero-point of (22), simultaneously, it is also the global minimum of (24). This is inconsistent with the former description. In the paper we select the first local minimum of (23) as the optimal time-delay $t_d$. But when $t=KT$ the equation (23) equals zero, and (23) still shows constantly increasing undulation with $t$ increasing.

**Table 1** Load values in full o'clock from 7th-Sep-2009 to 13th-Sep-2009

| Time | 2009.9.7 Mon | 2009.9.8 Tue | 2009.9.9 Wed | 2009.9.10 Thu | 2009.9.11 Fri | 2009.9.12 Sat | 2009.9.13 Sun |
|---|---|---|---|---|---|---|---|
| 00:00:00 | 19.65562 | 22.03054 | 19.50496 | 17.31560 | 18.40292 | 20.56363 | 17.91225 |
| 01:00:00 | 18.46718 | 20.03968 | 17.77087 | 16.75096 | 18.06091 | 19.01609 | 16.94366 |
| 02:00:00 | 17.58250 | 18.44542 | 16.11951 | 14.70013 | 14.77449 | 16.61128 | 14.76071 |
| 03:00:00 | 16.50782 | 17.23822 | 15.13741 | 13.86362 | 14.38104 | 15.47946 | 13.45124 |
| 04:00:00 | 16.31171 | 16.45588 | 15.32464 | 13.79082 | 14.53923 | 14.90607 | 13.23663 |
| 05:00:00 | 16.73845 | 16.98046 | 15.87528 | 14.74795 | 18.38729 | 15.43221 | 13.79529 |
| 06:00:00 | 20.42165 | 20.16048 | 19.11494 | 18.13952 | 19.92930 | 18.83209 | 15.06410 |
| 07:00:00 | 21.94692 | 21.94635 | 20.44828 | 19.74379 | 19.51349 | 19.91040 | 16.26883 |
| 08:00:00 | 23.09392 | 22.20051 | 20.23754 | 19.90303 | 21.63161 | 19.17505 | 15.49879 |
| 09:00:00 | 25.68276 | 24.27925 | 21.57800 | 21.40582 | 22.93236 | 19.53681 | 15.72068 |
| 10:00:00 | 27.01327 | 25.04792 | 22.05609 | 22.47287 | 23.96074 | 20.08721 | 16.46189 |
| 11:00:00 | 27.38191 | 25.03651 | 21.98844 | 23.22141 | 24.22040 | 20.57258 | 17.39085 |
| 12:00:00 | 27.59679 | 24.85070 | 22.65517 | 23.47164 | 24.26291 | 20.94662 | 17.92330 |
| 13:00:00 | 27.51493 | 24.06480 | 22.05525 | 22.37264 | 24.58419 | 20.29611 | 17.73231 |
| 14:00:00 | 26.52052 | 23.42715 | 21.79336 | 22.58452 | 24.33843 | 20.14876 | 17.79996 |
| 15:00:00 | 26.49043 | 24.01339 | 21.92775 | 22.83713 | 25.26148 | 20.25313 | 17.54841 |
| 16:00:00 | 27.42996 | 24.65630 | 22.77594 | 23.95268 | 25.56785 | 20.53909 | 18.25283 |
| 17:00:00 | 27.96801 | 25.44427 | 23.05518 | 24.12369 | 24.28073 | 21.51140 | 19.05809 |
| 18:00:00 | 26.07701 | 23.22383 | 22.07355 | 22.90213 | 24.11323 | 21.46275 | 19.03052 |
| 19:00:00 | 26.18509 | 24.00569 | 22.17559 | 23.20809 | 24.11323 | 21.54701 | 19.79276 |
| 20:00:00 | 26.69598 | 23.85331 | 22.15677 | 23.57331 | 24.11323 | 21.86266 | 19.79573 |
| 21:00:00 | 26.79810 | 23.52475 | 21.62288 | 22.85041 | 24.11323 | 21.67198 | 19.38401 |
| 22:00:00 | 25.52110 | 22.59386 | 19.58275 | 21.25477 | 22.64067 | 21.16397 | 18.54376 |
| 23:00:00 | 23.84008 | 21.00444 | 18.69471 | 19.91184 | 25.56778 | 19.35402 | 17.02838 |

**Table 2** Fractal dimension and *K*-entropy for every day in a week

| Time | Mon | Tue | Wed | Thu | Fri | Sat | Sun |
|---|---|---|---|---|---|---|---|
| $D_2$ | 0.9724 | 0.97 | 1.0136 | 1.0533 | 0.9965 | 0.9601 | 1.0293 |
| *K*-entropy | 0.4333 | 0.414 | 0.4275 | 0.4447 | 0.4428 | 0.4275 | 0.4139 |

**Table 3** Fractal dimension and *K*-entropy under same-space diverse-time scale

| Time | Day | Week | Month |
|---|---|---|---|
| $D_2$ | 1.4701 | 1.7946 | 1.9196 |
| *K*-entropy | 0．7330 | 0.1747 | 0.1673 |

When $t_d$ is larger, such high-frequency undulation can influence the selection of the first local minimum of (23). Hence, we require the improvement on C-C algorithm.

Compared (16) with (17), for fixed $m$ and $r$, when $N\to\infty$, they have same undulation rule as a whole. But as $t=KT$ formula (17) equals zero. Hence here we replace (17) with (16) to calculate $\Delta S_1(m,t)$, and find out the first local minimum of $\Delta S_1(m,t)$ as the optimal time-delay $t_d$.

# 4 Fractal Characteristic Analysis of Power System Loads

According to power system loads characteristics and fractal theory, self-comparability of power system loads curve is investigated here under diverse time scales and diverse space states. Below we may take power system loads in a southern city for example, apply G-P algorithm to calculate the conjunction dimension and Kolmogorov entropy defined as *K*-entropy. Table 1 shows the load values in full o'clock from 7th-Sep-2009 to 13th-Sep-2009, and the load unit is expressed using MWh.

## 4.1 Self-comparability of Power System Loads Under Same Space State

According to Table 1 and the former G-P algorithm, we may work out the fractal dimension and Kolmogorov entropy as shown Table 2.

We analyses fractal characteristic of power system loads under same time and space scale above, below we analyses the fractal characteristic of power system loads under same space and diverse time scale. The calculating result is shown in Table 3.

Seen from Table 2 and Table 3, fractal dimension and *K*-entropy of power system loads under same-space same-time scale are comparatively stable, but with time increasing, fractal dimension of power system loads also increases, the reason is that time-series data increase with time increasing, which leads to a increasing of fine degree of the arresting cell, consequently, fractal dimension increases. In addition, based on *K*-entropy, we can judge the operation of the system is in chaos state.

In Table 3, Fractal dimensions and *K*-entropy of power system loads under same-space diverse-time scale are quit approximate. Hence, power system loads under same-space diverse-time scale are quit approximate. Hence, power system loads under same-area diverse-time scale may be considered as the relationship between the part and the whole. In other words, power system loads in same area possess self-com comparability in time scale.

## 4.2 Self-comparability of Power System Loads Under Same Time Scale

To expound self-comparability of power system loads further, we investigate under same-time diverse-space scale, we select power loads of the four different areas under same day to implement fractal analysis, the results are shown in Table 4.

**Table 4** Fractal dimension and K-entropy under same-time diverse-space scale

| Area | Huining | Tangtai | Shatang | Guohui |
|---|---|---|---|---|
| $D_2$ | 1.4392 | 1.1104 | 1.059 | 1.3263 |
| *K*-entropy | 0.6632 | 0.9077 | 0.906 | 0.6675 |

Known from Table 4, fractal dimension of power system loads under diverse-area same-time are very approximate. If power system loads in diverse areas are considered as the composition parts of power grid loads, then self-comparability exists between each part. If the areas are expanded, and seen as diverse grids, then diverse grids possess self-comparability. Simultaneously, *K*-entropy is larger than zero in Table 4, i.e., system locomotion is in chaos.

### 4.3 Self-comparability of Power System Loads Under Diverse-time Diverse-area Scale

In order to make out self-comparability of power system loads under diverse-time diverse-area scale, fractal analysis is done in different areas according to diverse time scale as shown in Table 5.

**Table 5** Fractal dimension and *K*-entropy under diverse-time diverse-space scale

| Area/Time | Area A/hour | Area B/Day | Area C/Year |
|---|---|---|---|
| $D_2$ | 1.8142 | 1.7704 | 1.01 |
| *K*-entropy | 0.2336 | 0.2662 | 0.2937 |

Seen from Table 5, power system loads in diverse-area diverse-time scale possess definite self-comparability, but it is worse compared with the former ones. Seen from fractal dimension, area A possesses the best self-comparability, and area B is better, and area B is the worst, which is resulted in by power system load data in diverse sample time. In addition, based on *K*-entropy, we can know that the system locomotion is neither irregular nor random.

## 5 STLF Method Based on Fractal Theory

To implement STLF using fractal theory, we require to use the measuring load data to reconstruct phase-space. The evolutionary rule of the reconstructed phase-space can correctly reflect the change rule of time-series of power system loads. However in high-dimensional interspace, it is difficult to model track, and the modelling error on nonlinear function is quite large. To resolve it we require selecting the *k*-nearest neighbour points from *N* states to implement the local prediction for the point to be predicted using linear regression method. Such disposal is doable for both the linear system and the nonlinear system. The concrete prediction steps are described as follows.

1) To implement the pretreatment for the data of power system loads, i.e., to analyses whether the abnormal data and losing data exist in time-series data or not, and do corresponding disposal.

2) To confirm the time-delay $\tau$, conjunction dimension $D_2$, and embedded dimension $m$ of time-series data.

3) To reconstruct the phase-space corresponding to load time-series data.

4) To find out the $p$-nearest neighbour points of the point $x_k$ to be predicted using Euclidean distance method. The number of the nearest neighbour points is influential on predicting accuracy.

5) To calculate the weight value of each nearest neighbour point by

$$P_i = \frac{\exp(-a(d_i - d_m))}{\sum_{i=1}^{p} \exp(-a(d_i - d_m))} \tag{25}$$

where $a$ is a parameter, $a$=1; $d_i$ is a distance between the point $x_k$ and the nearest neighbour point $x_{ki}$, $d_m$=min($d_i$, $i$=1,2,⋯,$p$).

6) To implement the local linear fitting using the weighted 1-order local linear regression method described as follows.

$$\begin{bmatrix} x_{k1+1} \\ x_{k2+1} \\ \vdots \\ x_{kp+1} \end{bmatrix} = \begin{bmatrix} I & x_{k1} \\ I & x_{k2} \\ \vdots & \vdots \\ I & x_{kp} \end{bmatrix} \bullet \begin{bmatrix} a \\ b \end{bmatrix} \tag{26}$$

where $[a, b]^{\mathrm{T}}$ is estimated using the weight least mean square(LMS) described below.

$$\sum_{i=1}^{p} P_i (x_{ki+1} - a - bx_{ki})^2 = \min \tag{27}$$

To resolve partial differential coefficient on the two sides of (27), and after simplification, we get

$$\begin{cases} a\sum_{i=1}^{p} P_i x_{ki} + b\sum_{i=1}^{p} P_i x_{ki}^2 = \sum_{i=1}^{p} P_i x_{ki} x_{ki+1} \\ a + b\sum_{i=1}^{p} P_i x_{ki} = \sum_{i=1}^{p} P_i x_{ki+1} \end{cases} \quad (28)$$

To resolve the $a$, $b$ from (28), and substitute it into linear regression equation, we may then predict out load value of next time.

## 6 Examples

The load data come from one southern city in China in Sep-2009, whose load curve is shown in Fig.1.

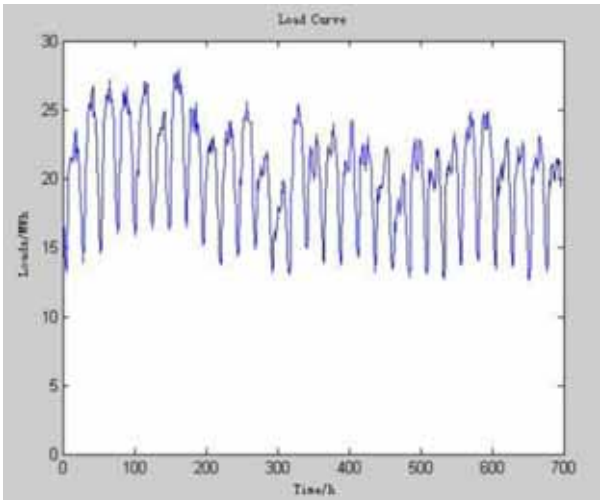


**Fig. 1** Load curve diagram

Seen from the Fig. 1, the loads curve possesses self-comparability. According to (11), we get the relation curve between ln$C(r)$ and ln$r$ with fixed $m$ as shown in Fig. 2

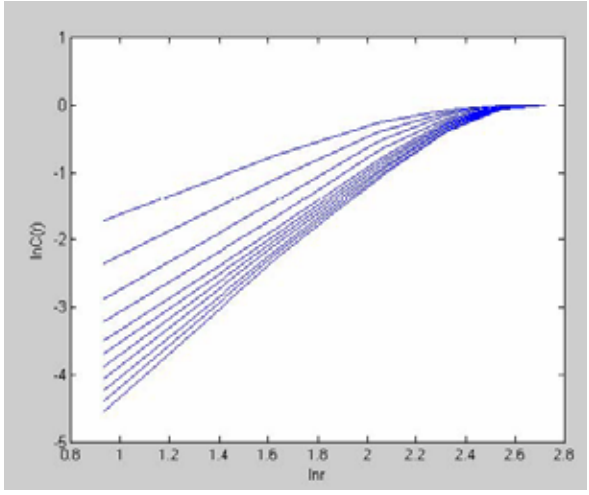


**Fig. 2** Relationship curve between ln$C(r)$ and ln$r$

According to (12), we may get the change of conjunction dimension $D(m)$ along with $m$ as shown in Fig. 3. Known from Fig. 3, when $m$>6, $D(m)$ hardly changes along with $m$, and goes towards 1.897, and 1.9 approximately. Hence here we give $m$=6. According to (1), $K$-entropy equals 0.023 greater than zero, and so the system is in chaos.

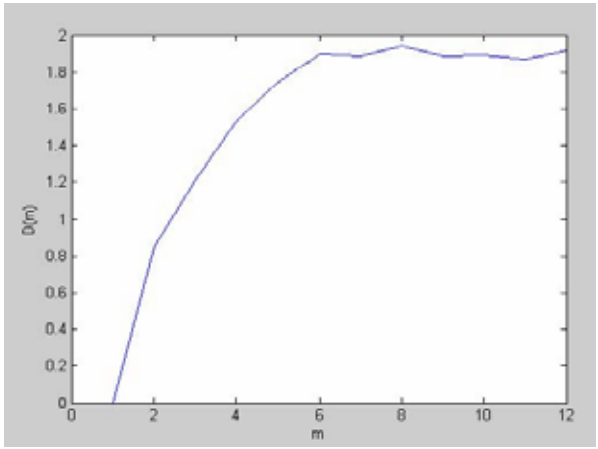


**Fig. 3** Change trend curve of $D(m)$ with $m$

In this paper there are 696 load data to be applied in all, where 672 data are used to learn, the remained 24 load data are used to test. According to C-C algorithm mentioned above, the time-delay $\tau$ is calculated as 4. After reconstructing phase-space we acquire a 6-dimensional phase-space.

The current phase-point is the load value $Y_{672}$ at 23:0:0 o'clock on 28th-9-2009, now $Y_{673}$ at 0:0:0 o'clock on 29th-9-2009 requires to be predicted. According to section 5, we require to find out the $k$-nearest neighbouring phase-point of $Y_{672}$ to work out prediction. Here we let $k$=6, then we work out prediction for 29th-9-2009, and make a comparison with practical values as shown in Table 6, where the error is calculated by

$$\text{error} = \frac{|\text{predicting\_value} - \text{practical\_value}|}{\text{practical\_value}} \times 100\% \quad (29)$$

And the average relative error is calculated by

$$\text{average\_relative\_error} = \frac{1}{N} \times \text{error} \quad (30)$$

After calculation, the average relative error is 0.97%, i.e., the forecasting accuracy arrives at 90% above using fractal theory, the desired result is acquired. At the same time, we find the forecasting error on 29th-9 is inside 3%, and 60% error is in range of 1%, and some individual errors are larger as

**Table 6** Load forecasting results at integral o'clock on 29th-9-2009

| Time | Practical values (MWh) | Predicting values (MWh) | Error(%) (MWh) |
|---|---|---|---|
| 00:00:00 | 20.72 | 20.56 | 0.77 |
| 01:00:00 | 16.29 | 16.4 | 0.67 |
| 02:00:00 | 15.53 | 15.39 | 0.89 |
| 03:00:00 | 13.45 | 13.78 | 2.5 |
| 04:00:00 | 13.41 | 13.52 | 0.76 |
| 05:00:00 | 13.27 | 13.16 | 0.86 |
| 06:00:00 | 13.69 | 13.93 | 1.74 |
| 07:00:00 | 17.82 | 17.76 | 0.28 |
| 08:00:00 | 19.64 | 19.79 | 0.76 |
| 09:00:00 | 19.45 | 19.53 | 0.45 |
| 10:00:00 | 20.57 | 20.42 | 0.71 |
| 11:00:00 | 20.86 | 21.02 | 0.76 |
| 12:00:00 | 20.92 | 20.75 | 0.83 |
| 13:00:00 | 21.11 | 21.0 | 0.51 |
| 14:00:00 | 20.05 | 20.06 | 0.04 |
| 15:00:00 | 19.9 | 20.0 | 0.50 |
| 16:00:00 | 20.01 | 19.83 | 0.9 |
| 17:00:00 | 21.12 | 21.25 | 0.63 |
| 18:00:00 | 21.41 | 21.16 | 1.15 |
| 19:00:00 | 21.19 | 21.36 | 0.78 |
| 20:00:00 | 21.15 | 20.94 | 0.98 |
| 21:00:00 | 21.23 | 21.36 | 0.61 |
| 22:00:00 | 21.38 | 21.2 | 0.82 |
| 23:00:00 | 20.73 | 20.47 | 1.26 |

**Table 7** Practical forecasting results of three kinds of methods

| Time | Practical value (MWh) | Fractal model | | BP networks model | | RBF netwoeks model | |
|---|---|---|---|---|---|---|---|
| | | Prediction (MWh) | Error (%) | Prediction (MWh) | Error (%) | Prediction (MWh) | Error (%) |
| 00:00:00 | 20.72 | 20.56 | 0.77 | 16.51 | 20.3 | 17.11 | 17.4 |
| 01:00:00 | 16.29 | 16.40 | 0.67 | 16.05 | 1.47 | 15.78 | 3.13 |
| 02:00:00 | 15.53 | 15.39 | 0.90 | 15.68 | 0.96 | 14.00 | 9.85 |
| 03:00:00 | 13.45 | 13.78 | 2.45 | 13.46 | 0.07 | 13.40 | 0.37 |
| 04:00:00 | 13.41 | 13.52 | 0.82 | 13.42 | 0.07 | 13.28 | 0.96 |
| 05:00:00 | 13.27 | 13.16 | 0.82 | 12.66 | 4.60 | 14.24 | 7.30 |
| 06:00:00 | 13.69 | 13.93 | 1.75 | 17.24 | 25.9 | 16.79 | 23.9 |
| 07:00:00 | 17.82 | 17.76 | 0.33 | 20.41 | 14.5 | 18.83 | 5.66 |
| 08:00:00 | 19.64 | 19.79 | 0.76 | 18.61 | 5.37 | 18.97 | 3.41 |
| 09:00:00 | 19.45 | 19.53 | 0.45 | 21.09 | 8.43 | 20.41 | 5.39 |
| 10:00:00 | 20.57 | 20.42 | 0.41 | 20.63 | 0.29 | 21.13 | 2.72 |
| 11:00:00 | 20.86 | 21.02 | 0.76 | 21.15 | 1.39 | 21.64 | 3.73 |
| 12:00:00 | 20.92 | 20.75 | 0.81 | 20.79 | 0.62 | 21.31 | 1.86 |
| 13:00:00 | 21.11 | 21.00 | 0.52 | 20.20 | 4.31 | 20.18 | 4.40 |
| 14:00:00 | 20.05 | 20.06 | 0.04 | 19.93 | 0.60 | 20.73 | 3.39 |
| 15:00:00 | 19.90 | 20.00 | 0.50 | 19.80 | 0.50 | 20.65 | 3.76 |
| 16:00:00 | 20.01 | 19.83 | 0.89 | 21.54 | 7.64 | 21.37 | 6.79 |
| 17:00:00 | 21.12 | 21.25 | 0.61 | 22.19 | 5.06 | 21.52 | 1.89 |
| 18:00:00 | 21.41 | 21.16 | 1.16 | 22.95 | 7.19 | 20.95 | 2.14 |
| 19:00:00 | 21.19 | 21.36 | 0.80 | 22.73 | 7.26 | 20.52 | 3.16 |
| 20:00:00 | 21.15 | 20.94 | 0.99 | 22.27 | 5.29 | 20.43 | 3.40 |
| 21:00:00 | 21.23 | 21.36 | 0.61 | 21.77 | 2.54 | 19.54 | 7.96 |
| 22:00:00 | 21.38 | 21.20 | 0.84 | 19.68 | 7.95 | 18.92 | 11.5 |
| 23:00:00 | 20.73 | 20.47 | 1.25 | 18.90 | 5.29 | 18.52 | 10.6 |

**Table 8** Statistic results of three kinds of methods

| Forecasting model | Relative error | Minimum error | Maximum error | Error <1% | Error <3% | Error >10% |
|---|---|---|---|---|---|---|
| Fractal model | 0.83% | 0.04% | 2.45% | 83.3% | 100% | none |
| BP networks | 5.87% | 0.07% | 25.9% | 29.2% | 42% | 12.5% |
| RBF networks | 6% | 0.37% | 23.9% | 8.33% | 25% | 16.6% |

history data are less which lead to larger distances between the nearest neighbouring points and it. With the increasing of the history data, the selection of the nearest neighbouring points can be improved, consequently, the forecasting accuracy is also improved.

To illuminate the advantages of fractal forecasting, below we make a comparison with BP neural networks and RBF neural networks, respectively. Let the structure of BP networks be 24-28-24, the transfer function in middle layer be *S*-type, and one in output layer be also *S*-type, and the training time be 1000, and the aim error be 0.01, and learning rate be 0.1. Likewise, the structure of RBF is 24-10-24, and aim error is zero, and expansion speed is 0.4. Fig. 4 shows the simulation results of the above three kinds of methods.

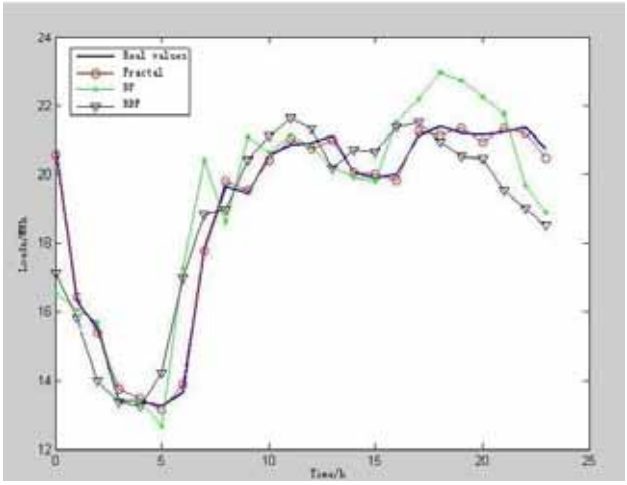


**Fig. 4** Forecasting results of three kinds of the methods

Table 7 shows the practical forecasting results of three kinds of methods, and Table 8 shows the statistics results of the three kinds of forecasting models.

Seen from Table and Table 8, forecasting effects of fractal model are the best. Moreover, fractal forecasting does not require learning and samples selection, and is inexistent not to be able to converge, whose prediction speed is quicker, and ubiquity is stronger.

In another example, we adopt the history load data between 1996 and 2008 in a China power grid Co. to implement the month load forecasting based on fractal forecasting model. Where there are the former 150 load data used to learn, and the latter 6 data for test. With the same process we calculate the fractal dimension and *K*-entropy, and the embedded dimension, and the conclusion is achieved the system is in chaos. Below we let $k$=8, then month load on 7-2008 to 12-2008 is predicted as shown in Table 9.

**Table 9** Load forecasting results($10^6 \times$MWh)

| Month | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|
| Practice | 21.67 | 22.61 | 19.78 | 12.57 | 23.27 | 25.72 |
| Prediction | 20.83 | 22.79 | 20.14 | 12.98 | 23.01 | 25.24 |
| Error(%) | 3.876 | 0.796 | 1.820 | 3.261 | 1.117 | 1.866 |

For convenient comparison, the number of the nearest neighbouring points is respectfully selected as 6 and 10 to make the same forecasting, the results are as shown in Table 10.

**Table 10** Load forecasting results($10^6 \times$MWh)

| Month | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|
| Practice | 21.67 | 22.61 | 19.78 | 12.57 | 23.27 | 25.72 |
| Prediction ($k$=6) | 20.41 | 22.87 | 20.27 | 13.34 | 22.83 | 24.96 |
| Prediction ($k$=8) | 20.83 | 22.79 | 20.14 | 12.98 | 23.01 | 25.24 |
| Prediction ($k$=10) | 20.38 | 22.92 | 20.24 | 13.22 | 22.94 | 24.17 |

Seen from Table 10, when the number of the nearest neighbouring points are selected as 8 the forecasting results are the most accurate, the larger *k* value such as 10 or the smaller *k* value such as 6 is selected, accordingly, the accuracy is lower, which indicates that the suitable *k* value to be selected can improve the forecasting precision.

## 7 Conclusion

The established STLF model based on fractal theory has higher forecasting precision and reliability than other nonlinear forecasting models, and can suit load forecasting demands for different area as long as we change the history data and the number of the nearest neighbouring points. It can be used by the power enterprises to guide their practice to improve their production efficiency and competitive power. It is particularly important that it may better meet the demand of TOU power price in power market, and save electric energy, and ensure safe reliability operation of power systems. The forecasting model presented in this paper can not only be applied in STLF, but also may be used in other similar cases, and possesses wide application prospect.

*References:*


[1] Chenghui Liu, *Theory and Method for Power System Load Forecasting*, Harbin Industry University Press, 1987.

[2] Guoquan Xiao, Chun Wang, Fuwei Zhang, *Power System Load Forecasting*, Chinese Electrical Power Press, 2001.

[3] K. Xie, J. Zhou, Combination Forecasting Models Using Regression Analysis Method, J*ournal of Chongqing University((Natural Science Edition)*, Vol.26, No.1, 2003, pp.62-65.

[4] Erking Yu, Fang Han, Kai Xie, *Power System Load Forecasting*, Chinese Electrical Power Press, 1999.

[5] Chaoming Huang, Chi-Jin Huang, Mingli Wang, A Particle Swarm Optimization to Identifying the ARMAX Model for Short-Time Load Forecasting, *IEEE Transactions on Power Systems*, Vol.20, No.2, 2005, pp.1126-1133.

[6] D.C. Park, M.A. El-Sharkawi, R.J. Marks II, Electric Load Forecasting Using an Artificial Neural Network, *IEEE Transactions on Power System*, Vol.6, No.2, 1991, pp.442-449.

[7] K. Wang, Short-term Load Forecasting for Special Days in Anomalous Load Conditions Using Neural Networks and Fuzzy Inference Method, *IEEE Transactions on Power Systems,* Vol.15, No.2, 2000, 559-565.

[8] H. S. Su, Chaos Quantum-Behaved Particle Swarm Optimization Based Neural Networks for Short-term Load Forecasting, *Procedia Engineering*, Vol.15, 2011, pp,199-203.

[9] Saifur Rahman, Rahul Bhatnagar, An Expert Systerm Based Algorithm for Short Term Load Forecasting, *IEEE Transactions on Power Systems*, Vol.2, No.3, 1998, pp.886-892.

[10] Kun-Long Ho, Yuan-Yih, Short Term Load Forecasting of Taiwan Power System Using a Knowledge-Based Expert System, *IEEE Trans. on Power Systems*, Vol.5, No.4, 1990, pp.1145-1152.

[11] W.L. Wang, X.Y. Wei, An Improved Method based on Grey Model in Load Forecasting of Power System, *Journal of Northeast China Institute of Electric Power Engineering*, Vol.17, No.2, 1997, pp.37-43.

[12] D.X. Niu, T.T. Zhang, L.R. Chen, B. Zhang, Grey Load Forecasting Models Based on Relational Analysis of Multi-factor, *Journal of North China Electric Power University (Natural Science Edition)*, Vol.33, No.5, 2006, pp.90-92.

[13] R.M. Shen, Fuzzy Model of Short Term Load Forecasting on Power Syestem, *Journal of East China Shipbuilding Institute(Natural Science Edition)*, Vol.4, No.2, 1989, pp.32-37.

[14] C.L. Zhang, Power System Short-term Load Forecasting Based on Fuzzy Clustering Analysis and Rough Sets, *Journal of North China Electric Power University(Natural Science Edition)*, Vol.35, No.3, 2008, pp.38-43.

[15] L. Zhang, X, Liu, H. Yin, Applications of Support Vector Machines Based on Time Sequence in Power System Load Forecasting, *Power System Technology*, Vol.28, No.19, 2004, pp.38-41.

[16] Y.J. Zhai, J.X. Wang, L.H. Zhou**,** Power System Mid-term Load Forecasting Based on Fuzzy Support Vector Machines, *Journal of North China Electric Power University (Natural Science Edition)*, Vol.35, No.2, 2008, pp.70-73.

[17] S. Fan, and L. Chan, Short-term Load Forecasting Based on an Adaptive Hybrid Method, *IEEE Transactions on Power Systems*, Vol.21, No.2, 2006, pp.392-401.

[18] K. B. Song, Y. S. Baek, D. H. Hong, G. Jang, Short-Term Load Forecasting for the Holidays Using Fuzzy Linear Regression Method, *IEEE Transactions on Power Systems*, Vol.19, No.1, 2004, pp.96-101.

[19] Tsakoumis A C, Fessas P, Mladenov V M, and et al, Application of Chaotic Time Series for Short-term Load Prediction, *WSEAS Trans on Systems*, Vol.2, No.3, 2003, pp.517-523.

[20] J. Y. Fan, J. D. McDonald, A Real-time Implementation of Short-term Load Forecasting for Distribution Power Systems, *IEEE Transactions on Power Systems*, Vol.9, No.2, 1994, pp.988-994.

[21] J. Wang, G. Ren, Load Forecasting Based on Chaotic Support Vector Machine with Incorporated Intelligence Algorithm, *Control and Decision*, Vol.18, No.1, 2003, pp.89-91.

[22] H. Su, Y. Zhang, Short-Term Load Forecasting Using H∞ Filter and Elman Neural Networks, *Proceedings of International Conference on Control and Automation*, Guangzhou, China, 2007, pp.1868-1872.

[23] Q. Cheng, Y. Wang, W. Chen, Modified Principal Component Analysis Based Short-Term Load Forecasting, *Power System Technology*, Vol.29, No.3, 2005, pp.64-67.

[24] H. Kassaei, A. Keyhani, T. Woung, A Hybrid Fuzzy Neural Network Bus Load Modeling and Predication, *IEEE Transactions on Power Systems*, Vol. 14, No.2, 1999, pp.718-724.

[25] G. Ji, J. Cheng, H. Mi, Short-Term Load Forecasting Using Chaos Phase Space Mode Linear Regression Model, *Systems Engineering Theory & Practice*, Vol. 21, No.6, 2001, pp.138-140.

[26] M. Zeng, J. Cheng, Y. Xu, D. Yuan, Short-Term Load Forecasting Based on Artificial Neural Network and Fuzzy Theory, *Journal of Hunan University(Natural Sciences)*, Vol.35, No.1, 2008, pp.58-61.

[27] S.Z. Zhang, L. Ma, Y.M. Sun, Load Forecasting Model Using Chaos Theory and Support Vector Machine, P*roceedings of the CSU-EPSA***,** Vol.20, No.6, 2008, pp.31-35.

[28] R. Fang, J. Zhou, Y. Zhang, L. Liu, Short-Term Probabilistic Load Forecasting Using Chaotic Time Series, *Journal of Huazhong University of Science and Technology(Nature Science Edition)*, Vol. 38, No.5, 2009, pp.125-128.

[29] B. Mandelbrot, *The Fractal Geometry of Nature*, W. h. Freeman, and Company, 1983.

[30] T. Vicsek, *Fractal Growth Phenomena*, World Scientific Publishing Co., 1989.

[31] E. Peters, Fractal Structure in the Capital Market, *Financial Analysis Journal*, Vol.52, No.3, 1989, pp:131-137.

[32] J. H. Lv, Y.A. Lu, S.H. Chen, *Analysis and Aplications on Chaos Time-Series*, Wuhan University Press, China, 2002

[33] Mikael, Bask, Dimensions and Lyapunoy Exponents form Exchange Rate Series, *Chaos, Solution & Fractals*, Vol.7, No.12, 1996, pp.199-202.

[34] Y. Li, and B. Yang, *Chaotic Oscillator Inspection Introduction*, Publishing House of Electronics Industry, 2004.

[35] P. Grassbeiger, I. Procaccis, Measuring the Strangeness of Strange Attractors, *Physicia*, No.9, 1983, pp.189-208.

[36] Z.L. Yang, K.Y. Lin, Improving Precision of Short Term Load Forecasting by Numerical Testing in Local Linearization Method of Phase Space Reconstruction, *Automation of Electric Power Systems*, Vol.27, No.16, 2003, pp.40-44, 2003.

[37] S. Yang, C. Jia, Two Practical Methods of Phase Space Reconstruction, *Acta Physica Sinica,* Vol.51, No.11, 2001, pp.2452-2458.