

A Study of Fractal Image Compression Based on an Improved Genetic Algorithm

Lifeng Xi*, Liangbin Zhang
Institute of Mathematics, Zhejiang Wanli University
Ningbo, 315100, P.R. China

(Received 24 December 2006, accepted 3 March 2007)

Abstract: Reducing the search complexity of matching between range block and domain block in fractal image compression is one of the most active research areas lately. This paper puts forward an improved genetic algorithm proposed for obtaining matching domain blocks to prevent the premature convergence of GA to a suboptimal individual, to make individual diversity and to improve GA's search efficiency. For each range block, an improved GA's strategy is to encode the position of searching domain block with Gray code, define fitness for minimum distance of current range block matching with searching domain block, select optimal individual protection strategy and adjust GA's controlled parameters of crossover and mutation probability adaptively. Experimental results show that compared with Jacquin coding scheme, our coding scheme achieves an average of 85% reduction in encoding time and improves the efficiency of search. At the same time, the compression ratio and quality of decoded images are guaranteed to the same as Jacquin coding scheme for the same images.

Key words: fractal image coding; genetic algorithms; gray coding; adaptive GA

1 Introduction

Compression and decompression technology of digital image has become an important aspect in the storing and transferring of digital image in information society. Fractal image coding, introduced by Barnsley and Jacquin [1-3], is the outcome of the study of the iterated function system [4] developed in the last decade. Because of its high compression ratio and simple decompression method, many researchers have done a lot of research on it. At present, researchers focus mainly on how to select and optimize the classification of the range blocks, balance the speed of compression and decompression, increase the compression ratio and improve the quality of image after decompression [5]. Especially in the field of reducing the complexity of search, many outstanding algorithms based on classified search are put forward after some thorough study, this paper puts forward an improved genetic algorithm for fractal image compression.

Genetic Algorithm (GA) is a stochastic algorithm simulating the process of natural evolution, which is usually applied to optimize controlled parameters and constrained functions. Especially GA is efficient to solve nonlinear multiple-extrema problems [6-8]. This paper puts forward an improved genetic algorithm proposed for obtaining matching domain blocks of fractal partition in image compression, which uses the partition iterated function system and fractal image compression. For each range block, an improved GA's strategy is to code the position of searching domain block with Gray code, define fitness for minimum distance of current range block matching with searching domain block, select optimal individual protection strategy and adjust GA's controlled parameters of crossover and mutation probability adaptively. In the end, our coding scheme utilizes a GA in finding the appropriate domain block for each range block.

* **Corresponding author.** Tel. : +86-574-8822 2249; Fax: +86-574-8822 2249 E-mail address: xilf@zwl.edu.cn.

Theoretical foundations are outlined in Section 2. Fractal image compression based on an improved genetic algorithm is described in Section 3. Experimental result is reported in Section 4. Discussion and conclusion are included in Section 5.

2 Theoretical Foundations

2.1 Self-affine and Self-similar Transformations

The fractal image compression algorithm is based on the fractal theory of self-similar and self-affine transformations [9]. Recall the following basic definitions of fractals (also see [16-20]):

Definition 1. A self-affine transformation $w : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a transformation of the form $w(x) = T(x) + b$, where T is a linear transformation on \mathbb{R}^n and $b \in \mathbb{R}^n$ is a vector.

Definition 2. A mapping $w : D \rightarrow D$, $D \subseteq \mathbb{R}^n$ is called a contraction on D if there is a real number $c \in (0, 1)$, such that $d(w(x), w(y)) \leq cd(x, y)$ for $x, y \in D$ and for a metric d on \mathbb{R}^n . The real number c is called the contractility of w .

Definition 3. If $d(w(x), w(y)) = cd(x, y)$, then w is called a similarity.

A family $\{w_1, \dots, w_m\}$ of contractions is known as a local iterated function system (LIFS). If there is a subset $F \subseteq D$ such that for a LIFS $\{w_1, \dots, w_m\}$,

$$F = \bigcup_{i=1}^m w_i(F), \quad (1)$$

then F is said to be invariant for that LIFS. If F is invariant under a collection of similarities, F is known as a self-similar set.

Let S denote the class of all non-empty compact subsets of D . The δ -parallel body of $A \in S$ is the set of points within distance δ of A , i.e.

$$A_\delta = \{x \in D : \text{there exists } a \in A \text{ such that } |x - a| \leq \delta\}. \quad (2)$$

Let us define the distance $d(A, B)$ between two sets A, B to be

$$d(A, B) = \inf\{\delta : A \subset B_\delta \text{ and } B \subset A_\delta\}. \quad (3)$$

The distance function is known as the Hausdorff metric on S (other distance functions can also be used).

Given a LIFS $\{w_1, \dots, w_m\}$, there exists a unique compact invariant set F , such that

$$F = \bigcup_{i=1}^m w_i(F). \quad (4)$$

This F is known as the attractor of the system.

If E is a compact non-empty subset such that $w_i(E) \subset E$ and

$$w(E) = \bigcup_{i=1}^m w_i(E). \quad (5)$$

We define the k -th iteration of w , $w^k(E)$, to be

$$w^0(E) = E, w^k(E) = w(w^{k-1}(E)), \quad (6)$$

for $k \geq 1$, then we have

$$F = \bigcap_{i=1}^{\infty} w^i(E) \quad (7)$$

The sequence of iteration $w^k(E)$ converges to the attractor of the system for any set E . This means that we can have a family of contractions that approximate complex images and, using the family of contractions, the images can be stored and transmitted in a very efficient way. Once we have a LIFS, it is straightforward to obtain the encoded image.

If we want to encode an arbitrary image in this way, we will have to find a family of contractions so that its attractor is an approximation to the given image. Barnsley's Collage Theorem states how well the attractor of a LIFS can approximate the given image.

2.2 Collage Theorem

Let $\{w_1, \dots, w_m\}$ be contractions on \mathbb{R}^n so that for any $x, y \in \mathbb{R}^n$ and any i ,

$$|w_i(x) - w_i(y)| \leq c|x - y| \quad (8)$$

where $c \in (0, 1)$ is a constant. Let $E \subset \mathbb{R}^n$ be any non-empty compact set. Then

$$d(E, F) \leq \frac{1}{(1-c)} d(E, \bigcup_{i=1}^m w_i(E)) \quad (9)$$

where F is the invariant set for the w_i , and d is the Hausdorff metric [9].

As a consequence of this theorem, any subset of \mathbb{R}^n can be approximated within an arbitrary tolerance by a self-similar set, i.e., given $\delta > 0$, there exist contracting similarities $\{w_1, \dots, w_m\}$ with invariant set F satisfying $d(E, F) < \delta$. Therefore, the problem of finding a LIFS $\{w_1, \dots, w_m\}$ whose attractor F is arbitrary close to a given image I is equivalent to minimize the distance $d(I, \bigcup_{i=1}^m w_i(I))$.

2.3 Fractal Image Coding

Fractal image coding makes good uses of image self-similarity in space by ablating image geometric redundant. Fractal coding process is quite complicated but decoding process is very simple, which makes use of potentials in high compression ratio. The main theory of fractal image coding is based on iterated function system, attractor theorem, and collage theorem. Regard original compressible image as attractor, how to get LIFS parameters is main problem of fractal coding.

We explain the basic procedure for the fractal image coding [9].

1. A given image I is divided into non-overlapping M range blocks of size $B \times B$ and into arbitrarily located N domain blocks of size $2B \times 2B$. The range blocks are numbered from 1 to M , and represented by $R_i (1 \leq i \leq M)$. Similarly, the domain blocks are from 1 to N , and represented by $D_j (1 \leq j \leq N)$.

2. For each range block R_i , the best matched domain $D_k (1 \leq k \leq N)$ and an appropriate contractive affine transformation τ_{ik} which satisfy the following equation are found through

$$d(R_i, \tau_{ik}(D_k)) = \min d(R_i, \tau_{ij}(D_j)) \quad (10)$$

Where τ_{ij} is an contractive affine transformation from the domain block D_j to the range block R_i ; the distortion measure $d(R_i, \tau_{ij}(D_j))$ is the mean square error (MSE) between the range block R_i and the contractive domain block $\tau_{ij}(D_j)$. The contractive affine transformation τ_{ij} is composed of two mappings ϕ_j and θ_{ij} as follows:

$$\tau_{ij} = \theta_{ij} \circ \phi_j. \quad (11)$$

The first mapping ϕ_j is the transformation of domain-block size to the same size as range blocks. This transformation can be described as follows: The domain block D_j is divided into non-overlapping unit blocks of size 2×2 ; and each pixel value of the transformed block $\phi_j(D_j)$ is an average value of four pixels in each unit block in D_j . The second mapping θ_{ij} consists of two steps: The first step transforms the block $\phi_j(D_j)$ by one of the following eight transformations: rotation around the center of the block $\phi_j(D_j)$, through $0^\circ, +90^\circ, +180^\circ$, and $+270^\circ$, and each rotation after orthogonal reflection about mid-vertical axis of the block $\phi_j(D_j)$. Those eight transformations are called isometries. The second step is the transformation p_{ij} of pixel values of a block obtained by the first step. This transformation p_{ij} is defined as

$$p_{ij}(v) = a_{ij}v + g_{ij}, \quad (12)$$

where v is a pixel value of the block obtained by the first step, and the parameters a_{ij} and g_{ij} are computed by the least square analysis of pixel values of the range block R_i and the block obtained by the first step. We call the parameters a_{ij} and g_{ij} a scaling coefficient and an offset, respectively.

The LIFS parameters listed below are encoded:

1. Parameters to indicate a location of the best matched domain block;

2. A parameter to indicate an isometry on the best matched domain block;
3. A scaling coefficient and an offset.

The proposed method quantizes these LIFS parameters [10].

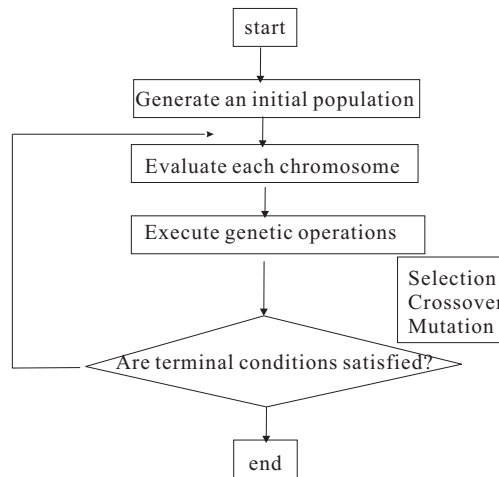


Figure 1: genetic algorithm

2.4 Genetic Algorithm

Genetic algorithms (GAs) [6-8] are search procedures based on the mechanics of natural selection and natural genetics. The GAs work as follows (Fig. 1): first code each individual in the search space as a finite-length string (chromosome), which consists of the characters (genes) 1's and 0's. A set of the chromosomes is a population. Second evaluate each chromosome with the fitness by an objective function (fitness function). Third apply basic operations to the population of the chromosomes. The basic operations compose of selection, crossover, and mutation. Selection is an operation which selects the chromosomes according to their fitness values. The higher the chromosome's fitness value is, the higher its probability to produce offsprings into the next generation is. Crossover creates new chromosomes by swapping genes of parent chromosomes for each pair of selected chromosomes. Mutation changes the gene of the chromosomes with a probability (mutation probability), 1 to 0 and 0 to 1. This process is repeated until terminal conditions are satisfied.

3 Proposed Methodology

The main aspect of fractal-based image coding is to find a suitable domain block and a transformation for a rough type range block. Thus, the whole problem can be regarded as a search problem. Instead of a global search mechanism we have introduced GA's to find the near optimal solution. Combined with fractal image compression, we design GA parameters to deal with the following problems.

3.1 Individual Chromosome Coding

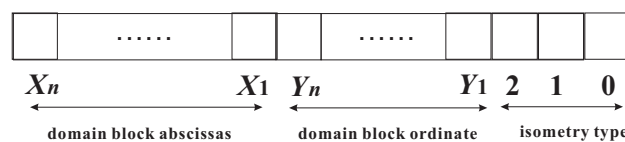


Figure 2: gray code for searching domain block

The LIFS parameters are composed of $\{x, y, z, a, g\}$, here $\{x, y\}$ is the location of the best matched domain block, $\{z\}$ is the isometry type of the best matched domain block, $\{a, g\}$ is the scaling coefficient

and an offset. Solution code is a representation of elements in the population. Generally, it is represented as binary code. In our scheme, chromosomes $\{x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n, z_1, z_2, z_3\}$ is applied with Gray code. Figure 2 shows Gray code of searching domain block.

During the search process of GA's execution, the search of domain block is completed in encoding space (one dimension), evaluation and selection of domain block is completed in decoding space (two dimension). See Figure 3.

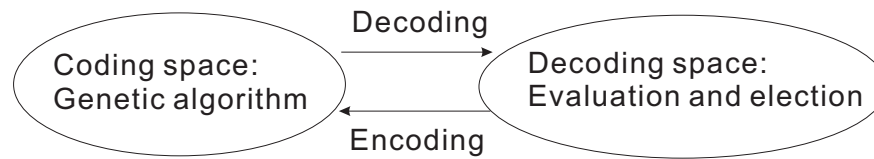


Figure 3: encoding and decoding execution

A given image $I(w \times w)$ is divided into non-overlapping range blocks of size $b \times b$, for each range block, the search of domain block position (usually top-left coordinate) is coded by Gray code $g_i (i = 1, 2, \dots, n)$, then the larger n is, the more accurate it is. Firstly, we convert Gray code $g_i (i = 1, 2, \dots, n)$ into binary code $X_i (i = 1, 2, \dots, n)$ with form (13)

$$\begin{aligned} X_n &= g_n \\ X_i &= X_{i+1} \oplus g_i, i = n - 1, n - 2, \dots, 1 \end{aligned} \quad (13)$$

then we get numerical value X with form (14)

$$X = 1 + \frac{\sum_{i=1}^n x_i \cdot 2^{i-1}}{2^n - 1} * (w - 2b) \quad (14)$$

It can effectively overcome crag during evolution process. Results show that applying gray code with individual chromosomes has the features of object, convenient, extract, fast and good application. It is a superior nonlinear optimal method that can locally meticulously search the global solution with great probability [11].

3.2 Initial Population

In our scheme, the number of the initial population is taken by splitting the difference. The initial population contains fifty chromosomes randomly generated.

3.3 Selecting the Fitness Function

The fitness function to be selected should reflect a desirable trait in the member of the population. In the case of fractal image compression, the fitness function is defined as follows:

$$fitness(x) = \begin{cases} \frac{1}{mse(x)+1}, & mse(x) > \varepsilon \\ \infty, & mse(x) \leq \varepsilon \end{cases} \quad (15)$$

where $mse(x)$ is MSE between the given range block and the domain block obtained by chromosome x .

3.4 Genetic Operations

Three basic operations (selection, crossover and mutation) executed in the GA of the proposed method are described below.

- **Selection**

Our selection scheme is to combine the proportion choice with the protection of the best chromosome. The probability of a given chromosome in the population being selected is then given by

$$Pis = Fi / \left(\sum_{i=1}^M Fi \right), (i = 1, 2, \dots, M), \quad (16)$$

where M is the population number, F_i is the fitness of the chromosome i . Therefore, the larger the fitness of the chromosome is, the more selected probability it has, which is called roulette method. Nevertheless, Markov link has been proved theoretically that the best fitness chromosome of the current generation is perhaps not selected and cannot get the optimization convergence [12]. Thus, good choice is adopted that the best fitness chromosome does not execute the crossover and mutation and replaces the worst fitness chromosome after the generation is executed.

• **Crossover**

The crossover is performed by selecting two elements in the present population as the parent. The selection of parents is made according to a probability distribution as above. Since a chromosome has three group genes, the crossover point is decided arbitrarily among every group, and the crossover operation is executed respectively. See Figure 4.

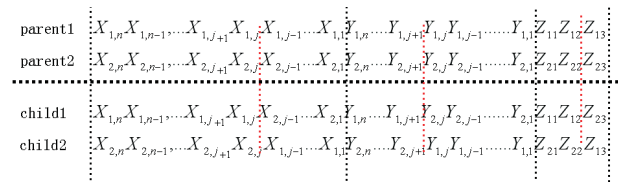


Figure 4: three group independent crossover

Where using the standard genetic algorithm, in every group genes each parent is divided into the two parts with the crossover point, and after it to create two new elements: one with the first part taken from the first parent and with the second part taken from the second parent, and the other in exactly the opposite way. See Figure 4.

• **Mutation**

Mutation changes the gene of the chromosomes with a mutation probability, 1 to 0 or 0 to 1. The role of mutation operation in GA is that of restoring lost or unexplored genetic material into the population to prevent the premature convergence of GA to suboptimal chromosome. The mutation operation is called a subtle process [13].

• **Crossover Probability P_c and Mutation Probability P_m**

Crossover probability P_c and mutation probability P_m play an important role in GA. Crossover causes a randomized exchange of genetic material between chromosomes. Crossover occurs only with some probability P_c which controls the rate at which chromosome is subjected to crossover. The larger value P_c is, the faster is the new chromosome introduced into the population. The smaller value P_c is, the lower the searching process is leading to stagnation. Typical initial value of P_c is in the range 0.5~1.0. The mutation probability P_m is varied according to the generations. The initial P_m is larger for the global search, and in some generations it is smaller for the local search. Finally it is larger again for avoidance of local optimum. Typical initial value of P_m is in the range 0.005~0.05 [14-15].

In our scheme, we put forward an adaptive varied values of P_c and P_m as follows

$$P_c = \begin{cases} p_{c1} - \frac{(p_{c1}-p_{c2})*(f'-f_{avg})}{f_{max}-f_{avg}}, & f' \geq f_{avg} \\ p_{c1}, & f' < f_{avg} \end{cases}, \tag{17}$$

$$P_m = \begin{cases} p_{m1} - \frac{(p_{m1}-p_{m2})*(f-f_{avg})}{f_{max}-f_{avg}}, & f \geq f_{avg} \\ p_{m1}, & f < f_{avg} \end{cases} \tag{18}$$

Here, f_{max} is the maximum fitness function of current generation, f_{avg} is the average fitness function of current generation, f' is larger fitness function of the two crossover chromosomes selected, f is the fitness function of mutation chromosome selected, p_{c1}, p_{c2} is crossover probability, and p_{m1}, p_{m2} is mutation probability. Our experimental parameters include: $p_{c1} = 0.9, p_{c2} = 0.6, p_{m1} = 0.1, p_{m2} = 0.001$.

Our scheme adjusts GA's controlled parameters of crossover and mutation probability adaptively in many aspects:

(1) Compared with the current generation average fitness function, those chromosomes with large fitness function are chosen to have small P_c and P_m in order to be protected to enter the next generation, and those chromosomes with small fitness function are chosen to have large P_c and P_m to be eliminated.

(2) If individual fitness function approaches the maximum fitness function of current generation, P_c and P_m are coming to decrease; if individual fitness function exceeds it, P_c and P_m become to P_{c1} and P_{m1} respectively. So that excellent individuals' P_c and P_m are increased to avoid evolution stagnancy.

- **Terminal Condition**

As for endless execution of genetic algorithms, we end searching process after 150 iterations and then the individual with maximum fitness function is chosen to be the matched domain block corresponding to the current range block.

4 Experimental Result

All the experiments are carried out on a computer with Intel 2.5Ghz and 512M RAM in the Win2000 professional operating system and VC6.0 language is used. Original image is classical 128×128 grey-level Lena face image coded with 8 bits per pixel.

An optimal bit allocation strategy is as follows: 14 bits for the location of the matched domain block (horizontal and vertical coordinate), 3 bits for isomorphic types, 5 bits for contrast scaling and 7 bits for the offset. For each of the range block, fractal code includes 29 bits allocation via writing into a text file as a fractal coding file. During the iteration process of the image decoding, those grey value either exceeding integer 255 or less than integer 0 are replaced by the average of its four neighbors to avoid block diverging.

Table 1 shows the genetic algorithm parameters. Table 2 shows the experimental data of our coding scheme compared with Jacquin coding scheme. Figure 5 shows that the decoded image our coding scheme compared with Jacquin coding scheme.

Table1. Genetic Algorithm's Parameters

Range block size	4*4	Adaptive p_{c1}, p_{c2}	$p_{c1} = 0.9, p_{c2} = 0.6$
Population size	50	Adaptive p_{m1}, p_{m2}	$p_{m1} = 0.1, p_{m2} = 0.001$
Initial crossover probability P_c	0.85	Fitness f	$f = \frac{1}{rms+1}$
Initial mutation probability P_m	0.1	Iterations	150

Table2. Our Coding Scheme Comparison with Jacquin Coding Scheme

Performance	Jacquin coding scheme	Our coding scheme
Range block size	4*4	4*4
Searching domain block amount	14641	7500
Coding time	250 S	31 S
Compression ratio	4.5:1	4.5:1
PSNR	36.6	31.2
Decode image	Figure 5.(b)	Figure 5.(c)

Table 2 and Figure 5 show that it is efficient to compute multi-extremum problem with an improved genetic algorithm for fractal image compression. After analyzing experimental results, we can draw a conclusion that compared with Jacquin coding, our coding scheme can greatly shorten image coding time (an

average of 85% reduction compared with Jacquin method) and improve the search efficiency, at the same time, the compression ratio and quality of the decode image are guaranteed to be the same as Jacquin coding scheme for the same images.

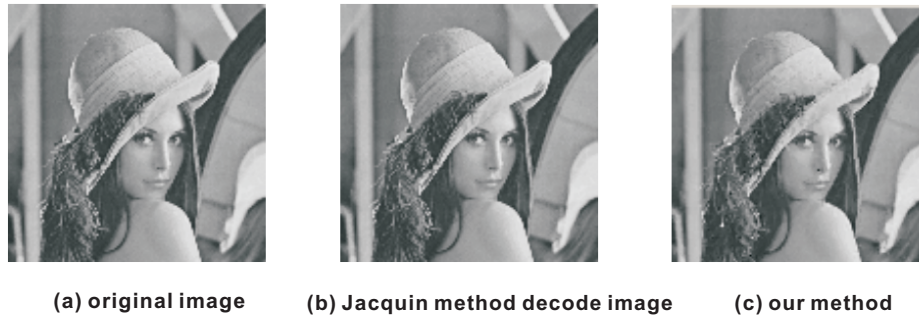


Figure 5: reconstruction comparison image

5 Conclusion

In this paper, our coding scheme tries to apply an improved genetic algorithm for fractal image compression, including using Gray code for individual chromosome and adaptive crossover and mutation probability, which can prevent the premature convergence of GA to suboptimal chromosome, make individual diversity and improve GA's search efficiency. From the experimental result, our coding scheme is feasible and effective. However, compared with enumerate algorithms, few individuals do search fail to get the global optimal result and a little block effect exists in the decode image. As for a searching method of GA, how to make use of the fractal characteristic to encode and decode image, such as fractal dimension[16] and other related topics[17-20], how to design GA's operators including selection, crossover and mutation, how to control the important GA's parameters, such as initial crossover and mutation probability, how to improve the compression ratio and decode quality, are our main work for future.

Acknowledgements

This research is supported by National Natural Science Foundation of China (Grant No.10671180), and Research Project (No.20060183) of Education Department of Zhejiang Province of China.

References

- [1] A. E. Jacquin: Fractal image coding a review. *Proceeding of the IEEE*. 81(10),212-223(1993)
- [2] M. F. Barnsley and A. E. Jacquin: Application of recurrent iterated function systems to images. *Proc SPIE*. 1001(3),122-131(1998)
- [3] A. E. Jacquin: A novel fractal block-coding technique for digital image. *Proceedings of ICASSP IEEE International Conference on ASSP*. 4(2),34-42(1990)
- [4] M. F. Barnsley and S.Demko: Iterated function system and the global construction of fractal. *Pro.R.Soc.Lond*. 39(4), 243-275(1985)
- [5] Mingshui Li, Shanhu Ou and Heng Zhang: The new progress in research approach of fractal image compression. *Journal of Engineering Graphics*. 4(3),143-152(2004)
- [6] Xiaoping Wang, Liming Cao: Genetic algorithms-theory, application and software realizability. *Xi'An Jiao Tong University Press*(2002)
- [7] J. H. Holland: Adaptation in nature and artificial systems. *University of Michigan Press: Michigan* (1975)

- [8] David Goldberg: Genetic algorithms in search, optimization and machine learning. *Addison-Wesley Publishing Company: MA*(1989)
- [9] Shouji Chen, Liming Zhang: Fractal and image compression. *Shanghai Science And Technology Education Publishing House*(1998)
- [10] Y. Fisher: Fractal image compression, *Fractals*. 2(3),25-36(1994)
- [11] Xiaohua Yang: Gray coding based accelerating genetic algorithm and its theory. *Theory and Application of System Engineering*. 3(4), 100-106(2003).
- [12] D. E. Goldberg, P. Segrest: Finite markov chain analysis of genetic algorithm. *Proceedings of the Second Conference on GA Hillsdale Lawrence Enlbaum Associates*.2,1-8(1987).
- [13] H. Muhlenbein: How genetic algorithms really work: mutation and hillclimbing parallel problem solving from nature,(PSNR2). *Elsvier Science Publishers: North-Holland,Amsterdam*(1992)
- [14] Rudolph: Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Network*. 5(3), 96-101(1994)
- [15] Haoyang Wu, Changchun Zhu: Adaptive genetic algorithm to improve group premature convergence, *Journal of Xi'An Jiao Tong University*.33(11),27-32(1999)
- [16] L. F. Xi: Hausdorff Dimension of level set of Rademacher series. *C. R. Acad. Sci. Paris Serie I* 327, 29-33 (2000)
- [17] Mei Sun, Lixin Tian, Jian Yin: Hopf bifurcation analysis of the energy resource chaotic System. *International Journal of Nonlinear Science*. 1(1), 49-53 (2006)
- [18] Guoliang Cai, Juanjuan Huang, Lixin Tian, Qingchao Wang: Adaptive control and slow manifold analysis of a new chaotic system. *International Journal of Nonlinear Science*. 2(1),50-60 (2006)
- [19] Xuedi Wang, Lixin Tian, Liqin Yu: Linear feedback controlling and synchronization of the chen's chaotic system. *International Journal of Nonlinear Science*. 2(1),43-49 (2006)
- [20] Meifeng Dai, Xi Liu: Lipschitz equivalence between two sierpinski gasketse. *International Journal of Nonlinear Science*. 2(2),77-82 (2006)