The Hausdorff Measure of a Sierpinski Carpet

QILI XIAO, LIFENG XI

School of Computer Science and Information Technology

Zhejiang Wanli University

Ningbo, Zhejiang, 315100

P. R. CHINA

Abstract: - The Hausdorff measure computation of fractals is very difficult in fractal. In this paper, we present a novel method using the genetic algorithm to compute the Hausdorff measure of a Sierpinski carpet. The encoding method, decoding method and fitness computation are discussed in detail. The exact Hausdorff measure of the Sierpinski carpet is concluded through the implementation of the genetic algorithm.

Key-Words: Sierpinski carpet; Hausdorff measure; genetic algorithm

1 Introduction

Take a unit square in the Euclidean plane R^2 and denote it by F_0. Dividing each side of F_0 into four equal parts, sixteen equal small squares are got with length 1/4. Removing the interior of all small squares except for the four ones lying on the vertexes of F_0, we get a set denoted by F_1. If the above procedure is repeated for each small square in F_1, the set F_2 is obtained. Repeating the above procedure infinitely (such as Fig. 1), we have

$F_0 \supset F_1 \supset ... \supset F_k \supset ....$ The non-empty set $F = \cap_{m=0}^{\infty} F_m$

is called the Sierpinski carpet yielded by F_0. By [1, 2, 3], the Hausdorff dimension s of F is 1, and the Hausdorff measure $H(F)$ of F meets the following condition:

$$H(F) = \lim_{m \to \infty} \inf_{U \in F_m^{'}} \frac{|U|}{\mu(U)} \qquad (1)$$

where there are the following four mappings:

$\quad S_1(x) = x/4$

$\quad S_2(x) = x/4 + (3/4, 0)$

$\quad S_3(x) = x/4 + (3/4, 3/4)$

$\quad S_4(x) = x/4 + (0, 3/4)$

and $\quad F_{i_1 i_2 ... i_m} = S_{i_1} \circ S_{i_2} \circ ... \circ S_{i_m}(F_0)$, $F_m^{'} = \{U \mid$

U is a union of some small squares $F_{i_1 i_2 ... i_m}$ in the m-th structure$\}$.
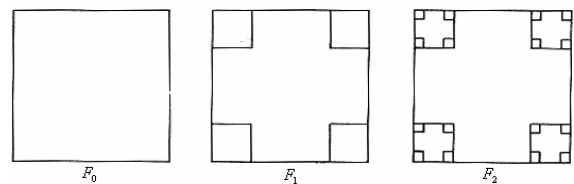

Fig. 1: the structure of the Sierpinski carpet

From the computing expression of $H(F)$, we know that it is very difficult to compute the exact of the Sierpinski carpet. The genetic algorithm, which is a self adaptive and global optimizing probability search algorithm inspired by evolution, is introduced and investigated by John Holland (1975) and by students of Holland (e.g., DeJong, 1975). The algorithm encodes a potential solution to a specific problem on a simple chromosome-like data structure and applies recombination operators to these structures so as to preserve critical information. The algorithm is robust, and the range of problems to which the genetic algorithm have been applied is quite broad [4, 5, 6]. Especially for the large complex nonlinear system, the genetic algorithm has particularly predominant capability.

In this paper, we will focus on the Sierpinski carpet with compression ratio 1/4. Section 2 discusses the encoding and decoding method, and fitness computation in detail. The experimental results and the future work will be given in Section 3.

2 Realization of Genetic Algorithm

2.1 encoding

During the process of calculating the Hausdorff measure, the method of binary code with fixed length is used. The encoding method is as follows:

if $F_{i_1 i_2 \dots i_m}$ is contained in the set $U \in F_m'$, then its corresponding code is 1, otherwise its corresponding code is 0. In this way, a code list of 4^m length can show whether each one of the 4^m equal small squares in the m-th structure is chosen or not. In order to solve the problem easily, the origin is regarded as the starting point here, and every small square is encoded according to the anticlockwise order.

Now let's state the detail encoding method with the second structure. In the second structure, there are $4^2=16$ equal small squares altogether. If we regard the origin as the starting point and encode every one according to the anticlockwise order, the marks of the 16 squares are as shown in Fig. 2. We choose every one of the 16 squares one by one according to their marks from small to large. If a square is chosen, we use 1 to represent on the corresponding position of the individual code, otherwise use 0. For example, 1001 0010 0100 0000 represents that four squares of the 16 squares in the second structure are chosen and their marks are 1, 4, 7 and 10 respectively (such as Fig. 2).
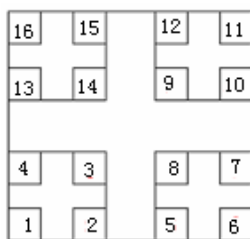

Fig. 2: small squares of the second structure

2.2 decoding

During the calculation of the Hausdorff measure, an individual represents a choice. Then it is the decoding method that any binary digit 1 of the individual represents which of the 4^m equal small squares is chosen, namely that the corresponding small square is the result which functions act on F_0.

The detail of decoding is as follows:

(1) Firstly, divide the whole code into four equal length groups, every small square that is represented by 0 or 1 in the first group is inside the square $S_1(F_0)$, and every small square that is represented by 0 or 1 in the second group is inside the square $S_2(F_0)$, and every small square that is represented by 0 or 1 in the third group is inside the square $S_3(F_0)$, and every small square that is represented by 0 or 1 in the forth group is inside the square $S_4(F_0)$.

(2) Secondly, respectively divide every group above into four equal length groups, every small square that is represented by 0 or 1 in the first group of every group above is respectively inside the square $S_1 \circ S_1(F_0)$ and $S_1 \circ S_2(F_0)$ and $S_1 \circ S_3(F_0)$ and $S_1 \circ S_4(F_0)$, and every small square that is represented by 0 or 1 in the second group of every group above is respectively inside the square $S_2 \circ S_1(F_0)$ and $S_2 \circ S_2(F_0)$ and $S_2 \circ S_3(F_0)$ and $S_2 \circ S_4(F_0)$, and the rest may be deduced by analogy.

(3) Finally, continue dividing every group of the second step into four equal length groups until there is only one digit 0 or 1 in every group.

2.3 calculation of fitness

Here fitness function is $F(U) = \mu(U)/|U|$, then finding minimum value is transformed into finding maximum value, where $\mu(U)$ represents the value of dividing the number of 1's in the individual code by 4^m, and $|U|$ represents the maximum distance of two random vertexes of eight vertexes of two random small squares. In order to calculate the maximum distance of any set U, a construction function series of every square in U must be found first, and then the coordinates of all vertexes of all small squares in U can be calculated.

Step 1: calculate the coordinates of all vertexes of all small squares. The detail calculation is as follows.

(1) Work out the construction function series of all small squares as introduced above.

(2) Calculate the coordinates of the left bottom vertex of all small squares. The detail is as follows:

Suppose A, B, C and D are four equal small squares in the m-th structure, and they are inside the square E that is one of the squares in the (m-1)-th structure (such as Fig. 3). If the coordinates of the left bottom vertex of the square E is (x_0, y_0), the coordinates of the left bottom vertex of small squares A, B, C and D are listed as follows:

$$(x_A, y_A) = (x_0 , y_0)$$
$$(x_B, y_B) = (x_0, y_0) + (3/4*l , 0)$$
$$(x_C, y_C) = (x_0, y_0) + (3/4*l , 3/4*l)$$
$$(x_D, y_D) = (x_0, y_0) + (0 , 3/4*l)$$

Where l represents the length of the side of the square E, and $l=1/4^{m-1}$.
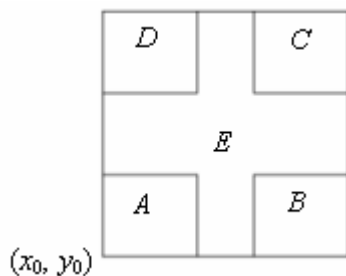

Fig. 3: square E and four small squares of the next structure

(3) Calculate the coordinates of the other vertexes of all small squares. The detail is as follows: If a small square is one of squares in the m-th structure and the coordinates of the left bottom vertex is (x, y), the coordinates of other vertexes are (x, y) + ($1/4^m$, 0), (x, y) + ($1/4^m$, $1/4^m$), (x, y) + (0, $1/4^m$) in anticlockwise order.

Remark All coordinates of all vertexes of all small squares are saved in a global variable whose name is "location" in turn in order to decrease the time of calculating individual fitness.

Step 2: Calculate the individual fitness.

(1) Find the coordinates of the small squares chosen in the current individual according to the result of step 1.

(2) Calculate the diameter $|U|$ and the number of 1's in the individual code.

(3) Calculate the corresponding fitness.

3 Experimental Results and Analysis

During the calculation of the Hausdorrf measure, the selection method is to preserve the best individual, the main crossover method is the one-point crossover, and the mutation method is the gene mutation. The values of the key parameters are the following: the crossover probability p_c=0.9, the mutation probability p_m=0.001, the population size *popsize*=50, the generation number *gen*=100. The initial population is generated randomly. According to above-mentioned ideas, we have programmed. The result of the experiment is what Table 1 shows.

Table 1: values of H (F)

m	$H(F)$	computation times (unit: second)
1	1.41421	1
2	1.41421	1
3	1.41421	1
4	1.41421	3
5	1.41421	120
6	1.41421	21106
7	1.41421	4200658

The main conclusions of this paper are the following.

(1) From Table 1, we can conclude that the Hausdorff measure of the Sierpinski carpet with compression ratio 1/4 H (F) =1.41421.

(2) It is predicated that the genetic algorithm will be an effective method to estimate the Hausdorff measure of fractals.

References:
[1] Xi Lifeng, Cen Zhongdi, Some Front Problems of Fractal- Computation of Hausdorff Measure,

Journal of Zhejiang Wanli University, Vol. 14, 2001, 1-3

[2] Zhou Zuoling, Hausdorff Measure of Sierpinski Carpet, *Science in China (A)*, Vol. l40, 1997, 1016-1021

[3] Zhou Zuoling, Hausdorff Measure of Self-similar set, *Science in China (A)*, Vol. 28, 1998, 103-107

[4] Zhou Ming, Sun Shudong, *Theory and Application of Genetic Algorithm*, National Defence Industry Press, 2002

[5] Chen Guoliang, Wang Xufa, *Genetic Algorithm and Its Application*, People Post Publishing Company, 2001

[6] Wang Xiaoping, Cao Liming, *Theory, Application and Software Realization of Genetic Algorithm*, Publishing Company of Xi'an Jiaotong University, 2002