Hybrid Genetic-Simulated Annealing Approach for Fractal Image Compression

Y.Chakrapani, K.Soundera Rajan

***Abstract*—**In this paper a hybrid technique of Genetic Algorithm and Simulated Annealing (HGASA) is applied for Fractal Image Compression (FIC). With the help of this hybrid evolutionary algorithm effort is made to reduce the search complexity of matching between range block and domain block. The concept of Simulated Annealing (SA) is incorporated into Genetic Algorithm (GA) in order to avoid pre-mature convergence of the strings. One of the image compression techniques in the spatial domain is Fractal Image Compression but the main drawback of FIC is that it involves more computational time due to global search. In order to improve the computational time along with acceptable quality of the decoded image, HGASA technique has been proposed. Experimental results show that the proposed HGASA is a better method than GA in terms of PSNR for Fractal image Compression.

***Keywords*—**Fractal Image Compression, Genetic Algorithm, HGASA, Simulated Annealing.

I. INTRODUCTION

COMPRESSION and decompression technology of digital image has become an important aspect in the storing and transferring of digital image in information society. Most of the methods in use can be classified under the head of lossy compression. This implies that the reconstructed image is always an approximation of the original image. Fractal image coding introduced by Barnsley and Jacquin [1-4] is the outcome of the study of the iterated function system 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. But the main drawback of their work can be related to large computational time for image compression. At present, researchers focus mainly on how to select and optimize the classification of the range blocks, balance the speed of compression, 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 have been proposed.

GA is a search and optimisation method developed by mimicking the evolutionary principles and chromosomal processing in natural genetics. Especially GA is efficient to solve nonlinear multiple-extrema problems [6-8] and is usually applied to optimize controlled parameters and constrained functions.

Simulated Annealing (SA) is a method for obtaining good solutions to difficult optimization problems which has received much attention over the last few years. The recent interest began with the work of Kirkpatrick and Cerny. They showed how a model for simulating the annealing of solids, as proposed by Metropolis could be used for optimization problems, where the objective function to be minimized corresponds to the energy of the states of the solid. Since then, SA has been applied to many optimization problems occurring in areas such as computer design, image processing and job scheduling. There has also been progress on theoretical results from a mathematical analysis of the method, as well as many computational experiments comparing the performance of SA with other methods for a range of problems.

Even though few investigations have been carried out involving GA and SA for fractal image compression [9], but till now a hybrid technique involving both GA and SA has not been proposed for FIC. Hence this paper proposes a hybrid technique which involves incorporation of SA into GA in order to avoid pre-mature convergence of strings.

The remainder of the paper is organized in detail as follows: Section (II) focuses on the theory of transformations and fractal image compression technique. In Section (III) the concept of GA is explained. Section (IV) explains the necessity of incorporating SA into GA. Section (V) explains the concept of SA and its incorporation into GA based FIC. The experimental results and discussions are furnished in Section (VI). In Section (VII) some conclusions are drawn.

II. FRACTAL IMAGE COMPRESSION

The fractal image compression algorithm is based on the fractal theory of self-similar and self-affine transformations.

A. Self-affine and Self-similar Transformations

In this section we present the basic theory involved in Fractal Image Compression. It is basically based on fractal theory of self-affine transformations and self-similar transformations. A self-affine transformation $W : R^n \to R^n$ is a transformation of the form $W(x) = T(x) + b$, where T is a linear transformation on R^n and $b \in R^n$ is a vector.

A mapping $W : D \to D, D \subseteq R^n$ is called a contraction on D if there is a real number $c, 0 < c < 1$ such that

Y.Chakrapani is with the Department of ECE, G.Pulla Reddy Engineering College, Kurnool-518002, Andhra Pradesh, India (phone: 09849595014; email: yerramsettychakri@ gmail.com).

K.Soundera Rajan is with the Department of ECE, J.N.T.U College of Engineering, Anantapur, Andhra Pradesh, India.

$d(W(x),W(y)) \le cd(x,y)$ for $x, y \in D$ and for a metric d on R^n. The real number c is called the contractivity of W.
$d(W(x),W(y)) = cd(x,y)$ then W is called a similarity.
A family $\{w_1,....w_m\}$ of contractions is known as Local Iterated function scheme (LIFS). If there is a subset $F \subseteq D$ such that for a LIFS $\{w_1,....w_m\}$

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

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.

$$A_\delta = \{x \in D : |x - a| \le \delta, a \in A\} \tag{2}$$

Let us define the distance $d(A,B)$ between two sets A, B to be $d(A,B) = \inf\{\delta : A \subset B_\delta \wedge B \subset A_\delta\}$
The distance function is known as the Hausdorff metric on S. We can also use other distance measures.
Given a LIFS $\{w_1,....w_m\}$, there exists an unique compact invariant set F, such that
$F = \bigcup_{i=1}^{m} w_i(F)$, this F is known as attractor of the system.
If E is compact non-empty subset such that $w_i(E) \subset E$ and

$$W(E) = \bigcup_{i=1}^{m} w_i(F) \tag{3}$$

We define the k-th iteration of W, $W^k(E)$ to be
$W^0(E) = E, W^k(E) = W(W^{(k-1)}(E))$. For $K \ge 1$ then we have that

$$F = \bigcap_{i=1}^{\infty} W^k(E) \tag{4}$$

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 easy 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.

B. Collage Theorm

If Let $\{w_1,....w_m\}$ be contractions on R^n so that

$|w_i(x) - w_i(y)| \le c|x - y|, \forall x, y \in R^n \wedge \forall i$,

Where $c < 1$. Let $E \subset R^n$ be any non-empty compact set. Then

$$d(E,F) \le d(E, \bigcup_{i=1}^{m} w_i(E)) \frac{1}{(1-c)} \tag{5}$$

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

C. Fractal Image Coding

The main theory of fractal image coding is based on iterated function system, attractor theorem and Collage theorem. Fractal Image coding makes good use 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.

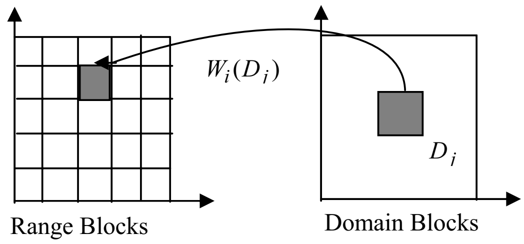

Fig. 1 Domain – Range block transformations

Fractal Image coding attempts to find a set of contractive transformations that map (possibly overlapping) domain cells onto a set of range cells that tile the image. The basic algorithm for fractal encoding is as follows

1. The image is partitioned into non overlapping range cells $\{R_i\}$ which may be rectangular or any other shape such as triangles. In this paper rectangular range cells are used.
2. The image is covered with a sequence of possibly overlapping domain cells. The domain cells occur in variety of sizes and they may be in large number.
3. For each range cell the domain cell and corresponding transformation that best covers the range cell is identified. The transformations are generally the affined transformations. For the best match the transformation parameters such as contrast and brightness are adjusted as shown in Fig 1
4. The code for fractal encoded image is a list consisting of information for each range cell which includes the location of range cell, the domain that maps onto that range cell and parameters that describe the transformation mapping the domain onto the range

One attractive feature of fractal image compression is that it is resolution independent in the sense that when decompressing, it is not necessary that the dimensions of the decompressed image be the same as that of original image.

III. Genetic Algorithm

Genetic algorithms are procedures based on the principles of natural selection and natural genetics that have proved to be very efficient in searching for approximations to global optima in large and complex spaces in relatively short time. The basic components of GA are:

- Representation of problem to be solved
- Genetic operators (selection, crossover, mutation)
- Fitness function
- Initialization procedure

GA starts by using the initialization procedure to generate the first population. The members of the population are usually strings of symbols (chromosomes) that represent possible solutions to the problem to be solved as shown in Fig 2

$$\underbrace{1\,1\,0\,1\,0}_{x_1}\ \underbrace{1\,0\,0\,1\,0\,0\,1}_{x_2}\ \underbrace{0\,1\,0}_{x_3}\ \text{------}\ \underbrace{0\,0\,1\,0}_{x_n}$$

Fig. 2 String representing 'N' variables

Each of the members of the population for the given generation is evaluated and according to its fitness value, it is assigned a probability to be selected for reproduction. Using this probability distribution, the genetic operators select some of the individuals. By applying the operators to them, new individuals are obtained. The mating operator selects two members of the population and combines their respective chromosomes to create offspring as shown in Fig 3. The mutation operator selects a member of the population and changes part of the chromosome as shown in Fig 4.

Parent A	0 0 **0 0 0 1 0 1**
Parent B	**1 1** 1 0 1 0 0 1
Child A	0 0 1 0 1 0 0 1
Child B	**1 1 0 0 0 1 0 1**

Fig 3: Binary crossover

Child A 0 1 **0** 1 1

$\downarrow$

New Child A 0 1 **1** 1 1

Fig. 4 Binary mutation

The algorithm for the genetic algorithm can be represented as follows

Step 1:As the genetic algorithm takes pairs of strings, we create a random number of strings depending upon our necessity and also note down their decoded values along with setting a maximum allowable generation number *t*max

Step 2:Using the mapping rule we next find out the corresponding values of above created strings.

Step 3: Using these values *the* fitness function values are found out.

Step 4: Next the process of reproduction is carried out on the strings to create a mating pool

Step 5: The process of crossover and mutation is also carried out on the strings with probabilities of 0.8 and 0.05 respectively.

Step6: After the termination criteria is met with, the value of string with minimum fitness function value is considered as optimum value.

IV. Necessity of Incorporation of SA

The performance of GA can be improved by introducing more diversity among the strings so that pre-mature convergence can be eliminated. This can be achieved by replacing weaker strings i.e. the strings having low fitness value with better strings i.e. strings having higher fitness value. Simulated Annealing may be used for this purpose.

V. Hybrid Genetic-Simulated Annealing Technique

Simulated Annealing method follows the cooling process of molten metals through annealing. At high temperature, the atoms can move freely and these movements get restricted to shape gradually a structure of crystal as the temperature is reduced slowly. If the temperature is not reduced properly, the system may end up in **a** polycrystalline state, which may have higher energy state than the crystalline state. Based on this annealing process, this algorithm begins with an initial point and a high temperature T. A second point is created at random .in the vicinity of the initial point and the difference in the function values (ΔE) at these two points is calculated. If the second point has a smaller function value, the point is accepted; otherwise the acceptance of that point is measured by Boltzmann probability distribution

In order to incorporate SA into GA, the strings resulted from GA after performing the mutation operation in every generation are modified by SA. New strings are generated from current strings via a perturbation mechanism. The similarity of the range block with the domain blocks computed from new string and current string are calculated. If the domain block constructed from the new string matches closely with the range block than the current string domain block, then the new string is carried to the next generation. If the new string is a deteriorated string, the probability of acceptance $P(T)$ is calculated as, $P(T) = \exp(-\Delta E / T)$ where ΔE is the difference between mean squared error of current string domain block and new string domain block when compared with the range block. T is the current temperature at which the new solution is generated. A random number between 0 and 1 is then generated. If the value of $P(T)$ is greater than or equal to the random number, the new string is passed to the next iteration; otherwise, if $P(T)$ is less than random number, the move is discarded and the current string is carried to the next iteration. This completes one iteration and in the next iteration, the temperature is lowered and the acceptance or rejection of the new strings selected by perturbation mechanism is decided by the same process.

The flowchart of HGASA based FIC is as shown in Fig 5.

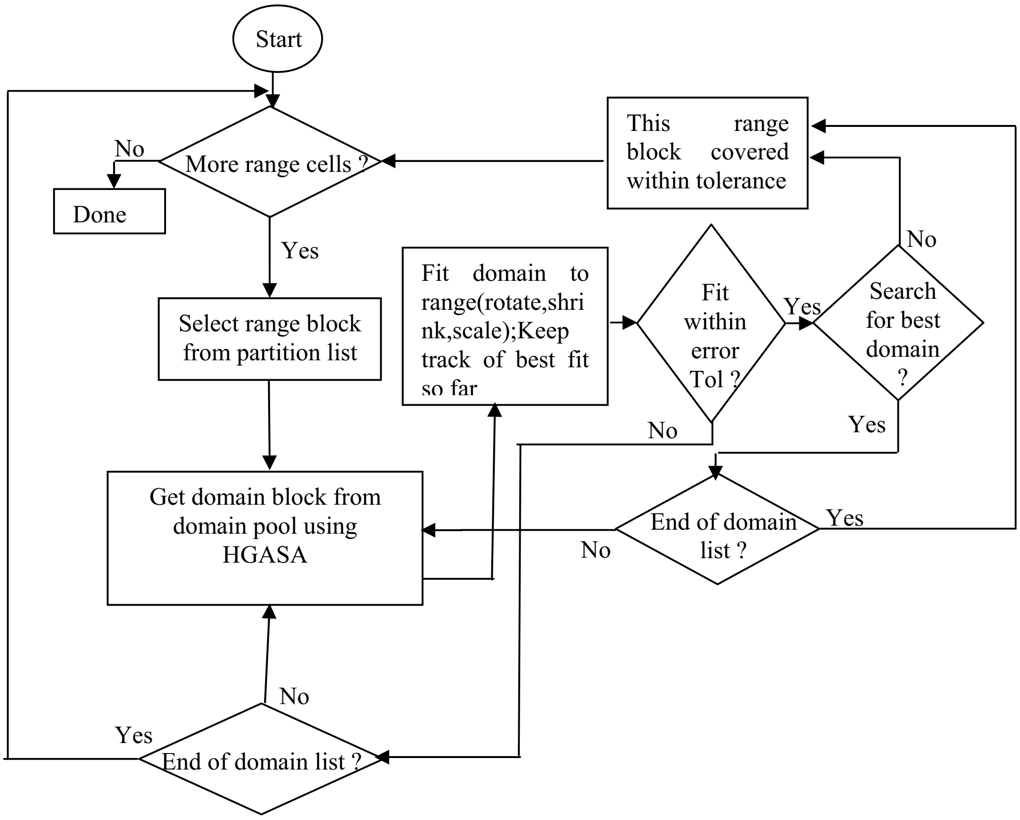

Fig. 5 Flow Chart of HGASA based Fractal Image Encoding

VI. Results and Discussions

In this paper a gray level image of 256×256 size with 256 gray levels is considered. A Range block of size 4×4 and Domain blocks of size 8×8 are considered. The domain blocks are mapped to the range block by affine transformations and the best domain block is selected using the above proposed technique. In the above technique the individual chromosome is coded as shown in Fig 6

$$\underbrace{1\,1\,0\,1\,0\,1\,0\,1}_{x}\ \underbrace{1\,0\,0\,1\,0\,0\,1\,1}_{y}\ \underbrace{0\,1\,0}$$

Domain block coordinates Isometry type

Fig 6: Binary code for searching domain block

The mean squared error (MSE) and PSNR considered in this work are given by

$$\text{MSE}= \frac{1}{N_{Rows} N_{Cols}} \sum_{i=1}^{N_{Rows}} \sum_{j=1}^{N_{Cols}} \left| f_{i,j} - d_{i,j} \right|^2 \tag{6}$$

$$\text{PSNR} = 10 \log_{10}\left(\frac{255^2}{MSE} \right) \tag{7}$$

Where $N_{Rows} = N_{Cols} = 256$

This work is carried out in MATLAB 7.0 version and the original image is classical 256×256 Lena and Barbara face image coded with 8 bits per pixel. An optimal bit allocation strategy for GA is as follows: 14 bits for the location of matched domain block (horizontal and vertical coordinate), 3 bits for isomorphic types. For each of the range block fractal coding includes 17 bits allocation. During the iteration process of the above proposed methods the gray level values beyond 0 and 255 are replaced by average of its four neighbours to avoid block diverging. Figs 7 and 8 show the reconstructed images using FIC with GA and HGASA as search algorithms along with the original images of Barbara and Lena. The various parameters of HGASA are shown in Table 1.The Coding scheme of FIC using GA and SA is given in Table 2.

As seen from Table 2 the PSNR of the reconstructed image using HGASA has a better value than the image reconstructed through normal GA. Since the proposed technique involves two evolutionary algorithms joined together, the computational time of HGASA is slightly more than that of normal GA. But this can be compensated through the improvement in the reconstructed image quality. So it can be concluded that the proposed method can be used in all applications of image compression where the quality of reconstructed image is demanding.

(a) (b) (c)

Fig. 7 (a) original image (b) reconstructed image using GA (c) reconstructed image using HGASA

(a) (b) (c)

Fig. 8 (a) original image (b) reconstructed image using GA (c) reconstructed image using HGASA

TABLE I HGASA PARAMETERS

Range block Size	4*4	Population size	50
Crossover Probability Pc	0.8	Fitness f	1/(1+MSE)
Mutation Probability	0.05	Temperature	5000

TABLE II FIC CODING SCHEME COMPARISON USING HGASA AND GA

	Image	FIC with HGASA	FIC with GA
Compression Ratio	Barbara	6.73:1	6.73:1
	Lena	6.73:1	6.73:1
PSNR (db)	Barbara	28.89	28.34
	Lena	28.86	26.22
Encoding Time (sec)	Barbara	5580	4470
	Lena	5390	4230

VII. CONCLUSION

The concept of HGASA has been successfully applied to fractal image compression of a gray level image. Instead of traditional exhaustive search which takes large computational time in FIC the evolutionary computational technique like GA is implemented which goes for random search thus reducing the computational time. In order to improve the performance of GA in terms of PSNR, a new technique which combines the concept of genetic algorithm and simulated annealing has been proposed. Experimental results show that the proposed HGASA scores over normal GA in the case of fractal image compression. Normally the PSNR ratio for a decoded image should be very high to have a better image. Based on Table 3 it can be seen that the PSNR is better in the case of decoded image using HGASA over the one obtained by GA.

REFERENCES

[1] A.E.Jacquin, ‘Image coding Based on a Fractal theory of Iterated contractive Image Transformations,’ IEEE Transactions on Image Processing. vol 1, Jan 92, pp 18-30

[2] A.E.Jacquin, “Fractal Image coding: A Review,” Proc. IEEE, vol 81, pp. 1451-1465, 1993.

[3] M.F.Barnsley, Fractals Everywhere. New York: Academic 1988.

[4] M.F Barnsley and A.E Jacquin, “Application of recurrent iterative function systems to images”, Proc SPIE, 1001(3),122-131 (1998)

[5] Mingshui Li, Shanhu Ou and Heng Zhang, “The new progress in research approach of fractal image compression” journal of engineering graphics, 4(3), 2004, pp 143-152 .

[6] Xiaoping Wang, Liming Cao, “Genetic Algorithms-theory, application and software reliability Xi” An Jiao Tong University Press (2002).

[7] David Goldberg, "Genetic algorithms in search, optimization and machine learning" Addison-Wesley Publishing Company: MA (1989).
[8] Kalyanmoy Deb, "Optimization for engineering design" Prentice Hall of India,2000.
[9] Y.Chakrapani, K.Soundera Rajan, "A comparative approach to fractal image compression using genetic algorithm and simulated annealing technique" Asian Journal of Information technology 7(7), pp 285-289.
[10] M.Hassaballah, M.M.Makky, Youssuf B. Mahdy, "A fast fractal image compression method based entropy" Electronic letters on computer vision and image analysis, 5(I) 2005, pp 30-40.
[11] Zhang Chao, Zhou Yiming, Zhang Zengke, "Fast fractal image encoding based on special image features" Journal of Tsinghua science and technology, Vol12, No 1, 2007,pp 58-62

Y.Chakrapani is presently with the Department of Electronics and Communications, G.Pulla Reddy Engineering college, Kurnool, A.P, India. Currently he is pursuing Ph.D at JNT University under the guidance of Dr K.Soundera Rajan. His research areas are image compression, image retrieval and evolutionary algorithms.

Dr K.Soundara Rajan , born on April 1953, obtained Post graduation from JNTU, Hyderabad and Ph.D from Indian Institute of Technology, Roorkee. He has been with JNT University for the past 22 years. Prof K.Soundara Rajan is presently the Principal of JNT College of Engg, Anantapur. He has published 45 papers in various reputed International and National Journals. His areas of Interest are Microcomputers, Instrumentation, Computer Operating systems and Digital Image Compression. He received Best Teacher award in 2006 by the State government of Andhra Pradesh. He is also member of many expert committees.