

Act of CVT and EVT in the Formation of Number Theoretic Fractals

Pabitra Pal Choudhury, Sk. Sarif Hassan
Applied Statistics Unit,
Indian Statistical Institute, Kolkata, 700108, INDIA
Email: pabitrpalchoudhury@gmail.com sarimif@gmail.com

Sudhakar Sahoo, Birendra Kumar Nayak
P.G. Department of Mathematics,
Utkal University, Bhubaneswar-751004, INDIA
Email: sudhakar.sahoo@gmail.com
bknatuu@yahoo.co.uk

Abstract— In this paper we have defined two functions that have been used to construct different fractals having fractal dimensions between 1 and 2. More precisely, we can say that one of our defined functions produce the fractals whose fractal dimension lies in $[1.58, 2)$ and rest function produce the fractals whose fractal dimension lies in $(1, 1.58]$. Also we tried to calculate the amount of increment of fractal dimension in accordance with base of the number systems. And in switching of fractals from one base to another, the increment of fractal dimension is constant, which is 1.58, it's quite surprising!

Keywords-Carry Value Transformation, Extreme Value Transformation, Fractals, Fractal dimension.

We have constructed two functions namely Carry Value transformation and Extreme Value Transformation; and on using these we have generated different fractals of fractal dimension lying between the interval (1, 2). Indeed, for any given number in between (1, 2) this functions could give one fractal having fractal dimension nearer to the given number. The maximum error could be 0.12 in approximating fractal dimension. Finally our observation centers on the fact that when we switch from one fractal to another fractal on using our said methodology, the amount of fractal dimension (amount of chaos) is remaining unaltered as 1.58.

I. INTRODUCTION

In this paper we have tried to make an association between natural number and fractals, with the help of two defined transformations. Numbers corresponding to different number systems with base b ($b=2, 3, 4, \dots$ etc) signifies different fractals. Here, we have defined two new transformations called as “Carry Value Transformation (CVT) and Extreme Value Transformation (EVT)”. With these two mapping we have generated fractals whose dimension lying in between the open interval $(1, 2)$ [figure 7, 14]. It should be noted that we are traversing this interval discretely, but very densely also. That is for any given number from $(1, 2)$ we could be able to give a fractal whose dimension is nearer to that given number. In our journey we have got a fractal whose dimension is 1.68, fortunately this very fractal dimension is the fractal dimension of music (Sri Lankan, Chariots of Fire) [7]. So this fractal could be a frame of lyrics for music. And undoubtedly there are a lot of fractals, which are of fractal dimension 1.68, and possibly these fractal-frames make new lyrics. In general, the fractal dimension of music is around the number 1.65, and our generated fractals could interpolate the number 1.65, as fractal dimension. In this paper also we have tried to calculate the amount of increment of fractal dimension in accordance with base of the number systems. And in switching of fractals from one base to another, the increment of fractal dimension is constant, which is 1.58, fractal dimension of Sierpinski Gasket.

The organization of the paper is as follows. Section 2 discusses some of the basic concepts on fractals, fractal dimension, which are used in the subsequent sections. The concept of CVT is defined in section 3. In section 4, we have explored the formation of fractals in different bases of the number systems. And then we have generalized the concept of formation of fractals in any arbitrary bases of the number system. In section 5, we have explained the amount of increment of fractal dimension in accordance with base of the number systems.

In the next sections, we have defined another transformation to generate fractals having fractal dimension lying in between (1, 1.58]. In addition, we have discussed formation of fractals in different bases of the number system and ultimately we have made it-generalized concept in any base of the number system as like for CVT we have done it. On highlighting other possible applications of CVT and some future research directions a conclusion is drawn in section 8.

II. REVIEW OF SOME FUNDAMNETALS OF FRACTALS

For quite a long time the scientific community was very much worried due to our inability to describe the shape of cloud, a mountain, a coastline or a tree on using the traditional Euclidean Geometry. In nature, clouds are not really spherical, mountains are not conical, coastlines are not circular, even the lightning doesn't travel in a straight line. More generally, we would be able to conclude that many patterns of nature are so irregular and fragmented, that, compared with *Euclid Geometry* –a term, can be used in this regard to denote all of the standard geometry. Mathematicians have over the years disdained this challenge and have increasingly chosen to flee nature by devising theories unrelated to natural objects we can see or feel. After a long time, responding to this challenge, Benoit Mandelbrot developed a new geometry of nature and implemented its use in a number of diverse arenas of science such as Astronomy, Biology, Mathematics, Physics, and Geography and so on [1, 2,3,4]. This new-born geometry can describe many of the irregular and fragmented (chaotic) patterns around us, and leads to full-fledged theories, by identifying a family of shapes, now-a-days which we people call 'FRACTALS'.

Fractal dimension

The fractal dimension alone does not give an idea of what “fractals” are really about Mandelbrot founded his insights in the idea of self similarity, requiring that a true fractal “fracture” or break apart into smaller pieces that resemble the whole. This is a special case of the idea that there should be a dynamical system underlying the geometry of the set. This is partly why the idea of fractals have become so popular throughout science; it is a fundamental aim of science to seek to understand the underlying dynamical properties of any natural phenomena. With this view, self similar fractals has been playing a dominant role, however it has now become more clearer that apart from self similarity,. Dynamical systems can produce many intricate shapes and behavior that occur throughout nature. There are different methods to calculate the fractal dimension of an object. But here we are only concentrating on Similarity dimension. Now let us try to define what fractal dimension (Similarity dimension) is. Given a self-similar structure [figure 3], there is a relation between the reduction factor (scaling factor) ‘S’ and the number of pieces ‘N’ into which the structure can be divided; and that relation is as follows...

$$N = 1/S^D, \text{ equivalently,}$$

$$\text{i.e. } D = \log(N) / \log(1/S)$$

This ‘D’ is called the Fractal dimension (Self-similarity dimension).

III. CARRY VALUE TRANSFORMATION (CVT)

The carry or overflow bits are usually generated at the time of addition between two n-bit strings. In the usual addition process, carry value is always a single bit and if generated then it is added column wise with other bits and not saved in its own place. But the carry value defined here are the usual carries generated bit wise and stored in their respective places as shown in fig 1.

$$\begin{array}{r} \text{carry value} = c_n \ c_{n-1} \dots\dots\dots c_1 \ 0 \\ a = \quad a_n \quad a_{n-1} \dots\dots\dots a_1 \\ b = \quad b_n \quad b_{n-1} \dots\dots\dots b_1 \\ a \oplus b = a_n \oplus b_n \ a_{n-1} \oplus b_{n-1} \dots\dots\dots a_1 \oplus b_1 \end{array}$$

Figure 1. Carry generated in ith column saved in (i-1)th column

Thus to find out the carry value we perform the bit wise XOR operation of the operands to get a string of sum-bits (ignoring the carry-in) and simultaneously the bit wise ANDing of the operands to get a string of carry-bits, the latter string is padded with a ‘0’ on the right to signify that there is no carry-in to the LSB. Thus the corresponding decimal value of the string of carry bits is always an even integer.

Now we can give a precise definition of CVT as follows:

Let $B = \{0,1\}$ and CVT is a mapping defined as $CVT: (B_n \times B_n) \rightarrow B_{n+1}$ where B_n is the set of strings of length n on $B = \{0,1\}$. More specifically, if $a = (a_n, a_{n-1}, \dots, a_1)$ and $b = (b_n, b_{n-1}, \dots, b_1)$ then $CVT(a, b) = (a_n \wedge b_n, a_{n-1} \wedge b_{n-1}, \dots, a_1 \wedge b_1, 0)$ is an $(n+1)$ bit string, belonging to \mathbb{Z} , set of non-negative integers, and can be computed bit wise by logical AND operation followed by a 0, which denotes no carry is generated in the LSB at the time of addition procedure. In other words, CVT is a mapping from $\mathbb{Z} \times \mathbb{Z} \rightarrow \mathbb{Z}$, where \mathbb{Z} is set of non-negative integers [7].

Example:

Suppose, we want to get the CVT of the numbers $(13)_{10} \equiv (1101)_2$ and $(14)_{10} \equiv (1110)_2$. Both are 4-bit numbers. The carry value is computed as follows:

Carry: 1 1 0 0 0

Augend: 1 1 0 1
 Addend: 1 1 1 0
 XOR: 0 0 1 1

Figure 2. Carry generated in ith column saved in (i-1)th column

Conceptually, in the general addition process the carry or overflow bit from each stage (if any) goes to the next stage so that, in each stage after the first (i.e. the LSB position with no carry-in), actually a 3-bit addition is performed instead of a 2-bit addition by means of the full adder. Instead of going for this traditional method, what we do is that we perform the bit wise XOR operation of the operands (ignoring the carry-in of each stage from the previous stage) and simultaneously the bit wise ANDing of the operands to get a string of carry-bits, the latter string is padded with a '0' on the right to signify that there is no carry-in to the LSB (the overflow bit of this ANDing being always '0' is simply ignored). In our example, bit wise XOR gives $(0011)_2 \equiv (3)_{10}$ and bit wise ANDing followed by zero-padding gives $(11000)_2 \equiv (24)_{10}$. Thus $CVT(1101,1110) = 11000$ and equivalently in decimal notation one can write $CVT(13,14) = 24$. In the next section we have used the carry value in decimal to construct the CV table.

IV. GENERATION OF SELF-SIMILAR FRACTAL USING CVT IN DIFFERENT BASES OF THE NUMBER SYSTEM

A table is constructed that contains only the carry values (or even terms) defined above between all possible integers a's and b's arranged in an ascending order of x and y-axis respectively. We observe some interesting patterns in the table. We would like to make it clear how the CV-table is constructed.

Step 1. Arrange all the integers 0 1 2 3 4 5 6 ... (as long as we want) in ascending order and place it in both, uppermost row and leftmost column in a table.

Step 2. Compute $CVT(a,b)$ as mentioned in 3.1 and store it in decimal form in the (a, b) position.

Then we look on the pattern of any integer, and we have made it color. This shows a very beautiful consistent picture, which we see as a fractal as by shown below.

A. Generation of Fractals

Let us do find the fractals in different domain of number system with the help of CVT.

• Production of Fractal in Binary(2-nary) Number System

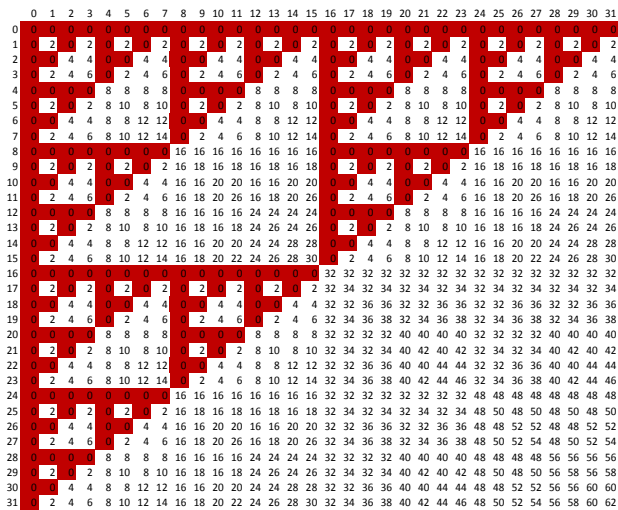


Figure 3. A Fractal Structure on using CVT of Different Integer Values in Binay Number System

Dimension of this fractal

For this fractal [see figure 3], number of self-similar copies $N=3$ and scaling factor $S=1/2$, where l is the initial length. So, referring the discussion in 2.1, Fractal dimension D is given by

$$3 = 1 / (1/2)^D$$

Or $D = \log 3 / \log 2 \approx 1.585$

This is same as the dimension of Sierpinski triangle. Thus CVT fractal as obtained by us can be regarded as a relative to *Sierpinski* triangle [6].

- *CV Table in Ternary Number(3-nary) System*

Here we are applying CVT on the domain of ternary (3-nary) number system and we are having the following table. It is mentioned that the procedure can be verbatim copied just by replacing binary by ternary (3-nary), as we have discussed earlier to construct CVT table in binary number system. Let us demonstrate how we are computing CVT of two numbers in ternary number system.

Suppose, we want to get CVT of the numbers $(13)_{10} \equiv (111)_3$ and $(14)_{10} \equiv (112)_3$. Both are 3-digit numbers. The carry value is computed as follows:

Carry: 0 0 1 0

Augend: 1 1 1

Addend: 1 1 2

Addition process in ternary: 2 2 0

Therefore, $CVT(13,14) = (0010)_3 = 3$.

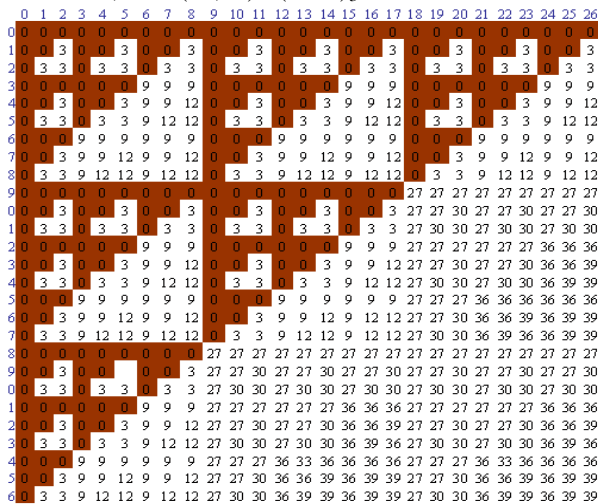


Figure 4. A Fractal Structure on using CVT of Different Integer Values in Ternary Number System

Dimension of this fractal

For this fractal [see figure 4], number of self-similar copies $N=3$ and scaling factor $S=1/2$, where l is the initial length.

So, referring the discussion in 2.1, Fractal dimension D is given by

$$6 = 1/(1/3)^D$$

$$\text{Or } D = \log 6 / \log 3 \approx 1.630929$$

- *CV Table in 4-nary System*

Here we are applying CVT on the domain of 4-nary number system and we are having the following table. It is mentioned that the procedure can be verbatim copied just by replacing binary by 4-nary, as we have discussed earlier to construct CVT table in binary number system.

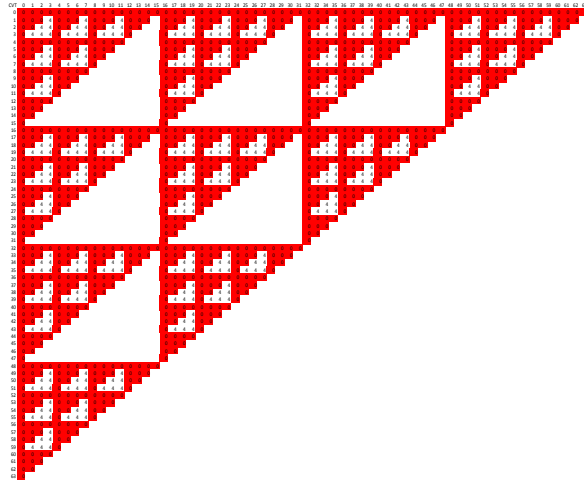


Figure 5. A Fractal Structure on using CVT of Different Integer Values in 4-nary Number System

Dimension of this pattern

For this fractal [see figure 5], number of self-similar copies $N=10$ and scaling factor $S=1/2$, where l is the initial length. So, referring the discussion in 2.1, Fractal dimension D is given by

$$10 = 1 / (1 / 2)^D$$

$$\text{Or } D = \log 10 / \log 2 = 1.6609$$

- *CV Table in 5-nary System*

Here we are applying CVT on the domain of ternary (3-nary) number system and we are having the following table. It is mentioned that the procedure can be verbatim copied just by replacing binary by ternary (3-nary), as we have discussed earlier to construct CVT table in binary number system.

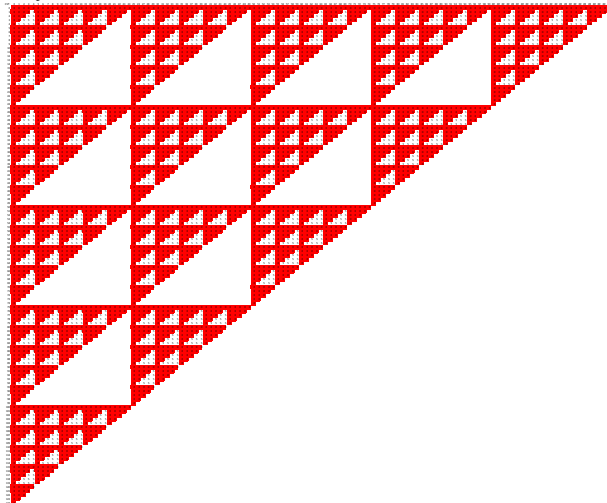


Figure 6. A Fractal Structure on using CVT of Different Integer Values in 5-nary Number System

Dimension of this pattern

For this fractal [see figure 6], number of self-similar copies $N=15$ and scaling factor $S=1/2$, where l is the initial length. So, referring the discussion in 2.1, Fractal dimension D is given by

$$15 = 1 / (1 / 2)^D$$

$$\text{Or } D = \log 15 / \log 2 = 1.682606$$

It should be noted that fractal dimension of music (Sri Lankan, Chariots of Fire) is also 1.68[7]. So this fractal could be a frame of lyrics for music. In general, fractal dimension of arbitrary music is nearly 1.65. And undoubtedly there are a lot of fractals which are of fractal dimension 1.68, and possibly these fractal-frames make new lyrics.

Similarly, it could be possible to generate fractals in any base of the number system.

B. Generalization of the concepts in Arbitrary Base of the Number Systems

Now, we are warmed up to construct the fractal in different number system just following the above procedure. It is also cleared that, when we have calculated the similarity dimension of those fractal in each case we have observed the scaling factor is $1/n$ and self-similar copies is $(1+2+3+\dots+n)$, for n -nary number system. So let us define our self-similarity dimension formula as follows...

Here, Number of self – similar copies,

$$N = \sum_{j=1}^n j, \text{ \& Scalling, } S = \frac{1}{n}, \text{ for } n - \text{nary number system}$$

$$\text{Fractal dimension, } S_D(n) = \left\{ \frac{\log(\sum_{j=1}^n j)}{\log n} \right\}$$

$$\text{i.e. } S_D(n) = \left\{ \log\left(\frac{n(n+1)}{2}\right) / \log n \right\}$$

Theorem 1: The fractal dimension S_D converges to the topological dimension (Euclidian dimension) 2 as the base ‘ n ’ of the number system diverges to infinity.

Proof: Let us try to find the limit when n tends to infinity,

$$\begin{aligned} \lim_n S_D(n) &= \lim_n \left\{ \log\left(\frac{n(n+1)}{2}\right) / \log n \right\} \left(\frac{\infty}{\infty} \text{ form} \right) \\ &= \lim \left\{ \frac{\left[\frac{2(2n+1)}{2n(n+1)} \right]}{\frac{1}{n}} \right\}; \text{ by L'hospital rule} = \lim \left\{ \frac{2n^2+n}{n^2+n} \right\} = 2 \end{aligned}$$

So, starting from the binary number system the fractal dimension of the generated fractal will go on increasing with the increase of the base of the number system and finally it converges to the topological dimension 2.

Let us consider a list of such dimensions...

Table 1: Shows a table of fractal dimensions of fractals according as base of the number system

Base of the number system	Fractal dimension of the obtained fractal
2	1.584962501
3	1.630929754
4	1.660964047
5	1.682606194
6	1.699180325
7	1.712414374
8	1.723308334
9	1.73248676
10	1.740362689
11	1.747221736
12	1.75326861
13	1.758654413
14	1.763493463

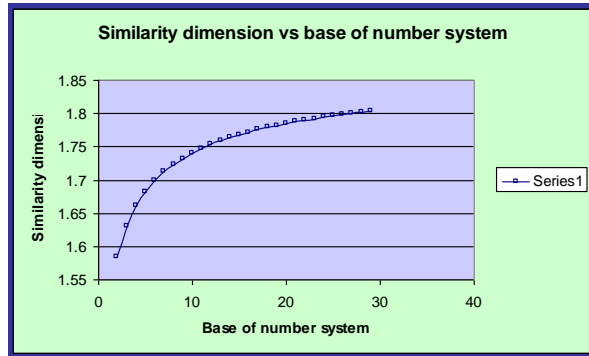


Figure 7. Shows a graph how fractal diemnsion increases in accordance with base of the number system, using table 1

V. INCREMENT OF FRACTAL DIMENSION OF CVT FRACTALS

Let us try to obtain the increment of fractal dimension of the CVT fractals in switching from one base to another base of the number system. In binary number system we have the fractal of fractal dimension 1.58, and in ternary number system we have the fractal of fractal dimension 1.63. So algebraically, it seems the increment of fractal dimension is $(1.63-1.58) = 0.05$. But it is really not!

To obtain the increment we proceed as follows...

First of all we paste one $(n-1)$ -nary CVT fractal to n -nary CVT fractal. Next, the overflowed portion is extracted which can be seen as a self-similar figure. These self-similar pieces derived from overflowed portion would lead to another fractal with some fractal dimension. What we observe, this very measure (fractal dimension) is the actual increment of fractal dimension in switching from one base to another base of the number system.

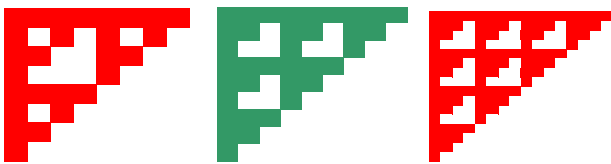


Figure 8. Shows Binary CVT Fractal generetors in Binary, Ternary, Four-nary form left to right.

A. Binary Generator over Ternary Generator

Here we paste binary CVT fractal over ternary CVT fractal [8(a),8(b)]. Next, the overflowed portion can be seen in figure 9, which is a self-similar figure as the CVT fractals were self-similar. These self-similar pieces derived from overflowed portion would lead to another fractal with fractal dimension 1.58 as shown below.

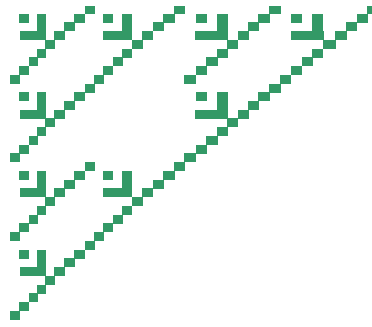


Figure 9. Binary CVT generator over Ternary generator lead to the another fractal.

This generator leads to another fractal; let us calculate the fractal dimension of the generated fractal as follows...

Dimension of this fractal

For this fractal, $N=3$, $S=1/2$, where l is the initial length.

$$\text{Fractal dimension } D \text{ is given by } \dots \quad 3 = 1/(1/2)^D$$

$$\text{Or } D = \log 3 / \log 2 \approx 1.585$$

This is same as the dimension of Sierpinski triangle. Thus CVT fractal as obtained by us can be regarded as a relative to *Sierpinski* triangle.

Therefore, we could say that the amount of increment is likely 1.58.

B. Ternary Generator over Four-nary Generator

Here we paste ternary CVT fractal over 4-nary CVT fractal [8(b),8(c)]. Next, the overflowed portion can be seen in figure 10, which is a self-similar figure as the CVT fractals were self-similar. These self-similar pieces derived from overflowed portion would lead to another fractal with fractal dimension 1.58 as shown below.

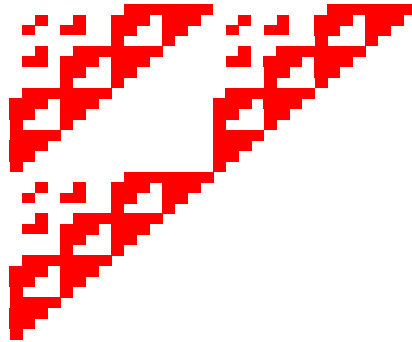


Figure 10. Ternary CVT generator over Four-nary generator lead to the another fractal.

This generator also leads to another fractal; let us calculate the fractal dimension of the generated fractal as follows...

Dimension of this fractal

For this fractal, $N=3$, $S=1/2$, where l is the initial length.

$$\text{Fractal dimension } D \text{ is given by } \dots \quad 3 = 1/(1/2)^D$$

$$\text{Or } D = \log 3 / \log 2 \approx 1.585$$

This is same as the dimension of Sierpinski triangle. Thus CVT fractal as obtained by us can be regarded as a relative to *Sierpinski* triangle. Here also we are having the increment same as above.

In fact, in general we could make a conjecture that if we like to paste an n -nary CVT generator over an $(n+1)$ -nary CVT generator, then we will be able to have another generator which led to the fractal of fractal dimension 1.58, i.e. the attractor fractal is Sierpinski Gasket.

So, the amount of increment is constant, which is 1.58.

So far we have discussed we are in position to achieve the fractals having fractal dimension lying in between [1.58, 2), with the help of CVT. But let us try to define another map named as Extreme Value Transformation (EVT) to achieve fractals whose fractal dimension lying in between (1, 1.58].

VI. EXTREEM VALUE TRANSFORMATION(EVT)

Now we can give a precise definition of EVT as follows:

Let $B = \{0,1\}$ then EVT is a mapping defined as $E_{\max}VT: B_n \times B_n \rightarrow B_n$ where B_n is the set of strings of length n on $B = \{0,1\}$.

More specifically, if $a = (a_n, a_{n-1}, \dots, a_1)$ and $b = (b_n, b_{n-1}, \dots, b_1)$ then

$E_{\max}VT: Z \times Z \rightarrow Z$ where a, b being non-negative integers.

$$E_{\max}(a, b) = ((\max(a_n, b_n), (\max(a_{n-1}, b_{n-1}) \dots \max(a_1, b_1)))$$

is an n bit string and can be computed bit wise maximum.

Now, people may have in confusion that why we have concentrate on bit-wise maximum in defining EVT. But we like to make you clear that if we use the bit wise minimum then also we will be having same fractal if we ignore the orientation (each of them what we are getting $E_{\max}VT$). That is we are producing a pair of functions to produce the same fractals.

Example:

Suppose, we want the EVT of the numbers $(13)_{10} \equiv (1101)_2$ and $(14)_{10} \equiv (1110)_2$. Both are 4-bit numbers. The extreme value is computed as follows...

$$\begin{aligned} 13 &= 1101 \\ 14 &= 1110 \\ \text{EVT}(13, 14) &= 1110 \end{aligned}$$

VII. GENERATION OF SELF-SIMILAR FRACTAL USING EVT IN DIFFERENT BASES OF THE NUMBER SYSTEM

A table is constructed that contains only integer defined above between all possible integers a's and b's arranged in an ascending order of x and y-axis respectively. We observe some interesting patterns in the table. We would like to make it clear how the EV-table is constructed.

Step-1. Arrange all the integers 0 1 2 3 4 5 6 ... (as long as we want) in ascending order and place it in both, uppermost row and leftmost column in a table.

Step-2. Compute EVT (a, b) as mentioned in 6.1 and store it in decimal form in the (a, b) position.

Then we look on the pattern of any integer, and we have made it color. This shows a very beautiful consistent picture, which we see as a fractal as by shown below.

A. Generation of Fractals

Let us do find the fractals in different domain of number system with the help of EVT.

- Production of Fractal in Binary(2-nary) Number System

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	1	3	3	5	5	7	7	9	9	11	11	13	13	15	15
2	2	3	2	3	6	7	6	7	10	11	10	11	14	15	14	15
3	3	3	3	3	7	7	7	7	11	11	11	11	15	15	15	15
4	4	5	6	7	4	5	6	7	12	13	14	15	12	13	14	15
5	5	5	7	7	5	5	7	7	13	13	15	15	13	13	15	15
6	6	7	6	7	6	7	6	7	14	15	14	15	14	15	14	15
7	7	7	7	7	7	7	7	7	15	15	15	15	15	15	15	15
8	8	9	10	11	12	13	14	15	8	9	10	11	12	13	14	15
9	9	9	11	11	13	13	15	15	9	9	11	11	13	13	15	15
10	10	11	10	11	14	15	14	15	10	11	10	11	14	15	14	15
11	11	11	11	11	15	15	15	15	11	11	11	11	15	15	15	15
12	12	13	14	15	12	13	14	15	12	13	14	15	12	13	14	15
13	13	13	15	15	13	13	15	15	13	13	15	15	13	13	15	15
14	14	15	14	15	14	15	14	15	14	15	14	15	14	15	14	15
15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15	15

Figure 11. A Fractal Structure on using EVT of Different Integer Values in Binay Number System

Dimension of this fractal

For this fractal [Figure 11], $N=3$, $S=1/2$, where l is the initial length.

$$7=1/(1/4)^D$$

$$\text{Or } D = \log 7 / \log 4 = 1.403677$$

VIII. GENERATION OF SELF-SIMILAR FRACTAL USING CVT IN DIFFERENT BASES OF THE NUMBER SYSTEM

Now, we are warmed up to construct the fractal in different number system just following the above procedure. It is also cleared that, when we have calculated the similarity dimension of those fractal in each case we have observed the scaling factor is $1/n$ and self-similar copies is $(1+2+3+\dots+n)$, for n -nary number system. So let us define our self-similarity dimension formula as follows...

Here, Number of self – similar copies,

$$N = n + (n - 1) = 2n - 1$$

$$S = \frac{1}{n}, \text{ for } n - \text{nary number system}$$

$$\text{Fractal dimension, } S_D(n) = \left\{ \frac{\log(2n - 1)}{\log n} \right\}$$

Theorem 2: The fractal dimension S_D converges to the topological dimension (Euclidian dimension) 1 as the base ‘ n ’ of the number system, diverges to infinity.

Proof:

Let us try to find the limit when n tends to infinity,

$$\lim_n S_D(n) = \lim_n \left\{ \frac{\log(2n-1)}{\log n} \right\} \left(\frac{\infty}{\infty} \text{ form} \right)$$

$$= \lim \left\{ \frac{\left[\frac{2}{(2n-1)} \right] / \frac{1}{n}}{\frac{1}{n}} \right\} \text{ by L'Hospital rule}$$

$$= \lim \left\{ \frac{2n}{2n-1} \right\} = 1$$

$$i. e. \lim_{n \rightarrow \infty} S_D(n) = 1$$

So, starting from the binary number system the fractal dimension of the generated fractal will go on decreasing with the increase of the base of the number system and finally it converges to the topological dimension 1.

Let us consider a list of such dimensions...

Table 2: Shows a table of fractal dimensions of fractals according as base of the number system

Base of the number system	Fractal dimension of the obtained fractal
2	1.584962501
3	1.464973521
4	1.403677461
5	1.365212389
6	1.338290833
7	1.318123223
8	1.302296865
9	1.289450962
10	1.278753601
11	1.269664473
12	1.261815697
13	1.254947126
14	1.248868992
15	1.24343922
16	1.238549078
17	1.234113756
18	1.230066012

19	1.226351756
20	1.222926921
21	1.219755197
22	1.21680636
23	1.214055019
24	1.211479669
25	1.209061955
26	1.206786106
27	1.20463848
28	1.202607215
29	1.195425616

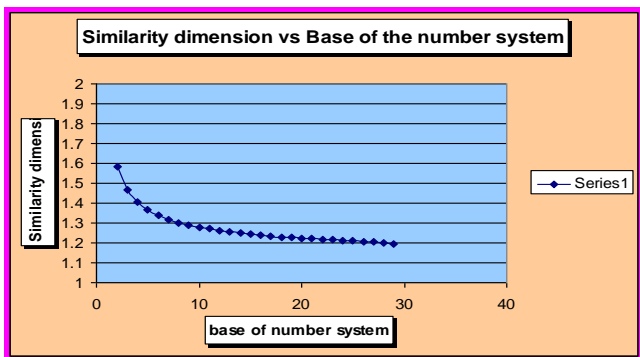


Figure 14. . Shows a graph how fractal diemnsion increases in accordance with base of the number system, using table 2.

Here it is noted that similar analysis for decrement of fractal dimension in switching of EVT fractals from one base to another base of the number system could be drawn.

IX. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

So far we have observed how CVT and $E_{\text{Extrem}}VT$ produce fractals. In this regard one natural question may be raised as to whether some other functions would be able to produce the same fractals. We are happy to produce those functions also. Already we have observed that in extreme value transformation we have considered the bit wise maximum, instead of bit-wise maximum if we consider bit wise minimum we will be having those fractals but different orientation. Then we are in firm conviction that corresponding to CVT there is also another twin like function, which also produces those fractals although we have not yet analyzed them. Our current effort will be to analyze those twin functions algebraically and thus finding the broader spectrum of these fractals.

REFERENCES

- [1] B.B. Mandelbrot, The fractal geometry of nature. New York, 1982.
- [2] P.P. Choudhury, S. Sahoo, M. Chakraborty, Implementation of Basic Arithmetic Operations Using Cellular Automaton, ICIT-08, IEEE CS Press, 11th International Conference on Information Technology, pp 79-80, 2008.
- [3] H. O. Pietgen, H. Jurgens, D. Saupe, Chaos and Fractals New Frontiers of Science, ISBN 3-540-97903-4, Springer Verlag, 1992.
- [4] A.K. Ghosh, P. P. Choudhury, R. Choudhury, Production of fractals by various means and measuring their dimensions with probable explanation, Laser Horizon, Journal of Laser Science and Technology Centre (LASTEC), vol. 6/No. 2, pp.50-59, 2003.
- [5] A. Bogomolny, Cut The Knot! An interactive column using Java applets <http://www.cut-the-knot.org/ctk/Sierpinski.shtml>
- [6] P. P. Choudhury, S. Sahoo, B. K Nayak, and Sk. S. Hassan, Carry Value Transformation: It's Application in Fractal Formation 2009 IEEE Explore 2.0, pp 971-976, 2009.