Robustness of Diffractal Architecture in Time Domain: A theoretical approach

PhD 1st Year Research Rotation Report

Submitted by

Sahana Das Bhattacharyya

(Banner ID 000139658)

Dr. Luat Vuong

Research Advisor

Robustness of Diffractal Architecture in Time Domain: A theoretical approach

Sahana Bhattacharyya (Banner ID 000138659)
3rd Rotation Report
Rotation Mentor: Professor Luat Vuong, Queens College

Introduction and Literature Review

Fractals are infinitely complex and self-repetitive patterns made by repetitive iterations. They are self-similar across different length scales over a whole space/plane [1, 2]. A surprising fact about fractals is their dimension can be non-integer and do not need to follow Euclidian geometry.

Diffractals, as defined by Berry, are the far-field profiles generated by Fraunhofer diffraction of plane wave through fractal-patterned apertures. Generated by fractals, diffractals also exhibit iterated, self-similar features each containing complete information about the fractal aperture. Verma et. Al. observed that any arbitrary section of the diffractals can reconstruct the whole fractal pattern without loss of meaning or information in a Cantor grating. [3] Since the complete information of the fractal aperture is present in the diffractals, by taking the Inverse Fourier Transform (IFT) (i.e, by taking the inverse propagation in the Fourier plane) one can reconstruct the whole pattern from even a very tiny piece of the fractal.

There are many examples of fractals in both natural and man-made systems like aggregated dielectric and metal colloids [4], crystals [5], and tissues [6]. More importantly, in nonlinear optics [7, 8] fractals are very competent to improve the response of materials [9–11]. Diffractals (diffraction by fractal patterns) are extremely useful for encrypting data using random-phase encoding and that can be used in communication system. [13]

Moocarme et. al. [12] studied a more complicated structure called Sierpinski Carpet both theoretically and experimentally. According to this study, using the Fast Fourier Transform (FFT) of the diffractals from this pattern, spatial multiplexing can be obtained. They have also studied the robustness and redundancy of the pattern.

Having a target to replicate the same thing in time domain, we theoretically generated an1-dimensional fractal pattern and observed how a signal transmitted through it can change. We also calculated the Bit-Error Ratio (BER) and studied how it changes with the fractal order.

Theoretical Work

Here are the codes for 1D input signal, 1D fractal and their plots.

Code: 1

```
%Code to write the Input Signal
function y=InpSigFrac(n)
%create a signal of size 1 x n
if rem(n+2,1)== 1
m = zeros(n+3,1);%displays a matrix of zeros
m(1:2:end,1) = 1 %extracts odd elements from column 1 and makes them a 1
m(2:2:end,1) = 1 %extracts even elements from column 1 and makes them 1
else
    mod(n+3,1)
m = zeros(n+3,1);%displays a matrix of zero
m(1:2:end) = 1 %extracts all odd elements and makes them a one
end
```

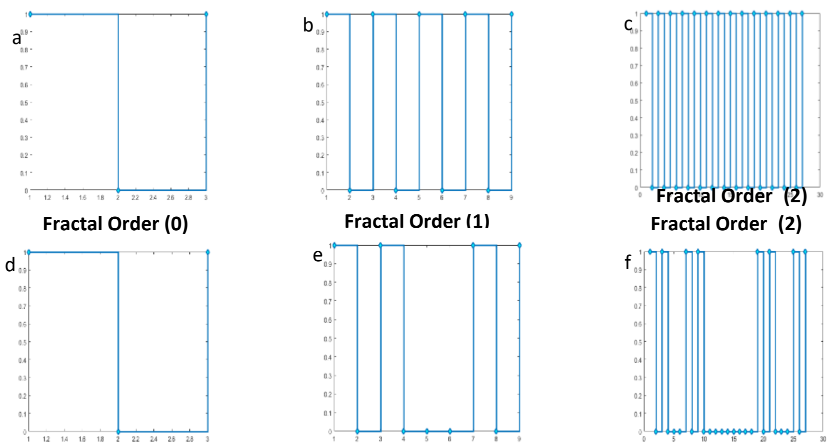
Code: 2

```
%Code to write the fractal vector
function Holo = Frac1D(fracOrder);
    Base = ones(3,1); % Initialize matrix
    Base(2,1) = 0;
    % Zero middle element
    if fracOrder==0;
        % For zero order case
        Holo = Base;
    else
        % For higher order cases
        Holo= kron(Base,Base);
        % For loop iterates the Kronecker tensor product
        for i = 1:fracOrder-1
            Holo = kron(Holo,Base);
         end
```

```
    end
end
```

Code: 3

```
%Code to Plot the Signal
function y = PlotVector(n)
x1 = [Frac1D(n)]
stairs(x1,'LineWidth',2,'Marker','d','MarkerFaceColor','c')
```



MATLAB generated plot of (a), (b),(c) Input Signal vs time and Fractals of order (d) 0, (e) 1 and (f) 2

So what we see here is with the increase in fractal order, we get change in the output signal. The regular input signal (1 0 1 0….) gets diffracted in the fractal pattern and the diffracts are compared in the following table (supporting document is attached in the excel sheet).

Order zero Output	Comment		Order 1 Output	Comment				Order 2 output	Comment
1	No change between Input and output		1		1	No change		1	No change when there are 3^0 number of zeros. Only change happens when we see 3^n number of zeros. This thing happened in all the orders (we calculated till 10^{th} order and got the same result). From this we can calculate the Bit-Error-Ratio of the signal and sample further accordingly.
3^0 zero			3^0 zero	No change	3^0 zero			3^0 zero	
1			1		1	Periodic change		1	
			3^1 zero	Only one change between I/P and O/P in the midpoint of 3^1 zeros	3^1 zero			3^1 zero	
			1		1	No change		1	
			3^0 zero		3^0 zero	Periodic change		3^0 zero	
			1		1			1	
					3^2 zero			3^2 zero	
					1			1	
					3^0 zero			3^0 zero	
					1			1	
					3^1 zero			3^1 zero	
					1			1	
					3^0 zero			3^0 zero	
					1			1	