Chaos Theory & Fractal Geometry

. . . but it is not always so; it may happen that small differences in the initial conditions produce very great ones in the final phenomena. A small error in the former will produce a large one in the later. Prediction becomes impossible.

Poincaré (1903)

Santosh Chapaneri,
Assistant Professor, EXTC, SFIT

Outline

- **Chaos Theory**
  - Definition of Chaos
  - Non-linear Dynamic Systems
  - Logistic Map
  - Bifurcation Diagram
  - Chaos v/s Random
  - Chaos Theory: Analogies & Applications

- **Fractal Geometry**
  - Concept of Dimension
  - Fractal Dimension
  - Examples of Fractal Objects
  - Affine Transformations
  - Iterated Function System
  - Application: Fractal Image Compression

Chaos Theory

- Dictionary Meaning of *Chaos* – "*a state of things in which chance is supreme; especially**:** the confused unorganized state of primordial matter before the creation of distinct forms*" (Webster).
- Chaos Theory represents a big jump from the way we have thought in the past – **a paradigm shift.**
- Traditional notion of chaos – *unorganized, disorderly, random*
- But *Chaos Theory has nothing do with this traditional notion*
- On the contrary, it actually tells you that not all that 'chaos' you see is due to chance, or random
- Oxymoron term coined "***Deterministic Randomness***"

Dynamic System

- A dynamic system is a set of **functions** (rules, equations) that specify how variables change over time.

- Example: $$\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \mathbf{y}_{\text{old}}$$ $$\mathbf{y}_{\text{new}} = \mathbf{x}_{\text{old}}$$

Dynamic System

- Important Distinctions:
  - **variables** (dimensions) vs. **parameters**
  - **discrete** vs. **continuous** variables
  - **stochastic** vs. **deterministic** dynamic systems

  How they differ:
  - **Variables** change in time, **parameters** do not.
  - **Discrete** variables are restricted to integer values, **continuous** variable are not.
  - **Stochastic** systems are one-to-many; **deterministic** systems are one-to-one

Terminology

The current **state** of a dynamic system is specified by the current value of its variables, x, y, z, ...

The process of calculating the new state of a *discrete* system is called **iteration**.

To evaluate how a system behaves, we need the functions, parameter values and **initial conditions** or **starting state.**

Illustration

- Classical Learning theory:

$$q_{n+1} = ß\ q_n$$

specifies how q_n, the probability of making an error on trial n, changed from one trial to the next

The new error probability is diminished by ß

Now we can calculate the dynamics by iterating the function

$q_1 = 1$

$ß = .9$

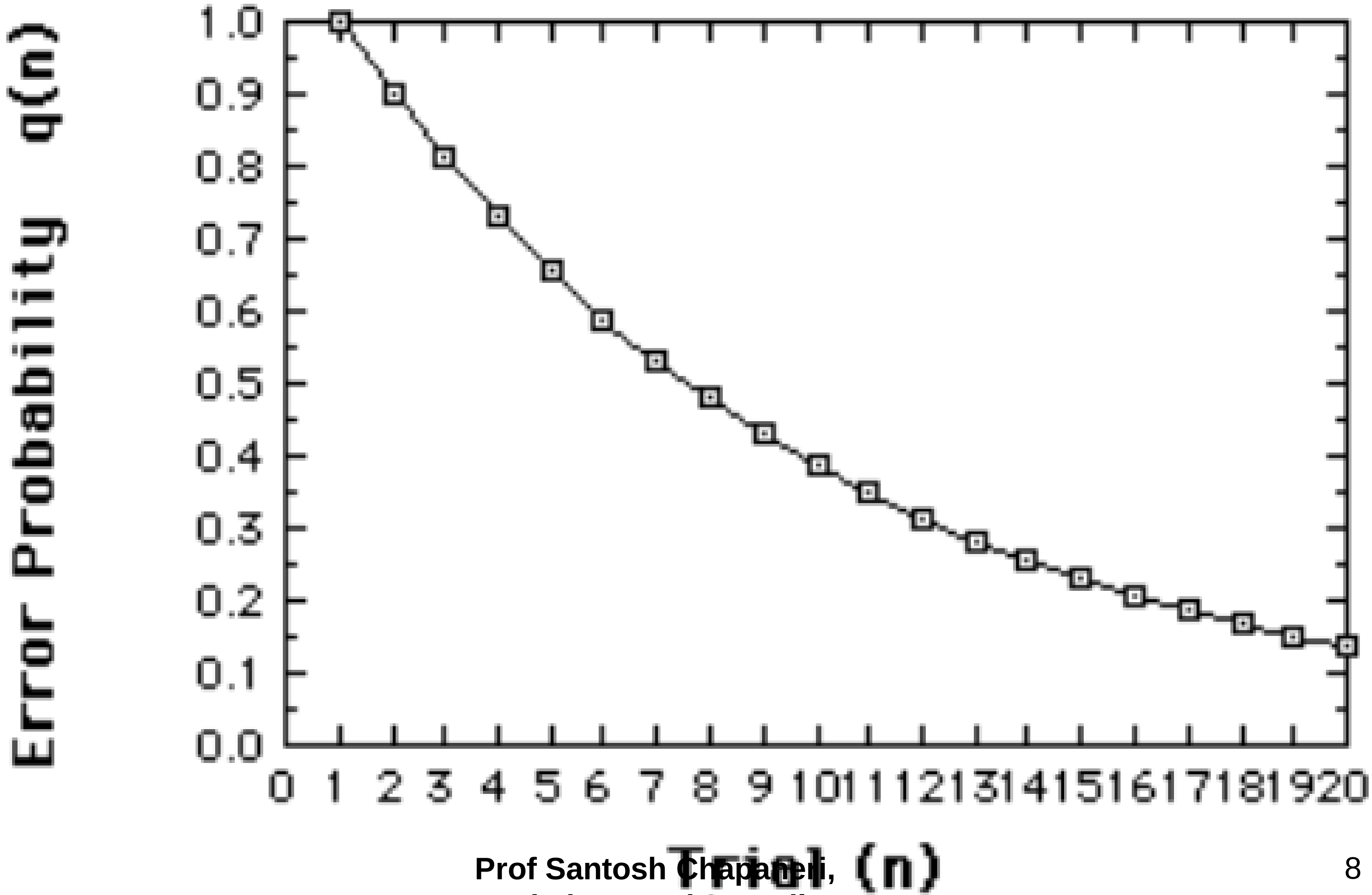
$q_2 = ßq_1 = (.9)(1) = .9$

$q_3 = (.9)q_2 = (.9)(.9) = .81$

etc. ...

Illustration – Dynamic System

- Classical Learning theory:



Non-linear Dynamic System

- Linear function: $y = mx + b$

- Non-linear function:

What makes a dynamic system *nonlinear* ....
is whether the function specifying the change is nonlinear.

And y is a nonlinear function of x if x is multiplied by another (non-constant) variable, or multiplied by itself (i. e., raised to some power).

Example: Non-linear Dynamic System

- **Growth Model**:

$$\mathbf{x}_{\text{new}} = \mathbf{r}\ \mathbf{x}_{\text{old}} \qquad\qquad \mathbf{x}_{n+1} = \mathbf{r}\ x_n$$

This says x changes from one time period, n, to the next, n+1, according to r.

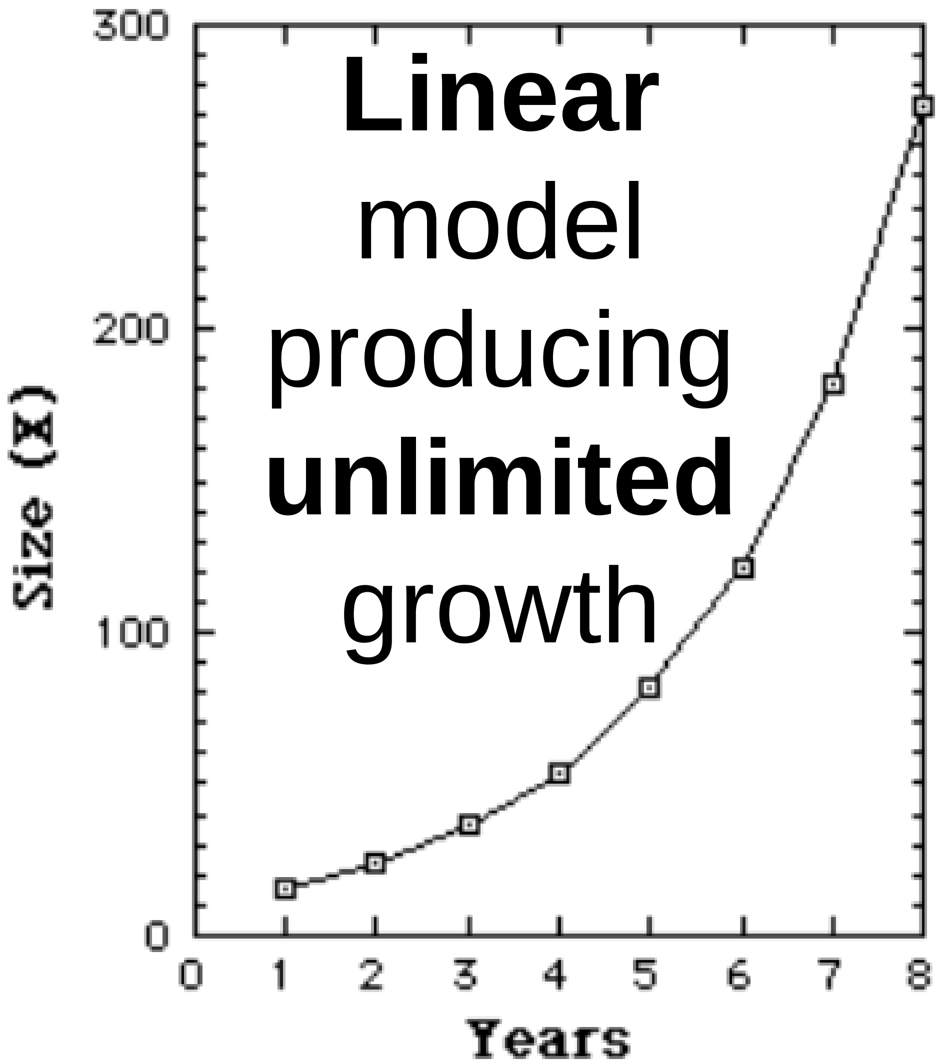
If r is larger than one, x gets larger with successive iterations.

If r is less than one, x diminishes.

Example: Non-linear Dynamic System

We start, year 1 (n=1), with a population of 16 [x_1=16], and since r=1.5, each year x is increased by 50%. So years 2, 3, 4, 5, ... have magnitudes 24, 36, 54, ...

Our population is growing exponentially. By year 25 we have over a quarter million.



Iterations of Growth model with r = 1.5

Limited Growth Model – Logistic Map

The Logistic Map prevents unlimited growth by inhibiting growth whenever it achieves a high level. This is achieved with an additional term, $[1 - x_n]$.

The growth measure (x) is also rescaled so that the maximum value x can achieve is transformed to 1.

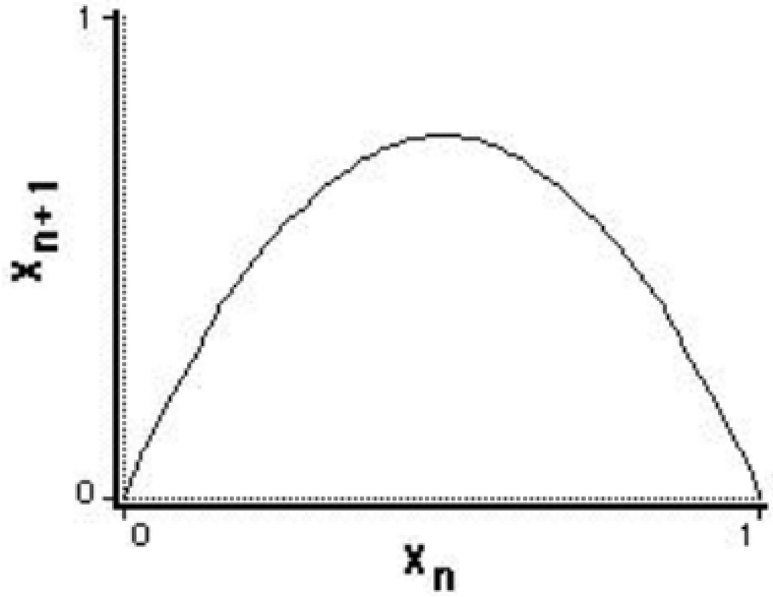
Our new model is

$$\boxed{x_{n+1} = r\, x_n\, [1 - x_n]}$$

[r between 0 and 4.]

The $[1-x_n]$ term serves to inhibit growth because as **x** approaches **1**, $[1-x_n]$ approaches **0**.

Limited Growth Model – Logistic Map



Analysis of Logistic Map

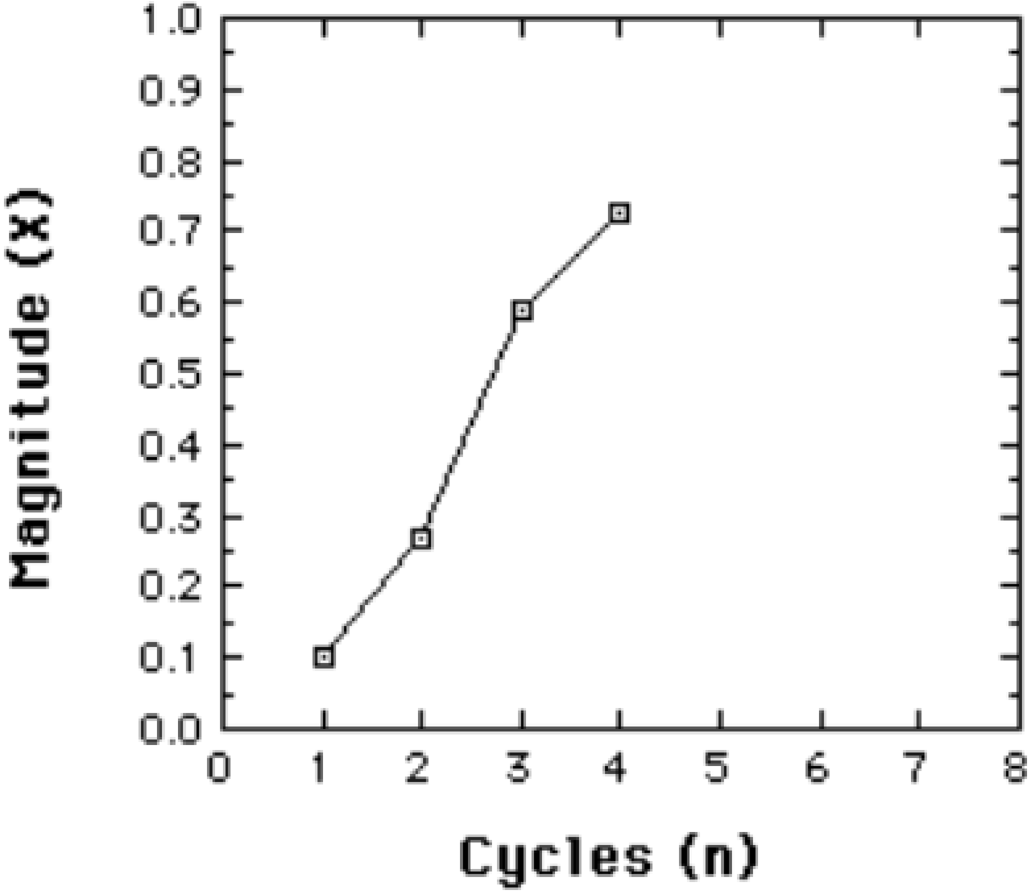
We have to **iterate this function** to see how it will behave ...

Suppose r=3, and x_1=.1

$x_2 = rx_1[1\text{-}x_1] =$
$3(.1)(.9) = .27$

$x_3 = r\,x_2[1\text{-}x_2] =$
$3(.27)(.73) =$
$.591$

$x_4 = r\,x_3[1\text{-}_3] =$
$3(.591)(.409) =$
$.725$



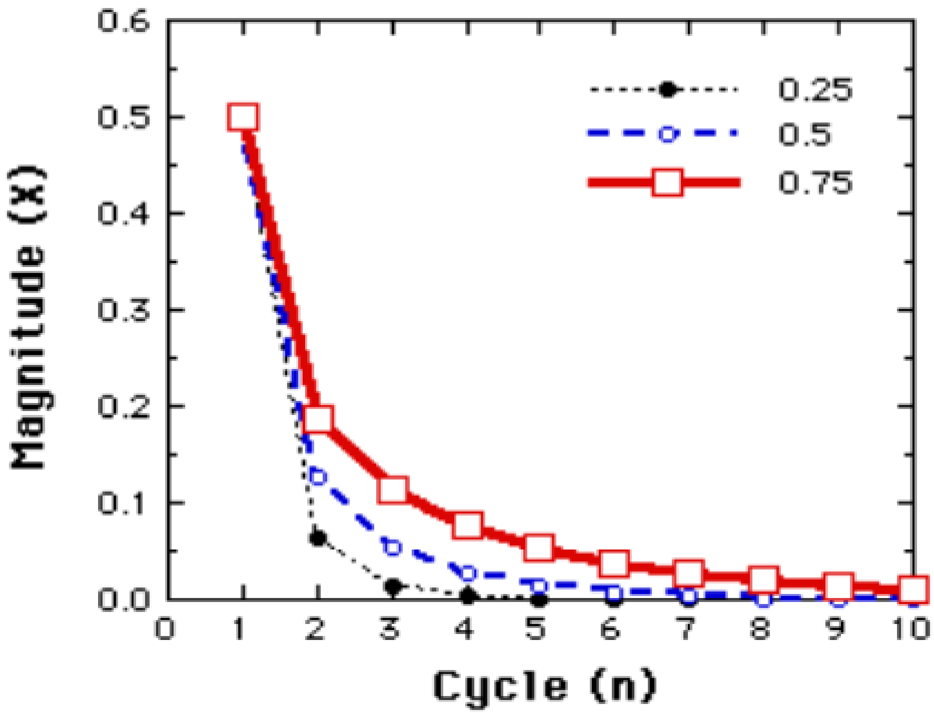
Prof Santosh Chapaneri,
santoshchapaneri@gmail.com

Analysis of Logistic Map

we next examine the time series produced at different values of r, starting near 0 and ending at r=4.

Along the way we see very different results, revealing and introducing major features of a chaotic system.

When r is less than 1



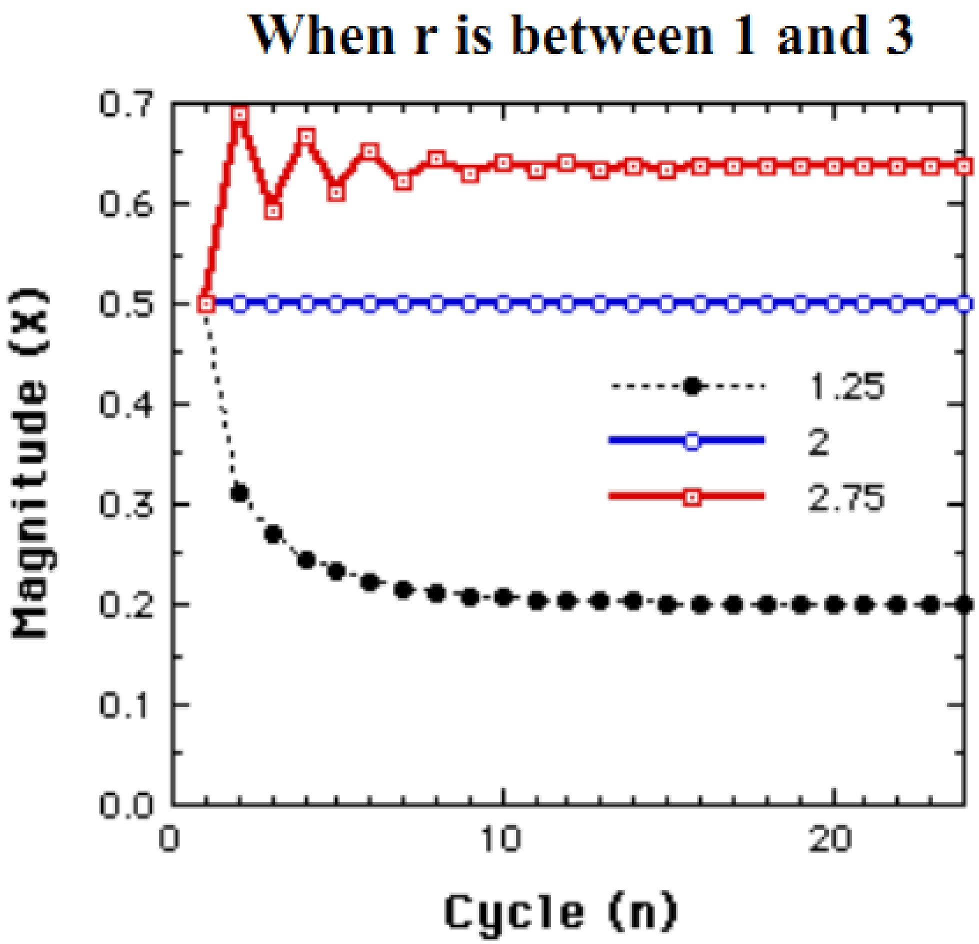
Behavior of the Logistic map for r=.25, .50, and .75.

In all cases x_0 = .5

As long as r < 1, x approaches 0.

One-point attractor

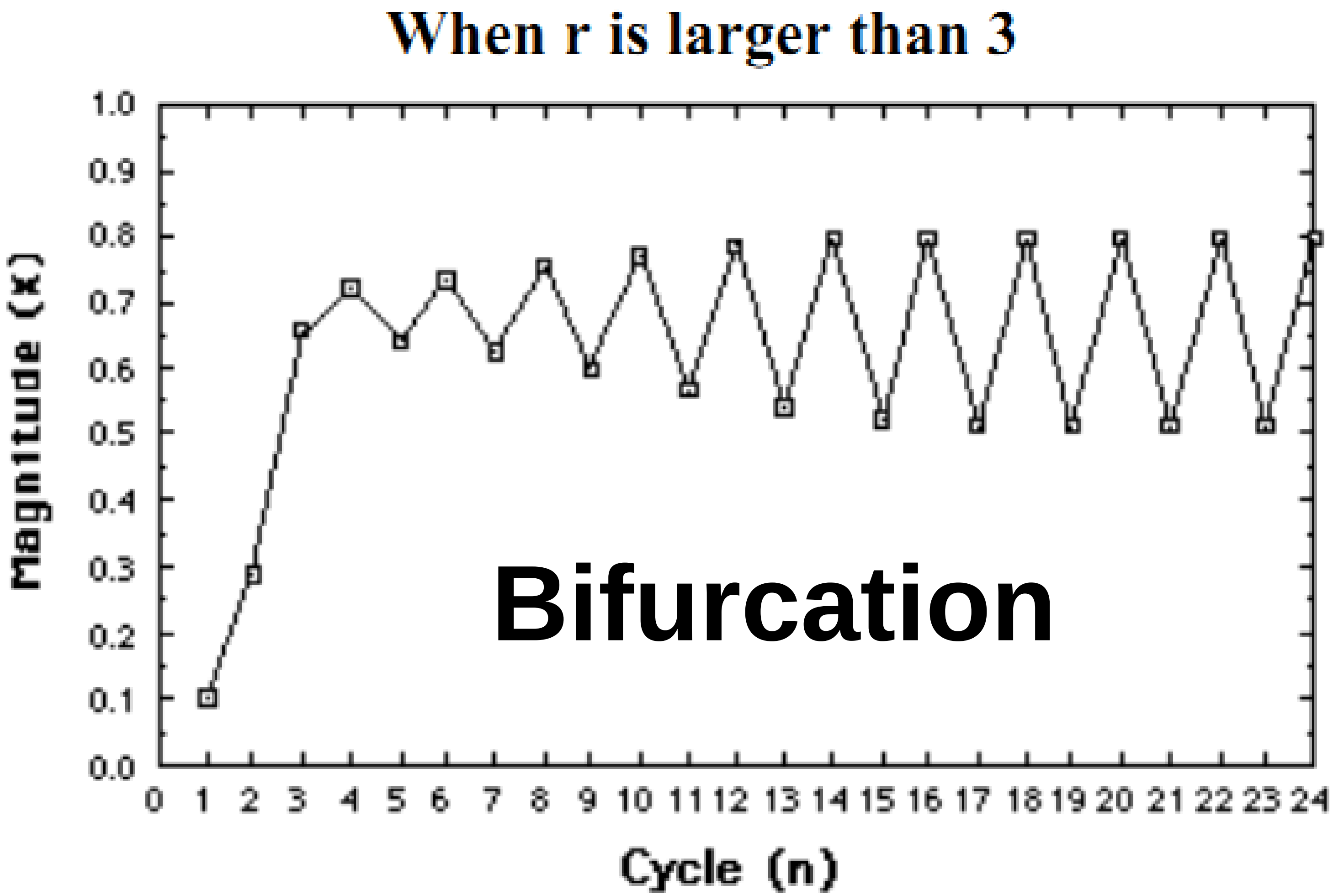
Analysis of Logistic Map



Behavior of the Logistic map for r=1.25, 2.00, and 2.75.
In all cases x_1=.5.

Non-zero One-point attractor

Analysis of Logistic Map

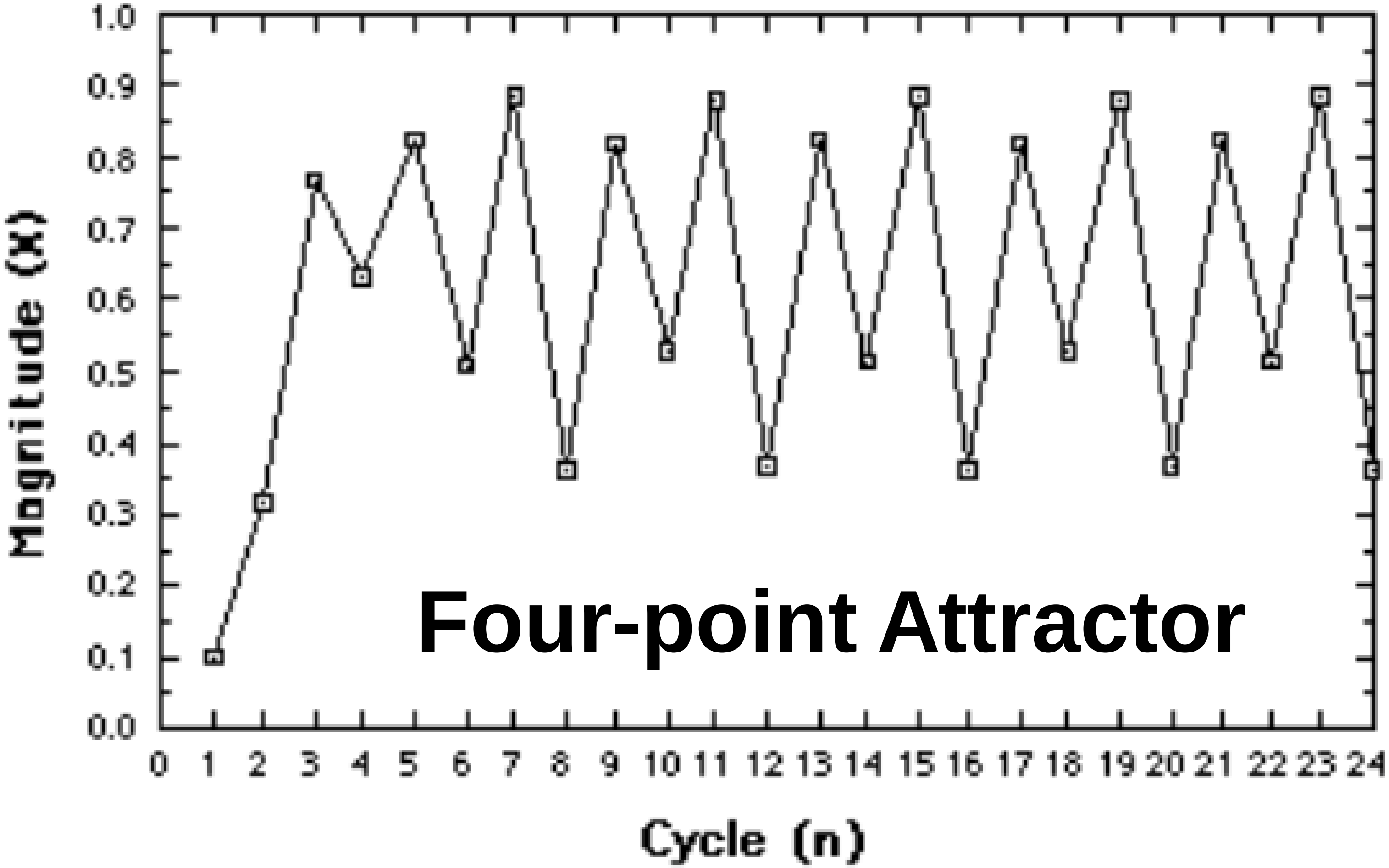


Behavior of the Logistic map for r=3.2

Alternating between two points

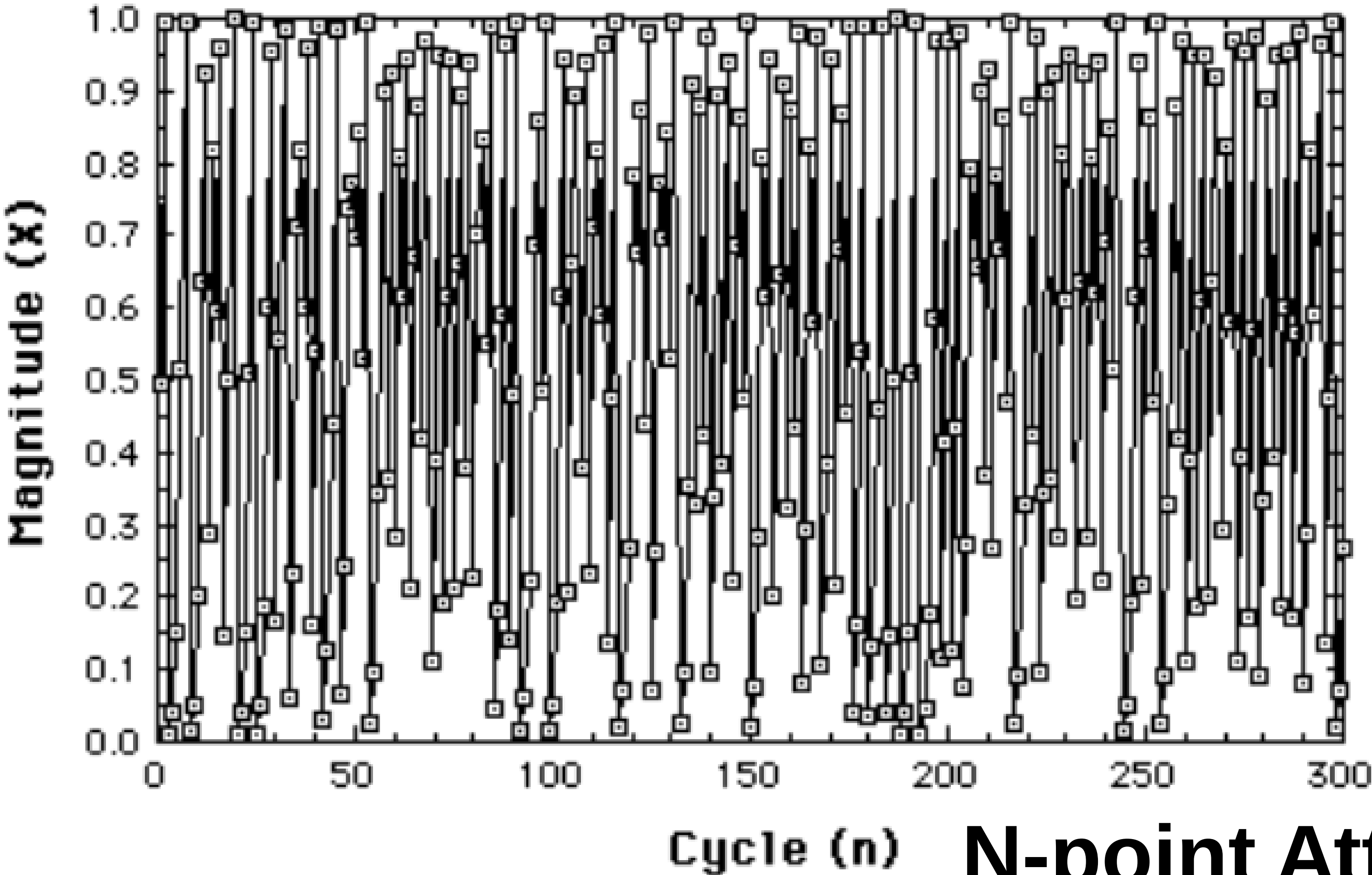
Two-point Attractor

Analysis of Logistic Map



Behavior of the Logistic map for r= 3.54.

Analysis of Logistic Map



N-point Attractor

Chaotic *behavior of the Logistic map at r= 3.99.*

Attractors

So, what is an **attractor**?

Whatever the system "settles down to".

Here is a very important concept from nonlinear dynamics:

A system eventually "settles down".

But what it settles down to, its attractor,

need not have 'stability'; it can be very 'strange'.

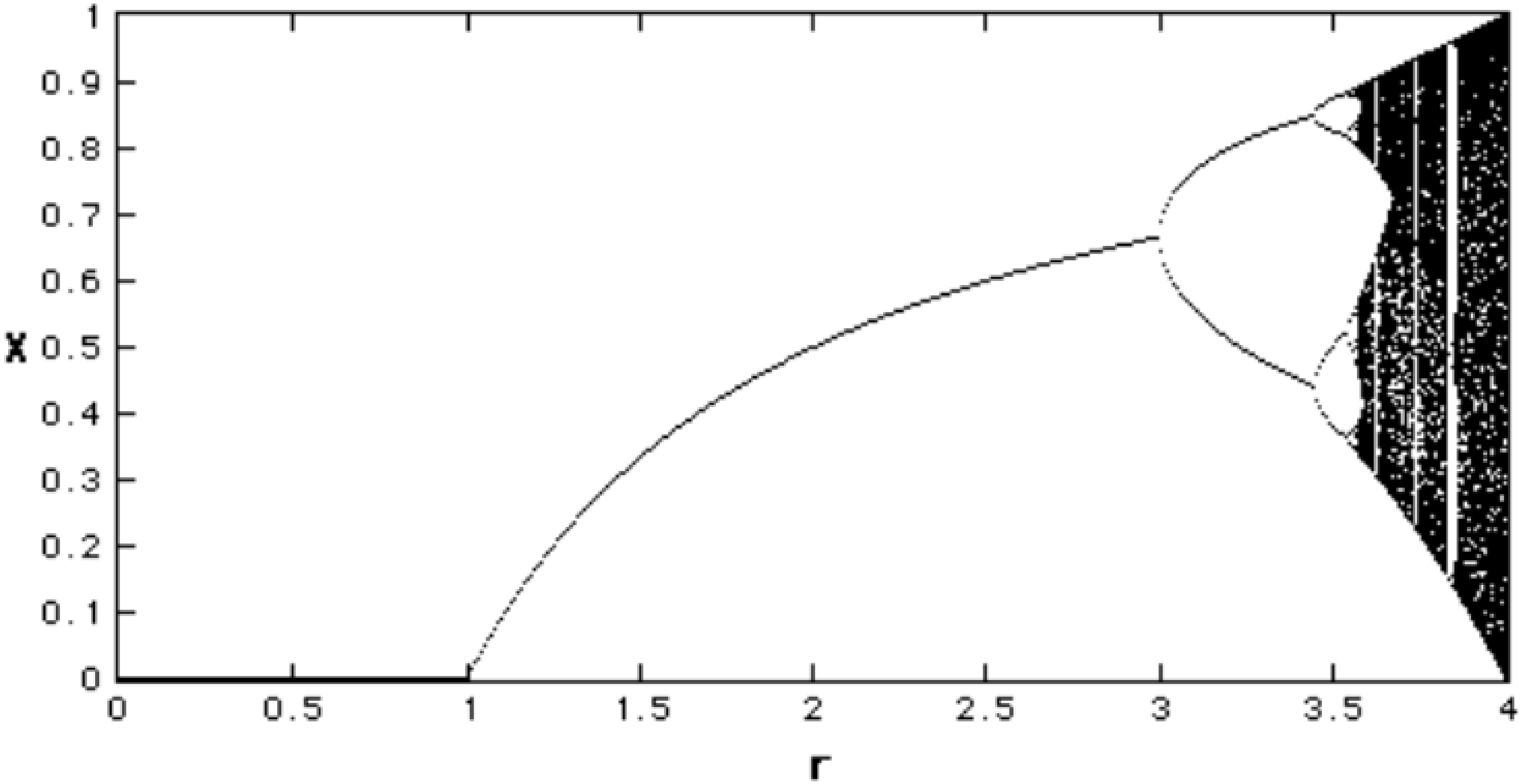
Bifurcation Diagram

A *bifurcation* is a period- doubling, a change from an N-point attractor to a 2N-point attractor, which occurs when the control parameter is changed.

A Bifurcation *Diagram* is a visual summary of the succession of period-doubling produced as r increases.

Bifurcation Diagram – Logistic Map

For each value of r the system is first allowed to settle down and then the successive values of x are plotted for a few hundred iterations.



Bifurcation Diagram r between 0 and 4

Bifurcation Diagram

We see that for r less than one, all the points are plotted at zero. Zero is the one point attractor for r less than one.

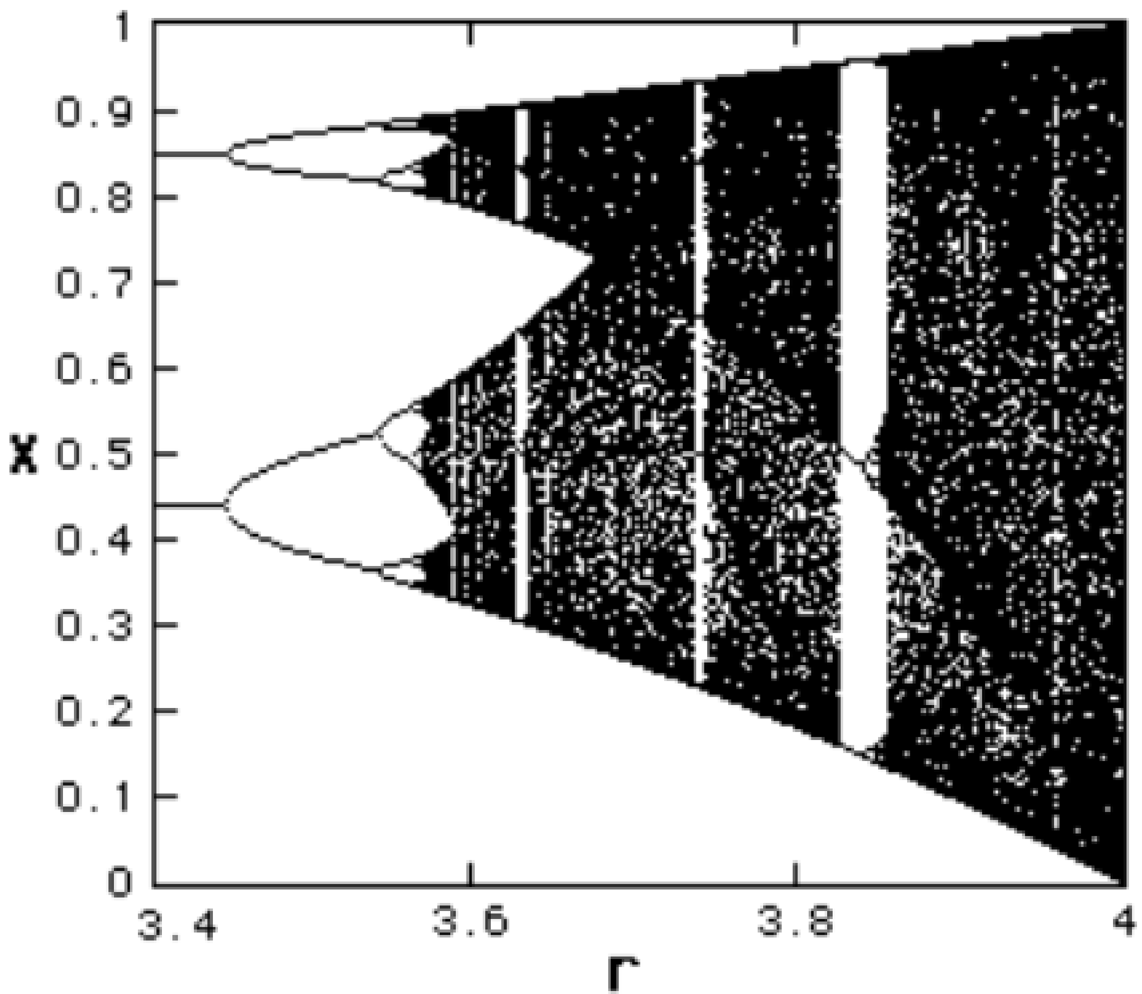
For r between 1 and 3, we still have one-point attractors, but the 'attracted' value of x increases as r increases, at least to r=3.

Bifurcations occur at r=3, r=3.45, 3.54, 3.564, 3.569 (approximately), etc., until just beyond 3.57, where the system is chaotic.

However, the system is not chaotic for all values of r greater than 3.57.

Bifurcation Diagram

Let's zoom in a bit.



Bifurcation Diagram r between 3.4 and 4

Bifurcation Diagram

Notice that at several values of r, greater than 3.57, a small number of x=values are visited. These regions produce the 'white space' in the diagram.

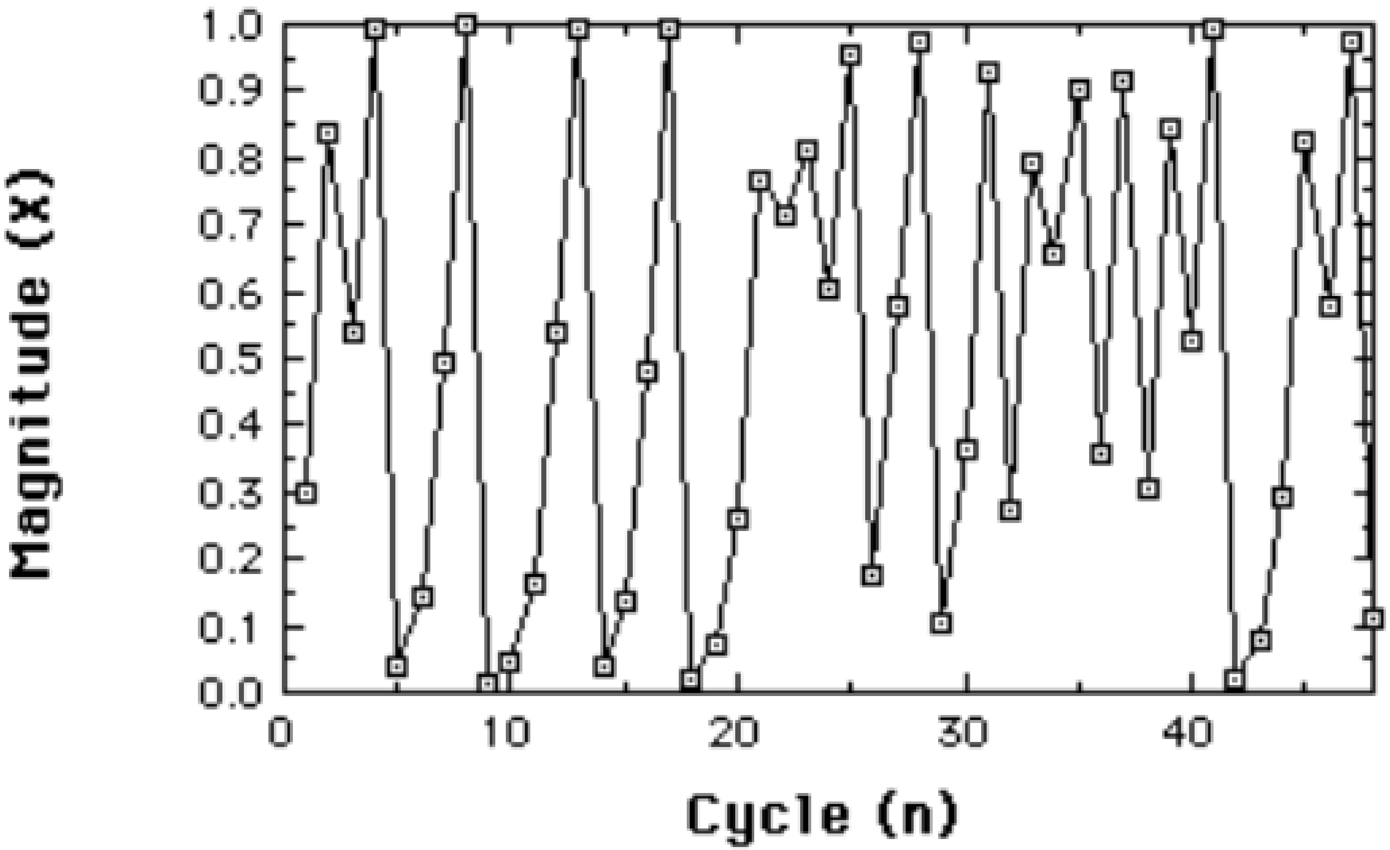
Look closely at r=3.83 and you will see a three-point attractor.

In fact, between 3.57 and 4 there is a rich interleaving of chaos and order.

A small change in r can make a stable system chaotic, and vice versa.

Sensitivity to Initial Conditions

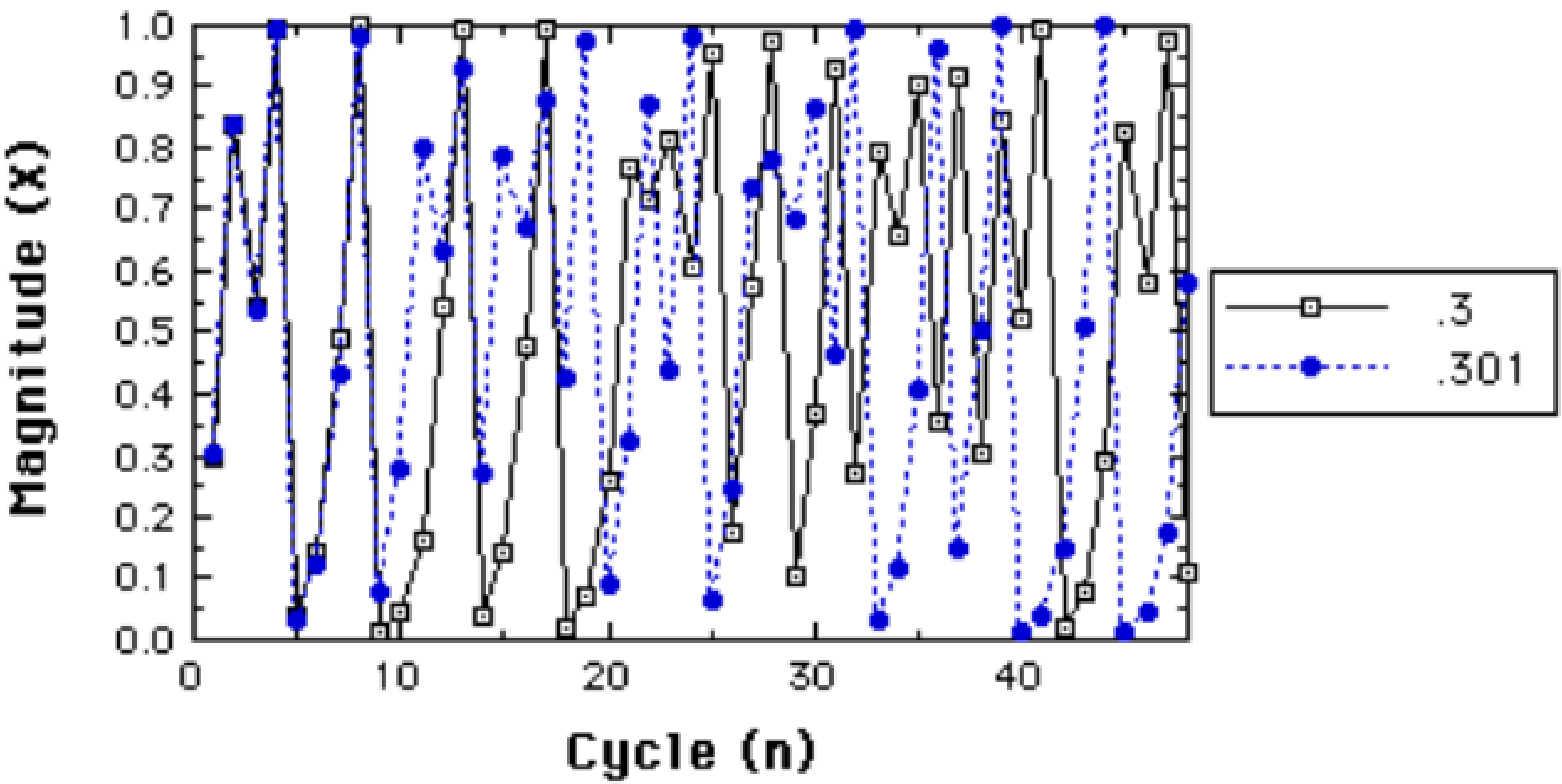
Another important feature emerges in the chaotic region ... To see it, we set r=3.99 and begin at x_1=.3. The next graph shows the time series for 48 iterations of the logistic map.



Time series for Logistic map r=3.99, x_1=.3, 48 iterations.

Sensitivity to Initial Conditions

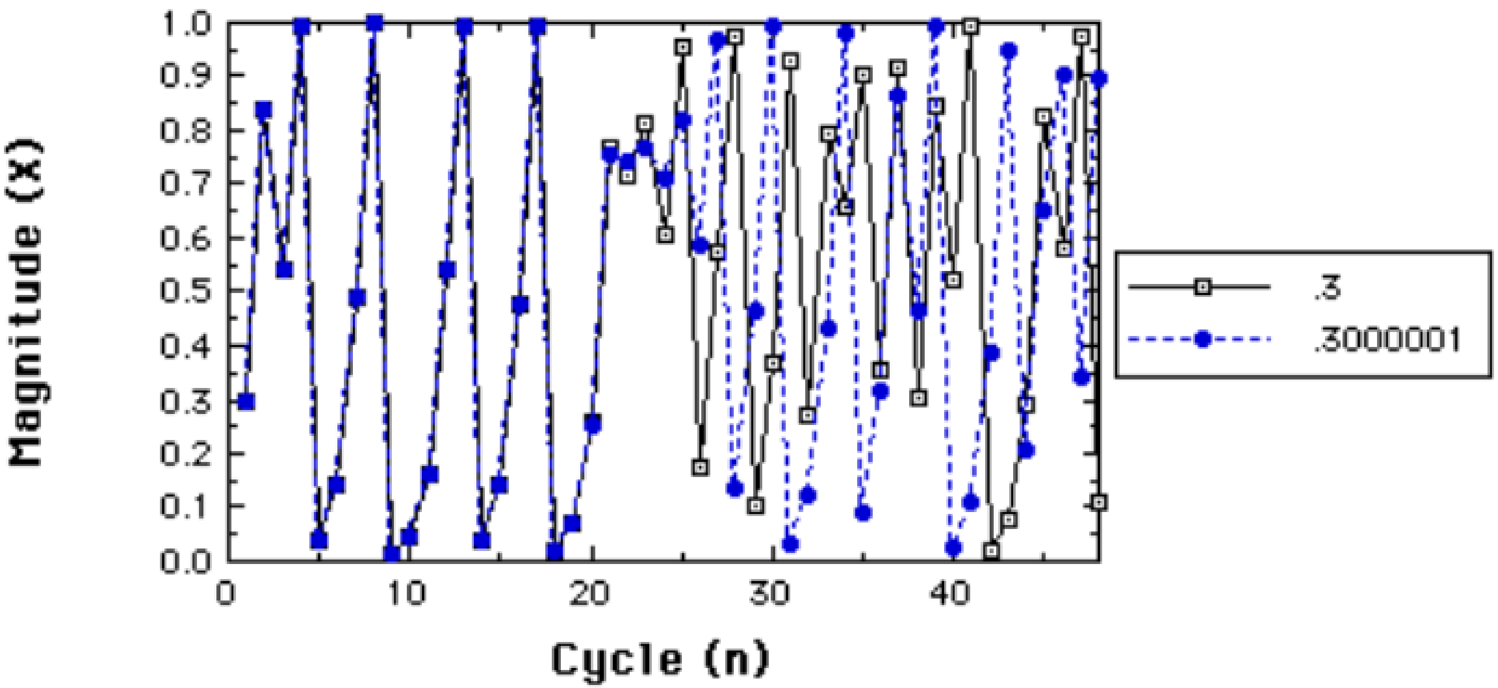
Now, suppose we alter the starting point a bit. The next figure compares the time series for x_1=.3 (in black) with that for x_1=.301 (in blue).



Two time series for $r=3.99$, $x_1=.3$ compared to $x_1=.301$

Sensitivity to Initial Conditions

The two time series stay close together for about 10 iterations. But after that, they are pretty much on their own. Let's try starting closer together. We next compare starting at .3 with starting at .3000001...



Two time series for r=3.99, x_1=.3 compared to x_1=.3000001

This time they stay close for a longer time, but after 24 iterations they diverge.

Chaotic Maps

- We have illustrated one of the **symptoms** of Chaos:

  A chaotic system is one for which the **distance** between two trajectories from nearby points in its state space **diverge** over time.

- The magnitude of divergence increases ***exponentially*** in a chaotic system.

- Implies that a chaotic system, even one determined by a simple rule, is in principle ***unpredictable***!
- In order to predict its behavior into the future, we must know its current value ***precisely***.

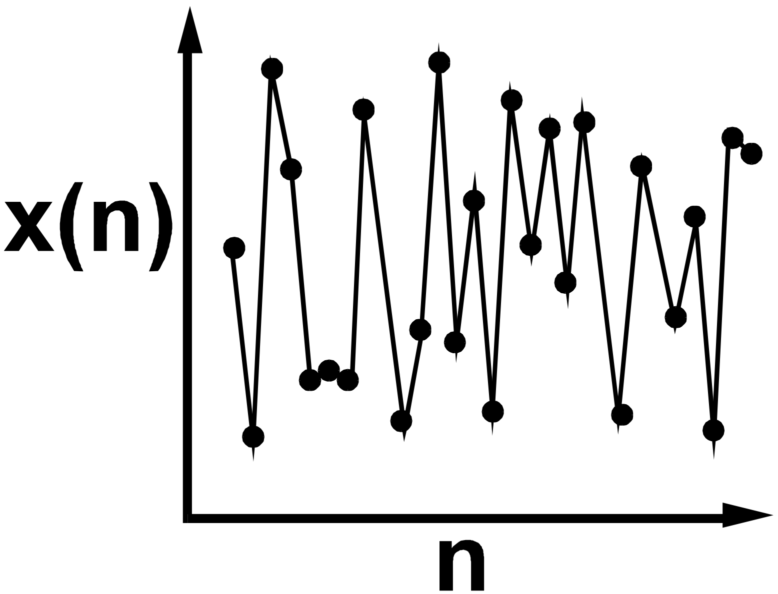
Chaos v/s Random

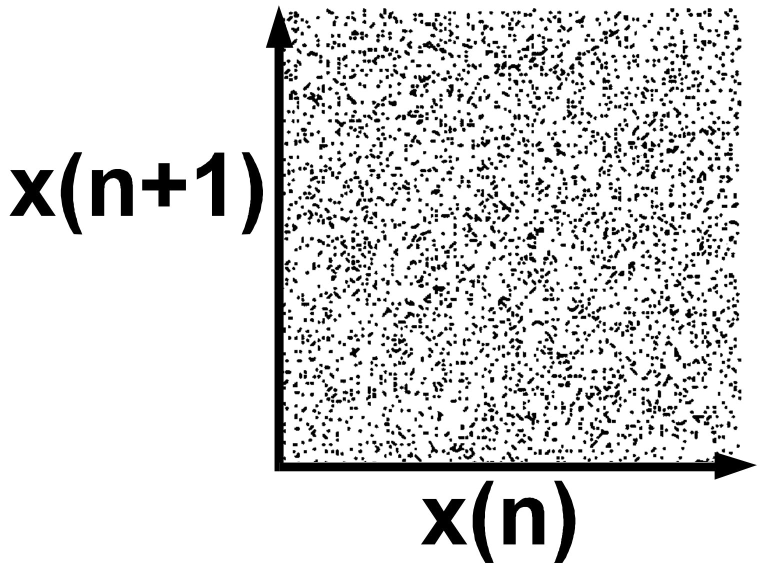
Data 1

RANDOM

random

$x(n) = RND$





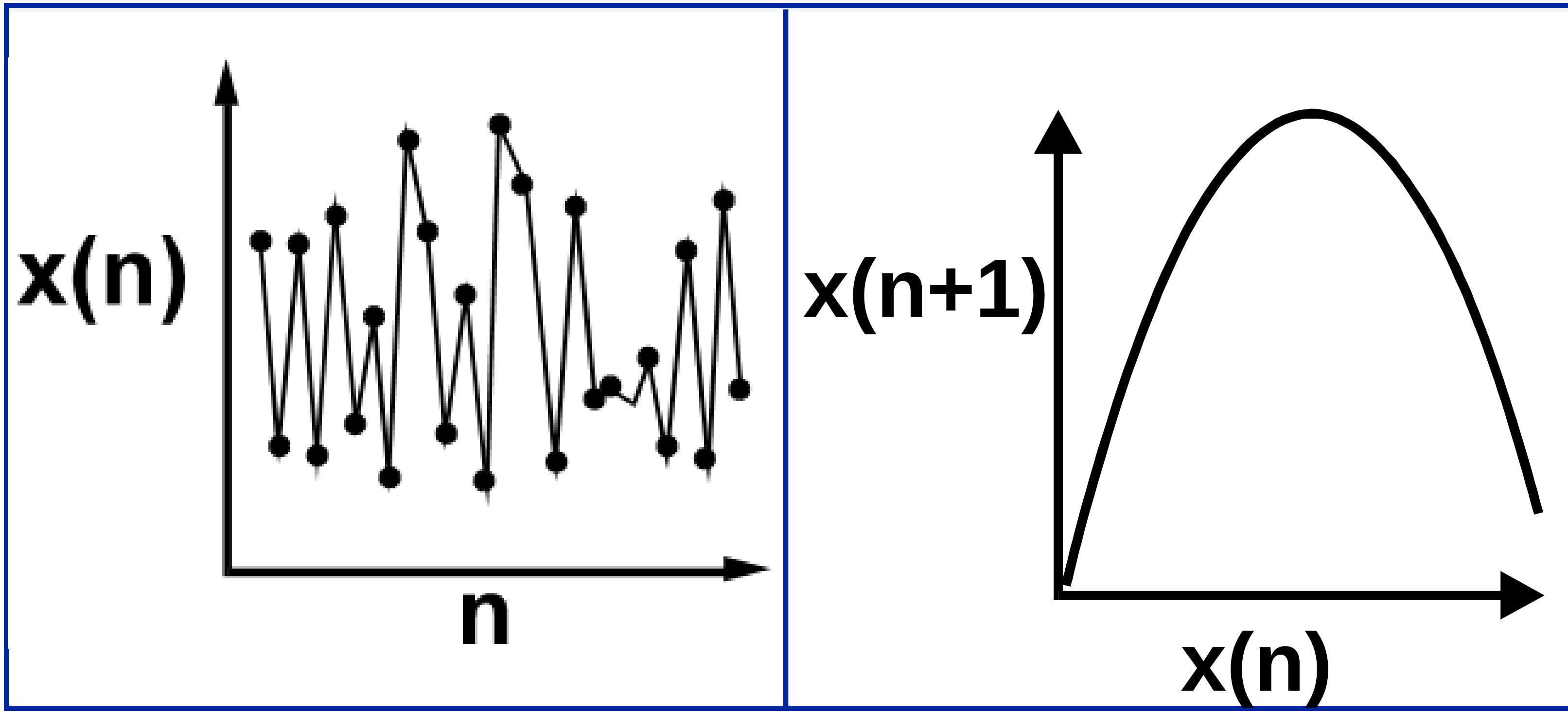
Chaos v/s Random

Data 2

CHAOS

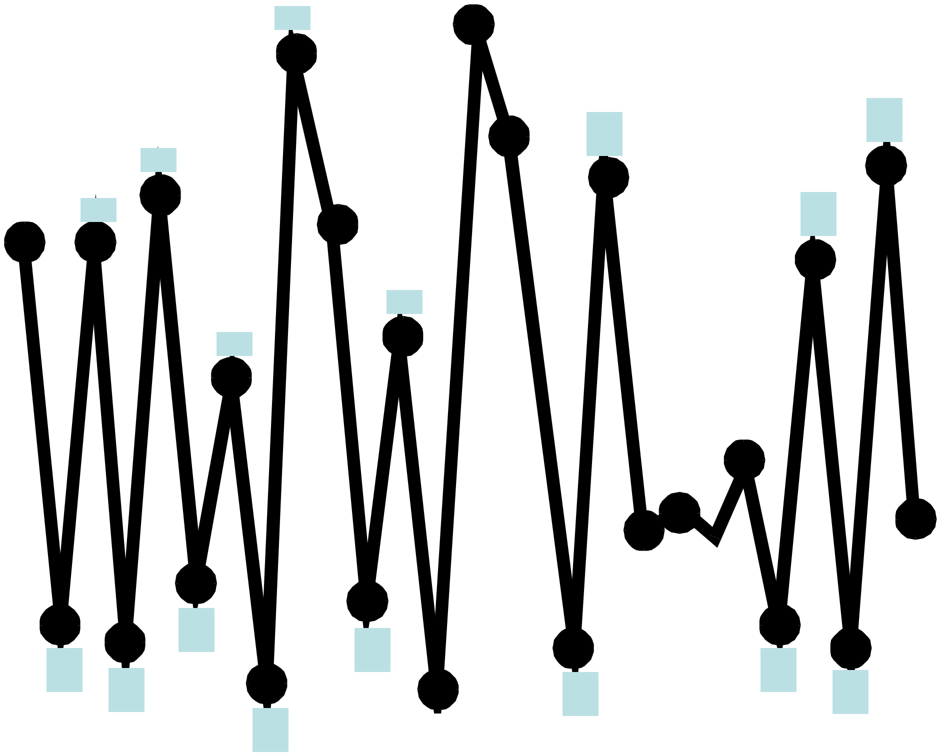
deterministic

$$x(n+1) = 3.95\, x(n)\, [1-x(n)]$$



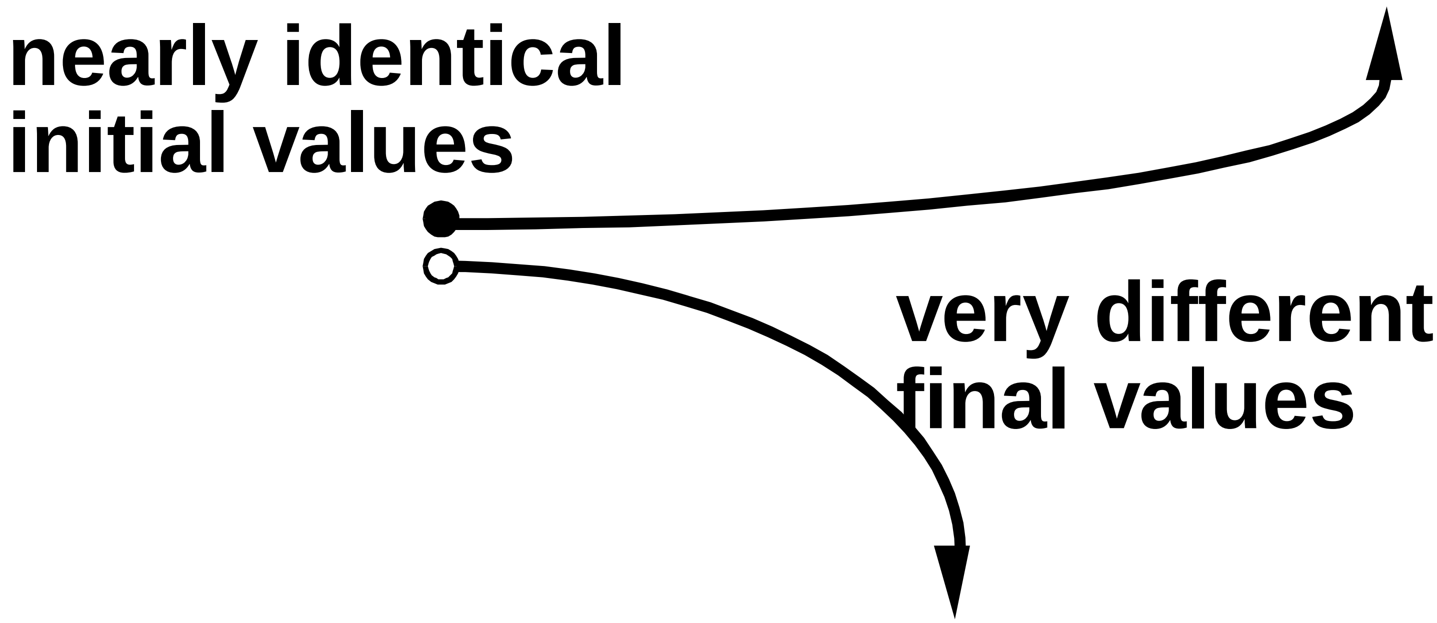
Chaos Theory

Complex Output

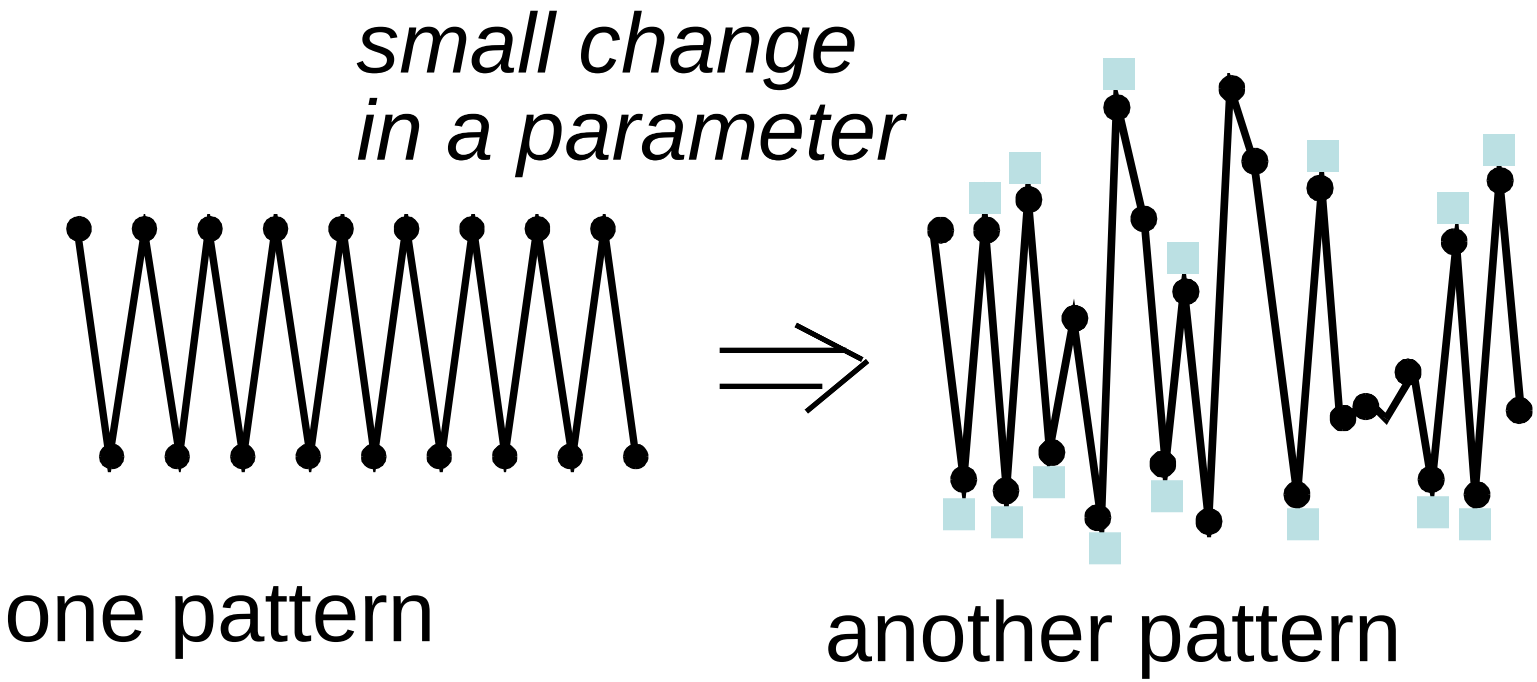


Chaos Theory

Sensitivity to Initial Conditions



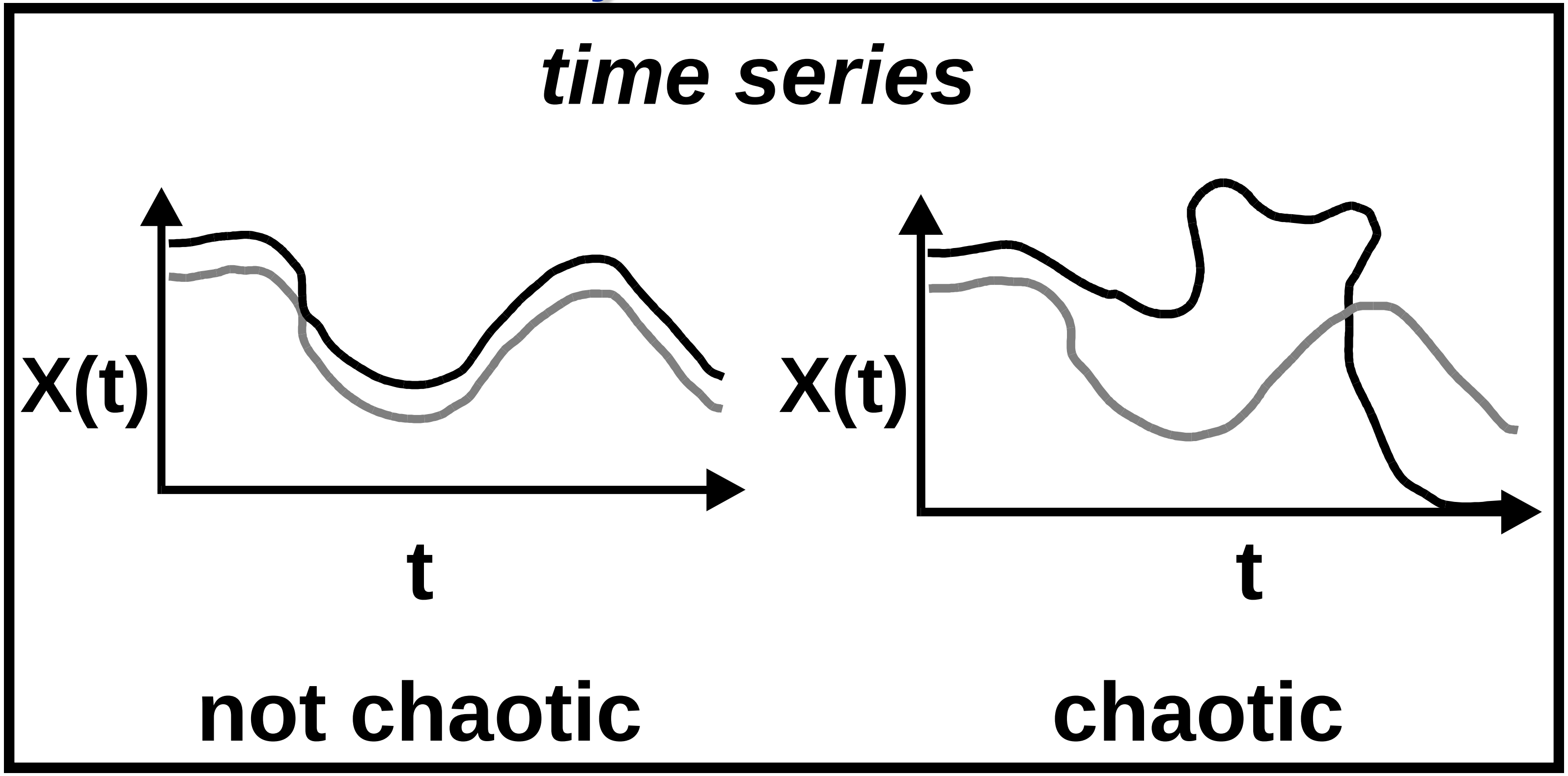
Bifurcations



Chaos Theory

"Chaotic"

sensitivity to initial conditions



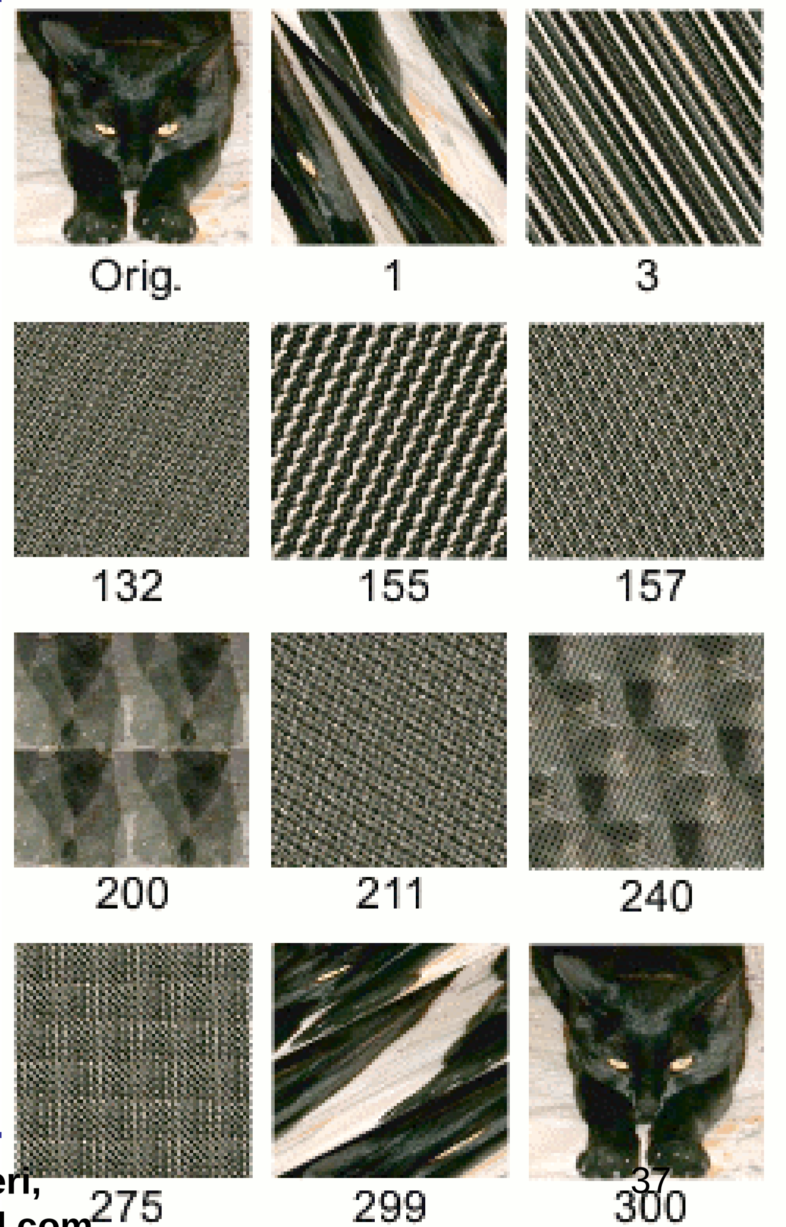
Chaotic Maps

- Other Chaotic Maps:
  - Arnold Map,
  - Henon Map,
  - Baker Map,
  - Standard Map,
  - Piecewise Linear Constant Map, etc.

Arnold Map

$$\Gamma : (x, y) \to (2x + y, x + y) \bmod 1$$

$$\Gamma\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \bmod 1$$



Chaos Theory - Analogies

- Think about the 100m sprint at the Olympics. Sprinters all start the same (supposedly the same initial conditions and they are all the best). Yet, one tiny change (like failing to hear or respond to the whistle on time) can cost them a medal.

- Or life itself – more chaotic. One tiny decision you take today (apparently tiny), you have no idea where it might take you in the long after an accumulation of the triggering effects.

Reference: Ian Stewart, *Does God Play Dice? The Mathematics of Chaos*

Chaos Theory - Applications

- Applied to all scientific disciplines:
  - Mathematics,
  - Geology,
  - Biology,
  - Computer Science,
  - Engineering,
  - Finance,
  - Pattern Recognition,
  - Physics,
  - Politics,
  - Robotics,
  - Electrical Circuits, etc.

Fractal Geometry

Concept of Dimension

So far we have used "dimension" in two senses:

- The three dimensions of Euclidean space (D=1,2,3)
- The number of variables in a dynamic system

Fractals, which are irregular geometric objects, require a third meaning:

The Hausdorff Dimension

If we take an object residing in Euclidean dimension D and reduce its linear size by 1/r in each spatial direction, its measure (length, area, or volume) would increase to $N=r^D$ times the original.

Concept of Dimension

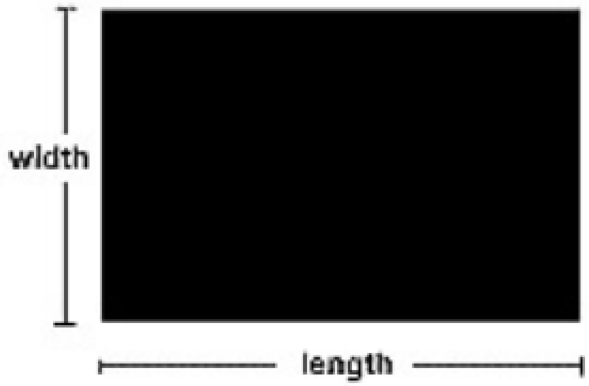
In geometry, a point has no dimension, since it has no length, no width and no depth.

· A point.

A line is one-dimensional because it has length.

A line.

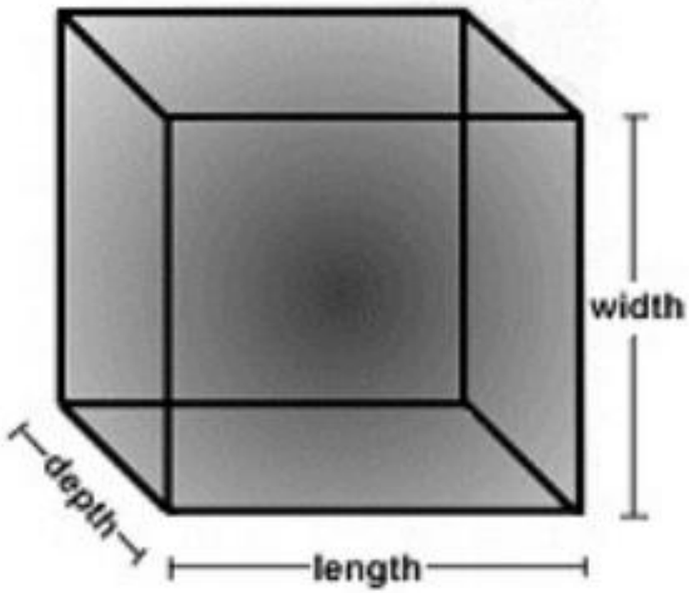
A plane is two-dimensional, since it has length and width.



A plane

Concept of Dimension

A box is three-dimensional: it has length, width and depth.



A cube.

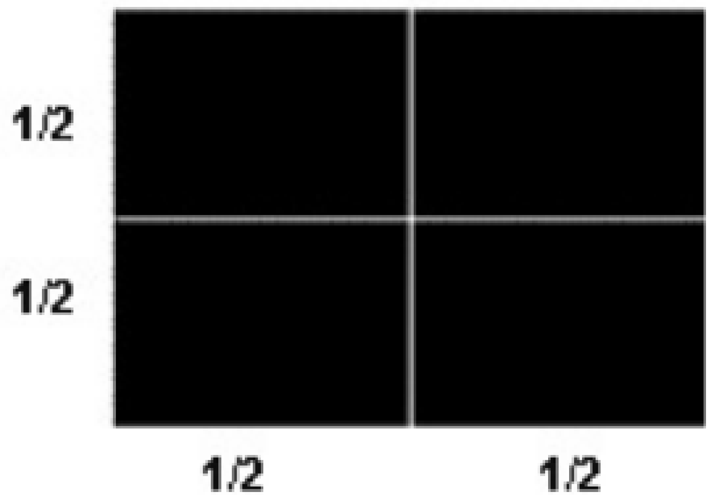
Concept of Dimension

If we divide a one-dimensional object in two smaller equal parts, we get two small versions of the same object.

Division of a line.

Concept of Dimension

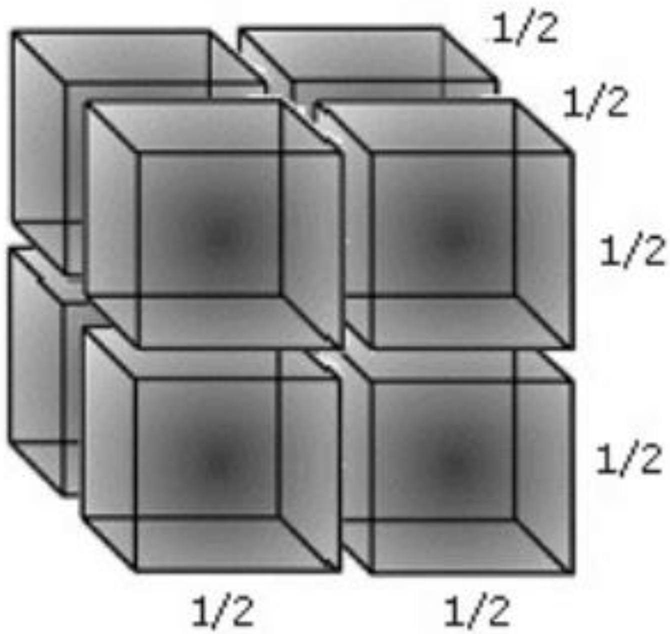
If we divide a two-dimensional object in half its length and width, we get four copies of the same object.



Division of a plane.

Concept of Dimension

If we divide a three-dimensional object in half its length, width and depth, we get eight copies of the same object.



Division of a cube.

Concept of Dimension

$$2 = 2^1$$

$$4 = 2^2$$

$$8 = 2^3$$

Examining the exponent in each case, we find that it is equal to the dimension of each object: 1, 2 and 3.

Concept of Dimension

$$N = r^D$$

We consider $N = r^D$, take the log of both sides, and get

$$\log(N) = D \log(r)$$

$$D = \log(N)/\log(r)$$

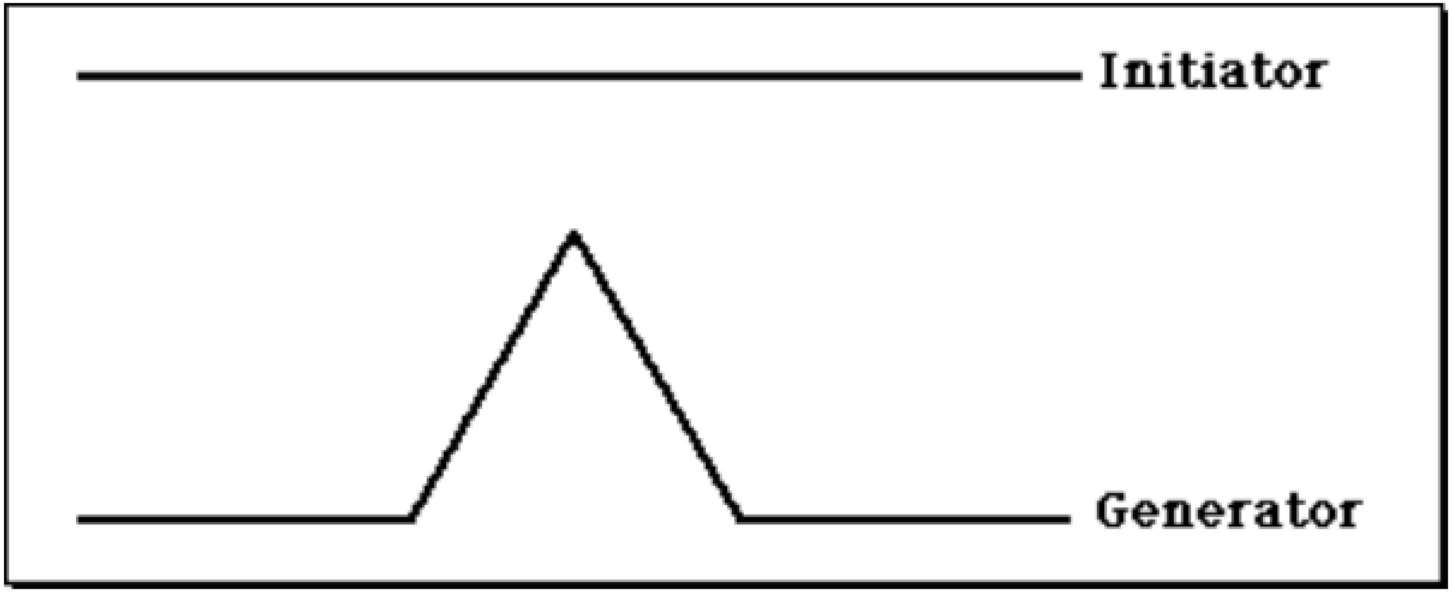
D need not be an integer, as it is in Euclidean geometry.

It could be a fraction, as it is in fractal geometry.

This generalized treatment of dimension is named after the German mathematician, Felix Hausdorff.

Example: Koch Curve

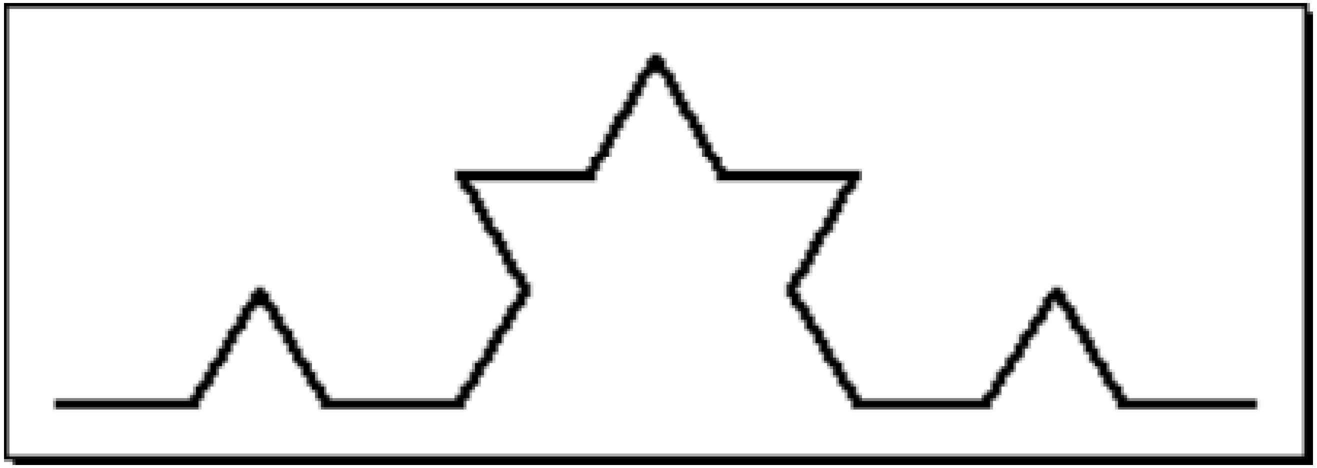
We begin with a straight line of length 1, called the **initiator**. We then remove the middle third of the line, and replace it with two lines that each have the same length (1/3) as the remaining lines on each side. This new form is called the **generator**, because it specifies a rule that is used to generate a new form.



The Initiator and Generator for constructing the Koch Curve.

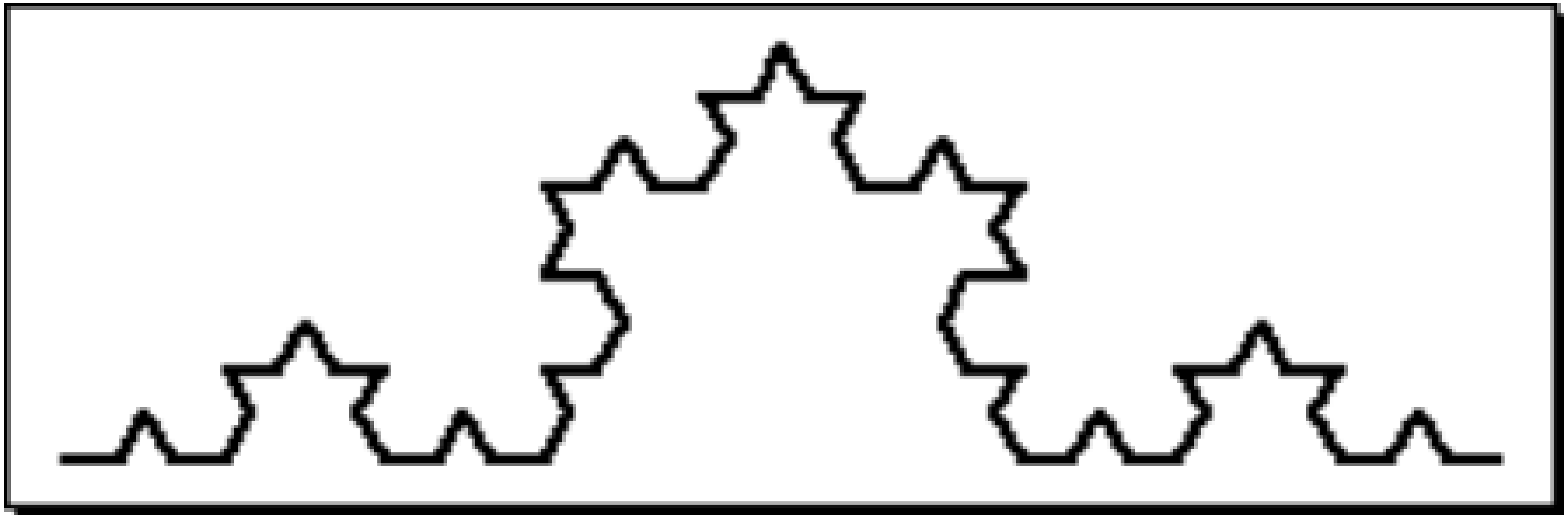
Example: Koch Curve

Level 2 in the construction of the Koch Curve.



The rule says to take each line and replace it with four lines, each one-third the length of the original.

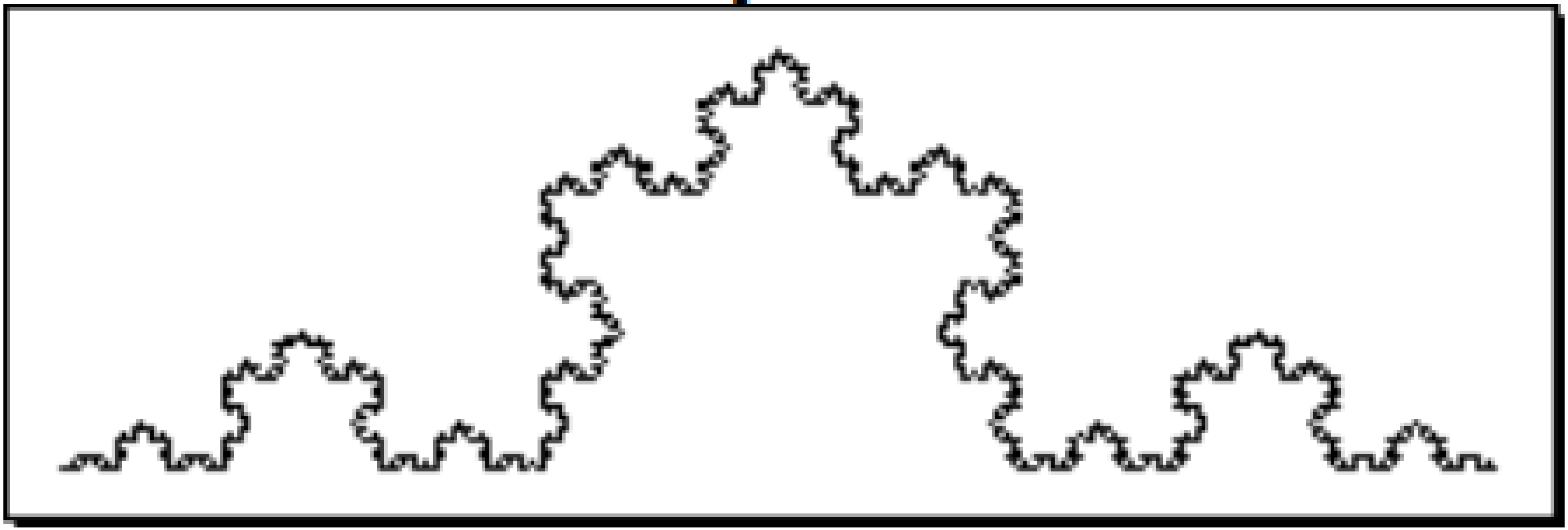
Example: Koch Curve



Level 3 in the construction of the Koch Curve.

Example: Koch Curve

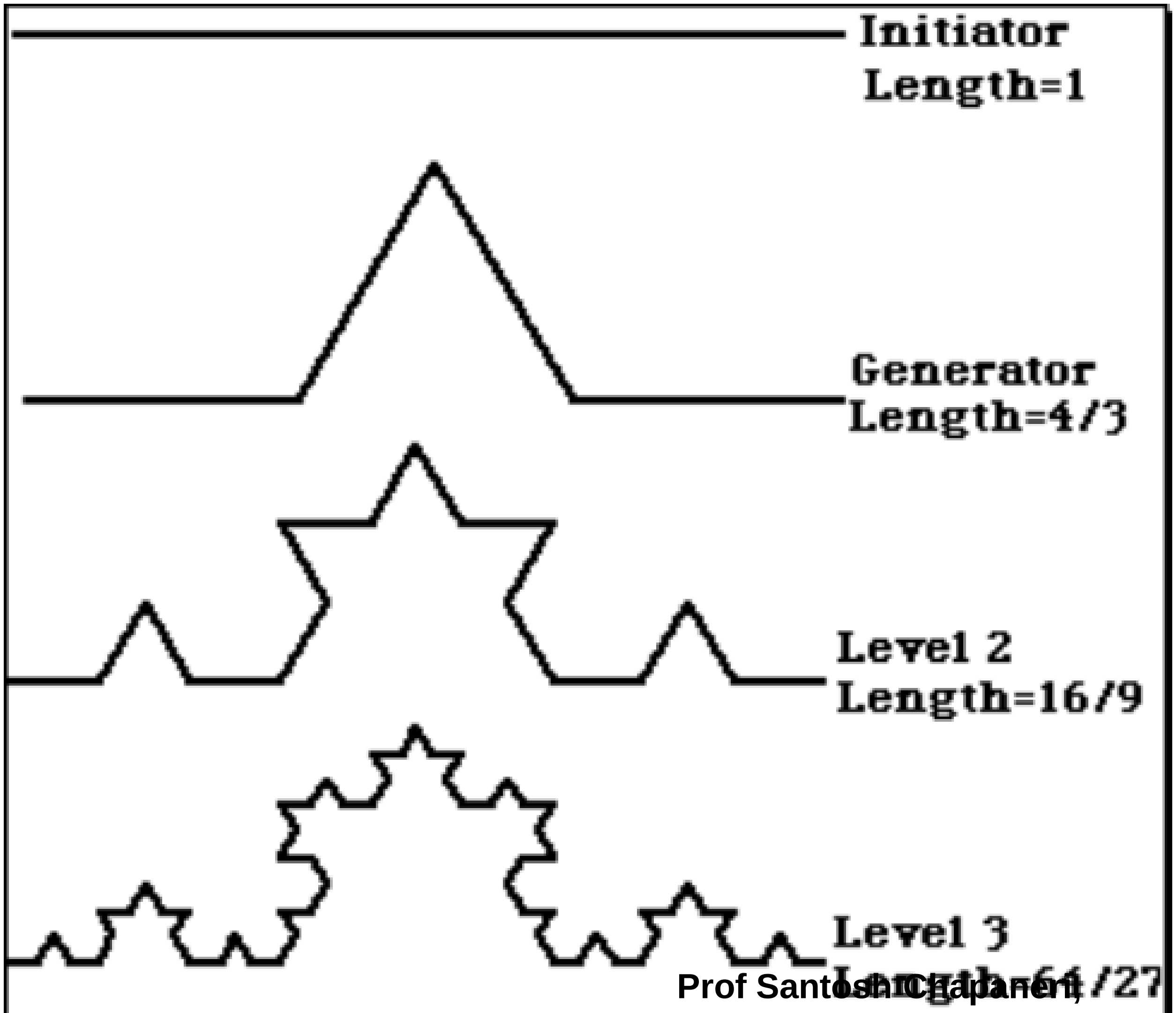
We do this iteratively ... without end.



The Koch Curve.

Example: Koch Curve

What is the **length** of the Koch curve?



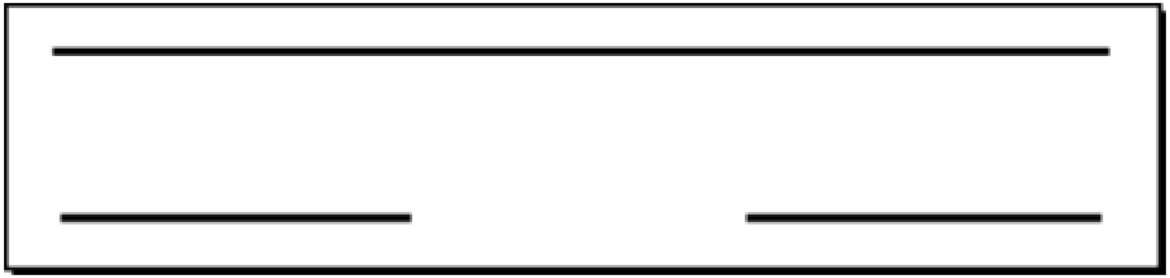
$$D = \log(N)/\log(r)$$

$$D = \log(4)/\log(3)$$

$$D = 1.26.$$

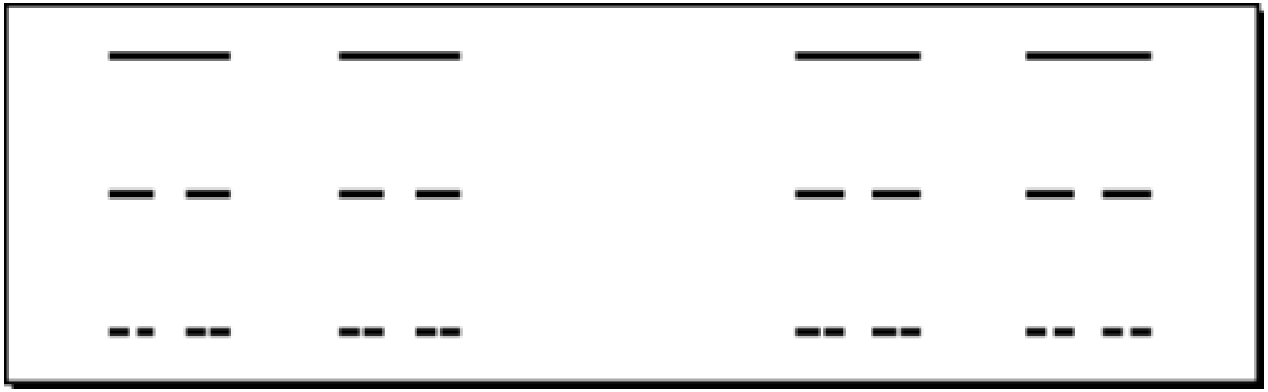
Example: Cantor Dust

Iteratively removing the middle third of an initiating straight line, as in the Koch curve, ...



Initiator and Generator for constructing Cantor Dust.

this time without replacing the gap...



$$D = \log(N)/\log(r)$$

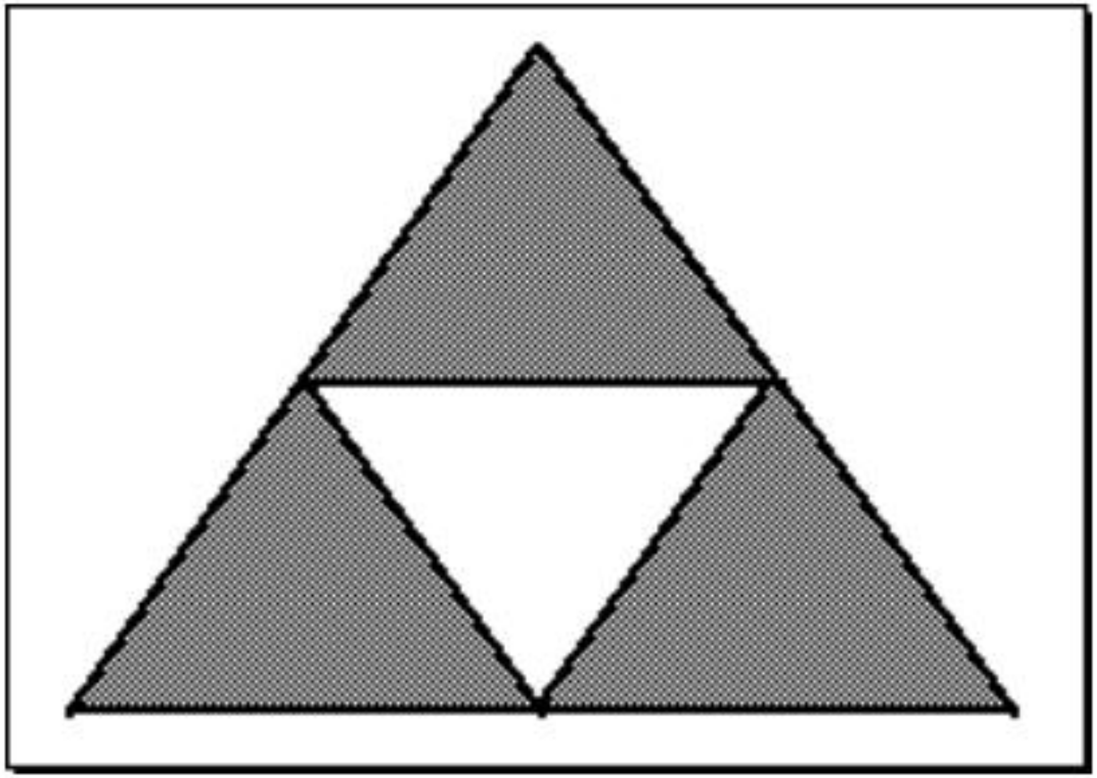
$$D = \log(2)/\log(3)$$

$$D = .63$$

Levels 2, 3, and 4 in the construction of Cantor Dust.

Example: Sierpinski Triangle

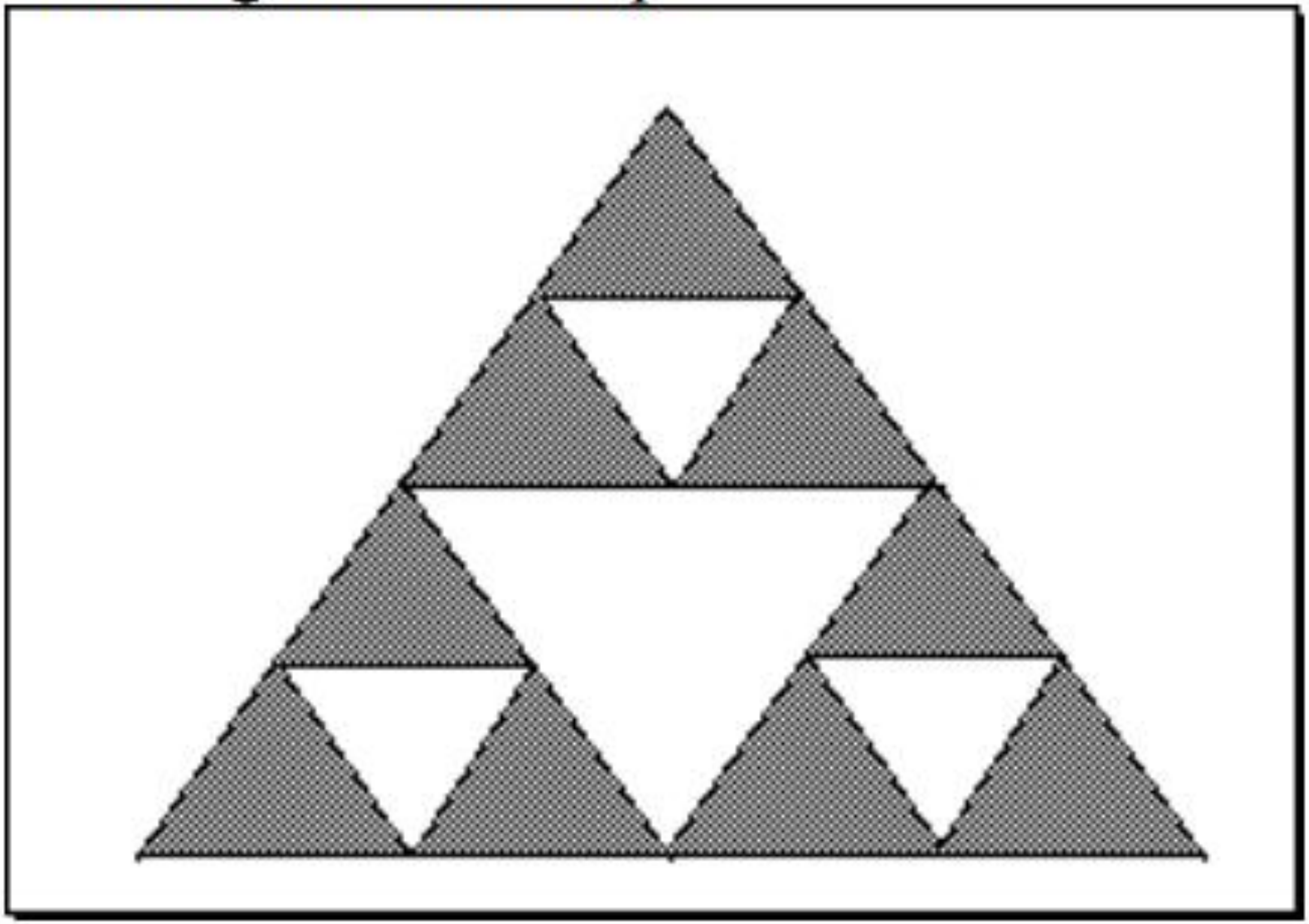
We start with an equilateral triangle, connect the mid-points of the three sides and remove the resulting inner triangle.



Constructing the Sierpinski Triangle.

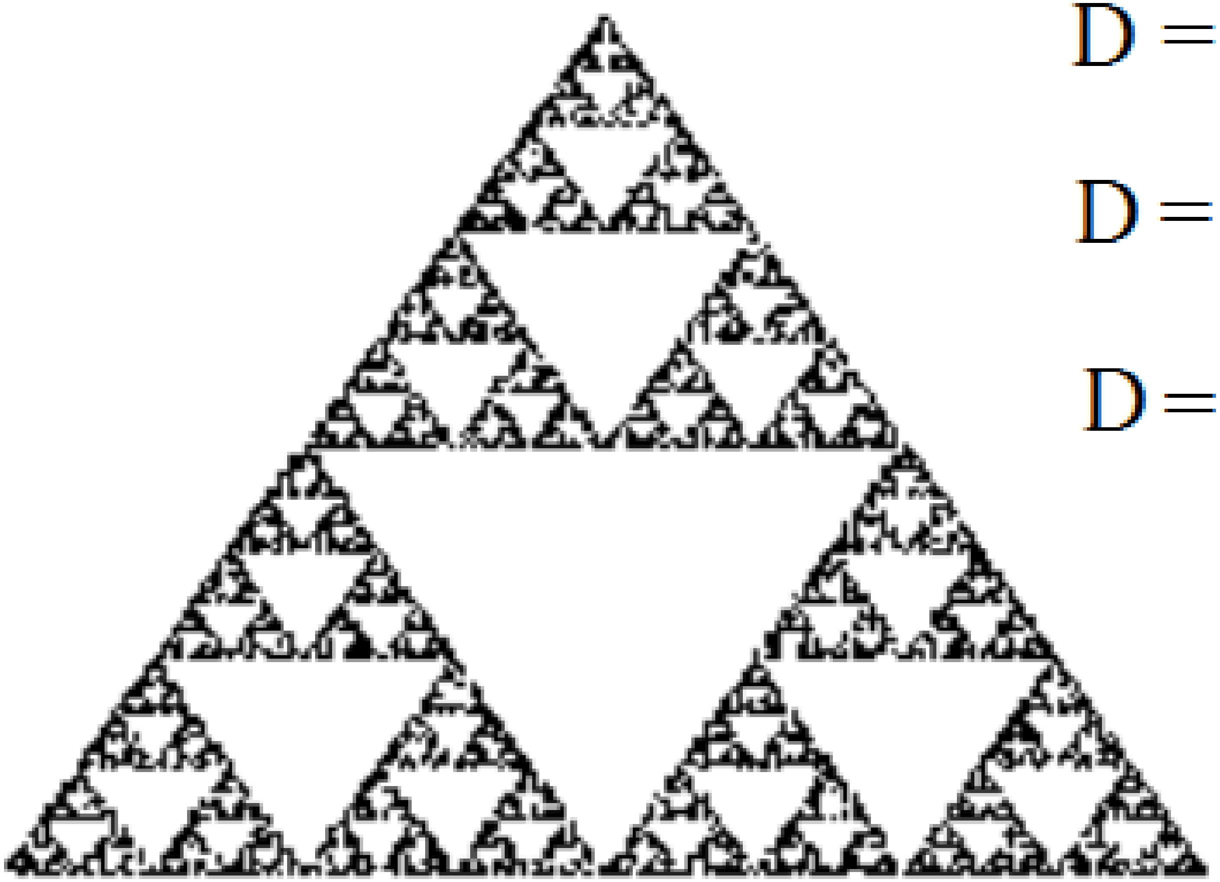
Example: Sierpinski Triangle

Iterating the first step.



Example: Sierpinski Triangle

Constructing the Sierpinski Triangle.



$$D = \log(N)/\log(r)$$

$$D = \log(3)/\log(2)$$

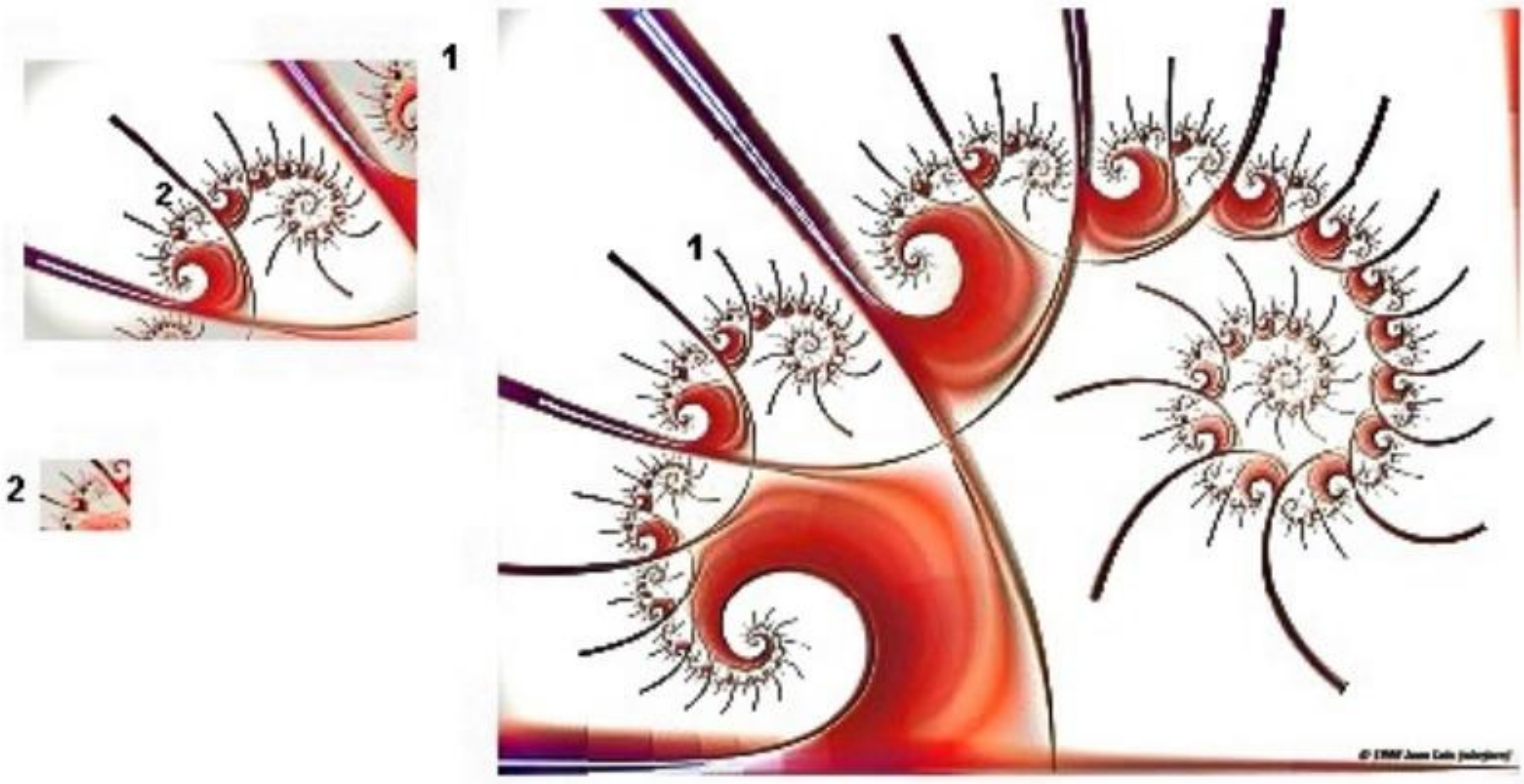
$$D = 1.585$$

Example: Sierpinski Triangle

Self-Similarity

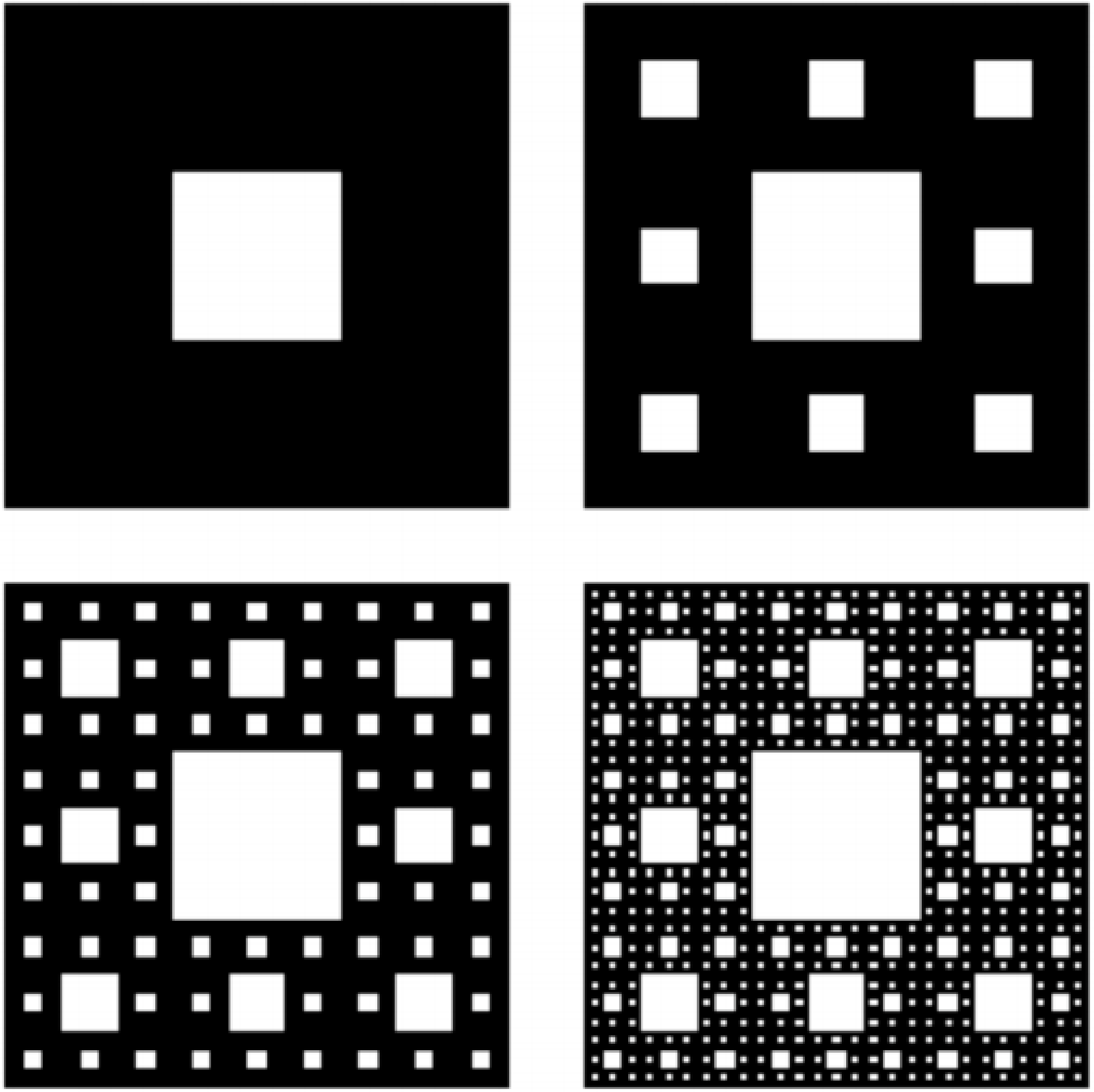
- A **fractal** is a rough or fragmented geometric shape that can be subdivided into parts, each of which is a reduced-size copy of the whole.
- A fractal is an object that displays ***self-similarity*** at various scales.
- In other words, if we zoom in any portion of a fractal object, we will notice the smaller section is actually a scaled-down version of the big one.
- **Fractals** are related to **chaos** because they are complex systems that have definite properties.

Example: Julia Fractal



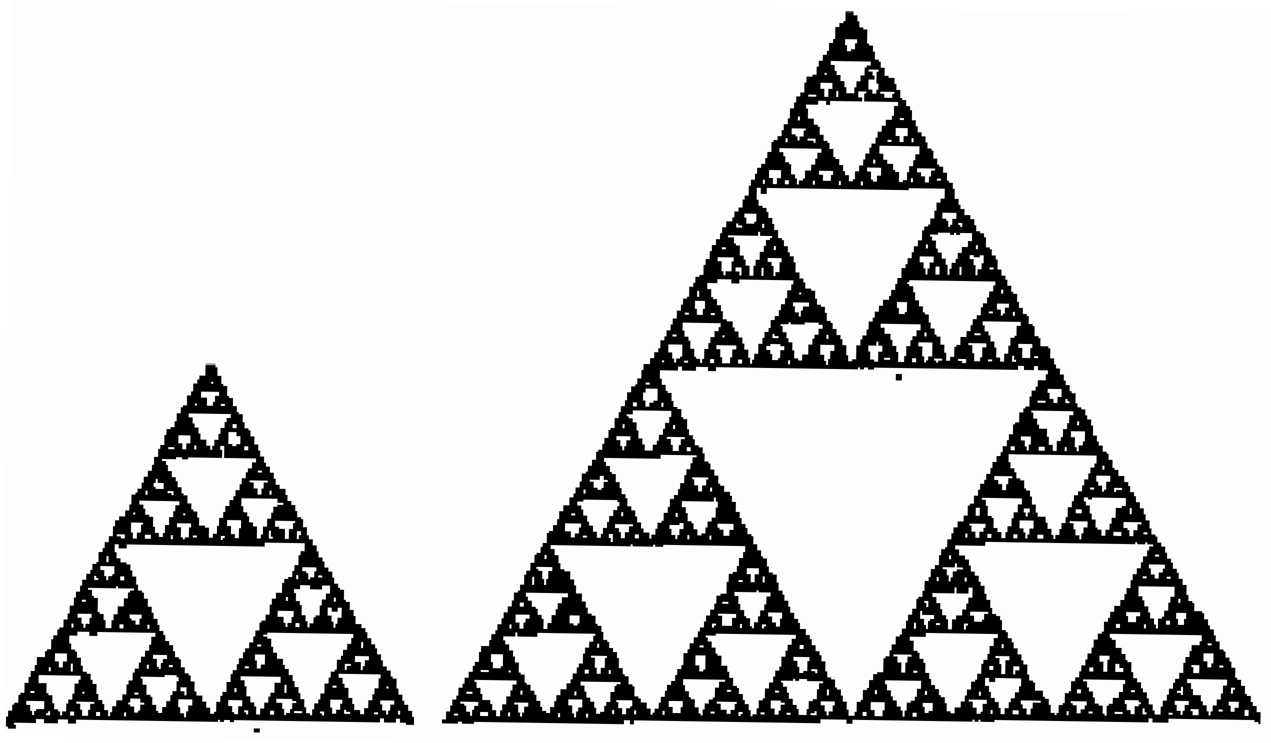
Julia fractal.

Example: Sierpinski Carpet



Self-Similarity

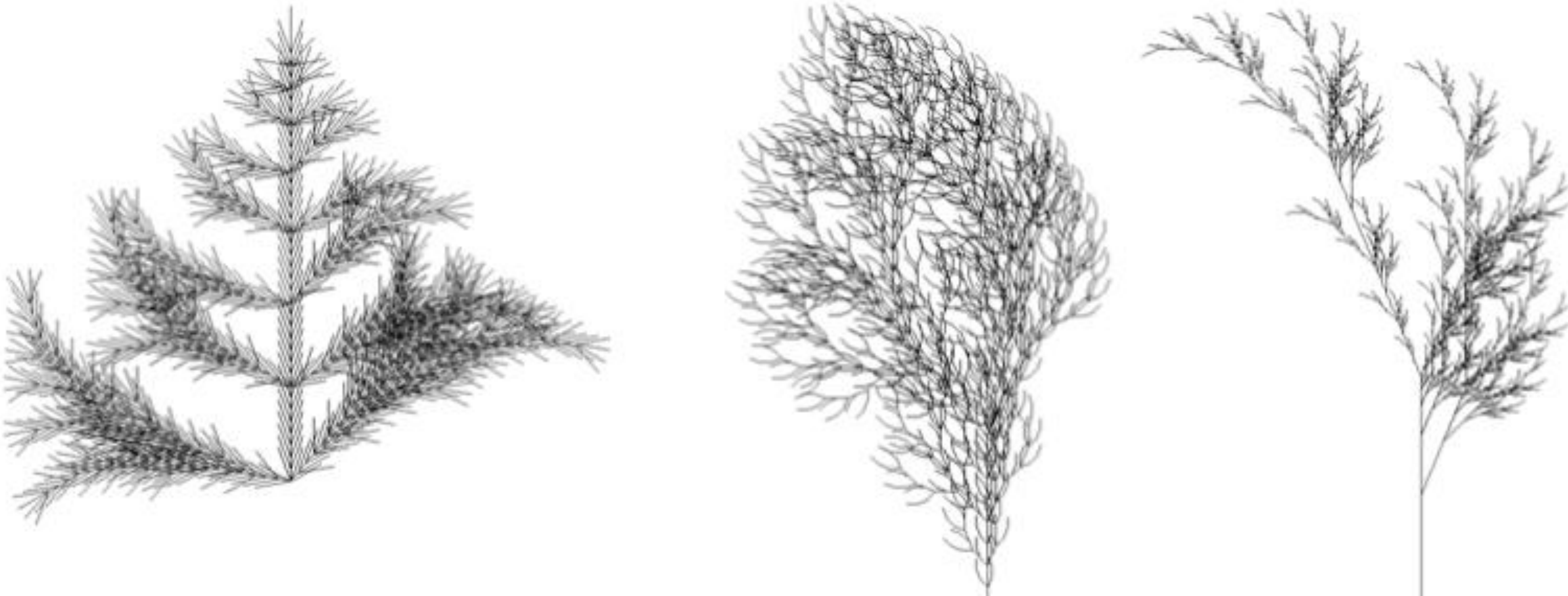
- Let ***A*** and ***B*** be some shapes.
- Then ***A*** is said to be similar to ***B*** if there is an **isomorphism** from ***A*** to ***B***, i.e. if ***B*** can be obtained by a sequence of translations, rescalings, and/or rotations of ***A***.



Fractal Tree

Fractal Food

Fractal – Computer Graphics

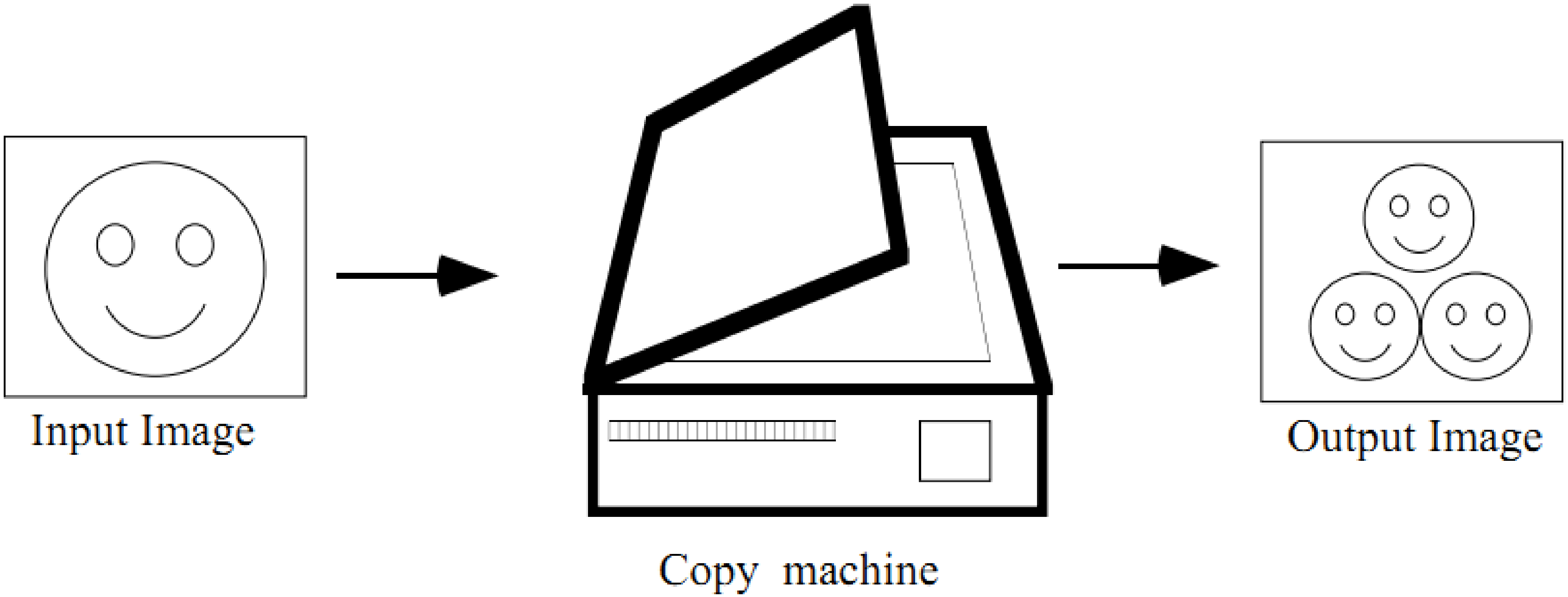


With computer graphics, it takes just a few bytes of data to store the code to generate the above patterns with fractal geometry!

Storing this relatively small amount of data is easier than creating these images themselves.

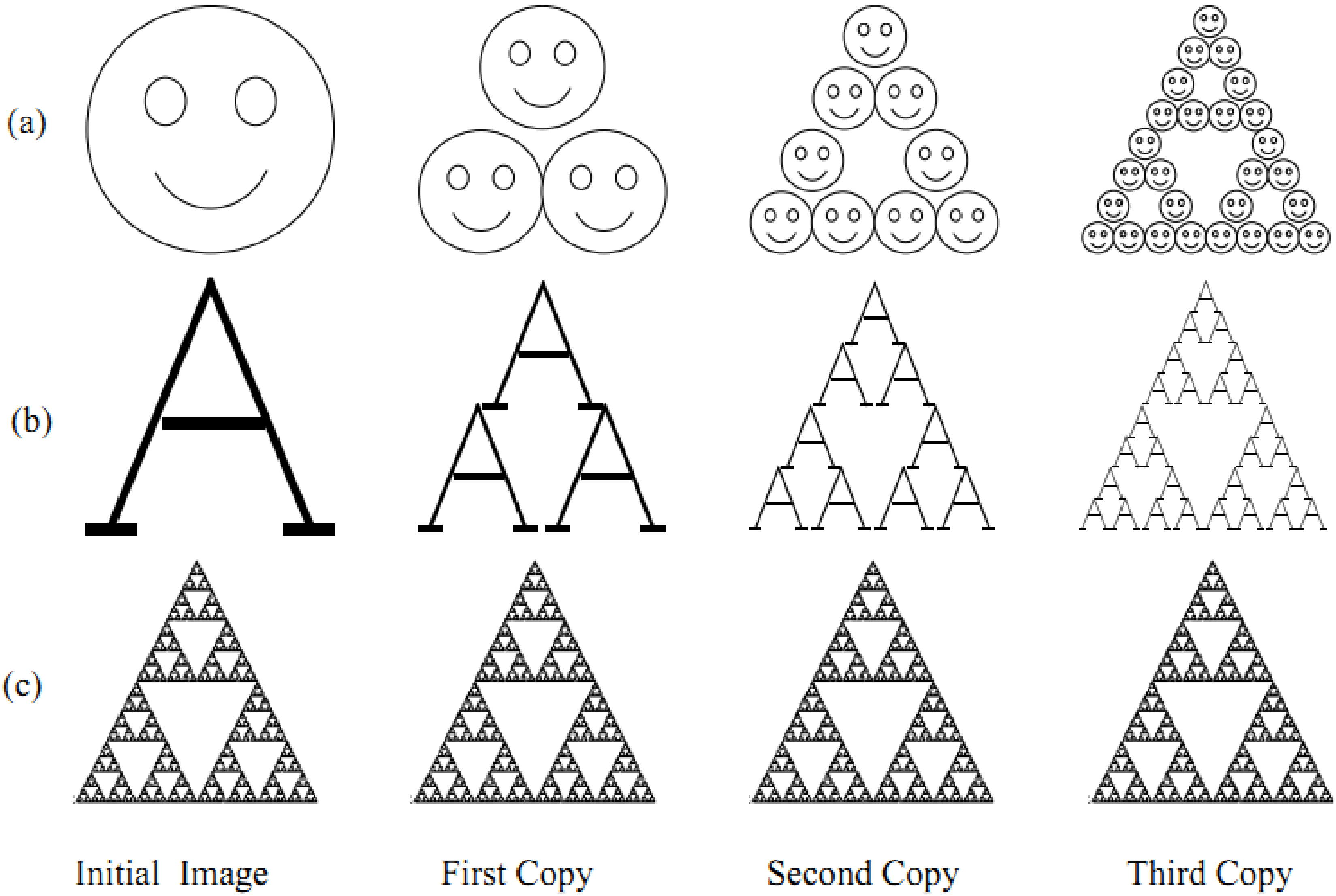
Fractal Geometry

Imagine a special type of photocopying machine that reduces the image to be copied by half and reproduces it three times on the copy.



A copy machine that makes three reduced copies of the input image.

Fractal Geometry – Attractor



Fractal Geometry – Transformations

- Different transformations lead to different attractors
- But the transformations must be ***contractive***, i.e. a given transformation applied to any two points in input image must bring them closer in the copy.

A transformation w is said to be contractive if for any two points $P1$, $P2$, the distance

$$d(w(P1),w(P2)) < sd(P1,P2)$$

for some $s < 1$, where d = distance.

Affine Transformations

- An affine transformation maps a plane to itself.
- The general form of affine transform is

$$w_i \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_i & b_i \\ c_i & d_i \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \end{bmatrix}$$

- Affine transforms can skew, stretch, rotate, scale and translate an input image.

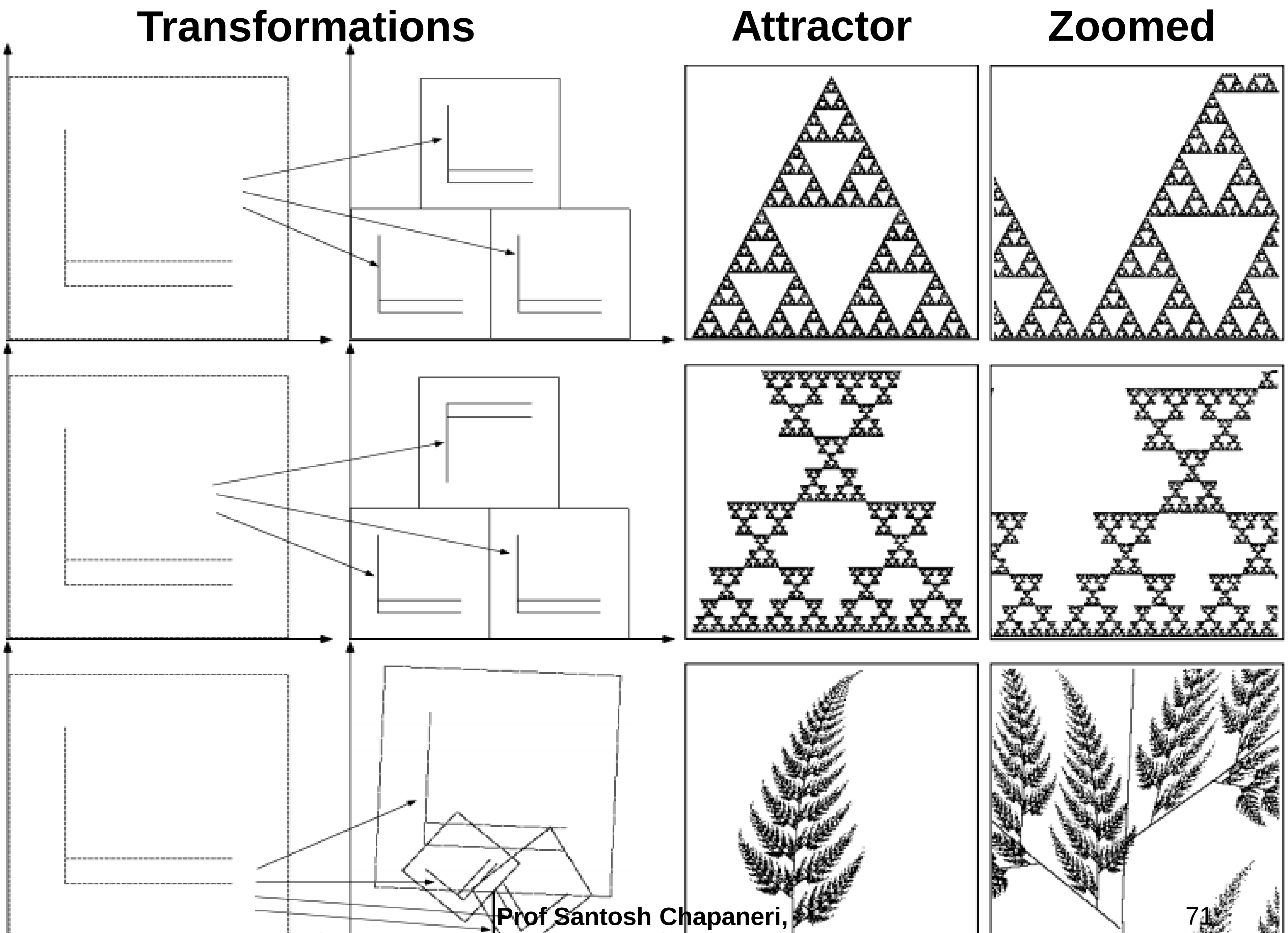
Affine Transformations

- An example of affine contractive transformation:

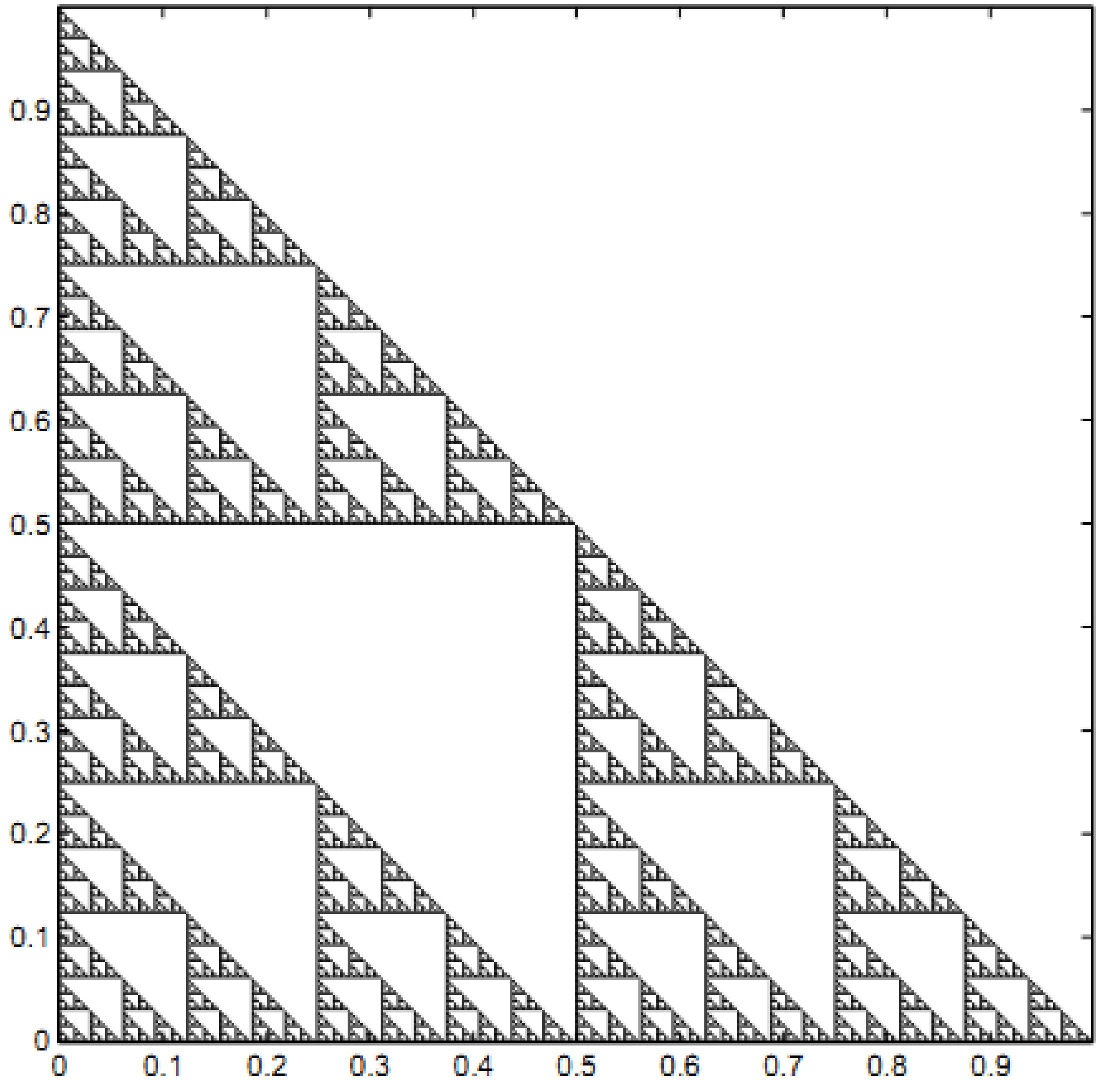
$$w\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix}$$

- This halves the distance between any two points.

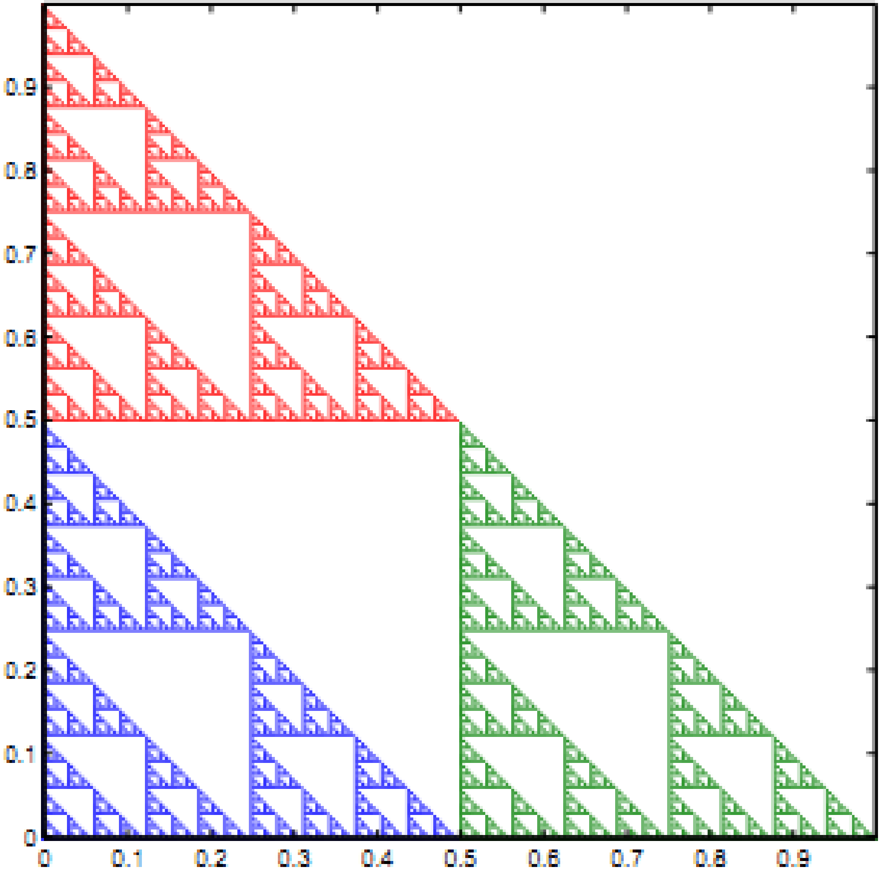
- **Property**: When applied repeatedly, they converge to a point which remains fixed upon further iteration.
- Eg. The map *w* above applied to any initial point (*x*, *y*) will yield (0.5*x*, 0.5*y*), (0.25*x*, 0.25*y*),…eventually converging to (0, 0) which remains fixed.


Transformations
Attractor
Zoomed

Sierpinski’s Gasket



Sierpinski's Gasket



$$w_1 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$w_2 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}$$

$$w_3 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}$$

Sierpinski's Gasket

- Sierpinski's gasket – three transforms, each at $\frac{1}{2}$ scale

$$\left(\frac{1}{2}\right)^s + \left(\frac{1}{2}\right)^s + \left(\frac{1}{2}\right)^s = 1$$

$$\left(\frac{1}{2}\right)^s = \frac{1}{3}$$

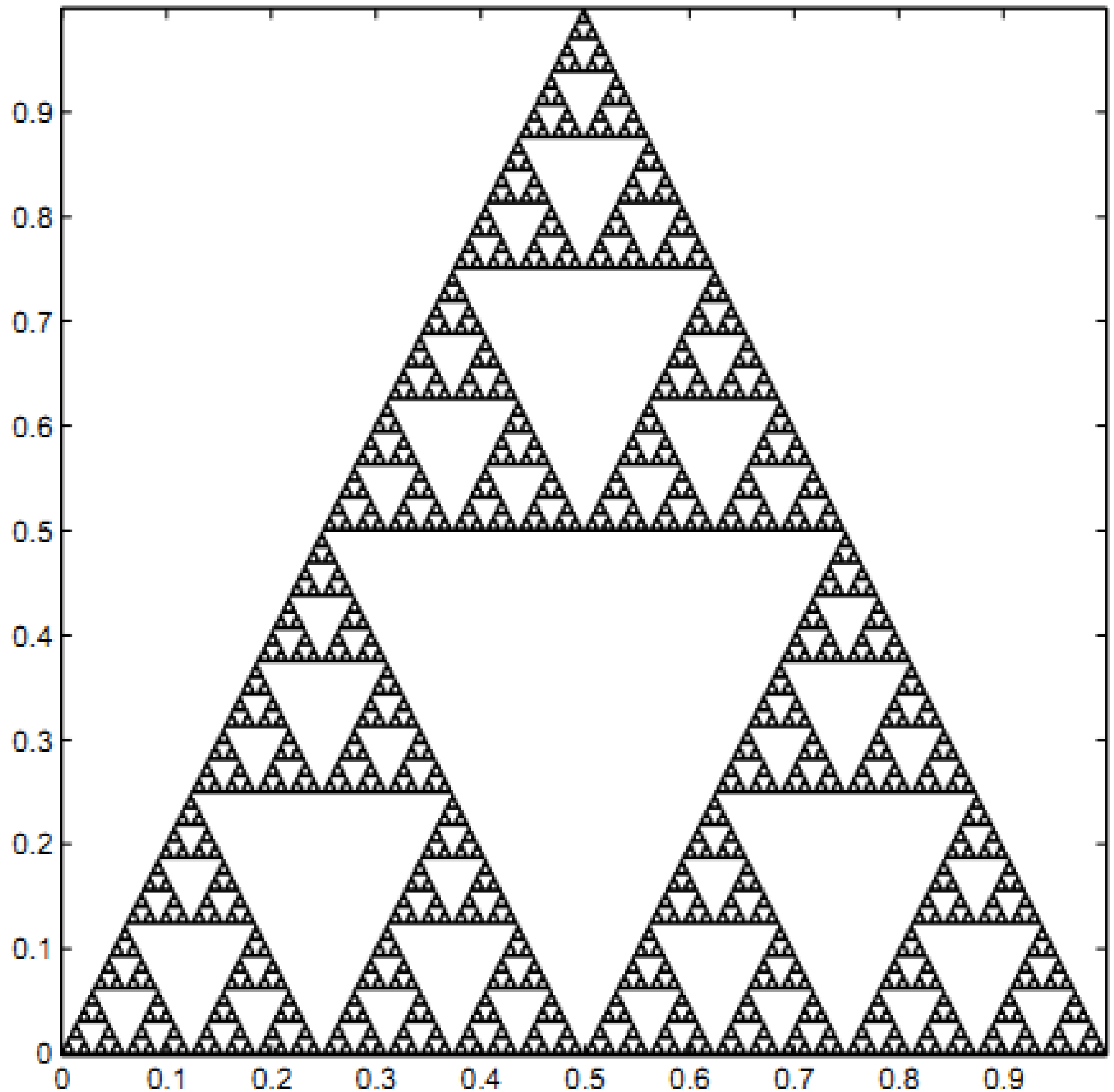
$$s \log \frac{1}{2} = \log \frac{1}{3}$$

$$s = \frac{\log \frac{1}{3}}{\log \frac{1}{2}}$$

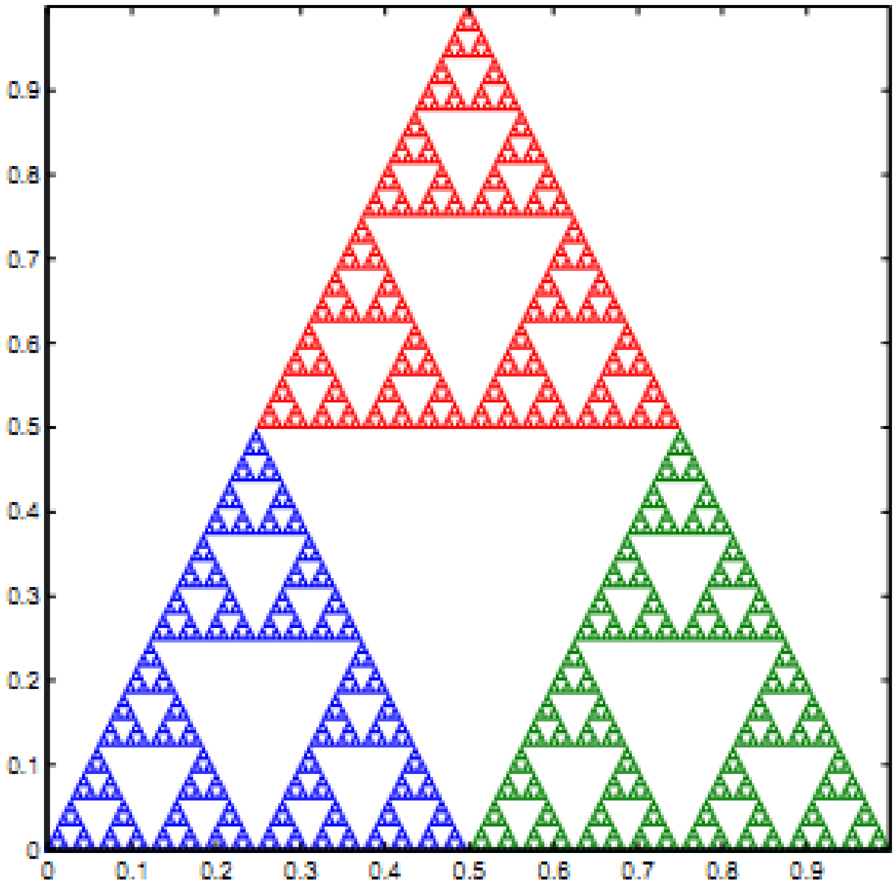
$$s = \frac{\log 3}{\log 2}$$

$$s \approx 1.585$$

Sierpinski's Triangle



Sierpinski's Triangle



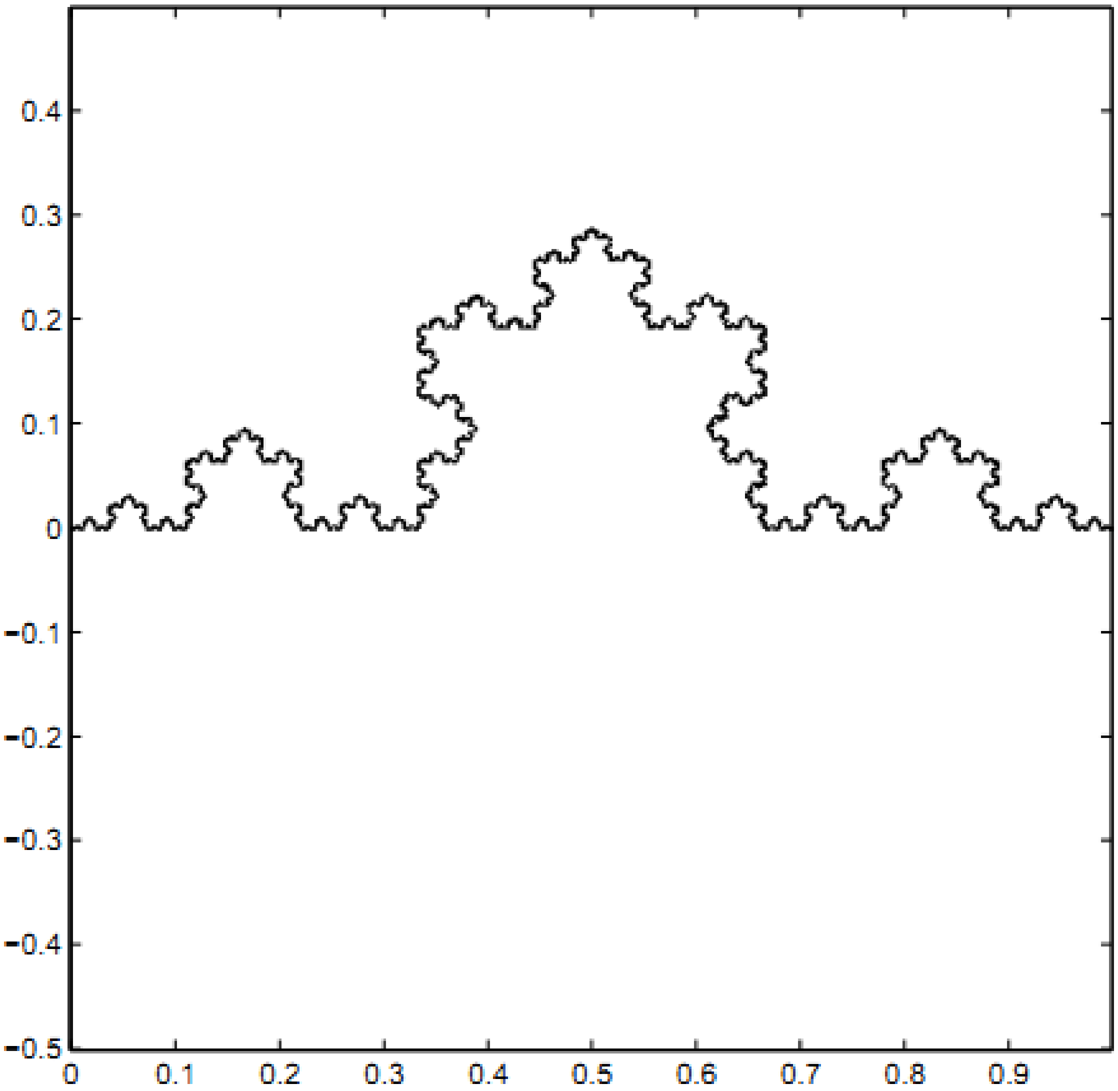
$$w_1 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$w_2 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}$$

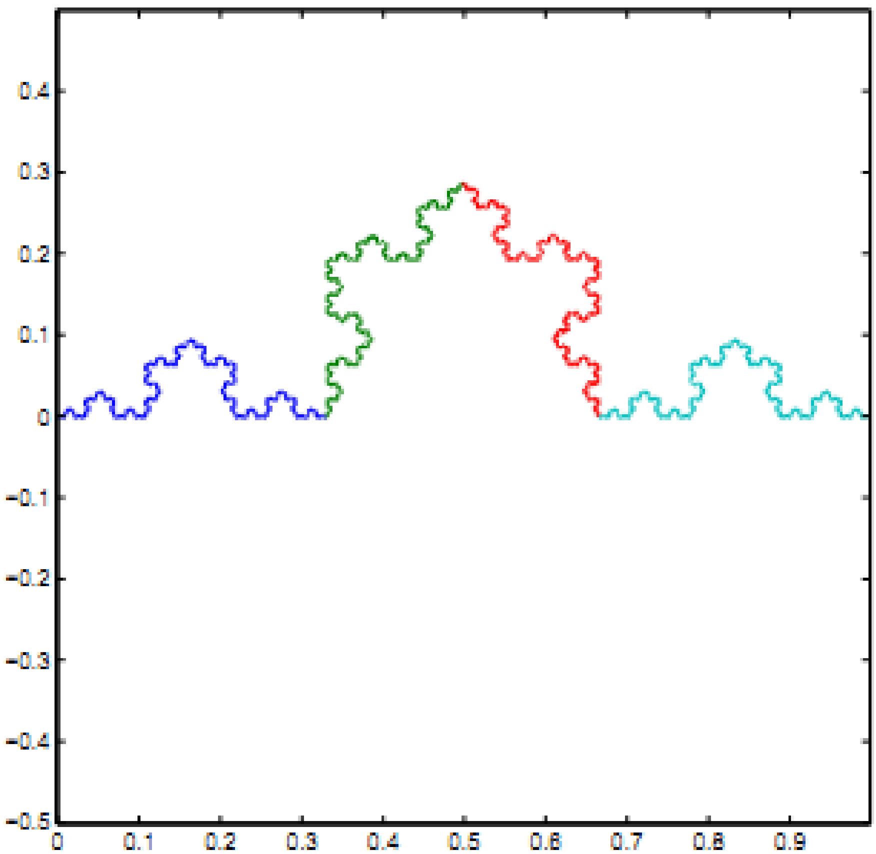
$$w_3 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.25 \\ 0.5 \end{bmatrix}$$

Fractal dimension still $\frac{\log 3}{\log 2}$

Koch Curve



Koch Curve



$$w_1 : \frac{1}{3}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$w_2 : \frac{1}{3}\begin{bmatrix} \frac{1}{2} & -\frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{1}{3} \\ 0 \end{bmatrix}$$

$$w_3 : \frac{1}{3}\begin{bmatrix} \frac{1}{2} & \frac{\sqrt{3}}{2} \\ -\frac{\sqrt{3}}{2} & \frac{1}{2} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{1}{2} \\ \frac{\sqrt{3}}{6} \end{bmatrix}$$

$$w_4 : \frac{1}{3}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} \frac{2}{3} \\ 0 \end{bmatrix}$$

Koch Curve

- Koch Curve – four transforms, each at $\frac{1}{3}$ scale

$$\left(\frac{1}{3}\right)^s + \left(\frac{1}{3}\right)^s + \left(\frac{1}{3}\right)^s + \left(\frac{1}{3}\right)^s = 1$$

$$\left(\frac{1}{3}\right)^s = \frac{1}{4}$$

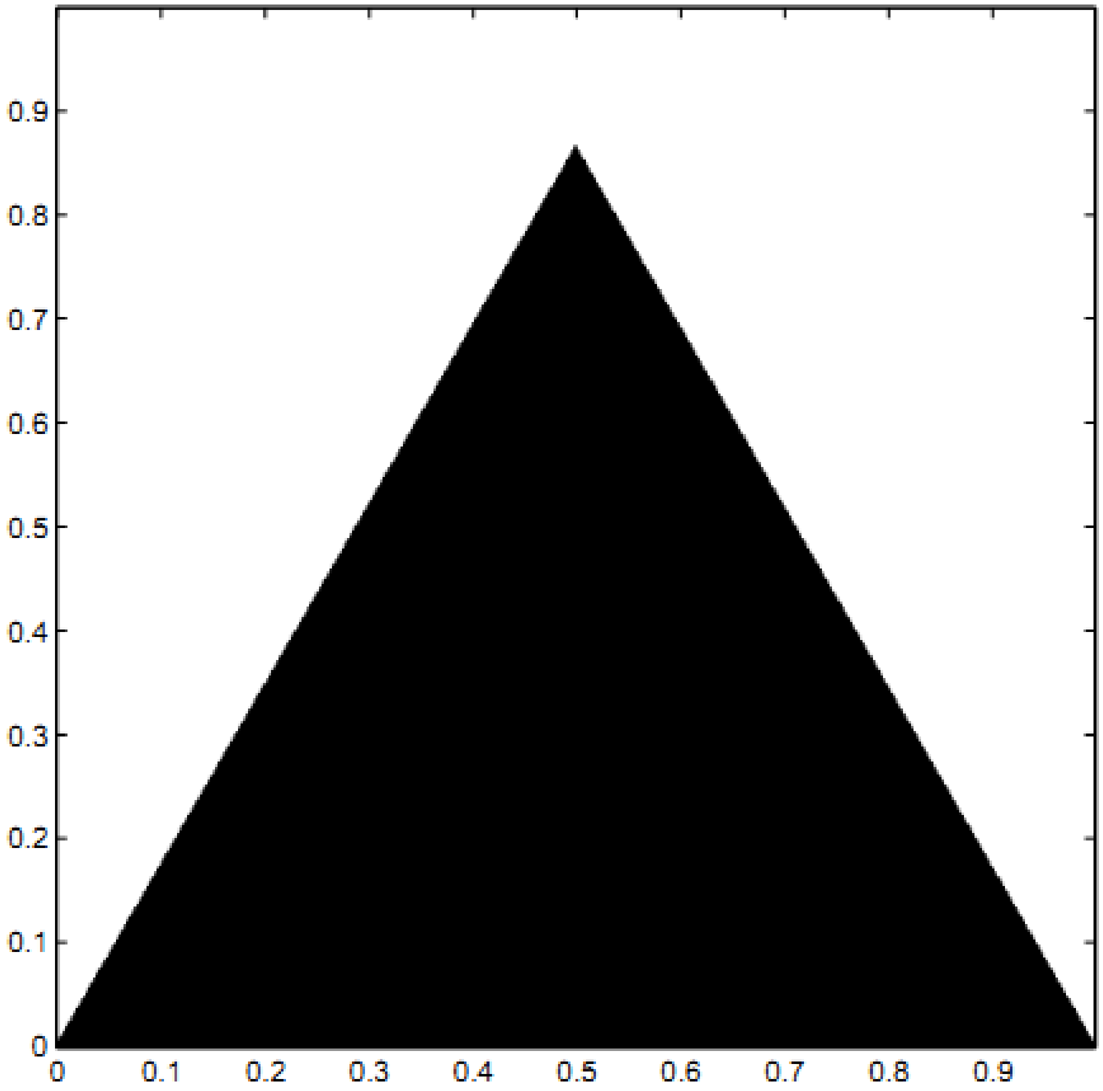
$$s \log \frac{1}{3} = \log \frac{1}{4}$$

$$s = \frac{\log \frac{1}{3}}{\log \frac{1}{4}}$$

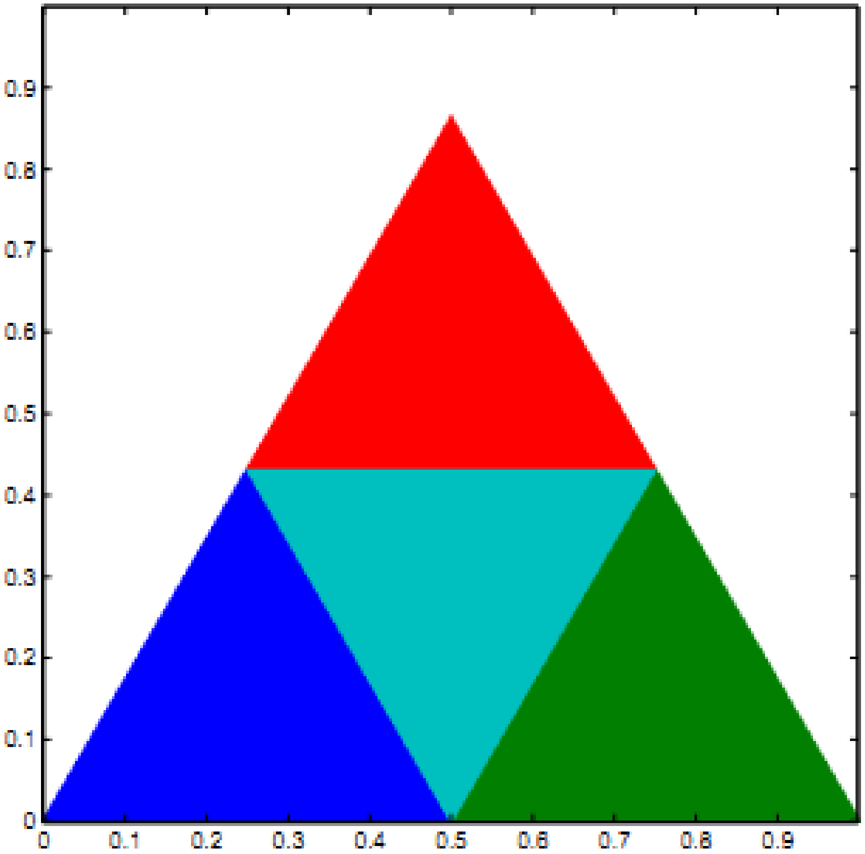
$$s = \frac{\log 4}{\log 3}$$

$$s \approx 1.262$$

Triangle



Triangle



$$w_1 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$w_2 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix}$$

$$w_3 : \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.25 \\ 0.433 \end{bmatrix}$$

$$w_4 : \begin{bmatrix} 0.5 & 0 \\ 0 & -0.5 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.25 \\ 0.433 \end{bmatrix}$$

Triangle

- Four transforms, each at $\frac{1}{2}$ scale

$$\left(\frac{1}{2}\right)^s + \left(\frac{1}{2}\right)^s + \left(\frac{1}{2}\right)^s + \left(\frac{1}{2}\right)^s = 1$$

$$\left(\frac{1}{2}\right)^s = \frac{1}{4}$$

$$s \log \frac{1}{2} = \log \frac{1}{4}$$

$$s = \frac{\log \frac{1}{2}}{\log \frac{1}{4}}$$

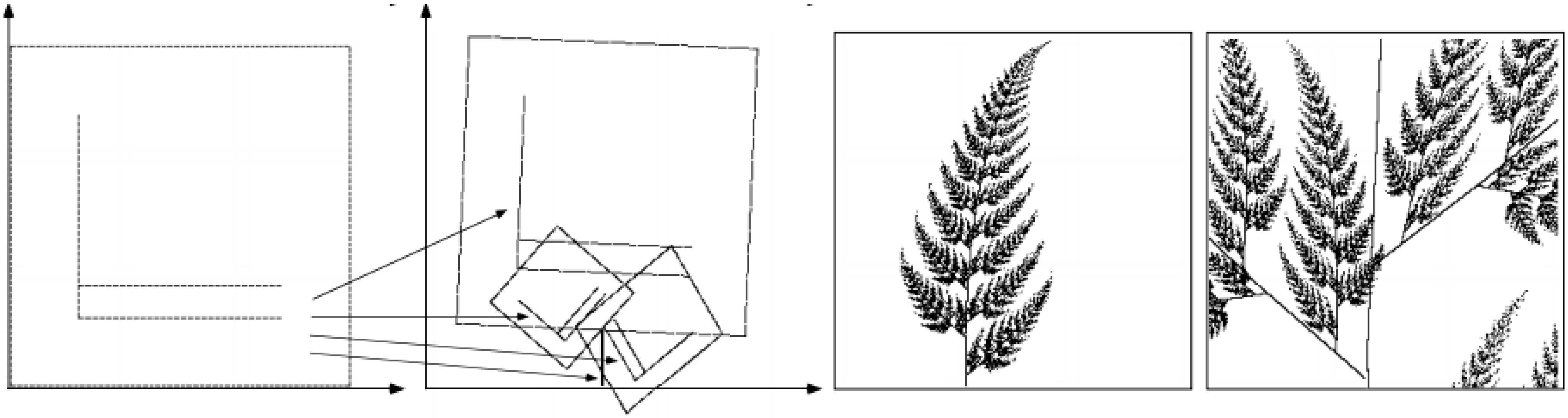
$$s = \frac{\log 4}{\log 2}$$

$$s = 2$$

For Triangle,
Fractal Dimension =
Topological Dimension

Fractal Image Compression

- M. Barnsley [1] suggested that storing images as collections of transformations could lead to image compression.



- Original image size = 65,536 bits.
- The above fern is made up of affine transformations consisting 6 numbers: 4 transformations x 6 numbers/transformation x 32 bits/number = 768 bits.

[1] Barnsley, M., ***Fractals Everywhere***, Academic Press, San Diego, 1989

Iterated Function System (IFS)

- Running the special photocopy machine in a feedback loop is a metaphor for a mathematical model called **Iterated Function System (IFS)**
- An IFS consists of a collection of **contractive** transformations

$$\{w_i : w_i \to R^2 \mid i = 1, \ldots, n\}$$

which map the plane R^2 to itself.

Self-Similarity in Images

- A typical image does not contain the type of self-similarity found in fractals. But it may contain **some form of self-similarity at different scales**.

Prof Santosh Chapaneri.
santoshchapaneri@gmail.com

Partitioned Iterated Function System

- For image compression, we extend the IFS to allow us to partition an image into **pieces/parts** which are each transformed separately.
- The Partitioned IFS is given by:

$$w_i \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} e_i \\ f_i \\ o_i \end{pmatrix}$$

- Here, s_i and o_i are the contrast and brightness adjustments for the transformations.

Ranges and Domains

- The problem of fractal image compression is to **find the best domain that will map to a range**.

- A *domain* is a region where the transformation maps from.

- A *range* is a region where the transformation maps to.

Fractal Image Encoding

- Consider a grayscale image of size 256 x 256 pixels
- Partition it into 8 x 8 non-overlapping ranges R_1, R_2, R_3, …, R_{1024}
- Also, partition it into 16 x 16 overlapping domains D_1, D_2, D_3, …, D_{58081}
- For each R_i, search through all D to find a D_i which minimizes some error threshold => i.e. try to find a part of the image that looks similar to R_i
- **Result**: Original image size = 65 kB, Transformation size = 3 kB, Compression ratio = **16.5:1**
- Fast processing techniques are available in the literature to reduce the time complexity from O(N) to O(log N)

Fractal Image Decoding

Start with *any* image

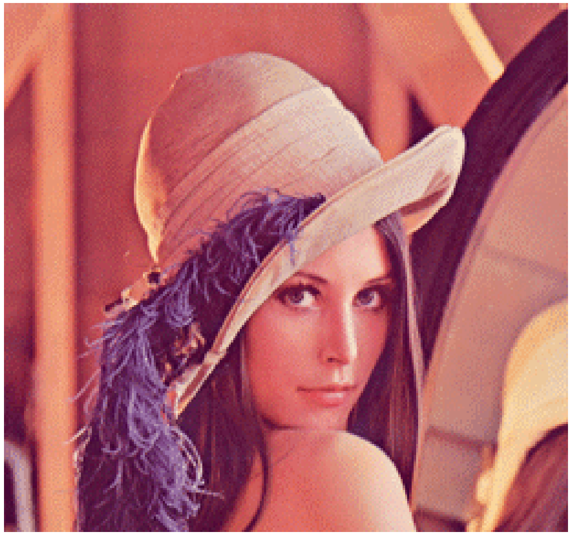
1st Iteration

2nd Iteration

10th Iteration

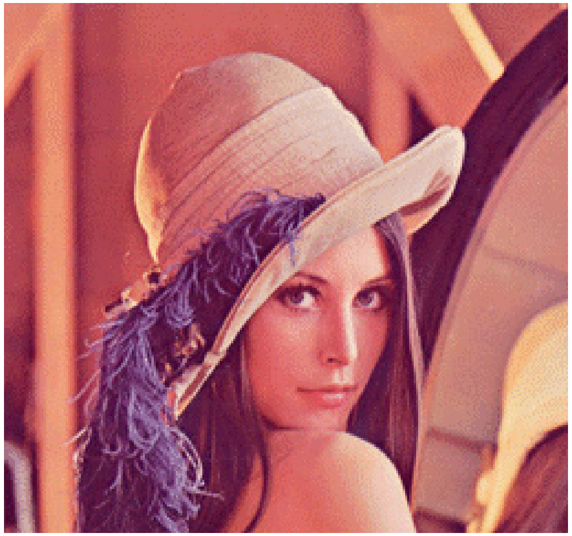
Prof Santosh Chapaneri, santoshchapaneri@gmail.com

Fractal Image Compression



Original Lena image

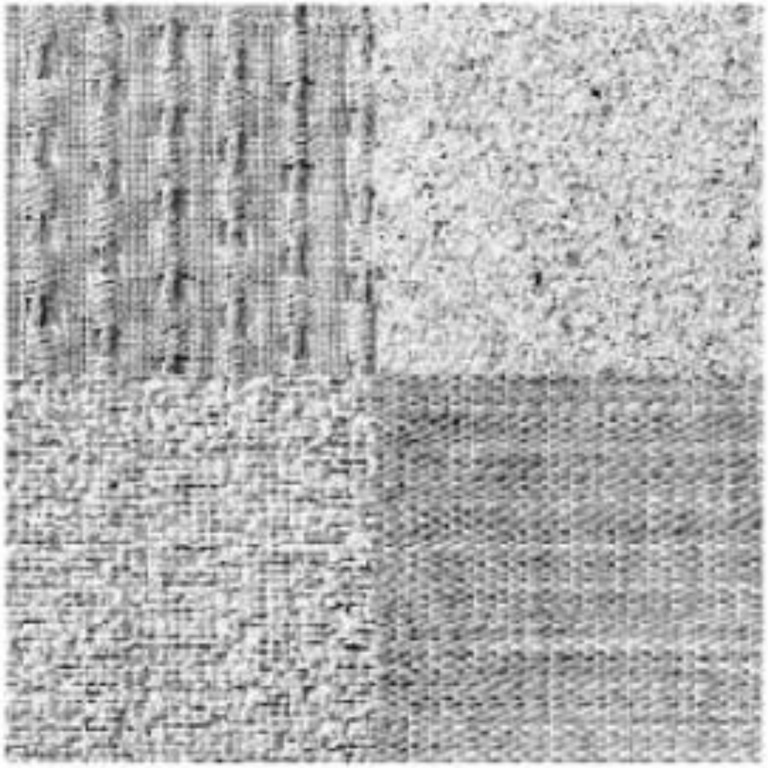
(184 kB)



JPEG Compression
comp. ratio: **5.75:1**

Fractal Compression
comp. ratio: **6.07:1**

Fractal Image Segmentation



(a)

(b)

Original Texture Image

Segmented Image using Fractal Geometry

References

1) Y. Fisher, Fractal Image Compression – Theory and Applications, Springer-Verlag, 1994
2) A. E. Jacquin, “Image coding based on a fractal theory of iterated contractive image transformations”, IEEE Trans. Image Processing, vol. 1, no. 1, pp. 18-30, Jan 1992
3) F. Davoine, M. Antonini, J. M. Chassey, M. Barlaud, “Fractal Image Compression Based on Delaunay Triangulation and Vector Quantization," IEEE Trans. on Image Processing, vol. 5, no. 2, pp. 338-346, Feb. 1996
4) D. Saupe, R. Hamzaoui, “A review of the fractal compression literature," Computer Graphics, vol. 28, no. 4, pp. 268-276, 1994
5) Y. Fisher, Fractal Image Compression, Siggraph 1992 course notes
6) Huang Q., Lorch J.R., Dubes R.C. ,“Can the fractal dimension of images be measured?”, Pattern Recognition, 27(3), 339-349