

# ***Fractales y Caos en Sistemas no Lineales. Aplicaciones***

---



**UC**



**INM**

**Dr. José Manuel Gutiérrez**

*Dpto. de Matemática Aplicada y  
Ciencias de la Computación*

**Universidad de Cantabria**

<http://personales.unican.es/gutierjm>

***UNPA, Río Gallegos, 29 de Septiembre del 2000***

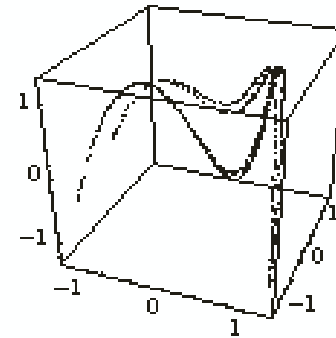
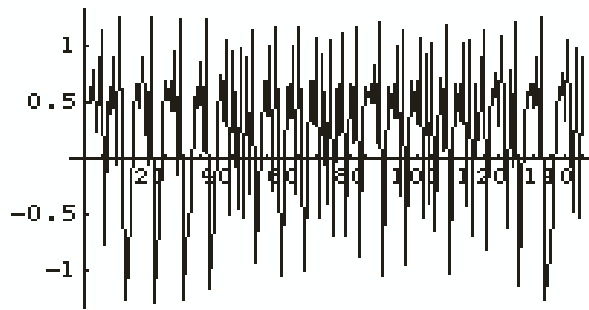
# Sistemas Dinámicos

Un sistema dinámico es cualquier proceso donde se tiene una magnitud que varía en el tiempo según una ley **determinista**, o **estocástica**.

- **Movimiento de los planetas.**
- **Evolución de la configuración atmosférica.**
- **Evolución de especies en un entorno.**
- **Lanzamiento de dados.**

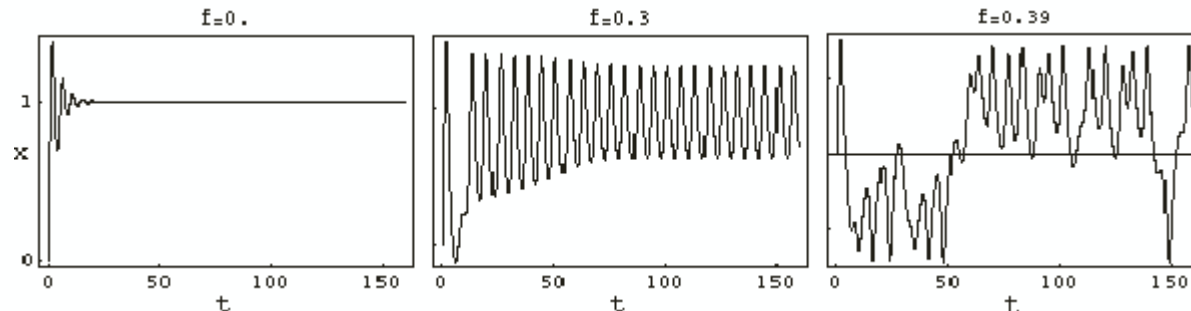
Un ejemplo es la serie de Henon.

$$x_{n+1} = 1 - 1.4 x_n^2 + 0.3 x_{n-1}$$



Otro ejemplo es un oscilador:

$$x'' + ax' + x^3 - x = f \cos(\omega t)$$



## Iteración. Conjunto de Mandelbrot

La iteración de un sistema lineal sólo puede dar lugar a una sucesión creciente, o a una sucesión que converge a cero.

La dinámica iterativa de ecuaciones no lineales puede dar lugar a comportamientos “extraños”.

Por ejemplo, consideremos la función  $z^2+c$ , para distintos valores de  $c$ .

Para iterar esta ecuación, comenzamos con lo que llamaremos una *semilla*  $z_0$   
Aplicando la función a obtenemos  $z_1 = z_0^2 + c$ ,  $z_2 = z_1^2 + c$ , ... “**órbita**”

- ¿Cuál es el destino de estas órbitas?
- ¿Convergen o divergen?
- ¿Son cíclicos o se comportan erráticamente

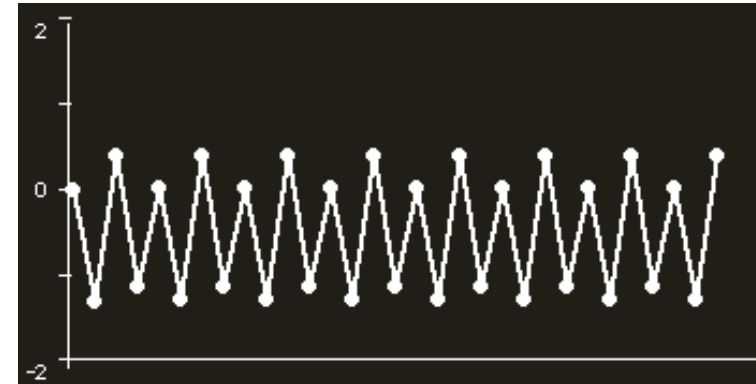
**Consideremos el valor  $c = 1$ .**

Si elegimos la semilla  $0$ , la órbita es:  $z_0 = 0$ ,  $z_1 = 1 = 0^2 + 1$ ,  $z_2 = 2$ ,  $z_3 = 5$ ,  $z_4 = 26$ ; es decir, la órbita diverge.

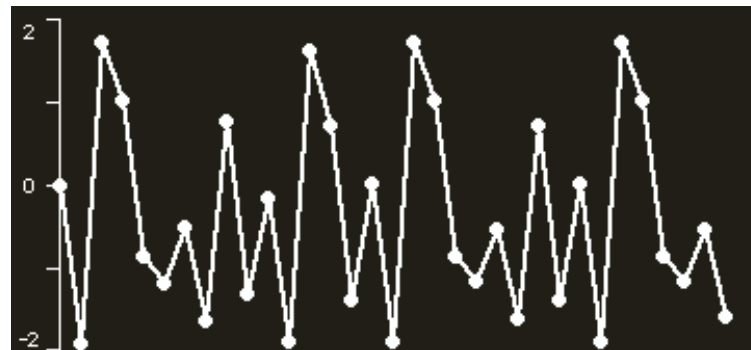
Consideremos el valor  $c = -1.0$



Consideremos el valor  $c = -1.3$



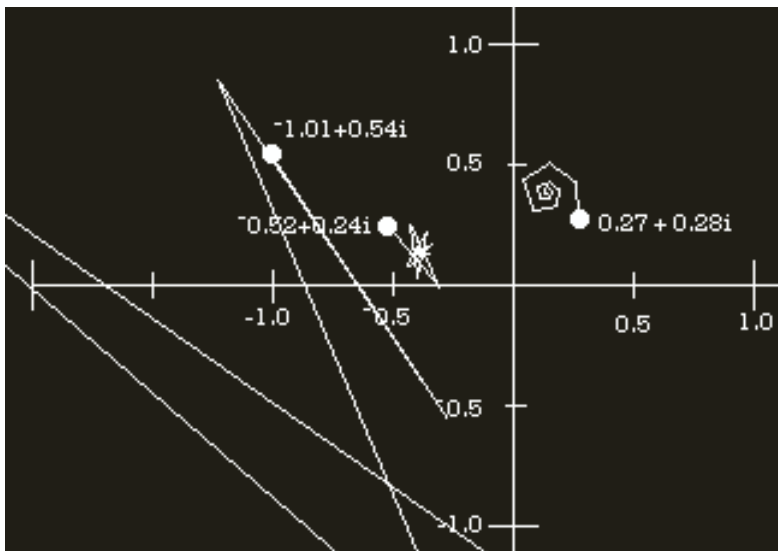
Consideremos el valor  $c = -1.9$



*Para distintos valores del parámetro se tienen distintos comportamientos del sistema: Divergencia, periodicidad y ?*

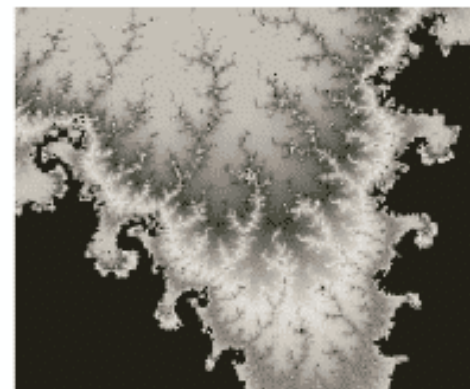
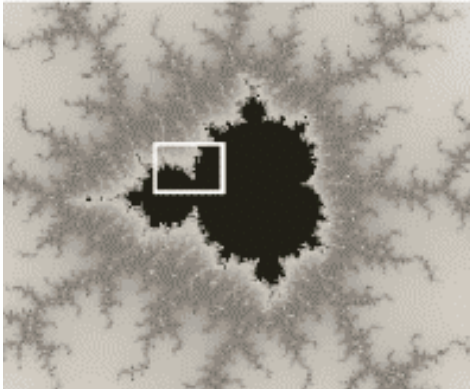
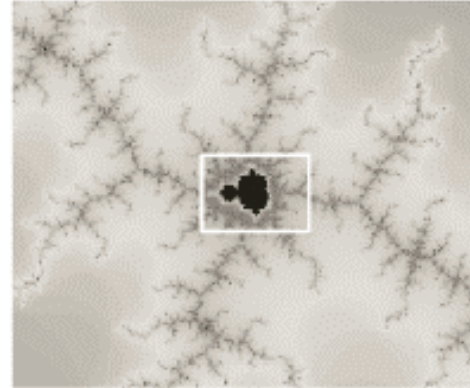
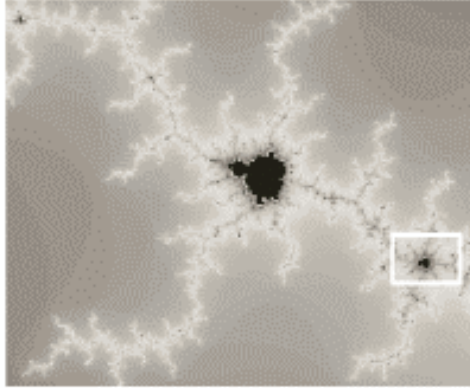
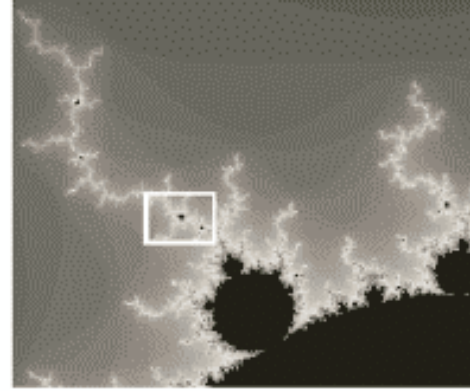
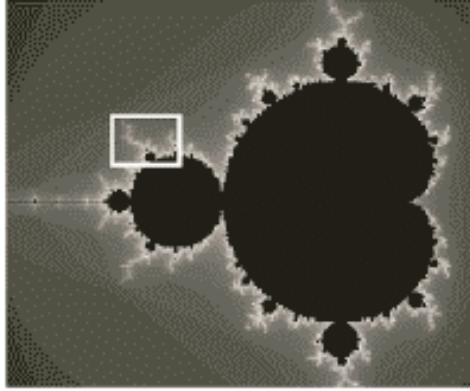
En general, la órbita asociada a un número complejo asociado será una órbita de números complejos, pero la dinámica es la misma.

La figura siguiente muestra los procesos iterativos asociados a tres condiciones iniciales distintas (los números complejos  $0.27+0.28i$ ,  $-0.52+0.24i$ , y  $-1.01+0.54i$ , respectivamente).



*El conjunto de Mandelbrot es una versión gráfica de los posibles comportamientos de este proceso iterativo. El conjunto de Mandelbrot está formado por aquellos valores  $c$  del plano complejo tales que la órbita de  $0$  está acotada.*





*Nos encontramos  
con un objeto  
sorprendente que  
posee estructura a  
cualquier escala y  
contiene copias de sí  
mismo.*

***FRACTAL !!!!***

```

#include <graphics.h>
#include <conio.h>

void main()
{
    int sx=320;           //Screenwidth
    int sy=240;           //Screenheight
    double xmin=-2;      //smallest real value (x-axis)
    double xmax=1.25;    //largest real value (x-axis)
    double ymin=-1.25;   //smallest imaginary value (y-axis)
    double ymax=1.25;    //largest imaginary value (y-axis)
    int maxiter=96;      //Max number of iterations

    double old_x;        //temporary variable to store x-value
    double fx,fy;
    int m;                //variable to store number of iterations

    int gd=VGA,gm=2;     //MS-DOS spesific graphics-init.

    double dx=(xmax-xmin)/sx; //how much to add for each x-pixel?
    double dy=(ymax-ymin)/sy; //how much to add for each y-pixel?

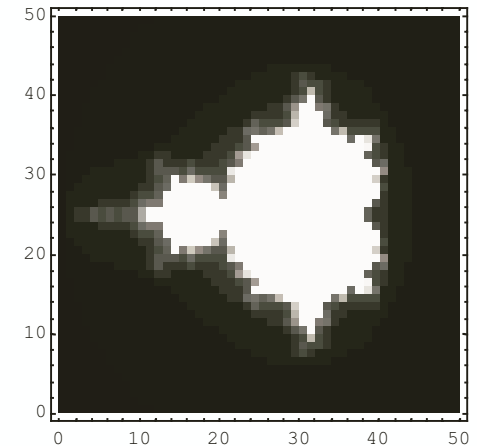
    int px;               //Variable storing current x-pixel
    int py=0;             //Variable storing current y-pixel
    double x;             //Variable storing current x-value
    double y=ymin;        //Variable storing current y-value
    initgraph(&gd,&gm,"c:\\tc\\bgi\\"); //code. Update the path!!!

    while (py<sy) {
        px=0;
        x=xmin;
        py++;
        while (px<sx) {
            px++;

            fx=0;
            fy=0;
            m=0;
            do {
                old_x=fx;
                fx=fx*fx-fy*fy+x;
                fy=2*old_x*fy+y;
                m++;
            } while (((fx*fx+fy*fy)<4) && (m<maxiter));
            putpixel(px,py,m); //MS-DOS spesific code to give pixel(x,y)
                               //the color (m)
            x+=dx;
        }
        y+=dy;
    }

    getch();
    closegraph();        //MS-DOS spesific code to close graphicscreen
}

```

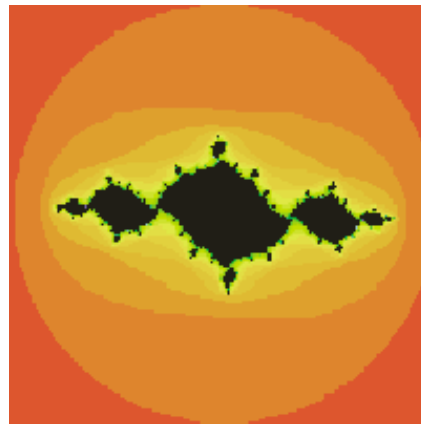


# Conjuntos de Julia

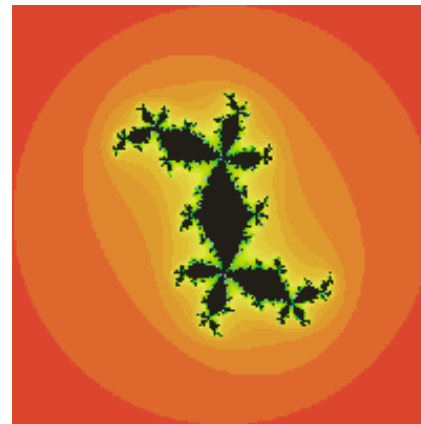
El conjunto Julia es sutilmente diferente del conjunto Mandelbrot. Para  $M$ , calculamos sólo la **órbita de 0 para cada valor de  $c$** .

Para los conjuntos Julia, **fijamos un valor de  $c$**  y luego consideramos el destino de todas las posibles semillas para ese valor fijo de  $c$ . Aquellas semillas cuyas órbitas no escapan, forman el conjunto Julia.

Por tanto obtenemos un conjunto Julia diferente para cada  $c$  diferente.



$c = -1.037 + 0.17i$



$c = 0.295 + 0.55i$

Uno de los teoremas más sorprendentes de la geometría fractal data de 1919 y fue comprobado por Gastón Julia y Pierre Fatou. La **Dicotomía Fundamental** para la ecuación es: para cada valor de  $c$ , el conjunto de Julia es o un conjunto conexo (una pieza) o un conjunto completamente desconexo (infinitas piezas).

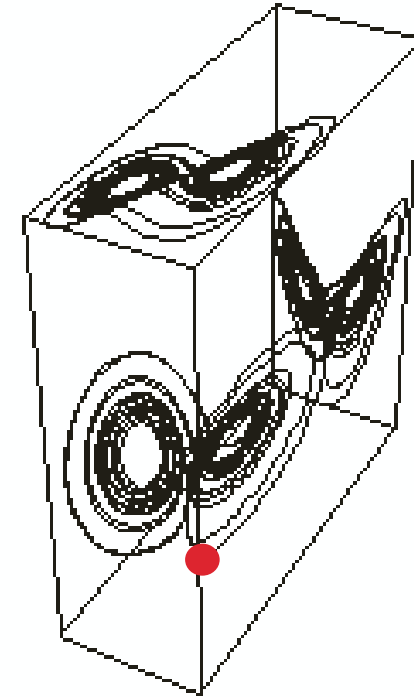
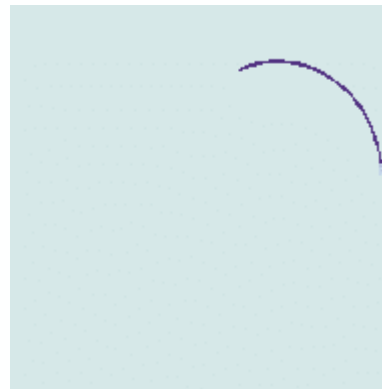


# El sistema de Lorenz. Origen del Caos.

**Sistema de Lorenz**  
 inspirado en un  
 modelo simple de  
 dinámica de fluidos,  
 donde se observó  
 caos por primera vez.

$$\begin{cases} \frac{dx}{d\tau} = \alpha(y - x - f(x)) \\ \frac{dy}{d\tau} = x - y + z \\ \frac{dz}{d\tau} = -\beta y - \gamma z \end{cases}$$

**Sensibilidad a  
 las condiciones  
 iniciales**

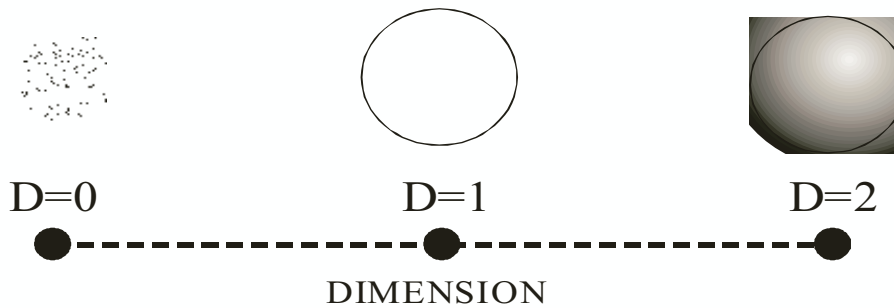


**Geometría compleja y “extraña”**

La geometría fractal y la teoría de los sistemas dinámicos están íntimamente ligadas, ya que la región del espacio hacia la que tiende asintóticamente una órbita caótica tiene estructura fractal (**atractores extraños**). Por tanto, la geometría fractal permite estudiar el soporte sobre el que se definen los sistemas dinámicos caóticos.

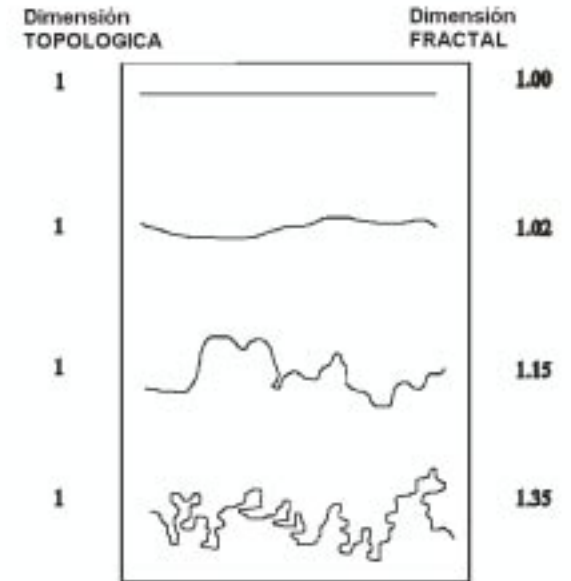
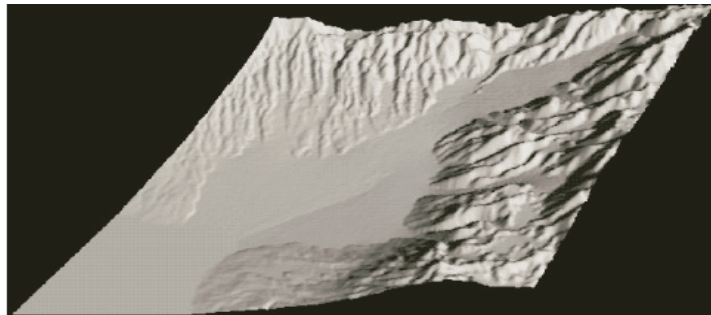
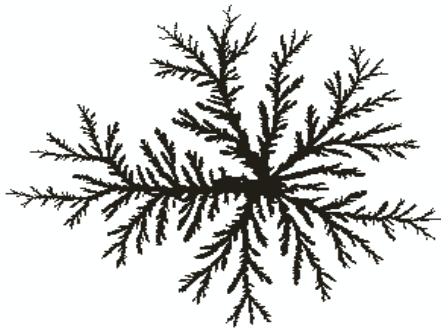
# La Geometría Fractal

La geometría tradicional (euclídea) se encarga de las propiedades y de las mediciones de objetos tales como **puntos**, **líneas**, **planos** y **volúmenes**.



Sin embargo, no todas las curvas son “iguales”.

Existen otros **objetos geométricos irregulares** con infinitas singularidades (puntos no diferenciables).



Clouds are not spheres, mountains are not cones, nor does lightning travel in a straight line.

***Benoit B. Mandelbrot***

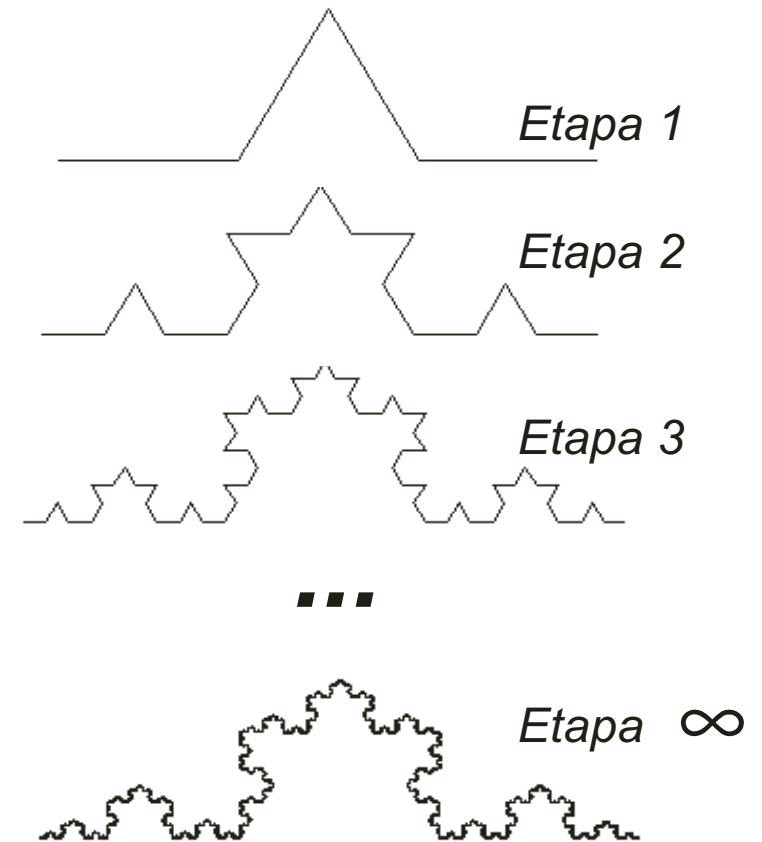
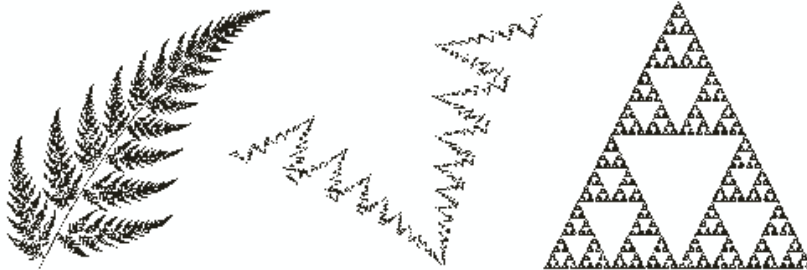
*Benoit B. Mandelbrot: The fractal geometry of nature. W.H. Freeman and Company, New York, 1983.*

# Fractales en las Matemáticas

Desde principios de siglo matemáticos como **Cantor**, **Poincaré** o **Julia**, se interesaron por el estudio de objetos extraños (monstruos matemáticos) que no encajaban en las ideas de la geometría clásica.

Muchos de estos objetos se construían mediante algoritmos iterativos, partiendo de un “iniciador” y aplicando reiterativamente un conjunto de transformaciones.

De esta forma se pueden definir gran cantidad de objetos matemáticos con propiedades comunes, como la **autosemejanza**.

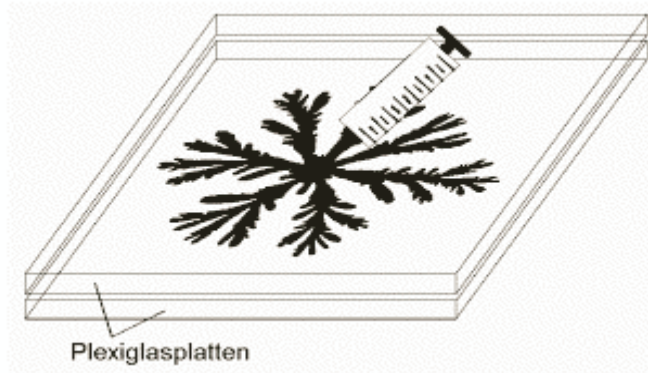


Longitud infinita encerrada en un área finita !!!!



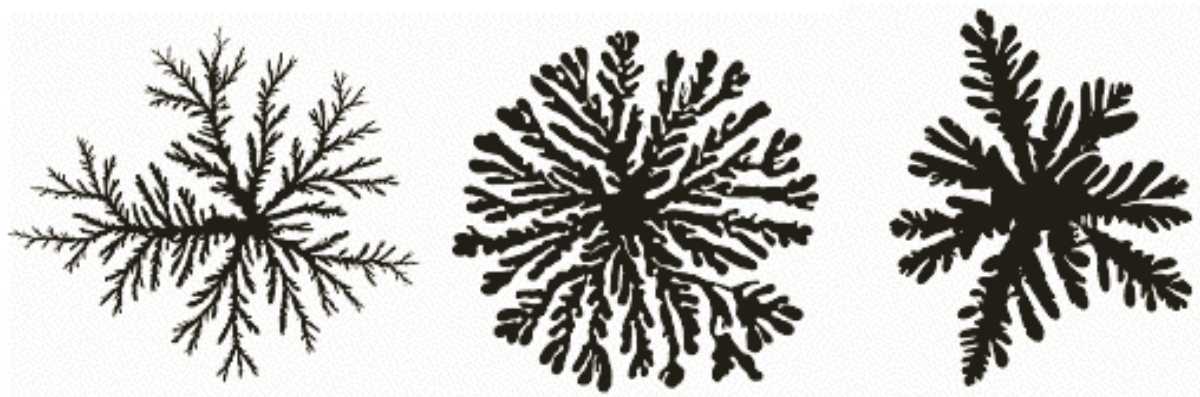
## Fractales en el “mundo real”

Un ejemplo clásico de estructura fractal se observa en hidrodinámica cuando un fluido poco viscoso desplaza a otro más viscoso. La interfase que se crea tiene estructura compleja.



### *Experimento de Henry S. Hele-Shaw*

Entre dos planchas de cristal muy próximas se dispone un fluido muy viscoso. A continuación se inyecta en un punto y con cierta velocidad un fluido menos viscoso.



- Viscosidad de los fluidos.
- Velocidad de inyectado.
- Tensión de la interfase.
- Etc.

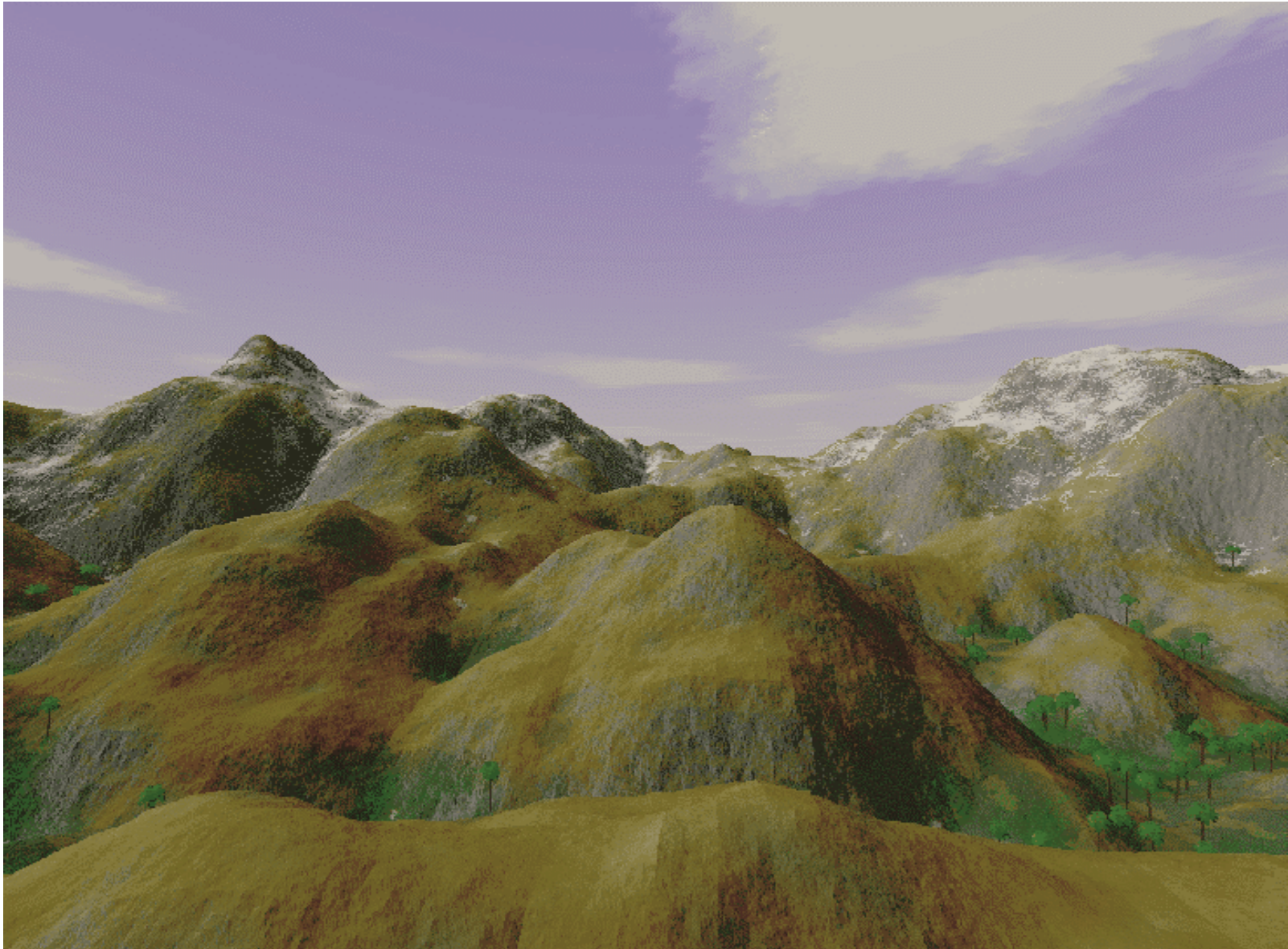
**Autosemejanza estadística !!!**

Tamás Vicsek: Fractal Growth Phenomena.  
World Scientific Publishing, 1992.

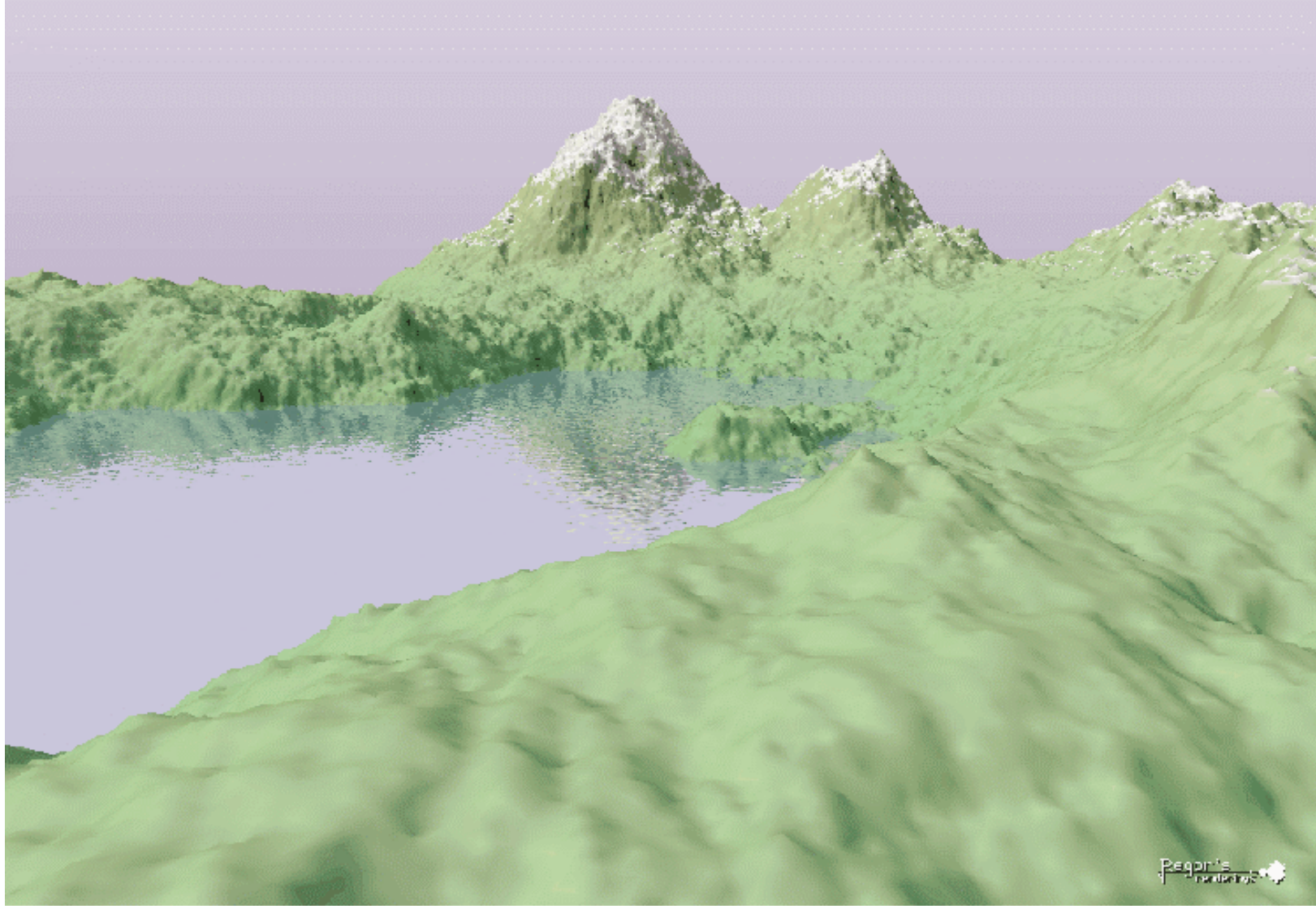


# Paisajes Fractales (realidad virtual)

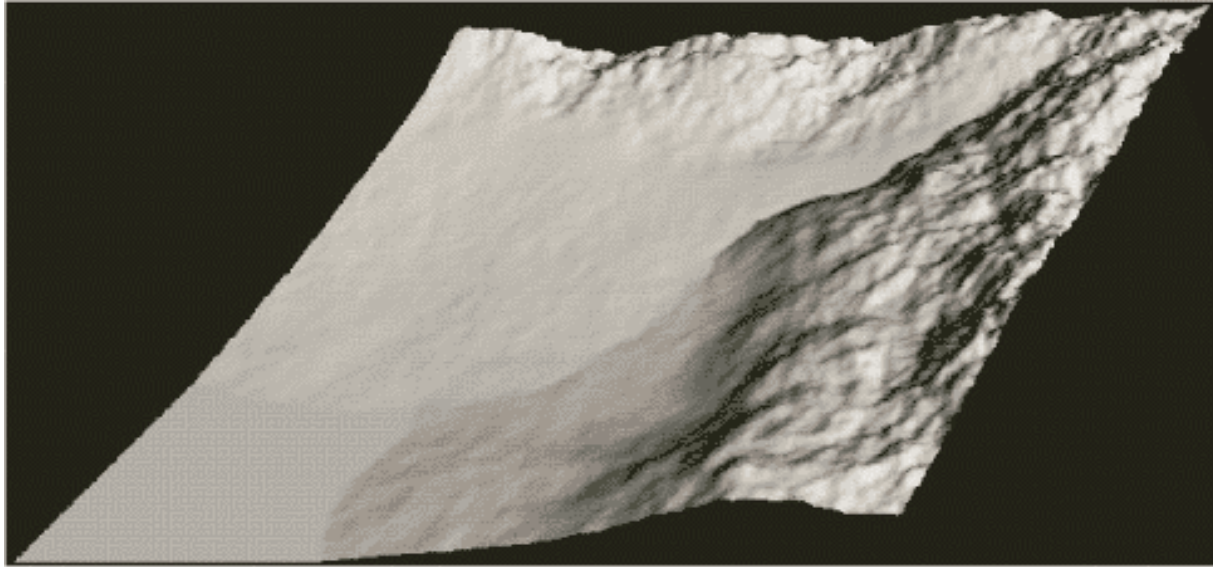
José Manuel Gutiérrez, Universidad de Cantabria (2000)



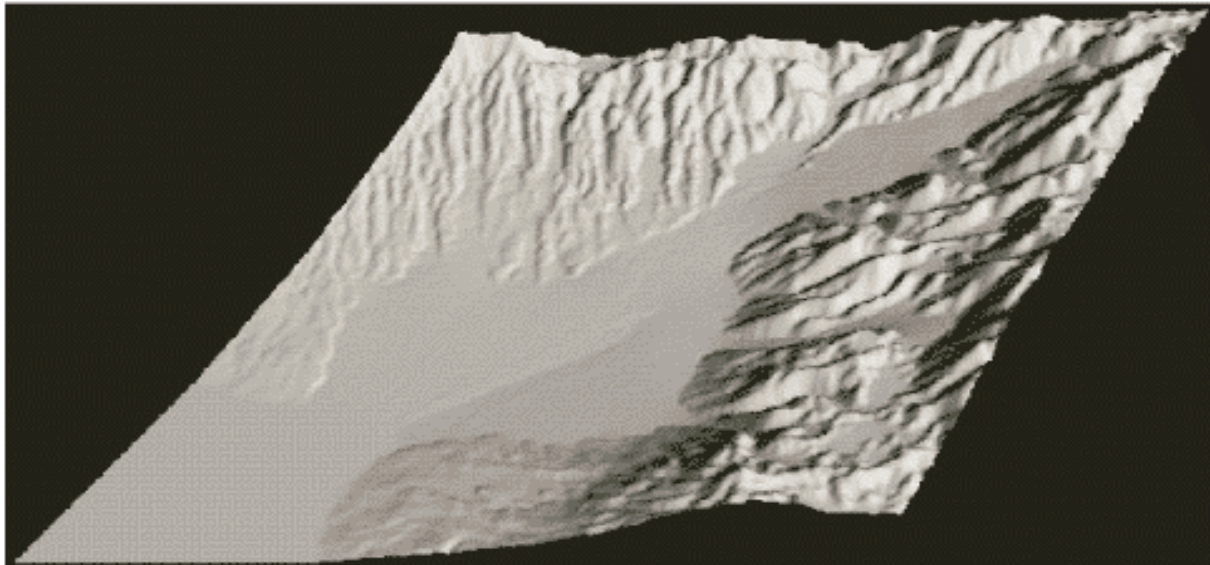




## Fractales en Modelización: Erosión del Terreno



También es posible modelizar mediante la geometría fractal diversos procesos naturales como, por ejemplo, los procesos erosivos sobre un terreno.





# Formalización del concepto de Fractal

*La geometría fractal permite estudiar fenómenos irregulares que no pueden ser caracterizados con las teorías geométricas clásicas.*

**Invarianza a cambios de escala.** Misma estructura (determinista o estadística) a cualquier escala.

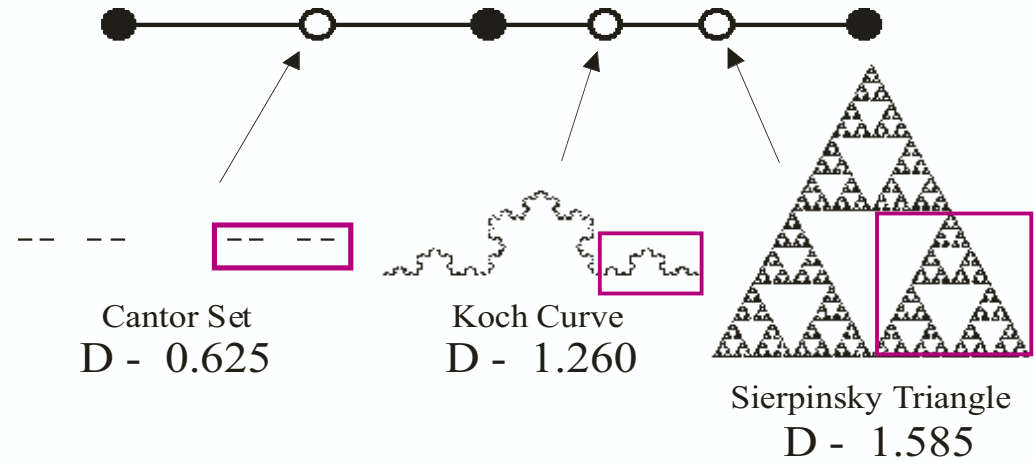
**Leyes de potencia.**

$$m(l) = l^D$$

**Soluciones de EDs**

**Curvas y superficies de interpolación**

**Sistemas de Funciones Iteradas**



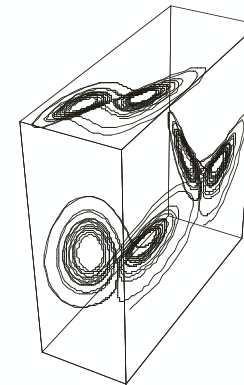
IFS



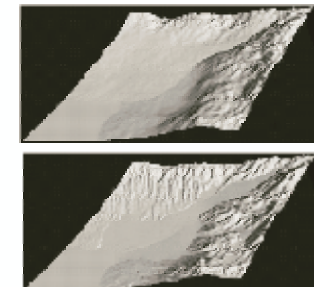
L-sistemas



Atractores extraños



Paisajes fractales



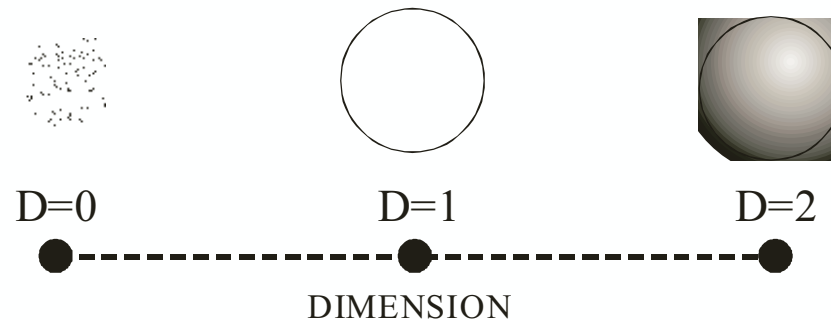
Curvas de interpolación fractal



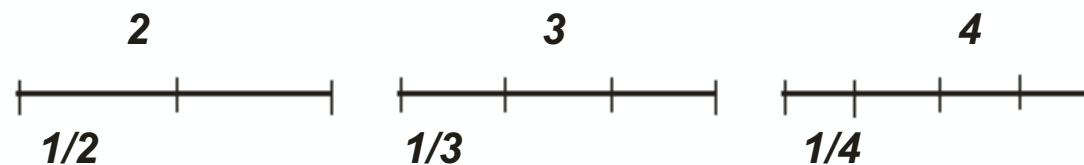
# La dimensión fractal

El desarrollo de la geometría fractal ha permitido obtener parámetros cuantitativos para definir el “**grado de irregularidad**” de un determinado objeto. Uno de los parámetros más representativos es el de **dimensión fractal**, una generalización de la dimensión euclídea para objetos autosemejantes.

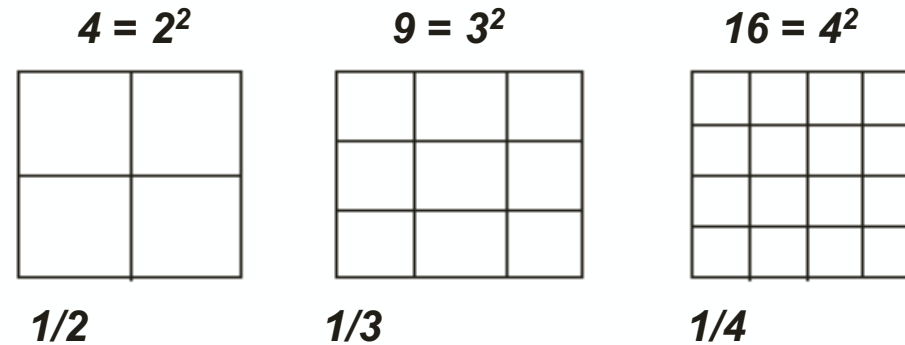
El concepto de **dimensión euclídea** asigna un número natural a los distintos objetos geométricos que pueden definirse en un espacio dado. Este concepto de dimensión tiene diversas interpretaciones intuitivas como, por ejemplo, el número de **parámetros** que son necesarios para definir el objeto.



Si partimos de un segmento de **longitud 1**, y lo partimos en segmentos de longitud  $L$  obtendremos  $N(L)$  partes, de manera que  $N(L) \cdot L^1 = 1$ ,



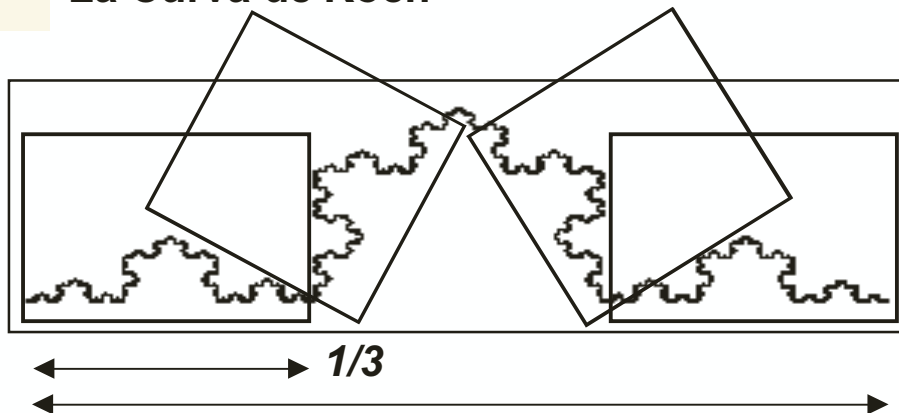
Si el objeto inicial es un cuadrado de **superficie 1**, y lo comparamos con unidades cuadradas, cuyo lado tenga de longitud  $L$ , el número de unidades que es necesario para recubrirlo  $N(L)$ , cumple  $N(L) \cdot L^2 = 1$ .



Podemos generalizar que la dimensión de un objeto geométrico es el número  $D$  que cumple:

$$N(L) \cdot L^D = 1 \iff D = \frac{\log N(L)}{\log(1/L)}$$

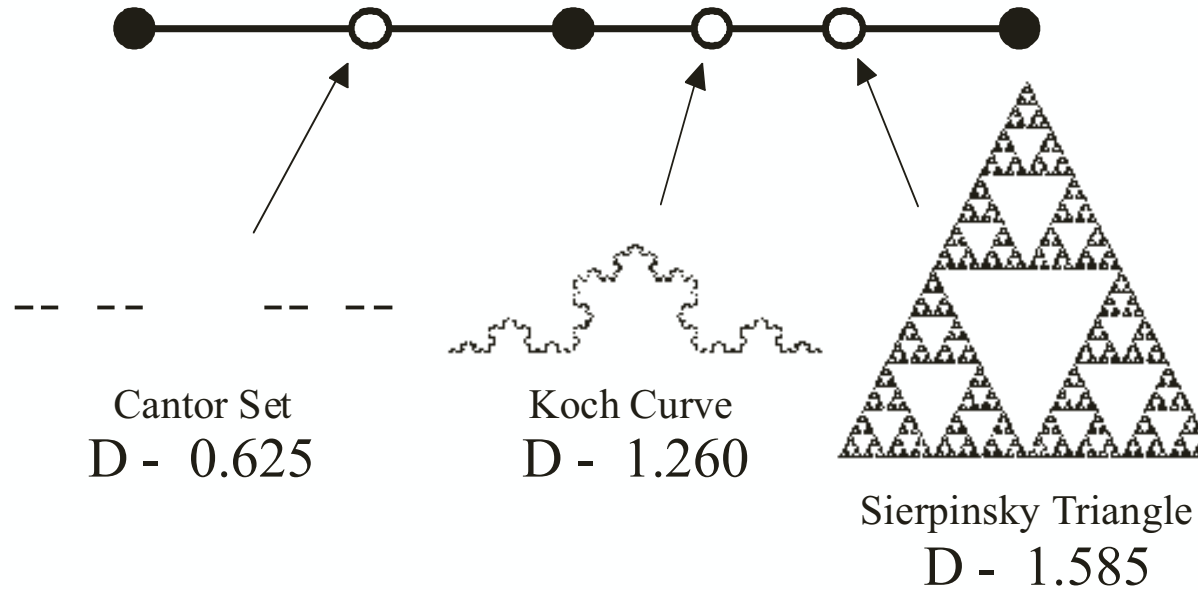
### La Curva de Koch



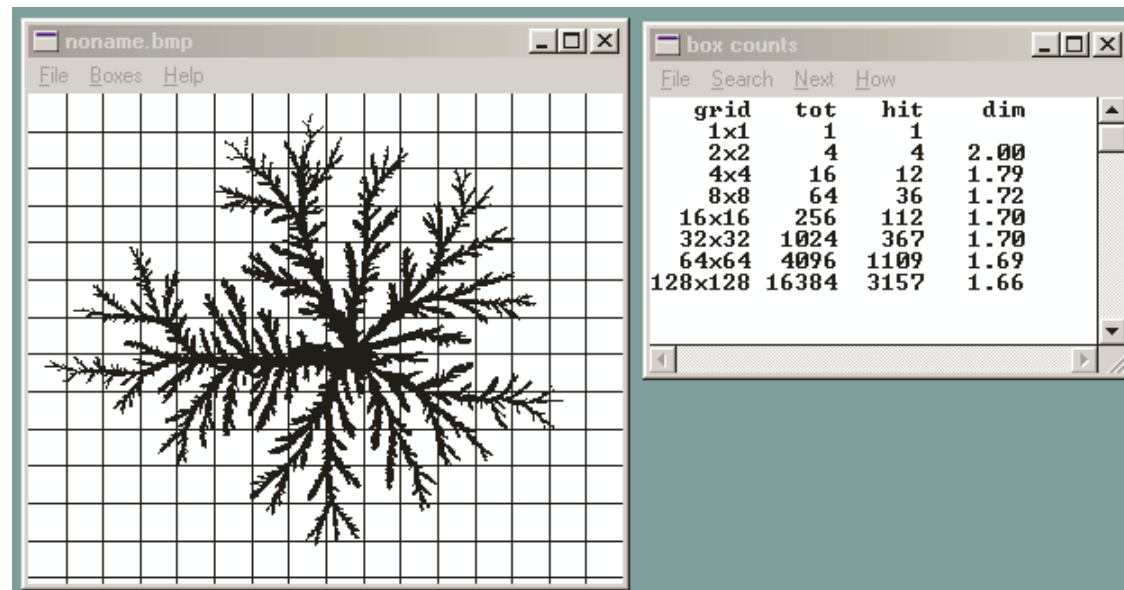
*Al reducir la escala de la curva de Koch a 1/3, nos encontramos con que se descompone en 4 partes.*

$$D = \log 4 / \log 3 = 1.2618\dots$$

**Dimensión Fractal !!!**



Basándose en estas ideas, existen algoritmos eficientes para el cálculo de la dimensión fractal de imágenes fractales obtenidas de cualquier estudio (urbanismo, biología, botánica, etc.).

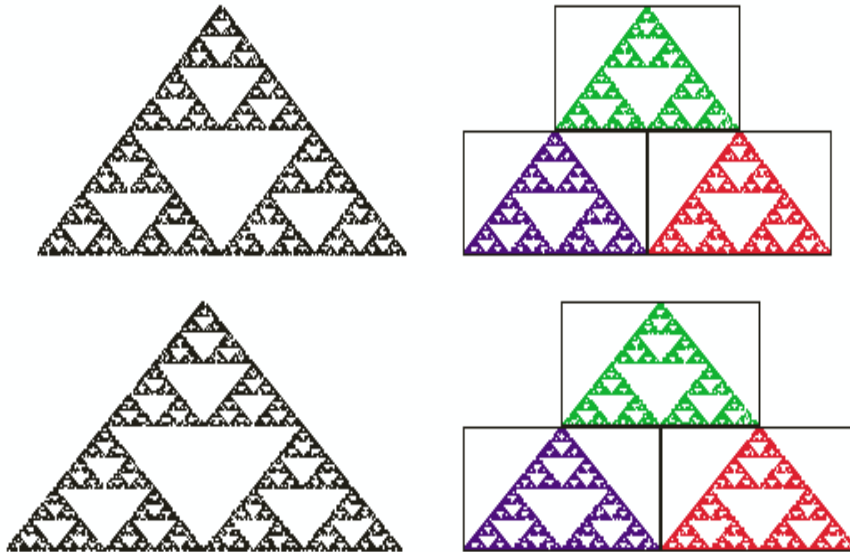
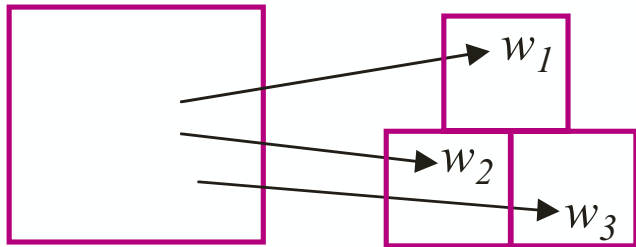


# Sistemas de Funciones Iteradas (SFI)

Un SFI consiste en  $N$  aplicaciones

$$\{w_1, \dots, w_N\}$$

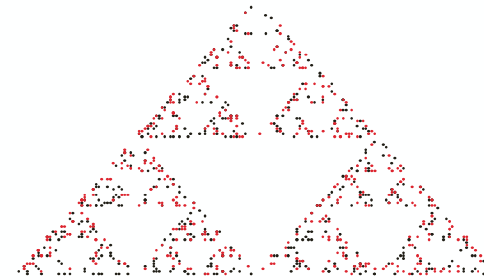
$$A = \bigcup_{i=1}^N w_i(A) \quad \text{“atractor”}$$



**IFS -> atractor (rendering)**

$$\{x_n\}_{n=1}^{\infty}, \quad x_{n+1} = w_{\sigma_n}(x_n)$$

$$P(\sigma_n = i) = p_i, \quad i = 1, \dots, N$$



**atractor -> IFS**

*El problema inverso de los sistemas de funciones iteradas (hallar las transformaciones que corresponden a un atractor dado) es un problema mucho más complejo que todavía no tiene solución.*



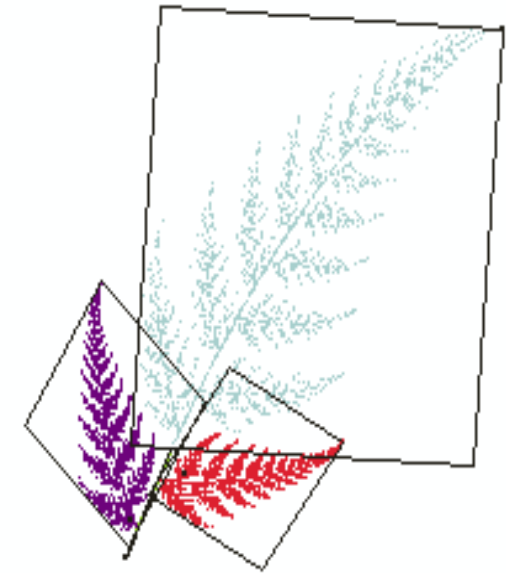
## El helecho de Barnsley

$$W_1 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0.81 & 0.07 \\ -0.04 & 0.84 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.12 \\ 0.195 \end{bmatrix}$$

$$W_2 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0.18 & -0.25 \\ 0.27 & 0.23 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.12 \\ 0.02 \end{bmatrix}$$

$$W_3 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0.19 & 0.275 \\ 0.238 & 0.14 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.16 \\ 0.12 \end{bmatrix}$$

$$W_4 \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0.0235 & 0.087 \\ 0.045 & 0.1666 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0.11 \\ 0 \end{bmatrix}$$



# Fractal Image Compression

Lenna 256 colors (gray scale)

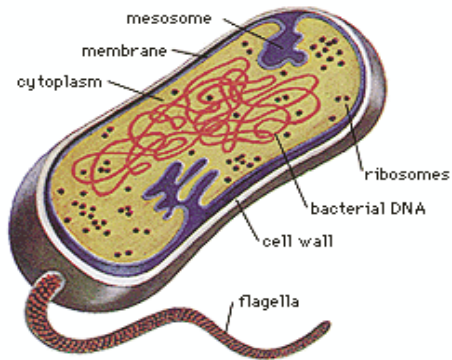
**BMP**  
66330 bytes

**Fractal**  
24619 bytes

**Fractal**  
2852 bytes



# Estructura de las Secuencias de ADN



Escherichia Coli  
176195 bases

gccattacct  
ctggtctgga  
agtagtctgg  
accagacgc  
cgaccagtg  
gagcaactat  
ttcttcgaga  
acctgttcaa  
gtatgagtgg

Secuencia  
Simbólica

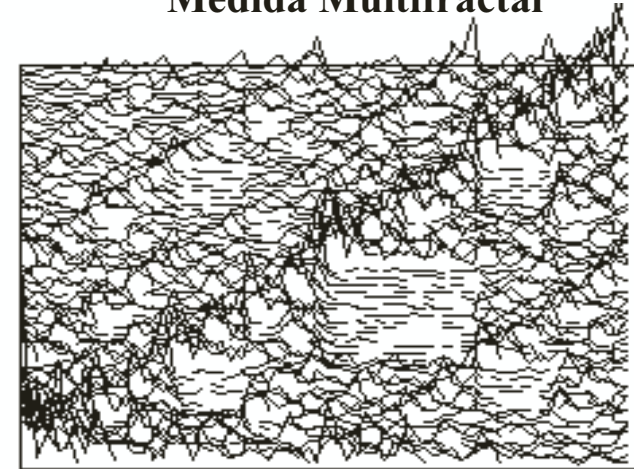
Alfabeto con 4 símbolos  $\{1, \dots, 4\}$



SFI con 4 transformaciones

$\{w_1, w_2, w_3, w_4\}$

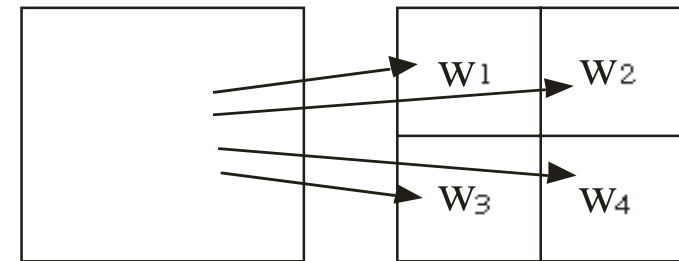
## Medida Multifractal



$$\{x_n\}_{n=1}^N,$$

$$x_{n+1} = w_{s_n}(x_n)$$

11	21	12	22
31	41	32	42
13	23	14	24
33	43	34	44



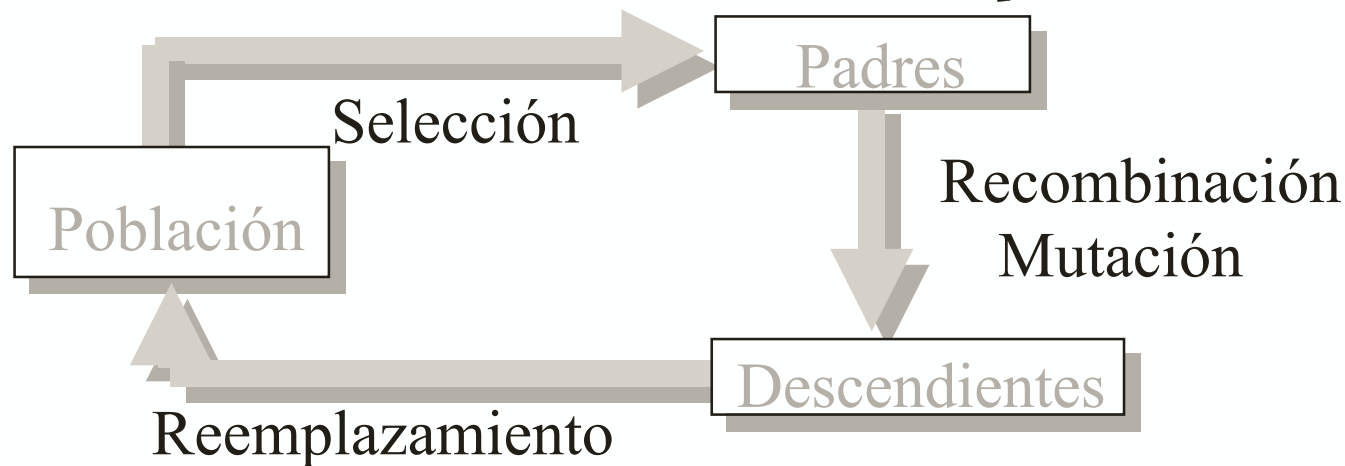


# Solución del Problema Inverso con Alg. Evolutivos

Las ideas básicas de la teoría de la evolución:

- **cruzamiento y mutación** de individuos y
- **selección** de los mejor adaptados

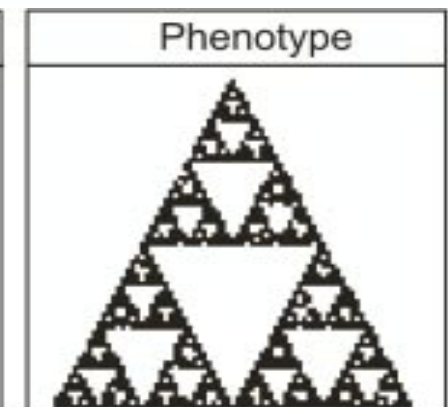
“Algoritmos evolutivos”.



**Una codificación de la población** del problema, ya sea binaria, o mediante números reales.

6 parámetros que componen cada una de sus transformaciones.

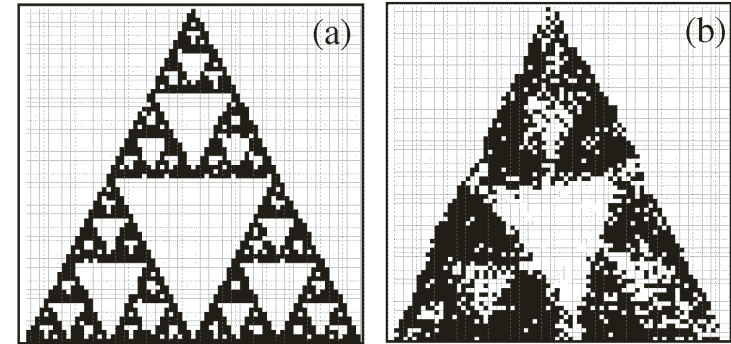
Genotype	
$ifs = (R^2; t_1, t_2, \dots, t_n)$	
$t_1$	$t_n$
$\dots$ $\theta_{i1}$ $\theta_{i2}$ $r_{i1}$ $r_{i2}$ $b_{i1}$ $b_{i2}$ $\dots$	
$\dots$ 01...001...001...001...0 01...001...0 $\dots$	
bit sequence	





**Una función de fitness (adaptación)** que cuantifique la optimalidad de cada elemento de la población.

El parecido (distancia de Hamming) entre la imagen buscada y el IFS.



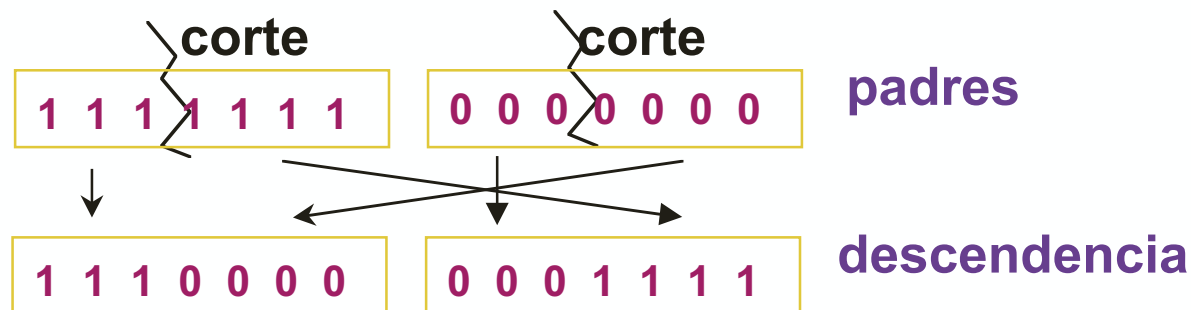
**Operadores genéticos de cruzamiento y mutación**, mediante los cuales se introducirá diversidad en la población.

La **mutación** se lleva a cabo variando aleatoriamente el valor de algún bit, o introduciendo un valor aleatorio en los números reales.

antes 1 1 1 1 1 1 1  
 después 1 1 1 0 1 1 1

$$x'_i = x_i + N(0, \sigma_i)$$

El **cruzamiento** trata de combinar elementos de la población para combinar las mejores características.



# Algoritmo Genético Híbrido

*La complejidad de los problemas hace que, en la práctica, pueda ser necesario utilizar algún tipo de estrategia híbrida para resolver un problema.*

*Primero calculamos transformaciones autosemejantes de la imagen dada.*

*Después combinamos las más adaptadas para iniciar un algoritmo genético.*

