



Control de Procesos – Ayudantía 00c Extra de Matlab (Problemas Resueltos)

Ejercicio R1. El juego del caos. Asumamos que tenemos tres puntos en los vértices de un triángulo equilátero, $P_1 = (0, 0)$, $P_2 = (0.5, \sqrt{2}/2)$, $P_3 = (1, 0)$. Generar otro conjunto de puntos $p_i = (x_i, y_i)$ tal que $p_1 = (0, 0)$ y p_{i+1} es el punto medio entre p_i y P_1 con un 33% de probabilidad, o el punto medio entre p_i y P_2 con un 33% de probabilidad, o el punto medio entre p_i y P_3 con un 33% de probabilidad. Escribir un programa que genere los puntos p_i para $i = 1, \dots, n$. Por ejemplo, se ilustran gráficos generados para $n = 100$ y $n = 10000$. (Tip: la función `rand` puede ayudar)

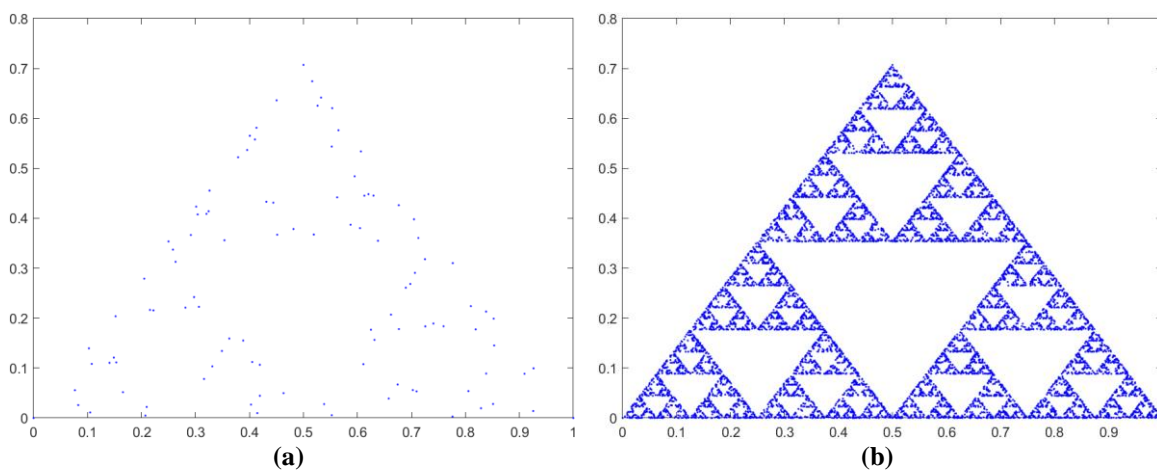


Figura 1. Triángulo de Sierpinski generado con puntos aleatorios para a) $n = 100$ puntos, b) $n = 10000$.

Nota: Más sobre el triángulo de Sierpinski en [wiki](https://es.wikipedia.org/wiki/Tri%C3%A1ngulo_de_Sierpinski).

Solución:

Este problema involucra conceptos de probabilidad y geometría. Para programar esta visualización deberemos seguir los siguientes pasos:

- Declarar los puntos iniciales.
- Crear un ciclo donde se vayan generando los siguientes puntos.
- Los puntos se generan con distancias medias entre el punto de la iteración anterior (o inicial p_1) y alguno de los que definen el triángulo equilátero inicial, ¿cuál? Elegir según las probabilidades dadas. Aquí será necesario el uso de condicionales `if-else` para realizar las comparaciones.
- Finalmente, podemos desplegar cada punto en cada iteración, o almacenarlos en un arreglo y desplegarlos al terminar el ciclo, lo primero tomará más tiempo, pero podremos seguir la actualización del fractal, esto es lo que haremos en este ejemplo.



Veamos la solución:

```
clear; clc; close

% El juego del caos, el triángulo de Sierpinski
clear; clc; close all;

% Primeros puntos que definen el triángulo equilátero
Px      = [0 0.5 1]; % Coordenadas x
Py      = [0 sqrt(2)/2 0]; % Coordenadas y

figure
plot(Px,Py, '.b', 'MarkerSize',3);
axis off; % borramos los ejes para visualizar solamente el fractal
set(gcf, 'color', 'w'); % cambiamos el color de fondo de la figura actual

Npoints = 10000; % número de puntos a dibujar
p        = [0 0]; % punto inicial pi = (xi,yi)
hold on % escribir este comando acá, hará que toda gráfica dentro del
        % ciclo se guarde en la misma figura

for k = 1:Npoints
    prob = rand;% números aleatorios entre 0 y 1 con distribución normal

    % Actualización de los puntos según las probabilidades dadas
    if prob < 0.33
        p(1) = (Px(1)+p(1))/2;
        p(2) = (Py(1)+p(2))/2;

    elseif prob < 0.66
        p(1) = (Px(2)+p(1))/2;
        p(2) = (Py(2)+p(2))/2;

    else
        p(1) = (Px(3)+p(1))/2;
        p(2) = (Py(3)+p(2))/2;

    end

    plot(p(1),p(2), '.b', 'MarkerSize',3);
    drawnow
end
```