1 Agregación limitada por difusión

Simule un proceso de agregación limitada por difusión en una red cuadrada con condiciones de borde periódicas. Indicativamente[1], use una red de 1024×1024, con una semilla quieta en el centro y del orden de 2×10^4 caminantes aleatorios que parten de posiciones distribuidas uniformemente al azar. Considere que los caminantes quedan pegados a la estructura por contactos a lo largo de los enlaces de la red y a 45°.

Haga un gráfico de la estructura dendrítica, coloreando las partículas de acuerdo al orden en que se pegaron.

Usando círculos de radio r con centro en la semilla, calcule la masa M contenida en el círculo en función del radio. Identifique en el gráfico log-log de $M(r)$ los tres regímenes, correspondientes a la escala de la red, la escala del agregado, y la región invariante de escala. Calcule la dimensión fractal de la estructura dendrítica. Observe que, para obtener un resultado sensato, los radios tienen que abarcar al menos un orden de magnitud en la región central.

Implemente el método de *box-counting* para cubrir el agregado. Grafique la cantidad de cajas, N, necesaria para cubrir la estructura, en función de la resolución r ($r = 1/d$ si d es el tamaño lineal de las cajitas; una posible elección de d son potencias de 2, usando cajas el doble de grandes en cada escala). En el gráfico log-log de $N(r)$ identifique nuevamente los tres regímenes y calcule la dimensión fractal en la región invariante. Vale la misma observación acerca del rango de magnitudes que tiene que abarcar r para obtener un buen resultado.

Si sus programas corren en un tiempo razonable, calcule la dimensión *box-counting* para un ensemble de agregados del mismo tamaño.

2 Use el programa que escribió para el sistema DLA para explorar el efecto de cambiar el número de semillas y partículas iniciales. ¿A qué densidad (número de partículas sobre número de sitios de la grilla) obtiene un agregado monolítico?

3 Haga crecer un agregado a partir de semillas ubicadas todo a lo largo de uno de los lados de la grilla. ¿Qué tipo de agregado se obtiene?

[1] Tarda unos 15 minutos en mi vieja notebook i5 1.8GHz de 3a generación, en Fortran escrito así nomás. Si quiere hacerlo más grande, o probar con varios tamaños, adelante. Ojo, que en Python o Mathematica es MUCHO más lento; recomendamos que usen un lenguaje compilado para esta simulación.

4 Modifique la regla de agregación, haciendo que los caminantes se peguen sólo si tienen el agregado en alguno de los cuatro primeros vecinos (es decir, sólo N, S, E, W, pero no a 45 grados). Use nuevamente una semilla central. ¿Qué cambios se observan en el agregado? Inspecciónelo visualmente y, si su programa es rápido, haga un agregado grande y calcule su dimensión fractal.

5 Este problema es todo un desafío, pero es muy interesante y, si tiene tiempo, le recomiendo que lo haga. No es difícil de programar.

Otro sistema que produce agregados interesantes es el de **percolación**. Dada una red de sitios "vacíos" (ceros) y un número $p \in (0, 1)$, se "ocupa" cada sitio (se pone en 1) con probabilidad p. El algoritmo es así de sencillo. Prográmelo para una red 2D de 512×512 o similar.

Observe el tipo de estuctura que se forma con $p = 0.25$, $p = 0.5$ y $p = 0.75$. Para las p menores, se ven una cantidad de agregados (*clusters*) disconexos, de distintos tamaños. Cuanto mayor sea p, más lleno queda el sistema, y más grandes son algunos de los clusters. Existe un valor de p crítico, p_c, tal que si $p > p_c$, el cluster más grande tiene un tamaño comparable al de todo el sistema, y abarca la red de lado a lado. Se dice que *percola*[2].

Identificar el cluster más grande no es trivial. Un método fácil de entender e implementar (si bien no súper eficiente) es el siguiente. Elija un sitio ocupado e imagine que lo pinta de rojo (ponele). A continuación, pinte de rojo todos sus primeros vecinos ocupados (en un vecindario de 4 vecinos, NSEO), y luego los vecinos de los vecinos, y así sucesivamente, hasta que no haya más vecinos ocupados, y guarde el tamaño. Luego repita con otro nodo ocupado, todavía no pintado, y mida el tamaño del nuevo cluster. Continúe haciendo esto hasta que no haya más sitios coupados sin pintar. Así habrá calculado el tamaño de todos los clusters. ¡Este algoritmo es mucho más complicado que el que usó para genera el sistema!

Calcule el tamaño del mayor cluster para valores de p entre 0 y 1. Para este sistema, $p_c \approx 0.59$, así que le conviene apretar los valores de p alrededor de p_c, y espaciarlos más lejos, para analizar el fenómeno de percolación. Grafique el tamaño del mayor cluster en función de p (pruebe distintas escalas, lineales y logarítmicas, *a piacere*). Observe que la cantidad de sitios ocupados depende de p, así que mejor le conviene *normalizar* el tamaño de los clusters, con la cantidad de sitios ocupados para cada valor de p. Grafique este tamaño normalizado y observe la transición de percolación.

[2] Si los elementos fuesen conductores, en tal caso la red entera conduce la electricidad de un lado al otro.

Ya que calculó todos los tamaños de cluster, haga un histograma normalizado que aproxime la función de distribución de tamaños de cluster. Hágalo para $p = 0.3$, $p = 0.59 \approx p_c$ y $p = 0.7$. Grafíquelas juntas en log-log, y observe cómo la del punto crítico obedece a una ley de potencia (una recta en log-log, mejor definida cuánto más grande sea el sistema, pero con 512 tendría que dar bastante bien) mientras que las otras no. Calcule el valor del exponente crítico (la pendiente, en log-log, debería dar -1.85).

Esta ley de potencia refleja la existencia de una invariancia de escala en los clusters. Muy cerca de p_c, los clusters son esponjosos, con filamentos y cavidades de muchos tamaños, y son fractales si el sistema es infinito. En 3D, es parecida a la esponja de materia (oscura y normal) que forma las galaxias y cúmulos de galaxias llenando el universo. Grafique los clusters más grandes para $p = 0.58$, 0.59 y 0.69 para ver la diferencia.

¿Cuánto vale p_c para una grilla 1D? (de memoria, sin programar nada).