

Conectividad y percolación

Álvaro Monclús, David Castro, Fernando Guirao, Aleix Ortigosa

24 de octubre de 2024

Algoritmia 2024-2025 Q1



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Facultat d'Informàtica de Barcelona



Resumen

La **teoría de la percolación** es un área de investigación dentro de la física estadística y las matemáticas que estudia el comportamiento de sistemas complejos y redes bajo procesos de conectividad aleatoria. En este proyecto se estudia el clásico problema de percolación en grafos que representan rejillas cuadradas (*squared lattice*), en grafos aleatorios geométricos (RGG), y grafos Kagome. El principal objeto de estudio es el **umbral de percolación**, a partir del cual se produce una **transición de fase** geométrica principalmente relacionada con la conectividad y la aparición de un gran clúster en la red.

En este documento, se presenta una introducción a la percolación, seguida por una modelización implementada basada en la literatura existente. Se realizan experimentos que permiten estimar este umbral y estudiar los fenómenos que lo rodean.

Índice

1. Introducción	3
2. Percolación	3
2.1. Modelando el problema	3
2.2. Algunas definiciones	5
2.2.1. Definición del modelo	5
2.2.2. <i>Coupling</i> de los procesos de percolación	6
2.2.3. Transición de fase	7
2.3. ¿Qué sabemos?	7
2.3.1. Umbráles de percolación conocidos	7
2.3.2. Exponentes críticos	8
2.3.3. Geometría fractal	9
3. Simulación del proceso de percolación	9
3.1. Percolación por nodos y aristas	9
3.2. Pseudocódigo del proceso de percolación	10
3.3. Optimizaciones del proceso de percolación con <i>Uniform Coupling</i>	11
3.4. Coste asintótico del proceso de percolación	11
3.5. Visualización del proceso	12
4. Experimentación	12
4.1. Metodología	12
4.2. Rejillas cuadradas	15
4.3. Grafos geométricos aleatorios	17
4.4. Grafos Kagome	20
5. Conclusiones	25
6. Aplicaciones	28
A. Glosario	28

1. Introducción

Empieza el otoño y estás paseando por un parque donde hay un gran tablero de ajedrez en el suelo. En esta época del año, las hojas caen de los árboles con frecuencia. Observas que algunas casillas del tablero han quedado completamente cubiertas por las hojas, mientras que otras permanecen vacías, de forma aparentemente aleatoria. Te das cuenta de que podrías cruzar el tablero pasando solo por casillas cubiertas de hojas. Eso es, casi, lo que llamamos percolación. [1]

Antes de continuar explicando qué es la percolación con más detalle, describimos brevemente el enfoque de este proyecto. Se ha implementado la simulación del proceso de percolación en grafos, basándonos en ideas obtenidas de la literatura existente, como la clusterización eficiente usando el algoritmo de Hoshen-Kopelman [2], (aunque nosotros implementamos una identificación de clústers generalizada para cualquier grafo, no solo rejillas, como en el caso de Hoshen-Kopelman). Posteriormente, estimamos el umbral de percolación en rejillas cuadradas, grafos geométricos aleatorios (RGG) y grafos Kagome, analizando los fenómenos asociados al umbral. Por último, exploramos algunas aplicaciones de la percolación, como la propagación de epidemias y la conectividad en redes de comunicación.

2. Percolación

Retomando el ejemplo del tablero de ajedrez, nos damos cuenta de que el problema de la percolación no se limita a contar cuántas casillas están cubiertas, sino que trata de la conectividad espacial entre ellas. Entonces, la pregunta clave es: ¿Cuántas hojas son necesarias para permitir el paso a través del tablero?

Puesto que las hojas se distribuyen aleatoriamente, podemos definir una probabilidad de ocupación, p , que mide la probabilidad de que una casilla esté ocupada. Así, la pregunta se reformula: ¿Existe un valor umbral de p (denotado como p_c) en el cual se forme un clúster gigante que permita el cruce a través de una red infinita?

El término percolación fue utilizado por primera vez en 1957 en publicaciones de Broadbent y Hammersley, quienes sentaron las bases de este campo de investigación. Desde entonces, la teoría de la percolación ha sido ampliamente estudiada en física estadística y matemáticas [3].

2.1. Modelando el problema

En esta sección explicamos de forma intuitiva el proceso de percolación, mientras que en la sección 2.2 damos algunas definiciones más formales, que se entienden mejor después de tener una idea intuitiva del proceso.

Dado un grafo $G = (V, E)$ y $p \in [0, 1]$. Hay dos modelos de percolación: por aristas (*bond percolation*) y por nodos (*site percolation*). Nos centramos en *bond percolation* para la explicación del modelo de percolación.

Al principio de este proyecto, nos topamos con dos formas de representar el proceso de percolación. Por un lado la percolación “clásica”, y por otro lado la percolación que sigue un “acoplamiento uniforme” (traducción literal del término *uniform coupling* presente en la literatura). Ambas formas de llevar a cabo este proceso nos conducen a los mismos resultados, sin embargo, utilizar *uniform coupling* es beneficioso a la hora de experimentar y visualizar la percolación (más adelante vemos por qué), pero sobre todo y más importante, aporta ciertas

propiedades que son necesarias para realizar demostraciones que dan respuesta a cuestiones básicas sobre la percolación, aunque en este proyecto no se profundiza en el plano teórico de este campo.

Percolación clásica

Supongamos que estamos trabajando con grafos que representan rejillas o cuadrículas cuadradas (*squared lattice*), denominados \mathbb{L}^d , donde \mathbb{L} indica el número de nodos que tiene cada lado, y d indica la dimensión de la rejilla.

Cada arista del grafo puede ser eliminada de forma aleatoria. Si una arista se elimina, decimos que queda cerrada, mientras que si no se elimina decimos que queda abierta. Por ahora supongamos que cada arista tiene una probabilidad 50-50 de quedar abierta o cerrada. En otras palabras, p es $1/2$. Cabe destacar que la decisión de eliminar o no cada una de las aristas es independiente del resto, es decir, lo que ocurra en una parte de la red no tiene ninguna influencia sobre lo que ocurra en cualquier otra parte.

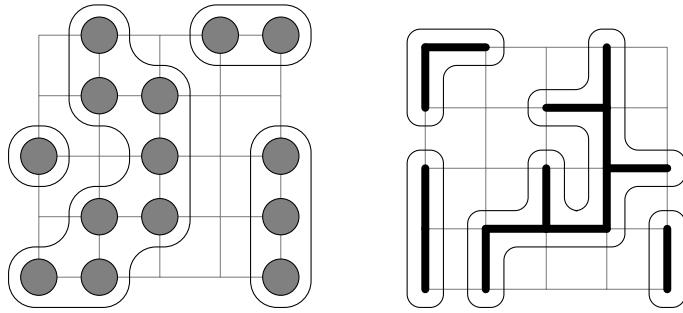


Figura 1: Rejillas \mathbb{L}^d tras aplicarles un proceso de percolación de nodos (izquierda) y un proceso de percolación de aristas (derecha). Las líneas delgadas definen los clústeres, y la línea sólida indica el clúster que percola.

Una vez se ha tratado cada arista, obtenemos un nuevo grafo, que es un subgrafo del original. Dado que la decisión tomada para cada arista es aleatoria, la estructura entera del nuevo grafo es aleatoria. Este proceso que hemos descrito es conocido como la **percolación de Bernoulli**, ya que cada arista de la rejilla actúa como una variable aleatoria de Bernoulli¹. Los ejemplos como los de la Figura 1 no son relevantes por sí solos, ya que lo que buscamos es estudiar todo el proceso aleatorio en general.

¿Qué ocurre si cambiamos p de modo que más o menos aristas queden abiertas? Repitiendo este proceso con distintos valores de p entre 0 y 1, veremos que para valores bajos de p tenemos menos aristas y más clústeres distintos y pequeños, mientras que para valores altos de p la rejilla estará poblada con más aristas y veremos menos clústeres y más grandes. p es ahora un parámetro de nuestro modelo, y tenemos que $\mathbb{P}_p[A] =$ la probabilidad de que un evento A ocurra en la percolación de Bernoulli con el parámetro p .

Percolación con *uniform coupling*

A través del proceso aleatorio descrito anteriormente de decidir con probabilidad p si una arista se queda o es eliminada del grafo. Podemos generar, dado G , para cada $p \in [0, 1]$ un subgrafo inducido $G' \subseteq G$. Sin embargo, para observar la aparición del clúster gigante que se mencionaba anteriormente en el umbral de percolación, deberíamos realizar este proceso

¹Una variable aleatoria de Bernoulli toma exactamente un valor entre dos diferentes

aleatorio repetidas veces alrededor de p_c . Seguramente seríamos capaces de observar la diferencia entre los grafos generados con $p < p_c$ y los grafos generados con $p > p_c$. No obstante, los clústers generados con $p < p_c$ no tendrían ninguna relación con los clústers generados con $p > p_c$. Si queremos observar el proceso de percolación en un grafo G' generado a partir de un valor bajo de p y ver cómo crecen los clústers de forma incremental hasta producirse la transición de fase, debemos introducir el concepto de *uniform coupling*.

Ahora, consideramos cerradas todas las aristas siendo a su vez todas ellas susceptibles de ser abiertas. Para cada arista generamos de manera uniforme e independiente un número aleatorio entre 0 y 1. Esto define un **nuevo proceso aleatorio**. Con los números asignados, podemos obtener un grafo diferente para cada valor de p . Para un determinado valor de p , una arista estará abierta si y sólo si su número asociado es $< p$.

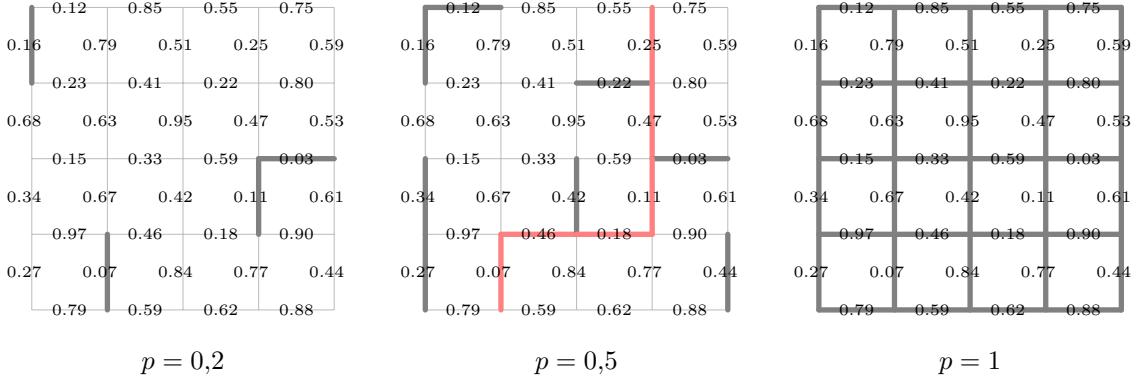


Figura 2: Proceso de percolación con *uniform coupling* en una red cuadrada.

Si fijamos un valor particular de p y consideramos sólo el grafo obtenido con ese parámetro, luego este proceso aleatorio es **idéntico** al de la percolación de Bernoulli con parámetro p .

$$\mathbb{P}[\text{un número aleatorio uniforme sea } < p] = p = \mathbb{P}_p[\text{una arista del grafo está abierta}]$$

Los grafos para diferentes valores de p ahora están “acoplados” de manera que ahora, dada una muestra de los números aleatorios que asignamos a las aristas, obtenemos una red diferente para cada valor de p con la propiedad de que la red correspondiente a un valor menor que p es siempre un subgrafo de la red correspondiente a un mayor valor de p .

2.2. Algunas definiciones

Habiendo introducido las ideas intuitivas sobre percolación y *uniform coupling*, podemos dar paso a una descripción más formal del proceso de percolación

2.2.1. Definición del modelo

Sea $G = (V, E)$ un grafo y $p \in [0, 1]$, consideramos el espacio de probabilidad² $\Omega = \{0, 1\}^E$, que representa cada una de las posibles configuraciones de las aristas del grafo. El subconjunto de elementos medibles de Ω (es decir, sucesos a los que se les puede dar una probabilidad) viene

²Es el conjunto de todos los posibles “estados” que un sistema puede tener.

definido por la σ -álgebra³ generada a partir de los cilindros⁴ de la forma

$$\{\omega(e) = x_e, \forall e \in A\},$$

donde A es un conjunto finito y $x_e \in \{0, 1\}$. Básicamente la expresión anterior representa un estado de la red, que denotaremos como $\eta_p \in \{0, 1\}^E$. La probabilidad asociada a esta medida es el producto de medida $\mathbb{P}_p(\cdot)$, esto significa que la probabilidad de una configuración η_p concreta es el producto de las probabilidades individuales de que cada arista esté abierta (p) o cerrada ($1 - p$).

En esta sección, seguimos asumiendo que $G = \mathbb{L}^d$ i. e. la rejilla Euclídea d -dimensional.

Escribimos $x \leftrightarrow y$ si existe un camino de aristas de x a y . Escribimos

$$\mathcal{C}(x) = \{y : y \leftrightarrow x\}$$

para indicar el clúster de x . También escribimos $x \leftrightarrow \infty$ si $|\mathcal{C}(x)| = \infty$, es decir, si x pertenece a un clúster infinito. Debido a la invariancia por traslación⁵ queda claro que $|\mathcal{C}(x)|$ tiene la misma distribución que $|\mathcal{C}(0)|$ para toda x . Ahora podemos hacernos las siguientes preguntas:

- ¿Existe $p \in (0, 1)$ tal que $\mathbb{P}_p(|\mathcal{C}(0)| = \infty) > 0$? (¿Puede existir un clúster infinito?)
- ¿Es la probabilidad $\mathbb{P}_p(|\mathcal{C}(0)| = \infty)$ monótona en p ? (¿La probabilidad de encontrar un clúster infinito es siempre creciente (o al menos no decrece) conforme aumentamos p ?)
- En el caso de que $\mathbb{P}_p(|\mathcal{C}(0)| = \infty) > 0$, ¿cuántos clústeres infinitos existen?

Para responder a las preguntas anteriores y más, se introducen diversas herramientas. El primer recurso que se utiliza con frecuencia para estudiar la percolación es un acoplamiento de todos los procesos de percolación, i. e. *uniform coupling* mencionado en la sección 2.1.

Definición 1. El acoplamiento (*Coupling*) de dos medidas de probabilidad μ y ν , definidas sobre espacios de probabilidad presumiblemente distintos, es una variable aleatoria (X, Y) definida en un único espacio de probabilidad tal que la distribución marginal de X es μ ⁶ y la marginal de Y es ν . En otras palabras, X e Y ahora toman valores en un mismo espacio de probabilidad.

El acoplamiento uniforme es fundamental para las demostraciones en el campo teórico, aunque el objetivo de este proyecto no es profundizar en ello. Para cadenas de Markov, el acoplamiento es útil para entender cómo los sistemas convergen a un comportamiento estable o estacionario, pero en nuestro caso lo usamos para modelar cómo varían las configuraciones de percolación conforme cambiamos p .

2.2.2. *Coupling* de los procesos de percolación

Denotamos como $\eta_p(e)$ el estado (abierto o cerrado) en el que se encuentra e cuando la probabilidad es p .

³Una σ -álgebra es una colección de subconjuntos medibles de Ω , es decir, que se les puede asignar una probabilidad

⁴Los conjuntos cilíndricos son subconjuntos de Ω en los que el estado de algunas aristas específicas está fijado. Por ejemplo, decir “las aristas e_1, e_2 están abiertas es un cilindro, mientras que decir: “El número total de aristas es par” no lo es, ya que no estamos fijando el estado de ninguna arista.

⁵El término invariancia por traslación indica que, en una red homogénea, las propiedades estadísticas son las mismas en cualquier punto de la red

⁶Que μ represente la distribución de probabilidad de X , quiere decir que μ nos indica cada probabilidad de que se de cada configuración posible (que son los valores que puede tomar X)

Sea $(U(e))_{e \in \mathbb{L}^d}$ una familia de variables i.i.d.⁷ en $[0, 1]$ (los valores que se asignan a cada una de las aristas). Definimos

$$\eta_p(e) = \begin{cases} 1 & \text{si } U(e) \leq p \\ 0 & \text{en caso contrario.} \end{cases}$$

A partir del acoplamiento, vemos inmediatamente que si $p \leq q$, entonces $\eta_p(e) \leq \eta_q(e)$, lo cual demuestra el

Lema 1. La probabilidad $\theta(p) = \mathbb{P}_p(|C(0)| = \infty)$ es creciente como función de p .

2.2.3. Transición de fase

Ahora definimos

$$p_c(d) = \sup\{p \in [0, 1] : \theta(p) = 0\}$$

que indica que el umbral crítico $p_c(d)$ es el valor más grande (supremo) de p en el intervalo $[0, 1]$ tal que la probabilidad $\theta(p)$ de que haya un clúster infinito es cero. En otras palabras, $p_c(d)$ es el mayor valor de p para el cual no existe un clúster gigante en la red.

Valiéndonos de la monotonía de $\theta(p)$, obtenemos que

$$\theta(p) = \begin{cases} 0 & \text{si } p \leq p_c \\ > 0 & \text{si } p > p_c \end{cases}$$

Es decir, que cuando pasamos el umbral crítico, existe la probabilidad de que encontremos un clúster infinito.

A partir de las definiciones mostradas y otras muchas técnicas, está demostrado que el clúster infinito **existe** y es **único**.

2.3. ¿Qué sabemos?

Como hemos dicho, no vamos a profundizar en demostraciones teóricas. Sin embargo, habiendo dado nociones básicas sobre percolación y algunos formalismos, podemos brevemente sumarizar más de 50 años de investigación y mencionar las conclusiones más relevantes, ya que algunas de ellas se verán reflejadas en la fase de experimentación de este proyecto.

2.3.1. Umbrales de percolación conocidos

Siguiendo con la percolación por aristas, para $d = 2$ sabemos que $\theta(p)$ es una función continua y el umbral de percolación p_c es exactamente 0.5 como se demostró en [10].

⁷i.i.d. significa independientemente e idénticamente distribuidas

Lattice	Site Percolation p_c	Bond Percolation p_c
Triangular	0.5	0.3473
Square	0.5927	0.5
Honeycomb	0.6970	0.6527
Kagomé	0.6527	0.5244
3D Cubic	0.3116	0.2488

Cuadro 1: Algunos umbrales de percolación conocidos. Los valores en cursiva han sido teóricamente demostrados.

2.3.2. Exponentes críticos

En grafos de tipo rejilla, los exponentes críticos en percolación describen cómo varias propiedades de un sistema divergen o se anulan al acercarse al umbral crítico de percolación, p_c , siguiendo una ley de potencias⁸.

- β : Describe cómo crece la probabilidad de que un sitio pertenezca al clúster gigante justo después de p_c . A partir de p_c , esta probabilidad sigue

$$\theta(p) \sim (p - p_c)^\beta$$

- ν : Describe cómo la longitud de correlación ξ , es decir, la escala máxima en la que los clústeres están correlacionados, diverge cerca de p_c

$$\xi \sim (p - p_c)^{-\nu}$$

- γ : Está asociado con la divergencia del número medio de sitios en clústeres finitos en p_c . Satisface

$$S \sim (p - p_c)^{-\gamma}$$

Para sistemas finitos, las propiedades como $\theta(n)$ o el tamaño de los clústeres dependen tanto de p como del tamaño del sistema \mathbb{L} . Para obtener los exponentes críticos, se analizan las simulaciones ajustando expresiones como

$$\theta(n) = \mathbb{L}^{-\beta/\nu} F(\mathbb{L}^{1/\nu}(p - p_c))$$

Cuando los datos de diferentes tamaños de rejilla colapsan en una curva, se confirman los valores de los exponentes.

Los exponentes críticos se determinaron mediante estudios teóricos y simulaciones numéricas. Se observó que ciertos comportamientos de las propiedades de los clústeres en modelos de percolación seguían leyes de potencias cerca de p_c . Además, al aplicar técnicas como el análisis de *finite size scaling*, se pudo ajustar los datos experimentales y obtener valores numéricos precisos para estos exponentes. En la percolación clásica 2D, los valores de β , ν , y γ **son universales y no dependen del tipo de red** ni del sistema específico.

Esta propiedad implica que, aunque los detalles microscópicos del sistema (como la estructura de la red) varíen, los comportamientos macroscópicos cerca del umbral crítico son los mismos.

⁸Una ley de potencias indica que una cantidad física X sigue una relación del tipo $X \sim (p - p_c)^\beta$, donde β es el exponente crítico.

El sistema se comporta de manera similar a grandes escalas, y las relaciones de escalado que describen el comportamiento del sistema son universales para una clase amplia de modelos en la misma dimensión espacial, lo que incluye redes cuadradas, triangulares, etc.

2.3.3. Geometría fractal

Otro interesante hallazgo es que en el umbral de percolación, el clúster más grande, C_{max} , tiene una geometría fractal⁹. Esto significa que el número de nodos del clúster más grande, S_{max} , escala con el tamaño del sistema, L , según la relación $S_{max} \sim L^{d_f}$, donde d_f es la dimensión fractal del clúster. En el caso de las rejillas bidimensionales ($d = 2$), se ha demostrado que $d_f = \frac{91}{48}$ [12].

Dado que $S_{max} = P_{max}L^d$, se puede derivar una ley de escalado que relaciona d_f , β y ν

$$d_f = d - \frac{\beta}{\nu}$$

Exponente	Valor
β	0.138
γ	2.38
ν	1.33
d_f	1.89

Cuadro 2: Exponentes críticos conocidos para la percolación clásica en rejillas bidimensionales.

3. Simulación del proceso de percolación

En esta sección, describimos los algoritmos desarrollados para simular el proceso de percolación tanto por nodos como por aristas en un grafo $G = (V, E)$. Estos algoritmos comparten ciertas ideas con el algoritmo de *Hoshen-Kopelman* (HK) en cuanto a la *clusterización* de la red, como por ejemplo el uso de la estructura Union-Find, pero presentan optimizaciones y características específicas al estar orientados a simular el proceso completo de percolación con *uniform coupling*.

3.1. Percolación por nodos y aristas

El objetivo de nuestros algoritmos es simular el comportamiento completo de los procesos de percolación con *uniform coupling* incrementando la probabilidad de ocupación p . Para ello, en cada iteración subimos el valor de p según un *step*, y recalculamos los clusters conectados en la red. Este proceso nos permite estudiar la evolución de los clústeres conforme nos acercamos al umbral crítico p_c , en el cual aparece el *clúster gigante* C_{max} .

Nuestros algoritmos no se limitan a grafos de tipo rejilla \mathbb{L}^d , sino que pueden aplicarse a cualquier tipo de grafo G (como los RGG). A continuación, presentamos el pseudocódigo del algoritmo para cada proceso de percolación.

⁹Un fractal es un objeto geométrico cuya estructura se repite a diferentes escalas y cuya dimensión no es necesariamente un número entero.

3.2. Pseudocódigo del proceso de percolación

El pseudocódigo a continuación describe ambos procesos:

Algorithm 1 Percolación por Aristas

```

1: Inicializar estructura Union-Find, inicialmente cada nodo tiene su clúster
2: Inicializar  $U(e) \sim \text{Uniform}(0, 1)$  para cada arista  $e \in E$ 
3: Inicializar  $p = 0$  y  $p_{\text{step}}$ 
4: Conectar súper nodos al top y bottom de la red                               ▷ Sólo en rejillas
5: while  $p \leq 1$  do
6:   // Unimos los nodos conectados por cada arista activada en esta iteración de p:
7:   for cada arista  $e$  con  $p_{\text{previo}} < U(e) \leq p$  do
8:     Actualizar clústers  $e$ : Unir  $C(e.\text{first})$  y  $C(e.\text{second})$  en Union-Find
9:   end for
10:  if  $C(\text{superTop}) == C(\text{superBottom})$  then                                ▷ Sólo en rejillas
11:    Percolación detectada
12:  end if
13:  Actualizar  $p \leftarrow p + p_{\text{step}}$ 
14: end while

```

Algorithm 2 Percolación por Nodos

```

1: Inicializar estructura Union-Find, inicialmente cada nodo tiene su clúster
2: Inicializar  $U(v) \sim \text{Uniform}(0, 1)$  para cada nodo  $v \in V$ 
3: Inicializar  $p = 0$  y  $p_{\text{step}}$ 
4: Inicializar conjunto de nodos activos  $A$ 
5: Conectar súper nodos al top y bottom de la red                               ▷ Sólo en rejillas
6: while  $p \leq 1$  do
7:   for cada arista  $e$  tal que  $U(e.\text{first}) \leq p$  y  $U(e.\text{second}) \leq p$  do
8:     // Comprobamos que ambos nodos no estuviesen activos ya y por lo tanto evitemos
     rehacer una unión:
9:     if  $\neg(e.\text{first} \in A \wedge e.\text{second} \in A)$  then
10:       Actualizar clústers: Unir  $e.\text{first}$  y  $e.\text{second}$  en Union-Find
11:     end if
12:   end for
13:   if  $C(\text{superTop}) == C(\text{superBottom})$  then                                ▷ Sólo en rejillas
14:     Percolación detectada
15:   end if
16:   // Actualizamos los nodos que están activados, esta es la estructura que nos permite
     evitar rehacer uniones de nodos que ya estaban activos:
17:   for cada nodo  $v$  con  $p_{\text{previo}} < U(v) \leq p$  do
18:      $A \leftarrow v$ 
19:   end for
20:   Actualizar  $p \leftarrow p + p_{\text{step}}$ 
21: end while

```

En ambos procesos, los componentes conexos se actualizan de manera eficiente utilizando *Union-Find*, y se verifica la existencia de un camino entre los *supernodos* para detectar la aparición del clúster gigante.

3.3. Optimizaciones del proceso de percolación con *Uniform Coupling*

El principal aspecto que diferencia nuestros algoritmos es la simulación del proceso de percolación a través de la técnica de *uniform coupling*. Gracias a ello, a partir de las definiciones vistas en la sección 2.2.2, concretamente, que $\eta_p(e) \leq \eta_q(e)$, podemos optimizar la actualización de clústers en cada iteración y hacerla realmente incremental, sin repetir operaciones *union*.

Por cada incremento de p :

- **Percolación por aristas:** Se activan solo las aristas cuyo valor $U(e)$ esté en el rango $[p_{\text{previo}}, p]$, y se actualizan los clústers en la red mediante la unión de los nodos conectados por dichas aristas.
- **Percolación por nodos:** Se evita conectar los nodos que ya estuviesen conectados en la iteración anterior. Utilizamos una lista auxiliar para saber qué nodos estaban activos en la iteración anterior.

Además, nuestros algoritmos incluyen una característica adicional para optimizar la detección del clúster percolante en el caso de las rejillas (cuando se conecta el *top* y el *bottom* de la red): el uso de *supernodos* para verificar si se ha producido la percolación. Estos *supernodos* se conectan a las primeras y últimas filas de una red bidimensional $L \times L$ para detectar si existe un camino entre el *top* y el *bottom*, indicando así la aparición del *clúster gigante*. De este modo, evitamos comparar el clúster de cada elemento de la fila del *top* de la red con cada clúster de cada elemento de la fila del *bottom* de la red, que tendría coste $O(L^2)$. En lugar de eso, basta con realizar una operación *find* sobre el *superTop* y el *superBottom* y ver si tienen el mismo representante, lo que tiene coste $O(\alpha(L^2))$, donde α es la **función de Ackermann**, que crece de forma extremadamente lenta y hace que el coste sea casi “constante”. Usamos α en la función de coste ya que nuestra implementación de Union-Find utiliza las optimizaciones de **compresión de caminos** y **unión por rango**.

Bien es cierto que el inconveniente de esto es mantener una estructura Union-Find auxiliar en memoria, ya que si hacemos la conexión de los *supernodos* en la estructura Union-Find principal, el número de clústers, especialmente en las primeras iteraciones, se ve alterado. No obstante, sacrificar más memoria, aunque sea el doble, es en nuestro caso preferible a realizar una operación temporalmente más cara, por cada iteración de p , especialmente en redes grandes.

3.4. Coste asintótico del proceso de percolación

En esta sección hacemos un análisis (muy general) del coste temporal de un proceso de percolación completo, específicamente para \mathbb{L}^2 . Tanto por nodos como por aristas tenemos un coste asintótico que depende del número de nodos $|V|$ y del número de aristas $|E|$ del grafo. A continuación, detallamos los factores que afectan al coste total del proceso:

- **Generación de la configuración aleatoria:** Asignar un valor aleatorio $U(e) \in [0, 1]$ a cada arista (o nodo) tiene un coste de $O(|E|)$ en la percolación por aristas y $O(|V|)$ en la percolación por nodos.
- **Unión y búsqueda de componentes:** Cada operación de unión y búsqueda en la estructura *Union-Find* tiene un coste amortizado de $O(\alpha(|V|))$, donde α es la inversa de la función de Ackermann. Este coste es casi constante en la práctica.

- **Actualización incremental:** En cada iteración, incrementamos p y procesamos las aristas (o nodos) activados. Esto tiene un coste de $O(|E|)$ en la percolación por aristas y $O(|V|)$ en la percolación por nodos por iteración.
- **Detección de percolación:** Verificar si los supernodos (conectando *superTop* y *superBottom*) están conectados en cada iteración tiene un coste de $O(\alpha(|V|))$, optimizando el proceso respecto a un coste $O(|V|)$ por iteración.

Coste total del proceso completo

El coste total del proceso completo de percolación, que implica procesar todas las iteraciones de p hasta que alcanza 1, está dado dominado por las operaciones de *union* y *find*:

$$O(|E| \cdot \alpha(|V|)) \text{ para la percolación por aristas}$$

$$O(|V| \cdot \alpha(|V|)) \text{ para la percolación por nodos}$$

Es decir, de forma muy general, en la percolación por nodos haremos una operación *union* por cada nodo, mientras que en la percolación por aristas, haremos una operación *union* por cada arista.

3.5. Visualización del proceso

En esta sección se muestra la visualización de un proceso de percolación completo por aristas en \mathbb{L}^2 .

4. Experimentación

4.1. Metodología

En esta sección definimos lo necesario para la parte práctica de este proyecto.

Generación de grafos

El primer paso es usar las librerías de Python NetworkX[8] y Netket[9] para generar los tres modelos de grafo que estudiaremos, rejillas cuadradas, RGG, y rejillas Kagome. Usando estas librerías, hemos generado archivos dimacs¹⁰ de 5 tamaños distintos para cada grafo.

Simulación del proceso de percolación

Con los archivos generados para cada grafo, podemos ejecutar los algoritmos que simulan el proceso completo de percolación (explicados en la sección 3) para cada p entre 0 y 1, dado un *step*. El *step* utilizado en los experimentos es de 0.01, para tener una resolución suficiente que nos permita detectar con relativa precisión el umbral de percolación.

¹⁰Un archivo dimacs es una forma de almacenar un grafo regulado por caracteres donde una línea que consiste de: e v1 v2 representa una arista entre el vértice v1 y v2.

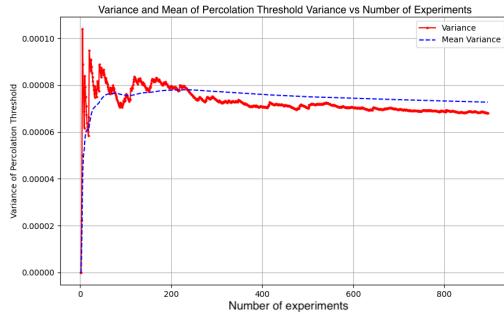
Cada proceso de percolación completo para una p concreta, nos devuelve el número de componentes conexos (N_{cc}), el tamaño del clúster más grande (S_{max}), y el tamaño normalizado del clúster más grande (N_{max}).

Para el caso de las rejillas, también obtenemos un valor de p crítico (p_c) que indica la iteración donde el clúster de mayor tamaño atraviesa la red de arriba a abajo, es decir, se conecta el *superTop* con el *superBottom*. En el caso de los RGG, esta p se determina cuando el grafo pasa a ser conexo.

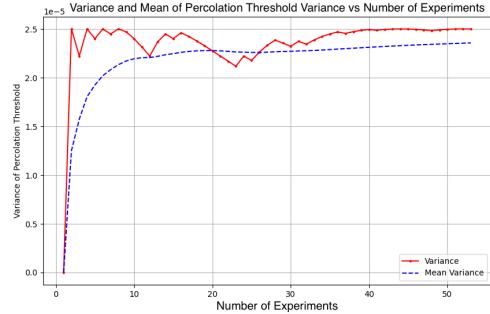
Número de experimentos

Dado que experimentamos sobre un proceso estocástico, realizaremos simulaciones de MonteCarlo. El método consiste en realizar un gran número de simulaciones para encontrar el número de simulaciones a partir del cual la varianza de p_c se estabiliza.

Concretamente, vamos haciendo simulaciones y actualizando la varianza de p_c , y cuando la variación promedio de la varianza en las últimas 15 simulaciones está por debajo de una $\epsilon = 5 \times 10^{-8}$ decimos que se ha estabilizado. Hemos probado esto con mallas $L = 200$ y $L = 1000$ simulando procesos de percolación por aristas , con grafos geométricos aleatorios y con mallas Kagome.

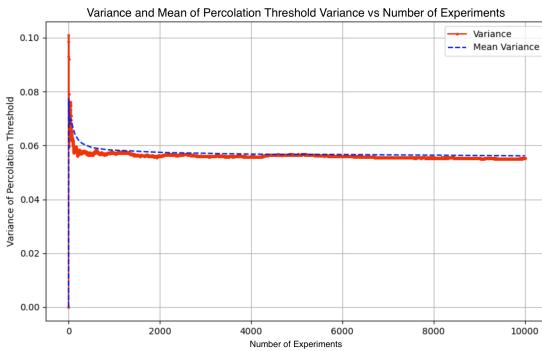


(a) Varianza en mallas de 200x200.

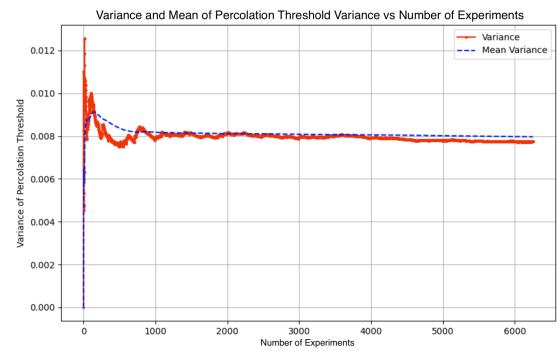


(b) Varianza en mallas de 1000x1000.

Figura 3: Análisis de varianza para diferentes tamaños de malla.



(a) Varianza en RGG.



(b) Varianza en grafos Kagome.

Figura 4: Análisis de varianza en RGG y grafos Kagome.

El hecho de que la varianza se estabilice antes en grafos grandes está relacionada con la ley de los grandes números y los efectos de las condiciones de frontera y la disminución de

las irregularidades locales. Bazant(2000) [6], señala que los clústeres más grandes crecen de manera logarítmica con el tamaño del sistema. Debido al mayor número de nodos y aristas, el sistema tiende a comportarse de manera más “promedio”, lo que reduce la variabilidad de los resultados entre simulaciones. Esto implica que el valor crítico p_c converge más rápidamente a su valor esperado, lo que lleva a que la varianza se estabilice más rápido. En contraste, en grafos pequeños, las fluctuaciones entre simulaciones son más significativas, y por lo tanto, la estabilización de la varianza tarda más en alcanzarse.

Además, podemos graficar también la distribución de p_c para tamaños considerables de muestra, y ver cierta relación con el teorema del límite central (TLC), ya que, a medida que aumentamos el tamaño de la muestra, la distribución de p_c se concentra en torno a un valor concreto de manera normal.

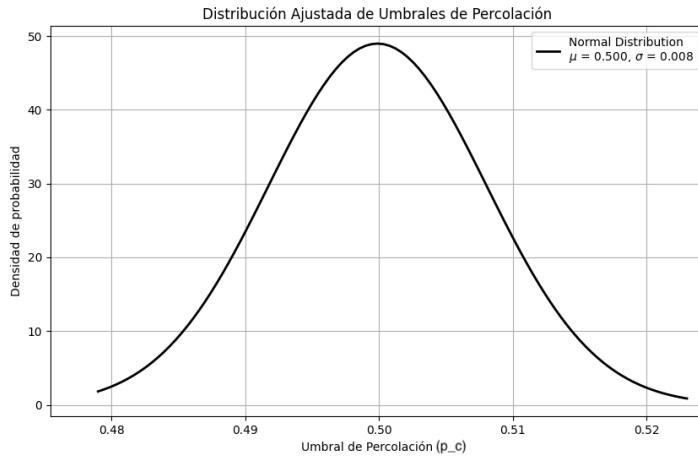


Figura 5: Distribución de p_c .

Finalmente, para mantener la simplicidad y estandarizar todos los experimentos realizados en este proyecto, decidimos usar un número de experimentos fijo, $N_{\text{exp}} = 1000$, para todas las configuraciones de grafos y tamaños de malla.

Para mantener la simpleza y reforzar la calidad de nuestros resultados, fijaremos el número de iteraciones en los grafos de mallas a **N=1000** para cada experimento. Fijaremos a **N=2000** iteraciones los RGG y fijaremos los Kagomé a **N=2000** ejecuciones para percolación por nodos y **N=4000** iteraciones para percolación por aristas.

Análisis de resultados

Con los resultados obtenidos en cada simulación N_{cc} , S_{\max} y N_{\max} por cada p , además de la p_c por cada simulación. Podemos calcular y graficar lo siguiente:

- N_{cc} Vs. p , donde por cada p graficamos el número de componentes conexos promedio.
- S_{\max} Vs. p , donde por cada p graficamos el tamaño del clúster más grande.
- N_{\max} Vs. p , donde por cada p graficamos el tamaño normalizado del clúster más grande.

Además, otro tipo de gráfico que podemos hacer, y quizás el más importante, es el de P_{\max} Vs. p . En este caso, graficamos para cada p la cantidad de veces que se ha producido

la percolación de forma **acumulada**. Es decir, si en 1000 experimentos, para $p = 0,4$ hemos detectado únicamente tres p_c inferiores a $p = 0,4$, entonces su correspondiente valor de P_{max} será $3/1000$.

4.2. Rejillas cuadradas

Los resultados obtenidos para la percolación por aristas en mallas cuadradas de diferentes tamaños ($L = 50, 100, 200, 500, 800, 1000$) muestran comportamientos consistentes con la teoría establecida. A continuación, se presenta un análisis detallado de los diferentes aspectos observados en las simulaciones.

Percolación por nodos

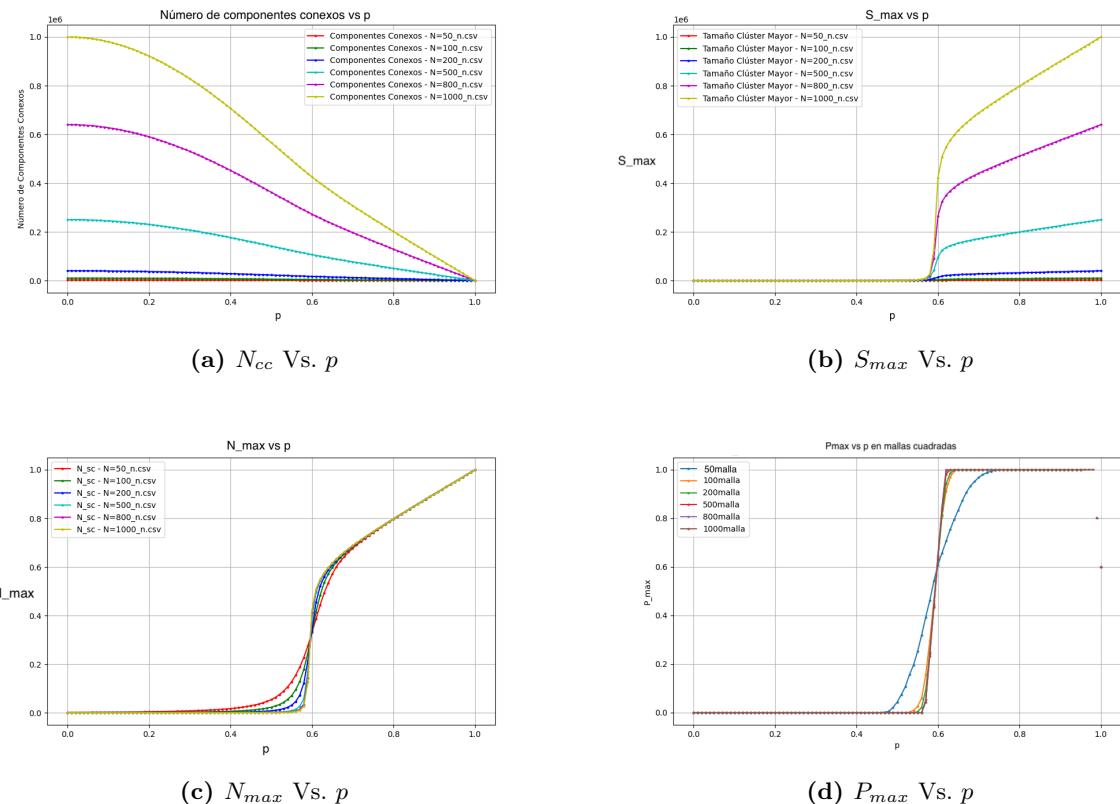


Figura 6: Resultados de la percolación por nodos en mallas cuadradas de distintos tamaños.

- **N_{cc} Vs. p :** En esta gráfica apreciamos cómo a medida que aumenta el valor de p , va decreciendo el número de componentes conexos. Mirando esta gáfica es difícil averiguar de forma exacta dónde ocurre la transición de fase, aunque sí se puede apreciar un cambio en la curvatura entre $p = 0,4$ y $p = 0,6$.
- **S_{max} Vs. p :** En esta gráfica se ve una clara transición de fase cerca del valor teórico mostrado por Akhunzhanov(2022) [11], en $p \approx 0,59$. Como hemos discutido en secciones previas, al ocurrir la transición de fase se genera un clúster gigante que recubre una gran parte del grafo.
- **N_{max} Vs. p :** Al normalizar el tamaño del clúster mayor C_{max} , vemos la influencia que tiene el tamaño de la rejilla en cuanto a la precisión del valor de p_c , siendo más preciso

con rejillas más grandes. El comportamiento observado sigue la forma funcional predicha por la teoría de scaling:

$$S_\infty \sim |p - p_c|^\beta \quad \text{para } p > p_c \quad (1)$$

- **P_{max} Vs. p :** De nuevo aparece la transición de fase en la p esperada. También vemos cómo para un tamaño menor de rejilla, la transición de fase ocurre en un rango mayor de valores. Este rango se reduce drásticamente al llegar a tamaños mayores de grafo.

Percolación por aristas

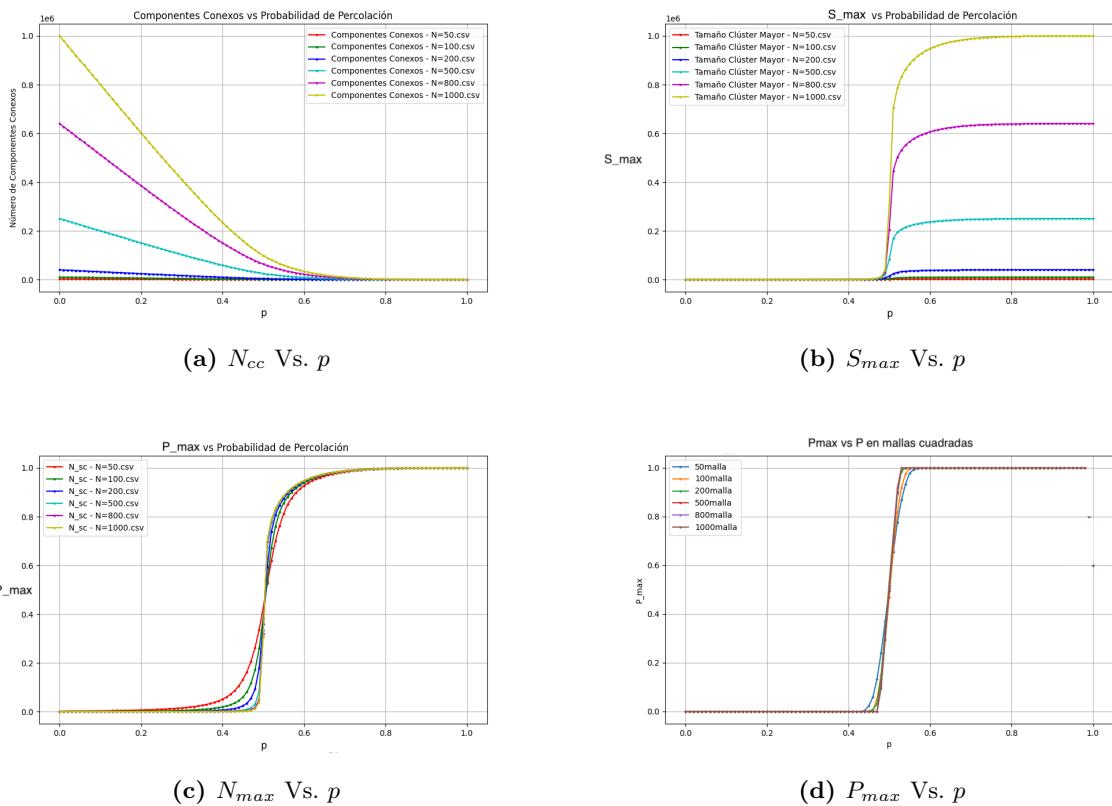


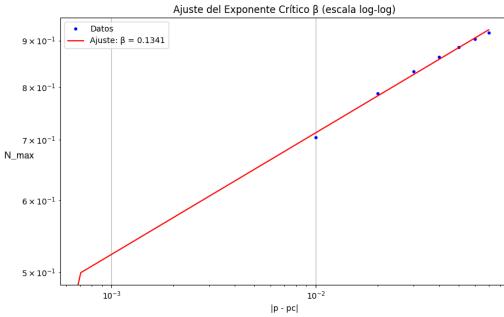
Figura 7: Resultados de la percolación por aristas en mallas cuadradas de distintos tamaños.

- **N_{cc} Vs. p :** En comparación con la percolación por nodos, aquí podemos apreciar una transición mucho más clara de los componentes conexos, de nuevo en el valor teórico demostrado en la malla como se muestra en Kesten(1980) [10].
- **S_{max} Vs. p :** De nuevo vemos unos resultados muy similares a aquellos de la percolación por nodos, con una clara diferencia en la rapidez de crecimiento del superclúster C_{max} .
- **N_{max} Vs. p :** Al igual que en la gráfica anterior, la gran diferencia ocurre en la rapidez de crecimiento del clúster. El comportamiento observado sigue la forma funcional predicha por la teoría de scaling:

$$S_\infty \sim |p - p_c|^\beta \quad \text{para } p > p_c \quad (2)$$

- **P_{max} Vs. p :** Podemos observar que la dispersión entre diferentes tamaños de rejillas es menor que al hacer percolación por nodos. De todos modos, sigue siendo claro que al tener rejillas mayores, los datos son más concretos.

Exponentes críticos



(a) Recta del exponente crítico β .

Figura 8: Cálculo exponente crítico β

En la Figura 8, se puede ver el cálculo del exponente crítico β que representa el crecimiento de N_{max} para los valores cercanos. Los exponentes críticos observados son consistentes con los valores universales esperados para la clase de universalidad de percolación 2D:

- $\nu = 4/3$ [17] para la longitud de correlación
- $\beta = 5/36$ [16] para el parámetro de orden

En nuestro caso, al calcular β , usando un ajuste por regresión log-log, obtenemos un valor de 0.1341 (Figura 8a), algo muy similar al valor teórico de 0.138.

En cuanto a los efectos de tamaño finito, la Figura 17c (N_{max} vs p) y la Figura 14c, evidencian claros efectos de tamaño finito al hacer percolación por aristas y por nodos, consistentes con la teoría de *finite-size scaling* [13].

- Se observa un redondeo característico de la transición cerca de p_c
- Las desviaciones del comportamiento de scaling son más pronunciadas para N pequeños
- Existe una convergencia gradual hacia el comportamiento termodinámico al aumentar N

En resumen, los resultados numéricos obtenidos muestran una excelente concordancia con las predicciones teóricas fundamentales [12]. En particular:

1. El valor del umbral crítico coincide con el valor exacto $p_c = 1/2$ y el aproximado de $p_c \approx 0.59$
2. Los exponentes críticos reproducen los valores esperados para la clase de percolación 2D
3. El comportamiento de *scaling* sigue las formas funcionales predichas por la teoría

4.3. Grafos geométricos aleatorios

Definiciones

Un grafo geométrico aleatorio (RGG, de *random generated graph*) es un tipo de grafo con aplicaciones en el mundo real, como la epidemiología o las representaciones de redes. Se modelan

de la siguiente forma:

- V se reparte siguiendo una distribución sobre el plano bidimensional PL para el cual:

$$PL = \{p = (x, y) \in \mathbb{R} \times \mathbb{R} \mid x \geq 0 \wedge x < 1 \wedge y \geq 0 \wedge y < 1\}$$

- E es el conjunto de aristas, en el que se forma una arista entre dos vértices si y solo si, la distancia euclídea entre ambos es menor al radio. Formalmente:

$$E = \{(a, b) \in V \times V \mid d_e(a, b) < R\} | E | = \left\{ \frac{n^2 \pi R^2}{2} \right\}$$

Cabe destacar que, por la definición de E , se podría obtener un grafo no conexo,(p.ej. un grafo con todos sus nodos conectados excepto uno que estará alejado del resto a una distancia mayor que R), por lo tanto para este tipo de grafos hay que asegurarse de que cada grafo generado sea conexo. Para ello, primero se tiene en cuenta una estimación para R donde, con alta probabilidad, el RGG será conexo[20]. Esta R crítica es:

$$R_c = \sqrt{\frac{\ln n \pm \mathcal{O}(1)}{\pi n}}$$

Para la generación, se distribuirán los nodos aleatoriamente y se empezará con una R_c con su término $\mathcal{O}(1)$ con un valor de 1. Si el grafo resultante no es conexo, se modifica el término y se repite el proceso de distribución de nodos. Dada la precisión demostrada de esta R_c , la obtención de un RGG conexo es rápida y eficaz.

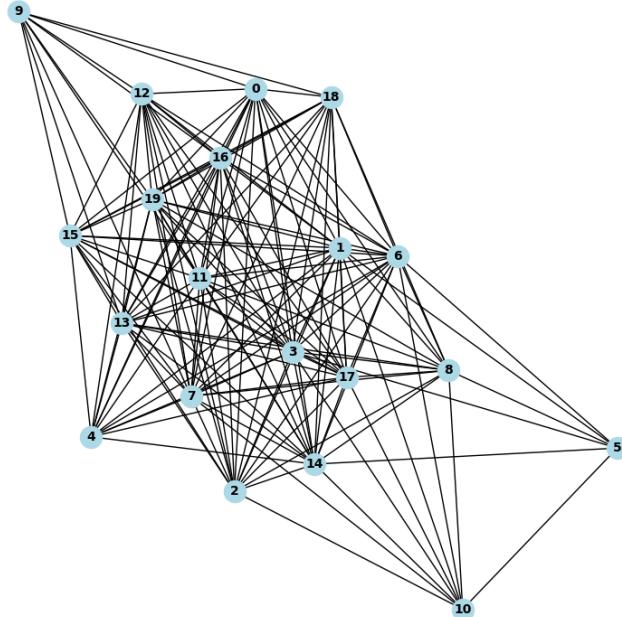


Figura 9: Representación de un RGG de 20 nodos conexo

Resultados y análisis

Para los RGG se ha realizado la percolación por aristas en grafos de 6 tamaños ($n = 50, 100, 200, 500, 800, 1000$). Con el fin de presentar unos resultados más concisos, el intervalo

de probabilidades ha sido $\Omega[0, 0, 2]$, ya que a partir de el último valor el porcentaje de grafos con una cantidad de componentes conexos considerable es negligible.

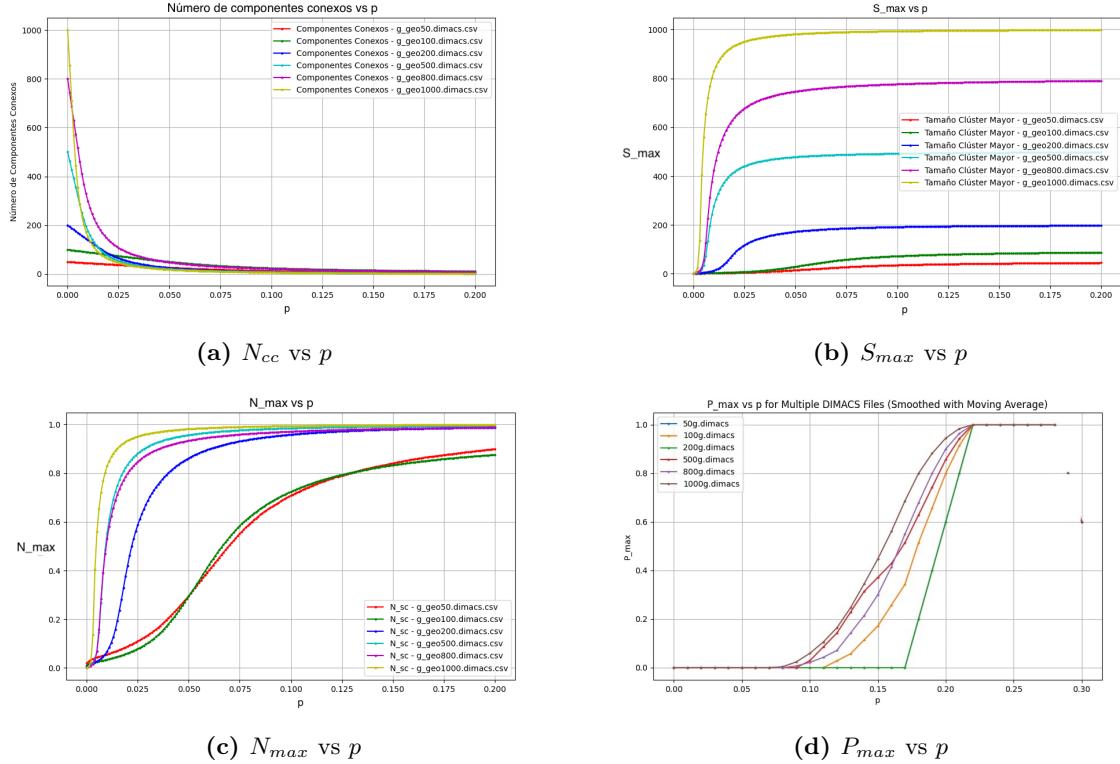


Figura 10: Resultados de la percolación por aristas en RGG

Umbral de percolación y transición de fase

Para los RGG, una estimación para su transición de fase es la p tal que el grafo es conexo. Para este experimento, el umbral está en ($p \approx 0,15$). Pese a parecer una probabilidad ínfima, por la estructura de los RGG hay cuantiosas alternativas para obtener un grafo conexo, causando entonces este comportamiento. Además, esta p se encuentra en el intervalo $[\frac{1}{\pi}, 0,25]$ que comprende valores encontrados en los diversos experimentos de Donelly(2004) [23].

Análisis de componentes conexos

El número de componentes conexos en función de la probabilidad de percolación p (Figura 11a) se desarrolla de la siguiente forma:

- En ($p \approx 0,01$) tenemos la reducción de clústeres más rápida.
- A partir de la p anterior, disminuye el ritmo al que se reducen las componentes, pero se sigue apreciando que la cantidad decrece.
- Cuando hemos llegado a ($p \approx 0,1$) ya se ve que la gran mayoría del grafo está conectada.

Comportamiento del clúster mayor C_{max}

La Figura 10b muestra el tamaño del clúster de mayor tamaño sobre el grafo percolado.

Destaca el hecho de que para las n grandes el tamaño de este clúster aumenta hasta los valores más altos sobre la ($p \approx 0,01$), ya que va acorde a la reducción de componentes conexos vista antes. Las diferencias entre la función para las n pequeñas y grandes se deben a la habitual densidad de este tipo de grafos[21]. En la Figura 10c, como se está tratando el tamaño del clúster respecto al número de nodos, se aprecia una menor diferencia para las n grandes y pequeñas, pese a seguir siendo muy notable.

Efecto del radio sobre p_c en RGG

En la Figura 10d, podemos ver el valor de p_c para diferente densidad de grafos, es decir, para distintos valores de R . Por la aleatoriedad subyacente de este tipo de grafos, no hay un consenso para una estimación de la p para la fase de transición. No obstante, suele haber un valor para un elemento ampliamente utilizado, este es el valor para R definido en la descripción del modelo del grafo. Es usado en varias investigaciones sobre percolación y conectividad de grafos, como puede ser [22]. Como aún teniendo esto en cuenta los resultados varían considerablemente (debido a la densidad de aristas y el tamaño del grafo sobretodo), nos centraremos en los resultados empíricos obtenidos con la experimentación.

En las siguientes figuras, podemos ver el efecto de R sobre los parámetros estudiados.

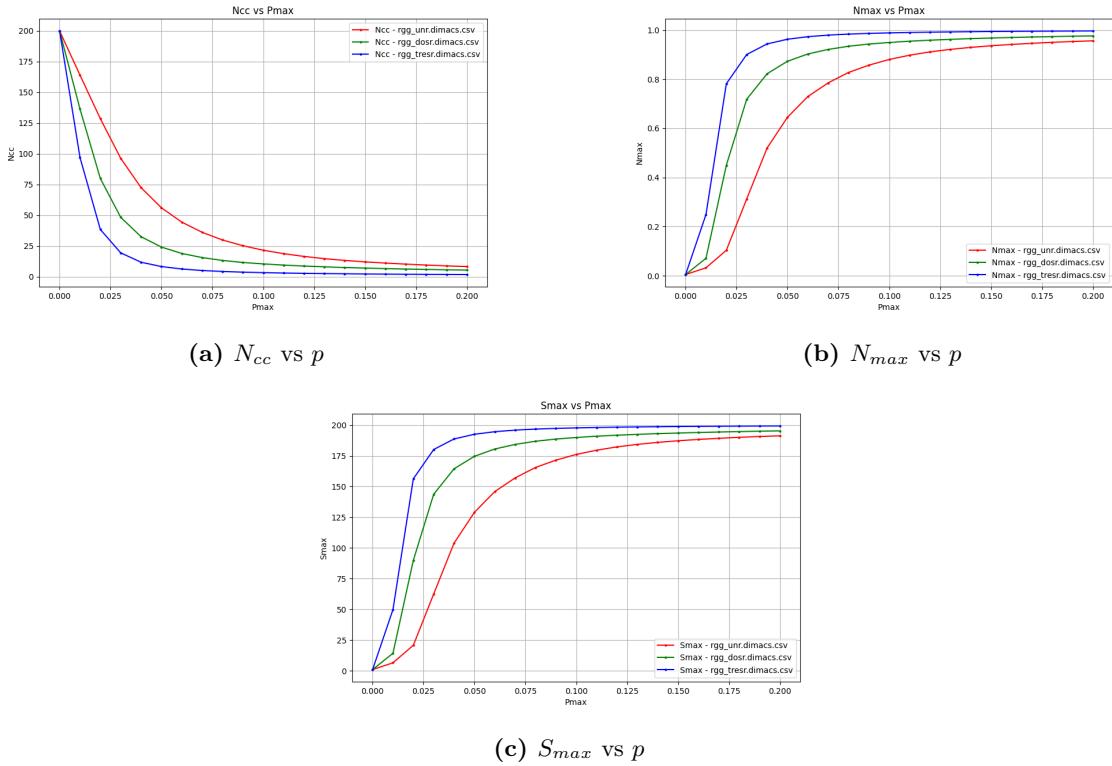


Figura 11: Efecto de R sobre los RGG

4.4. Grafos Kagome

Definiciones

Los grafos Kagome o mallas Kagome son unas malla donde cada vértice forma parte de un triángulo, hexágono, triángulo y hexágono, es decir es una malla (3, 6, 3, 6). Se puede pasar

de un grafo a triangular a *Kagome* añadiendo aristas para formar hexágonos y se puede pasar de un grafo *honeycomb* o hexagonal a kagome formando triángulos entre los nodos. Es un grafo estudiado en la percolación y está presente en algunos minerales como las Jarositas y la Herbertsmithita.

En la función que usamos para generar los grafos kagome, especificamos la altura y la anchura de la malla, y la cantidad de nodos del grafo es igual a $3 * \text{altura} * \text{anchura}$. Os mostramos ejemplos para $1 \times 1, 2 \times 2, 3 \times 3$ y 10×10 para que los podáis entender mejor, el 1×1 es una excepción ya que solo es un triángulo.

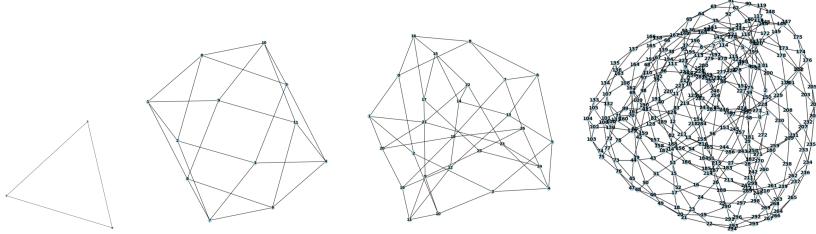


Figura 12: Ejemplo de grafos Kagome.

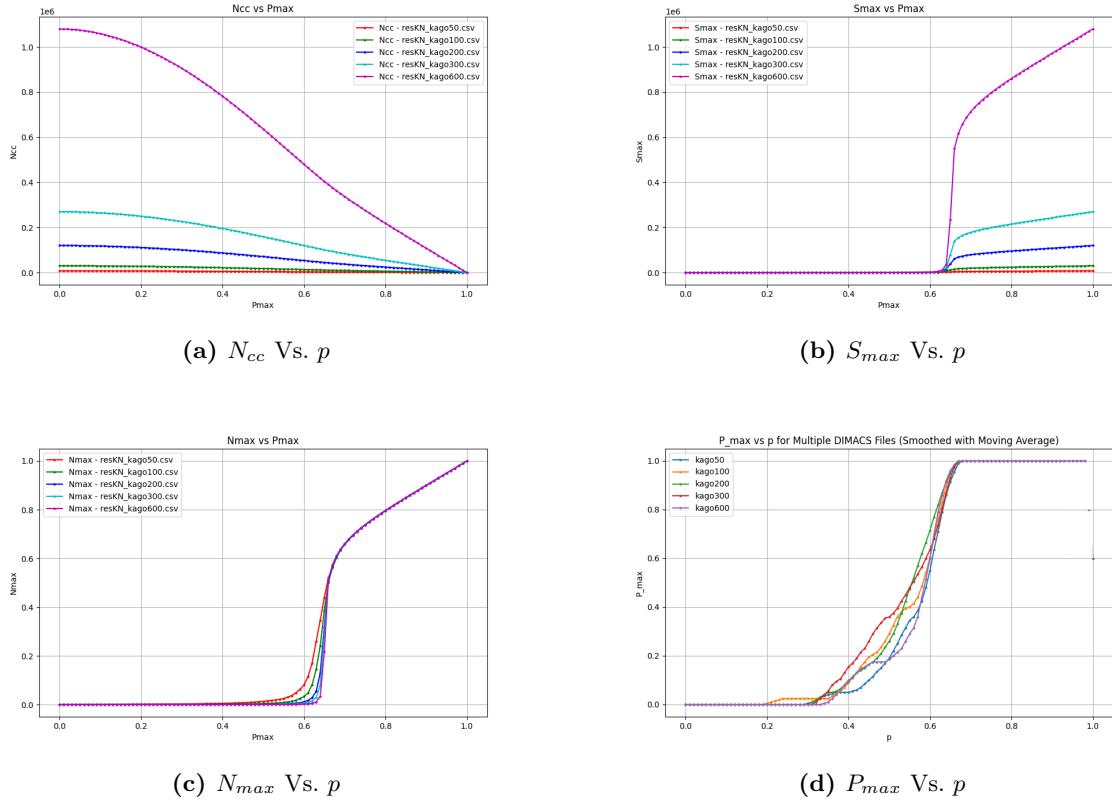
Una pequeña aclaración: Aunque las imágenes de la Figura 12 puedan llegar a parecer tridimensionales, el concepto de los grafos Kagome sigue siendo bidimensional. Se puede ver 3D por la representación gráfica que usa NetKet[9], en la que los nodos pueden estar distribuidos en el espacio de manera que parezcan tener profundidad. Por definición, las mallas Kagome son estructuras bidimensionales, más allá de la ilusión óptica que se puede formar al visualizarlos.

Podemos definir los grafos Kagome de la siguiente forma: Al elegir los vectores de red directa primitivos $\mathbf{a}_1 = \hat{x}$ y $\mathbf{a}_2 = \frac{1}{2}\hat{x} + \frac{\sqrt{3}}{2}\hat{y}$, entonces escribimos

$$a_R, b_R, c_R = \frac{1}{\sqrt{N}} \sum_k e^{i\mathbf{k} \cdot \mathbf{R}} (a_k e^{i2\mathbf{k} \cdot \mathbf{a}_1} b_k e^{i2\mathbf{k} \cdot \mathbf{a}_2} c_k).$$

Los vectores base son $\mathbf{0}$, $\frac{1}{2}\mathbf{a}_1$ y $\frac{1}{2}\mathbf{a}_2$. \mathbf{a}_1 y \mathbf{a}_2 son los vectores primitivos de la red Kagome, que definen la disposición geométrica. Usando estos vectores primitivos, puedes construir toda la red triangular en dos dimensiones.

Percolación por nodos



Percolación por aristas

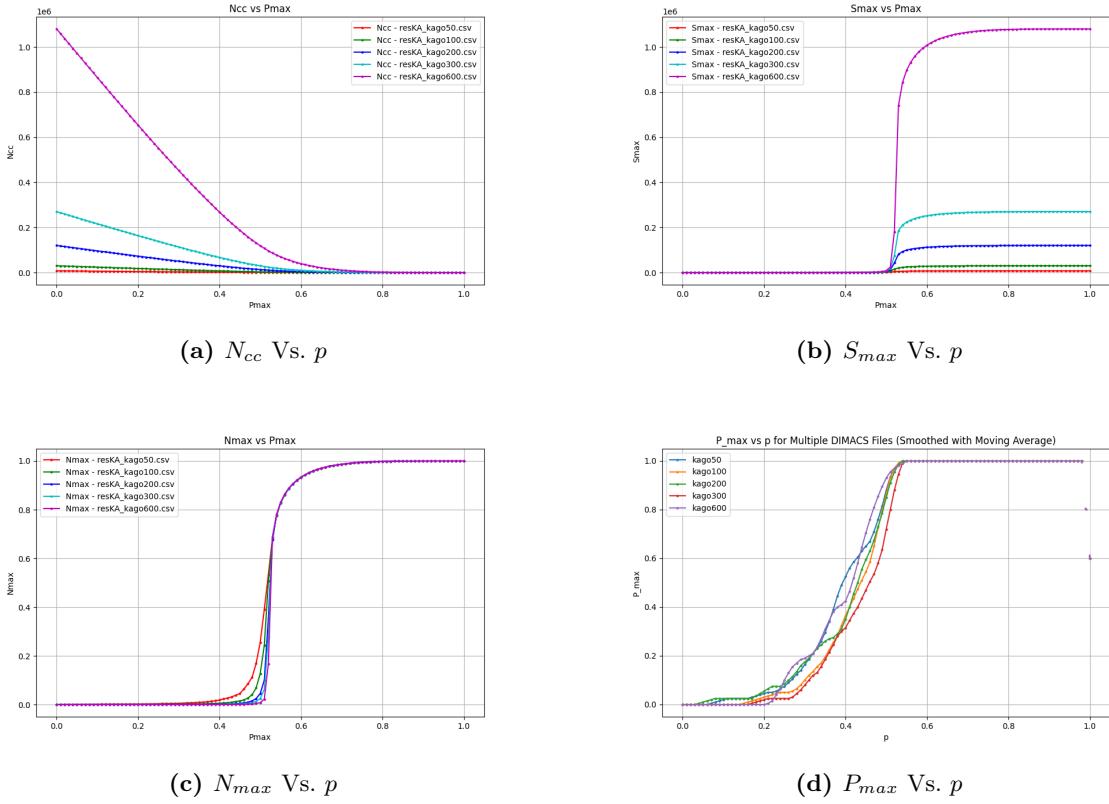


Figura 14: Resultados de la percolación por aristas en mallas Kagome de distintos tamaños.

Resultados y análisis

Los resultados obtenidos para la percolación por aristas y por nodos en grafos Kagome de distintos tamaños ($X = Y = 50, 100, 200, 300, 400, 475$) muestran comportamientos coherentes con la teoría conocida. A continuación, se presenta un análisis detallado de los diversos aspectos observados en las simulaciones.

Umbral de percolación y transición de fase

En la percolación por aristas se detecta una clara transición muy cerca del $p = 0,5$ (Figura 14a), que encaja con nuestros valores calculados de umbral de percolación y con los valores del umbral de percolación de otros estudios $p_c = 0,5244$ [18]. En el caso de la percolación por nodos (Figura 13a), no se distingue bien la transición de fase, y eso que eran los grafos más grandes que podíamos generar con la librería. Se puede observar un sutil cambio en las componentes conexas, sobre todo en 300×300 y 400×400 , alrededor de $p = 0,65$. Ese valor coincide con el valor de p_c calculado por nuestro programa y con el valor demostrado por nodos en redes Kagome, $0,652703645 \dots = 1 - 2 \times \sin(\frac{\pi}{18})$. [18] Solo vemos todos esos resultados cuando las dimensiones de la malla superan 300×300 .

Comportamiento del clúster mayor C_{max}

Como en las mallas cuadradas, en cuanto se llega al umbral de percolación p_c , el S_{max} incrementa sustancialmente (Figura 14b). En la percolación de aristas, también el exponente crítico β característico de la percolación en dos dimensiones se manifiesta en la pendiente cerca del punto crítico. Este comportamiento es especialmente notable en las simulaciones de tamaño 300×300 .

Igual que en las redes cuadradas, en la percolación por nodos la gráfica es distinta, al llegar al punto crítico hay un crecimiento instantáneo pero luego crece de forma lineal hasta el máximo.

Exponentes críticos

Usaremos los exponentes críticos universales de la percolación en dos dimensiones mencionados en el apartado 2.3.2.

También calcularemos β , usando un ajuste por regresión $\log - \log$, obtenemos un valor de 0,1263, ligeramente alejado del valor teórico 0,138. Igualmente fijándonos en la Figura 15, podemos ver que al aplicar el ajuste encontrado, obtenemos el valor exacto de $p_c = 0,5244$, que coincide con aquel calculado para mallas infinitas.

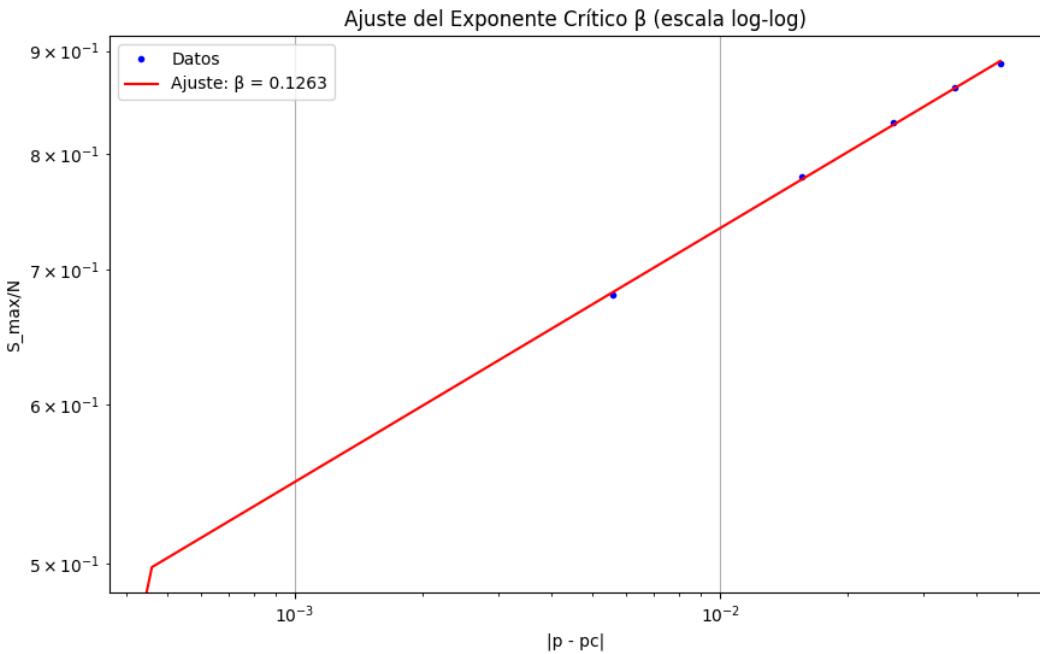


Figura 15: Exponente crítico β en grafos Kagome.

Comparación con resultados teóricos

Los valores de los umbrales críticos (p_c) se reproducen exactamente, tanto en percolación por aristas donde el umbral se acerca al valor aproximado de 0,5244 y en percolación por nodos donde el umbral se acerca al valor exacto demostrado de $0,652703645\dots = 1-2\times\sin(\pi/18)$. Además estos valores coinciden con las transiciones de fase observadas sobre el número de componentes conexos. Aunque el exponente crítico β no nos da su valor teórico exacto, vemos

como al ajustar nuestros datos a ella, se observan los comportamientos de *scaling*, igual que en la gráfica S_{max} (Figura 13b).

5. Conclusiones

Como se menciona en los libros, la teoría de la percolación atrae especialmente el interés entre los matemáticos, ya que es un problema cuya descripción es sencilla, pero cuyas soluciones y análisis requieren herramientas complejas y técnicas avanzadas. Esto ha llevado a una profunda exploración tanto teórica como computacional, destacando la belleza y dificultad intrínseca de fenómenos como la transición de fase y la aparición del clúster gigante en diferentes tipos de redes.

Este proyecto nos ha permitido asomarnos a un área de investigación muy interesante, que desde sus inicios hace más de 50 años, no ha dejado de estar activa, ya que muchas de las referencias consultadas son bastante recientes.

Sin duda un proyecto como este no de para profundizar mucho en un tema tan extenso. De hecho, durante la fase de investigación, la cantidad de papers y libros sobre el tema nos ha abrumado un poco y teníamos muchas cosas que nos hubiera gustado probar, pero el querer descubrir más sobre la percolación incluso nos ha podido ralentizar. No obstante, eso al final significa que es un proyecto que nos ha motivado bastante, y creemos que esta memoria, pese a los errores que pueda haber, lo refleja.

Sobre el trabajo realizado, queremos destacar la corrección de nuestros algoritmos, ya que, especialmente en las rejillas, hemos podido estimar el umbral de percolación y exponentes críticos de forma muy satisfactoria, ya que coinciden casi a la perfección con los valores expuestos en la literatura estimados teóricamente o numéricamente.

También hemos querido visualizar los procesos de percolación en mallas cuadradas confirmar más si cabe que nuestros algoritmos funcionan bien y las transiciones de fase ocurren cuando toca, siendo coherentes con los gráficos generados. En la Figura 16 podemos ver una rejilla cuadrada sometida a una percolación por nodos. Se aprecia que el umbral de percolación p_c ocurre entre los valores de $p = 0,59$ y $p = 6$, cuando el *superTop* y el *superBottom* se conectan. A partir de ese punto, vemos cómo el superclúster crece rápidamente en el momento en el que $p = 0,61$. Esta visualización refuerza los datos que hemos proporcionado, ajustándose al umbral de percolación teórico.

Lo mismo ocurre al aplicar percolación por aristas, con el umbral de percolación en $p = 0,5$.

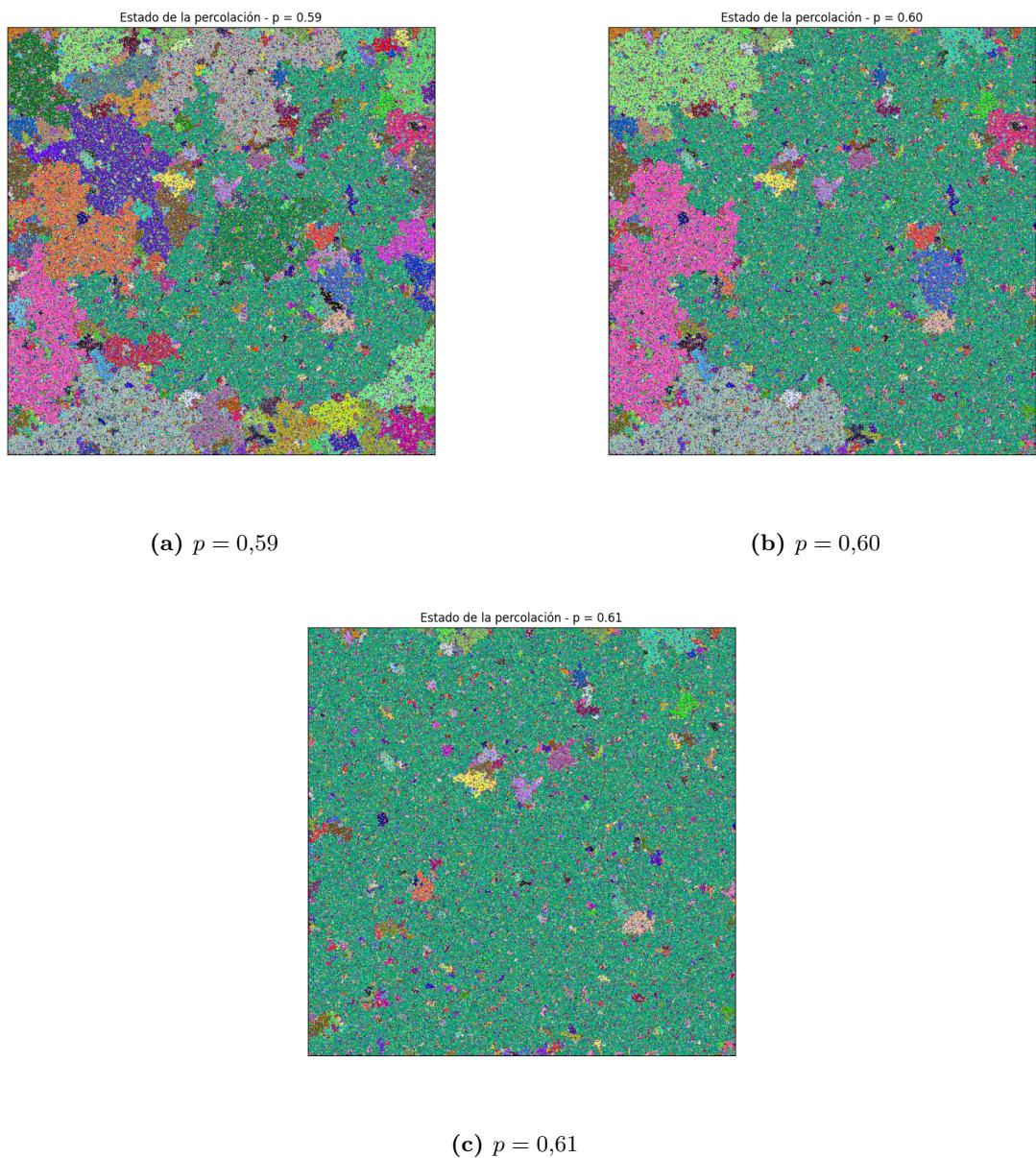


Figura 16: Visualización percolación por nodos en mallas cuadradas

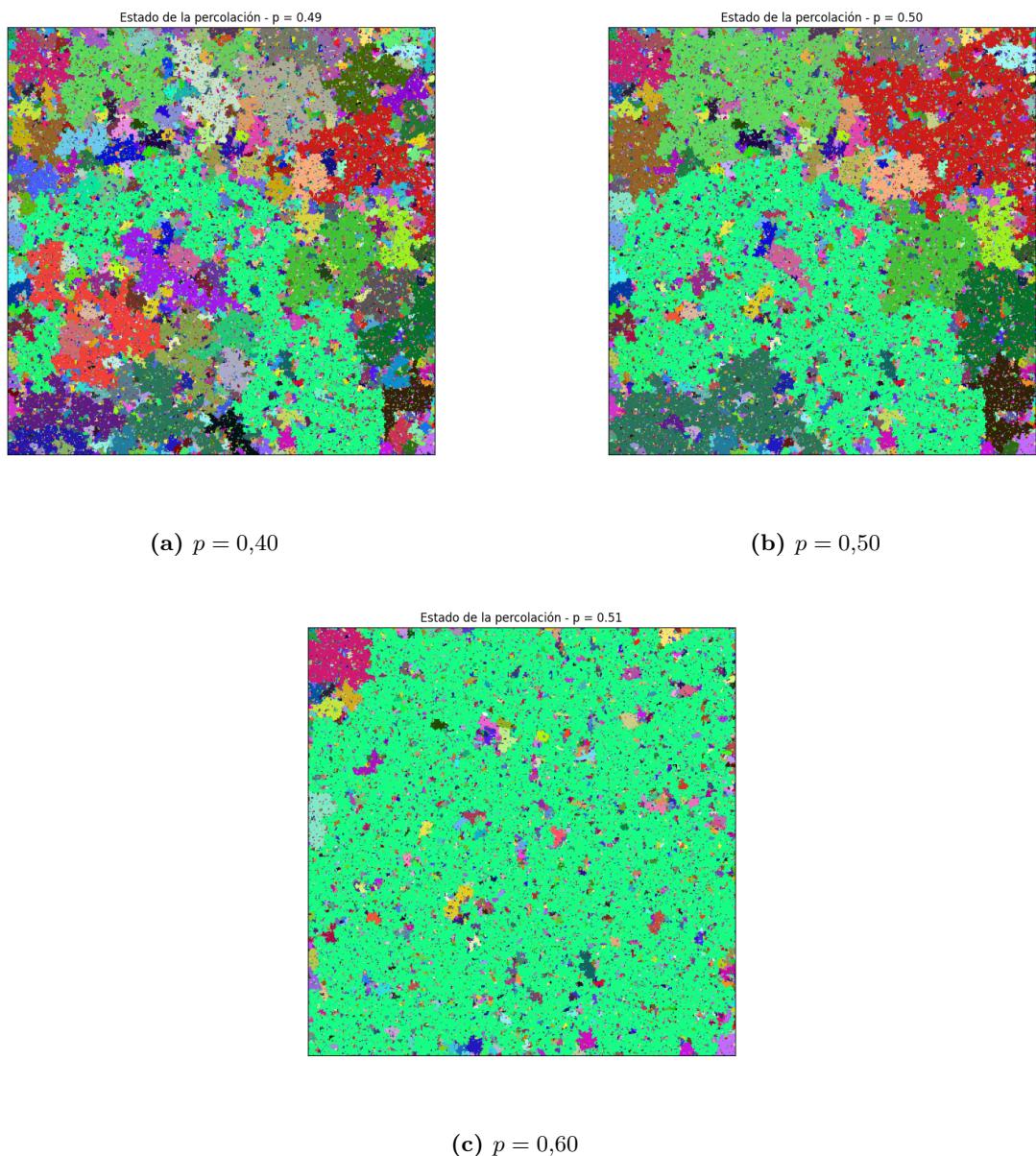


Figura 17: Visualización percolación por aristas en mallas cuadradas

6. Aplicaciones

Por último, querríamos mencionar algunas aplicaciones que tiene el hecho de simular procesos de percolación para explicar o predecir la realidad. Algunos ejemplos que hemos encontrado en la literatura son los siguientes.

- **Transporte de liposomas por redes de filamentos de actina.** En el estudio de transporte intracelular, simular la percolación de motores como Myosin Va a través de redes de actina ayuda a comprender cómo la densidad de filamentos de actina afecta el transporte. En este modelo, los nodos son las uniones de los filamentos de actina y los bordes representan las conexiones entre ellos, determinando si un liposoma logra desplazarse o queda atrapado en una caminata aleatoria debido a la alta densidad de la red.
- **Relación con el Target Set Selection Problem (TSS).** En este problema, el objetivo es identificar un conjunto mínimo de nodos inicialmente activos que logren propagar la activación a todo el grafo. Esta propagación puede modelarse como un proceso de percolación, donde los nodos son elementos susceptibles de activarse y las aristas representan la influencia entre ellos. Esto tiene aplicaciones en la difusión de información en redes sociales, la propagación de epidemias, o la penetración de nuevos productos en mercados.
- **Simulación del mercado de valores.** Como se menciona en The Statistical Mechanics of Capital Markets de J. Voit, los procesos de percolación pueden explicar comportamientos de mercado, ya que el comportamiento de los inversores y la correlación entre los precios de los activos exhiben propiedades similares a la percolación. En este contexto, los nodos pueden representar a los inversores o los activos, y las aristas las correlaciones entre ellos. A medida que aumenta la conectividad en el mercado, se podría producir una transición de fase hacia una volatilidad elevada o un crash financiero.
- **Detección de congestión en el tráfico.** Los procesos de percolación también se utilizan para modelar la congestión en redes de tráfico. Los nodos representan intersecciones o vehículos, mientras que las aristas son las carreteras o trayectos. Cuando un número crítico de aristas se sobrecarga, el sistema experimenta un comportamiento de percolación, lo que puede resultar en congestiones graves.

A. Glosario

A continuación se muestra un glosario con las notaciones más utilizadas en este documento:

Notación	Descripción
$G = (V, E)$	Grafo con un conjunto de nodos V y un conjunto de aristas E .
p	Probabilidad de que un nodo o arista esté activo (probabilidad de ocupación).
p_c	Umbral de percolación, valor de p en el que se produce la transición de fase.
Ω	Espacio de probabilidad.
$U(e)$	Número entre 0 y 1 asociado a la arista e en el proceso de percolación con acoplamiento uniforme.
$\omega(e)$	Estado (abierto (1) o cerrado (0)) de la arista e . También x_e .
$\eta_p(e)$	Estado (abierto (1) o cerrado (0)) en el que se encuentra e cuando la probabilidad es p .
η_p	Estado de un grafo durante el proceso de percolación en p (representa todos los $\eta_p(e)$).
$\mathbb{P}_p(\cdot)$	Producto de medida. i. e. probabilidad de que se produzca un estado η_p .
\mathbb{L}^d	Rejilla euclídea d -dimensional (<i>lattice</i>). Por simplicidad técnica, a veces usamos L en lugar de \mathbb{L} .
$L \times L$	Dimensiones de una red o rejilla cuadrada con L filas y L columnas.
$\mathcal{C}(x)$	Clúster al que pertenece un nodo x . Por simplicidad técnica, a veces usamos C en lugar de \mathcal{C} .
$\mathbb{P}_p(\mathcal{C}(0) = \infty)$	Probabilidad de que exista el clúster infinito (clúster gigante en la práctica) o probabilidad de que un sitio (nodo) aleatorio pertenezca a este. También expresado como la función $\theta(p)$, o por simpleza P_∞ o P_{max} .
C_{max}	Clúster de mayor tamaño, que a partir de p_c es el clúster gigante. Desde un punto de vista teórico es el clúster infinito C_∞ .
S_{max}	Número de nodos del clúster de mayor tamaño. Desde un punto de vista teórico es S_∞ .
N_{max}	Tamaño normalizado del clúster de mayor tamaño. Desde un punto de vista teórico es N_∞ .
N_{cc}	Número de componentes conexos que tiene cierto grafo.
S	Número promedio de nodos por clúster (no lo utilizamos).
ξ	Longitud de correlación, escala máxima en la que los clústeres están correlacionados (justo antes de p_c).
β	Exponente crítico que describe como crece P_{max} justo después de p_c .
ν	Exponente crítico que describe como ξ diverge cerca de p_c .
γ	Exponente crítico asociado con la divergencia de S cerca de p_c .
R	Radio de conexión en un grafo geométrico aleatorio (RGG).

Referencias

- [1] Rudolf A. Roemer. Percolation, Renormalization and the Quantum-Hall Transition. Institut für Physik, Technische Universität, Chemnitz, 2001.
- [2] Hoshen, J., & Kopelman, R. (1976). Percolation and cluster distribution. I. Cluster multiple labeling technique and critical concentration algorithm. Physical Review B, 14(8), 3438.
- [3] Broadbent, S. R., & Hammersley, J. M. (1957). Percolation processes: I. Crystals and mazes. *Mathematical Proceedings of the Cambridge Philosophical Society*, 53(3), 629–641. <https://doi.org/10.1017/S0305004100032680>
- [4] Umargono, E., Suseno, J. E., & Gunawan, S. (2019, March). K-Means clustering optimization using the elbow method and early centroid determination based-on mean and median. In Proceedings of the International Conferences on Information System and Technology (pp. 234-240). Setubal, Portugal: SCITEPRESS—Science and Technology Publications.
- [5] Abellán, J.L. (2021) Teorema Central del Límite (TCL), Economipedia. Available at: https://economipedia.com/definiciones/teorema-central-del-limite.html#google_vignette (Accessed: 22 October 2024).
- [6] Bazant, M. Z. (2000). Largest cluster in subcritical percolation. Physical Review E, 62(2), 1660
- [7] D. Stauffer and A. Aharony, *Introduction to Percolation Theory*, 2nd ed., Taylor & Francis, London, 1994.
- [8] Graph generators (no date)Graph generators - NetworkX 3.4.2 documentation. Available at: <https://networkx.org/documentation/stable/reference/generators.html> (Accessed: 22 October 2024).
- [9] Netket.graph.kagome (no date) NetKet. Available at: https://netket.readthedocs.io/en/latest/api/_generated/graph/netket.graph.Kagome.html (Accessed: 22 October 2024).
- [10] Kesten, H. (1980). The Critical Probability of Bond Percolation on the Square Lattice Equals 1/2. Communications in Mathematical Physics, 74, 41–59.
- [11] Akhunzhanov, R. K., Eserkepov, A. V., & Tarasevich, Y. Y. (2022). Exact percolation probabilities for a square lattice: Site percolation on a plane, cylinder, and torus. Journal of Physics A: Mathematical and Theoretical, 55(20), 204004.
- [12] Stauffer, D., & Aharony, A. (1994). *Introduction to Percolation Theory*. Taylor & Francis.
- [13] Fisher, M. E., & Barber, M. N. (1972). *Scaling Theory for Finite-Size Effects in the Critical Region*. Physical Review Letters, 28(23), 1516.
- [14] Cardy, J. (1996). *Scaling and Renormalization in Statistical Physics*. Cambridge University Press.
- [15] Sykes, M. F., & Essam, J. W. (1964). *Exact critical percolation probabilities for site and bond problems in two dimensions*. Journal of Mathematical Physics, 5(8), 1117-1127.
- [16] Sykes, M. F.; M. Glen; D. S. Gaunt (1974). "The percolation probability for the site problem on the triangular lattice". J. Phys. A: Math. Gen. 7 (9): L105–L108.

- [17] Levenshtein, M. E.; B. I. Shklovskii; M. S. Shur; A. L. Efros (1975). "The relation between the critical exponents of percolation theory". *Zh. Eksp. Teor. Fiz.* 69: 386–392.
- [18] Percolation threshold (2024) Wikipedia. Available at: https://en.wikipedia.org/wiki/Percolation_threshold#cite_note-SykesEssam-20 (Accessed: 23 October 2024).
- [19] Pearson, M.D.(2014); Random Generated Graphs"
- [20] J.Díaz;D.Mitsche;X.Pérez-Gímenez; .“On the Connectivity of Dynamic Random Geometric Graphs”
- [21] S.Walcott;D.M.Warshaw(2022);”Modeling myosin Va vesicle transport through actin filament networks reveals a percolation threshold that modulates transport properties”
- [22] Bollobás B. et al.;(2006);”Threshold Functions for Random Graphs and Random Geometric Graphs. In *Random Structures and Algorithms*, 28(4)”
- [23] Donelly,P.;(2004);”Phase transitions in random geometric graphs”