

**UNIVERSIDAD MAYOR DE SAN ANDRES**  
**FACULTAD DE CIENCIAS PURAS Y NATURALES**  
**CARRERA DE INFORMATICA**



**PERFIL DE TESIS DE GRADO**

**GENERADOR DE GRAFOS ALEATORIOS PARA CASOS DE PRUEBA**

Para obtener el Título de Licenciatura en Informática  
Mención Ingeniería de Sistemas Informáticos

**POR: CARLOS MIJAEL TOLA APAZA**

**TUTOR: LIC. JORGE TERAN**

**LA PAZ - BOLIVIA**

**Noviembre, 2021**

# Índice

<b>1. Introduccion</b>	<b>3</b>
<b>2. Antecedentes</b>	<b>3</b>
<b>3. Planteamiento del problema</b>	<b>4</b>
3.1. Problema central . . . . .	5
<b>4. Hipótesis</b>	<b>5</b>
<b>5. Definicion de objetivos</b>	<b>5</b>
5.1. Objetivo general . . . . .	5
5.2. Objetivos secundarios . . . . .	5
<b>6. Justificación</b>	<b>5</b>
6.1. Justificación económica . . . . .	5
6.2. Justificación social . . . . .	5
6.3. Justificación científica . . . . .	5
<b>7. Alcances y limites</b>	<b>6</b>
7.1. Alcances . . . . .	6
7.2. Límites . . . . .	6
<b>8. Metodología</b>	<b>6</b>
<b>9. Bibliografía</b>	<b>7</b>
<b>10. Anexos</b>	<b>8</b>
10.1. Cronograma de avance . . . . .	8
10.2. Indice de trabajo . . . . .	8

## 1. INTRODUCCION

Una red o grafo es un sistema compuesto por objetos discretos (vertices) y una relación (arista) entre algunos pares de ellos. Son herramientas de modelado ubicuas en muchos campos de investigación. Algunos ejemplos comunes incluyen redes sociales, neuronales, de proteínas y de comunicación. Hoy en día con el poder de computo disponible, han hecho que el estudio de redes y herramientas para analizarlas sean temas en alza dentro de la comunidad académica.

Los grafos aleatorios son construcciones probabilísticas que permiten definir realizaciones aleatorias de grafos. El uso generalizado de grafos aleatorios se ha destacado en el contexto de las aplicaciones de bases de datos y análisis de algoritmos durante varios años. Esto se debe a que tales estructuras de datos resultan ser muy útiles en muchas aplicaciones, desde el análisis de redes complejas hasta el estudio de algoritmos aleatorios. Generar estos grafos de manera aleatoria tiene un nivel de dificultad bastante alta, ya que al momento de generar un grafo se tienen que controlar algunas propiedades dependiendo el tipo de grafo que deseamos generar, por ejemplo: si el objetivo es construir un Arbol (Grafo conexo con  $E$  vertices y  $E - 1$  aristas) se debe tener control sobre donde se genera un nuevo nodo y arista, verificando que no existan ciclos, caso contrario el modelo no cumpliría con su objetivo. También podríamos mencionar que construir un Grafo Completo (Grafo con  $E$  vertices y  $\frac{E*(E-1)}{2}$  aristas) tiene una complejidad No Polinomial NP, haciendo que sea solo posible generar grafos completos con un límite de 20 nodos aproximadamente. Otro factor importante es que al ser este un modelado de grafos aleatorios, cada vertex a ser agregado, tiene cierta probabilidad de ser tomado en cuenta en el grafo generado o no.

Existen varios algoritmos generadores de grafos, como ser los modelos de grafos de Erdős-Rényi, y algunas variantes de dicho modelado, que no generan grafos representativos (no se asemejan a redes del mundo real), no pueden generar grafos con grandes cantidades de vertices y se toman bastante tiempo en el generado. Dados estos argumentos, el objetivo de este trabajo es crear un modelo generador de grafos aleatorios que sean representativos de forma eficiente. Cuando decimos eficiente, nos referimos a: mejores tiempos de ejecución para el generado y el mínimo uso de CPU, GPU y memoria.

Debido a que en Teoría de Grafos existe una infinidad de tipos de grafos, nos enfocaremos simplemente en un conjunto de grafos que son: Grafos directos acíclicos (DAG), Grafos Bipartitos, Grafos completos, Grafos no dirigidos conexos y no conexos, y árboles.

## 2. ANTECEDENTES

El primer modelo de grafos aleatorios fue introducidos por P. Erdős en 1959 como una herramienta para aplicar el método probabilístico para probar una relación asintótica entre la cintura y el número cromático de grafos generales. Más tarde, junto con A. Rényi, continuó el estudio de este modelo. Este es uno de los dos modelos conocidos hoy en día como modelos de Erdős-Rényi. Consiste en la construcción de grafos finitos con un número fijo de aristas distribuidas uniformemente a lo largo del conjunto de pares de vértices; mientras que el segundo modelo que comparte el mismo nombre, se debe a E. Gilbert, y consiste en grafos finitos obtenidos al establecer, independientemente y con la misma probabilidad, una arista entre cada par de vértices. A partir de ahí, se desarrollaron muchos modelos de grafos aleatorios diferentes. Estos probaron ser no sólo construcciones interesantes, sino también muy útiles en aplicaciones: como herramientas para estudiar escenarios promedio en

redes complejas y como una forma de explorar posibles mecanismos que dan lugar a muchas de las características distintivas de las redes observadas en el mundo real. Por ejemplo, incluso un modelo tan simple como el de Erdős-Rényi exhibe la propiedad del mundo pequeño (prevalencia de distancias relativamente cortas entre la mayoría de los pares de vértices) observada en muchas redes sociales y de comunicación.

Al respecto de las investigaciones relacionadas con el tema podemos destacar las siguientes:

- **Titulo:** Desempeño asintótico de algoritmos secuenciales en grafos aleatorios - Tesis presentada para optar al título de Doctor.  
**Autor:** Lic. Manuel Sáenz  
**Año:** 2019  
**Institucion:** Universidad de Buenos Aires  
En el mencionado trabajo, se estudia la evolución y convergencia de algoritmos generadores de grafos, así de esa forma lograr delimitar el algoritmo de modelado de grafos Erdős-Rényi. Para demostrar sus limitaciones se demuestra que para los grafos generados por el modelo Erdős-Rényi es posible construir un conjunto independiente máximo en tiempo polinomial. Pese a que el problema de construir Conjuntos Independientes Máximos es del tipo NP-difícil.
- **Titulo:** Generación rápida de gráficos aleatorios  
**Autor:** Sadegh Heyrani, Nobari Xuesong Lu  
**Año:** 2011  
**Institucion:** Universidad Nacional de Singapore  
En el mencionado trabajo, se plantea un modelo generador de grafos, usando programación paralela y GPU junto a su algoritmo ER y ZER (creado por ellos). Los grafos que pueden ser generados por este método son: Grafos dirigidos y no dirigidos.
- **Titulo:** Implementación de un generador de topologías aleatorias en emulador de red Mininet.  
**Autor:** Rosa Alejandra García Grisales, Juan Carlos Agudelo Calderón  
**Año:** 2015  
**Institucion:** Universidad Católica De Pereira  
En el mencionado trabajo, se plantea implementar un generador de grafos para el emulador Mininet, que es una aplicación desarrollada para la investigación de prototipos SDN en tiempo real. Este método usa un Modelo de Waxmax (variante de Erdős-Rényi), que construye grafos en tiempo lineal, pero usando bastante memoria limitando el generado a aproximadamente 1000 nodos solamente.
- **Titulo:** Modelado de grafos aleatorios: una revisión de los conceptos  
**Autor:** Mikhail Drobysheskiy, Denis Turdakov  
**Año:** 2019  
**Institucion:** Instituto de Física y Tecnología de Moscú  
En este trabajo, se hace un análisis a profundidad de el Algoritmo Erdős-Rényi. Se identificó seis direcciones donde este modelo tiene sus aplicaciones: comprensión de redes, análisis, extrapolación, puntos de referencia, modelos nulos y aleatorización.

### 3. PLANTEAMIENTO DEL PROBLEMA

El modelado de grafos juega un rol importante en el análisis de redes complejas. Nos ayudan a entender, controlar predecir fenómenos que pueden ocurrir en redes sociales, redes biológicas, internet y muchos más. Modelar estos grafos manualmente para cada aplicación podría requerir

demasiado tiempo y esfuerzo. Aunque ya existen, varios modelos generadores de grafos, esto no suelen ser representativos y al mismo tiempo son costosos en tiempo y memoria.

### **3.1. PROBLEMA CENTRAL**

¿Con que grado de fiabilidad los modelos generadores de grafos aleatorios actuales puede construir grafos representativos en tiempo y coste de memoria minimos?

## **4. HIPÓTESIS**

El metodo propuesto para la generacion de grafos aleatorios usando GPU y programacion paralela genera grafos representativos en tiempo y memoria eficiente con una fiabilidad del 90 %.

## **5. DEFINICION DE OBJETIVOS**

### **5.1. OBJETIVO GENERAL**

Desarrollar un modelo generador de grafos aleatorios representativos y eficiente en tiempo y memoria.

### **5.2. OBJETIVOS SECUNDARIOS**

- Analizar y diseñar un modelo generador de grafos aleatorios para: Grafos directos acíclicos (DAG), Grafos Bipartitos, Grafos completos, Grafos no dirigidos conexos y no conexos, y árboles.
- Verificar que los tiempos al generar los grafos sean eficientes experimentalmente.
- Verificar que el generador de grafos sea eficiente según su complejidad algorítmica.

## **6. JUSTIFICACIÓN**

### **6.1. JUSTIFICACIÓN ECONÓMICA**

Para el analisis de diferentes redes, se requiere tener acceso a distintas redes de prueba, para su simulacion y estudio. Generar estas redes o grafos de manera eficiente significa que se reduzcan los costos de energia y se eliminara el tiempo invertido en dicho modelado.

### **6.2. JUSTIFICACIÓN SOCIAL**

El impacto que tienen las redes sociales hoy en día en la sociedad ha marcado un hito en la forma de comunicarnos, debido a su crecimiento actual estas redes sociales siempre esta a la mejora de sus algoritmos. Para esto es necesario investigar y probar dichos algoritmos en grafos aleatorias.

### **6.3. JUSTIFICACIÓN CIENTÍFICA**

El presente trabajo combinara tecnicas algorítmicas, mascarar de bits, con tecnicas matematicas, Hashing, de tal manera que el modelado de las redes o grafos sea eficiente, esto se puede verificar usando notaciones de complejidad algorítmicas.

## 7. ALCANCES Y LIMITES

### 7.1. ALCANCES

- El presente trabajo diseñara un modelo generador de grafos aleatorios para DAGs, Grafos bipartitos, Grafos completos, Grafos no dirigidos conexos y no conexos, y arboles de manera eficiente.

### 7.2. LÍMITES

- Algunos tipos de grafos con construcciones y solucion no polinomial NP no seran tomados en cuenta.

## 8. METODOLOGÍA

Para el desarrollo del presente trabajo se utilizara un **Metodo de investigacion experimental** para cumplir con los objetivos propuestos para el modelo generador de grafos aleatorios. Esta se dividiran en las siguientes etapas.

1. Recopilacion de informacion.
2. Elaboracion del Modelo generador de grafos aleatorios.
3. Pruebas de funcionamiento y/o presentación de mejoras para el modelo.
4. Comparaciones y analisis de los datos.
5. Conclusiones.

La primera etapa se recopilara la informacion necesaria y se estudiara temas relacionadas al mismo.

La segunda etapa se realizara el Modelo generado de grafos tomando encuesta los alcances y limites.

La tercera etapa se verificara la funcionalidad del modelo.

En la cuarta etapa se analizaran los datos y se los comparara con otros modelos.

En la ultima etapa, se presentaran los resultados finales, las conclusienes y recomendaciones con respecto al proyecto

ETAPAS	ACTIVIDADES
Recopilacion de informacion	<ul style="list-style-type: none"> <li>- Investigar la existenciaa de modelos generadores de grafos.</li> <li>- Estudiar el proceso de generado de dichos modelos investigados.</li> </ul>
Elaboracion del modelo	<ul style="list-style-type: none"> <li>- Analizar y diseñar las etapas que necesitaran los modelados.</li> <li>- Modelar el proceso generico, para construir los grafos de forma aleatoria (Dirigidos y no Dirigidos).</li> <li>- Modelaro el proceso para los distintos grafos descritos en los alcances (DAG, Completos y Arboles).</li> </ul>
Pruebas de funcionamiento y mejoras.	<ul style="list-style-type: none"> <li>- Verificar el funcionamiento pleno del modelo.</li> <li>- Aplicar las mejoras encontradas y propuestas.</li> </ul>
Comparacion y Analisis	<ul style="list-style-type: none"> <li>- Comparar el modelado propuesto con respector al modelos ya existentes.</li> <li>- Analizar de formar estadisticaa los datos devuelto de las comparaciones.</li> <li>- Generalizar y encontrar la complejidad Algoritmica.</li> </ul>
Conclusiones	<ul style="list-style-type: none"> <li>- Presentar las conclusiones obtenias al finalizar el trabajo.</li> <li>- Compartir algunas recomendaciones obtenidas del trabajo realizado.</li> </ul>

## 9. BIBLIOGRAFÍA

Manuel Zaens. (2019). Desempeño asintotico de algoritmos secuenciales en grafos aleatorios. 2019, de Universidad de Buenos Aires Sitio web:

[http://cms.dm.uba.ar/academico/carreras/doctorado/Tesis\\_Saenz\\_corregida\(1\).pdf](http://cms.dm.uba.ar/academico/carreras/doctorado/Tesis_Saenz_corregida(1).pdf)

Sadegh Heyrani, Nobari Xuesong Lu. (2011). Generación rápida de gráficos aleatorios. 2019, de Universidad Nacional de Singapore Sitio web: <https://www.cs.au.dk/~karras/rg.pdf>

Rosa Alejandra García Grisales, Juan Carlos Agudelo Calderón. (2015). Implementacion de un generador de topologias aleatorias en emulador de red mininet.. 2019, de Universidad Católica De Pereira Sitio web: <https://repositorio.ucp.edu.co/bitstream/10785/3678/1/CDMIST122.pdf>

Mikhail Drobyshevskiy, Denis Turdakov. (2019). Modelado de grafos aleatorios: una revisión de los conceptos. 2019, de Mikhail Drobyshevskiy, Denis Turdakov Sitio web: <https://dl.acm.org/doi/pdf/10.1145/3369782>

## 10. ANEXOS

### 10.1. CRONOGRAMA DE AVANCE

ACTIVIDADES	DURACION (DIAS)	DEL 1 DE FEBRERO A 30 DE JUNIO																							
		Diciembre				Enero				Febrero				Marzo				Abril				Mayo			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Redacción del Capitulo II Marco Teórico	14																								
Desarrollo del Capítulo III Marco Aplicativo	21																								
Modelo generador de grafos DAG	28																								
Modelo generador de grafos Bipartitos	35																								
Modelo generador de grafos completos	28																								
Algoritmo validador del modelo	21																								
Redacción del Capitulo III Marco Aplicativo	21																								
Redacción del Capitulo IV Estado de la Hipótesis	21																								
Redacción del Capitulo V Conclusiones y Recomendaciones	14																								

### 10.2. INDICE DE TRABAJO

#### 1. CAPÍTULO 1: MARCO REFERENCIAL

- 1.1 Introduccion
- 1.2 Antecedentes
- 1.3 Planteamiento del problema
  - 1.3.1 Problema central
  - 1.3.2 Problemas secundarios
- 1.4 Objetivo
  - 1.4.1 Objetivo principal



- 1.4.2 objetivos secundarios
- 1.5 Hipótesis
- 1.6 Justificación
  - 1.6.1 Justificación económica
  - 1.6.2 Justificación social
  - 1.6.3 Justificación científica
- 1.7 Alcances y Límites
  - 1.7.1 Alcances
  - 1.7.2 Límites
- 1.8 Metodología
- 2. CAPITULO 2: MARCO TEORICO
  - 2.1 Mascaras de bits
    - 2.1.1 Operaciones con mascarar de bits
  - 2.2 Programación dinámica (DP)
    - 2.2.1 DP con mascarar de bits
  - 2.3 Teoría de grafos
    - 2.3.1 Grafos acíclicos no diridos (DAG)
    - 2.3.2 Grafos Bipartitos
    - 2.3.3 Grafos completos
    - 2.3.4 Caminos y ciclos hamiltonianos
  - 2.4 Grafos aleatorios
    - 2.4.1 Modelo de Erdős-Rényi
    - 2.4.2 Modelo de Configuraciones
    - 2.4.3 Otros modelos de grafos aleatorios
  - 2.5 Cadenas de Markov
- 3. CAPITULO 3: MARCO APLICATIVO
  - 3.1 Modelo generador de grafos DAG
  - 3.2 Modelo generador de grafos Bipartitos
  - 3.3 Modelo generador de grafos completos
  - 3.4 Algoritmo validdor del modelo
- 4. CAPITULO 4: ANÁLISIS Y RESULTADOS
  - 4.1 Complejidad algoritmica.
  - 4.2 Demostracion de la mejora con respecto a otros modelos.
- 5. CAPITULO 5:
  - 5.1 Conclusiones
  - 5.2 Recomendaciones
- 6. Bibliografía
- 7. Anexos