Universidad de Granada

Grado en Ingeniería Informática. Cuarto.

Práctica 3: Poda y Visualización de Redes.

Nombre de la asignatura:
Redes y Sistemas Complejos. Lunes de 10:30 a 12:30.

Realizado por:
Néstor Rodríguez Vico. DNI: 75573052C.
email: nrv23@correo.ugr.es

ETSIIT
Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE TELECOMUNICACIÓN.

Granada, 19 de noviembre de 2017.

Índice

1. Aclaración.

Debido a la complejidad de cargar todos los resultados obtenidos en Gephi y ir, de forma manual, calculando todos los datos que se nos pide en la práctica, he hecho un programa en *Python* usando la librería *networkx* (la cual implementa los mismos algoritmos usados por Gephi) para automatizar el proceso. Dicho programa lee un fichero *.net* (al igual que Gephi) y calcula los datos necesarios. Este proceso lo hace para todos los ficheros de mi directorio *resultados*. De la misma manera, se a automatizado el proceso de visualización usando dicha biblioteca. El programa no se ha entregado porque no es parte de la práctica; pero si lo desea, avíseme y se lo mando.

2. Parte 1.

Los diez cienciogramas elegidos son los correspondientes a Argentina 2005, Canada 3005, China 2002, France 2002, Germany 2002, Spain 1996, Spain 1998, Spain 2002, Spain 2004 y World. Los resultados que voy a mostrar a continuación son los obtenidos con la versión original de Pathfinder.

Argentina 2005	**Lados/Densidad**	**Distancia media <d>**
Red Original	17938/0,50895	1,49661
q=2	324/0,00919	4,70024
q=3	277/0,00786	5,72629
q=4	269/0,00763	6,07998
q=5	268/0,0076	6,1138
q=n-1	267/0,00758	6,21107

Canada 2005	**Lados/Densidad**	**Distancia media <d>**
Red Original	30574/0,73979	1,2604
q=2	340/0,00823	4,9933
q=3	301/0,00728	5,93169
q=4	290/0,00702	6,73432
q=5	287/0,00694	7,78934
q=n-1	287/0,00694	7,78934

China 2002	**Lados/Densidad**	**Distancia media <d>**
Red Original	20661/0,61839	1,38314
q=2	306/0,00916	4,98327
q=3	268/0,00802	5,98614
q=4	262/0,00784	6,82308
q=5	260/0,00778	7,59349
q=n-1	260/0,00778	7,59349

France 2002	Lados/Densidad	Distancia media <d>
Red Original	23986/0,67545	1,3252
q=2	312/0,00879	4,97032
q=3	275/0,00774	5,87753
q=4	271/0,00763	6,51922
q=5	270/0,0076	6,71843
q=n-1	268/0,00755	7,52178

Germany 2002	Lados/Densidad	Distancia media <d>
Red Original	25395/0,70452	1,29587
q=2	313/0,00868	5,18055
q=3	277/0,00768	6,00871
q=4	272/0,00755	6,84695
q=5	270/0,00749	7,69833
q=n-1	269/0,00746	7,90781

Spain 1996	Lados/Densidad	Distancia media <d>
Red Original	5967/0,20294	1,9648
q=2	394/0,0134	4,54668
q=3	313/0,01065	5,87797
q=4	303/0,01031	6,68432
q=5	300/0,0102	6,98738
q=n-1	300/0,0102	6,98738

Spain 1998	Lados/Densidad	Distancia media <d>
Red Original	12971/0,39125	1,63795
q=2	320/0,00965	5,02775
q=3	279/0,00842	6,42853
q=4	267/0,00805	8,05815
q=5	267/0,00805	8,05815
q=n-1	267/0,00805	8,05815

Spain 2002	Lados/Densidad	Distancia media <d>
Red Original	21807/0,62815	1,37234
q=2	320/0,00922	4,79197
q=3	274/0,00789	5,82898
q=4	265/0,00763	6,54606
q=5	263/0,00758	6,87032
q=n-1	263/0,00758	6,87032

Spain 2004	**Lados/Densidad**	**Distancia media <d>**
Red Original	24991/0,69331	1,30744
q=2	332/0,00921	4,77562
q=3	280/0,00777	5,88096
q=4	272/0,00755	7,03493
q=5	271/0,00752	7,19861
q=n-1	270/0,00749	7,71223

World	**Lados/Densidad**	**Distancia media <d>**
Red Original	20154/0,85207	1,14793
q=2	280/0,01184	4,93836
q=3	233/0,00985	6,236
q=4	223/0,00943	6,93756
q=5	220/0,0093	7,19012
q=n-1	217/0,00917	7,69792

Una vez tenemos los datos recogidos, vamos a analizar lo que sucede. Podemos ver que, independientemente del cienciograma, se cumple que, al usar un valor de $q = 2$, se realiza una poda bastante fuerte en el número de enlaces. Esto también lo podemos ver reflejado en cómo varía la densidad de la red en la red original frente a la densidad de la red podada con $q = 2$. Según vamos subiendo el valor de q, se sigue podando pero no de una forma tan fuerte.

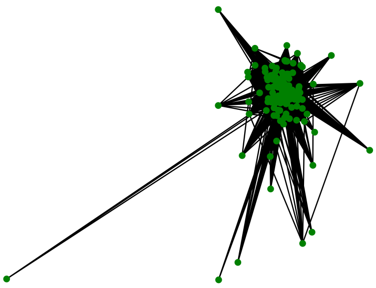
Al igual que se reduce la densidad de la red obtenida, aumenta el valor de la distancia media. ¿Por qué sucede esto? Esto se debe a que, al podar enlaces entre nodos, se reduce el número de caminos entre los nodos y por lo tanto, en algunas ocasiones hay que dar más saltos (de media) para ir de un nodo a otro.

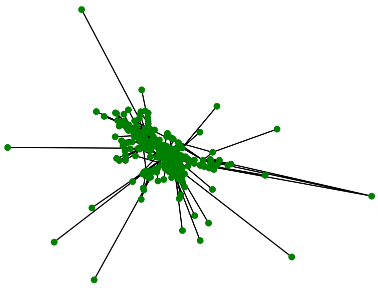
Hay en ciertos cienciogramas que, a pesar de subir el valor de q, no se produce más poda:

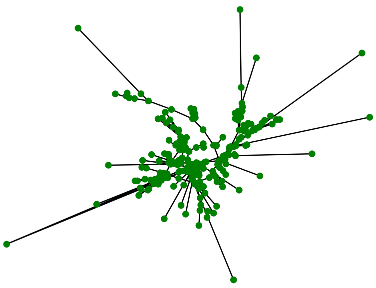
- Argentina 2005: este valor se alcanza con $q = n - 1$.
- Canada 2005: este valor se alcanza con $q = 5$. Por mucho que subamos el valor de q, no vamos a obtener una mayor poda.
- China 2002: este valor se alcanza con $q = 5$. Por mucho que subamos el valor de q, no vamos a obtener una mayor poda.
- Francia 2002: este valor se alcanza con $q = n - 1$.
- Germany 2002: este valor se alcanza con $q = n - 1$.
- Spain 1996: este valor se alcanza con $q = 5$. Por mucho que subamos el valor de q, no vamos a obtener una mayor poda.

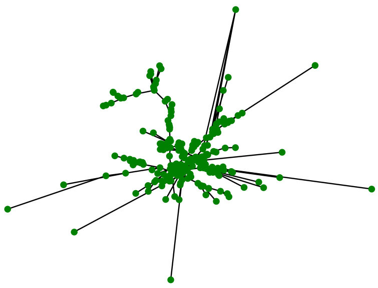
- Spain 1998: este valor se alcanza con q = *4*. Por mucho que subamos el valor de q, no vamos a obtener una mayor poda.
- Spain 2002: este valor se alcanza con q = *5*. Por mucho que subamos el valor de q, no vamos a obtener una mayor poda.
- Spain 2004: este valor se alcanza con q = n - *1*.
- World: este valor se alcanza con q = n - *1*.

A continuación vamos a visualizar algunas de las PFNETs obtenidas. Primero vamos a ver la representación de la red *Spain 2004*. Primero vamos a visualizarlas con Fruchterman Reingold. En la fila de arriba tenemos a la izquierda la red original y a la derecha la red podada con q = *2*. En la fila de abajo tenemos a la izquierda la red podada con q = *5* y a la derecha la red podada con q = n - *1*.

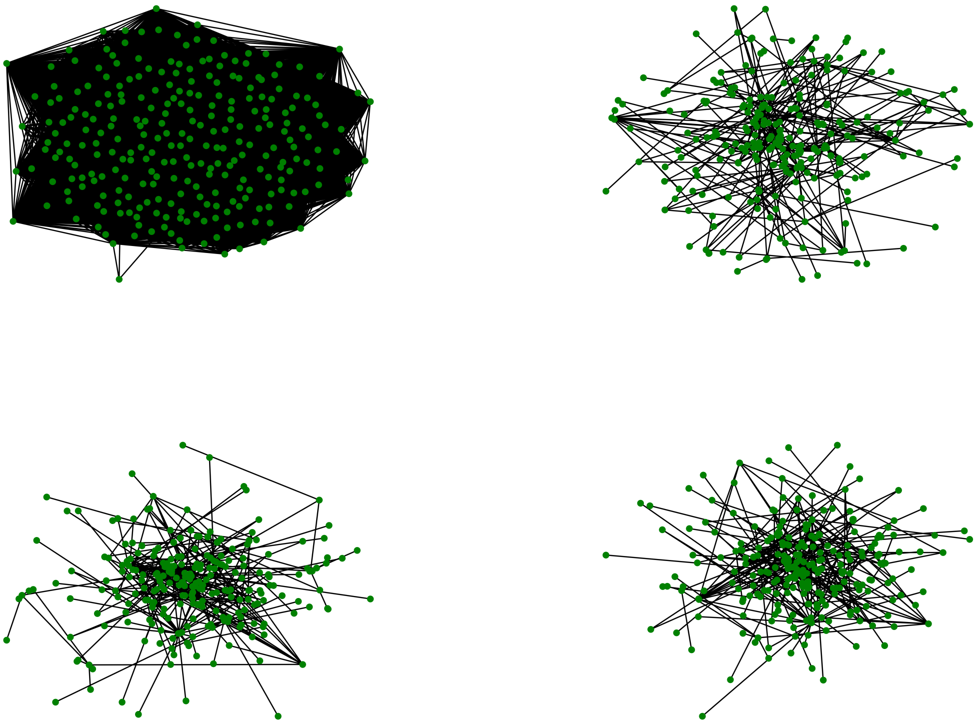




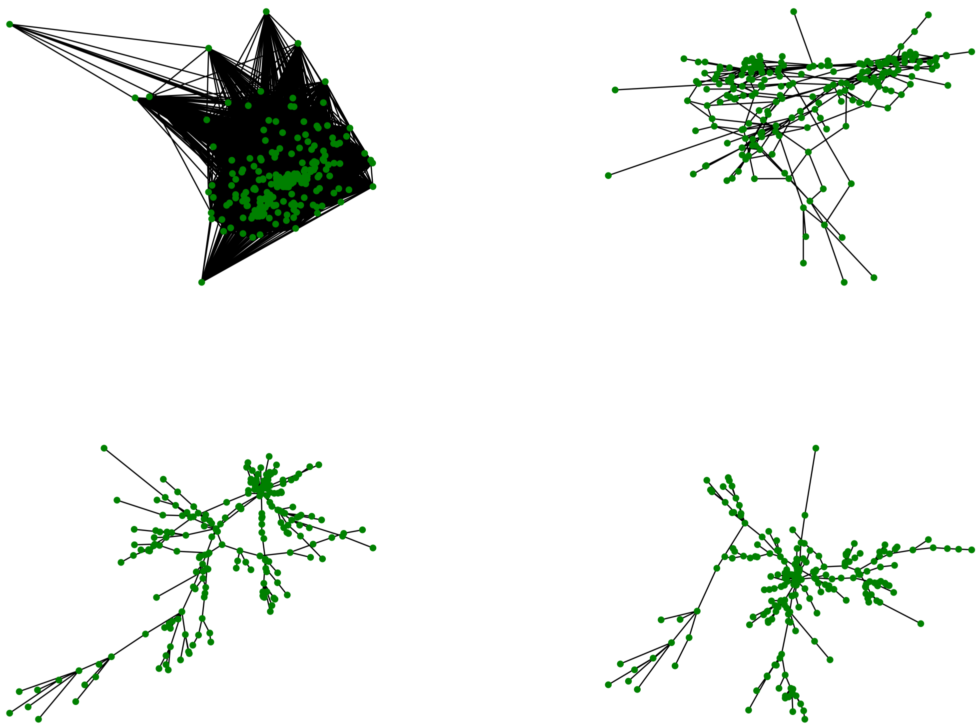




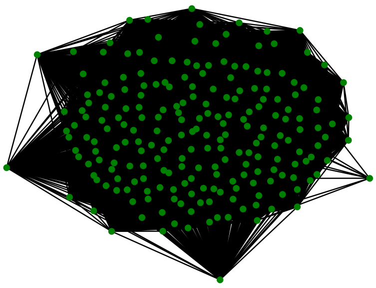
Primero vamos a visualizarlas con Kamada Kawai. En la fila de arriba tenemos a la izquierda la red original y a la derecha la red podada con q = *2*. En la fila de abajo tenemos a la izquierda la red podada con q = *5* y a la derecha la red podada con q = n - *1*.

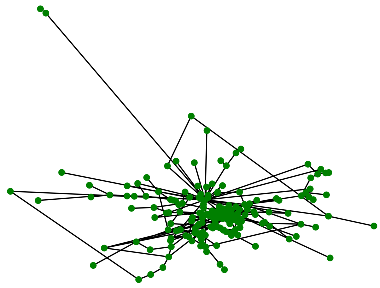


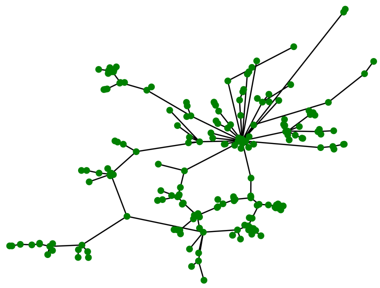
A continuación vamos a ver la red *World*. Primero vamos a visualizarlas con Fruchterman Reingold. En la fila de arriba tenemos a la izquierda la red original y a la derecha la red podada con $q = 2$. En la fila de abajo tenemos a la izquierda la red podada con $q = 5$ y a la derecha la red podada con $q = n - 1$.

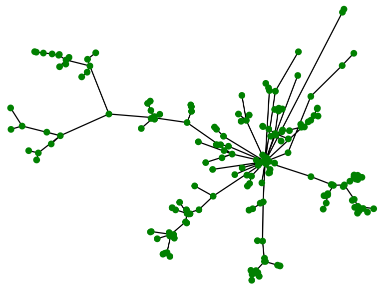


Primero vamos a visualizarlas con Kamada Kawai. En la fila de arriba tenemos a la izquierda la red original y a la derecha la red podada con $q = 2$. En la fila de abajo tenemos a la izquierda la red podada con $q = 5$ y a la derecha la red podada con $q = n - 1$.









A mí, personalmente, me gusta más la representación que obtenemos con el algoritmo de visualización Kamada Kawai. Evidentemente, a un mayor valor de q obtenemos una red con menos lados y, por lo tanto, obtenemos una red que se puede visualizar más cómodamente. Aún así, desde un punto de vista meramente estético, la red de *World* con q = n - *1* es la que mejor se ve.

3. Parte 2.

Las redes han sigo generadas todas de la misma forma, usando los siguientes parámetros:

- size: tamaño correspondiente.
- 1: indica que la red es no dirigida.
- 0: indica que tiene ceros en la diagonal.
- 1: pesos reales aleatorios en la red.
- 1.5, 10.5: los pesos varían entre 1.5 y 10.5 .
- 0.1: probabilidad de 0.1 de crear enlace.

Los resultados obtenidos son los siguientes:

N	Media E	Media D	Original	Binary	Fast	MST-T	MST-P
500	499	1,996	60,915	5,591	0,4	0,002	0,002
1000	999	1,998	1207,948	53,858	2,872	0,007	0,007
2000	1999,2	1,999	Timeout	1227,097	23,938	0,033	0,036
5000	4999,4	2	Timeout	Timeout	352,938	0,334	0,297
10000	10002,4	2	Timeout	Timeout	Timeout	1,407	1,744

Como podemos ver, los algoritmos menos eficientes son la versión Original, la versión Binary y, en una última instancia, la versión Fast. Si observamos, el valor de media del número de aristas se corresponde con el número de nodos. Esto nos indica que se podan bastantes aristas, obteniendo una red totalmente conectada sin necesidad de tener multiples aristas entre dos nodos (sólo tenemos una arista entre cada nodo). Debido a esta reducción de las aristas hasta llegar al número de nodos menos uno, el grado, sea cual sea el tamaño, es prácticamente 2. En cuanto a tiempos, podemos ver la gran diferencia entre los algoritmos MST (en sus dos versiones) frente a los demás, sacando una diferencia notable incluso en redes de tamaños pequeños.

En mi caso, de las dos versiones de MST la más rápida es la versión teórica, la cual tiene un orden de complejidad $O(n^2 log\ n)$. Aún así, tal y como podemos ver en al PDF que explica los distintos algoritmos de Pathfinder puede darse el caso de que la versión práctica, con un orden de complejidad $O(n^3)$ sea más rápido. Esto se debe a las estructuras de datos que usa esta versión práctica. Aún así, esto no sucede en mi caso, ya que es más rápido la versión teórica que la práctica.

A continuación, vamos a visualizar (esta vez si es con Gephi) las redes obtenidas por el algoritmo MST en su versión teórica para los tamaños 2000, 5000 y 10000. Vamos a usar el algoritmo de visualización Fruchterman Reingold. Los resultados son:

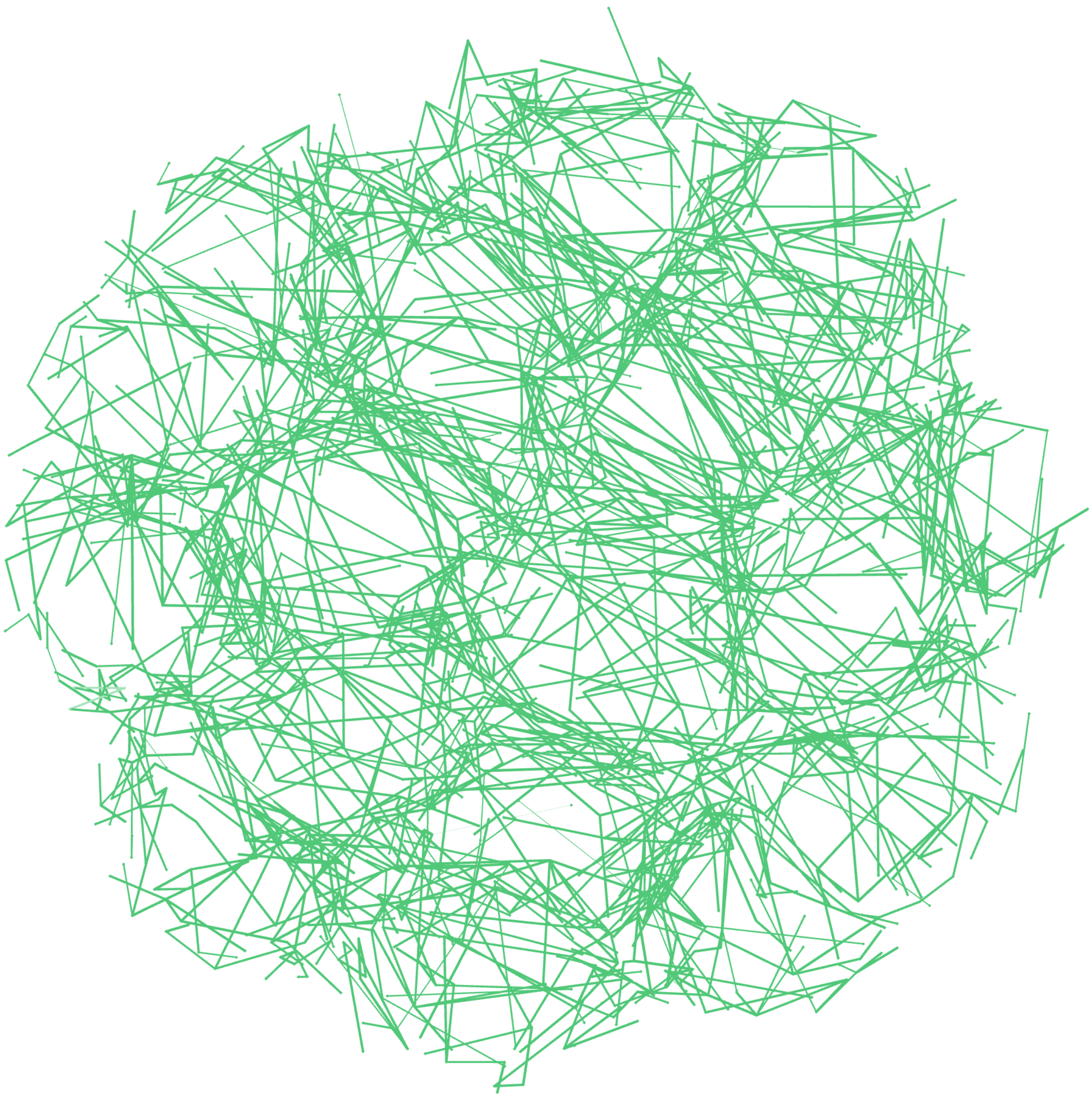


Figura 3.1: Tamaño 2000. Parámetros: Gravedad = 5.0 y Velocidad = 300.0

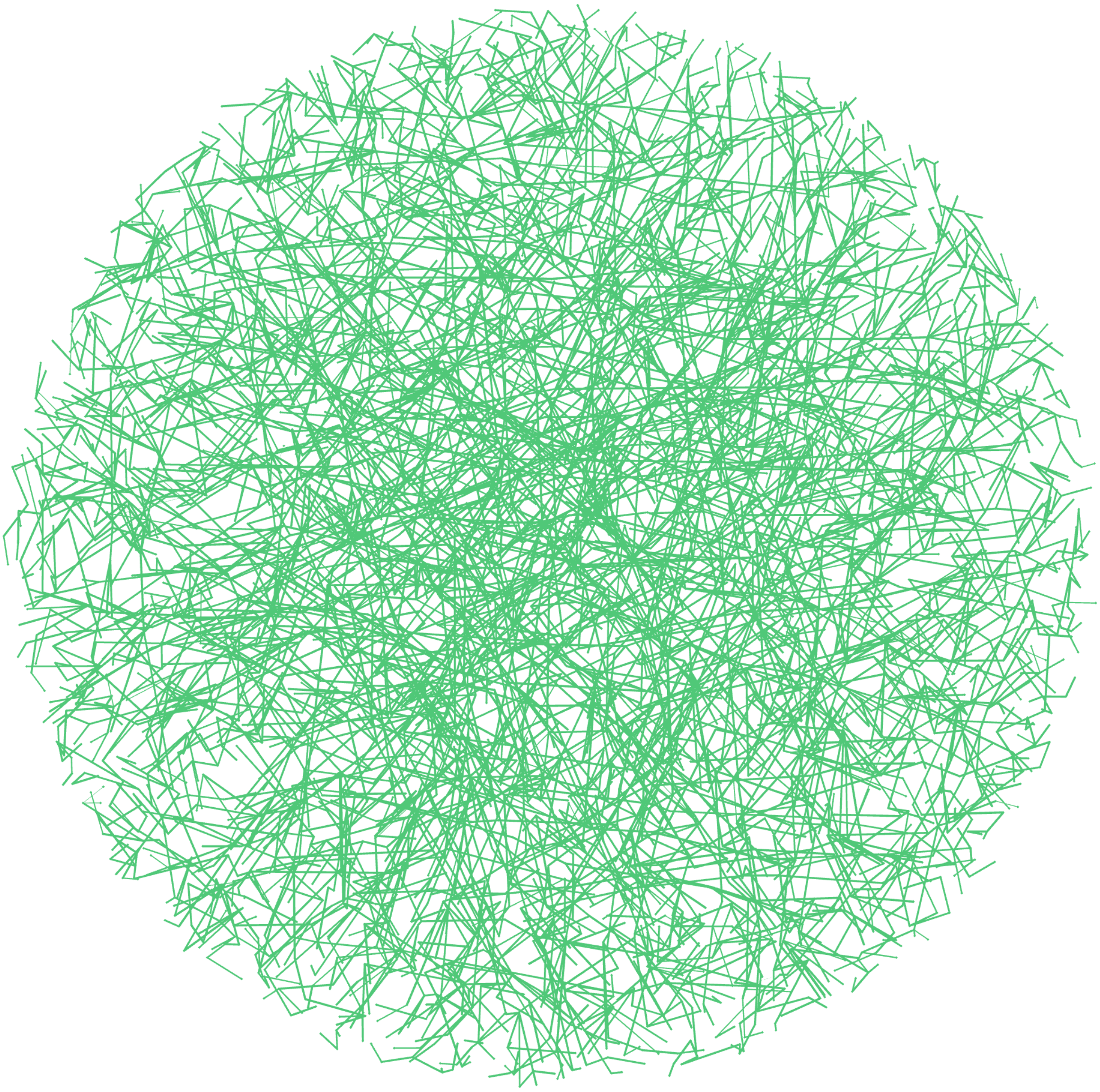


Figura 3.2: Tamaño 5000. Parámetros: Gravedad = 5.0 y Velocidad = 300.0

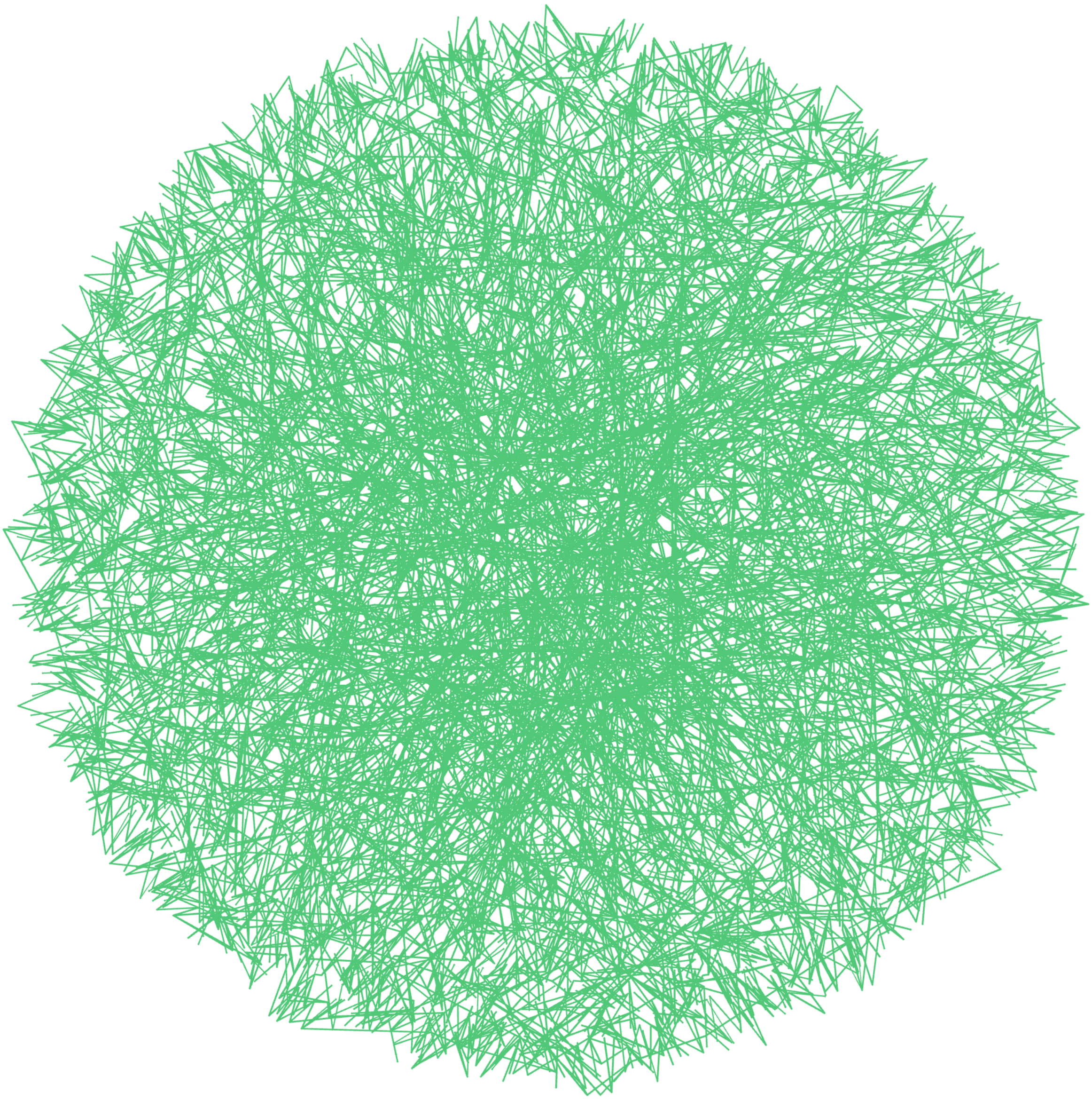


Figura 3.3: Tamaño 10000. Parámetros: Gravedad = 5.0 y Velocidad = 300.0

El resultado no es muy descriptivo. Vamos a probar ahora con Kamada Kawai:

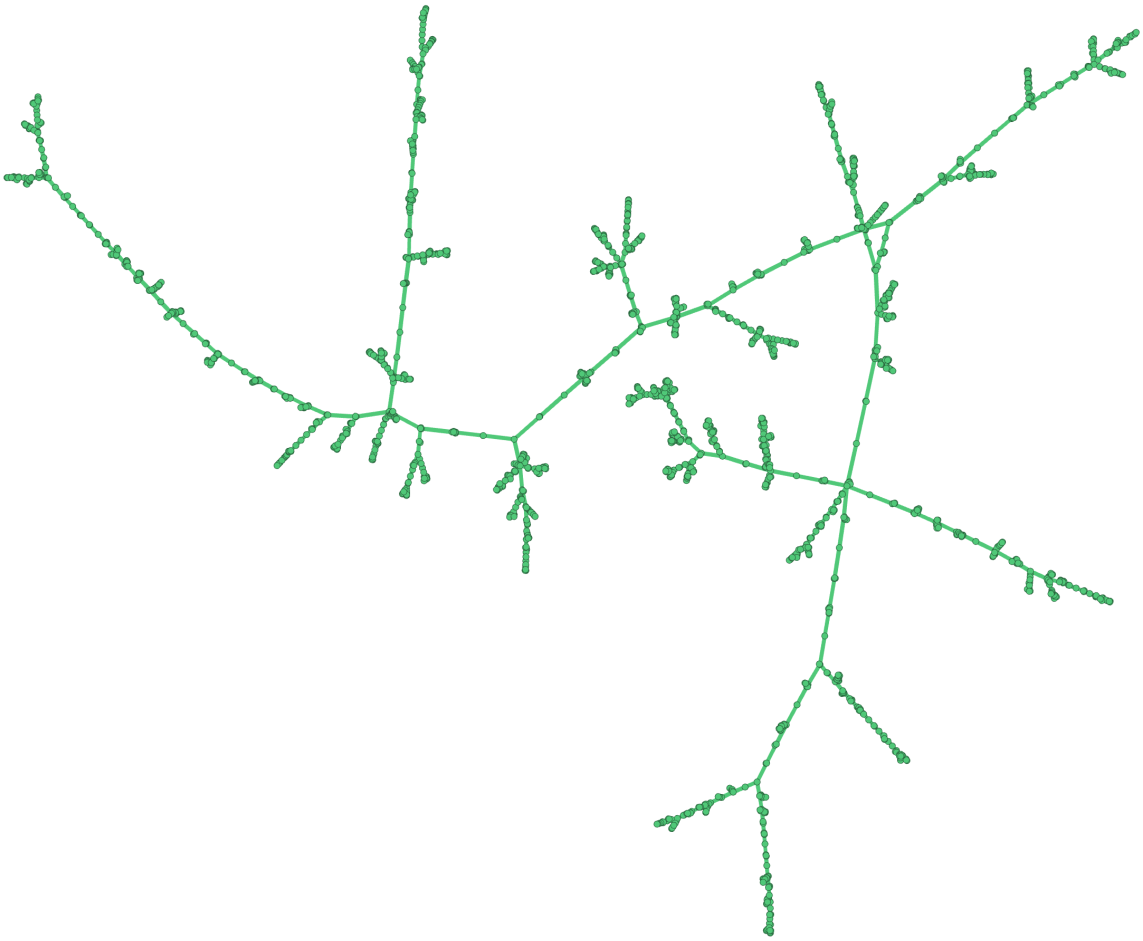


Figura 3.4: Tamaño 2000. Parámetros: Gravedad = 5.0

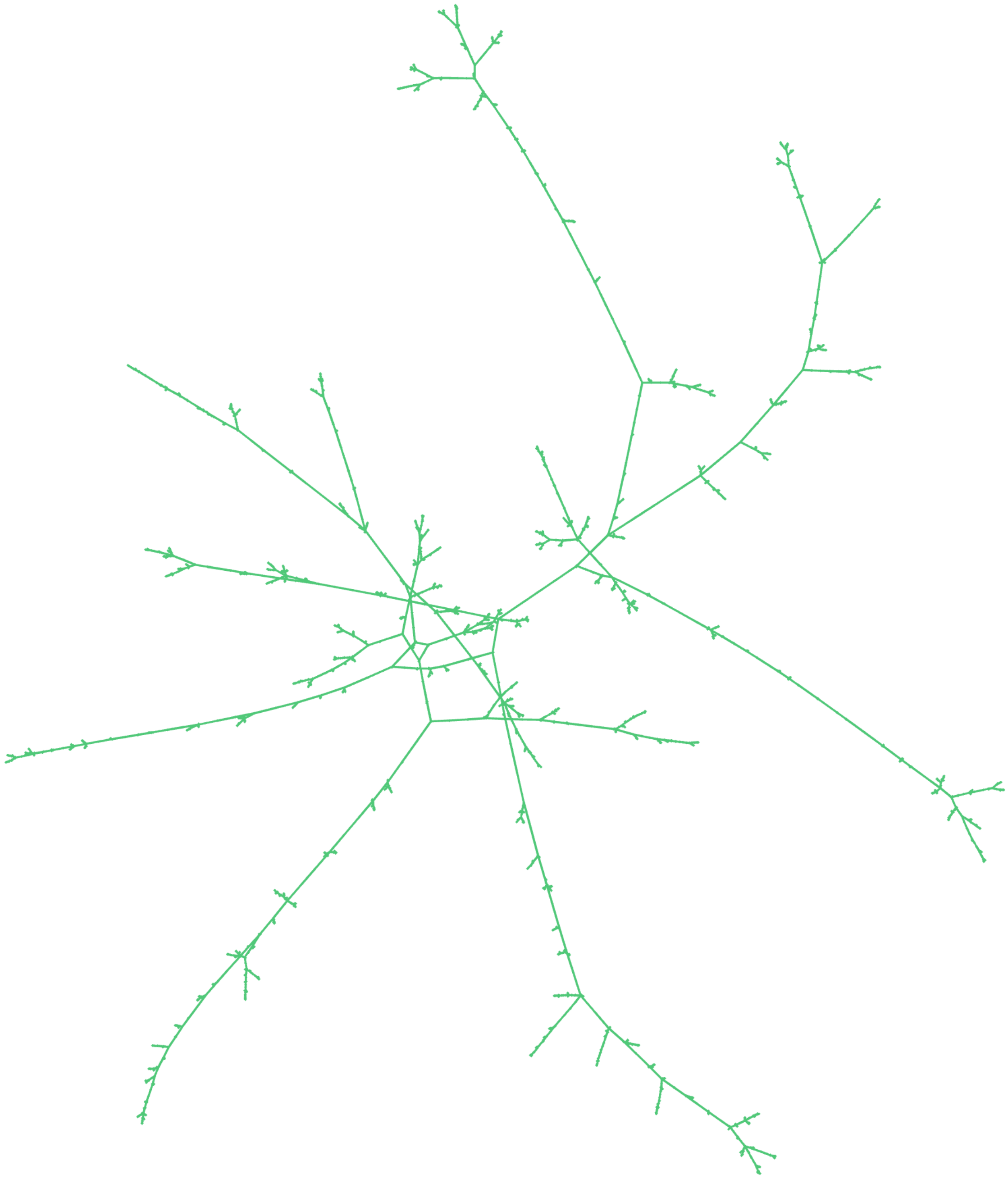


Figura 3.5: Tamaño 5000. Parámetros: Gravedad = 5.0

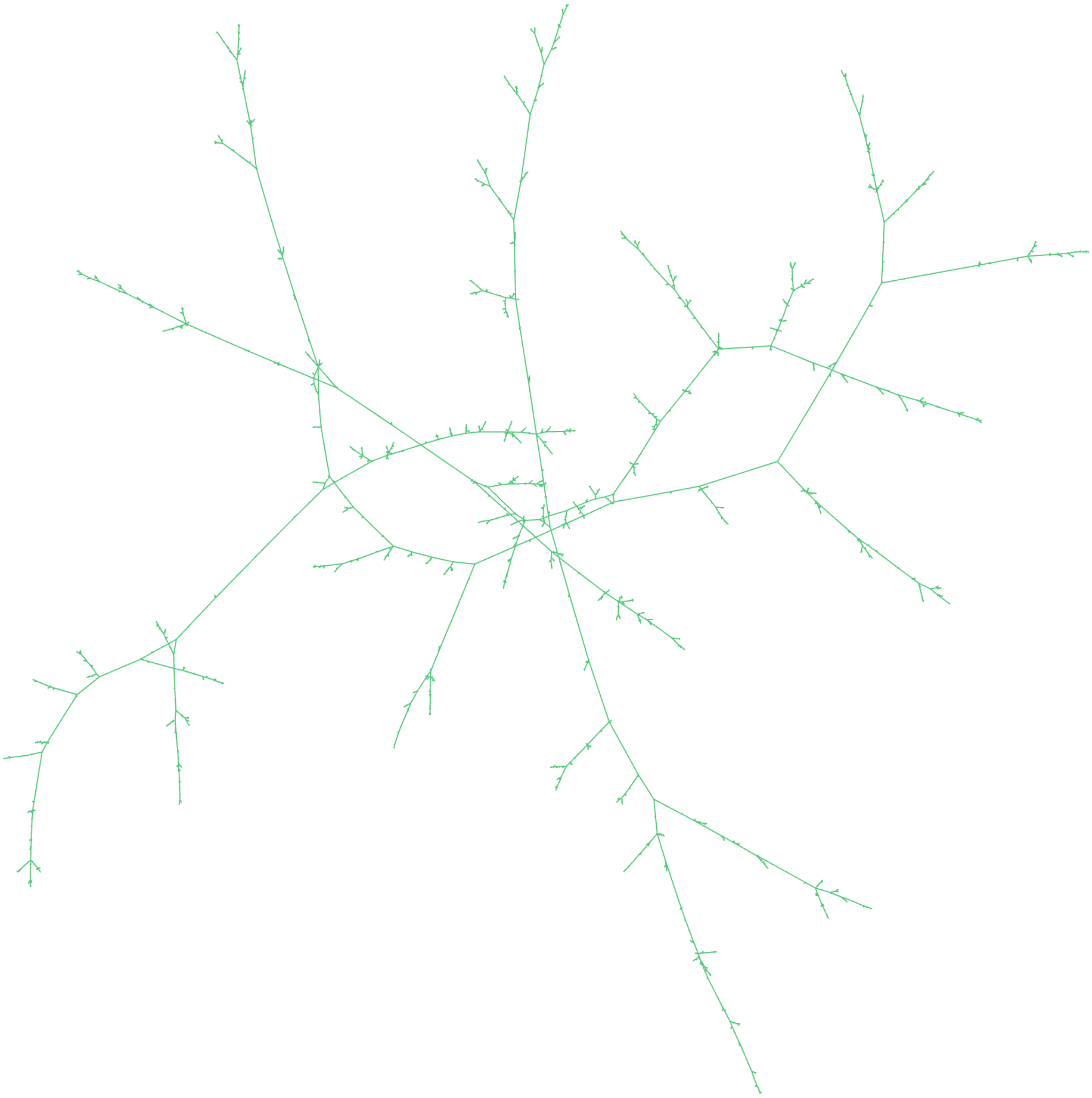
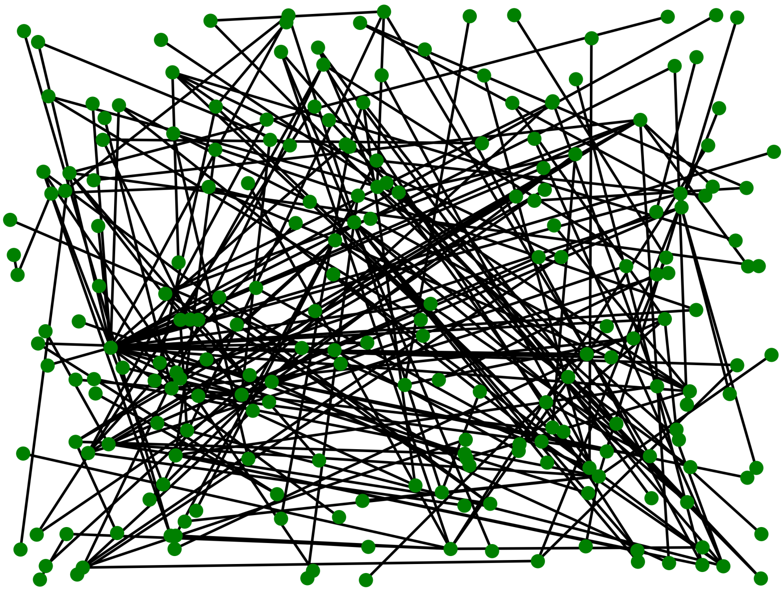


Figura 3.6: Tamaño 10000. Parámetros: Gravedad = 5.0

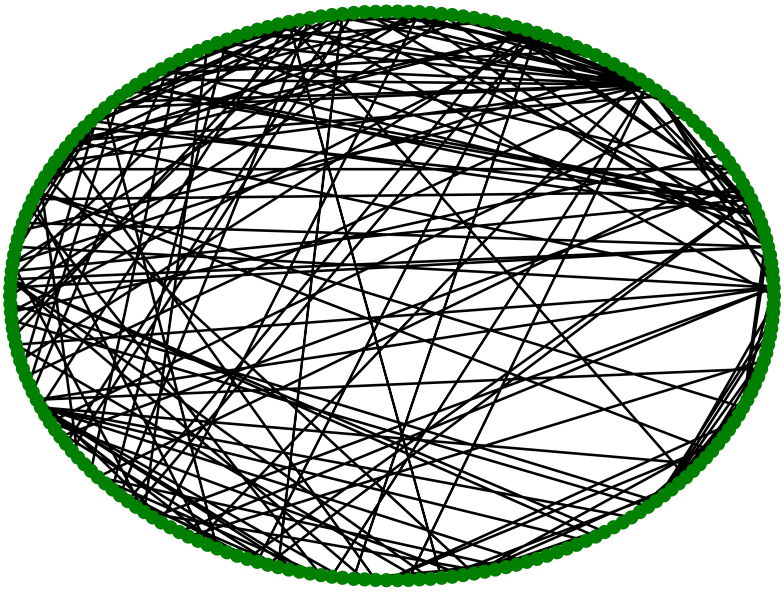
Como podemos ver, con Kamada Kawai si que obtenemos una representación más bonita y limpia. Esto nos hace ver que, a pesar de ser la misma red, la decisión de elegir un algoritmo de visualización u otro es bastante importante para obtener una mejor representación de la red que estamos analizando.

4. Extra.

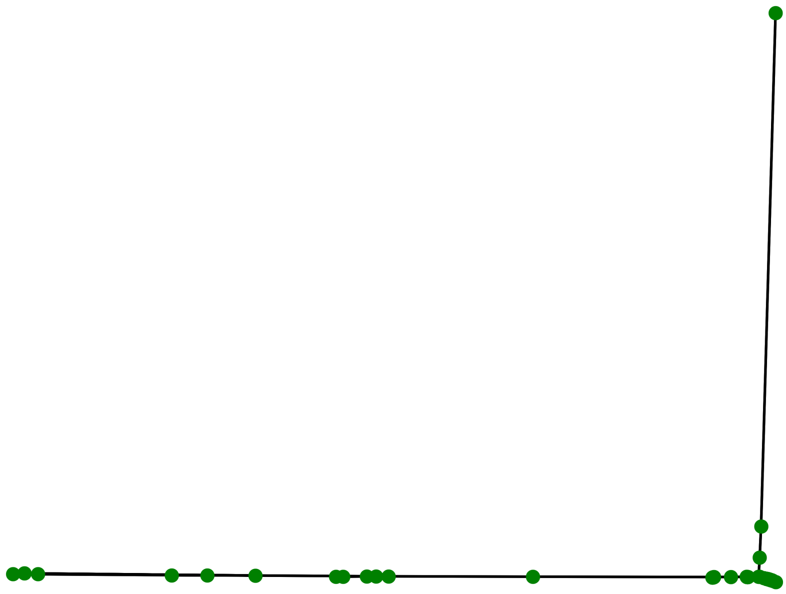
Para la parte extra voy a visualizar (con la biblioteca mencionada en la introducción) la red World podada con $q = n$ - *1* con otros algoritmos de visualización. El primero de ellos es uno aleatorio:



Luego vamos a ver un layout que lo que hace es poner los nodos en círculo (en elipse para ser más concretos). El resultado es el siguiente:



A continuación vamos a usar un layout que obtiene un resultado muy curioso. Se llama *spectral_layout*. Lo que hace es colocar los nodos usando los autovectores del grafo Laplaciano asociado. El resultado es bastante diferente a lo visto hasta ahora:



5. Bibliografía.

Como he comentado en la introducción, he usado la librería networkx para Python para agilizar el cómputo de datos sobre las redes.