QT 2020-2021

CONNECTIVITAT I PERCOLACIÓ

PROJECTE D'ALGORÍSMIA
FIB-UPC

Realitzat per:
Álvaro Mañoso alvaro.manoso@estudiantat.upc.edu
Edgar Pérez edgar.perez.baz@estudiantat.upc.edu
Ferran Villar ferran.villar.rodriguez@estudiantat.upc.edu

Estructura del projecte

El projecte està format per la classe **main** y les següents dos classes:

- **Grafo**: Aquesta classe conté totes les funcions necessàries per realitzar la creació i percolació dels 2 tipus de grafs tractats en aquest projecte:
  - Graf multidimensional de N nodes de costat
    - Amb percolació d'arestes
    - Amb percolació de nodes
  - Graf geomètric amb percolació d'arestes

  Y per suposat, les funcions necessàries per obtenir el numero de components connexes del graf.
- **Experimento**: Aquesta classe conté 2 funcions diferents per realitzar els experiments pertinents per cadascun dels grafs anteriorment mencionats.

Instruccions d'execució

Per tal de generar els arxius binaris seguim les següents indicacions:

- Ens dirigim al directori *src/* i executem la comanda **make**

  *" En el cas d'estar en un sistema linux, l'arxiu **Makefile** s'hauria de substituir per l'arxiu **MakefileLinux** "*

- També podem utilitzar la comanda **make clean** per esborrar els arxius binaris després de la compilació. Això ens serveix per detectar errors.

Per tal de poder realitzar experiments i rebre dades amb el programa executem la següent comanda al directori *compiled/*:

- **Windows:** sacarDatos.bat
- **Linux:** sacarDatos.sh

S'indicarà en tot moment com va el procés de generació i un cop acabat, les dades estaran al directori *datos/*.

En el cas de voler generar qualsevol graf manualment només caldria executar el binari *main.x* de la carpeta *compiled/* i aquest ens indicaria com fer-ho

Format de les dades

Per cada experiment es treuen 100 mostres per cada tipus de graf, probabilitat de fallida, mida, i si es el cas també distància mínima de connexió.

Les dades es generen al directori *datos/* en format csv, si s'han seguit les indicacions anteriors.

Hem escollit aquest format perquè les plataformes com Google Drive o Excel puguin representar en forma de full de càlcul les dades obtingudes.

Els arxius resultants contenen la següent informació:

Per valors de N de 10, 25, 50 i 100. (Dos dimensions)

- Grafs de NxN nodes amb percolació d'arestes
- Grafs de NxN nodes amb percolació de nodes

Per valors de N de 5, 10, 25, 50 i 100. (Dos dimensions)

- Grafs geomètrics de N nodes amb percolació d'arestes.

Per valors de N de 10 i 25. (Tres dimensions)

- Grafs de NxNxN nodes amb percolació d'arestes
- Grafs de NxNxN nodes amb percolació de nodes

Per valors de N de 10 i 25. (Quatre dimensions)

- Grafs de NxNxNxN nodes amb percolació d'arestes
- Grafs de NxNxNxN nodes amb percolació de nodes

Documentació

El codi realitzat per aquest projecte esta correctament documentat utilitzant Doxygen a la carpeta de *documentació/.* Ingressant al arxiu index.html podem visualitzar-la.

Cal mencionar que els algorismes estan pensats íntegrament per l'equip i la única font d'informació que hem considerat ha sigut la facilitada al enunciat del projecte, de la universitat de Princeton.

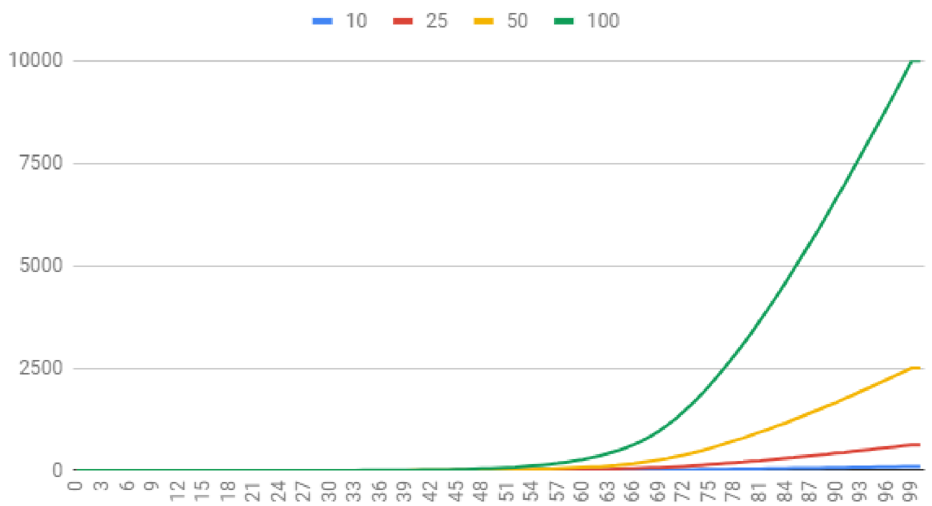
Anàlisi de les dades bidimensionals

Percolació per arestes en grafs en graella

Hem realitzat experiments per les següents mides de grafs: 10x10, 25x25, 50x50 i 100x100.

Com podem observar al gràfic, el creixement de les components connexes te un comportament diferent en funció de la mida del graf.

Quan més gran es el graf, la inclinació de la corba es més pronunciada però sempre partint d'un valor de l'índex de fallida proper al 0.5, o 50% com està expressat a la gràfica.



Components connexes en funció de la probabilitat de fallida d'una aresta en grafs de NxN nodes. Els colors identifiquen la N

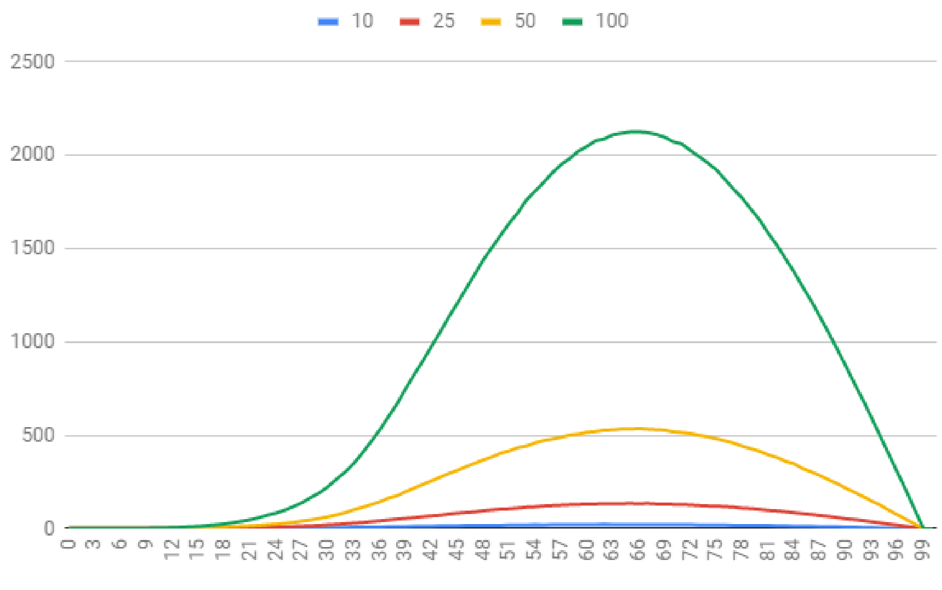
índex de fallida expressat en % (valors del 0 al 100)

Percolació per nodes en grafs en graella

Hem realitzat experiments per les següents mides de grafs: 10x10, 25x25, 50x50 i 100x100.

Com podem observar al gràfic, es produeix un creixement i decreixement de les components connexes d'una manera molt regular, succeint d'una manera més pronunciada als grafs més grans

D'aquest gràfic podem extreure, que el màxim de components connexes d'un graf amb percolació de nodes s'obté amb un índex de fallida calculat que es proper al 0.67, o 67% com està expressat a la gràfica.



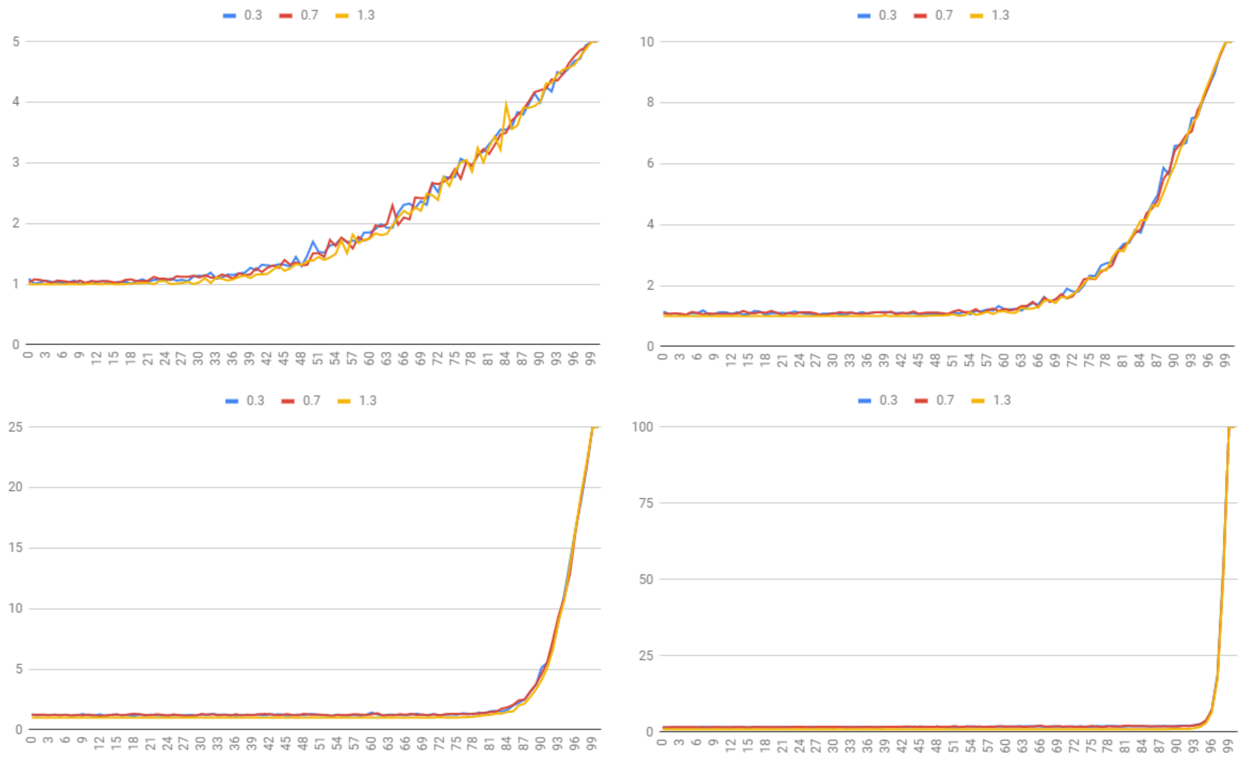
Components connexes en funció de la probabilitat de fallida d'un node en un graf de NxN nodes.
Els colors identifiquen la N

índex de fallida expressat en % (valors del 0 al 100)

Percolació per arestes en grafs geomètrics

Hem realitzat experiments per les següents mides de grafs: 5, 10, 25, 50 i 100 nodes. A continuació us mostrem 4 dels següents per ressaltar les diferències entre els més grans i més petits

Respecte a la distància mínima per que un node pugui connectar-se a un altre, aquesta sembla ser poc rellevant al procés de percolació. En canvi, a continuació comparem les mides 5, 10, 25 y 100 respectivament.



Podem observar que la corba de creixement de les components connexes podria tenir una transició de fase sempre i quan el graf sigui suficientment gran.

Això passa perquè els grafs amb més nodes tenen més **cohesió**. Que vol dir això? Significa que els nodes, al estar més junts, es connecten més entre ells i l'índex de fallida de les arestes ha de ser més gran per poder trencar les components connexes perquè hi ha més arestes per cada node.

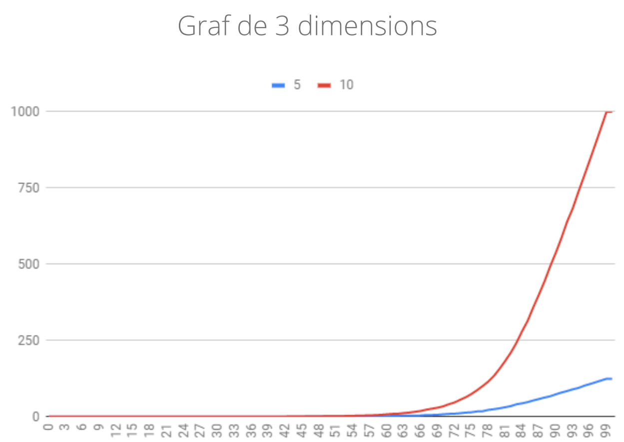
Anàlisi de les dades multidimensionals

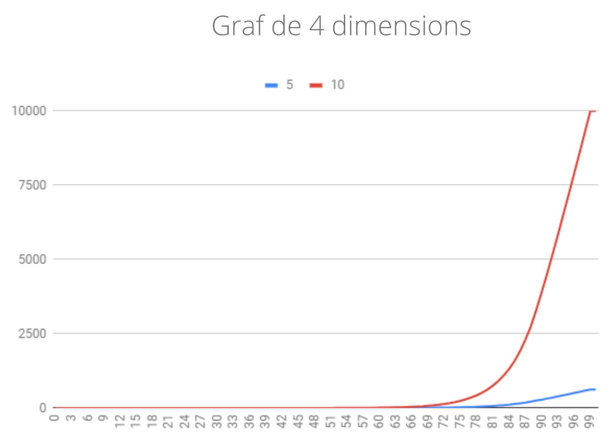
Percolació per arestes

Hem realitzat experiments per les següents mides de grafs:

- Tridimensionals de 5, 10 i 25 de costat.
- Quadridimensionals de 5 i 10 de costat.

A continuació la representació de les dades obtingudes pels mateixos nodes de costat.





Components connexes en funció de la probabilitat de fallida d'una aresta.
Els colors identifiquen la N

índex de fallida expressat en % (valors del 0 al 100)

Com podem observar, el comportament es molt similar al dels grafs bidimensionals amb una petita diferencia, la velocitat de creixement de la corba.

Hem observat que quantes mes dimensions té un graf, el creixement de les components connexes a mesura de l'índex de fallida es més elevat i s'enredereix cada cop més respecte l'índex de fallida passant de començar a créixer a partir del 0.5 en els bidimensionals fins començar des del 0.57 i 0.66 en els de 3 i 4 dimensions respectivament.

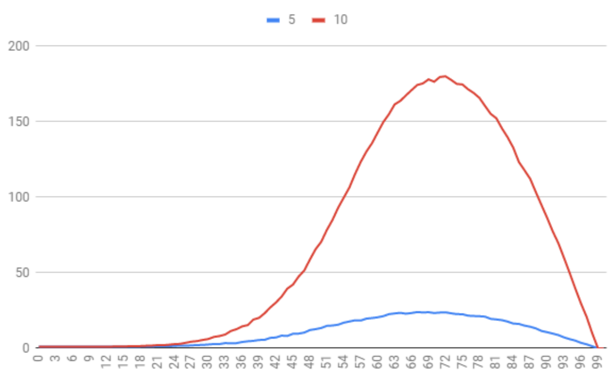
Percolació per nodes

Hem realitzat experiments per les següents mides de grafs:

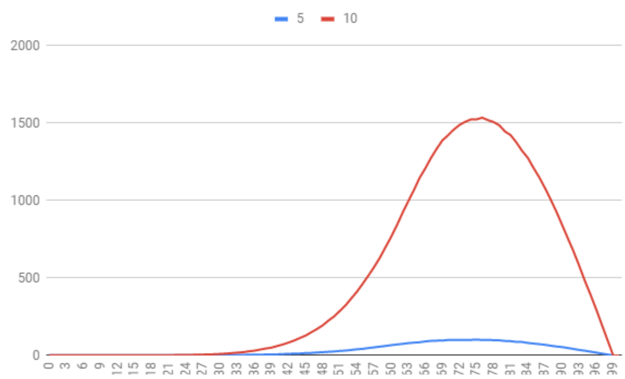
- Tridimensionals de 5, 10 i 25 de costat.
- Quadridimensionals de 5 i 10 de costat.

A continuació la representació de les dades obtingudes pels mateixos nodes de costat.

Graf de 3 dimensions



Graf de 4 dimensions



Components connexes en funció de la probabilitat de fallida d'una aresta.
Els colors identifiquen la N

índex de fallida expressat en % (valors del 0 al 100)

Com podem observar, el comportament es molt similar al dels grafs bidimensionals, però les transicions de fase calculades anteriorment s'enredereixen en augmentar l'índex de fallida per grafs de més dimensions, i la xifra calculada anteriorment pel nombre màxim de components connexes passa de 0.67 en els bidimensionals a 0.72 i 0.76 en els grafs de 3 i 4 dimensions respectivament.

Conclusió respecte la percolació de grafs en graella

Percolació per arestes

Hem observat que per aquest tipus de percolació podem identificar una possible transició de fase. Hem sigut capaços d'adonar-nos d'això en la nostra última part del projecte, l'augment de les dimensions.

Hem observat que aquest valor de l'índex de fallida pel que comença a créixer la corba, es cada cop més llunyà en augmentar les dimensions i podem deduir que en infinites dimensions aquest valor podria tenir tendència al 1.0.

Percolació per nodes

Pel que respecta a la percolació per nodes hem pogut identificar 2 possibles transicions de fase, on també, augmentar les dimensions del graf ens ha ajudat a adonar-nos d'aquest fenomen.

La primera sembla ser, igual que amb la percolació per arestes, el moment en el que la corba de components connexes comença a créixer. Y per altra banda la segona transició seria el valor en el que el graf assoleix el número màxim de components connexes.

Aquests dos valors sembla que tenen una tendència cap a valors cada cop més grans, però a diferència de la percolació per arestes aquests valors no podem assegurar que tenen tendència al 1.0 ja que no es una procés tan deductiu com la percolació per arestes. Per tant una major potencia computacional ens podria ajudar a seguir investigant amb dimensions més nombroses i així poder estudiar aquestes transicions de fase.

Conclusió respecte la percolació d'arestes en grafs geomètrics

Hem arribat a un important descobriment, sembla que els grafs geomètrics percolats per arestes pateixen una transició de fase que tendeix al 1.0 o 100% de l'índex de fallida en augmentar la mida del graf fins a l'infinit.

Ens hem adonat d'això gràcies a que en augmentar la mida del graf considerablement, la gràfica que interpretava les dades s'aproxima cada cop més a la forma d'una L inversa. Això vol dir que el punt on les components connexes es disparen es cada cop més llunya fins arribar al hipotètic cas d'ésser 1.0.

Cost dels algorismes

<table>
<tr><th></th><th>Creació</th><th>Percolació</th></tr>
<tr><td rowspan="2">Graella | Arestes i nodes</td><td>n</td><td rowspan="2">n log n</td></tr>
<tr><td>c^d</td></tr>
<tr><td>Geomètric | Arestes</td><td>n^2</td><td>n log n</td></tr>
</table>

On n identifica el número de nodes, c el número de nodes de costat i d les dimensions.
Per això el graf en graella es pot calcular de dos maneres ja que c^d = n

Què hem aprés amb aquest projecte?

Durant el procés hem desenvolupat diferents aptituds que no teníem anteriorment.
Ens ha sigut de molta ajuda saber formatar les dades en *.csv* per tal de que fossin visibles per programes d'anàlisi de dades en taules com Google Drive o Excel. En aquests serveis també hem aprés com filtrar, endreçar, tractar i representar dades de milers de fileres.

Pel que respecte al procés de programació dels algorismes, hem descobert l'algorisme UnionFind que ens ha permès tenir el compte, en tot moment, de les components connexes d'un graf.

Per últim també cap destacar que hem après a interpretar amb estructures de dades els diferents tipus de grafs que hem treballat.

Ha sigut una experiència on ens emportem coneixements que ens acompanyaran d'aquí en endavant per futurs projectes o reptes.