# Percolation and network resilience

Argimiro Arratia & Ramón Ferrer-i-Cancho

Universitat Politècnica de Catalunya

Complex and Social Networks (2020-2021)
Master in Innovation and Research in Informatics (MIRI)

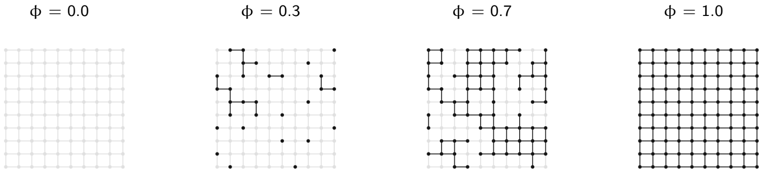
Official website: `www.cs.upc.edu/~csn/`

Contact:

- Ramon Ferrer-i-Cancho, rferrericancho@cs.upc.edu, `http://www.cs.upc.edu/~rferrericancho/`
- Argimiro Arratia, argimiro@cs.upc.edu, `http://www.cs.upc.edu/~argimiro/`

# Percolation: modeling random node or edge failures

## From Chapter 16 of [Newman, 2010]




- Site percolation:
  - With occupation probability $\phi$, keep nodes (black)
  - With probability $1 - \phi$, remove nodes (gray) and their incident edges
- Site percolation studies size of largest connected remaining component as $\phi$ changes (the giant cluster)
- Originally studied by physicists when networks are lattices

# In today's lecture

Uniform node removal

Non-uniform node removal

# Network resilience

## Uniform removal of nodes

If we remove nodes uniformly at random with probability $\phi$, will the remaining network still consist of a large connected cluster (aka "**the giant cluster**")?

If so, then we say that the network is resilient (or robust) to random removal of nodes

# Quantifying network resilience I

## Uniform removal of nodes in the configuration model

Consider a configuration model network with degree distribution $p_k$ and a percolation process in which vertices are present with occupation probability $\phi$

We'll use the *generating function* for the degree distribution

$$g_0(z) = \sum_{k=0}^{\infty} p_k z^k$$

Consider a node that has survived the random removal

- if it is to belong to the giant cluster, then at least one of its neighbors must belong to it as well

# Quantifying network resilience II

## Uniform removal of nodes in the configuration model

Let $u$ be the average probability that a vertex is not connected to the giant cluster via a specific neighbor

Then, for a vertex of degree $k$, the total probability of not being in the giant cluster is $u^k$

The average probability of not belonging to the giant cluster is $\sum_k p_k u^k = g_0(u)$

And so the average probability that a surviving node belongs to the giant cluster is $1 - g_0(u)$

Finally, the fraction of vertices (out of the original ones) that belong to the giant cluster is $S = \phi\ (1 - g_0(u))$

# Quantifying network resilience III

## Uniform removal of nodes in the configuration model

Now we compute $u$, the probability that a given neighbor is not in the giant cluster

For a neighbor (let's call it $A$) not to be part of the giant cluster, two things can happen

- either $A$ has been removed (w.p. $1 - \phi$), or
- $A$ is present (w.p. $\phi$), but none of $A$'s other neighbors are part of it (w.p. $u^l$ assuming $A$ has $l$ *other* neighbors)

# Quantifying network resilience IV

## Uniform removal of nodes in the configuration model

So, total probability of $A$ not being in the giant cluster is

$$1 - \phi + \phi \ u^l$$

The number of $A$'s other neighbors is distributed according to the excess degree distribution

$$q_l = \frac{(l+1)p_{l+1}}{\langle k \rangle}$$

where $\langle k \rangle$ is the average degree of the original network

# [An aside: excess degree distribution]

We want to compute the probability that by following an edge we reach a node of degree $l$.

Notice this is *different* from the degree distribution $p_l$

The probability of *reaching* a node of degree $l$ by following any edge is

$$\frac{\text{stubs adjacent to nodes of deg } l}{\text{stubs remaining}} = \frac{n\ p_l\ l}{2m-1} \approx \frac{n\ p_l\ l}{2m} = \frac{l\ p_l}{\langle k \rangle}$$

where $\langle k \rangle = \sum_l l\ p_l$ is the average degree

# Quantifying network resilience V

Averaging over $q_l$, we arrive at:

$$
\begin{aligned}
u &= \sum_l q_l \left(1 - \phi + \phi u^l\right) \\
&= 1 \sum_l q_l - \phi \sum_l q_l + \phi \sum_l q_l u^l \\
&= 1 - \phi + \phi\, g_1(u)
\end{aligned}
$$

since $\sum_l q_l = 1$ and where

$$g_1(z) = \sum_k q_k z^k$$

# Quantifying network resilience VI

Not always possible to derive closed form solution for

$$S = \phi\ (1 - g_0(u)) \qquad u = 1 - \phi + \phi\ g_1(u)$$

Observations:

- $g_1(u) = \sum_k q_k u^k$ is a polynomial with non-negative coefficients
  - $g_1(u) \geqslant 0$ for all $u \geqslant 0$
  - all derivatives are non-negative as well
  - so in general it is an increasing function of $u$ curving upwards

# Quantifying network resilience VII

Solution of equation is $u$ such that

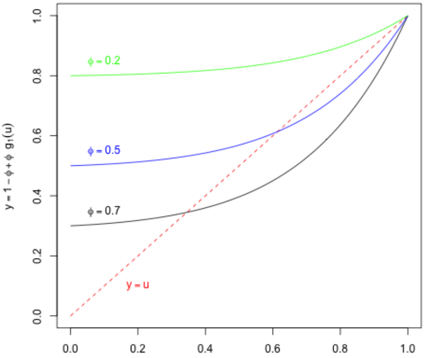
$$u = 1 - \phi + \phi \; g_1(u)$$

*(homework: check that $u = 1$ is always a solution for which $S = 0$)*

# Quantifying network resilience VIII

Depending on the value of $\phi$, two possibilities:

- $u = 1$ is the only solution (so no giant cluster), or
- there is another solution at $u < 1$ (and there is a giant cluster)



# Quantifying network resilience IX

## Uniform removal of nodes in the configuration model

**Another threshold phenomenon!**

The percolation threshold occurs at the critical value of $\phi$ s.t.

$$\left[\frac{d}{du}(1 - \phi + \phi\ g_1(u))\right]_{u=1} = 1$$

and so

$$\phi_c = \frac{1}{g_1'(1)} = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle}$$

- $g_1'(u) = \frac{d}{du}\sum_k q_k u^k = \sum_k kq_k u^{k-1} = \sum_k \frac{k(k+1)}{\langle k \rangle} p_{k+1} u^{k-1}$
- $g_1'(1) = \frac{1}{\langle k \rangle}\sum_k k(k+1)p_{k+1} = \frac{1}{\langle k \rangle}\sum_k (k-1)\ k\ p_k = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}$

# Quantifying network resilience X

## Uniform removal of nodes in the configuration model

The threshold $\phi_c = \frac{\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle}$ tells us the fraction of nodes that we must keep in order for a giant cluster to exist

So, if we want to make a network robust against random failures we'd want that $\phi_c$ is low, namely $\langle k^2 \rangle \gg \langle k \rangle$

# Uniform node removal

## Specific network types

### Erdös-Rényi networks

For large ER networks (with Poisson degree distribution) we have that $p_k = e^{-c}\frac{c^k}{k!}$ where $c$ is the mean degree, thus $\langle k \rangle = c$ and $\langle k^2 \rangle = c(c+1)$ and so $\phi_c = \frac{1}{c}$

So for large $c$ we will have networks that can withstand the loss of many of its vertices while keeping main connectivity

### Scale-free networks

For networks following a power-law degree distribution s.t. $2 \leqslant \alpha \leqslant 3$ we have that $\langle k \rangle$ is finite but $\langle k^2 \rangle$ diverges (in the limit). So, $\phi_c = 0$ in this case and it is very hard to break a scale-free network

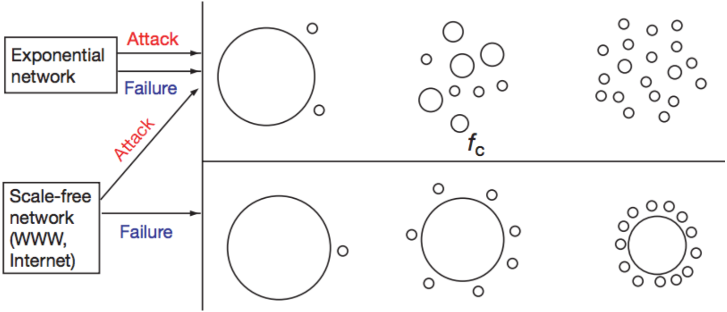
# In today's lecture

Uniform node removal

Non-uniform node removal

# Random vs. targeted attacks

From [Albert et al., 2000]




(By the way, giant cluster is not always good: think vaccination in the spread of an epidemic!)

# What if removal of nodes is not uniform?

## Targeted attack!

Now we generalize: let $\phi_k$ be the probability of occupation for nodes of degree $k$. Many possible scenarios:

- if $\phi_k = \phi$ for all $k$, then we recover the previous model
- if $\phi_k = 1$ for $k < 3$ and $\phi_k = 0$ for $k \geqslant 3$, then we remove all nodes of degree 3 and above

# Quantifying the size of the giant cluster I

## Targeted attack!

As before, the probability of a node of degree $k$ belonging to the giant cluster is $\phi_k(1-u^k)$, where $u$ is the average probability of not being connected to the giant cluster via a specific edge.

Now, we average over the degree probability distribution to find the average probability of being in the giant cluster

$$\begin{aligned} S &= \sum_k p_k \phi_k (1-u^k) = \sum_k p_k \phi_k - \sum_k p_k \phi_k u^k \\ &= f_0(1) - f_0(u) \end{aligned}$$

where

$$f_0(z) = \sum_{k=0}^{\infty} p_k \phi_k z^k$$

# Quantifying the size of the giant cluster II

## Targeted attack!

Notice that $f_0(z)$ is not normalized in the usual sense:

$$f_0(1) = \sum_k p_k \phi_k = \bar{\phi}$$

where $\bar{\phi}$ is the average probability that a node is occupied.

Now, the probability $u$ of not being part of the giant cluster via a particular neighbor can be computed as follows. Assume neighbor has excess degree $l$

- either the neighbor is not occupied (w.p. $1 - \phi_{l+1}$), or
- it is occupied (w.p. $\phi_{l+1}$) but it is not connected to the giant cluster (w.p. $u^l$)

# Quantifying the size of the giant cluster III

## Targeted attack!

So, adding these up: $1 - \phi_{l+1} + \phi_{l+1}\ u^l$

Now we average over the excess degree distribution $q_l$ to obtain value of $u$:

$$u = \sum_l q_l \left\{1 - \phi_{l+1} + \phi_{l+1}\ u^l\right\} = 1 - f_1(1) - f_1(u)$$

# Quantifying the size of the giant cluster IV

## Targeted attack!

where

$$
\begin{aligned}
f_1(z) &= \sum_{k \geqslant 0} q_k \phi_{k+1} z^k \\
&= \frac{1}{\langle k \rangle} \sum_{k \geqslant 0} (k+1) p_{k+1} \phi_{k+1} z^k \\
&= \frac{1}{\langle k \rangle} \sum_{k \geqslant 1} k p_k \phi_k z^{k-1}
\end{aligned}
$$

So, given $p_k, q_k$, and $\phi_k$, our solution is:

$$S = f_0(1) - f_0(u) \quad \text{for } u \text{ s.t.} \quad u = 1 - f_1(1) + f_1(u)$$

# Size of the giant cluster in a targeted attack I

## Special case: exponential networks

In an exponential network, $p_k = (1 - e^{-\lambda})e^{-\lambda k}$ for $\lambda > 0$

Suppose we remove vertices of degree greater than $k_0$, that is

$$\phi_k = \begin{cases} 1 & \text{if } k < k_0 \\ 0 & \text{otherwise} \end{cases}$$

Then

$$\begin{aligned} f_0(z) &= \sum_{k \geqslant 0} p_k \phi_k z^k = (1 - e^{-\lambda}) \sum_{k=0}^{k_0 - 1} e^{-\lambda k} z^k \\ &= (1 - e^{-\lambda k_0} z^{k_0}) \frac{e^{\lambda} - 1}{e^{\lambda} - z} \end{aligned}$$

# Size of the giant cluster in a targeted attack II

## Special case: exponential networks

where we have used: $\sum_{k=0}^{n} z^k = \frac{1-z^{n+1}}{1-z}$

Moreover,

$$
\begin{aligned}
f_1(z) &= \frac{f_0'(z)}{g_0'(1)} \\
&= \left[(1-e^{-\lambda k_0} z^{k_0}) - k_0 e^{-\lambda(k_0-1)} z^{k_0-1}(1-e^{-\lambda} z)\right]\left(\frac{e^{\lambda}-1}{e^{\lambda}-z}\right)^2
\end{aligned}
$$

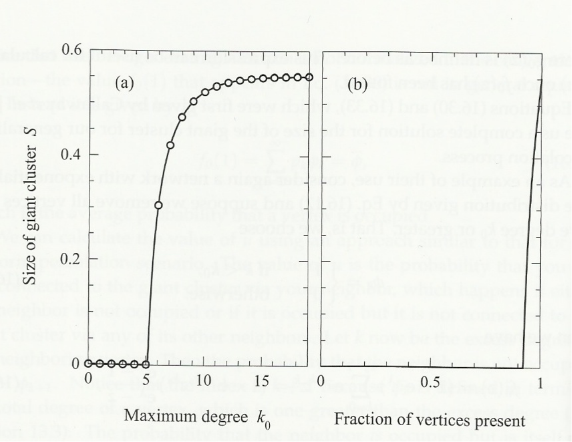
$f_1(z)$ is a polynomial on $z$, therefore

- to solve $u = 1 - f_1(1) + f_1(u)$
- we need to find $u^*$ s.t. $0 = 1 - u - f_1(1) + f_1(u)$,
- ie $u^*$ is a root of the polynomial $1 - u - f_1(1) + f_1(u)$

# Size of the giant cluster in a targeted attack III

## Special case: exponential networks

Knowing $0 \leqslant u^* \leqslant 1$ we can find the root numerically



# References I

Albert, R., Jeong, H., and Barabási, A.-L. (2000).
Error and attack tolerance of complex networks.
*Nature*, 406(6794):378–382.

Newman, M. (2010).
*Networks: An Introduction*.
Oxford University Press, USA, 2010 edition.