Too big to fail

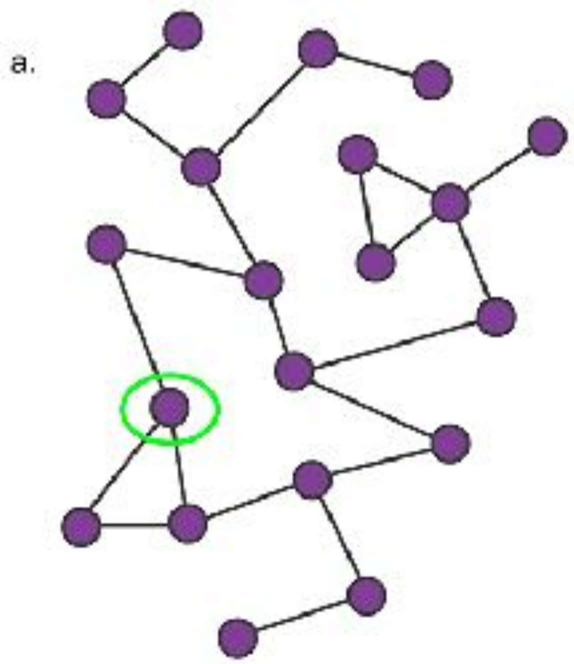
Learning outcomes

Robustness of different networks

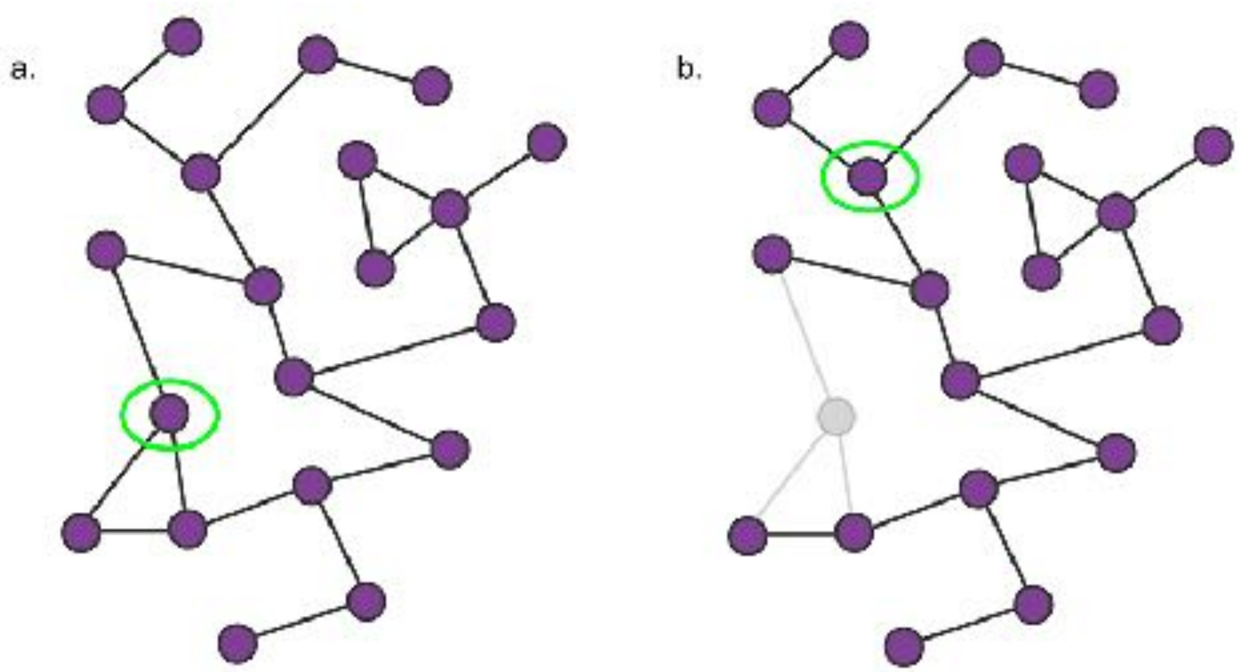
Financial networks and systemic risk

Overview of **Targeting** strategies

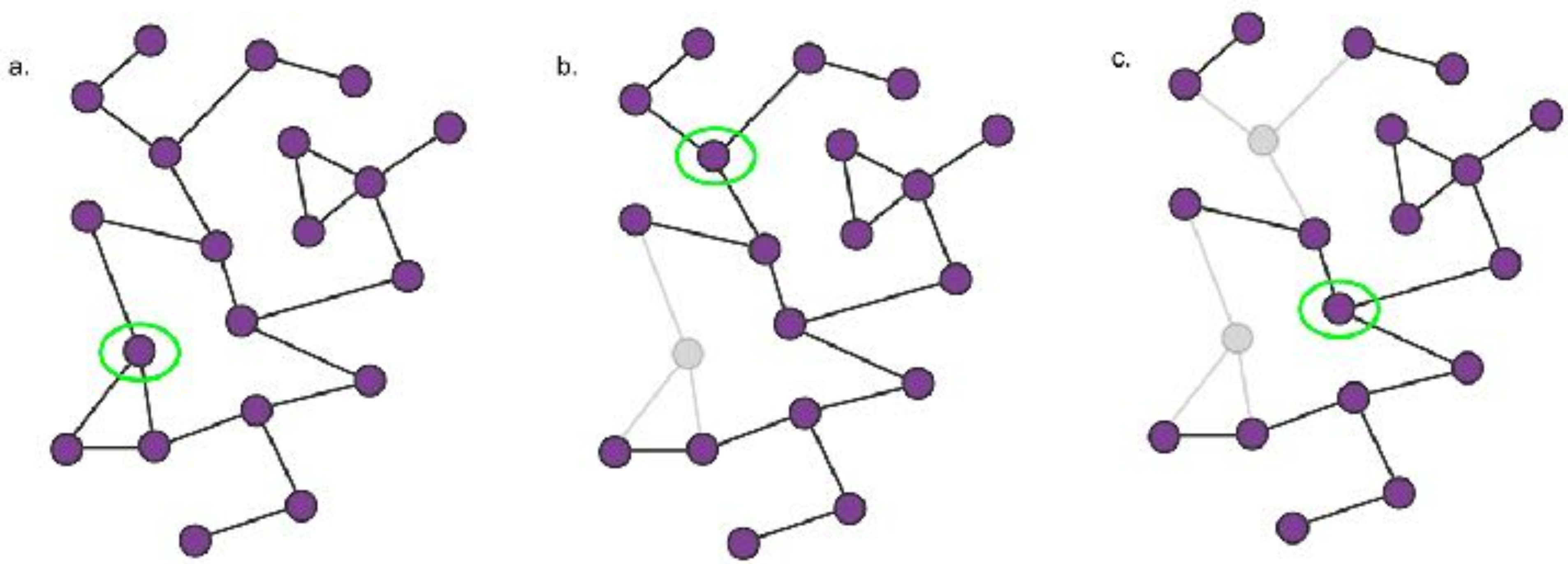
Percolation



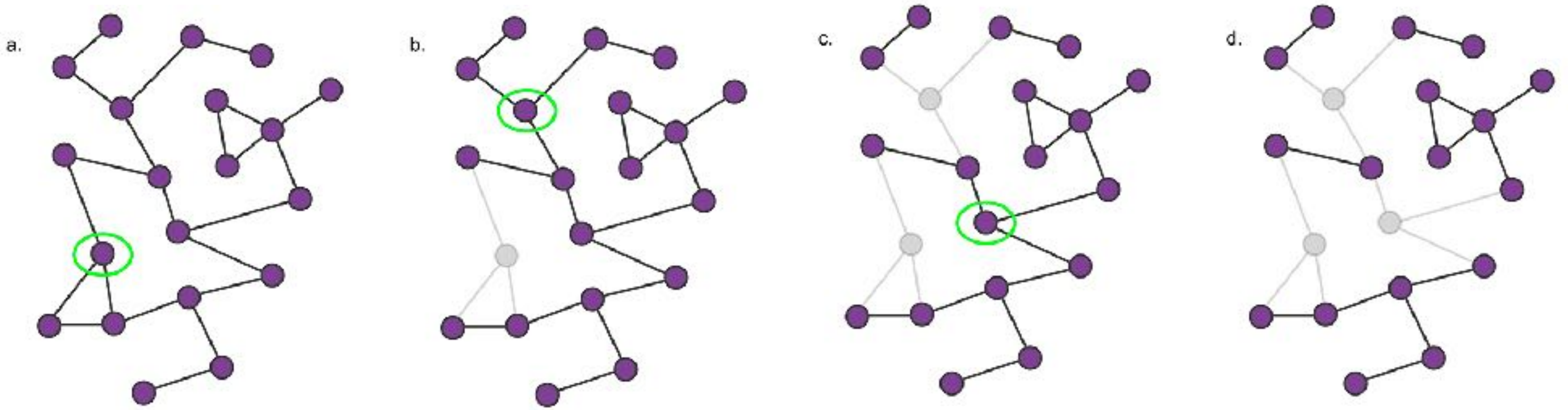
Percolation



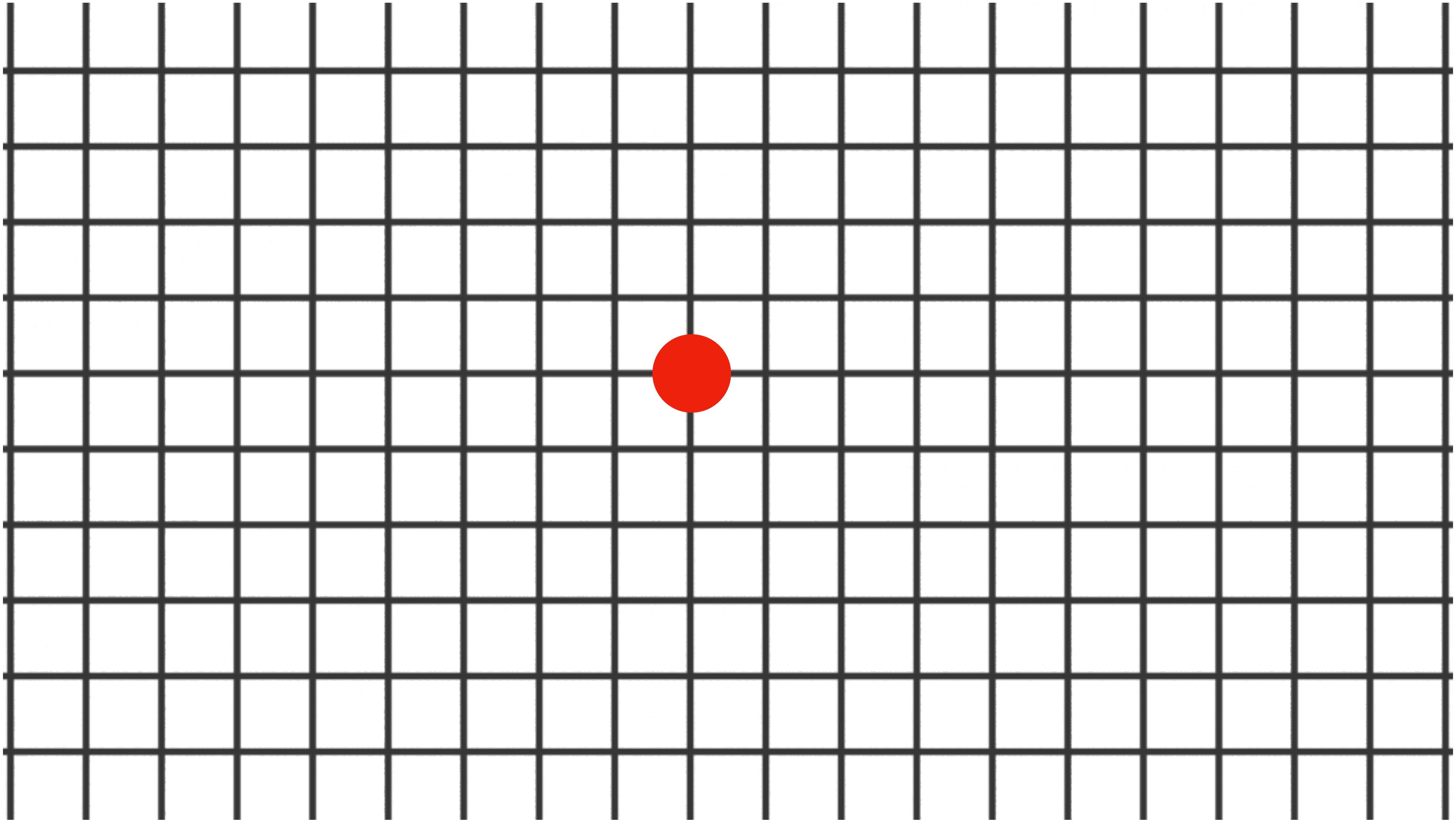
Percolation



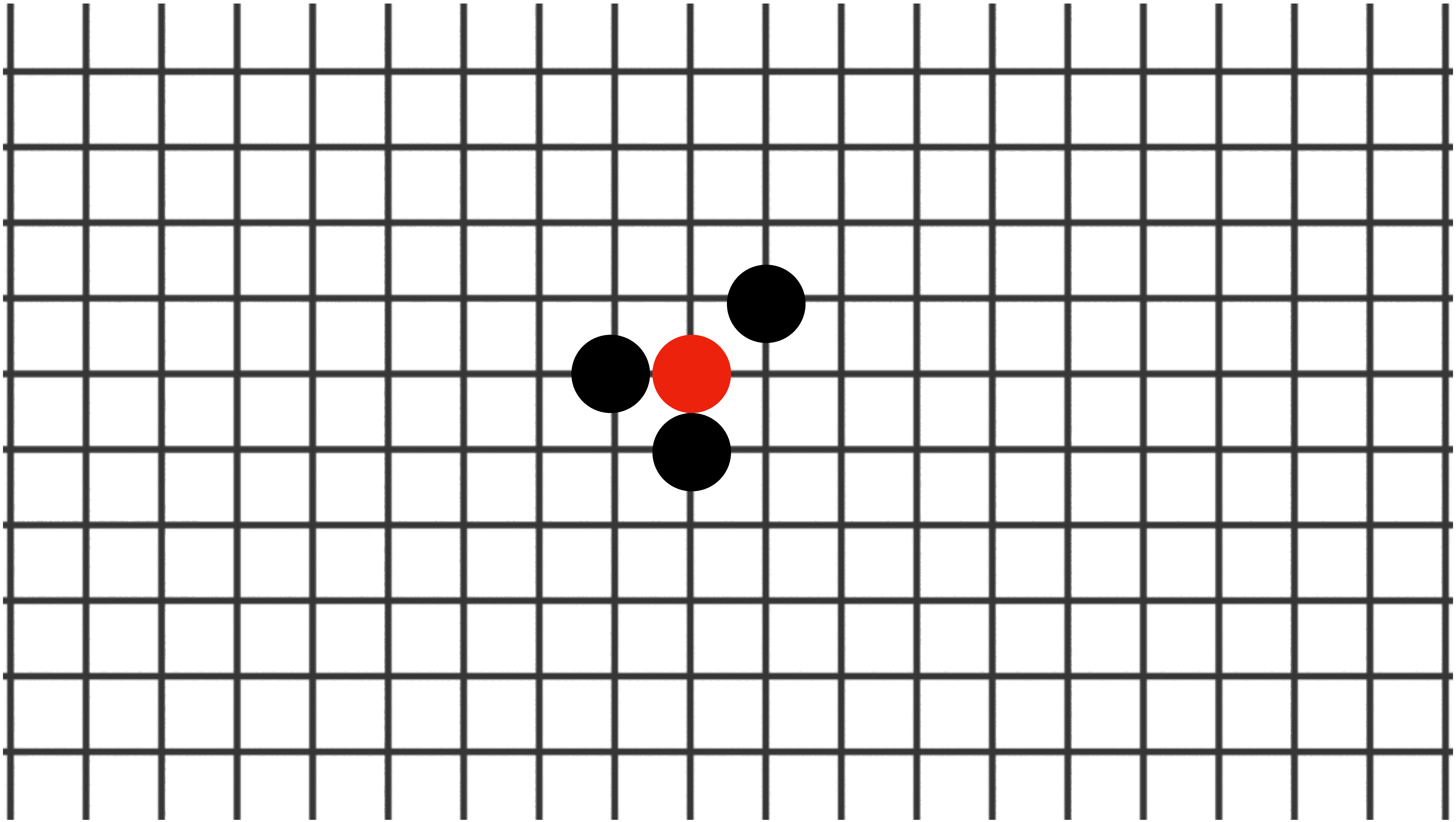
Percolation



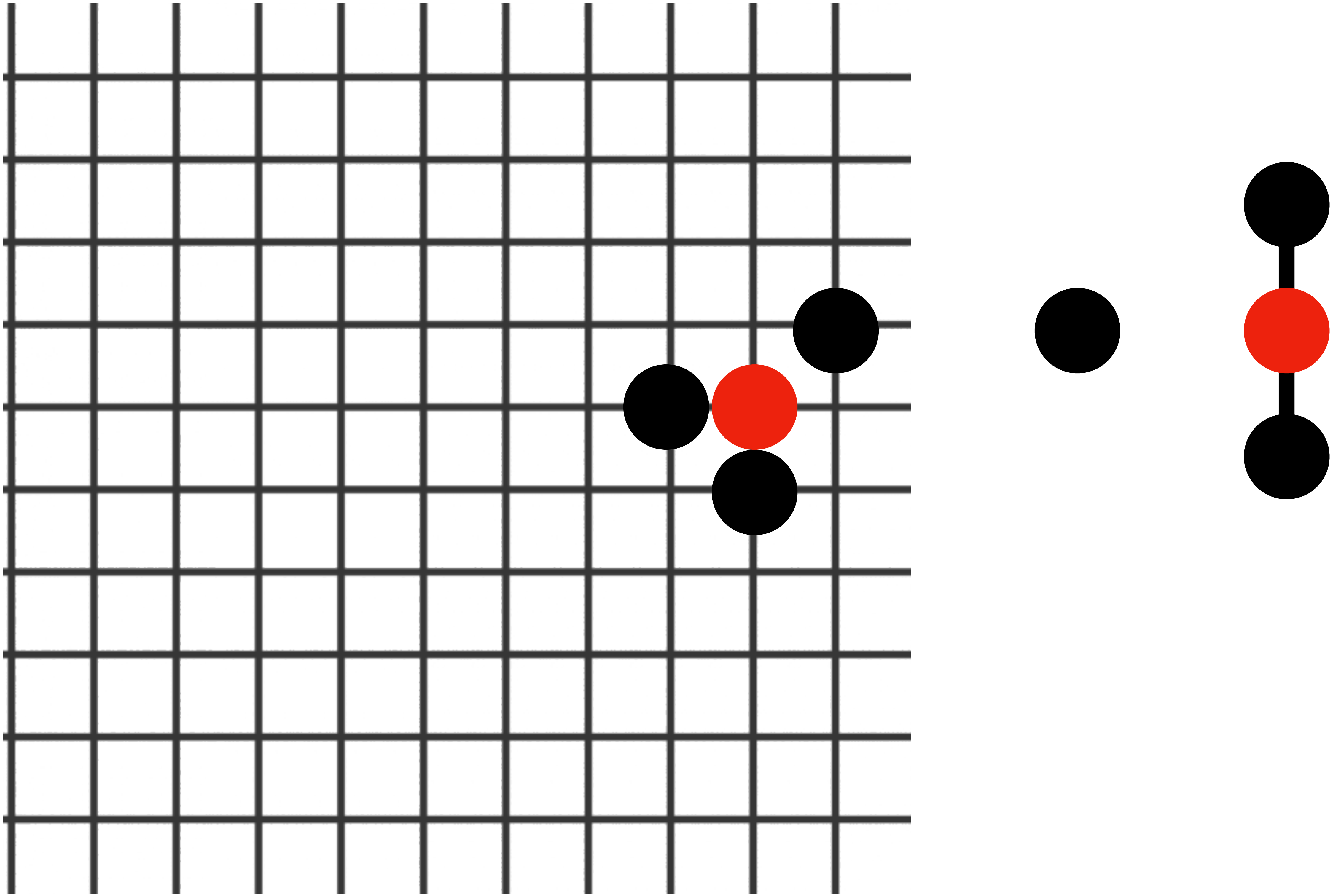
2d lattice



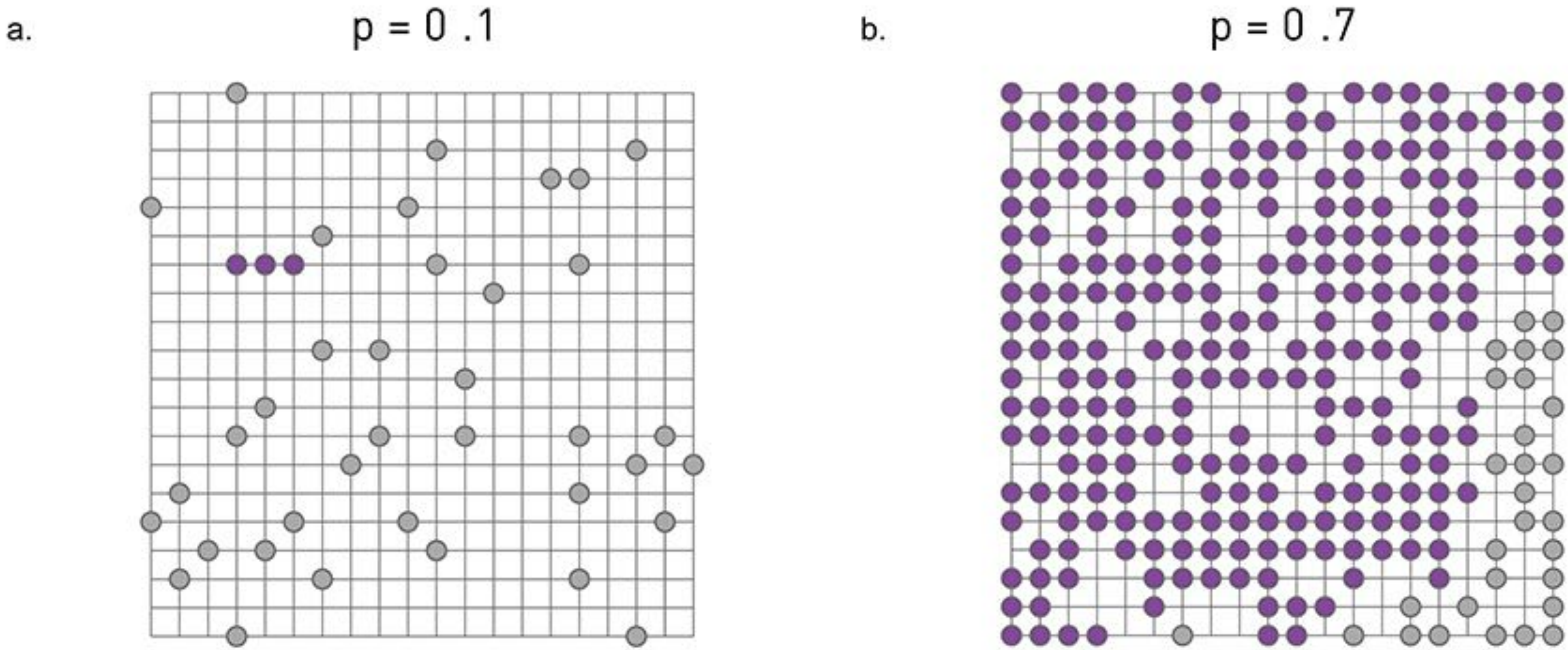
2d lattice



2d lattice



2d lattice



2d lattice

Average cluster size

$$\langle s \rangle \sim |p - p_c|^{-\gamma_p}$$

Order parameter

$$p_\infty \sim (p - p_c)^{\beta_p}$$

Correlation length

$$\xi \sim |p - p_c|^{-\nu}$$

p_c **Critical probability**

γ_p, β_p, ν **Critical exponents**

2d lattice

Average cluster size

$$\langle s \rangle \sim |p - p_c|^{-\gamma_p}$$

Order parameter

$$p_\infty \sim (p - p_c)^{\beta_p}$$

Correlation length

$$\xi \sim |p - p_c|^{-\nu}$$

Depends on lattice geometry

p_c **Critical probability**

γ_p, β_p, ν **Critical exponents**

Depend on lattice dimension (eg 2d, 3d) up to 6d

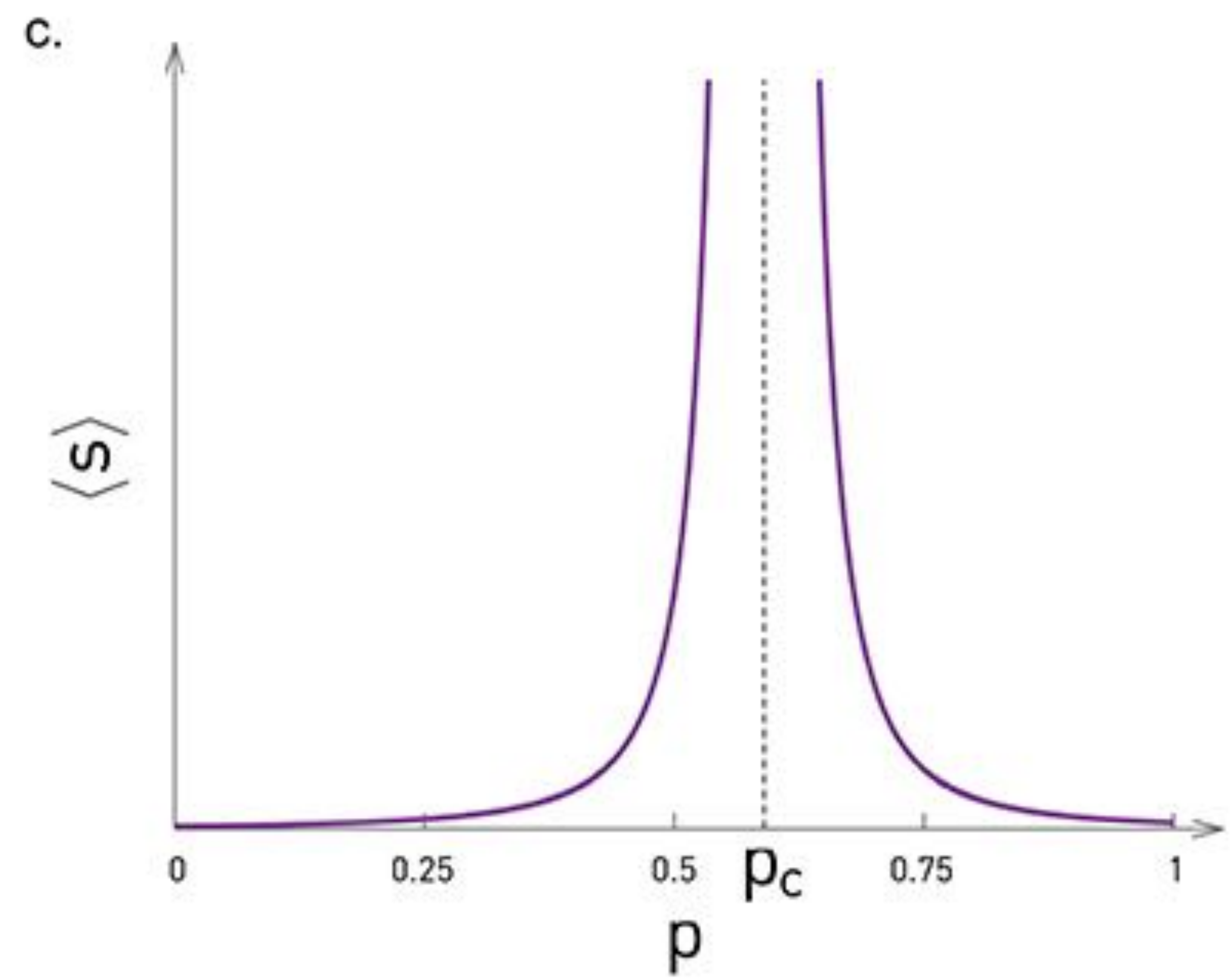
2d lattice

Average cluster size

$$\langle s \rangle \sim |p - p_c|^{-\gamma_p}$$

Correlation length

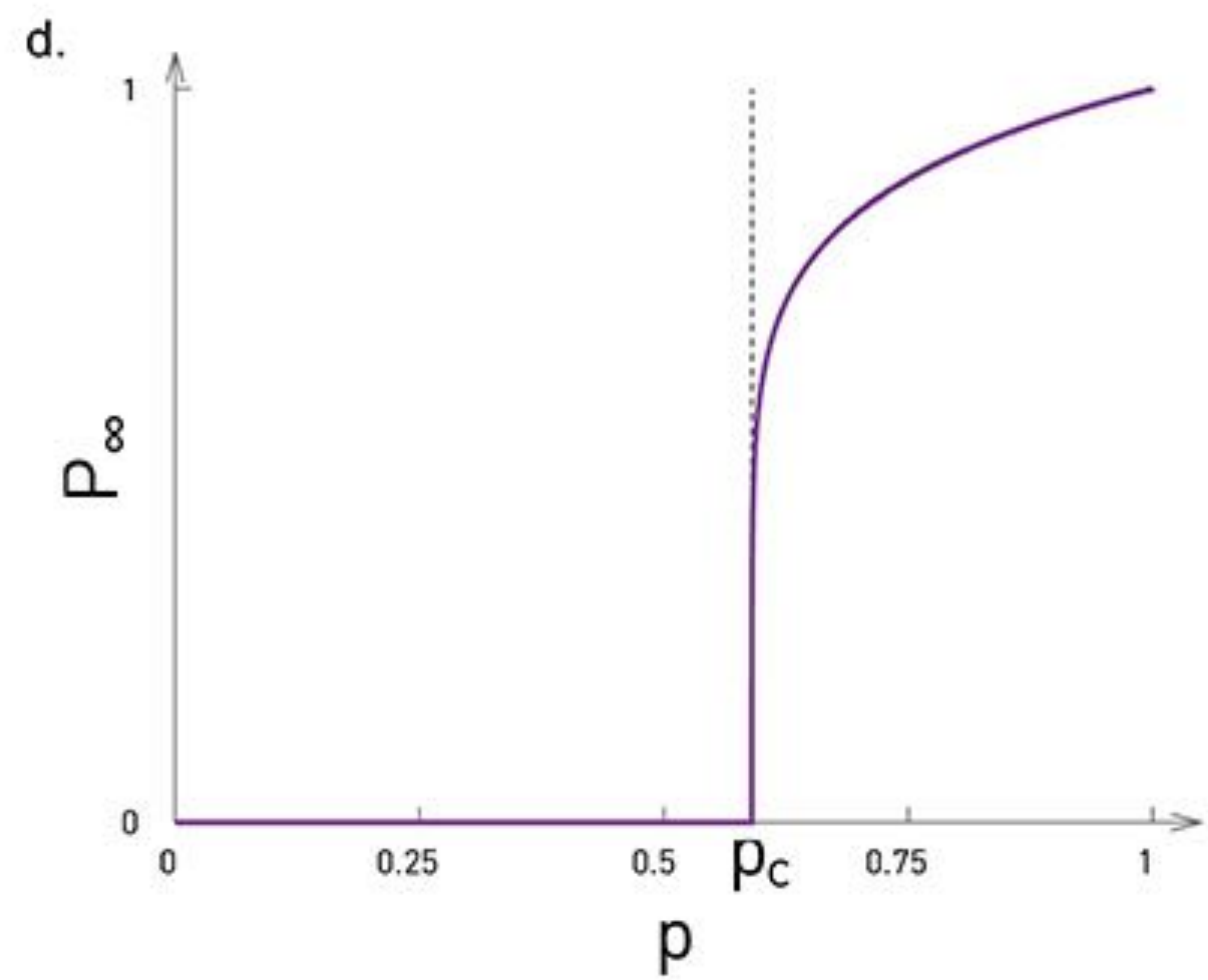
$$\xi \sim |p - p_c|^{-\nu}$$



2d lattice

Order parameter

$$p_\infty \sim (p - p_c)^{\beta_p}$$



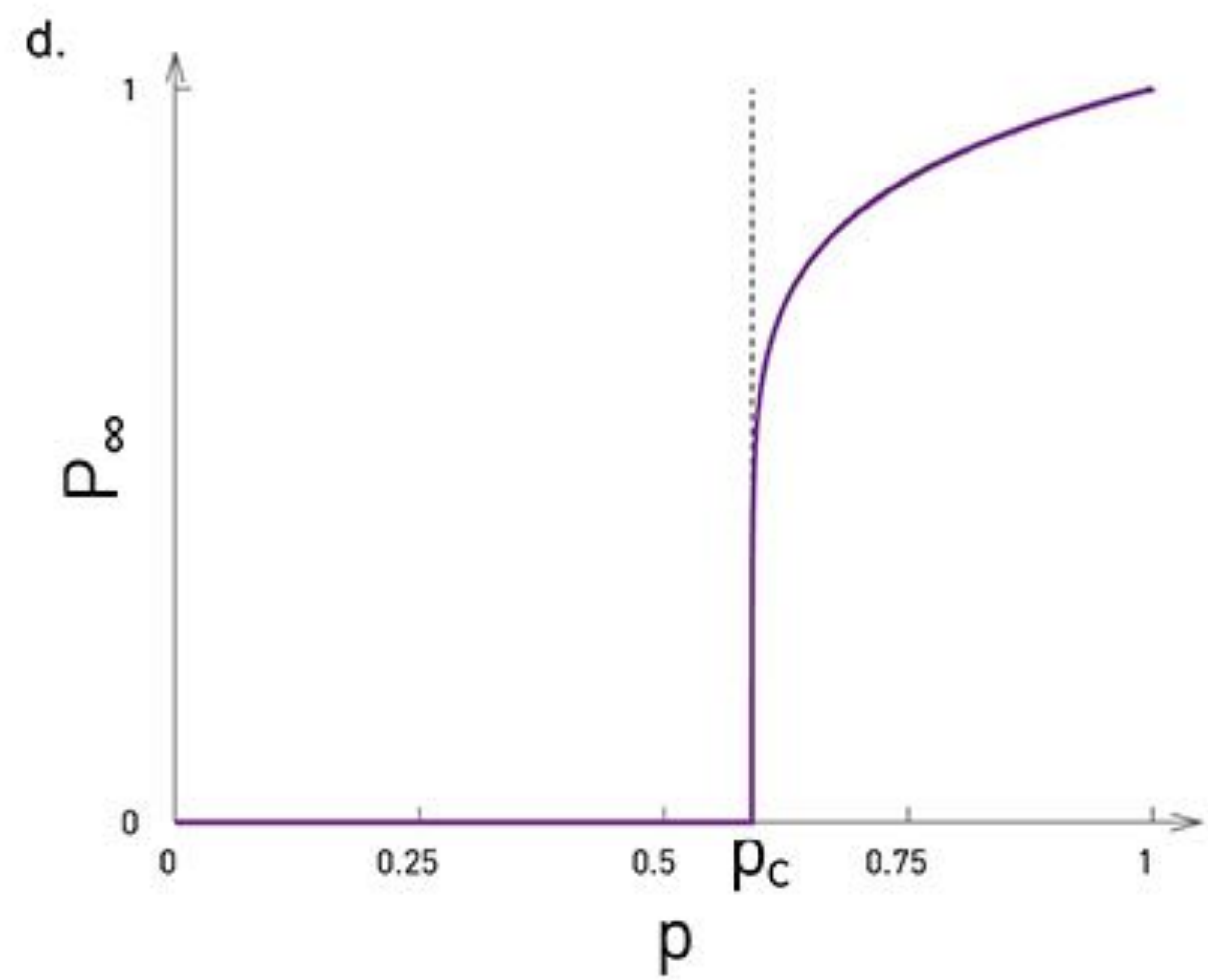

OMG Who The Hell Cares!

Systemic risk

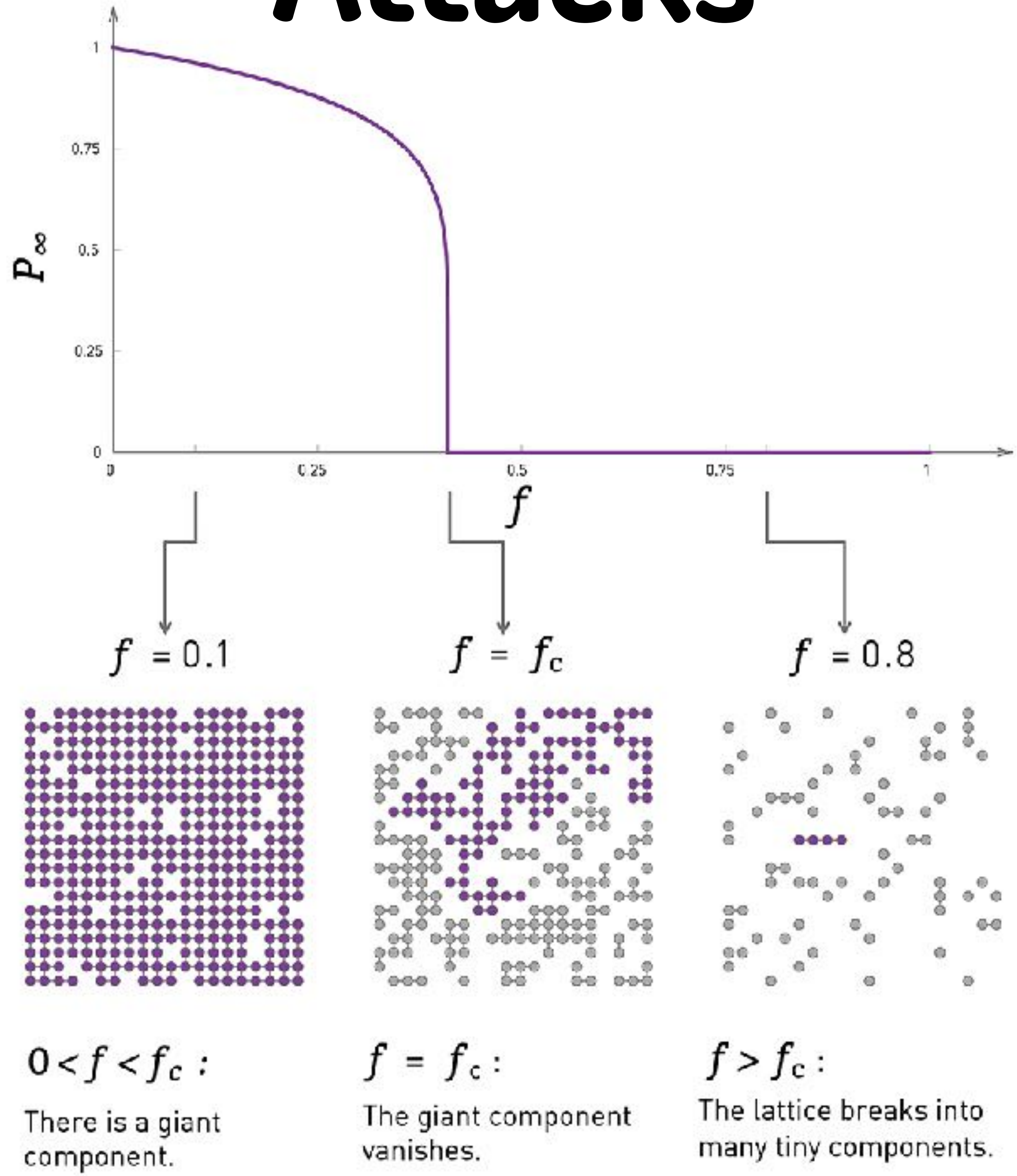
2d lattice

Order parameter

$$p_\infty \sim (p - p_c)^{\beta_p}$$



Attacks



1
0.75
0.5
0.25
0
P_∞
0
0.25
0.5
0.75
1
f
$f = 0.1$
$f = f_c$
$f = 0.8$
$0 < f < f_c$:
There is a giant component.
$f = f_c$:
The giant component vanishes.
$f > f_c$:
The lattice breaks into many tiny components.

Network structure comparison

What network do you think is more robust?

Network structure comparison

Scale-free networks are **more robust**

Most nodes have **low degrees**

Hubs are highly connected and central

Targeted removal

Robustness of different networks

Targeting strategies

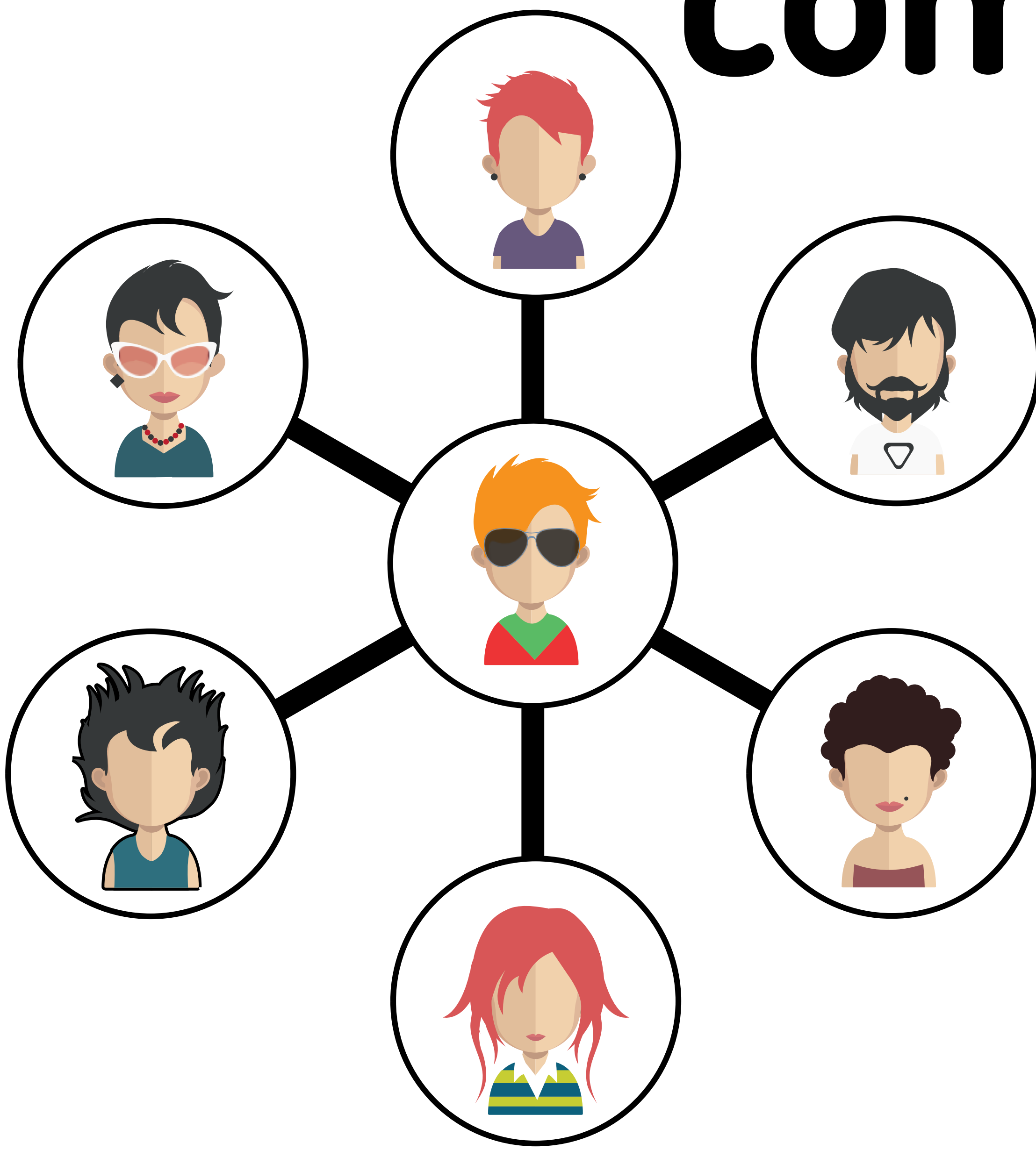
Financial networks and systemic risk

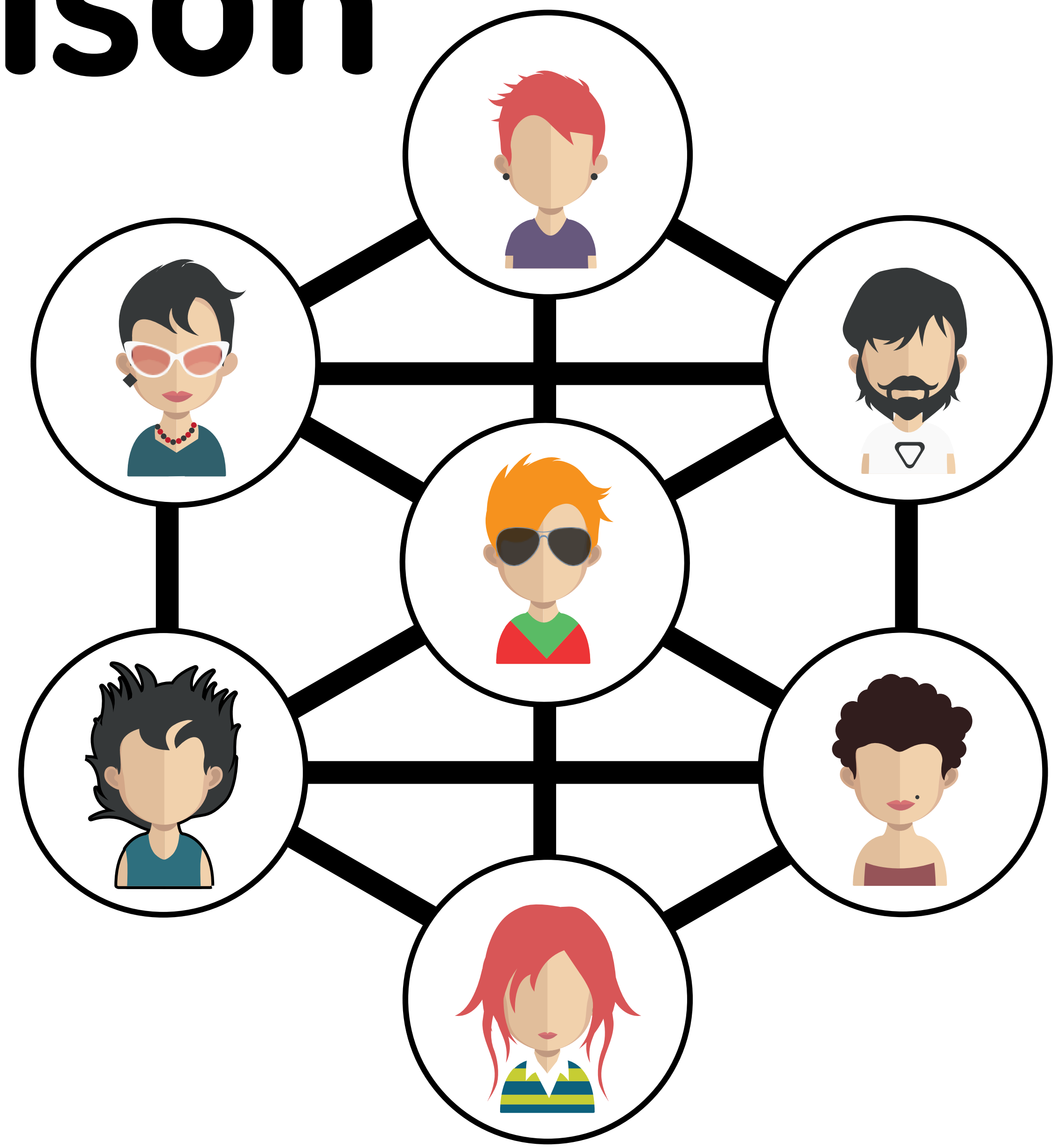
Targeted removal

If we consider **targeted attacks** everything changes!

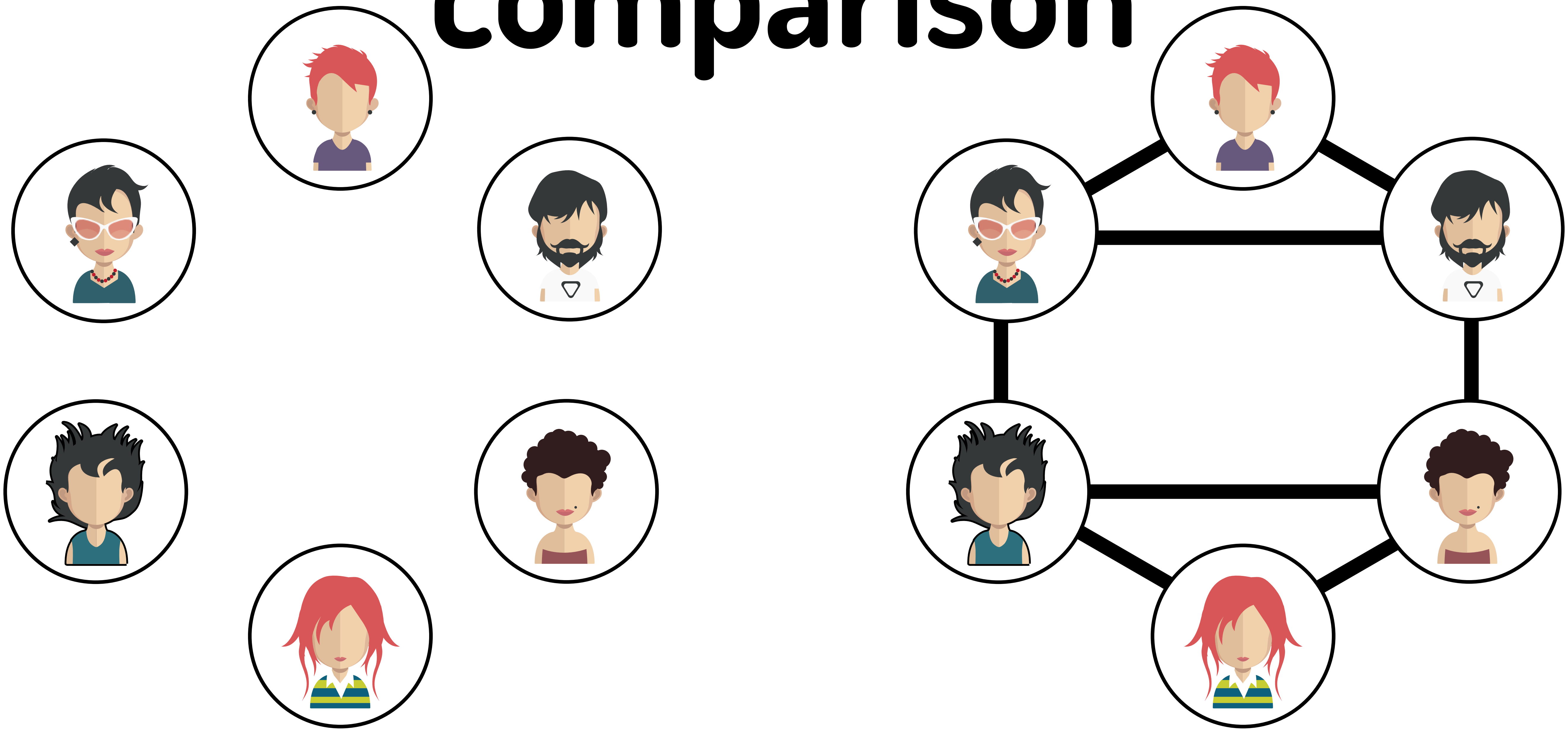
Hubs are highly connected and central

Network structure comparison



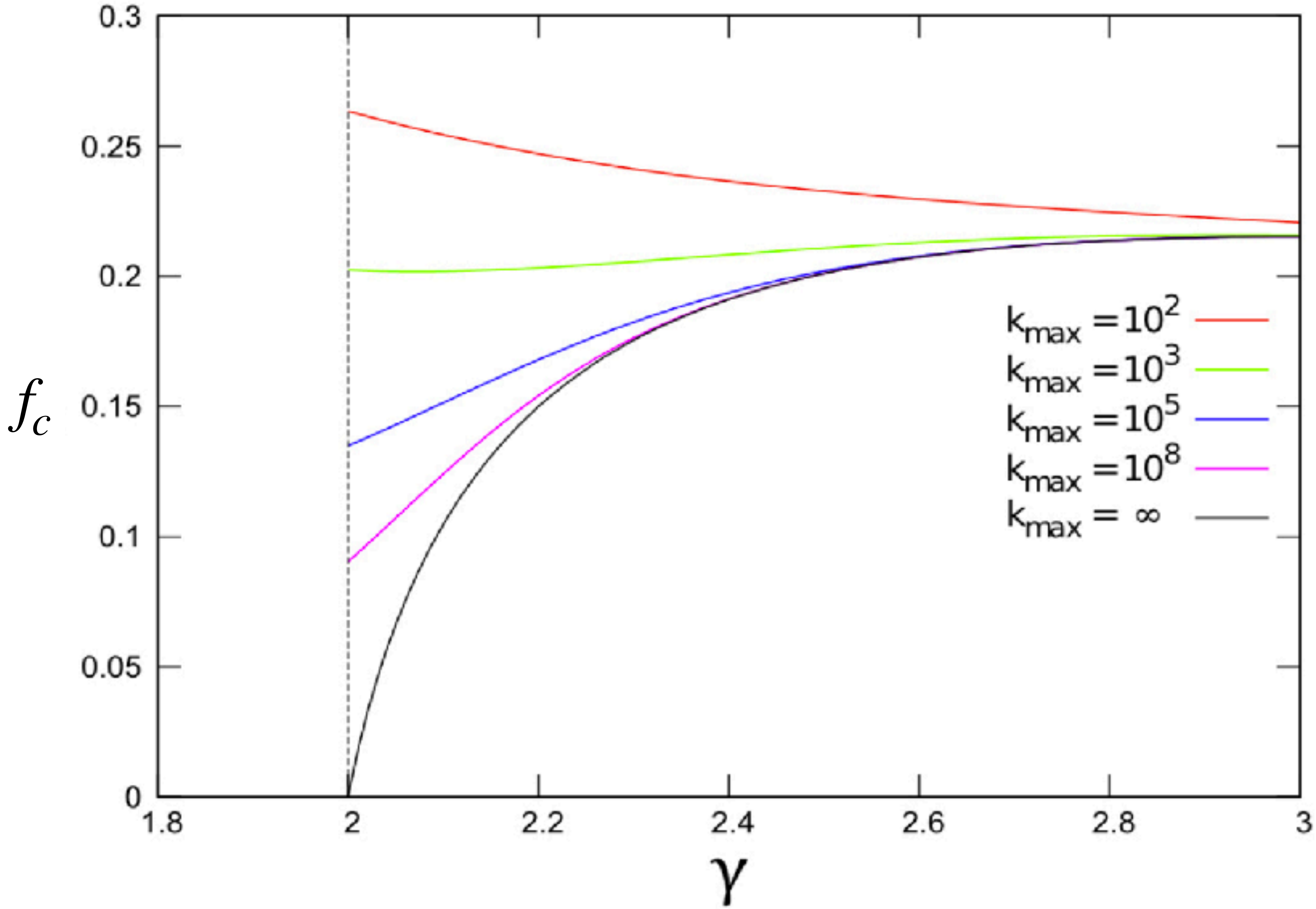


Network structure comparison



Network structure

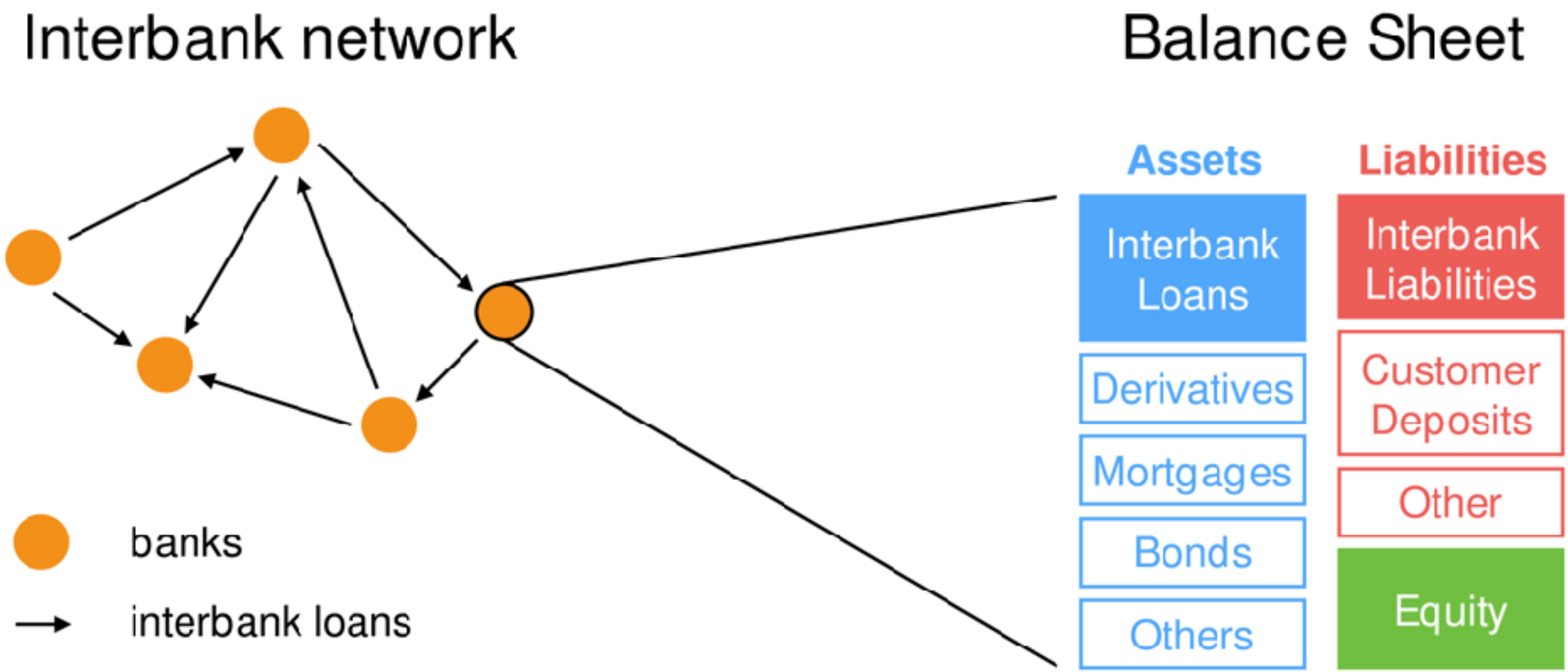
•



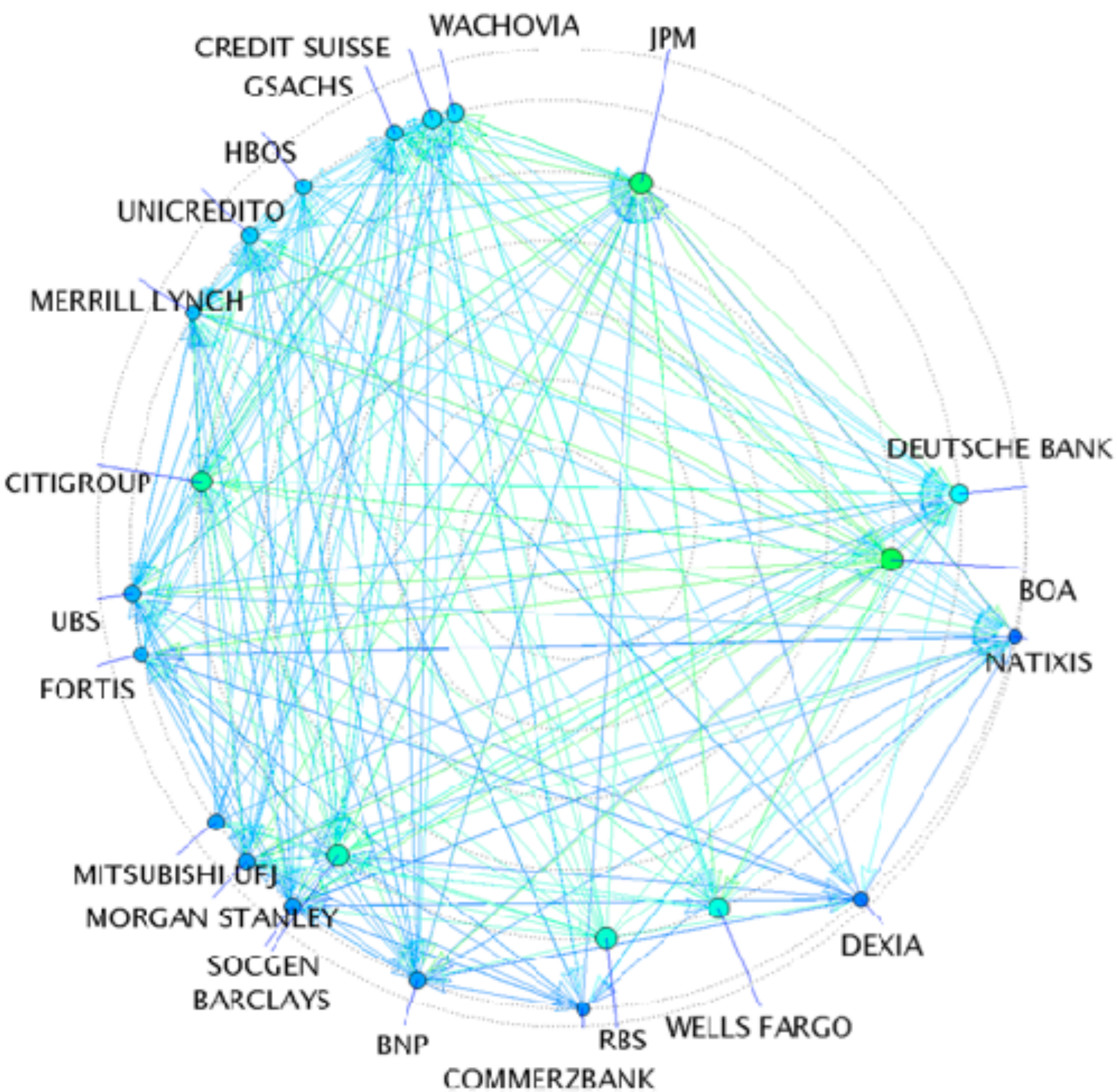
Example: Systemic risk

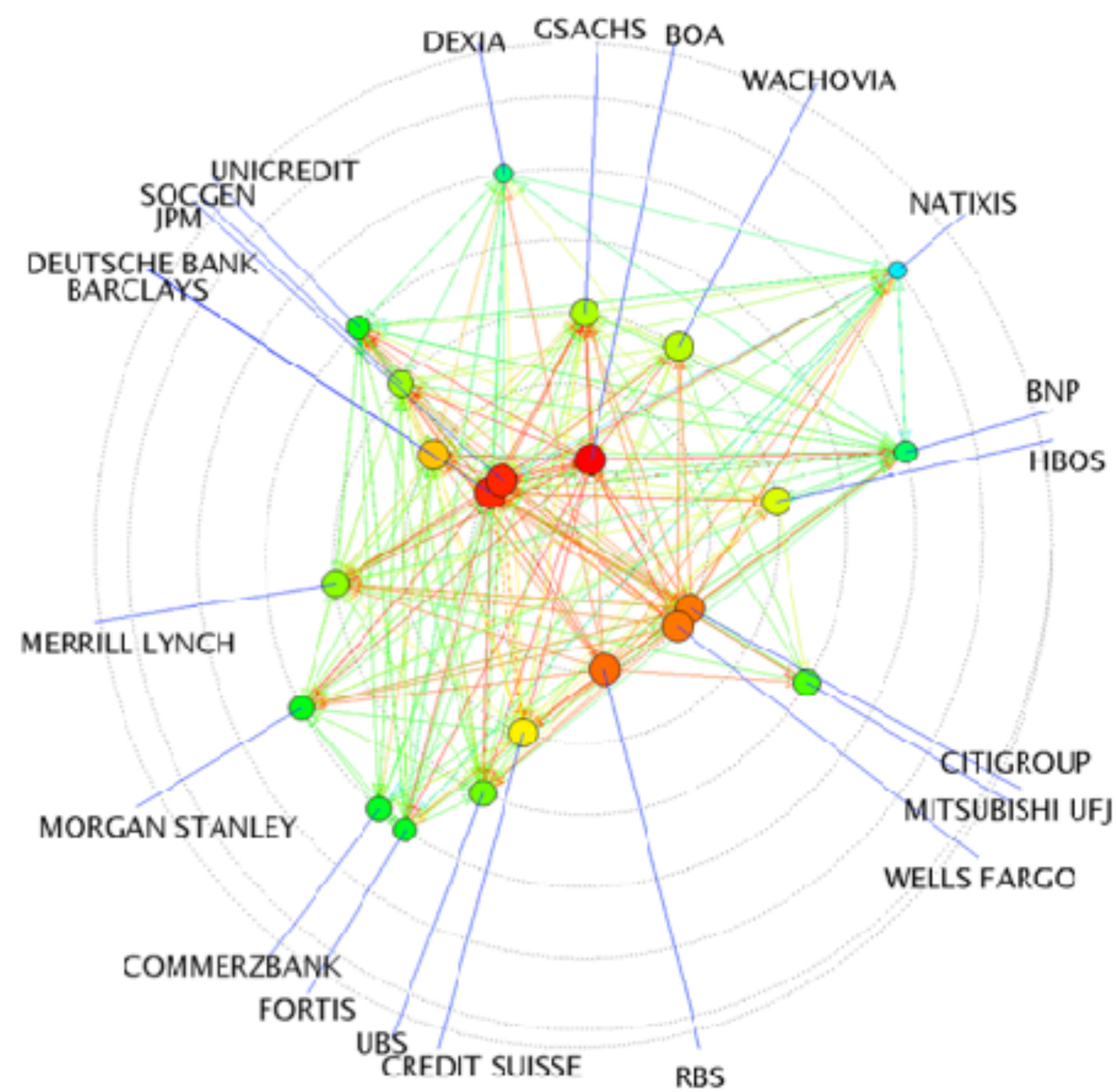
risk that default or stress **of one** or more financial institutions ("banks") **will trigger default** or stress **of further** banks.

Systemic risk



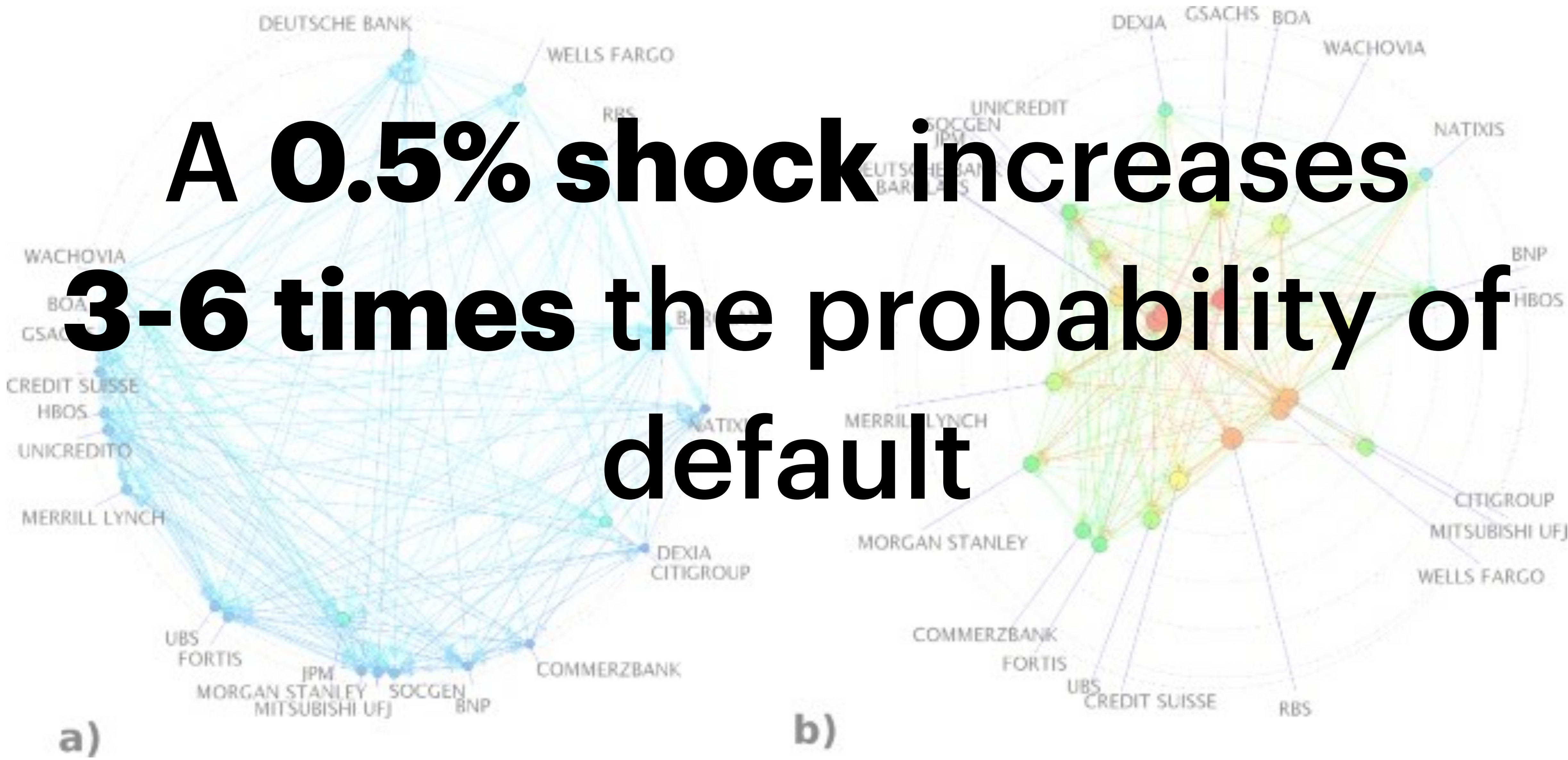
Systemic risk



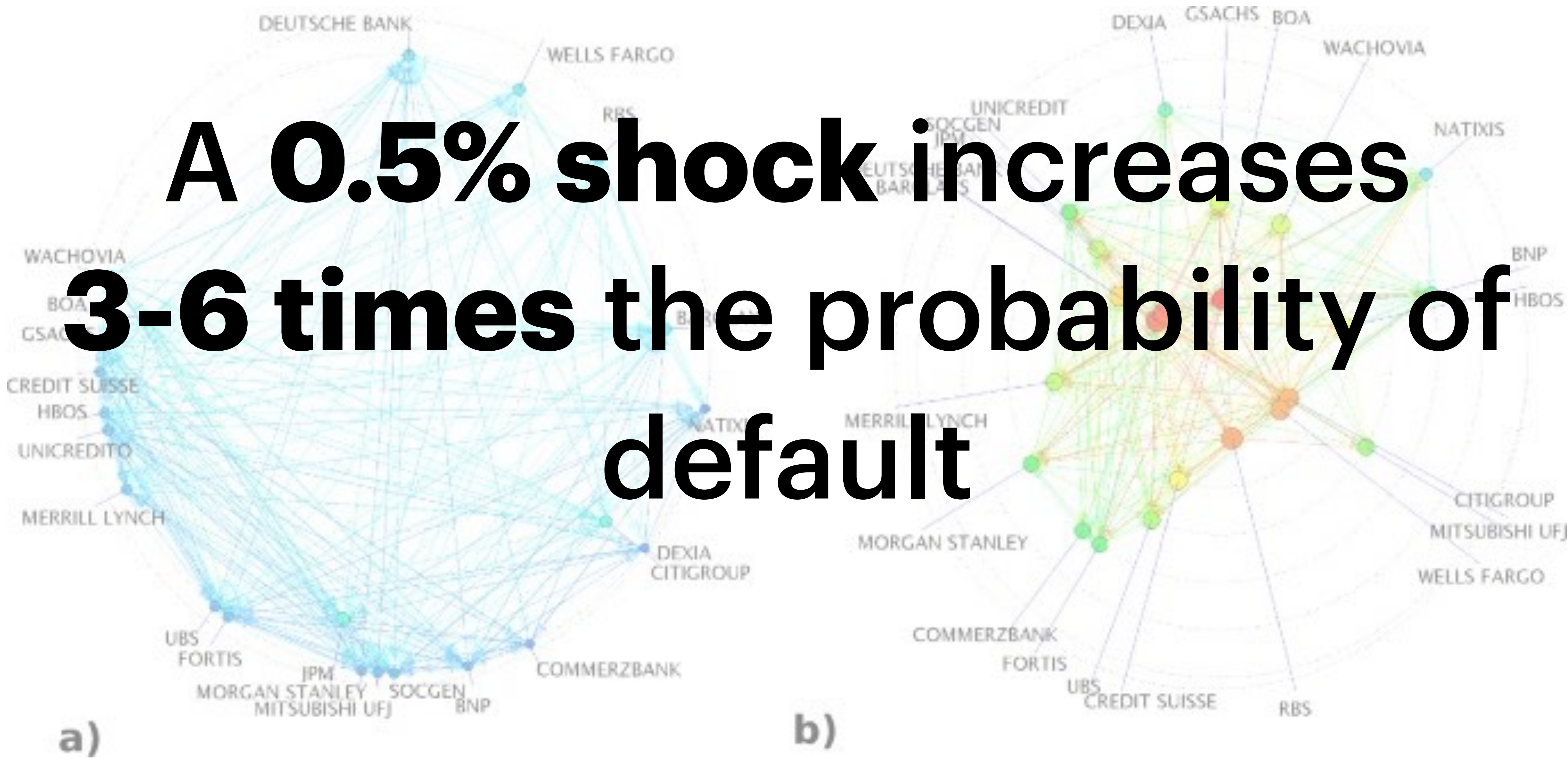


Systemic risk

A **0.5% shock** increases **3-6 times** the probability of default



Systemic risk



A **0.5% shock** increases **3-6 times** the probability of default

Battiston, S., Puliga, M., Kaushik, R. *et al*. DebtRank: Too Central to Fail? Financial Networks, the FED and Systemic Risk. *Sci Rep* 2, 541 (2012). https://doi.org/10.1038/srep00541

Think of a topic you like

Think of a topic you like

Think of an example of maximising/ minimising propagation

Influence maximisation

Selection of k nodes that best trigger a cascade

Heuristic strategies

Rule of thumb strategies that make sense


ROUND
Score Multiplier:
(type only 1, 2, or 3)
1
0
1
40
2
30
3
20
4
5
0
0
0
0
>>
<<
SHOW QUESTION
TIMER: 10
X
XXXXX

ROUND

Score 1
Multiplier:
(type ... 1, 2,
...r 3)

0

High-degree	40		0
Centrality	30		0
Friendship paradox	20		0
Random	5		0

SHOW QUESTION

TIMER: 10

X XXXXX

Kempe et al

First “influence maximisation” algorithm

Greedy algorithm - Theoretical **guarantee**

Works well with **unrealistic** assumptions

Kempe et al

Algorithm 1 Greedy Approximation Algorithm

Start with $A = \emptyset$.
while $|A| \leq k$ **do**
  For each node x, use repeated sampling to approximate $\sigma(A \cup \{x\})$ to within $(1 \pm \varepsilon)$ with probability $1 - \delta$.
  Add the node with largest estimate for $\sigma(A \cup \{x\})$ to A.
end while
Output the set A of nodes.

Kempe et al

Set of nodes

Algorithm 1 Greedy Approximation Algorithm

```
Start with A = ∅.
while |A| ≤ k do
    For each node x, use repeated sampling to approximate σ(A ∪ {x}) to within (1 ± ε) with probability 1 − δ.
    Add the node with largest estimate for σ(A ∪ {x}) to A.
end while
Output the set A of nodes.
```

Kempe et al

Set of nodes

Maximum n. of nodes in seed

Algorithm 1 Greedy Approximation Algorithm

Start with $A = \emptyset$

while $|A| \leq k$ **do**

  For each node x, use repeated sampling to approximate $\sigma(A \cup \{x\})$ to within $(1 \pm \varepsilon)$ with probability $1 - \delta$.

  Add the node with largest estimate for $\sigma(A \cup \{x\})$ to A.

end while

Output the set A of nodes.

Kempe et al

Set of nodes

Maximum n. of nodes in seed

Algorithm 1 Greedy Approximation Algorithm

Start with $A = \emptyset$

while $|A| \leq k$ **do**

  For each node x, use repeated sampling to approximate $\sigma(A \cup \{x\})$ to within $(1 \pm \varepsilon)$ with probability $1 - \delta$.

  Add the node with largest estimate for $\sigma(A \cup \{x\})$ to A.

end while

Output the set A of nodes.

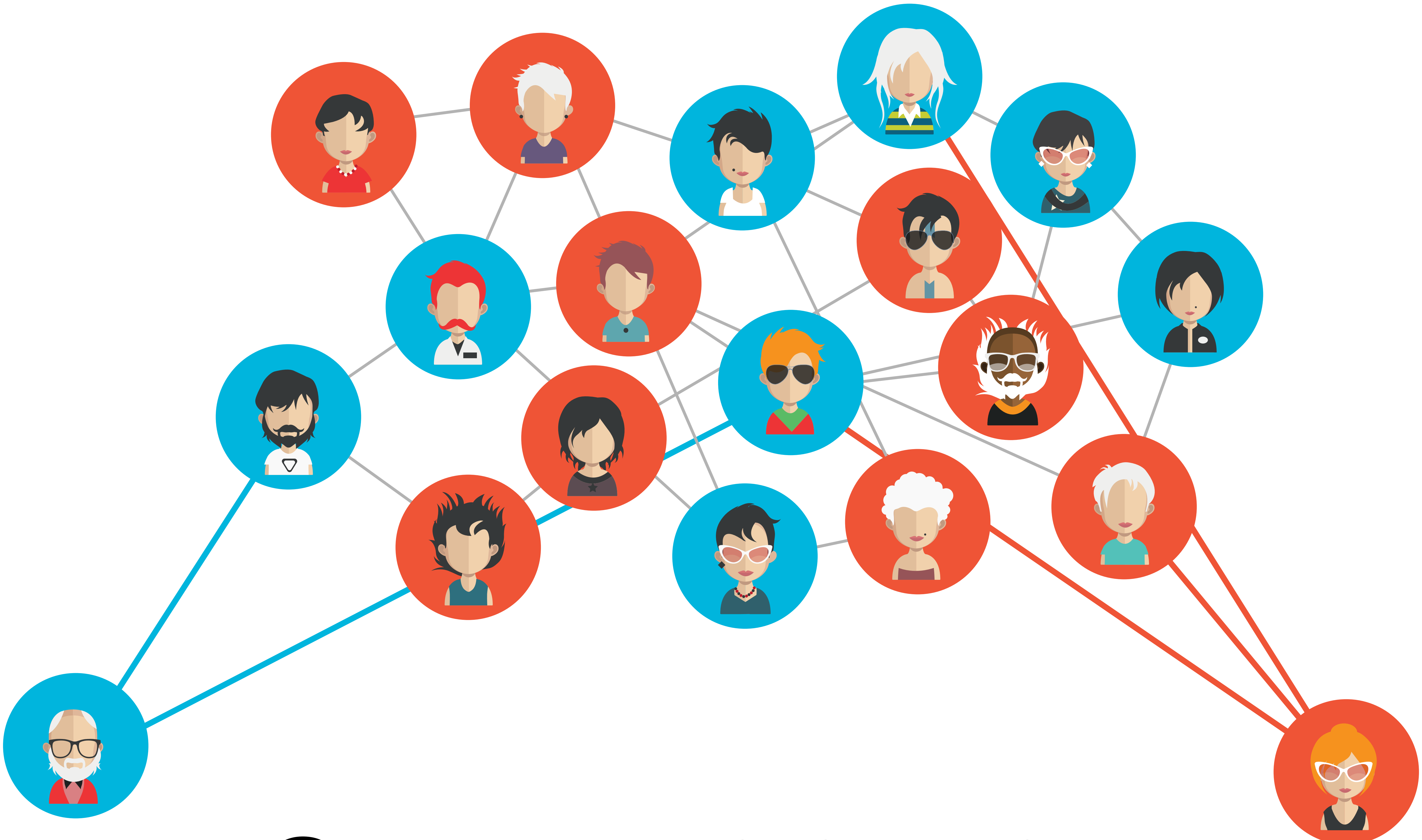
Influence of set of nodes a+x

Competitive im

Two or more parties **compete** for influence

Classical setting: **2 parties**, opposite sides

Easy to study on **voter model**



Zealot

Competitive im

Zealot

Competitive im on voter model

$$\Delta_i \frac{dx_i}{dt} = (1 - x_i)(\sum_j a_{ji} x_j + p_{A,i}) - x_i(\sum_j a_{ji}(1 - x_j) + p_{B,i})$$

Competitive IM on voter model

Probability being in state A

$$\Delta_i \frac{dx_i}{dt} = (1 - x_i)(\sum_j a_{ji}x_j + p_{A,i}) - x_i(\sum_j a_{ji}(1 - x_j) + p_{B,i})$$

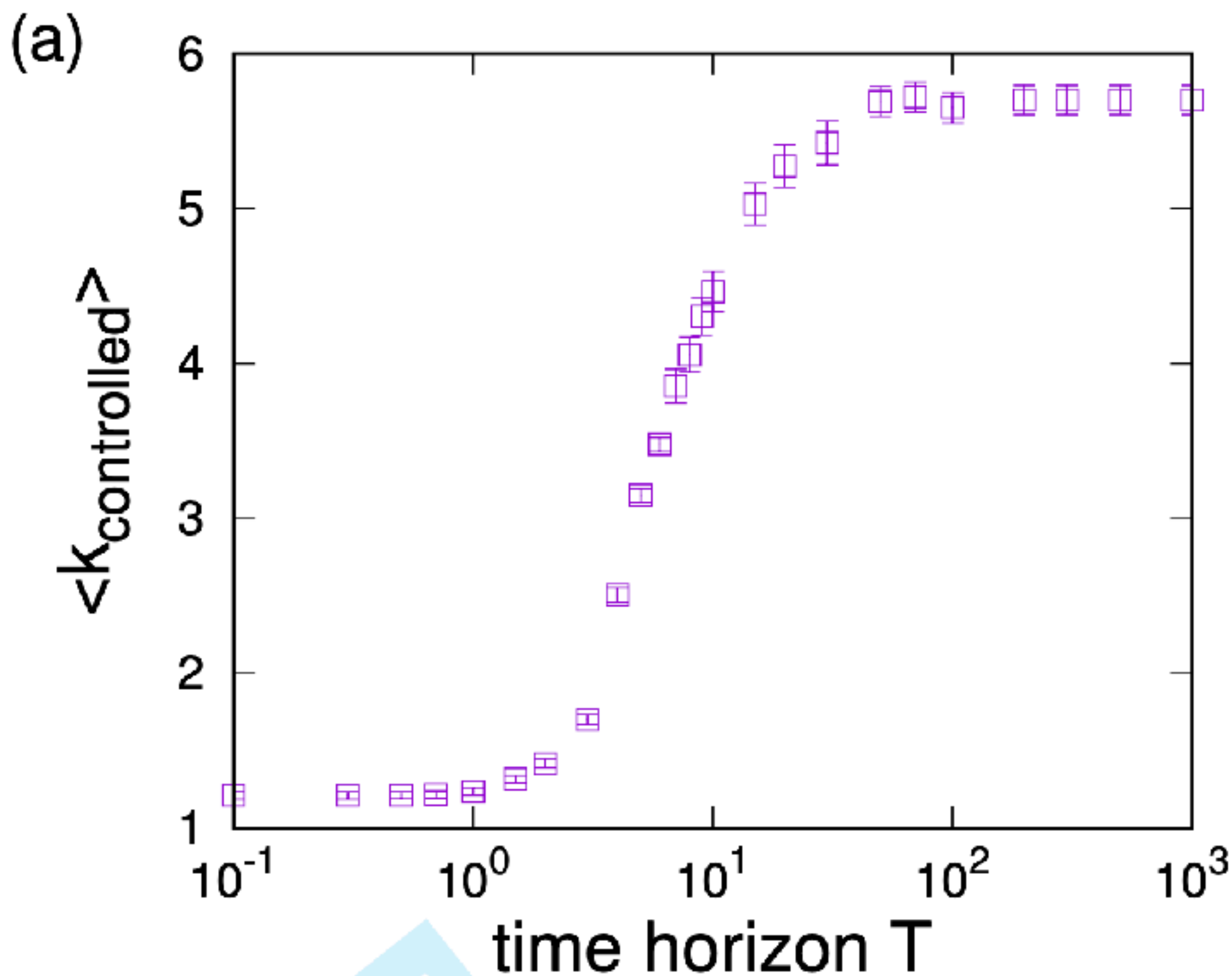
Influence of zealot A

Normalisation factor

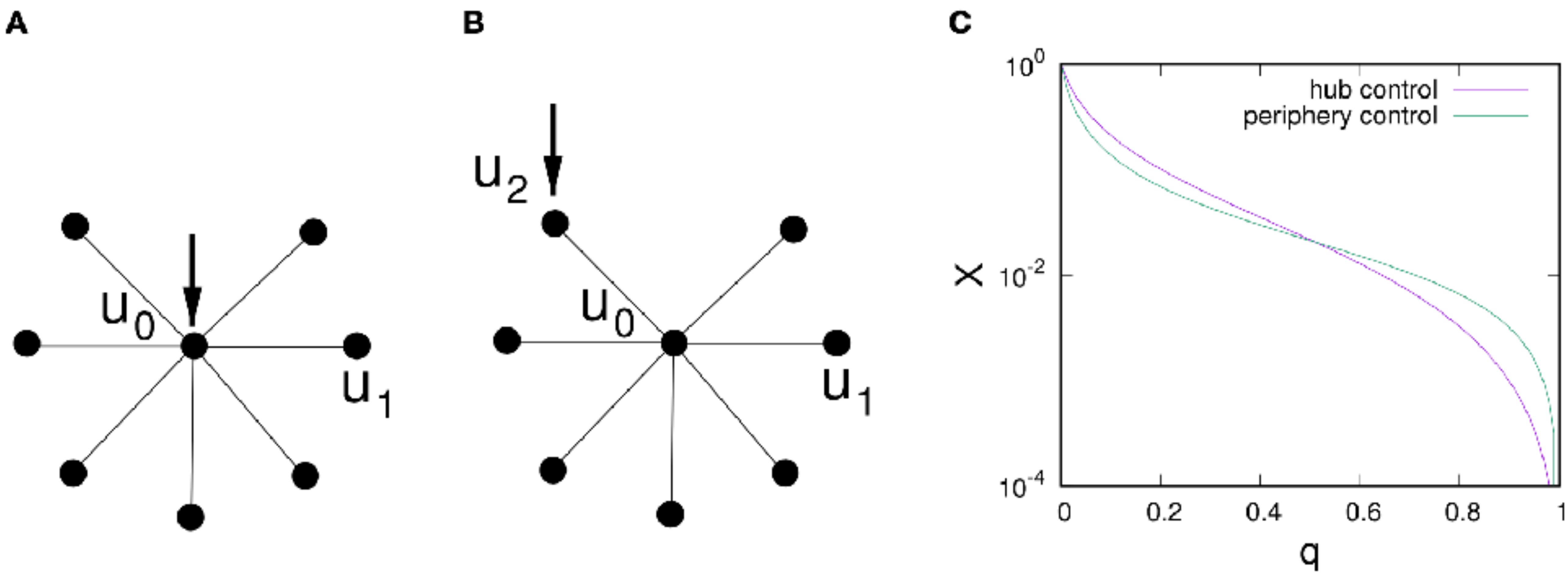
Influence of neighbours

$$\Delta_i = \sum_j a_{ji} + p_{A,i} + p_{B,i}$$

Competitive IM on voter model

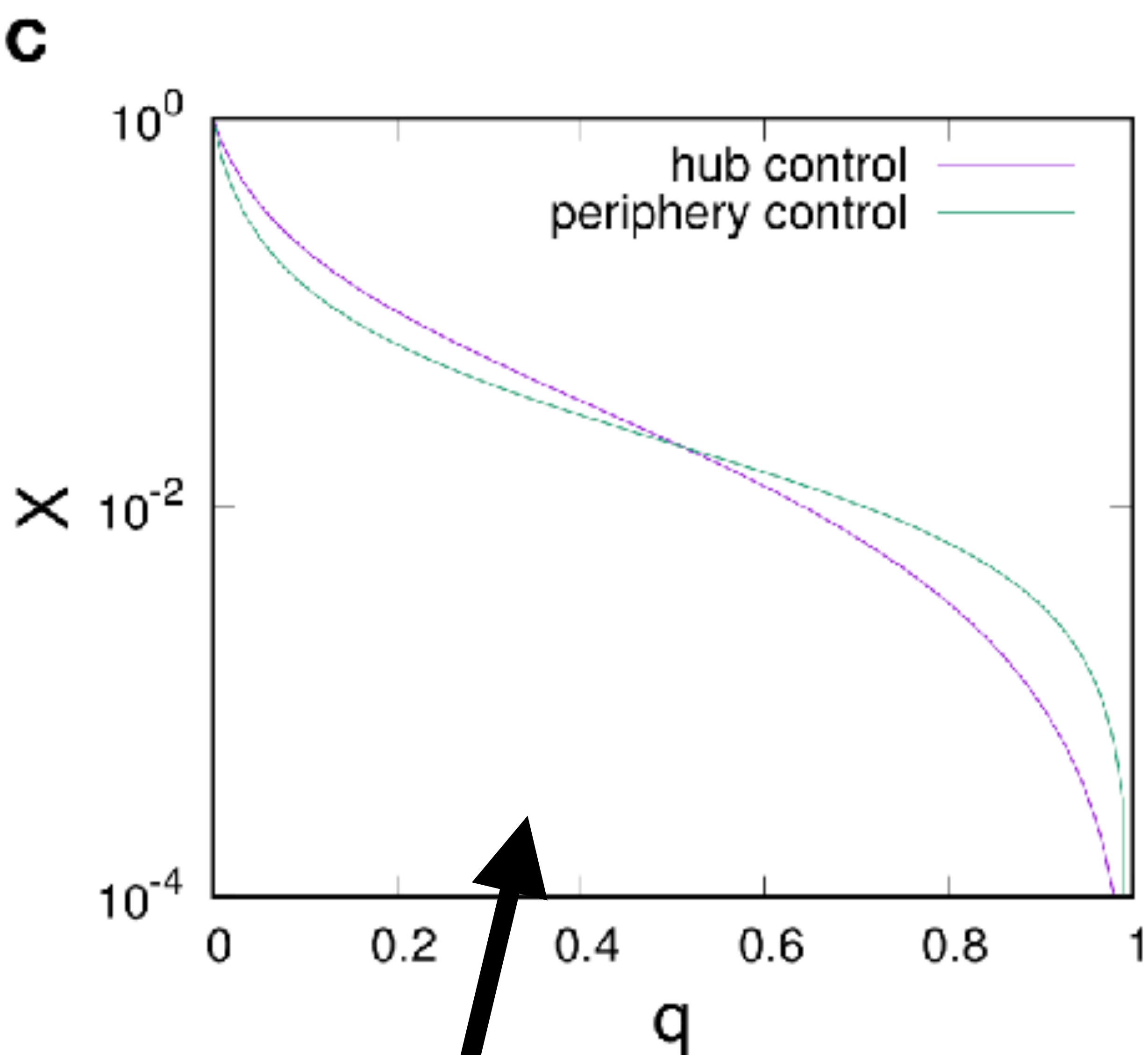


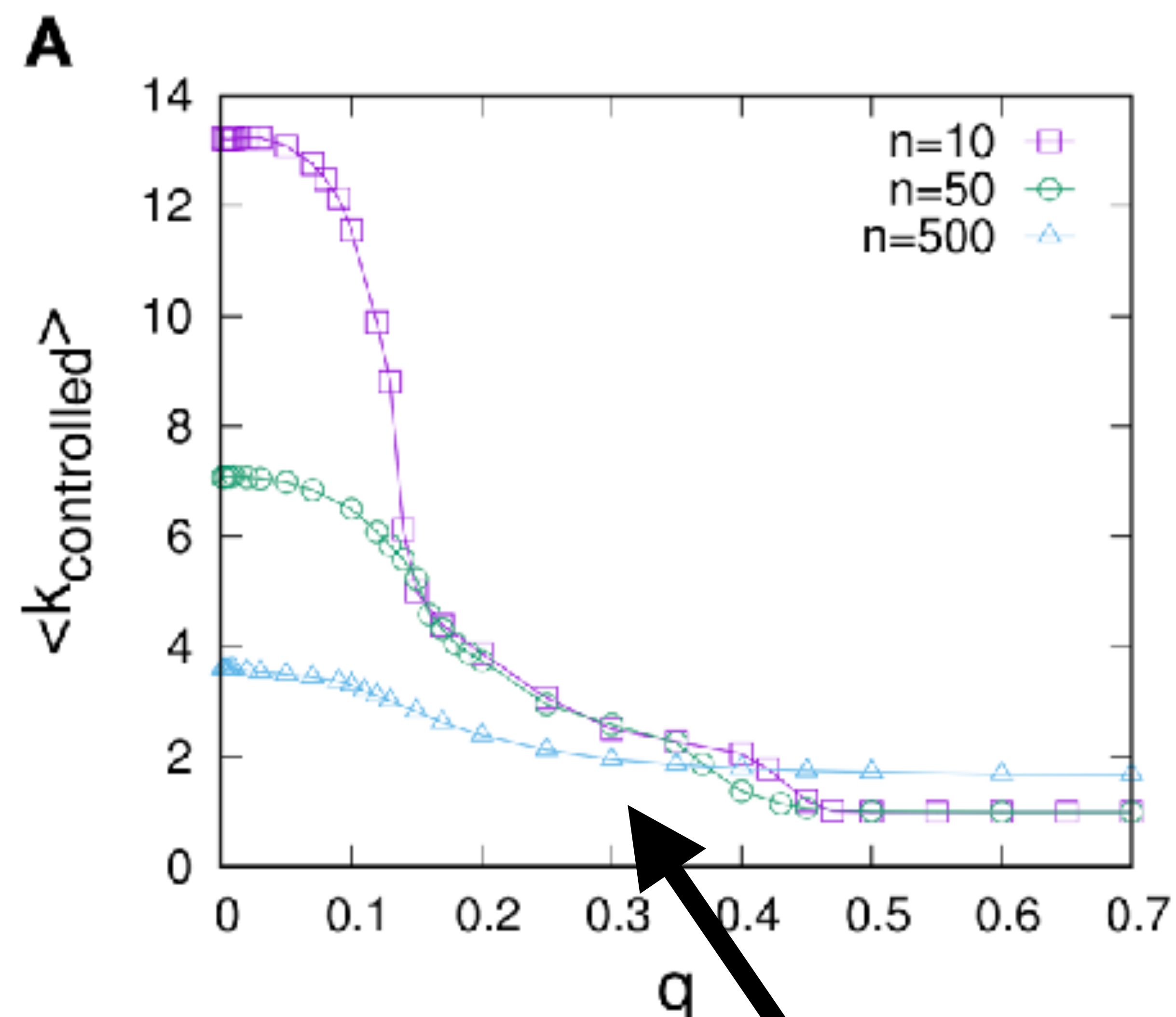
Temporary influence



q = probability of flipping back to pre-influence state

Temporary influence





Star network

Scale-free network

Summary

Percolation and its implications
Systemic risk and instability of finance
Influence maximisation