Clique Percolation and Resolution Limits in Community Detection

Yu Chen
Department of Computer Science
Rensselaer Polytechnic Institute
11/30/2015

Modularity: Resolution Limits

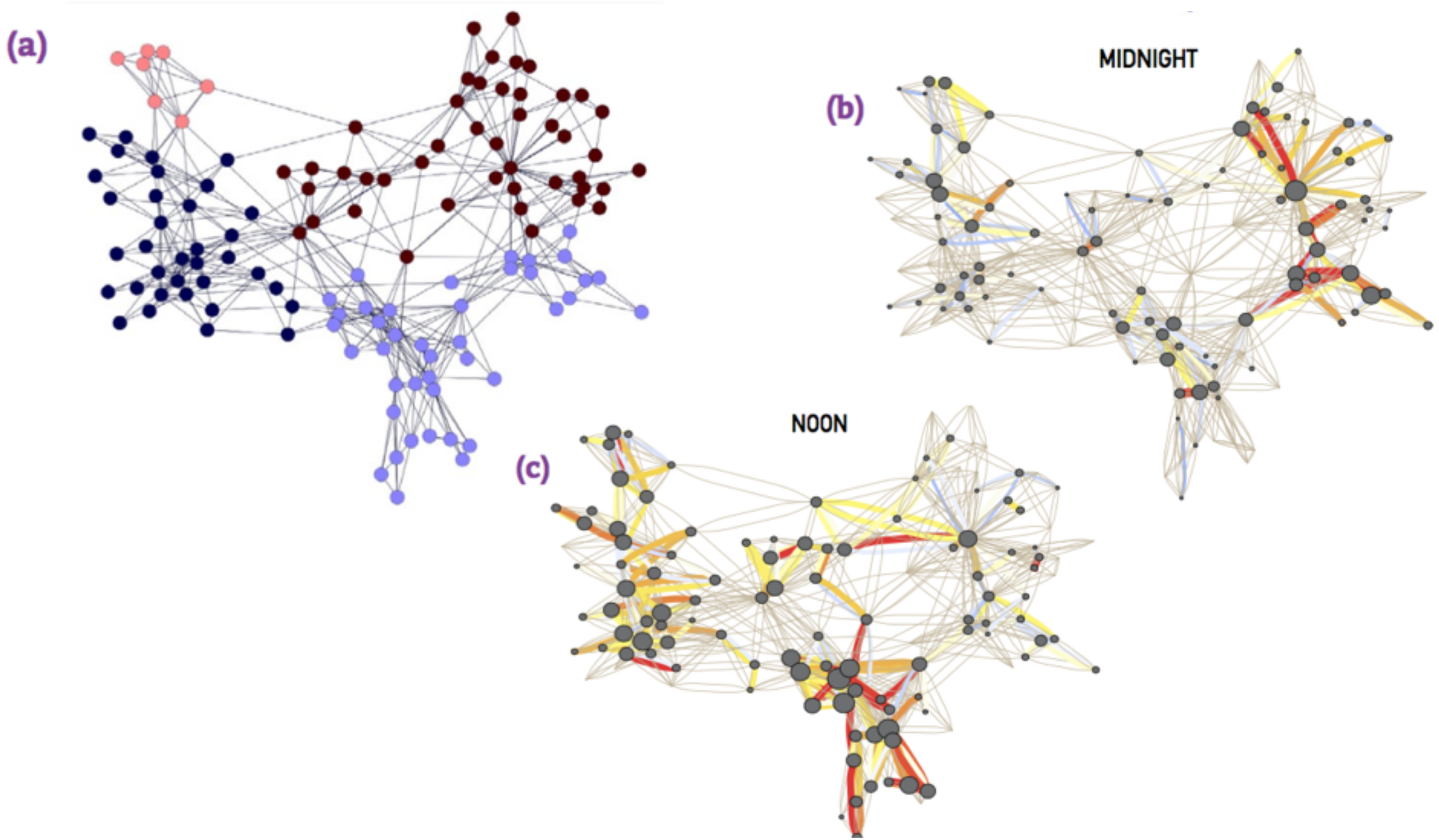
S Fortunato, M Barthélemy, Resolution limit in community detection, Proceedings of the National Academy of Sciences, 2007

Finding and evaluating community structure in networks MEJ Newman, M Girvan Physical review E, 2004

Albert-Laszlo Barabasi & Mauro Martino, Network Science.

Why Communities Matter

Uncovering the relationships between **structure** and **function** in complex networks.



Communities and call patterns

Modularity

➢ **Modularity (Q):** the difference between **the observed fraction of edges inside the communities** and **the expected value** in an equivalent subgraph with edges placed **at random**.

$$Q = \sum_{s=1}^{m} \left[\frac{l_s}{L} - \left(\frac{d_s}{2L} \right)^2 \right]$$

l_s : the number of edges inside the module s

L : the total number of edges in the network

d_s : the total degree of the nodes in the module s

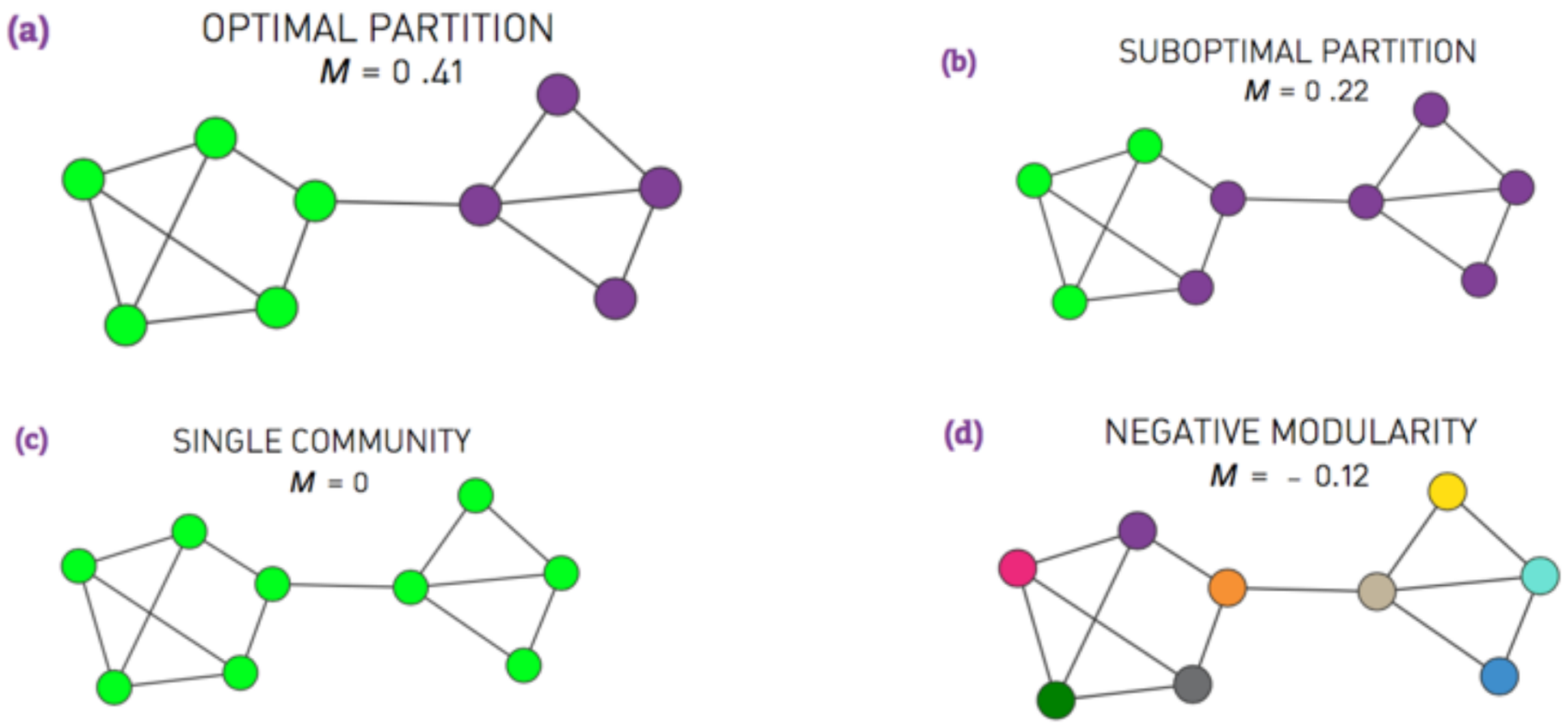
➢ **Random hypothesis:** **Randomly wired networks lack an inherent community structure.**

➢ **Properties:** Higher modularity implies better partition.

Maximal modularity Hypothesis

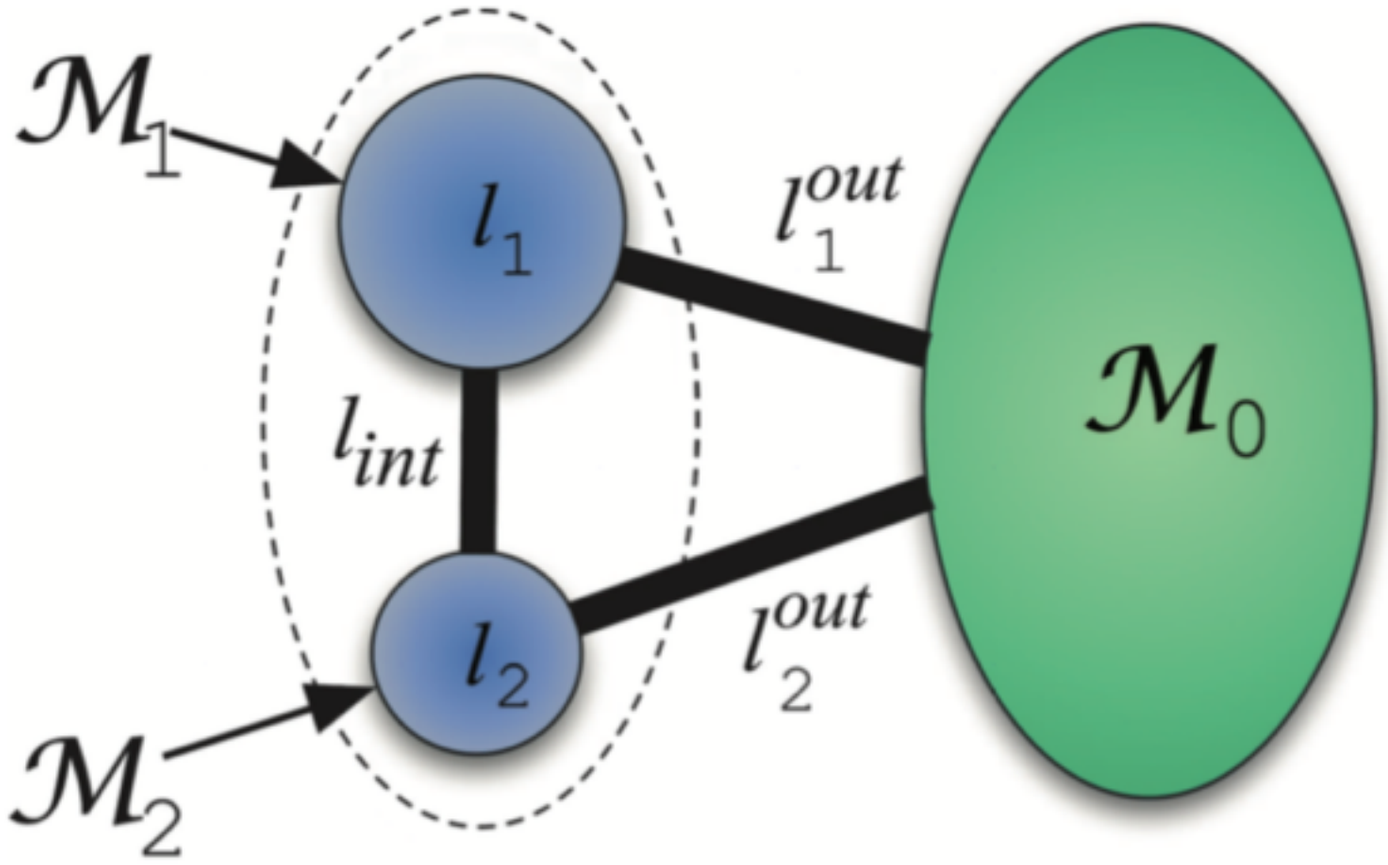
- **Maximal modularity hypothesis:** **For a given network, the partition with maximum modularity corresponds to the optimal community structure.**

- **Modularity-driven community detection algorithms:** Select the partition with maximal overall modularity.



Limits of Modularity: Resolution Limit

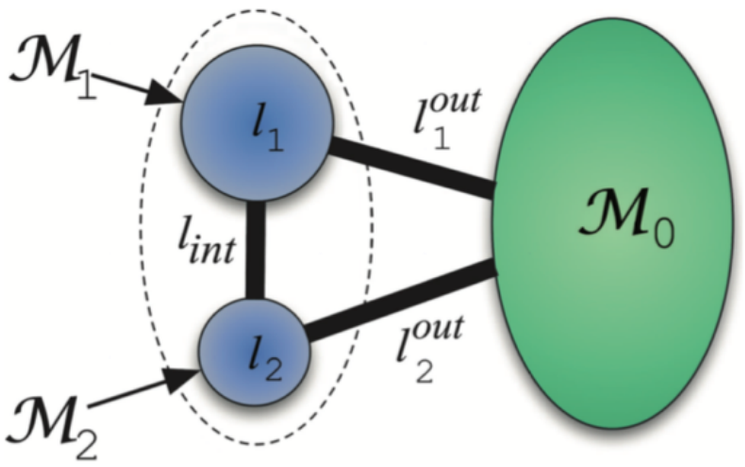
➢ **Resolution Limit:** Modularity maximization **forces small communities into larger ones** even if they are clearly distinct communities.



Limits of Modularity: Resolution Limit (cont'd)

If we merge communities 1 and 2 together, the network's overall modularity changes with ΔQ_{12}

$$\Delta Q_{12} = \left[\frac{L_{12}}{L} - \left(\frac{k_{12}}{2L}\right)^2\right] - \left[\frac{L_1}{L} - \left(\frac{k_1}{2L}\right)^2 + \frac{L_2}{L} - \left(\frac{k_2}{2L}\right)^2\right]$$



where

$$L_{12} = L_1 + L_2 + l_{12}$$
$$k_{12} = k_1 + k_2$$

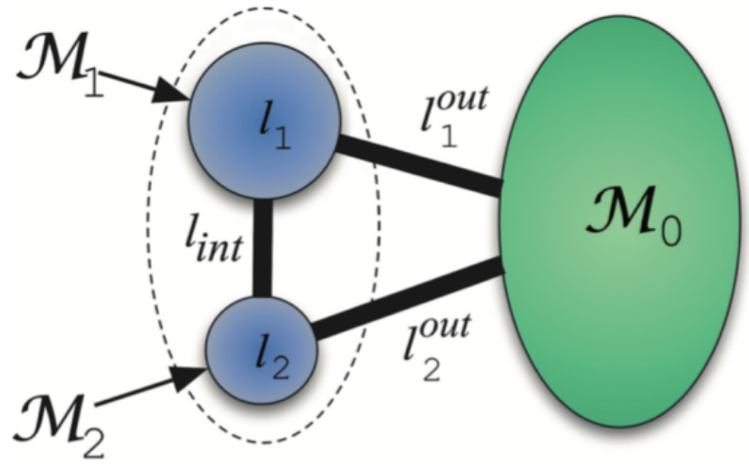
l_{12} is the number of edges between the nodes of communities 1 and 2.

Finally, we obtain

$$\Delta Q_{12} = \frac{l_{12}}{L} - \frac{k_1 k_2}{2L^2}$$

Limits of Modularity: Resolution Limit (cont'd)

$$\Delta Q_{12} = \frac{l_{12}}{L} - \frac{k_1 k_2}{2L^2}$$



Consider the case when there is at least one edge between communities 1 and 2 $(l_{12} >= 1)$ and $k_1 k_2 / 2L < 1$. Hence, we must merge communities 1 and 2, because

$$\Delta Q_{12} > 0$$

in this case even if communities 1 and 2 are two complete graphs connected by a single edge.

That is, we might **miss some important local structures at smaller scales** through modularity maximization.

For example, for the WWW sample with L=1,497,134, modularity maximization will have difficulties resolving communities with total degree $k_c <= 1730$.

Resolution Limits: Summary

- **Modularity maximization** helps us uncover local structures in complex networks.
- But it naturally fails to discover local structures **at smaller scales (limit resolution)**.
- Its role should probably be limited to the comparison of partitions with the same number of modules.
- We need more robust metrics or strategies.

Clique Percolation in Community Detection

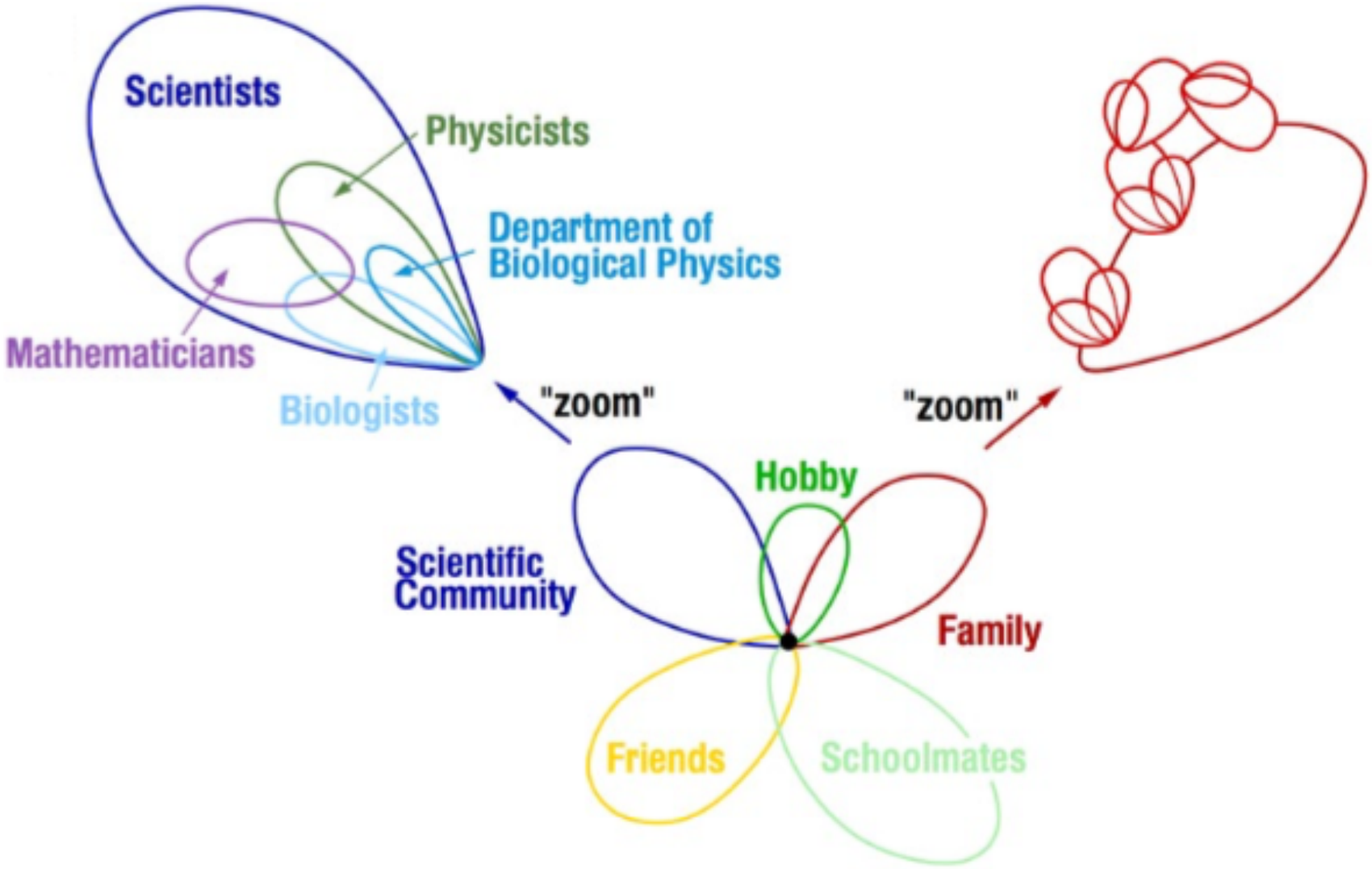
I Derényi, G Palla, T Vicsek, Clique percolation in random networks, Physical review letters, 2005

G Palla, I Derényi, I Farkas, T Vicsek, Uncovering the overlapping community structure of complex networks in nature and society, Nature, 2005

Albert-Laszlo Barabasi & Mauro Martino, Network Science.

Overlapping Communities

Many existing community detection algorithms **force each node into a single community**.



Schematic representation of the communities surrounding a scientist

Clique Percolation: Terminology

- **k-cliques:** complete (fully connected) subgraphs of k vertices.
- **k-clique adjacency:** two k-cliques are adjacent if they share k - 1 vertices.
- **k-clique chain:** a subgraph, which is the union of a sequence of adjacent k-cliques.
- **k-clique connectedness:** two k-cliques are k-clique-connected if they are parts of a k-clique chain.
- **k-clique percolation cluster (or k-clique community):** a maximal k-clique-connected subgraph.

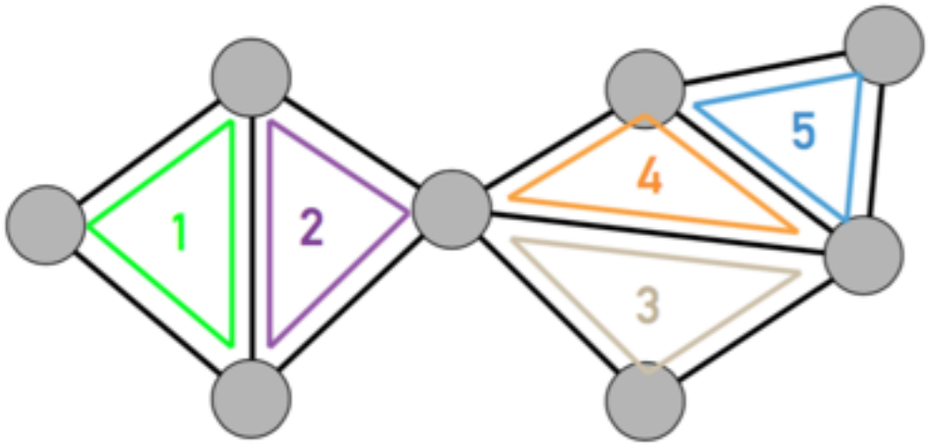
Rolling a k-clique template

- Rolling a k-clique template is the process that we roll a k-clique to an adjacent k-clique **by relocating one of its vertices and keeping its other k – 1 vertices fixed.**

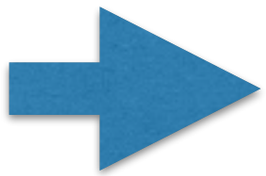
- **A k-clique community** of a network are all those subgraphs that can be fully explored but cannot be left by rolling a k-clique template in them. (Think about a connected component)

- Finding a k-clique community is somehow similar to finding a connected component.

An equivalent interpret of k-clique percolation



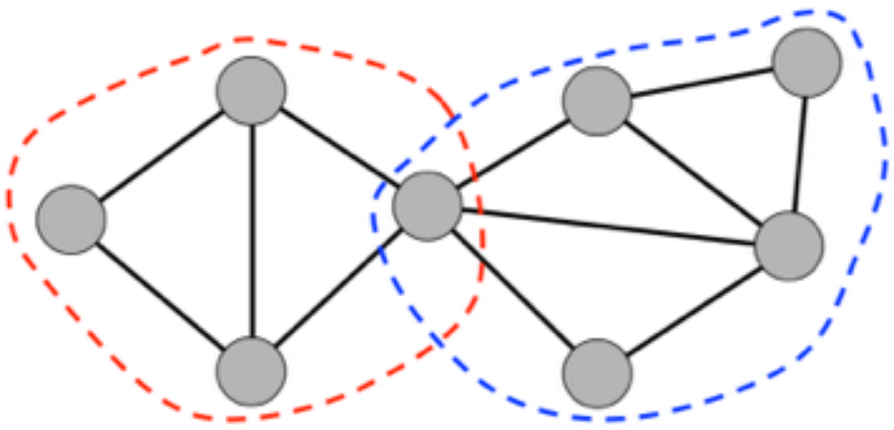
(a)



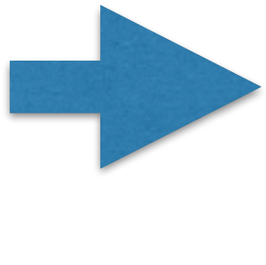
O=

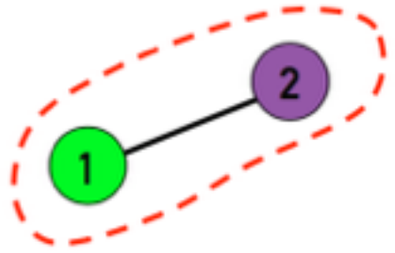
	1	2	3	4	5
1	0	1	0	0	0
2	1	0	0	0	0
3	0	0	0	1	0
4	0	0	1	0	1
5	0	0	0	1	0

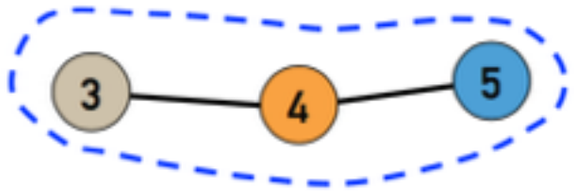
(b)



(c)







(d)

Threshold probability of k-clique percolation

➢ Recall the threshold probability that a giant component appears in ER networks, there should be a threshold probability of k-clique percolation which could be regarded as the generalized version of the above case (k=2).

➢ We find that **a giant k-clique component appears in an ER network** at the threshold probability $p = p_c(k)$, where

$$p_c(k) = \frac{1}{[(k-1)N]^{\frac{1}{k-1}}}$$

Threshold probability of k-clique percolation (cont'd)

- Consider after rolling a k-clique template from a k-clique to an adjacent one (by relocating one of its vertices), the **expectation value of the number of adjacent k-cliques**, where the template can roll further.
- A smaller expectation value would result in **premature** k-clique percolation clusters.
- A larger expectation value would allow an infinite series of bifurcations for the rolling, ensuring that **a giant cluster** is present in the system.

- This expectation value can be estimated as

$$(k-1)\,(N-k-1)\,p^{k-1}$$

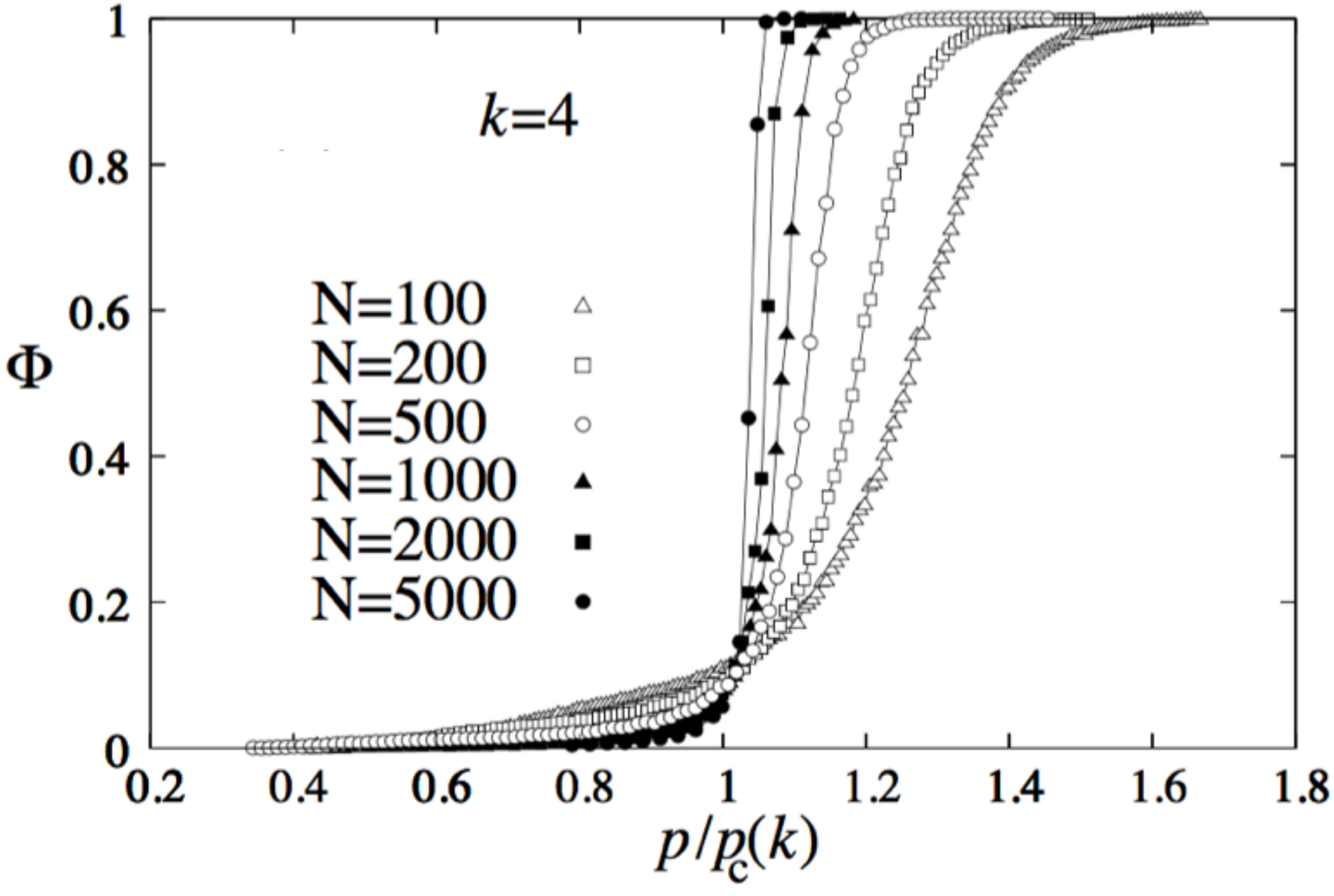
For large N, this equation can be written as

$$(k-1)\,Np^{k-1}$$

Let $(k-1)\,Np^{k-1}=1$, we obtain the threshold probability

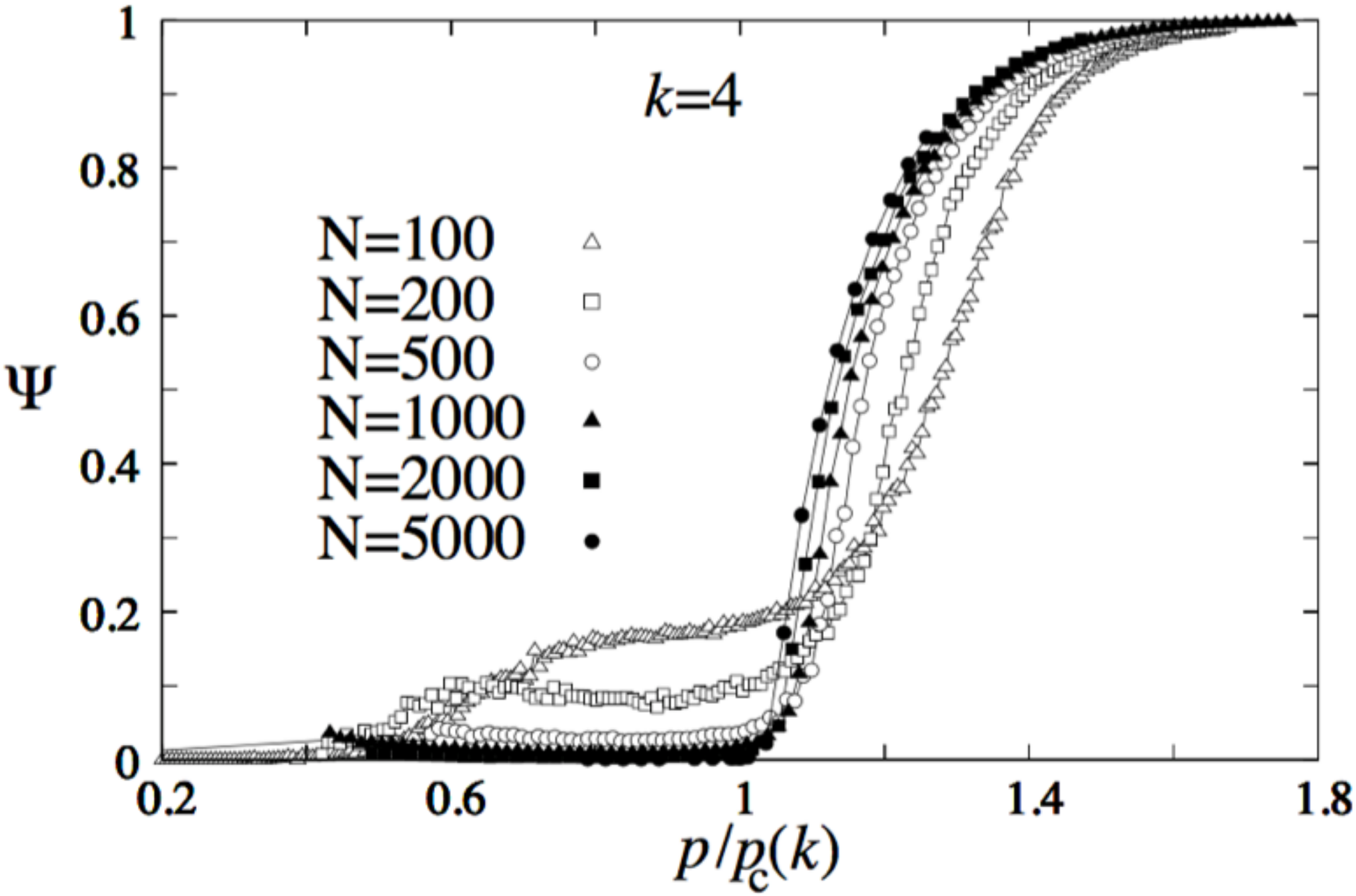
$$p_c\,(k)=\frac{1}{\left[(k-1)\,N\right]^{\frac{1}{k-1}}}$$

Numerical Simulations of the Threshold Probability



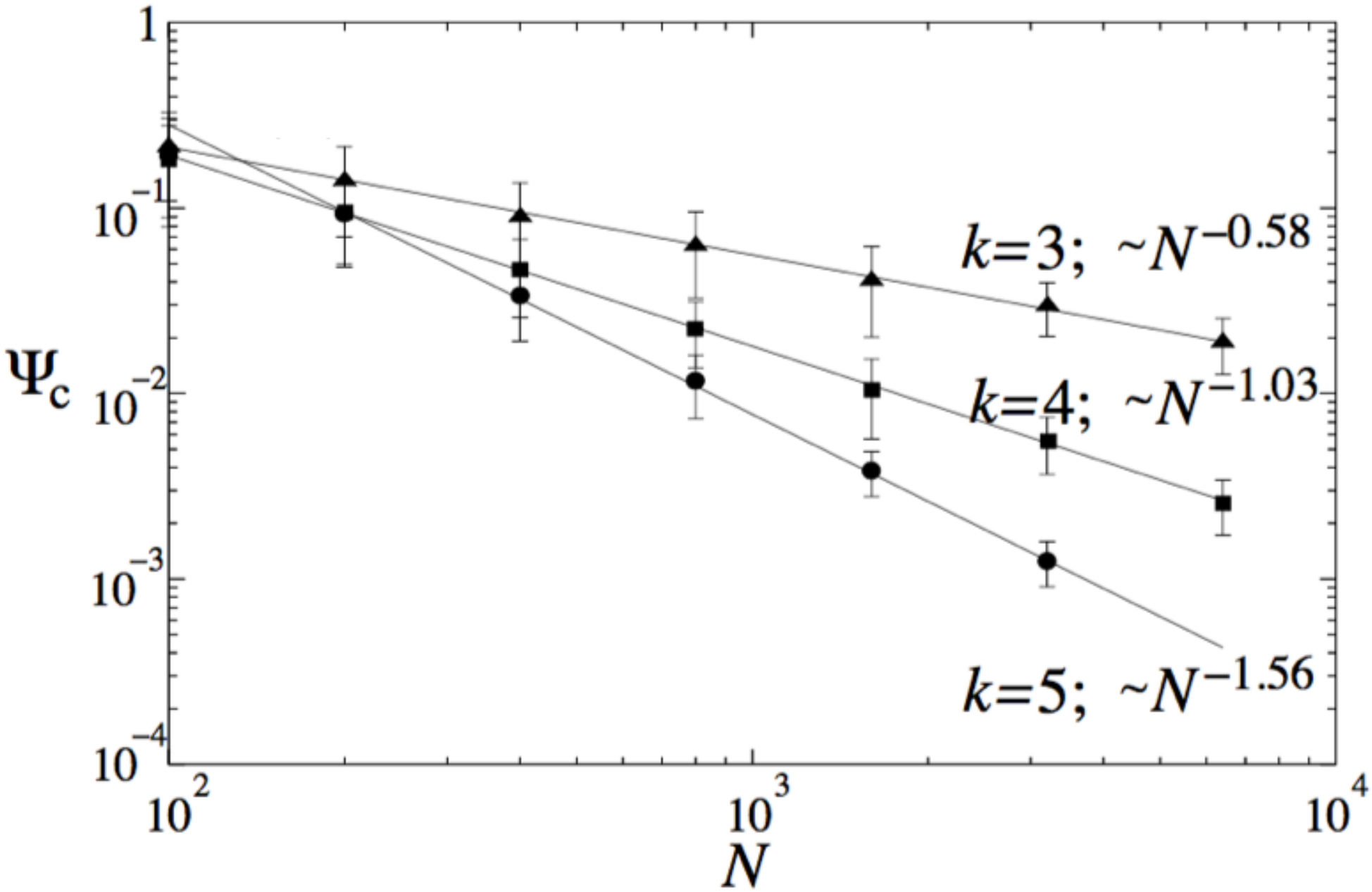
Φ denotes the fraction of vertices belonging to a giant k-clique community

Numerical Simulations of the Threshold Probability (cont'd)



Ψ denotes the fraction of k-cliques belonging to a giant k-clique community

Numerical Simulations of the Threshold Probability (cont'd)



Ψc denotes Ψ at the threshold p_c

The Clique Percolation Algorithm (CFinder)

- Clique percolation can be used to find **overlapping communities** in a network.

- If we start rolling a k-clique template from an initial k-clique, we can find all the k-cliques which are “reachable” from the initial k-clique. Thus we obtain a k-clique community. With a similar process, we can find other k-clique communities (allow overlaps) in the network.

- With different values of k we can **identify communities of different strength**.

The Theoretic Applicability of CFinder

- The **sharp percolation transition** of ER network guarantees the theoretic applicability of CFinder.

- For example, if a network with 1, 000, 000 nodes is completely random (ER network), and we set k = 4, the threshold probability that a giant k-clique community appears would be 0.0069, which means there must be at least 3.5 billion edges in the network! In other words, the network should be very dense. If we set k even larger, e.g., k = 6, there must be 22.9 billion edges in the network!!! That's almost impossible in a real network.

- Hence, **if large k-clique communities do appear, they must correspond to locally dense structures, i.e., real communities**, which means CFinider is applicable to find real communities in a real network.

Clique Percolation: Summary

- The clique percolation process.
- The **threshold probability** of clique percolation in an ER network.
- Clique percolation algorithm (CFinder) is applicable to find **overlapping** communities.

Thanks!

Q & A