The Research on Approximating the Real Network Degree Distribution Level Based on DCSBM

Tianyu Qi[1], Hongwei Zhang[2], Yufeng Zhan[1], Yuanqing Xia[1]

1. Beijing Institute of Technology, Beijing 100081, P. R. China
E-mail: qitianyu@bit.edu.cn, yu-feng.zhan@bit.edu.cn, xia_yuanqing@bit.edu.cn
2. Fudan University, Shanghai 200433, P. R. China
E-mail: zhanghw.hongwei@gmail.com

Tianyu Qi 2022.07.25

CONTENTS

Research Background:

- Many things in the real world can be simplified as a complex system composed of nodes and the relationships between nodes like a graph.

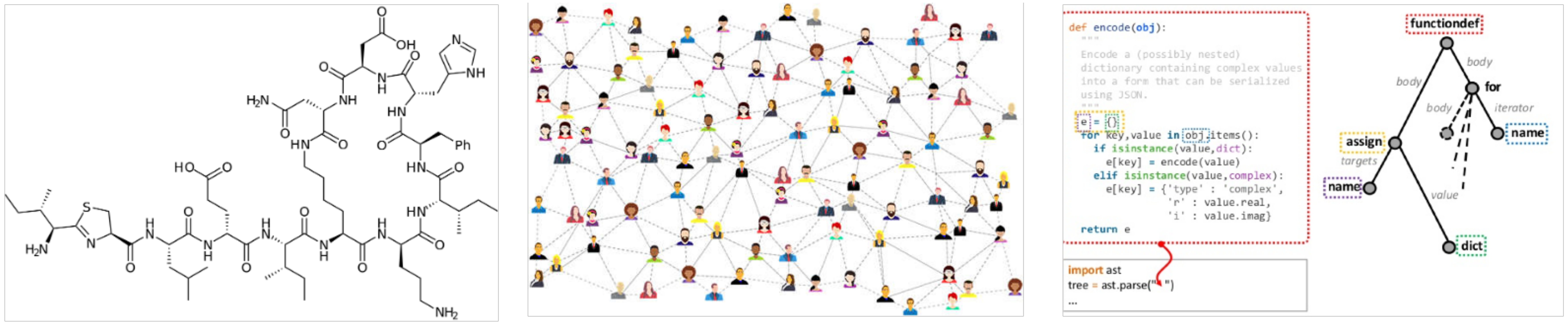


Fig 1: Application of common graph topologies

Challenge:

- The real graph topology we can get is limited.
- Real networks in different domains have statistical properties.

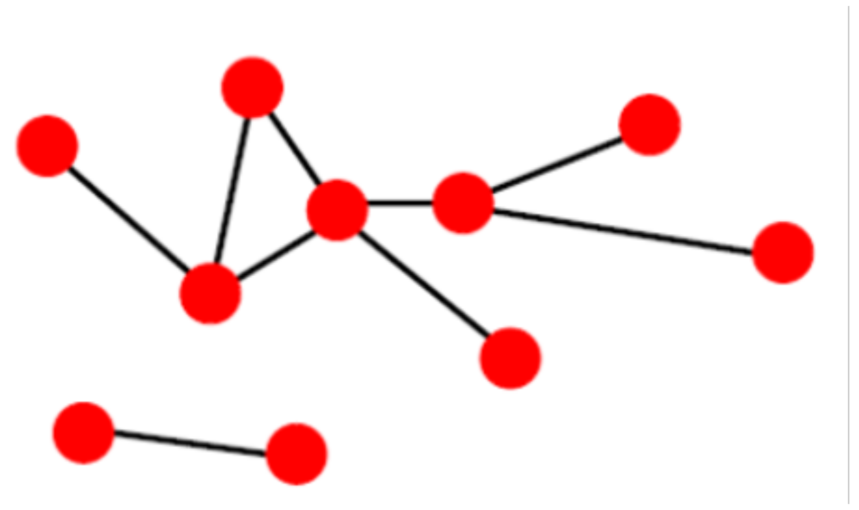


Fig 2: Random Graph Model

Research Status:

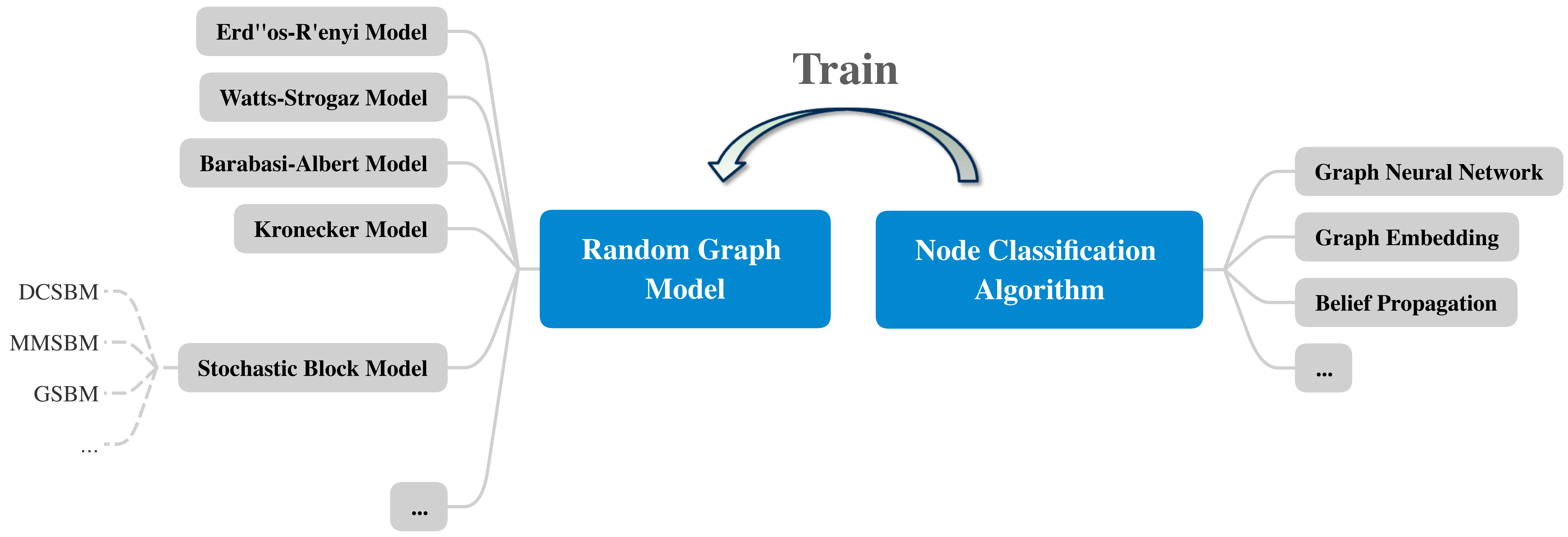


Fig 3: Research Status

Our Contributions:

- Infer the phase transition of DCSBM using a physics method called Belief Propagation (BP) algorithm.
- Test how similar the DCSBM is to the real graphs in the distribution level.
- Explore the effect of different community structure parameters on the phase transition.

CONTENTS

How to construct SBM:

- Suppose a graph has N nodes and the adjacency matrix A_{ij} is represented as an edge between node i and node j.
- Suppose there are a group of SBM, and there is a $q \times q$ matrix P_{ab} that represents the probability of edge between group a and b, where the matrix element is p_{in} when $a = b$, and the matrix element is p_{out} when $a \neq b$.

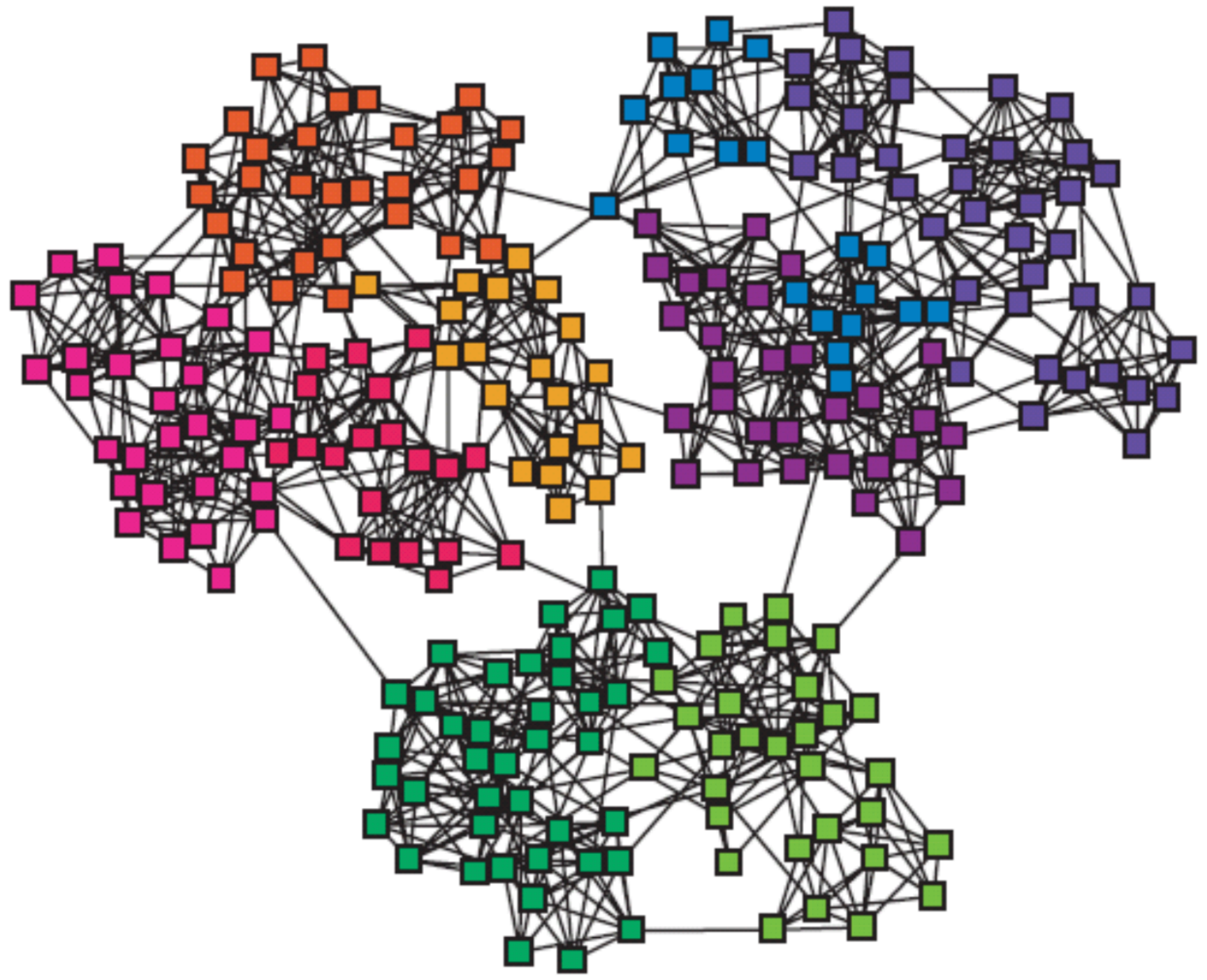


Fig 4: Stochastic Block Model

- The group of node i is t_i, then the probability of edge between i and j is $p_{t_i t_j}$, and the probability of non-edge is $1 - p_{t_i t_j}$.
- Since $P_{ab} = O(1/N)$ exists in the sparse graph generation, an matrix $C_{ab} = NP_{ab}$ is defined, which can be expressed as:

$$p_{in} = \frac{c_{in}}{N} \qquad p_{out} = \frac{c_{out}}{N}$$

SBM degree distribution:

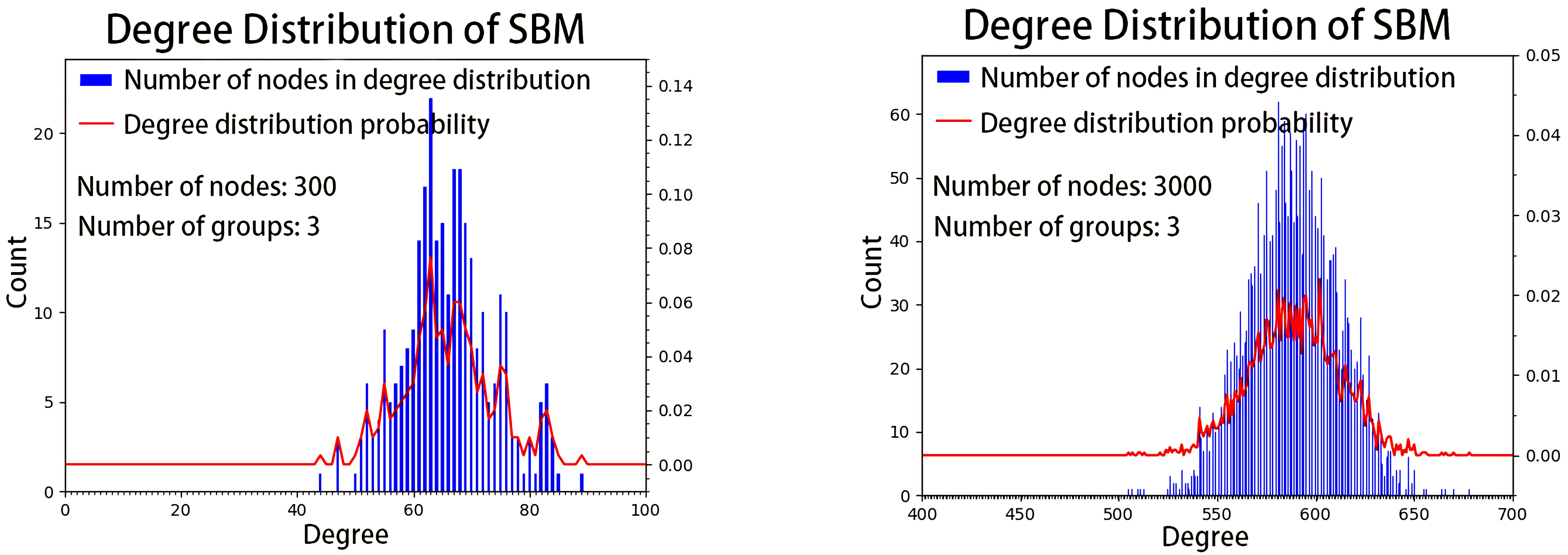


Fig 5: Traditional SBM degree distribution image

- In the random graph, the distribution is:

$$P(\deg(v)=k)=\binom{N-1}{k}p^k(1-p)^{N-1-k} \qquad N\to\infty: P(k)\to\frac{Np^k}{k!}e^{-Np}=\frac{c^k}{k!}e^{-c}$$

CONTENTS

Datasets in Real Networks

Datasets:

Tab 1: Dataset Properties

Datasets	Nodes	Edges	Features	Labels
Cora	2,708	5,278	1,433	7
Citeseer	3,327	4,552	3,703	6
Pubmed	19,717	44,324	500	3
OGB-arxiv	169,343	1,166,234	128	40

- ***Cora***: A subset of the scientific and technical literature citation network.
- ***Citeseer***: A part of papers from Digital Paper Library.
- ***Pubmed***: The publications from the Pubmed database.
- ***OGB-arxiv***: A data set for machine learning on graphs.

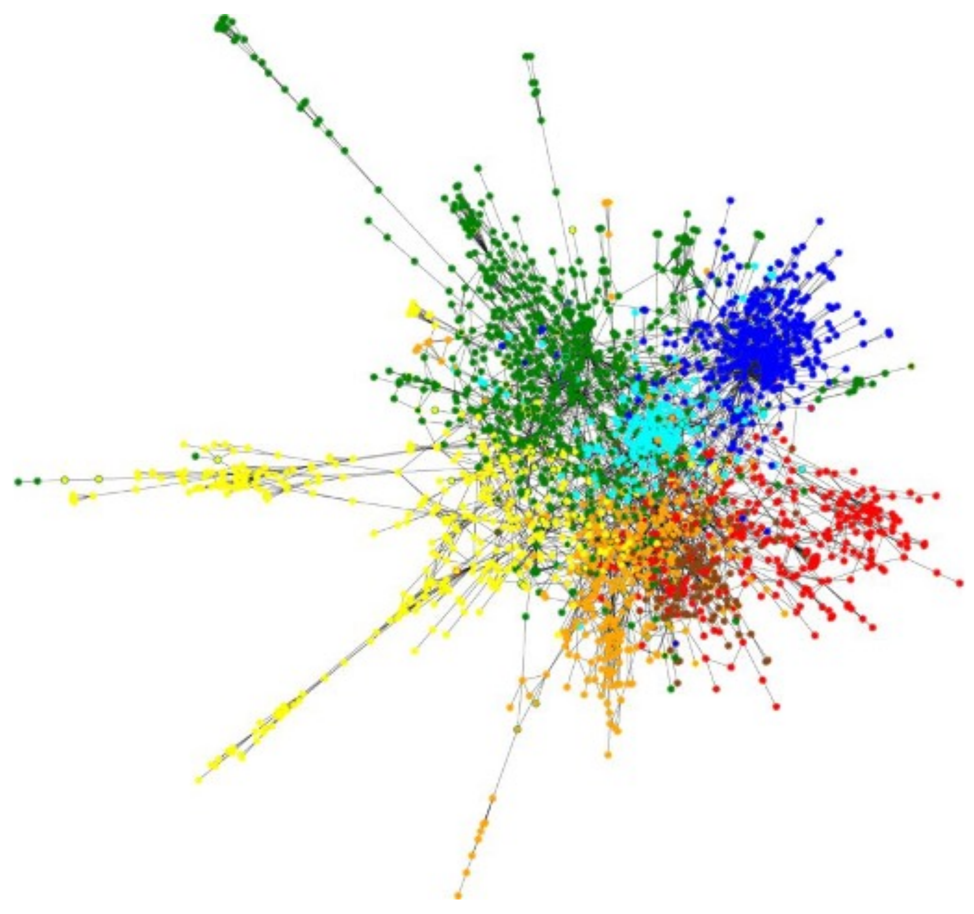


Fig 6: Cora dataset

- Common real benchmark data sets are converted into data information for storage by *PyG*.
- Using *networkx* to construct graph network of data and calculate its degree distribution.

Degree Distribution:

- Scale-free network: Most nodes in the network are connected to few nodes, and very few nodes are connected to very many nodes.
- The real network joins new nodes over time, and the earlier the nodes appear, the easier it is to connect with other nodes. (The rich get richer)
- Power law distribution:

$$P(k) \sim k^{-\gamma}$$

$$\lg P(k) \sim -\gamma \lg k$$

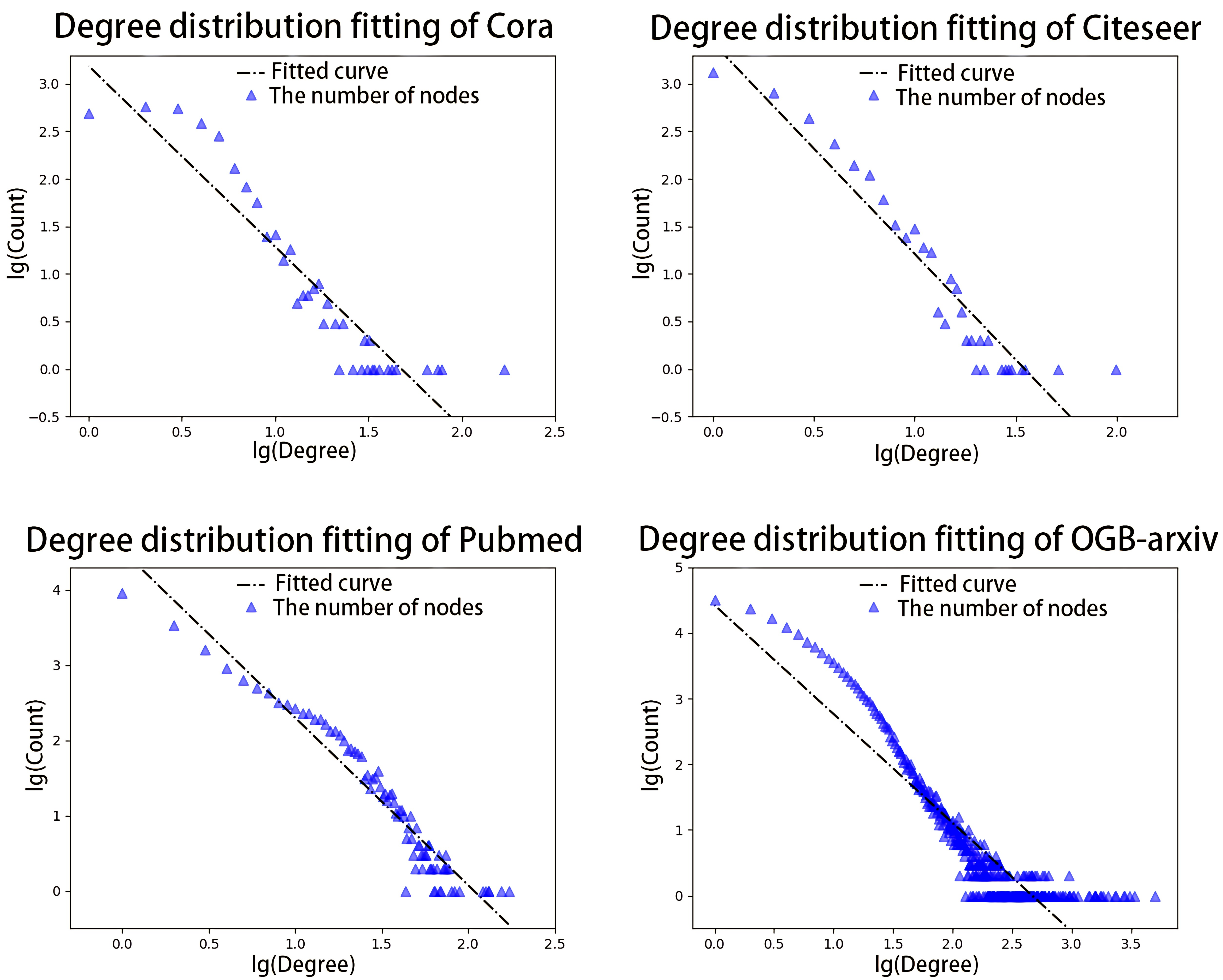


Fig 7: Real graph network degree distribution

CONTENTS

Construction of DCSBM:

SBM: $$P(A_{ij}=1)=P_{ab}=\frac{C_{ab}}{N}=\begin{cases}\frac{c_{in}}{N}(t_i=t_j)\\ \frac{c_{out}}{N}(t_i\neq t_j)\end{cases}$$

Node weight

DCSBM: $$P(A_{ij}=1)=\frac{\boxed{\theta_i\theta_j}C_{ab}}{N}=\begin{cases}\frac{\theta_i\theta_j c_{in}}{N}(t_i=t_j)\\ \frac{\theta_i\theta_j c_{out}}{N}(t_i\neq t_j)\end{cases}$$

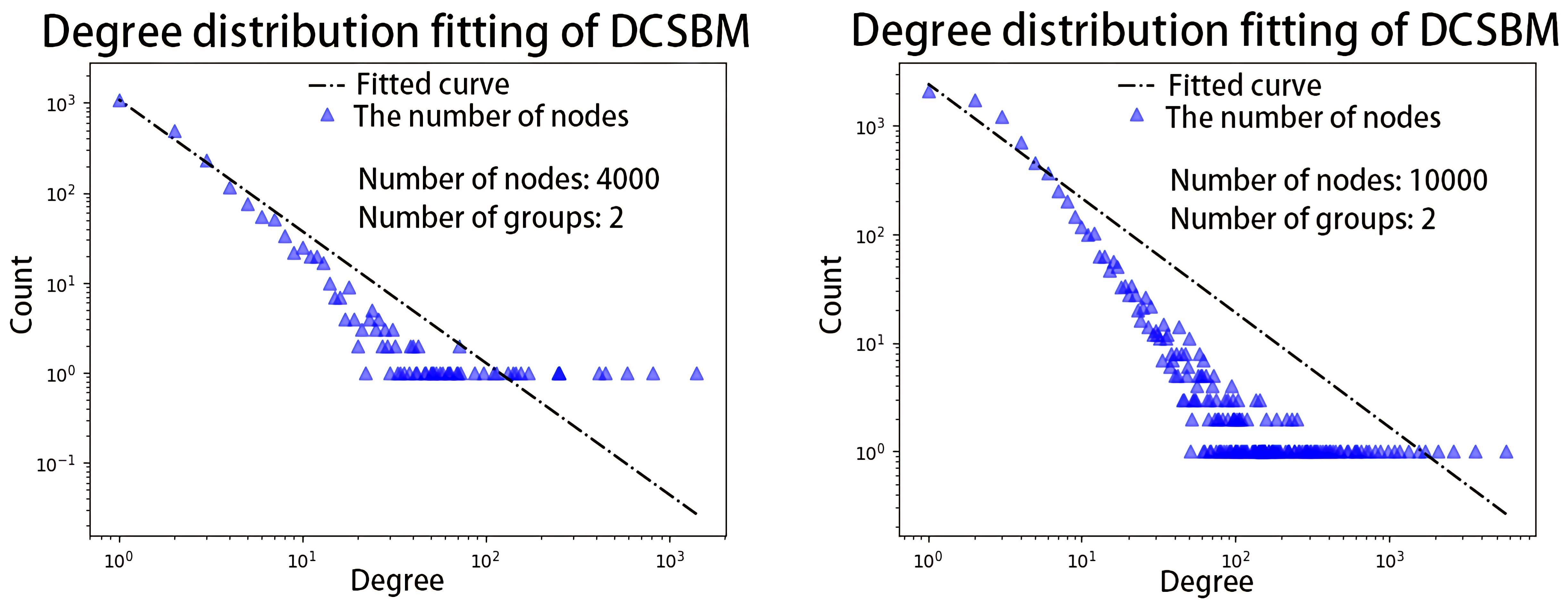


Fig 8: DCSBM based on random power-law sequences

Double Constraint:

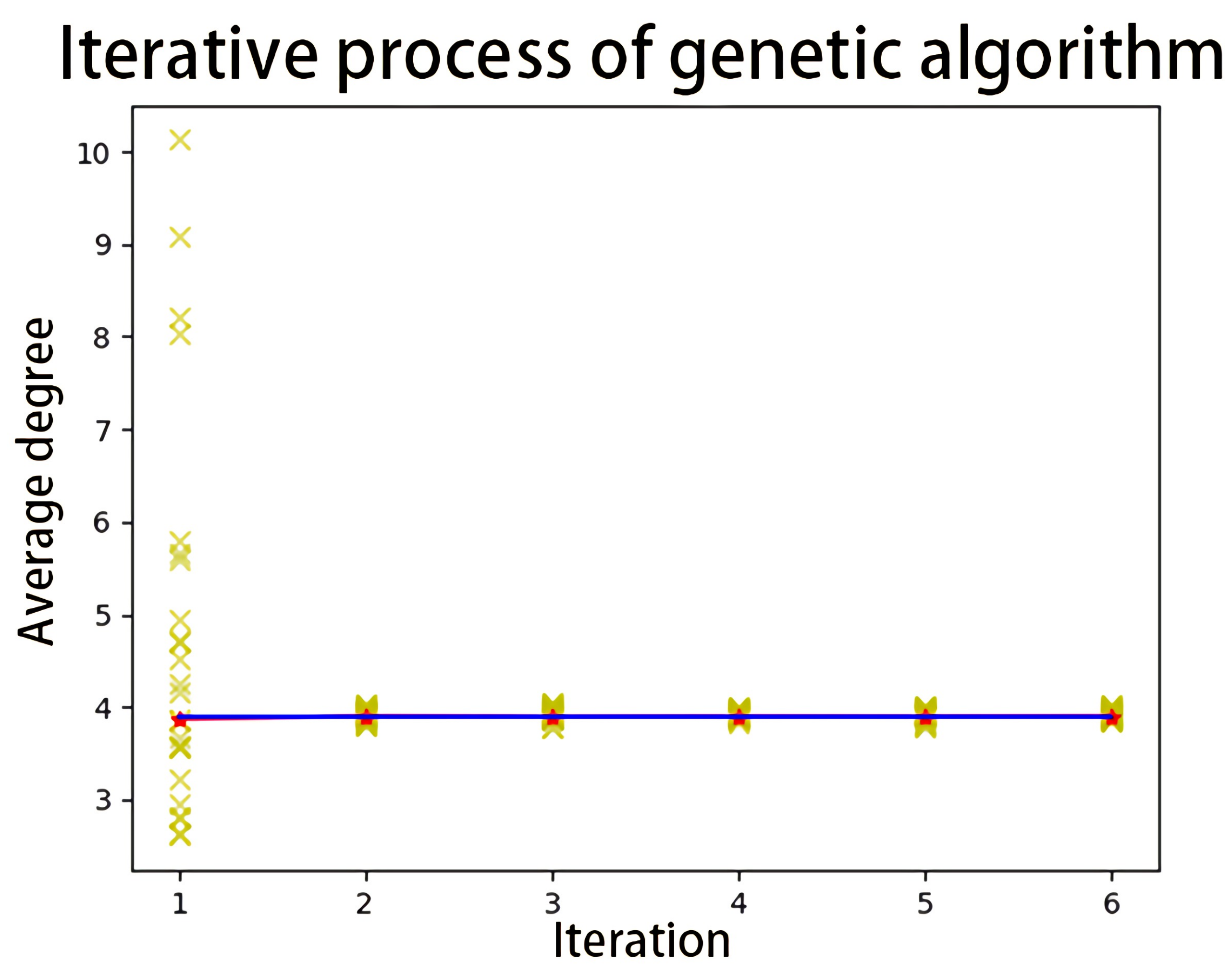
- Intra-class and inter-class connection probability P_{ab} and node power-law weights θ.

- A random set of power-law sequences as weights θ.
- Disadvantage: The larger the degree value, the smaller the weight.

CONTENTS

Construction of DCSBM by GA:

- Extract degree values in real network nodes, randomly shuffle and constrain as $\{\theta_u/x\}_{u=1}^{N}$.



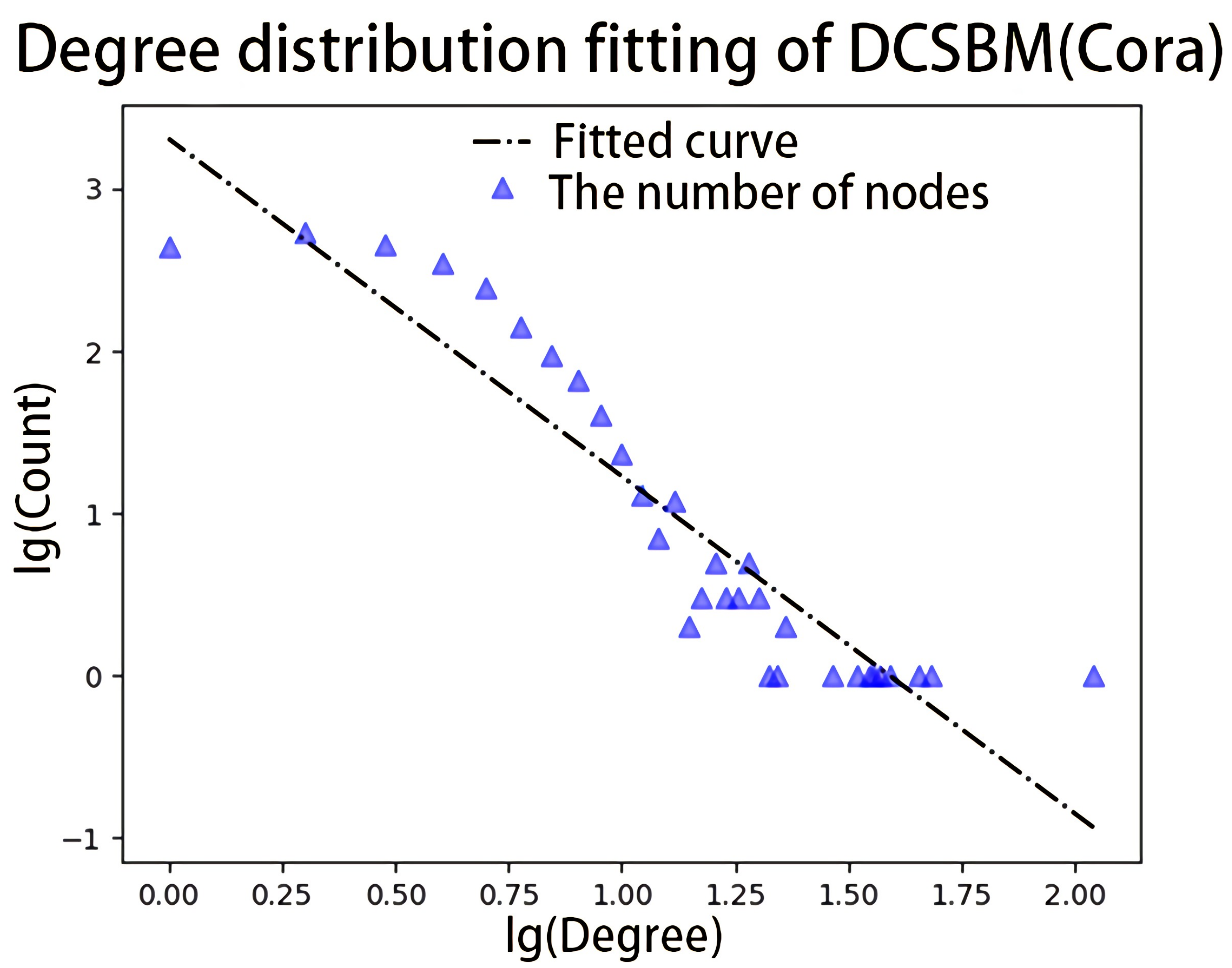


Fig 9: DCSBM based on Genetic Algorithm

- Treat the constraint parameter x as the population, the graph $G(x)$ constructed by extract θ from one individual dataset.
- The fitness function is:

$$f(x) = \left| \frac{1}{N} \sum_{i=1}^{N} deg(i) - c' \right|, i \in G(x)$$

- For the convenience of calculation, the final constraints are:

$$\frac{1}{N} \sum_{i=1}^{N} \theta_i = 1$$

$$\frac{1}{N} \sum_{i=1}^{N} \theta_i^2 = \phi = O(1)$$

CONTENTS

Node Classification Algorithm:

- Bayesian algorithm applied to graphs:

$$P(\{t_i\} \mid G,\theta) = \frac{P(G \mid \{t_i\},\theta)\, P_0(\{t_i\})}{\boxed{\sum_{t_i} P(G \mid \{t_i\},\theta)\, P_0(\{t_i\})}}$$

NP hard problem

- The summation term in the denominator needs to be solved by Boltzmann distribution.

How to solve?

- The KL divergence of ***Belief Propagation*** is similar to Bayesian algorithm.

Tab 2: Belief propagation algorithm formulas

	SBM	DCSBM
Auxiliary External Field	$h_{t_i} = \frac{1}{N}\sum_k \sum_{t_k} p_{t_k t_i}\psi_{t_k}^k$	$h_{t_i} = \frac{1}{N}\sum_k \sum_{t_k} \theta_k\theta_i p_{t_k t_i}\psi_{t_k}^k$
Belief Propagation	$\psi_{t_i}^{i\to j} = \frac{1}{Z^{i\to j}} n_{t_i} e^{-h_{t_i}} \prod_{k\in\partial i\backslash j}\left[\sum_{t_k} p_{t_i t_k}\psi_{t_k}^{k\to i}\right]$	$\psi_{t_i}^{i\to j} = \frac{1}{Z^{i\to j}} n_{t_i} e^{-h_{t_i}} \prod_{k\in\partial i\backslash j}\left[\sum_{t_k} p_{t_i t_k}\theta_k\theta_i\psi_{t_k}^{k\to i}\right]$
Marginal Probability	$\psi_{t_i}^{i} = \frac{1}{Z^{i}} n_{t_i} e^{-h_{t_i}} \prod_{j\in\partial i}\left[\sum_{t_j} p_{t_j t_i}\psi_{t_j}^{j\to i}\right]$	$\psi_{t_i}^{i} = \frac{1}{Z^{i}} n_{t_i} e^{-h_{t_i}} \prod_{j\in\partial i}\left[\sum_{t_j} p_{t_j t_i}\theta_k\theta_i\psi_{t_j}^{j\to i}\right]$

- The BP algorithm was developed to deal with the spin glass phase, that is, to define the edge probability from the point of physics.

CONTENTS

4 Phase Transition

Derivation of Phase Transitions:

- The BP transfer process will eventually converge to a fixed point

$$\psi_{t_i}^{i\to j} = \psi_{t_i}^{i} = \alpha_{t_i}.$$

- Impose a disturbance η_t^k on the leaf node. The influence T of the disturbance message passing and the disturbance on the whole graph are:

$$T_{t_i t_j}^{i\to j} = \left.\frac{\partial \psi_{t_i}^{j\to x}}{\partial \psi_{t_i}^{i\to j}}\right|_{\alpha_{t_i}} = \alpha_{t_i}\left(\frac{np_{t_i t_j}}{c} - 1\right) \qquad \eta_{t_0}^{k_0} = \sum_{\{t_i\} i=1,\ldots,d} \left[\prod_{i=0}^{d-1} T_{t_i t_j}^{i\to j}\right] \eta_{t_d}^{k_d} = T^d \eta^{k_d} \approx \lambda^d \eta^{k_d}$$

- The variance of the disturbance:

SBM: $$\left\langle \left(\eta_{t_0}^{k_0}\right)^2 \right\rangle \approx \left\langle \left(\sum_{k=1}^{c^d} \lambda^d \eta_t^k\right)^2 \right\rangle \approx \boxed{c^d \lambda^{2d}} \left\langle \left(\eta_t^k\right)^2 \right\rangle$$

DCSBM: $$\left\langle \left(\eta_{t_0}^{k_0}\right)^2 \right\rangle \approx \left\langle \left(\sum_{k=1}^{c\phi^d} \lambda^d \eta_t^k\right)^2 \right\rangle \approx \boxed{(c\phi)^d \lambda^{2d}} \left\langle \left(\eta_t^k\right)^2 \right\rangle$$

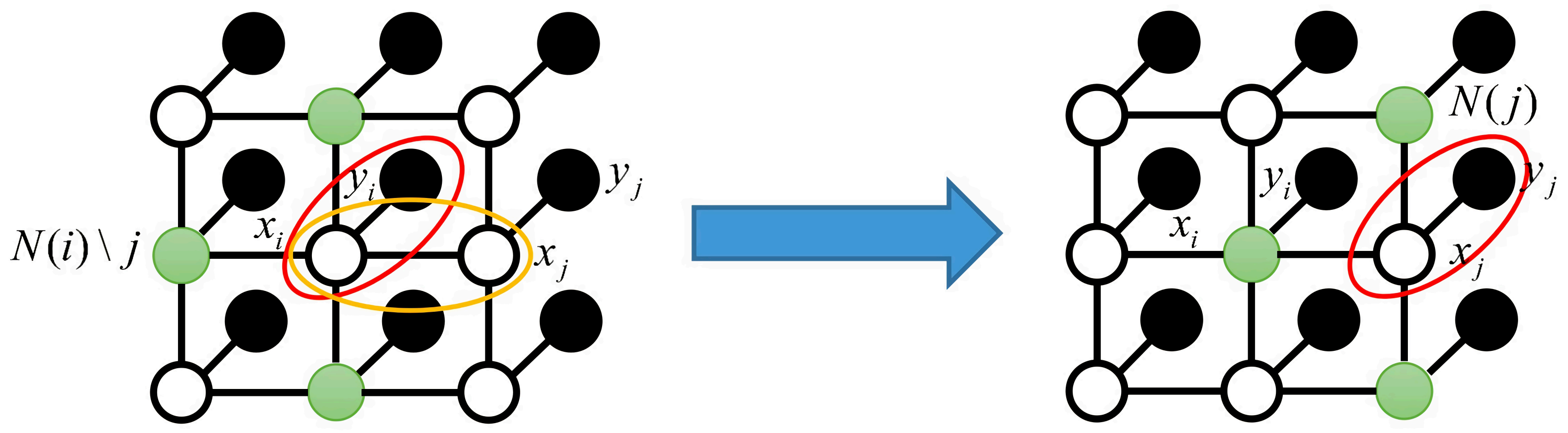
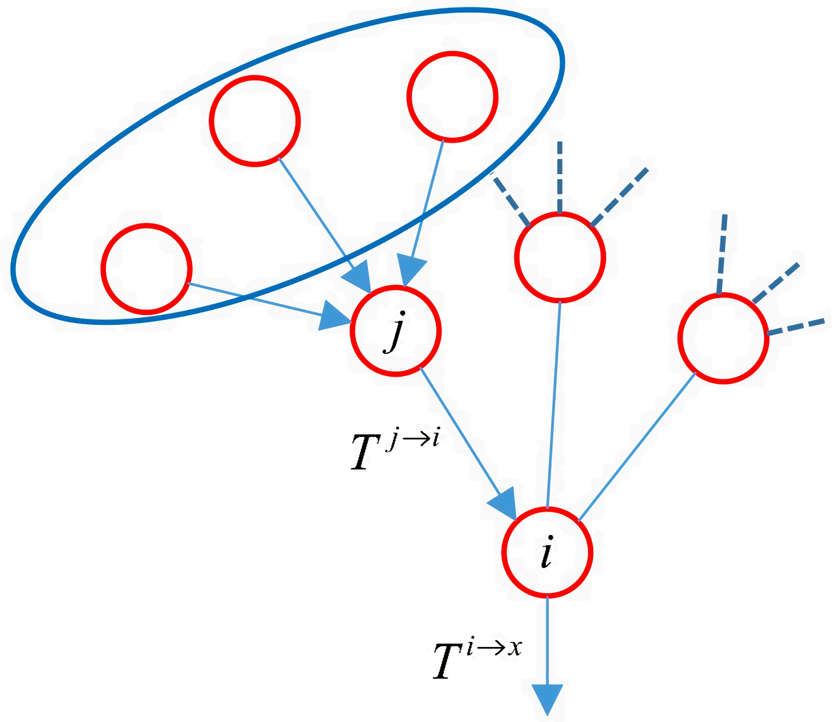


Fig 10: BP algorithm process



The number of leaf nodes:

- **SBM:** c
- **DCSBM:** $c\phi$

Fig 11: The tree structure of SBM or DCSBM

The Phase Transition

SBM: $c\lambda^2 = 1$ DCSBM: $c\phi\lambda^2 = 1$

CONTENTS

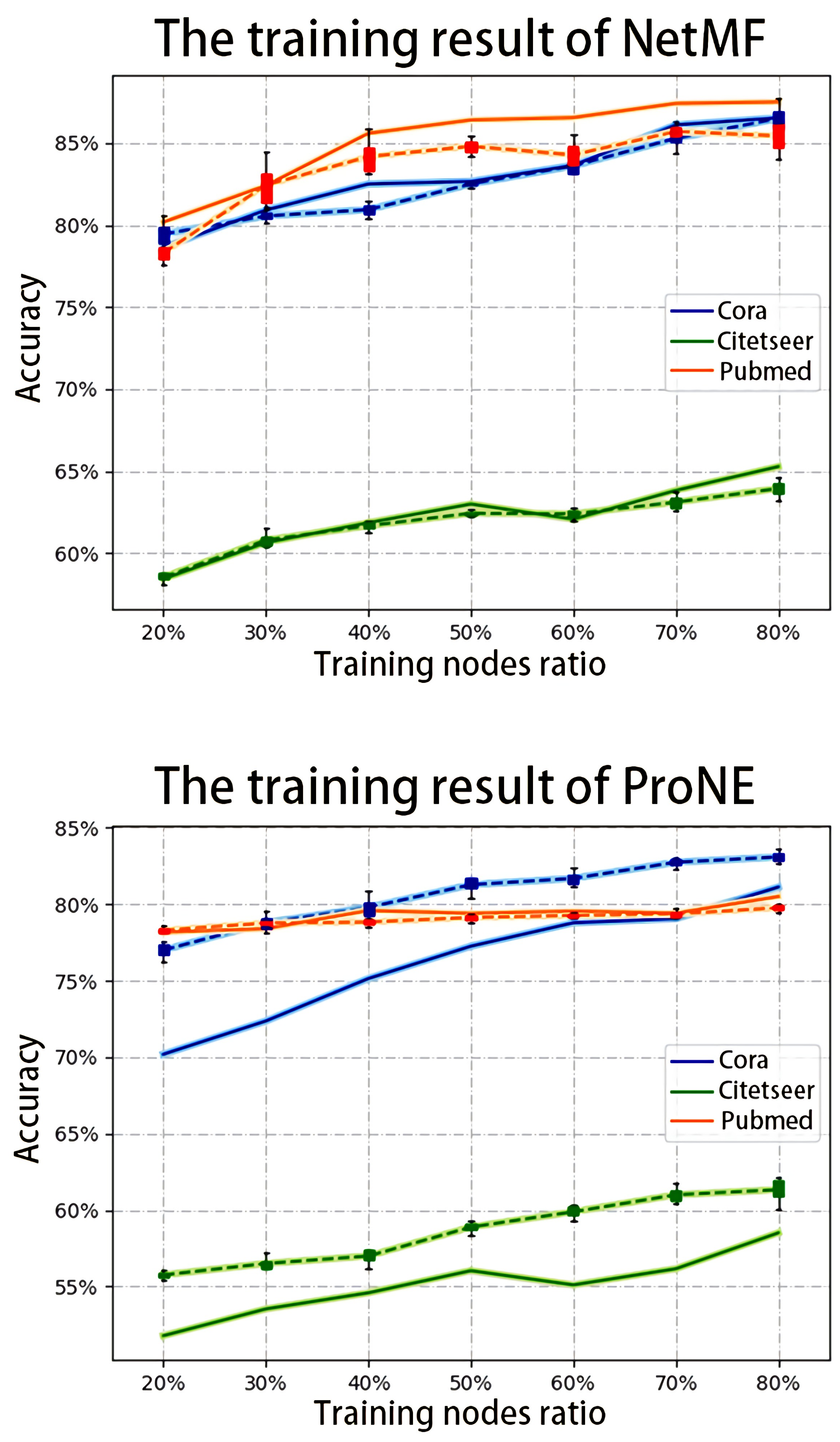


Fig 12: Graph embedding algorithm test

Q1: How to detect DCSBM approximation?

Datasets:

- Cora, Citeseer and Pubmed.

Method:

- NetMF and ProNE.

Idea:

- The results will be approximate when we use the same algorithm testing on models and datasets if they are similar enough.

Result:

- The node classification accuracy of the real dataset is basically the same as that of DCSBM.
- Real datasets are more complex than DCSBM.

Q2: How to prove the phase transition?

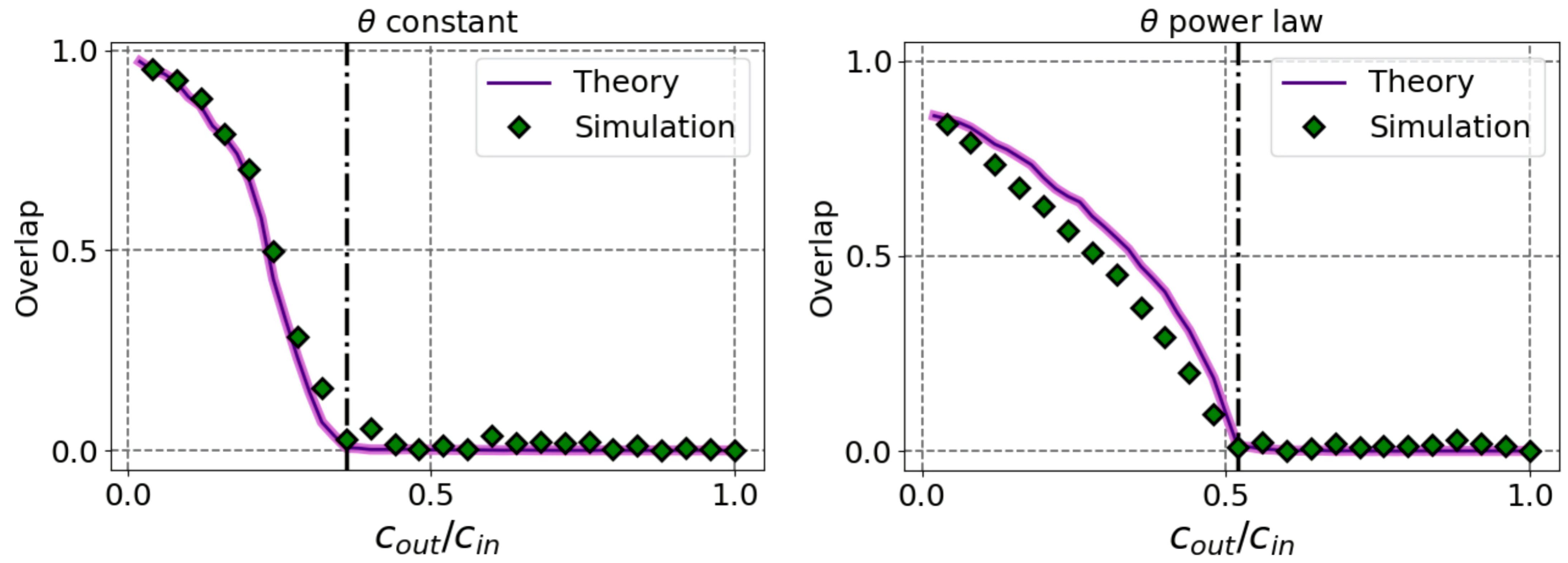


Fig 13: Graph embedding algorithm test

- The solid line is the theoretical value obtained in the BP algorithm. The green dot is the test result of the actual model.
- The position corresponding to the black dotted line is the phase transition critical.

Q3: How to test the effect of parameters on the model?

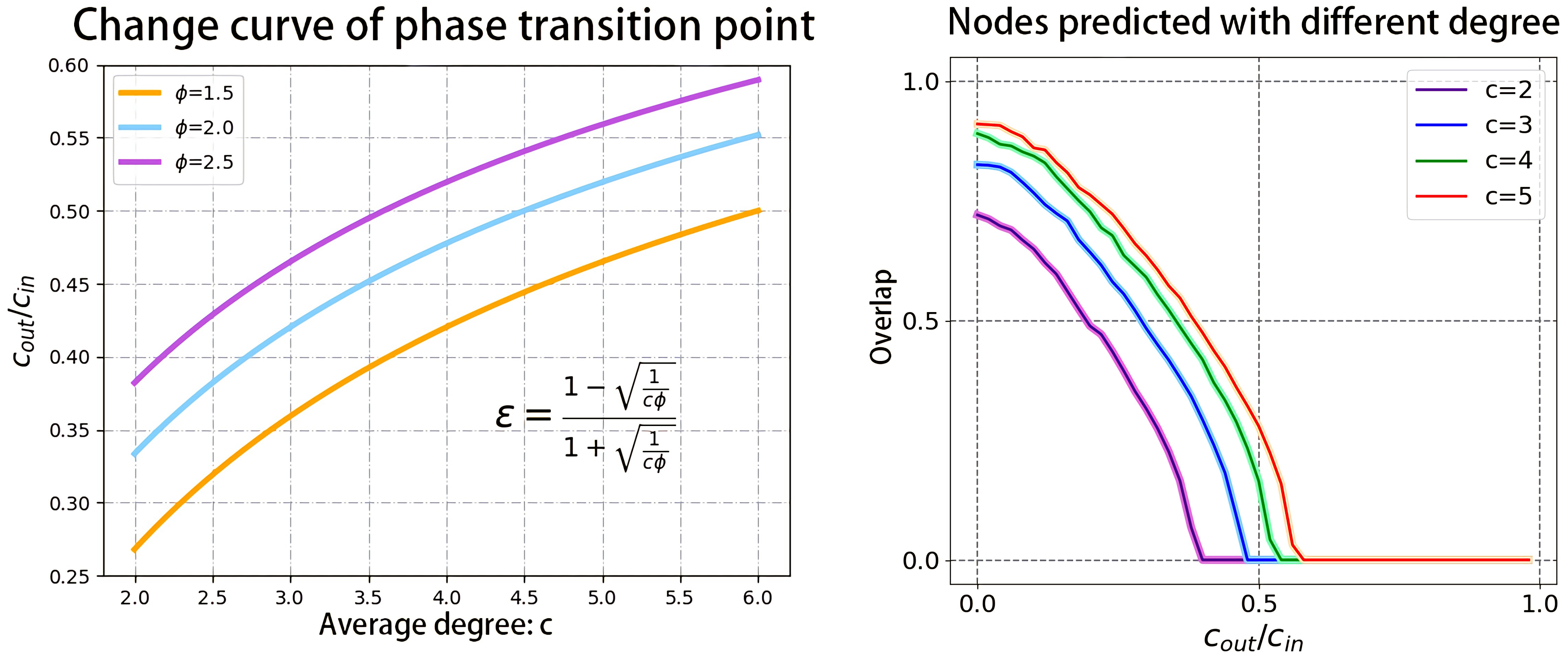


Fig 14: Graph embedding algorithm test

- When the weight θ is fixed and the average degree value c increases, the node classification accuracy improves.
- When the second moment ϕ increases, the probability ratio ε of the phase transition point increases.

CONTENTS

Conclusion:

- The differences and approachable directions of traditional SBM and real datasets are analyzed and compared.
  - SBM: Poisson distribution.
  - Real graphs: Power law distribution
- We build DCSBM in two ways, and the comparison test is done with the real datasets.
  - Random sequence: Need to be constrained by phase transition.
  - Genetic Algorithm: More accurate but more complex.
- The phase transition of DCSBM is derived by using BP algorithm and stochastic process

Thanks