Canadian Association of Physicists Congress 2014
Laurentian University, Sudbury, Ontario
2014.06.16–20

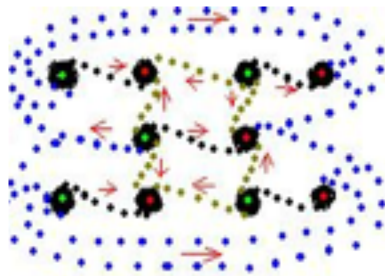
# Feedback Vertex Set
## spin glass theory & algorithms

Hai-Jun Zhou ( 周海军 )

Institute of Theoretical Physics,
Chinese Academy of Sciences,
(中国科学院理论物理研究所)
Beijing

# Trends of StatPhysics Researches

macroscopic collective phenomena <—> microscopic interactions



Statistical Physics

**Simple → Complex**

heterogeneous & competing interactions,

multiple time scale,

glasses, colloids, ...

complex networks, ...

**Equilibrium→ non-equilibrium**

Fluctuation relations,

active matter, ...

ageing dynamics, ...

**Physics→ Interdisciplinary**

game process,

computational complexity,

optimization, ...

# Spin Glass & Optimization

Fu, Anderson (1986):
Traveling Salesman Problem, Graph Partitioning

Kirkpatrick, Gelatt, Vecchi (1983):
*Simulated Annealing*

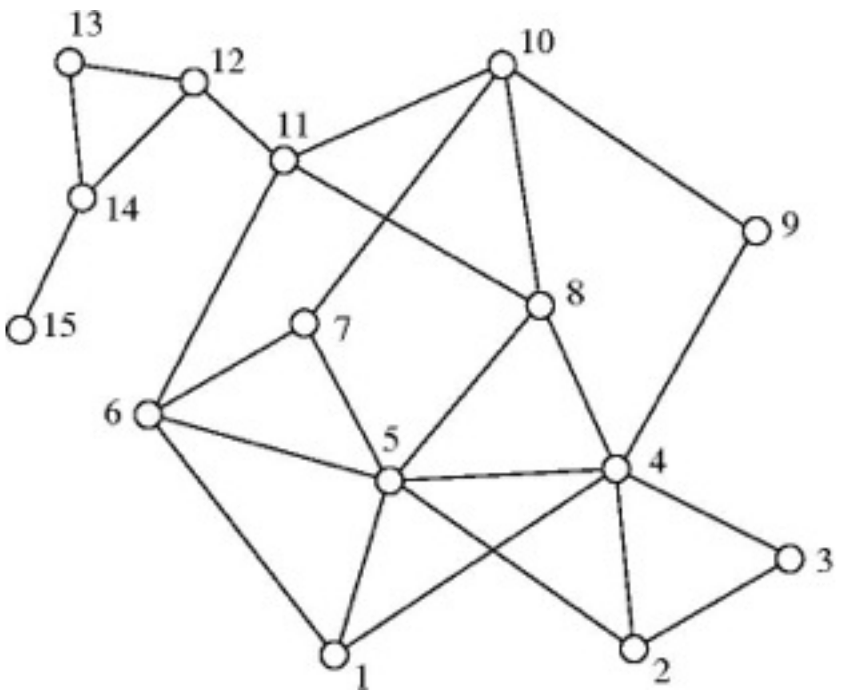
Mezard, Parisi, Zecchina (2002):
K-SAT & Survey-Propagation

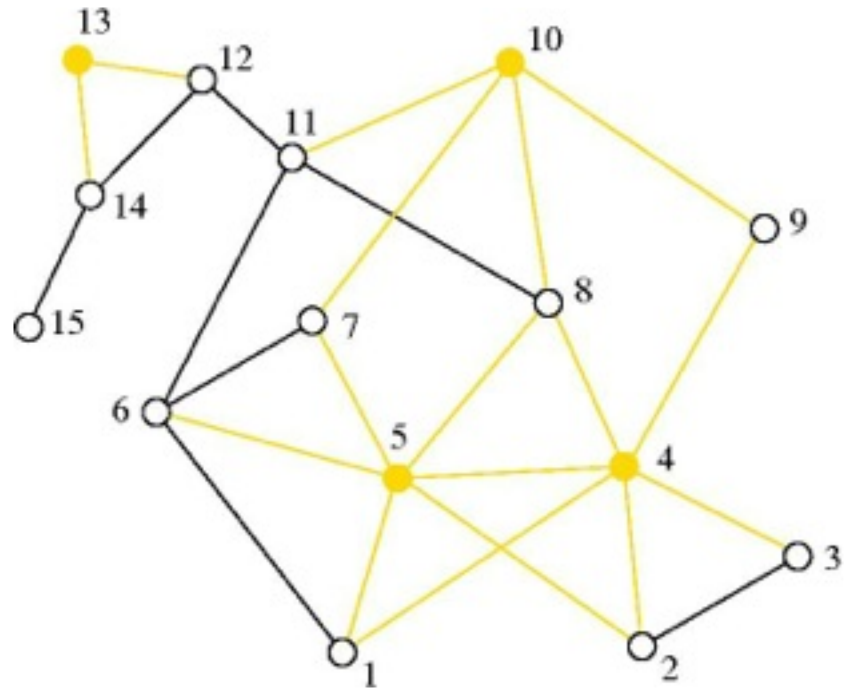
Mezard, Montanari (2009):
"Information, Physics, and Computation" (Oxford Univ Press)

# Feedback Vertex Set

# (a NP-hard problem with global constraints)

# Undirected graph:
# What is a FVS?




A feedback vertex set (FVS): $\{4, 5, 10, 13\}$

- FVS:

  a set of vertices, removal of which breaks all cycles (loops)

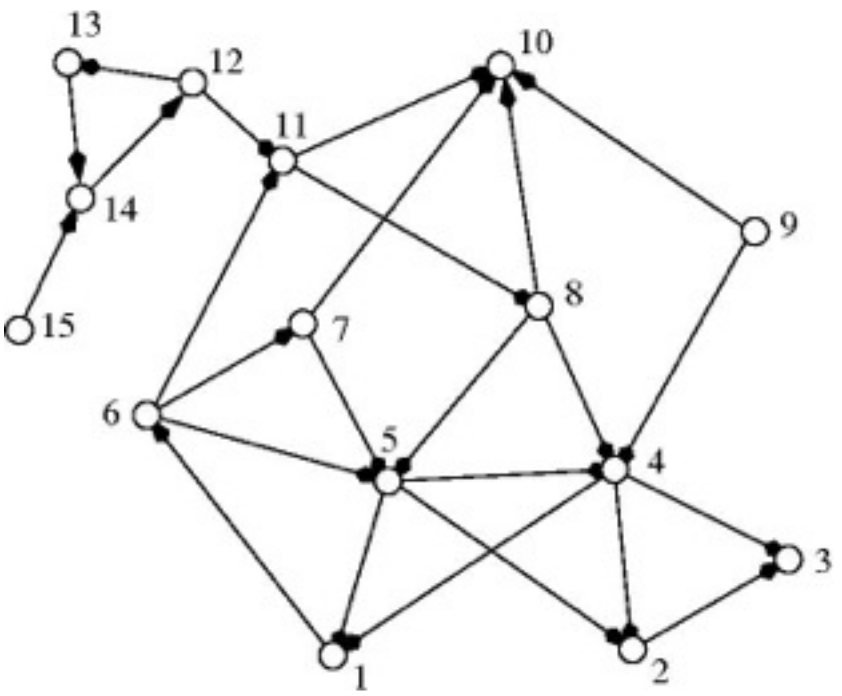
- **minimum FVS:**

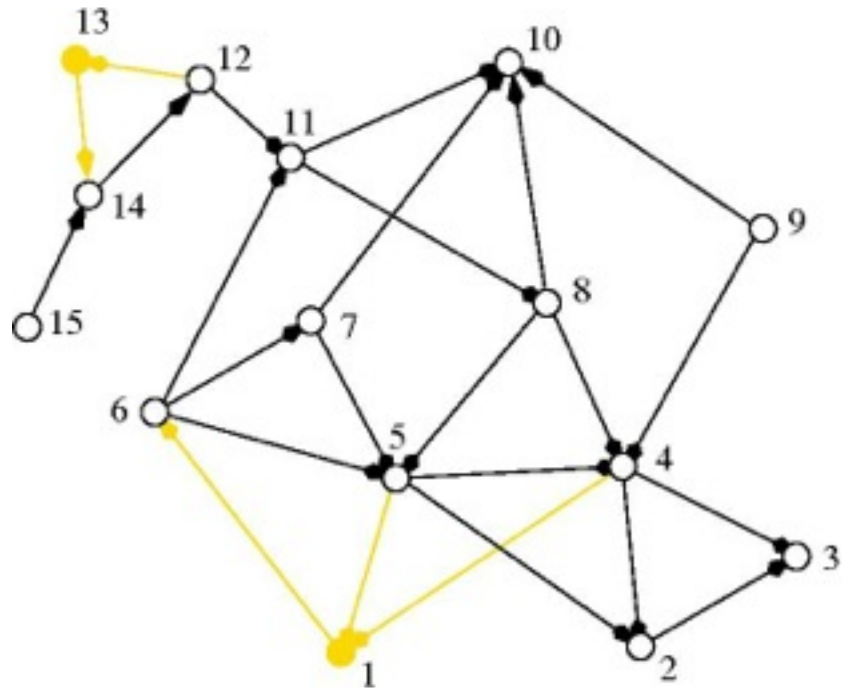
  FVS of global minimum size or total vertex-weight

- NP-complete problem:

  unlikely to have an guaranteed efficient algorithm

# Directed graph: What is a FVS?




A feedback vertex set (FVS): $\{1, 13\}$

- FVS:

  a set of vertices, removal of which breaks all ***directed*** cycles

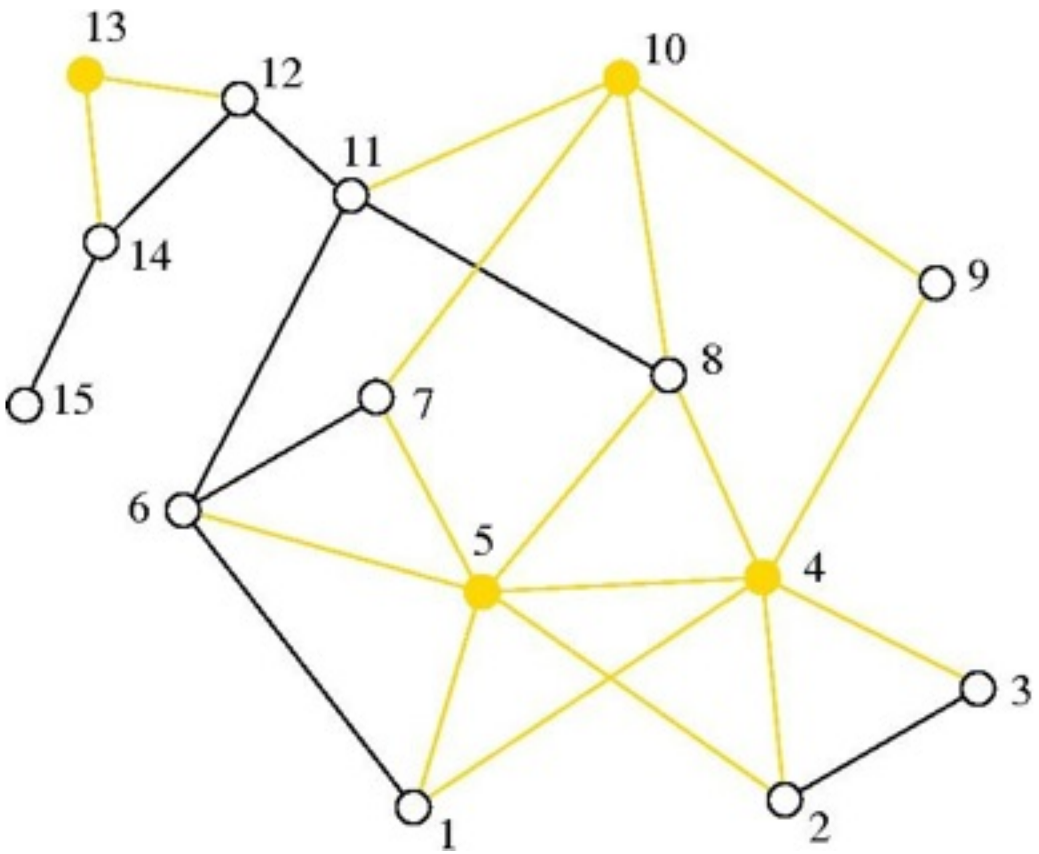
- **minimum FVS:**

  FVS of global minimum size or total vertex-weight

- NP-complete:

  *try not for best solution, but for better solution*

# Vertices in minimum FVS dynamically important




1. FVS as boundary
2. Given a boundary state, dynamics of each tree component easy to obtain
3. FVS vertices cause feedback to dynamics

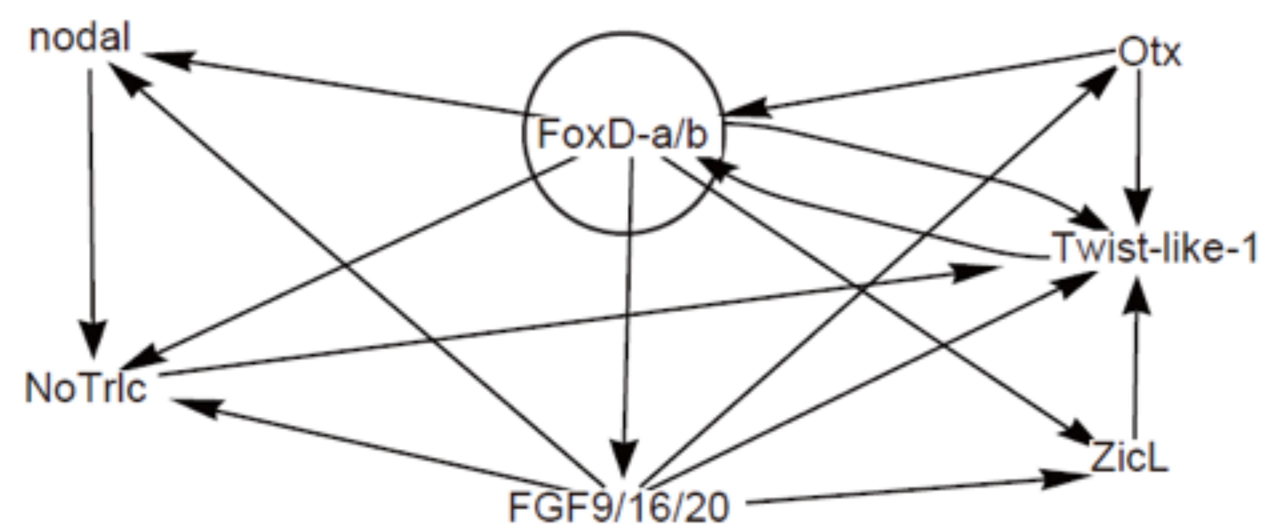
# Feedback vertex sets as informative and determining nodes of regulatory network dynamics

Bernold Fiedler*
Atsushi Mochizuki**
Gen Kurosawa**
Daisuke Saito**

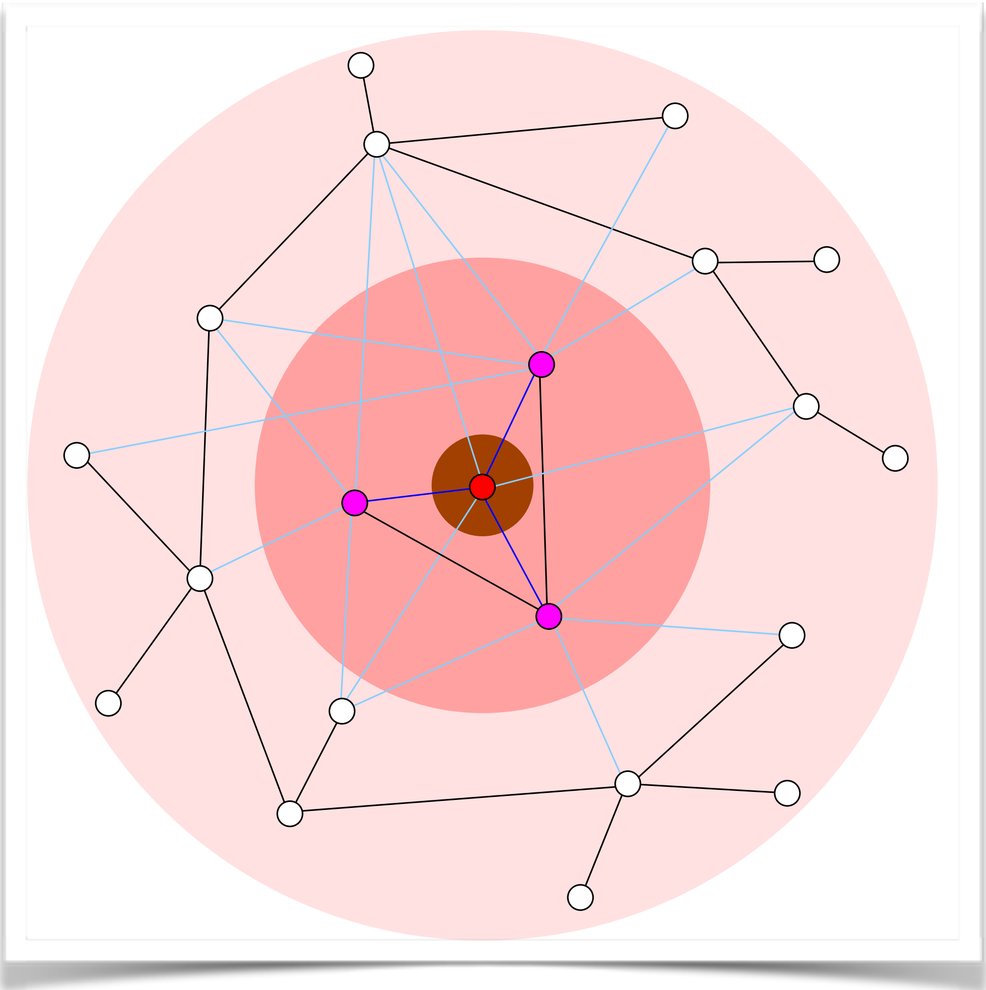
April 23, 2012

An application:

gene/protein interaction



# Decompose a network into different layers



**1st layer:**
The largest forest

**2nd layer:**
The largest forest in min-FVS subgraph

**3rd layer:**
…

## Goal: construct a FVS of (close to) minimum size

why **not** easy from statistical physics approaches?

Main reason:

Statistical physics methods best suited to problems with local (few-body) interactions, ……

…… but, **cycles are global properties**, can not be judged from looking just at single vertices or edges

# Previous algorithmic approach

SIAM J. DISCRETE MATH.
Vol. 12, No. 3, pp. 289–297



## A 2-APPROXIMATION ALGORITHM FOR THE UNDIRECTED FEEDBACK VERTEX SET PROBLEM*

VINEET BAFNA†, PIOTR BERMAN‡, AND TOSHIHIRO FUJITO§

**Abstract.** A feedback vertex set of a graph is a subset of vertices that contains at least one vertex from every cycle in the graph. The problem considered is that of finding a minimum feedback vertex set given a weighted and undirected graph. We present a simple and efficient approximation algorithm with performance ratio of at most 2, improving previous best bounds for either weighted or unweighted cases of the problem. Any further improvement on this bound, matching the best constant factor known for the vertex cover problem, is deemed challenging.

The approximation principle, underlying the algorithm, is based on a generalized form of the classical local ratio theorem, originally developed for approximation of the vertex cover problem, and a more flexible style of its application.

Bafna, Berman, Fujito, SIAM J. Discrete Math. 12, 289-297 (1999).

**Input:** an undirected graph $G = (V, E)$ with vertex weights $w : V \to Q_+$
**Output:** a feedback vertex set $F$

# FEEDBACK

```
Initialize F = {u ∈ V : w(u) = 0}, V = V − F. [i = 0]
Cleanup(G)
While V ≠ ∅ do
    [i ← i + 1]
    If G contains a semidisjoint cycle C, then
        Let γ ← min{w(u) : u ∈ V(C)}.
        Set w(u) ← w(u) − γ, ∀u ∈ V(C).
            [G_i = C and w_i(u) = γ, ∀u ∈ V(C)]
    Else [G is clean and contains no semidisjoint cycle]
        Let γ ← min{w(u)/(d(u) − 1) : u ∈ V}.
        Set w(u) ← w(u) − γ(d(u) − 1), ∀u ∈ V.
            [G_i = G and w_i(u) = γ(d(u) − 1), ∀u ∈ V]
    For each u ∈ V with w(u) = 0 do
        Remove u from V, add it to F, and push it onto STACK.
    Cleanup(G)
While STACK ≠ ∅ do
    Let u ← pop(STACK).
    If F − {u} is an FVS in original G, then [u is redundant]
        Remove u from F.

Cleanup(G):
    While G contains a vertex of degree at most 1, remove it along with
        any incident edges.
```

FIG. 3.1. *2-approximation algorithm* FEEDBACK *for the FVS problem.*

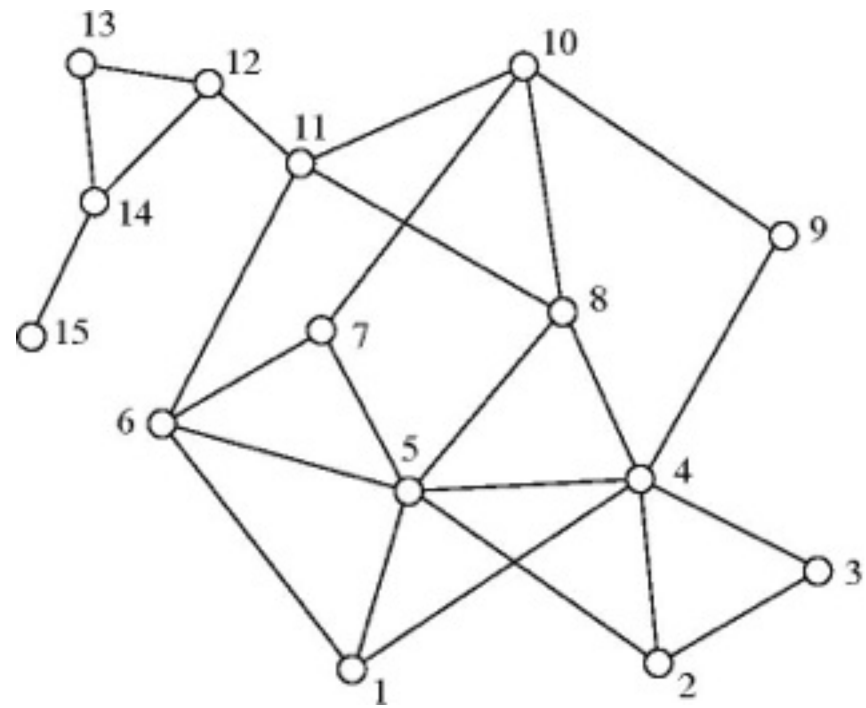
It's a local algorithm, recursively removes vertices and then simplify the graph. (**vertex chosen based on current degree**).

Guarantees that the size of the FVS is
At most two times of
The minimum size
**( so far best guarantee)**

# In this talk

Minimum FVS as spin-glass

- Define a spin model;
- mean-field theory;
- Message-passing algorithm;
- Application to random graphs and lattices




**We consider mainly undirected graphs in this work**
(directed graphs briefly discussed in the end)

# global cycle constraints → local edge constraints

Main observations:

»
Vertex either deleted (belongs to FVS) or remains in graph.

»
If a vertex remains in graph, it must belong to a tree component.

»
For a tree component, after assigning one vertex as root, a parent-child relationship can be defined for any two neighboring vertices.

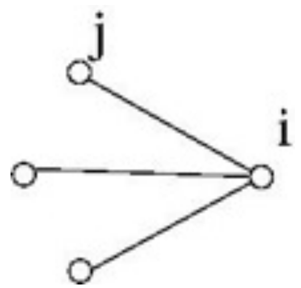
## global cycle constraints → local edge constraints

(a) Define an integer state $A_i$ for each vertex $i$

$A_i = 0$: vertex $i$ unoccupied ($\in FVS$)

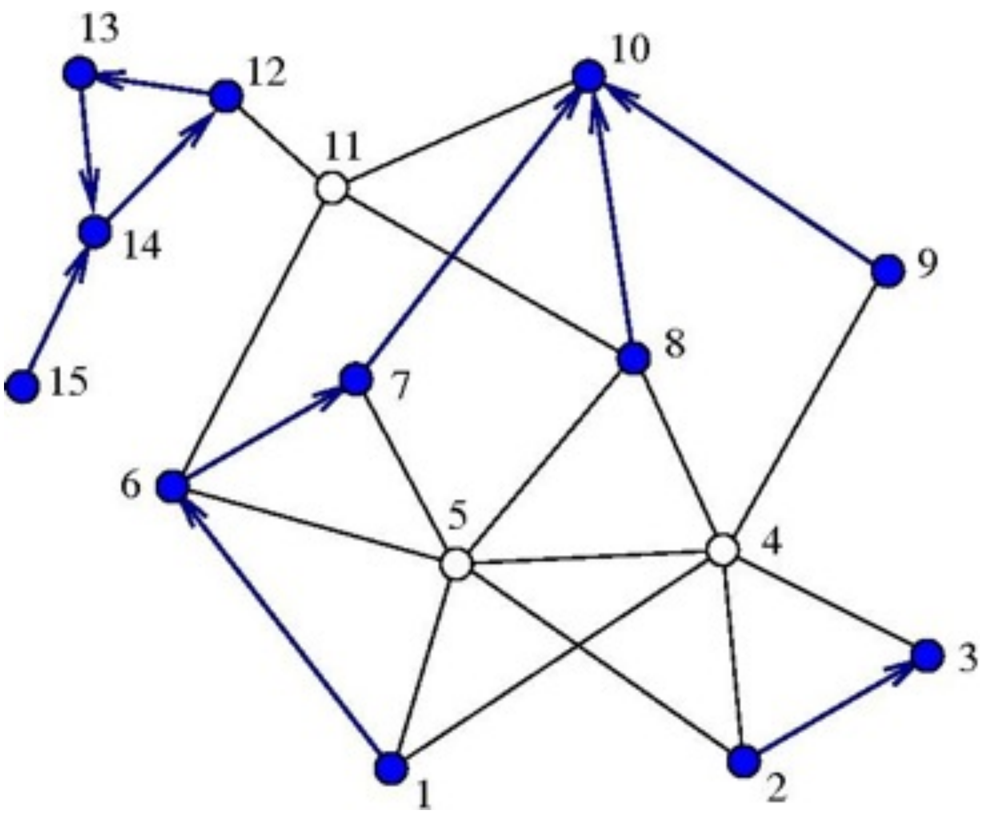
$A_i = i$: vertex $i$ occupied ($\notin FVS$) and is a root of a tree component

$A_i = j \in \partial i$: vertex $i$ occupied ($\notin FVS$) and $j$ is its parent vertex in a tree comp.




$\partial i$: the set of neighbors of vertex $i$

# A graphical representation of states




Closed circle: occupied
open circle: unoccupied
arrow on edge: pointing from child to parent

$A_1 = 6$
$A_2 = 3$
$A_3 = 3$
$A_4 = 0$

$A_5 = 0$
$A_6 = 7$
$A_7 = 10$
$A_8 = 10$

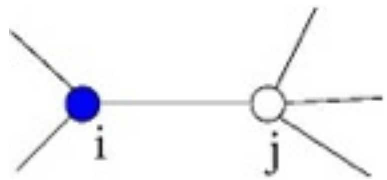
$A_9 = 10$
$A_{10} = 10$
$A_{11} = 0$
$A_{12} = 13$

$A_{13} = 14$
$A_{14} = 12$
$A_{15} = 14$

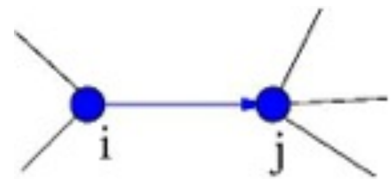
# global cycle constraints → local edge constraints

## (b) Define an edge factor $C_{ij}$ for each edge $(i, j)$

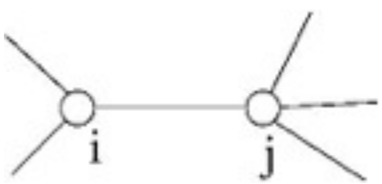
$$C_{ij}(A_i, A_j) \equiv \delta^0_{A_i}\delta^0_{A_j} + \delta^0_{A_i}(1 - \delta^0_{A_j} - \delta^i_{A_j}) + \delta^0_{A_j}(1 - \delta^0_{A_i} - \delta^j_{A_i}) + \delta^j_{A_i}(1 - \delta^0_{A_j} - \delta^i_{A_j}) + \delta^i_{A_j}(1 - \delta^0_{A_i} - \delta^j_{A_i})$$



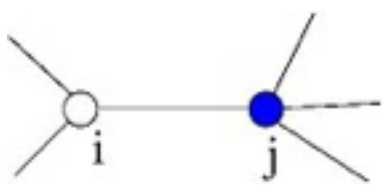

$A_j = 0,$
$A_i = i$ or $A_i = k \neq j$



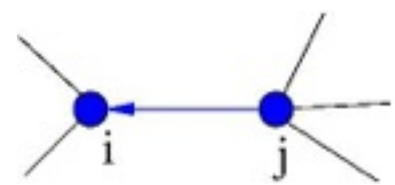

$A_i = j,$
$A_j = j$ or $A_j = l \neq i$




$A_i = 0, A_j = 0$




$A_i = 0,$
$A_j = j$ or $A_j = l \neq i$



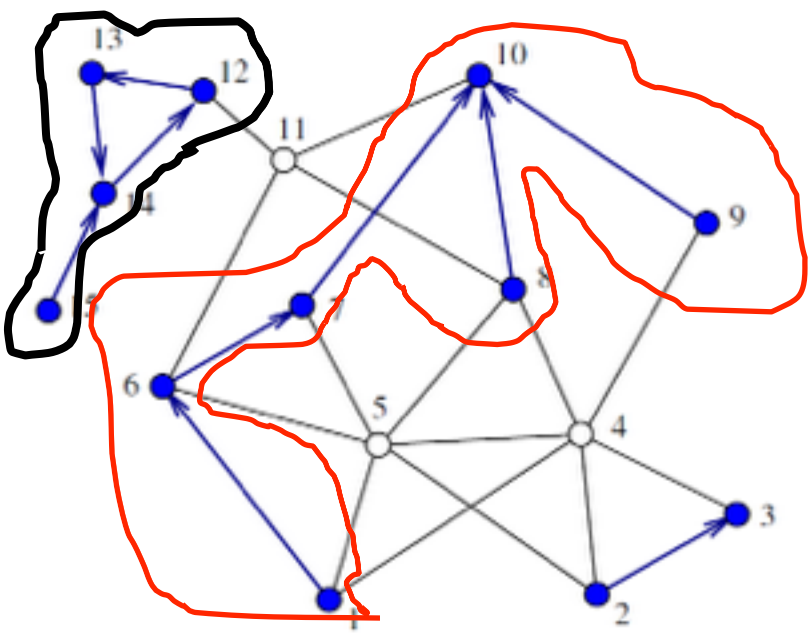

$A_j = i,$
$A_i = i$ or $A_i = k \neq j$

# partition function

$$Z(x) = \sum_{\underline{A}} \exp\left[x \sum_{i=1}^{N} (1 - \delta^0_{A_i}) w_i\right] \prod_{(i,j)\in G} C_{ij}(A_i, A_j)$$

$w_i$: (non-negative) weight of vertex $i$

Only configurations $\underline{A} = \{A_1, A_2, \ldots, A_N\}$ which corresponds to a collection of disjoint trees and c-trees have non-vanishing contribution to $Z(x)$.



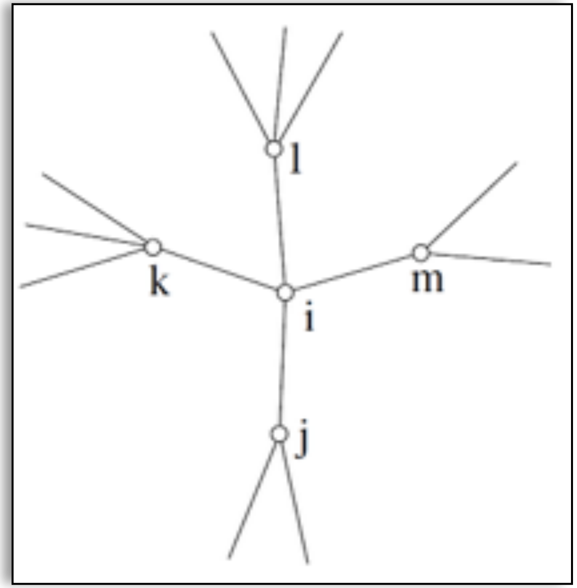

**tree**:
a connected component with $n$ vertices and $n-1$ edges.

**c-tree**:
a connected component with $n$ vertices and $n$ edges (there is one and only one ycle).

# belief propagation equations

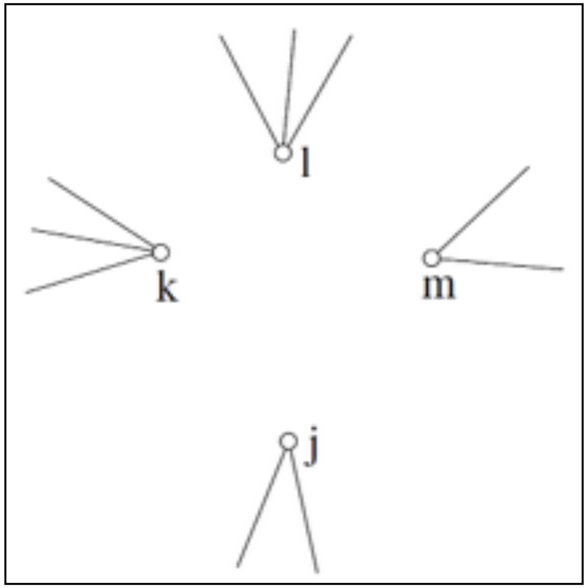
We want to obtain marginal probabilities for each vertex.

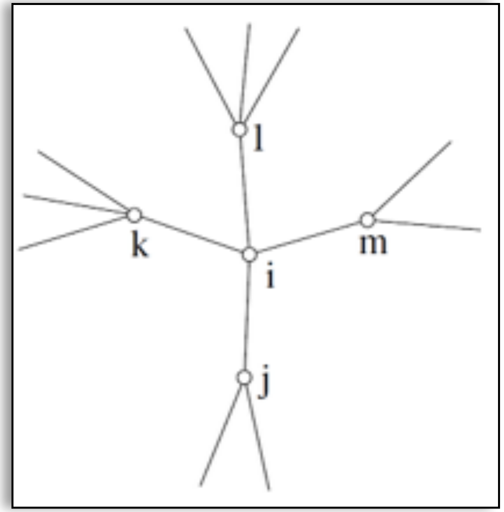
$q_i^{A_i}$ : probability of vertex $i$ to have state $A_i$.




Bethe-Peierls Approximation:

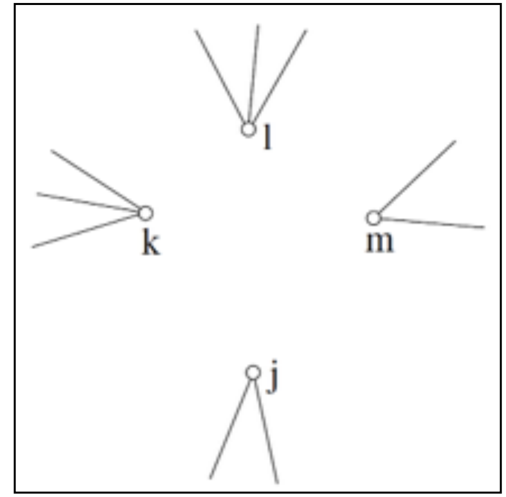
$$P_{\setminus i}\left(\{A_j : j \in \partial i\}\right) \approx \prod_{j \in \partial i} q_{j \to i}^{A_j}$$






**Bethe-Peierls Approximation:**

$$P_{\backslash i}\left(\{A_j : j \in \partial i\}\right) \approx \prod_{j\in\partial i} q_{j\to i}^{A_j}$$




Single vertex marginal probability:

$$q_i^0 = \frac{1}{1 + e^{x w_i}\left[\prod_{j\in\partial i}\left(q_{j\to i}^0 + q_{j\to i}^j\right) + \sum_{j\in\partial i}\left(1 - q_{j\to i}^0\right)\prod_{k\in\partial i\backslash j}\left(q_{k\to i}^0 + q_{k\to i}^k\right)\right]}$$

$$q_i^i = \frac{e^{x w_i}\prod_{j\in\partial i}\left(q_{j\to i}^0 + q_{j\to i}^j\right)}{1 + e^{x w_i}\left[\prod_{j\in\partial i}\left(q_{j\to i}^0 + q_{j\to i}^j\right) + \sum_{j\in\partial i}\left(1 - q_{j\to i}^0\right)\prod_{k\in\partial i\backslash j}\left(q_{k\to i}^0 + q_{k\to i}^k\right)\right]}$$

Belief-propagation (BP) equation:

$$q^0_{i\to j} = \frac{1}{1 + e^{x w_i}\Big[\prod_{k\in\partial i\backslash j}(q^0_{k\to i} + q^k_{k\to i}) + \sum_{k\in\partial i\backslash j}(1 - q^0_{k\to i})\prod_{m\in\partial i\backslash j,k}(q^0_{m\to i} + q^m_{m\to i})\Big]}$$

$$q^i_{i\to j} = \frac{e^{x w_i}\prod_{k\in\partial i\backslash j}(q^0_{k\to i} + q^k_{k\to i})}{1 + e^{x w_i}\Big[\prod_{k\in\partial i\backslash j}(q^0_{k\to i} + q^k_{k\to i}) + \sum_{k\in\partial i\backslash j}(1 - q^0_{k\to i})\prod_{m\in\partial i\backslash j,k}(q^0_{m\to i} + q^m_{m\to i})\Big]}$$

HJZ: European Physical Journal B (2013)

# Computing thermodynamic quantities

Free entropy: $$\Phi(x) = \frac{1}{x}\ln Z(x) = N\,\phi(x)$$

$$\Phi(x) = \sum_{i=1}^{N}\phi_i - \sum_{edge\,(i,j)}\phi_{ij}$$

$$\phi_i = \frac{1}{x}\ln\Big[1 + e^{xw_i}\prod_{j\in\partial i}[q^0_{j\to i} + q^j_{j\to i}] + e^x\sum_{j\in\partial i}(1 - q^0_{j\to i})\prod_{k\in\partial i\backslash j}(q^0_{k\to i} + q^k_{k\to i})\Big],$$

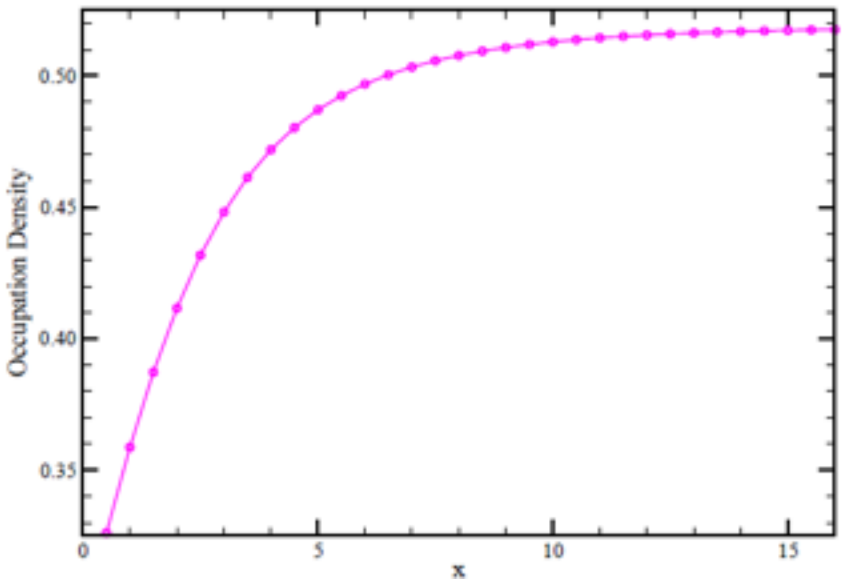
$$\phi_{ij} = \frac{1}{x}\ln\Big[q^0_{i\to j}q^0_{j\to i} + (1 - q^0_{i\to j})(q^0_{j\to i} + q^j_{j\to i}) + (1 - q^0_{j\to i})(q^0_{i\to j} + q^i_{i\to j})\Big].$$

# Results: I

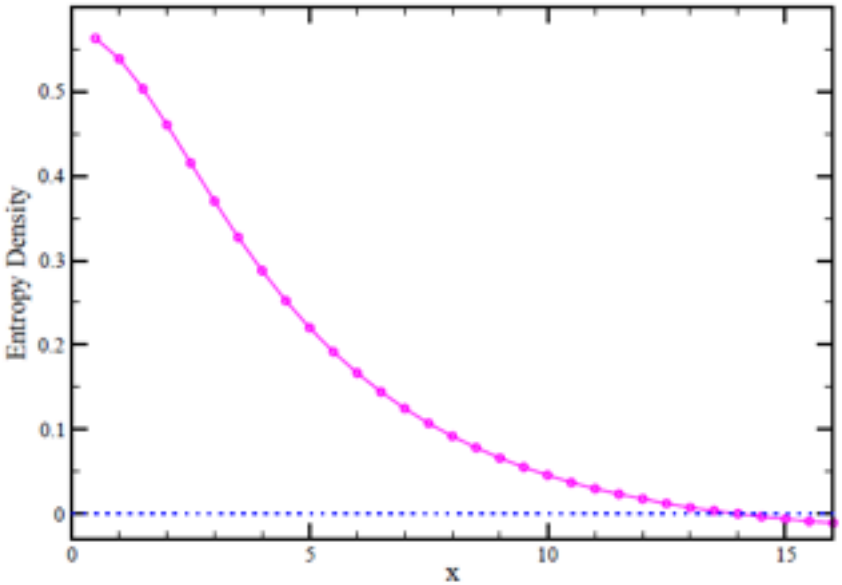
Ensemble-averaging on Erdos-Renyi random graph

**(mean degree $c = 10$)**

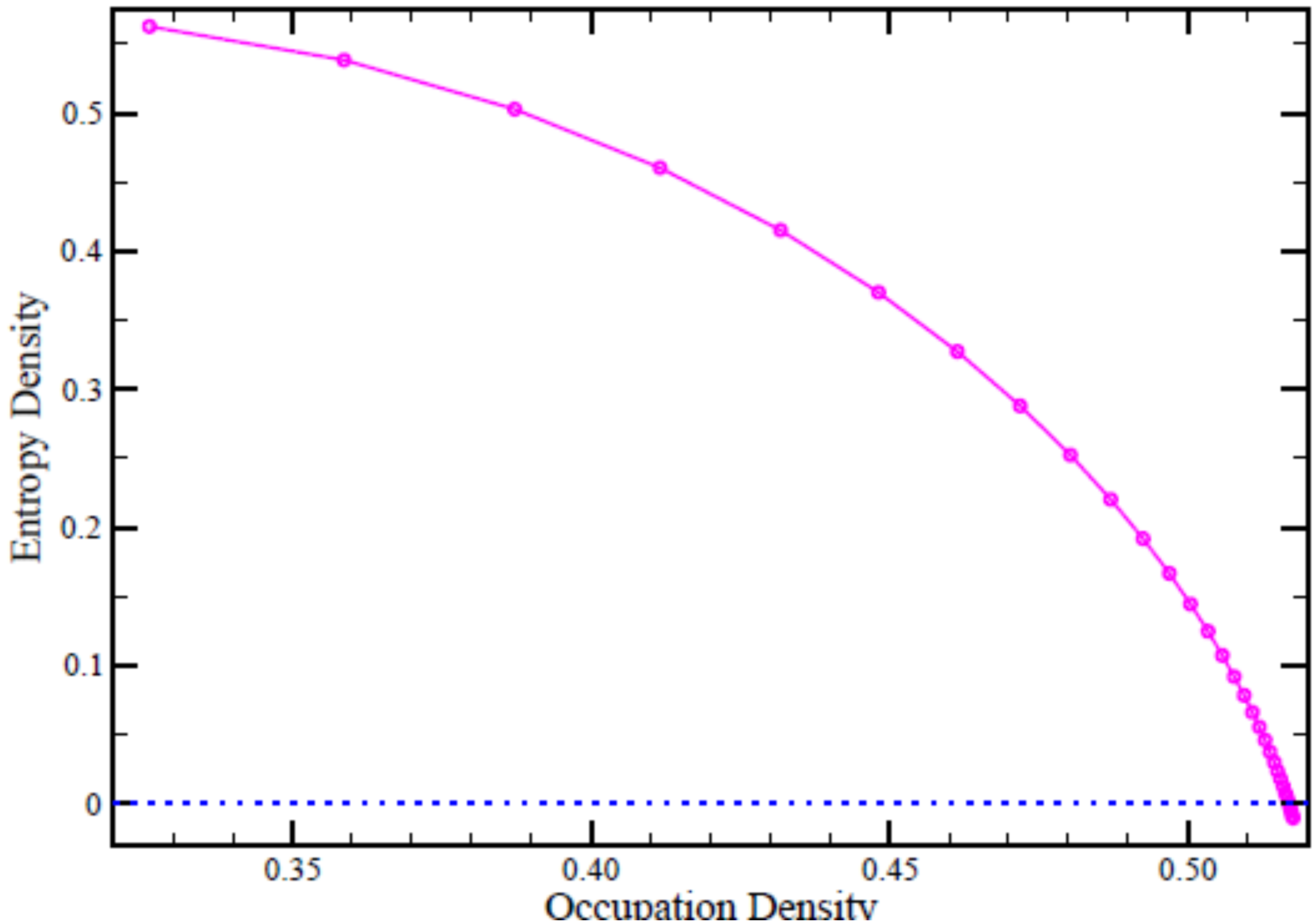
occupation desity




entropy density



# Ensemble-averaging on Erdos-Renyi random graph
## (mean degree $c = 10$)



# belief propagation-guided decimation

(0). Input a graph G and initialize randomly the edge messages $(q^0_{i\to j}, q^i_{i\to j})$ and $(q^0_{j\to i}, q^j_{j\to i})$ for each edge $(i, j)$ of the graph $G$. The feedback vertex set $\Gamma$ is initialized to be empty. The re-weighting parameter $x$ is set to an appropriate value.

(1). Perform the BP iteration process a number $T$ of rounds (in each round of the iteration, the vertices of the graph $G$ are randomly ordered and their output messages are then updated). We then compute the empty probability $q^0_i$ of each vertex $i$ of the graph $G$ based on the current inputting messages to vertex $i$. Then the $fN$ vertices with the highest empty probability values are added to the set $\Gamma$, and these vertices are then removed from the graph $G$ together with all the edges attached to them.

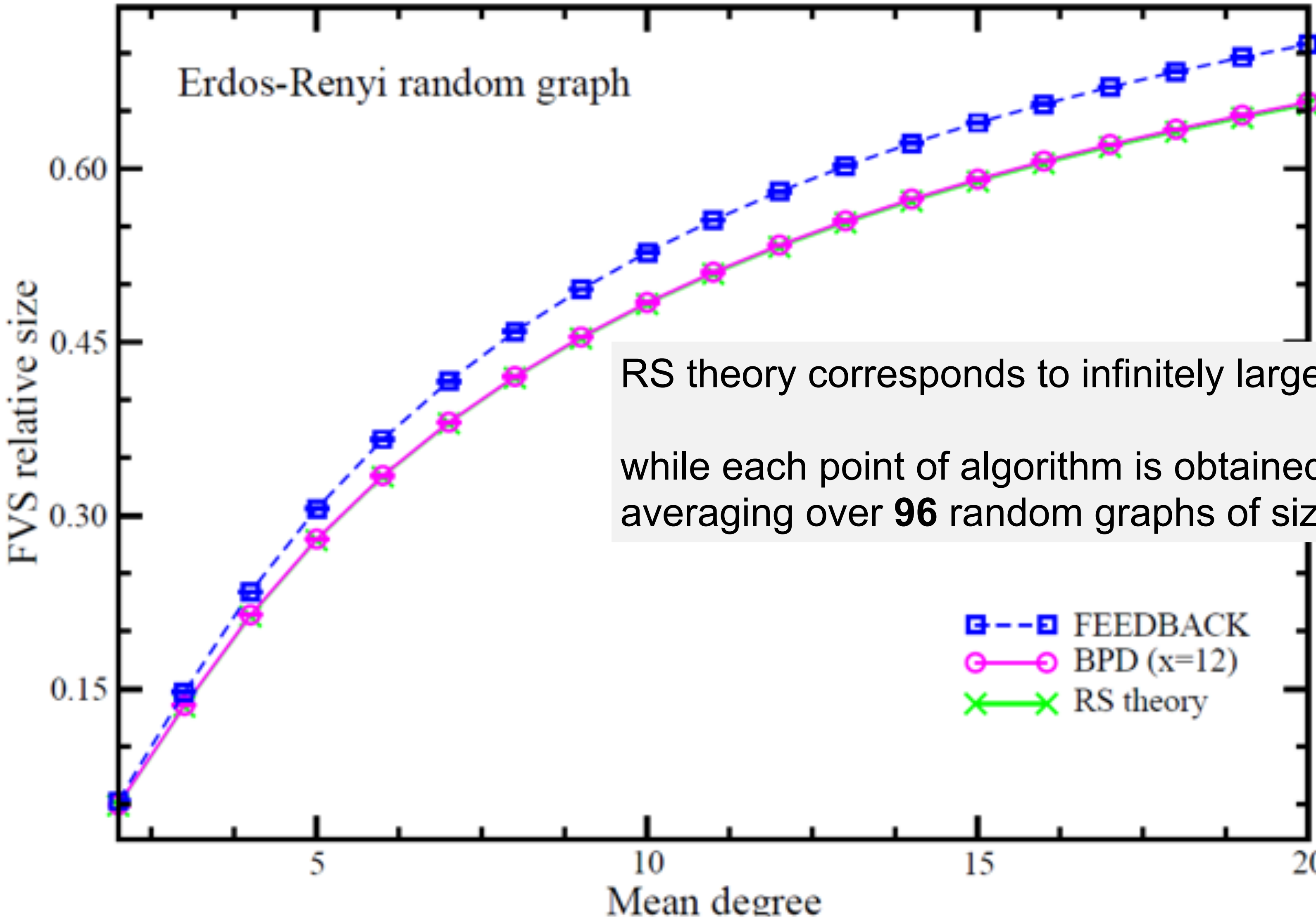
(2). Then we further simplify the graph $G$ by recursively removing vertices of degree 0 or 1 until all the remaining vertices of the graph have two or more attached edges.

(3). If the graph $G$ is non-empty, we return back to step (1).

(4). Output the constructed feedback vertex set $\Gamma$.

# Results: II

## Comparison between theory and algorithms
## Erdos-Renyi Random graphs

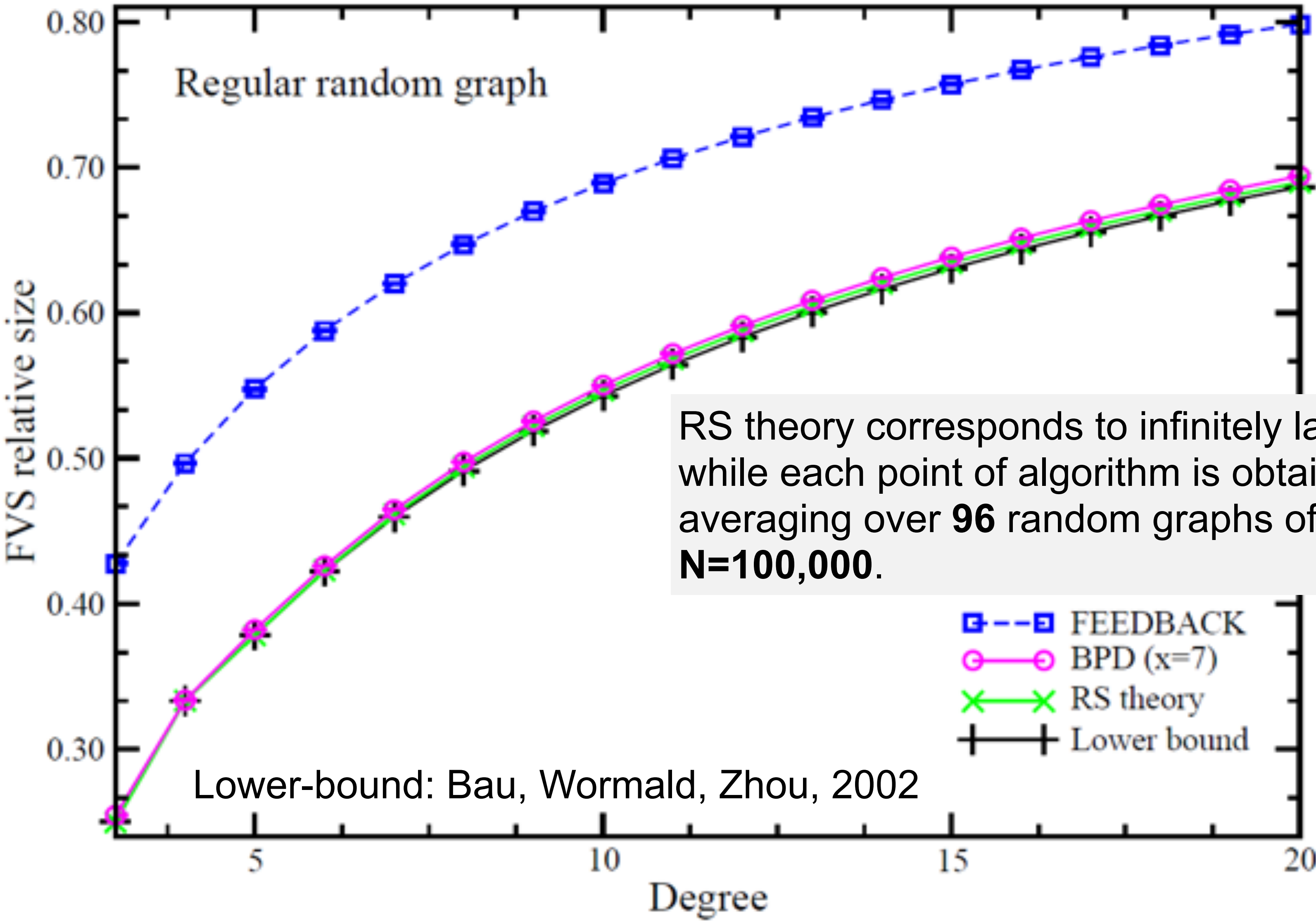



RS theory corresponds to infinitely large system;

while each point of algorithm is obtained by averaging over **96** random graphs of size **N=100,000**.

# Results: III

Comparison between theory and algorithms
Regular Random graphs




RS theory corresponds to infinitely large system; while each point of algorithm is obtained by averaging over **96** random graphs of size **N=100,000**.

Lower-bound: Bau, Wormald, Zhou, 2002

# Results. IV

**Belief-propagation decimation works also good in regular lattices:**

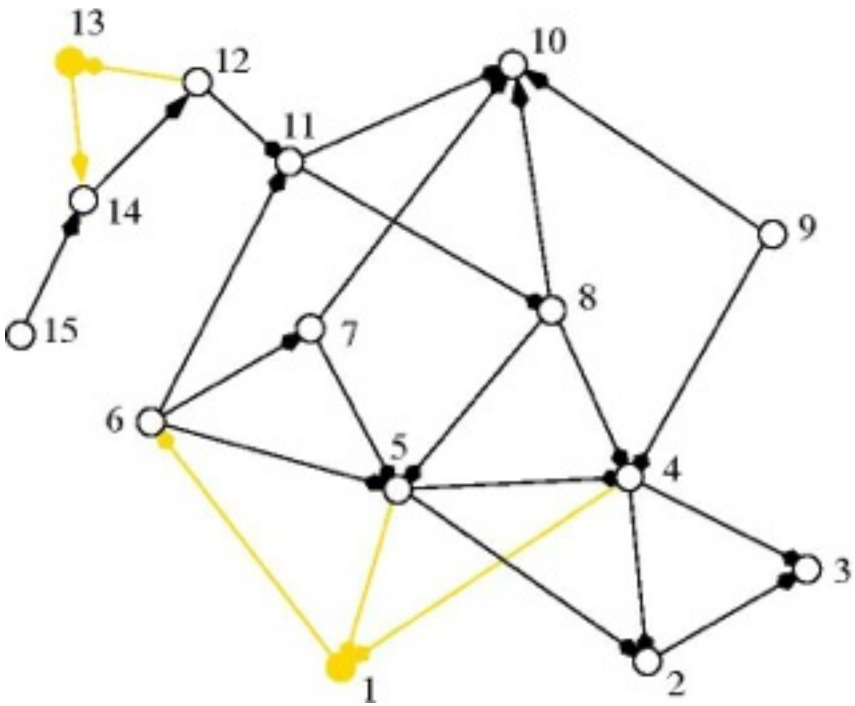
**➔2D square lattice, 35.2% vertices in constructed FVS as compared to 50% by FEEDBACK.**

**This value is close to the rigorous lower-bound 33.3%**

**➔ 3D cubic lattice, 42.2% vertices in constructed FVS as compared to 50% by FEEDBACK.**

**This value is close to the rigorous lower-bound 40.0%**

# Model for FVS of directed graph




- Height $h_i$ of vertex $i$:
  - $h_i = 0$: [$\in$ FVS]
  - $h_i > 0$: [$\notin$ FVS]

Lucas, Arxiv (2013)

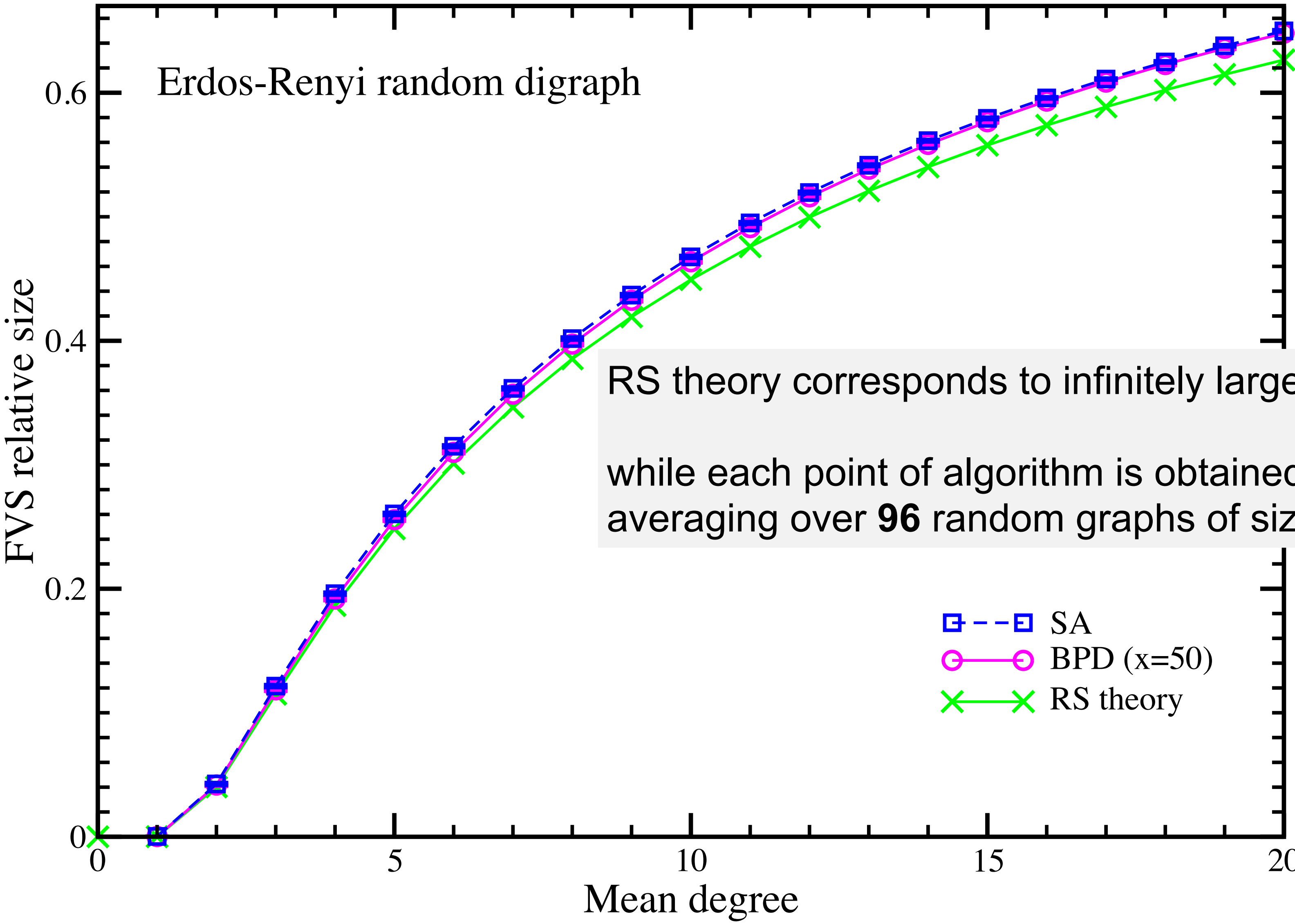
Edge $(i \to j)$ constraint:

if $h_i > 0$ and $h_j > 0$,

then $h_i < h_j$

$$Z(x) = \sum_{\underline{h}} \exp\left[x \sum_{i=1}^{N} (1 - \delta^0_{h_i}) w_i\right] \prod_{(i \to j) \in G} C_{i \to j}(h_i, h_j)$$

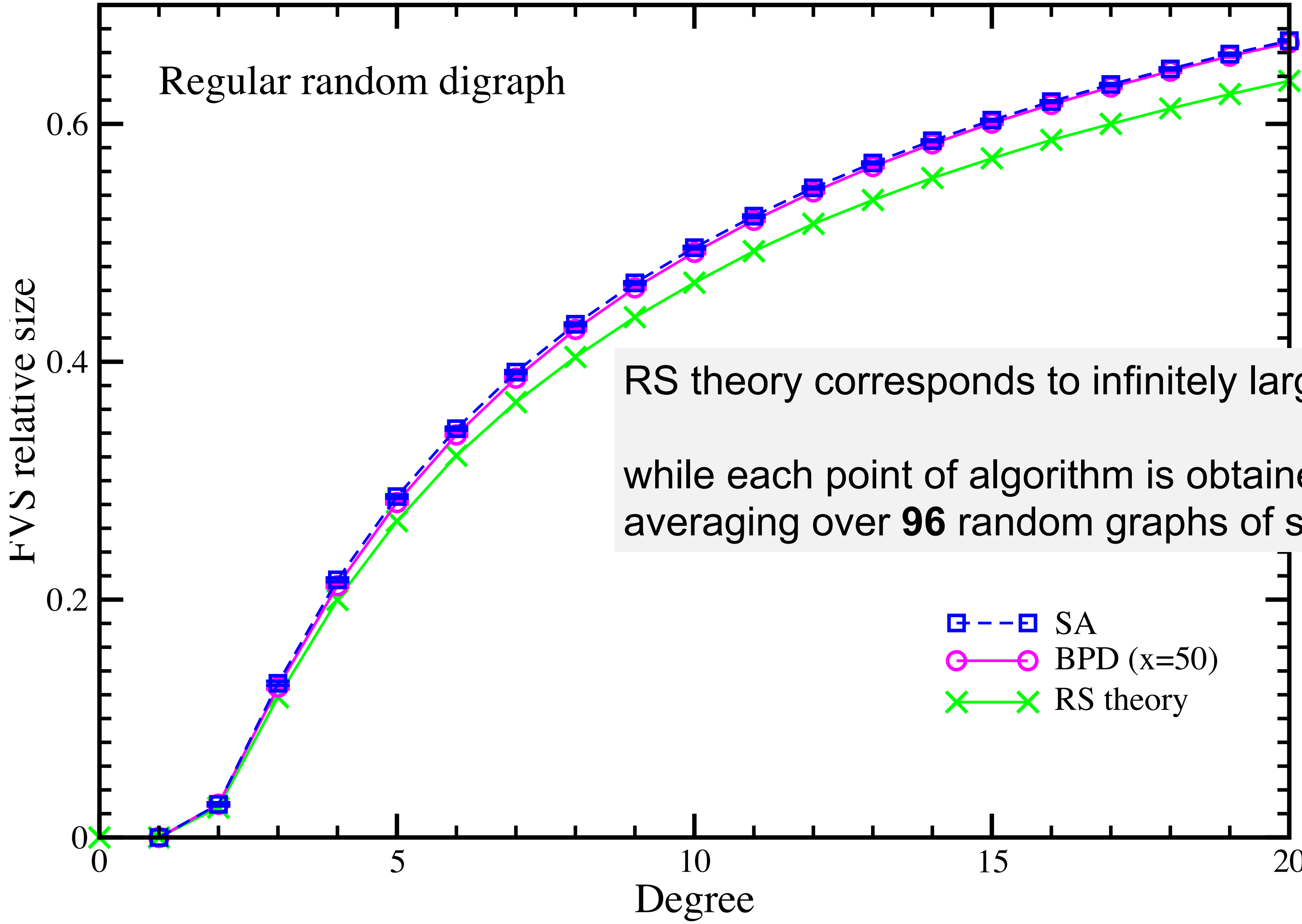
# DFVS: Comparison between theory and algorithms
# Erdos-Renyi Random directed graphs




RS theory corresponds to infinitely large system;

while each point of algorithm is obtained by averaging over **96** random graphs of size **N=100,000**.

# DFVS: Comparison between theory and algorithms Regular Random directed graphs




RS theory corresponds to infinitely large system;

while each point of algorithm is obtained by averaging over **96** random graphs of size **N=100,000**.

# Conclusion

A. Undirected and directed Feedback Vertex Set (FVS) problem solved by replica-symmetric mean-field theory.

B. Belief propagation-guided decimation (BPD) message-passing algorithm achieves good performance on random problem instances.

# On-Going Work:

- Computing the phase diagrams of random graph ensembles by 1RSB mean-field theory.
- Apply simulated annealing and other local optimization algorithms to FVS problem.
- Apply the FVS problem to studying the dynamical property of some processes on complex networks.