KAIST

ICDM 2025
November 12–15, 2025
Washington DC, USA
IEEE International Conference on Data Mining

USC University of Southern California

Edge Probability Graph Models Beyond Edge Independency:

Concepts, Analyses, and Algorithms




Fanchen Bu
(KAIST)

Ruochen Yang
(USC)

Paul Bogdan
(USC)

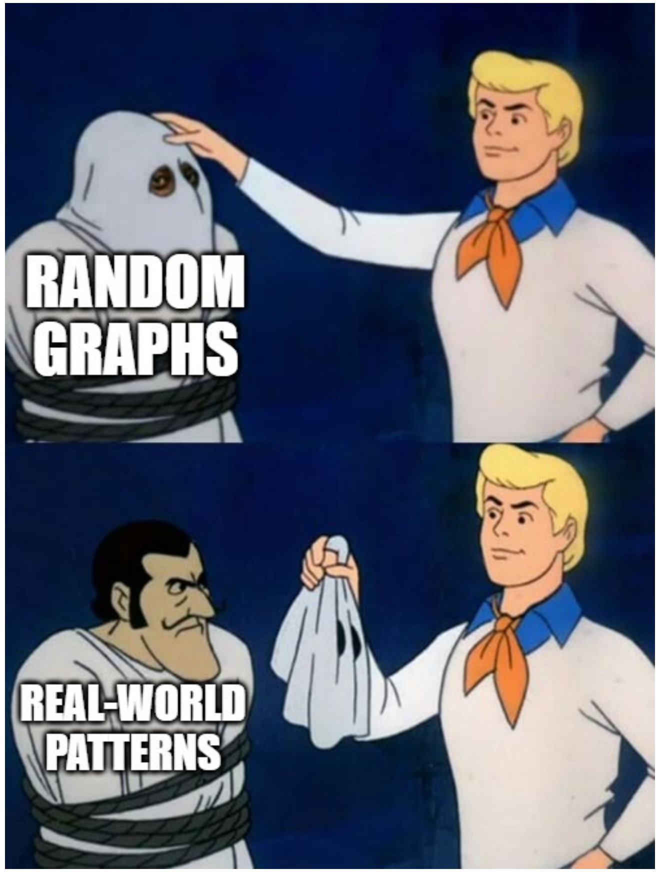
Kijung Shin
(KAIST)

Random Graph Models (RMGs)

WHAT WE CALL RANDOM IS JUST PATTERNS WE CAN'T DECIPHER.

- Chuck Palahniuk (1962 – ; American novelist)

- Random graph models **(RGMs)** are about generating random graphs that reproduce patterns observed in real-world graphs
- **Good RGMs should generate graphs that…**
  (1) reproduce patterns commonly observed in real-world networks,
  (2) stay variable (i.e., not too similar), and
  (3) are feasible for computing and controlling graph statistics



Real-World Application of RGMs

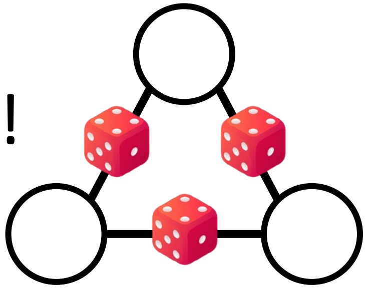
- Generate **synthetic** but **similar-to-real-world** graphs
- Can be used as substitutes, especially when real-world data are **scarce** or **unavailable** due to some practical concerns (e.g., privacy)

Real-World Application of RGMs

- **Graph algorithm testing:** If the algorithm works well on random graphs, we expect it to work well on real-world graphs too
- **Statistical testing:** We examine the statistical significance of some observations by comparing real-world graphs with random ones
- **Graph anonymization:** Release random graphs instead of the original data that might be sensitive, private, etc.

Examples of RGMs

- **Erdős-Rényi model:** Given overall density
  - **Reproducible patterns:** Not really. Mainly used for mathematical purposes
- **Chung-Lu model:** Given degree sequences
  - **Reproducible patterns:** Heavy-tailed degrees, small-world phenomenon…
- **Stochastic block model:** Given node partitions and densities between/within partitions
  - **Reproducible patterns:** Community structure, core-periphery, assortativity…
- **Kronecker model:** Use Kronecker power as edge probabilities
  - **Reproducible patterns:** Fractal structure, heavy-tailed degrees, small-world phenomenon…
- **Common point:** Edge existences are independent to each other!



[Reference] Paul Erdős and Alfréd Rényi. "On Random Graphs I." Publicationes Mathematicae Debrecen (1959).

[Reference] Fan Chung and Linyuan Lu. "Connected Components in Random Graphs with Given Expected Degree Sequences." Annals of Combinatorics (2002).

[Reference] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. "Stochastic Blockmodels: First Steps." Social Networks (1983).

[Reference] Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. "Kronecker Graphs: An Approach to Modeling Networks." Journal of Machine Learning Research (2010).

Edge Independent Graph Models: Merits

- **Edge independent graph models:** All random graph models that assume independent edge existences
- **Simplicity:** Mathematically concise and elegant
- **Tractability:** Easy to analyze and compute graph statistics

Edge Independent Graph Models: Merits

- **Edge independent graph models:** All random graph models that assume independent edge existences
- **Simplicity:** Mathematically concise and elegant
- **Tractability:** Easy to analyze and compute graph statistics

Edge Independent Graph Models: Limitation

- **Overlap:** The similarity between generated random graphs
  - High overlap = Low variability
- **Theorem:** For edge independent graph models, the expected number of triangles is bounded by an increasing function on overlap
  - **Implication:** If you want high triangle-density, you must sacrifice variability
- **High triangle-density** is a common pattern in real-world networks

[Reference] Sudhanshu Chanpuriya, Cameron Musco, Konstantinos Sotiropoulos, and Charalampos Tsourakakis. "On the Power of Edge Independent Graph Models." NeurIPS 2021.

Our Idea: Go Beyond Edge Independency

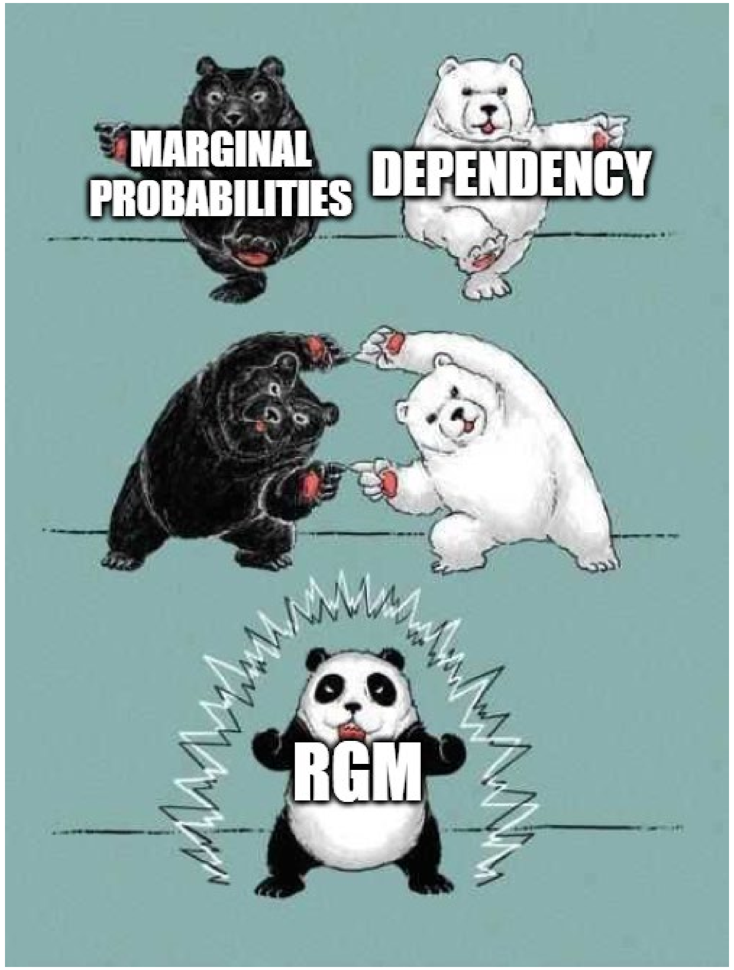
ADOPT WHAT IS USEFUL AND DISCARD WHAT IS USELESS.

- Qichao Liang (1873 – 1929; Chinese journalist)

- **Q:** Can we go beyond edge independency, breaking through the limitations, but still keeping the merits of the edge independent graph models?

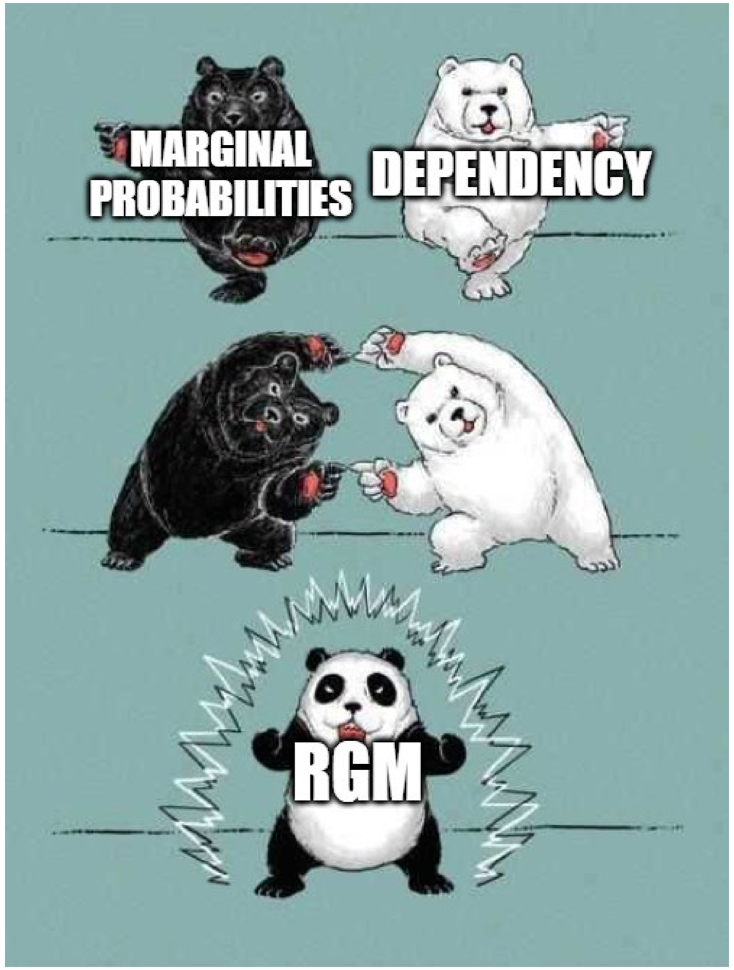
Novel Perspective: RGMs as Decomposition of Distribution

- Random graph model (RGM)
- = distribution of graphs
- = multivariate distribution on edges
- = marginal probability of each edge + (in)dependency between edges



Novel Perspective: RGMs as Decomposition of Distribution

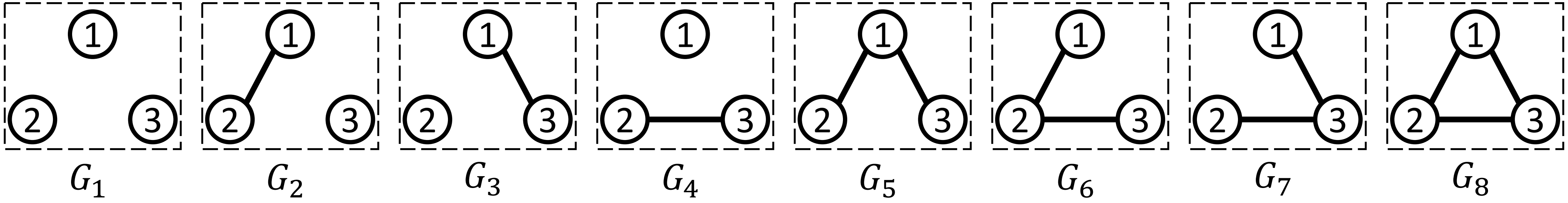
- Random graph model (RGM)
- = distribution of graphs
- = multivariate distribution on edges
- = marginal probability of each edge + (in)dependency between edges
- **Eureka!** We can keep the marginal probabilities but introduce dependency between them!



Edge Probability Graph Models (EPGMs): Definition

- Consider RGMs that generate graphs on n nodes $(v = 1, 2, \dots, n)$
- Given marginal edge probabilities p: $\binom{n}{2} \to [0,1]$, the set of *EPGMs* w.r.t. p consists of all the RGMs that *satisfy the edge probabilities* p
- Such EPGMs share the same marginal edge probabilities p, but vary in how the edge existences depend on each other

Edge Probability Graph Models (EPGMs): Examples

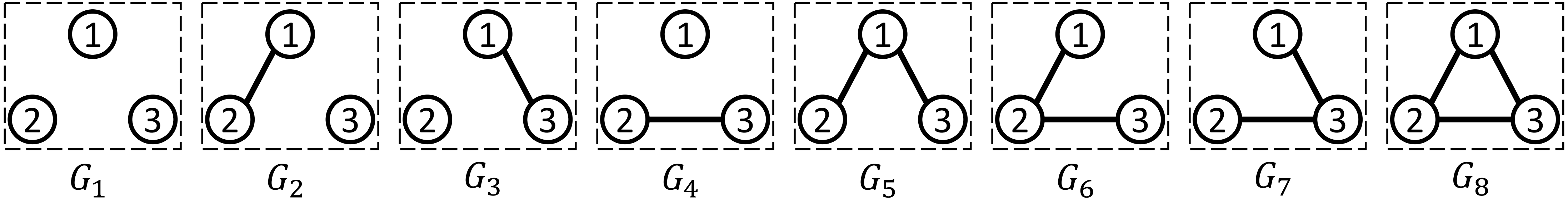


Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_1: Edge independent (*minimally* dependent)
- RGM_2: Between RGM_1 and RGM_3 (*intermediately* dependent)
- RGM_3: *Maximally* dependent

Edge Probability Graph Models (EPGMs): Examples

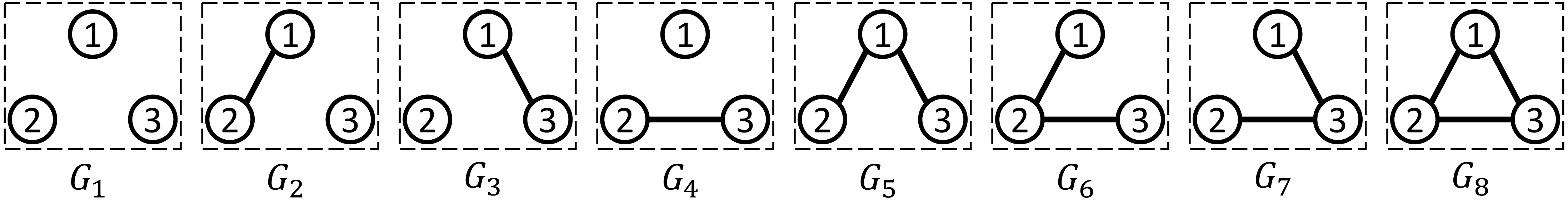


Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_1: Edge independent (*minimally* dependent)
- RGM_2: Between RGM_1 and RGM_3 (*intermediately* dependent)
- RGM_3: *Maximally* dependent

Edge Probability Graph Models (EPGMs): Example

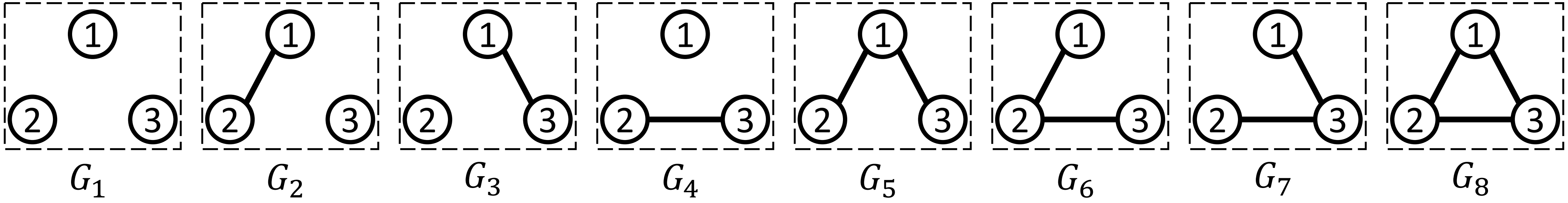


Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_1: Edge independent (*minimally* dependent)
- RGM_2: Between RGM_1 and RGM_3 (*intermediately* dependent)
- RGM_3: *Maximally* dependent

Edge Probability Graph Models (EPGMs): Example



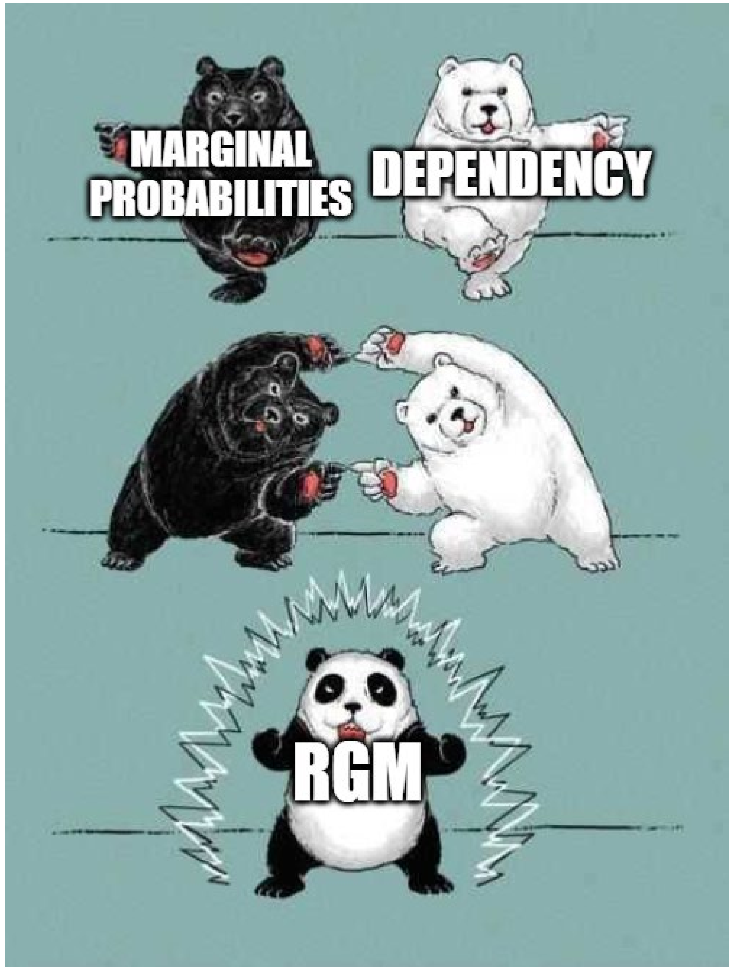
Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_1: Edge independent (*minimally* dependent)
- RGM_2: Between RGM_1 and RGM_3 (*intermediately* dependent)
- RGM_3: *Maximally* dependent

Edge Probability Graph Models (EPGMs): Basic Properties

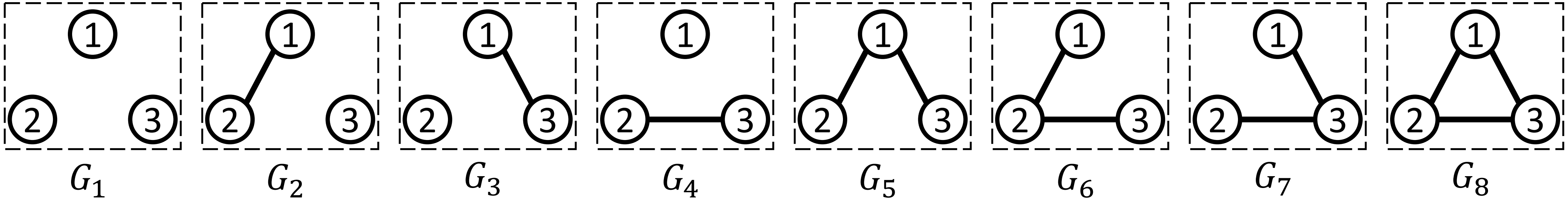
- **EPGMs are general:** *Any* RGM can be decomposed into its marginal edge probabilities p and edge dependency → can be represented as an EGPM w.r.t. p



Edge Probability Graph Models (EPGMs): Basic Properties

- **Recall:** For edge independent graph models, If you want high triangle-density, you must sacrifice variability
- **EPGMs have constant overlap for given $\boldsymbol{p}$:** Given any edge probabilities p, all EPGMs w.r.t. p have the same overlap
  - Specifically, the *same overlap as the edge independent graph model* with marginal edge probabilities p
- While keeping the same overlap (i.e., same variability), we can have higher expected number of triangles!

Edge Probability Graph Models (EPGMs): Basic Properties



Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- Expected number of triangles = $\Pr[G_8]$
- All three RGMs share the same marginal edge probabilities p → they share the same overlap (variability) → but they have different $\Pr[G_8]$ (number of triangles)

Edge Probability Graph Models (EPGMs): Basic Properties

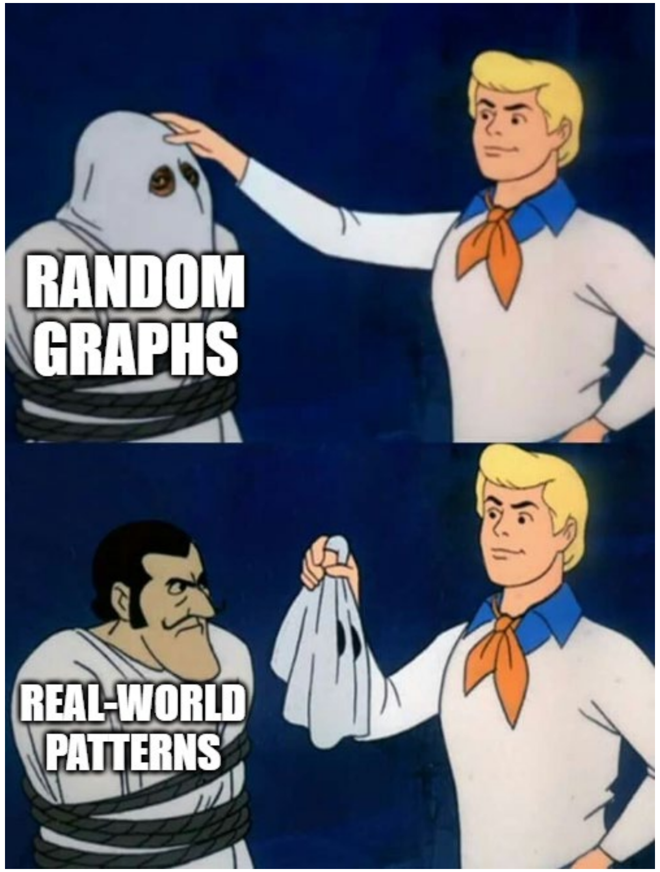
To summarize, EPGMs are…

- **General:** Any RGM can be represented as an EPGM
- **Variable:** The overlap is maintained as the same as the corresponding edge independent graph model
- **Potential to have high triangle-density:** Even with the same marginal edge probabilities, we are able to have higher triangle-density compared to edge-independent models
    - **High triangle-density** is a common pattern in real-world networks

Recall: Random Graph Models (RMGs)

What we call random is just patterns we can't decipher.
- Chuck Palahniuk (1962 – ; American novelist)

- Random graph models **(RGMs)** are about generating random graphs that reproduce patterns observed in real-world graphs
- **Good RGMs should generate graphs that…**
  (1) reproduce patterns commonly observed in real-world networks,
  (2) stay variable (i.e., not too similar), and
  (3) are feasible for computing and controlling graph statistics

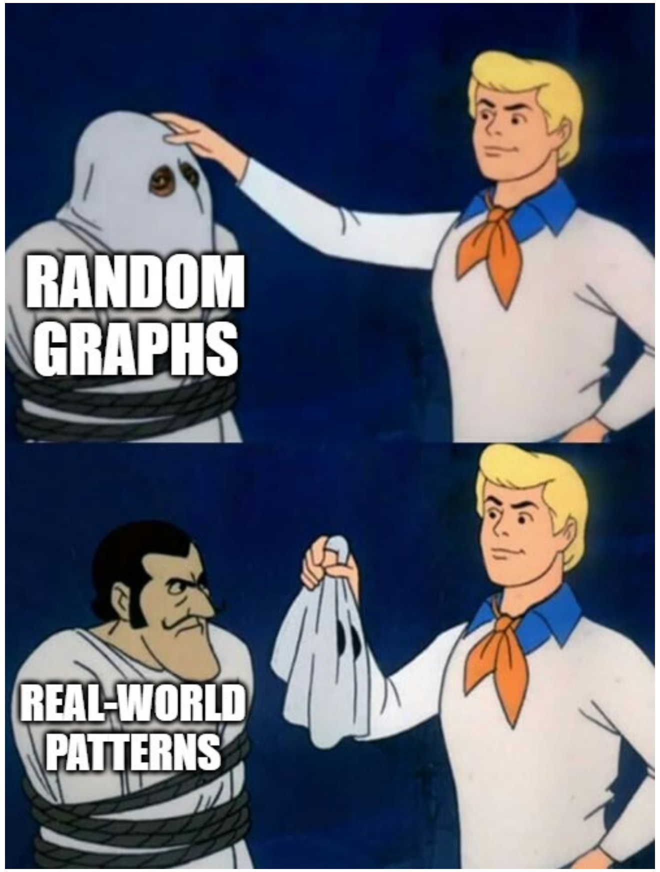


Recall: Random Graph Models (RMGs)

WHAT WE CALL RANDOM IS JUST PATTERNS WE CAN'T DECIPHER.

- Chuck Palahniuk (1962 – ; American novelist)

- Random graph models **(RGMs)** are about generating random graphs that reproduce patterns observed in real-world graphs
- **Good RGMs should generate graphs that…**
  (1) reproduce patterns commonly observed in real-world networks,
  (2) stay variable (i.e., not too similar), and
  (3) are feasible for computing and controlling graph statistics

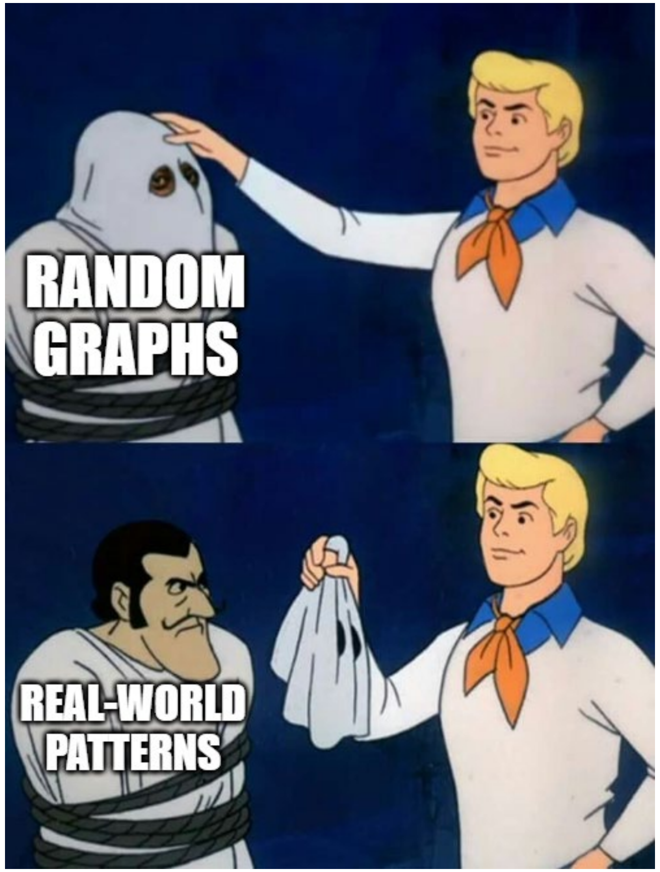


Recall: Random Graph Models (RMGs)

WHAT WE CALL RANDOM IS JUST PATTERNS WE CAN'T DECIPHER.

- Chuck Palahniuk (1962 – ; American novelist)

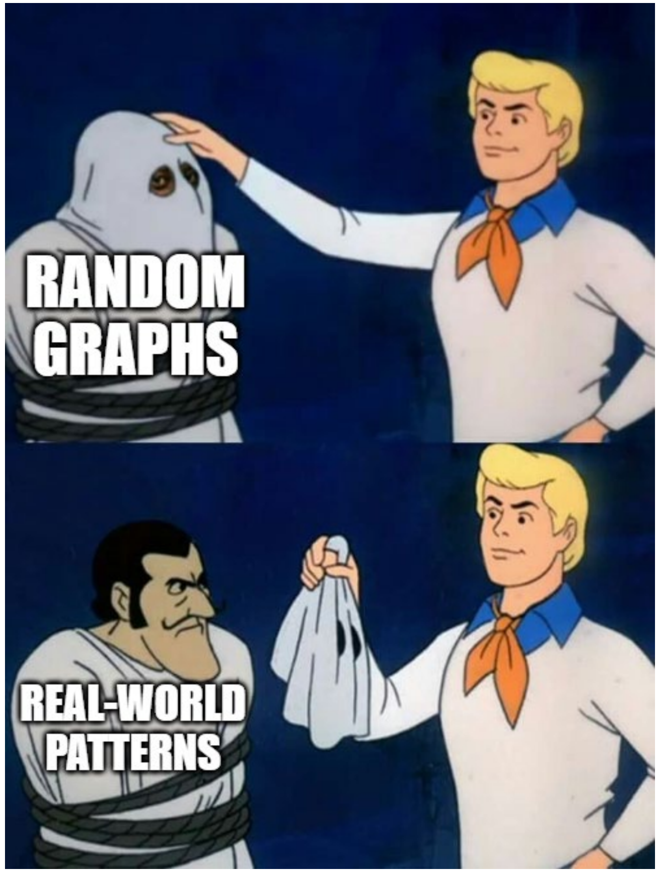
- Random graph models **(RGMs)** are about generating random graphs that reproduce patterns observed in real-world graphs
- **Good RGMs should generate graphs that…**
  (1) reproduce patterns commonly observed in real-world networks,
  (2) stay variable (i.e., not too similar), and
  (3) are feasible for computing and controlling graph statistics



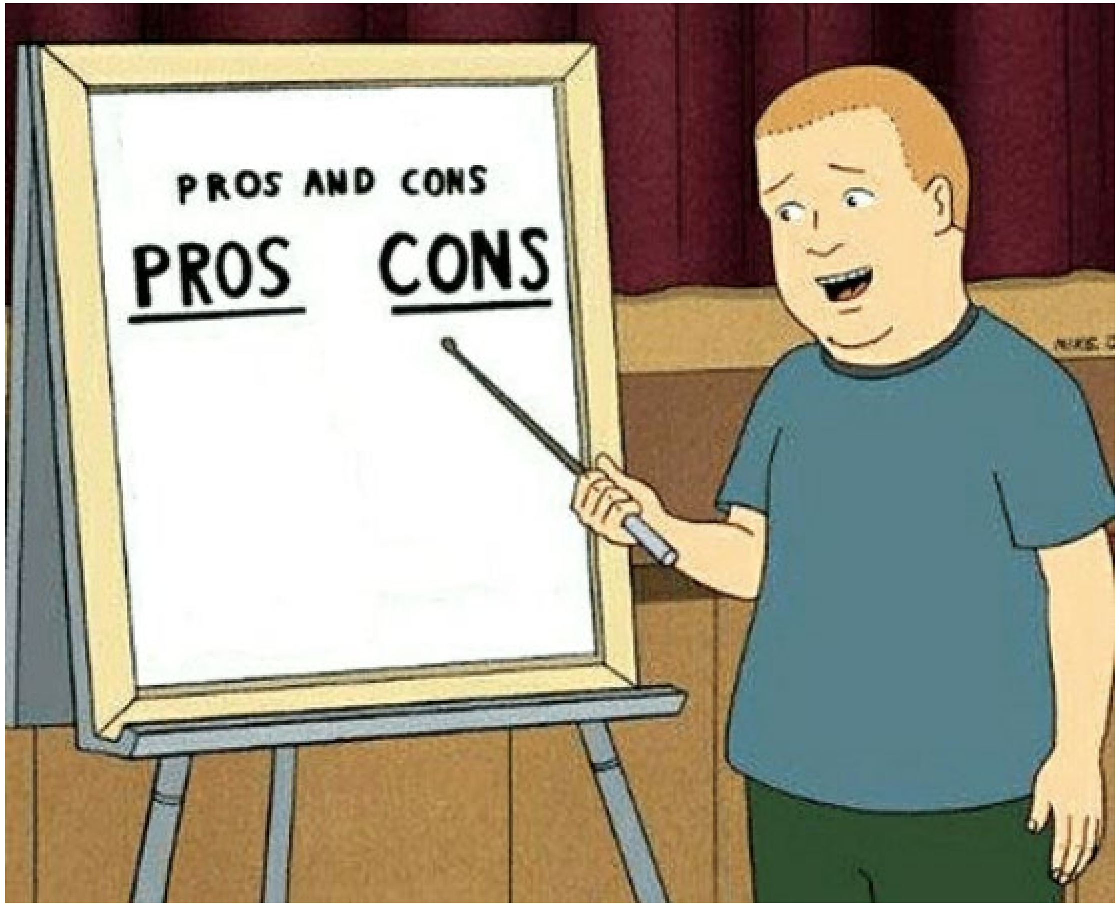
Recall: Random Graph Models (RMGs)

What we call random is just patterns we can't decipher.
- Chuck Palahniuk (1962 – ; American novelist)

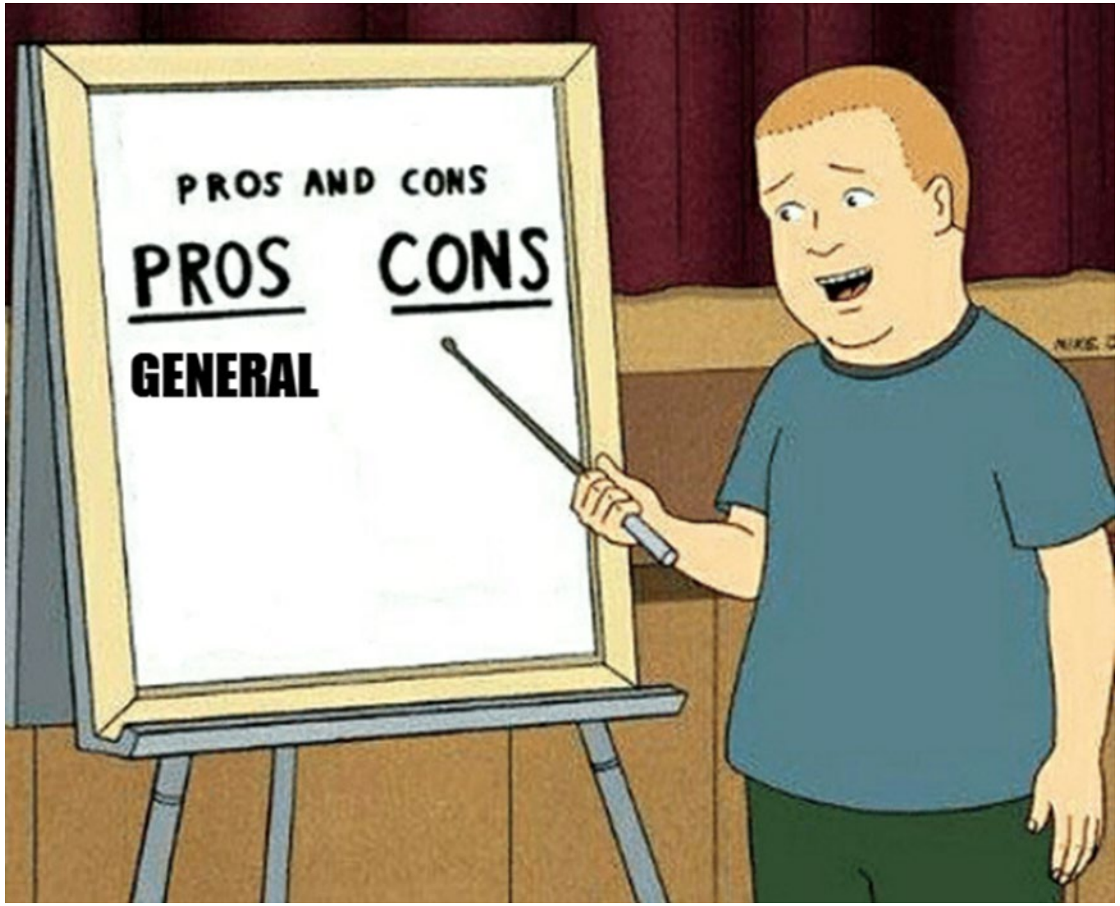
- Random graph models **(RGMs)** are about generating random graphs that reproduce patterns observed in real-world graphs
- **Good RGMs should generate graphs that…**
  (1) reproduce patterns commonly observed in real-world networks,
  (2) stay variable (i.e., not too similar), and
  (3) are feasible for computing and controlling graph statistics



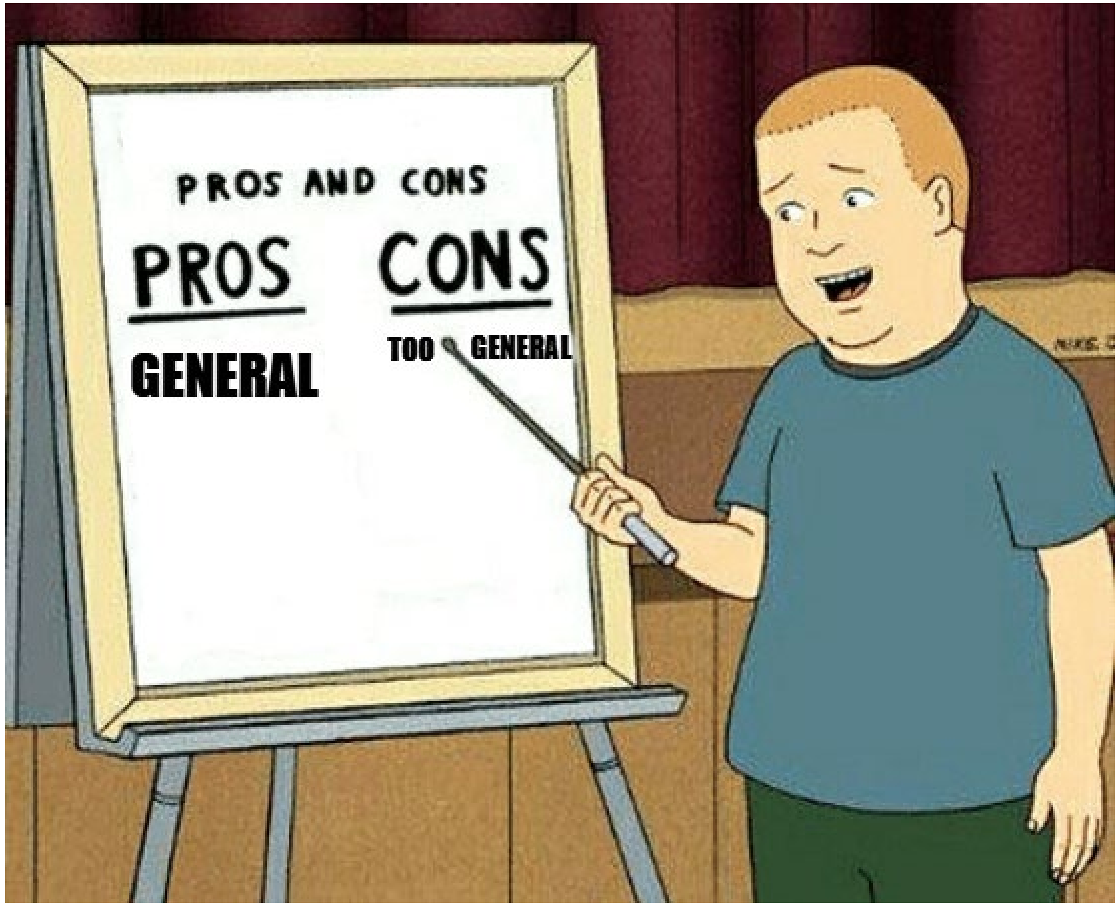
EPGMs: Pros and Cons



EPGMs: Pros and Cons



EPGMs: Pros and Cons

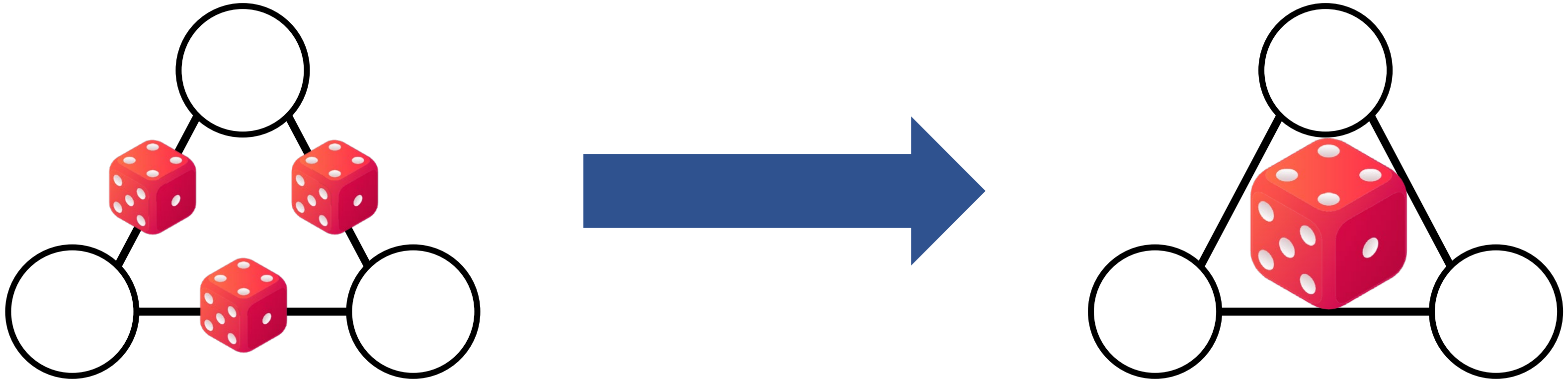


Research Questions

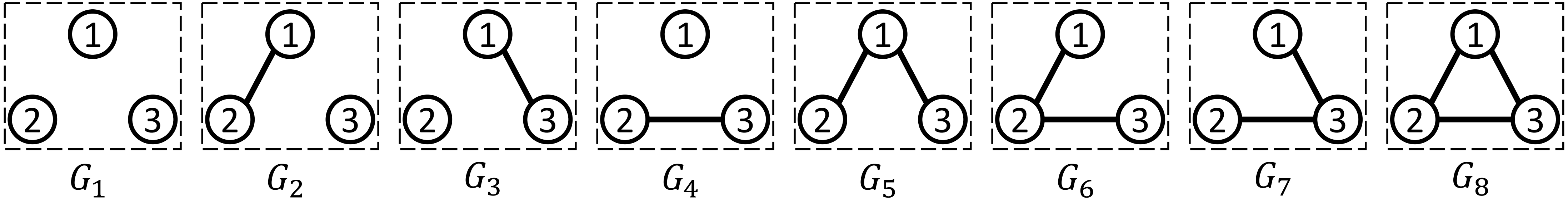
- **Theory:** How to find good subsets of EPGMs that are…
  - **Realistic:** Reproduce common patterns in real-world graphs
  - **Flexible:** Allow us to control the level of edge dependency
  - **Tractable:** Allow us to compute graph statistics
- **Practice:** How to design efficient algorithms…
  - For parameter fitting and graph generation

Binding: Systematic Edge Dependency Imposition

- Group node pairs and decide them together
- **Realistic:** Reproduce common patterns in real-world graphs
  - Specifically, higher clustering (e.g., triangle-density)
- **Flexible:** Allow us to control the level of edge dependency
  - Specifically, by adjusting the extensiveness of binding
- **Tractable:** Allow us to compute graph statistics
  - Specifically, we can compute the closed-form number of triangles



Binding: Example Revisited

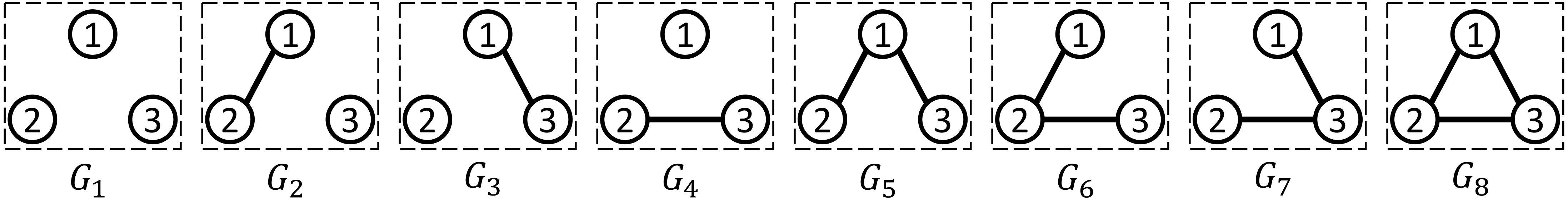


Node pair	Probability
$(1,2)$	$p(1,2) = 1/2$
$(1,3)$	$p(1,3) = 1/4$
$(2,3)$	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_1: Edge independent (*minimally* dependent)
- RGM_2: Between RGM_1 and RGM_3 (*intermediately* dependent)
- RGM_3: *Maximally* dependent

Binding: Example Revisited

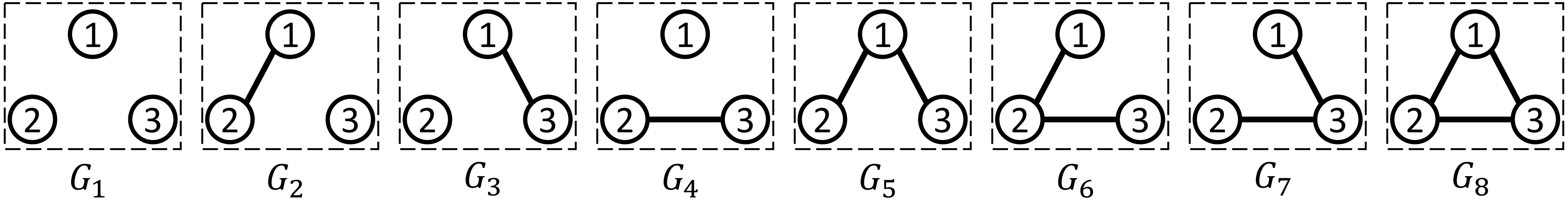


Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_1: Each node pair alone forms a group itself

Binding: Example Revisited

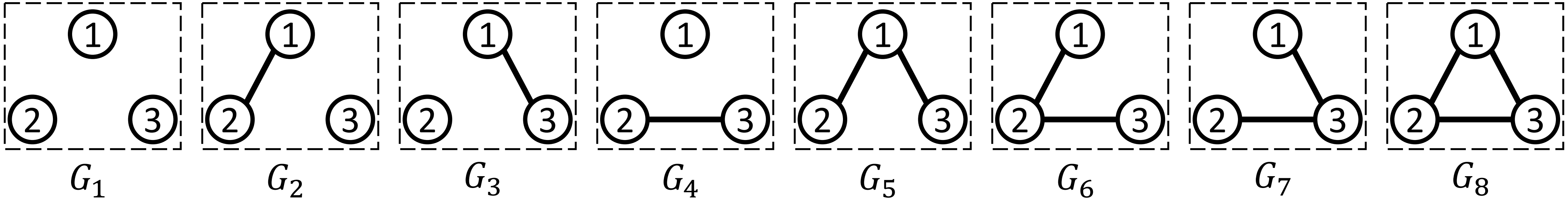


Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_2: $(1,2)$ and $(1,3)$ are grouped and decided together
- Sample a single random number $s \in [0,1]$ for *both pairs*
- Either $(1,2)$ or $(1,3)$ exists if $p(i,j) \geq s$
  - $(2,3)$ is decided independently

Binding: Example Revisited

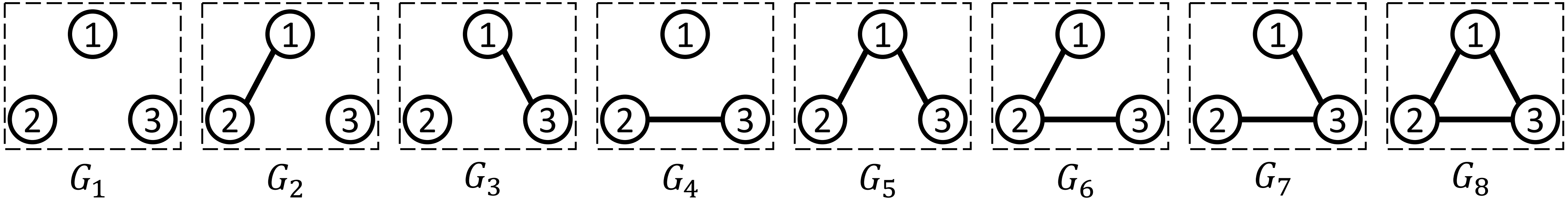


Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_2: Sample a single random number $s \in [0,1]$ for $(1,2)$ and $(1,3)$
- Either $(1,2)$ or $(1,3)$ exists if $p(i,j) \geq s$
  - (1) $0 \leq s \leq 1/4$: Both $(1,2)$ and $(1,3)$ exist (G_5 or G_8)
  - (2) $1/4 < s \leq 1/2$: Only $(1,2)$ exists (G_2 or G_6)
  - (3) $1/2 < s \leq 1$: Neither exists (G_1 or G_4)

Binding: Example Revisited

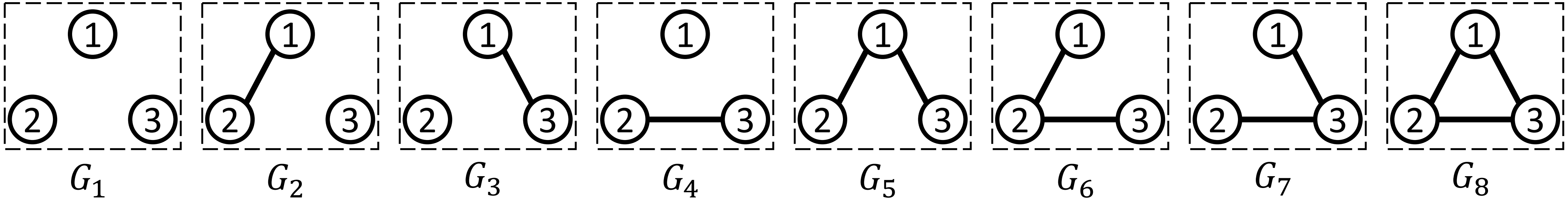


Node pair	Probability
$(1,2)$	$p(1,2) = 1/2$
$(1,3)$	$p(1,3) = 1/4$
$(2,3)$	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_3: All three node pairs are grouped and decided together
- Sample a single random number $s \in [0,1]$ for *the whole group*
- Each edge (i, j) exists if $p(i, j) \geq s$

Binding: Example Revisited

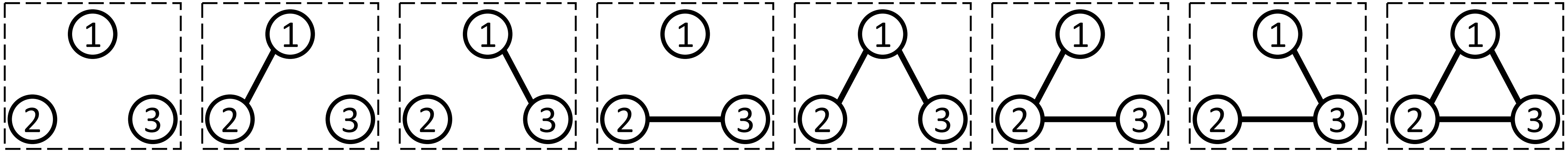


Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 1/4$
(2,3)	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

- RGM_3: Sample a single random number $s \in [0,1]$ for *the whole group*
- Each edge (i,j) exists if $p(i,j) \geq s$:
  - (1) $0 \leq s \leq 1/4$: All three edges exist (G_8)
  - (2) $1/4 < s \leq 1/2$: Only $(1,2)$ and $(2,3)$ exist (G_6)
  - (3) $1/2 < s \leq 1$: None exists (G_1)

Binding: Example Revisited



G_1 G_2 G_3 G_4 G_5 G_6 G_7 G_8

Node pair	Probability
$(1,2)$	$p(1,2) = 1/2$
$(1,3)$	$p(1,3) = 1/4$
$(2,3)$	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

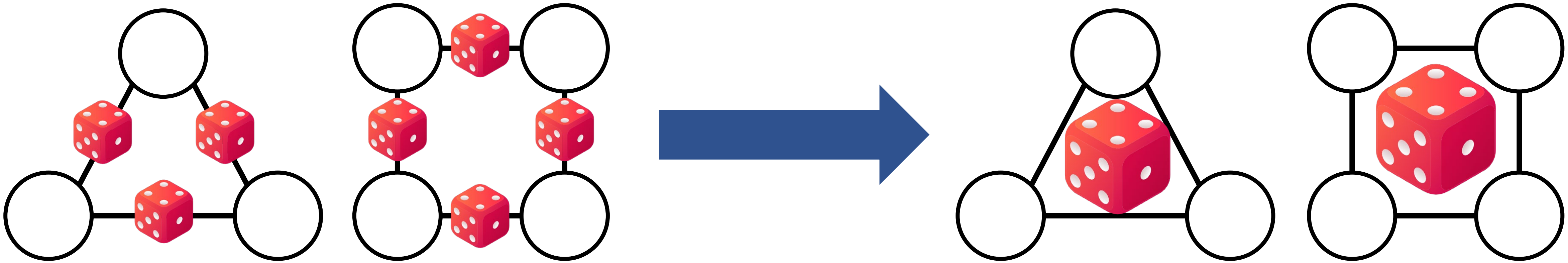
- RGM_3: Sample a single random number $s \in [0,1]$ for *the whole group*
- Each edge (i, j) exists if $p(i, j) \geq s$
  - For each (i, j), its marginal probabilities is maintained
  - While edge dependency is imposed among nodes pairs in the same group

Binding: Intuitions

- More node pairs are grouped together
  → "Stronger" edge dependency
  → Higher triangle-density (and general clustering)
- **Maximal:** All node pairs are grouped together
- **Minimal:** Each node pair alone forms a group (edge independent)
- Between the two extreme cases, we have various ways to group the node pairs and thus impose edge dependency

Local Binding: Node-Oriented Grouping

- **Q:** How can we decide which node pairs to group together?
- **Challenge:** There are *too many possible ways* to group them, and *many of them are not meaningful* (e.g., grouping irrelevant pairs)!
- We propose to use *node-oriented* grouping
- We first group nodes, and then bind the node pairs between them



Local Binding: Node-Oriented Grouping

- **It is realistic:** In real-world social networks, we have *group interactions*, where multiple nodes (people) form a group and the interaction between them depend on each other

Local Binding: Iterative Framework

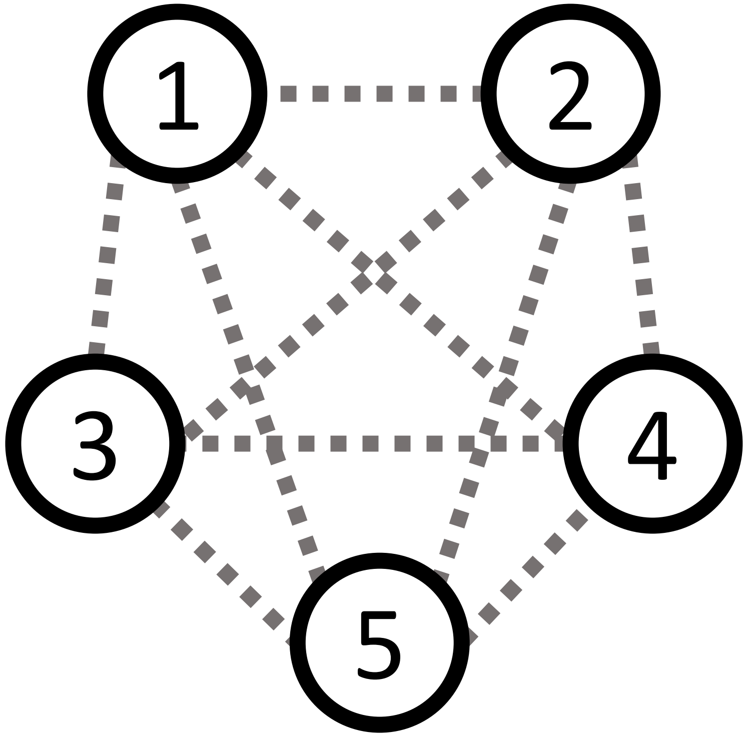
- **Challenge:** But there are still many ways to group nodes
- Consider RGMs that generate graphs on n nodes $(v = 1,2,\ldots,n)$
- **Given:** (1) Edge probabilities $p: \binom{n}{2} \to [0,1]$, (2) node-sampling probabilities $g: [n] \to [0,1]$, (3) maximum number of rounds: R
- Initialize the set of remaining (i.e., not-yet-grouped) node pairs P_{rem}
- Repeat for each round $i = 1,2,\ldots,R$:
  - Sample each node $v \in [n]$ with probability $g(v)$ → Grouped nodes V_i
  - Get the not-yet-grouped pairs among V_i → Node pairs $P_i = \binom{V_i}{2} \cap P_{\mathrm{rem}}$
    - Exclude those grouped pairs: $P_{\mathrm{rem}} \leftarrow P_{\mathrm{rem}} \setminus P_i$
  - Bind those pairs together and generate edges → Generated edges E_i
- Independent sampling for the remaining pairs after R rounds (if any)

Local Binding: Example

- The edge probabilities p and the node-sampling probabilities g are in the tables. We sample for $R = 2$ rounds for $n = 5$ nodes

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

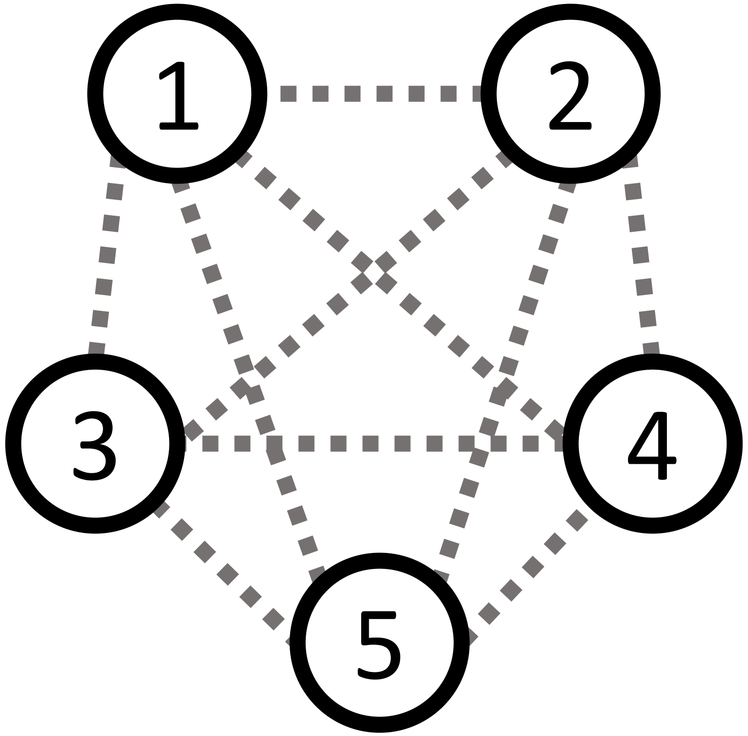


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = ?$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

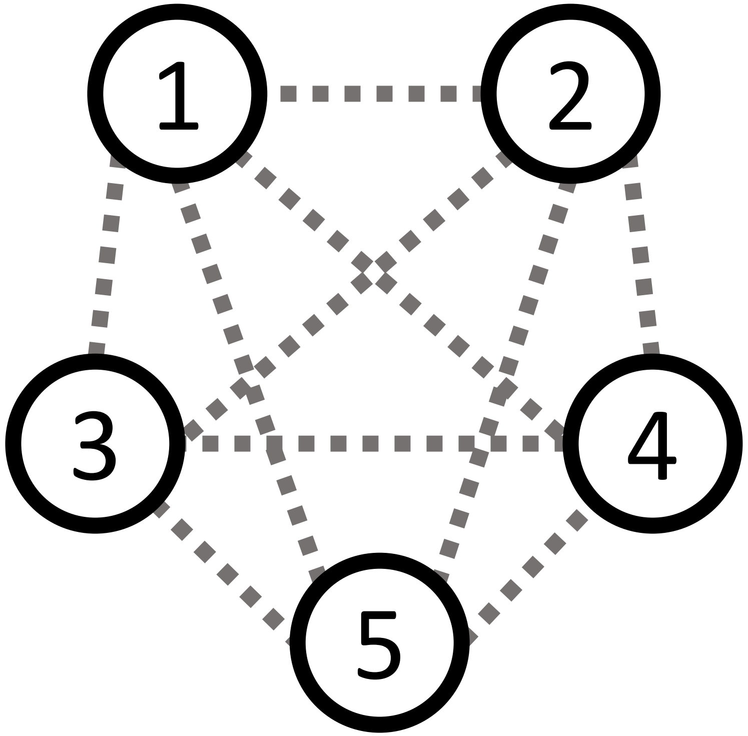


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = ?$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

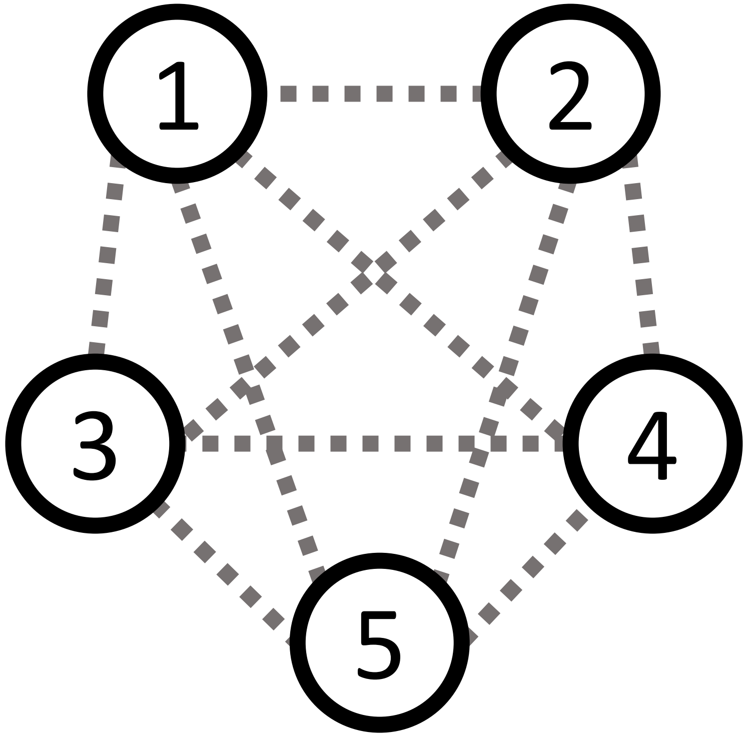


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2, \ldots ?\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

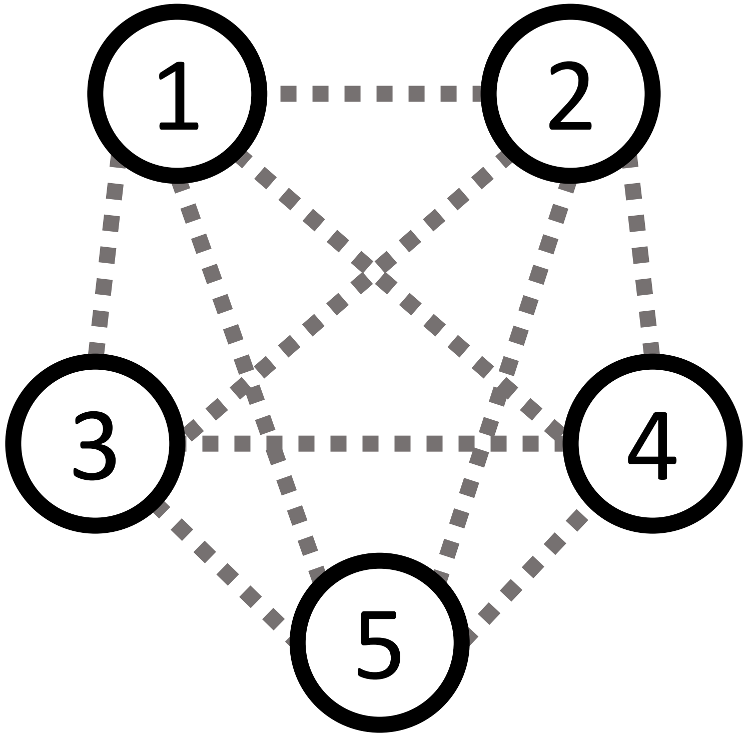


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3, \dots ?\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

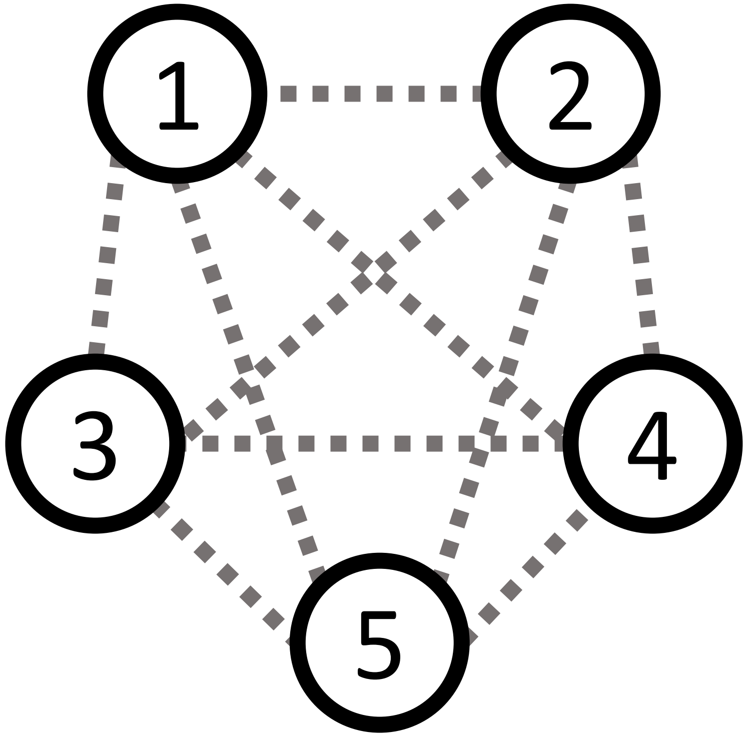


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3,4, \ldots ?\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

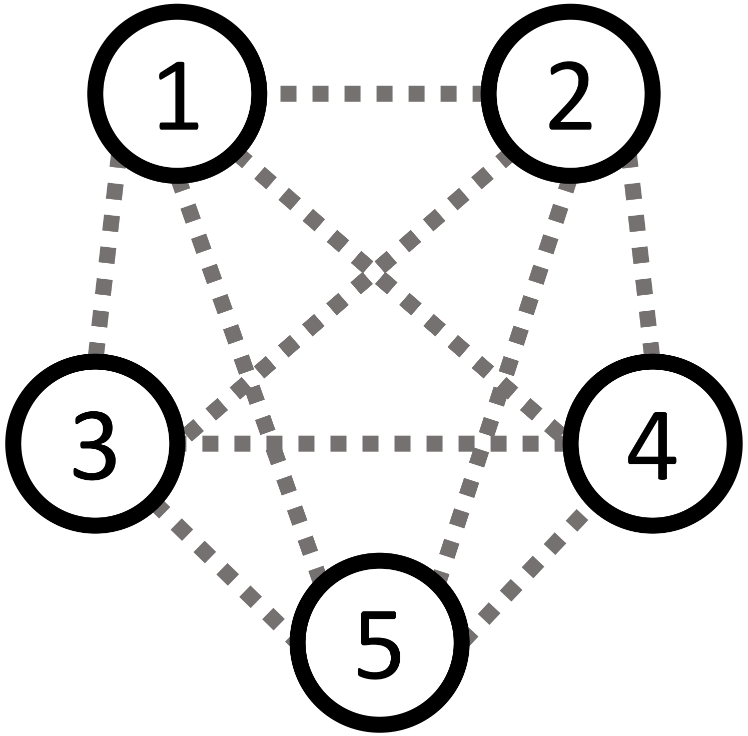


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3,4,5\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

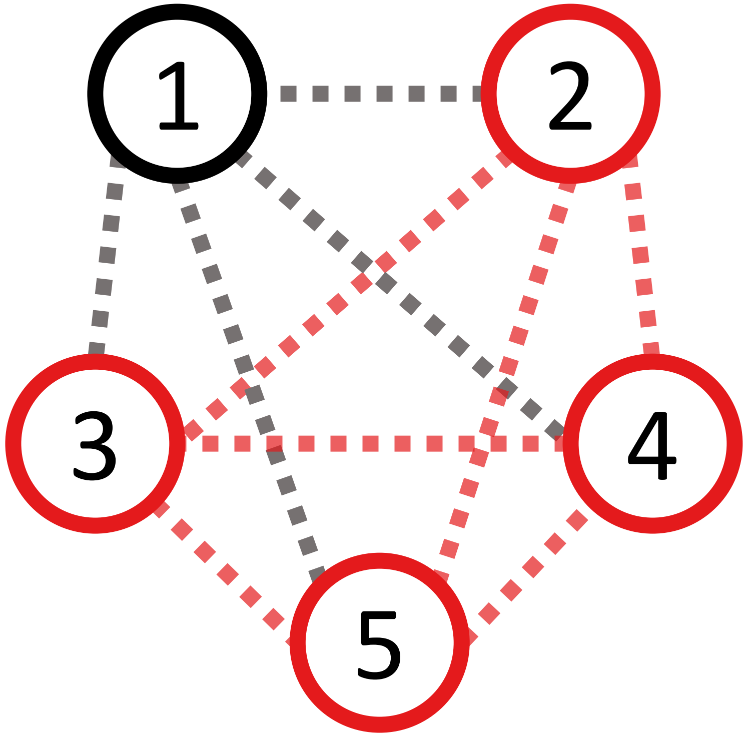


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3,4,5\}$ → Grouped pairs $P_i = \{\{2,3\},\{2,4\},\{2,5\},\{3,4\},\{3,5\},\{4,5\}\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

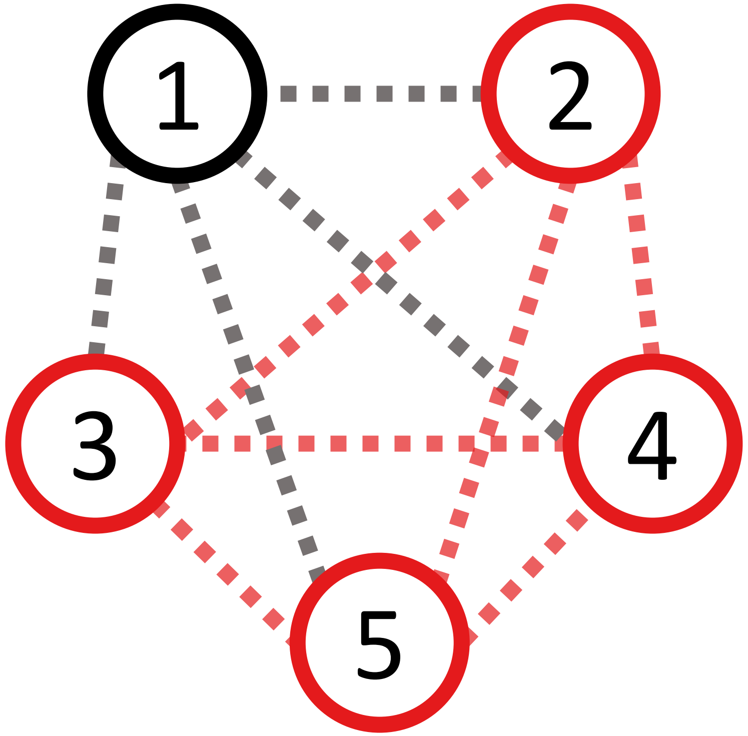


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3,4,5\}$ → Grouped pairs $P_i = \{\{2,3\},\{2,4\},\{2,5\},\{3,4\},\{3,5\},\{4,5\}\}$ → Sampled $s = 0.47$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

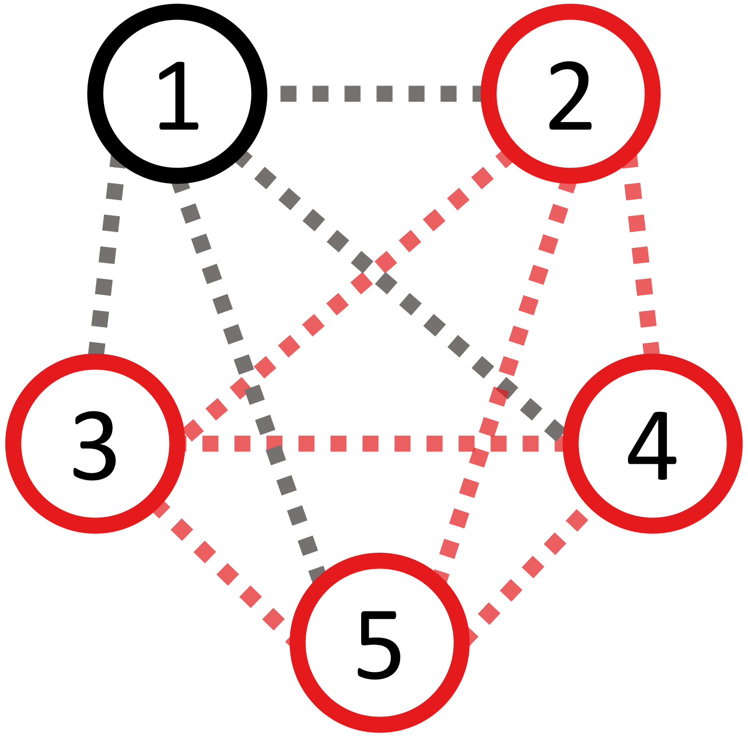


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3,4,5\}$ → Grouped pairs $P_i = \{\{2,3\},\{2,4\},\{2,5\},\{3,4\},\{3,5\},\{4,5\}\}$ → Sampled $s = 0.47$ → Generated edges $E_i = ?$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

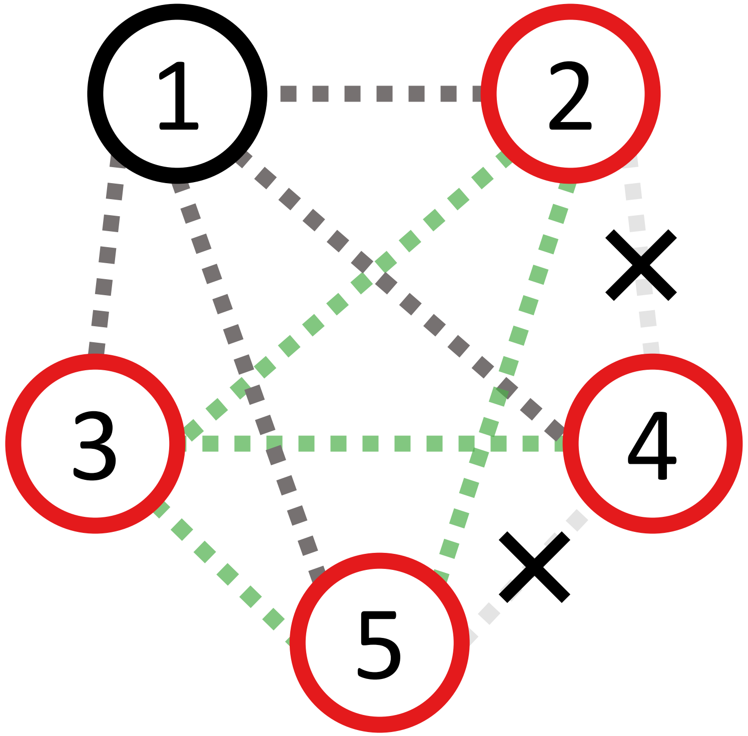


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3,4,5\}$ → Grouped pairs $P_i = \{\{2,3\},\{2,4\},\{2,5\},\{3,4\},\{3,5\},\{4,5\}\}$ → Sampled $s = 0.47$ → Generated edges $E_i = \{\{2,3\},\{2,5\},\{3,4\},\{3,5\}\}$

Node pair	Probability	
(1,2)	$p(1,2) = 1/2$	
(1,3)	$p(1,3) = 2/5$	
(1,4)	$p(1,4) = 1/3$	
(1,5)	$p(1,5) = 1/4$	
(2,3)	$p(2,3) = 3/4$	✓
(2,4)	$p(2,4) = 1/4$	✗
(2,5)	$p(2,5) = 2/3$	✓
(3,4)	$p(3,4) = 3/5$	✓
(3,5)	$p(3,5) = 1/2$	✓
(4,5)	$p(4,5) = 1/5$	✗

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

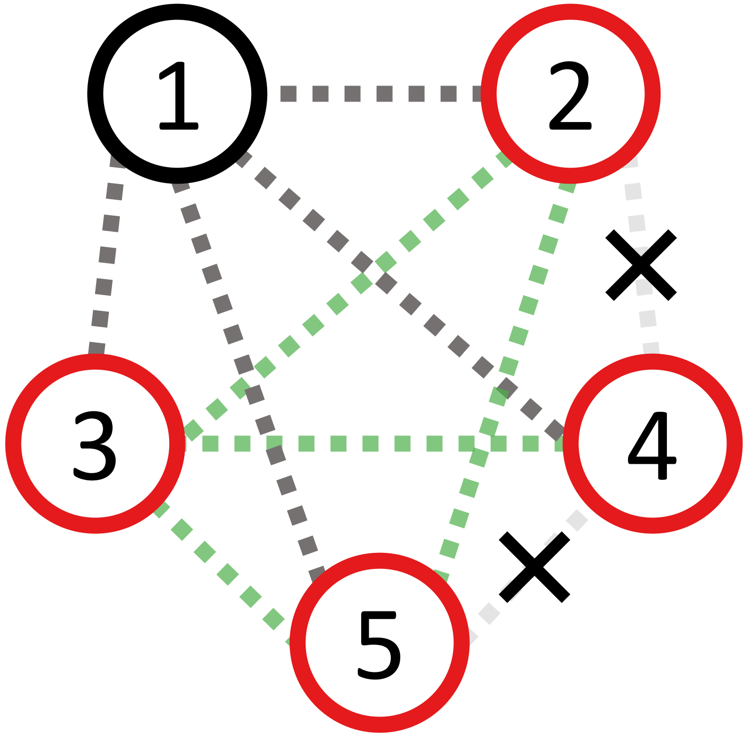


Local Binding: Example (Round $i = 1$)

- Sampled nodes $V_i = \{2,3,4,5\}$ → Grouped pairs $P_i = \{\{2,3\},\{2,4\},\{2,5\},\{3,4\},\{3,5\},\{4,5\}\}$ → Sampled $s = 0.47$ → Generated edges $E_i = \{\{2,3\},\{2,5\},\{3,4\},\{3,5\}\}$ → Round 1 over!

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

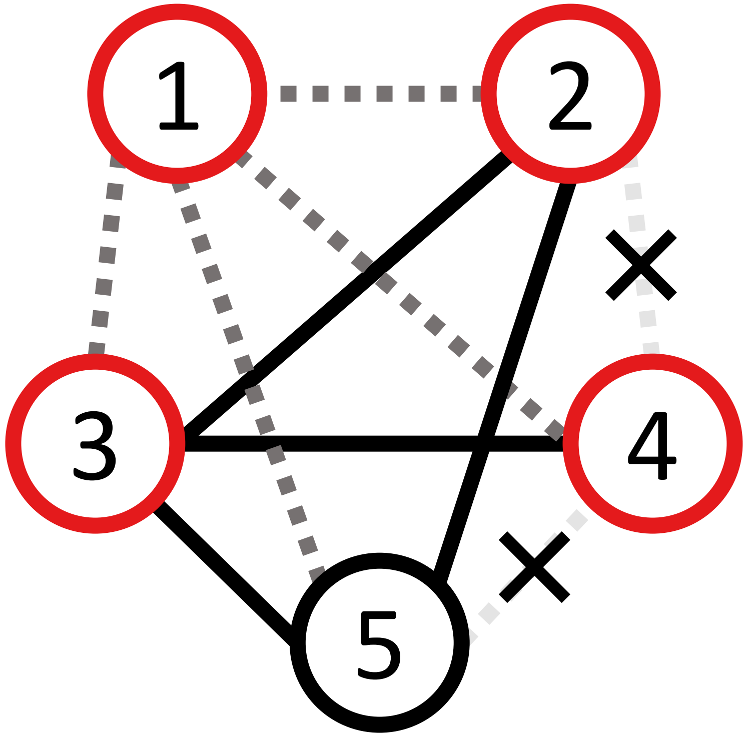


Local Binding: Example (Round $i = 2$)

- Sampled nodes $V_i = \{1,2,3,4\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

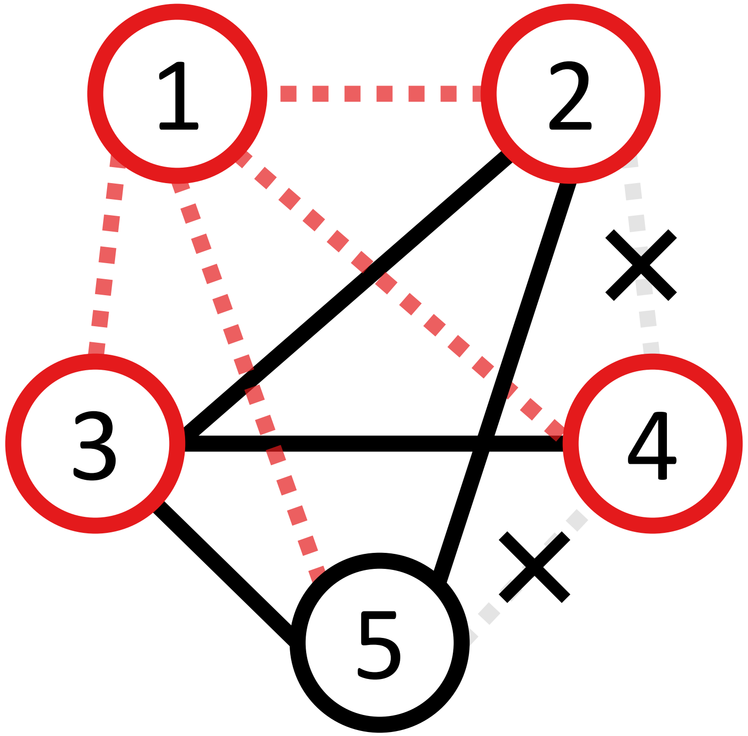


Local Binding: Example (Round $i = 2$)

- Sampled nodes $V_i = \{1,2,3,4\}$ → Grouped pairs $P_i = \{\{1,2\},\{1,3\},\{1,4\},\{1,5\}\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

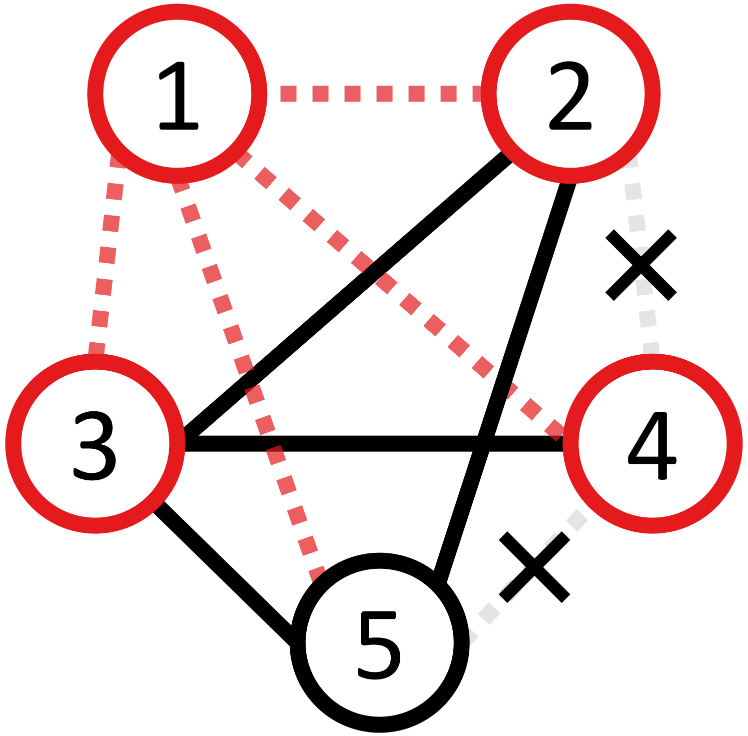


Local Binding: Example (Round $i = 2$)

- Sampled nodes $V_i = \{1,2,3,4\}$ → Grouped pairs $P_i = \{\{1,2\},\{1,3\},\{1,4\},\{1,5\}\}$ → Sampled $s = 0.39$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

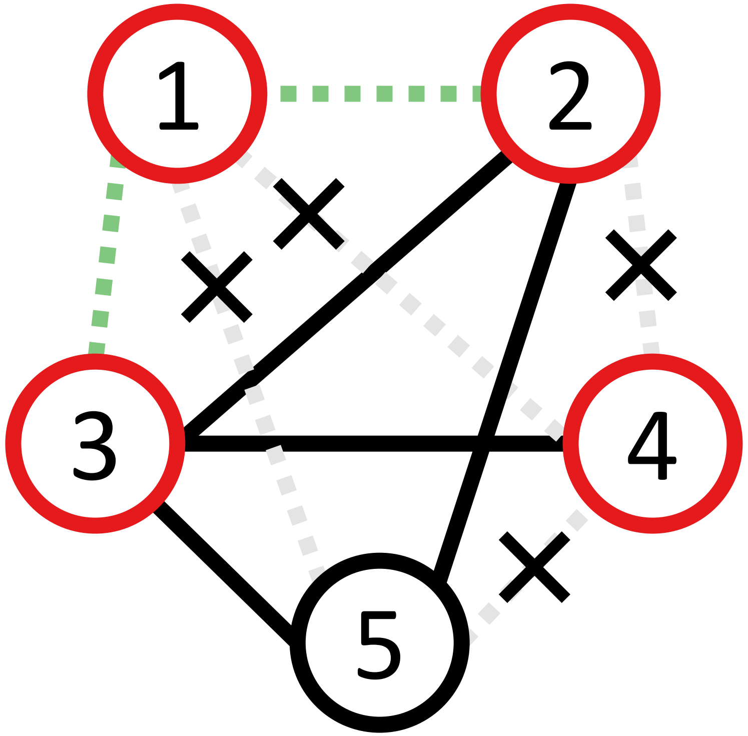


Local Binding: Example (Round $i = 2$)

- Sampled nodes $V_i = \{1,2,3,4\}$ → Grouped pairs $P_i = \{\{1,2\},\{1,3\},\{1,4\},\{1,5\}\}$ → Sampled $s = 0.39$ → Generated edges $E_i = \{\{1,2\},\{1,3\}\}$

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

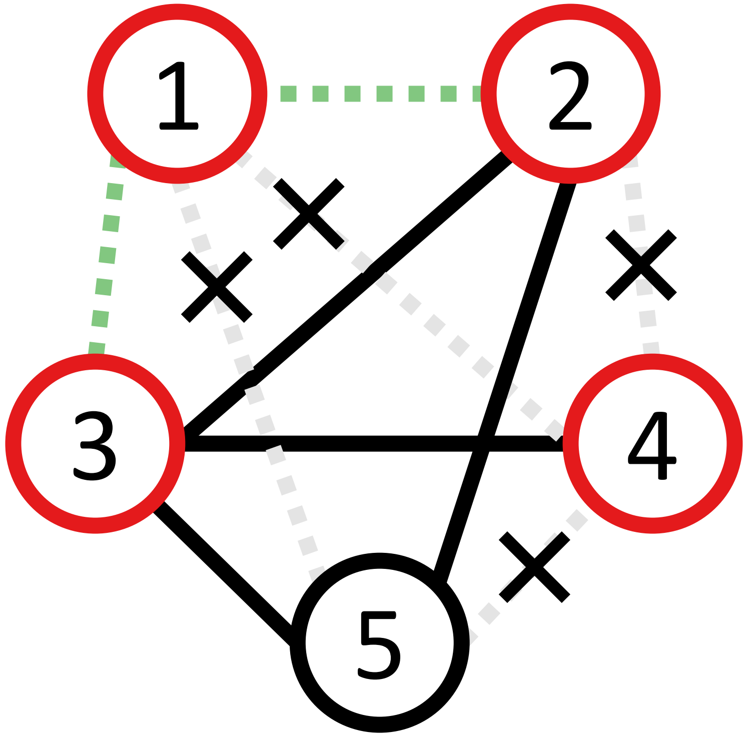


Local Binding: Example (Round $i = 2$)

- Sampled nodes $V_i = \{1,2,3,4\}$ → Grouped pairs $P_i = \{\{1,2\}, \{1,3\}, \{1,4\}, \{1,5\}\}$ → Sampled $s = 0.39$ → Generated edges $E_i = \{\{1,2\}, \{1,3\}\}$ → Round 2 over!

Node pair	Probability
(1,2)	$p(1,2) = 1/2$
(1,3)	$p(1,3) = 2/5$
(1,4)	$p(1,4) = 1/3$
(1,5)	$p(1,5) = 1/4$
(2,3)	$p(2,3) = 3/4$
(2,4)	$p(2,4) = 1/4$
(2,5)	$p(2,5) = 2/3$
(3,4)	$p(3,4) = 3/5$
(3,5)	$p(3,5) = 1/2$
(4,5)	$p(4,5) = 1/5$

Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$

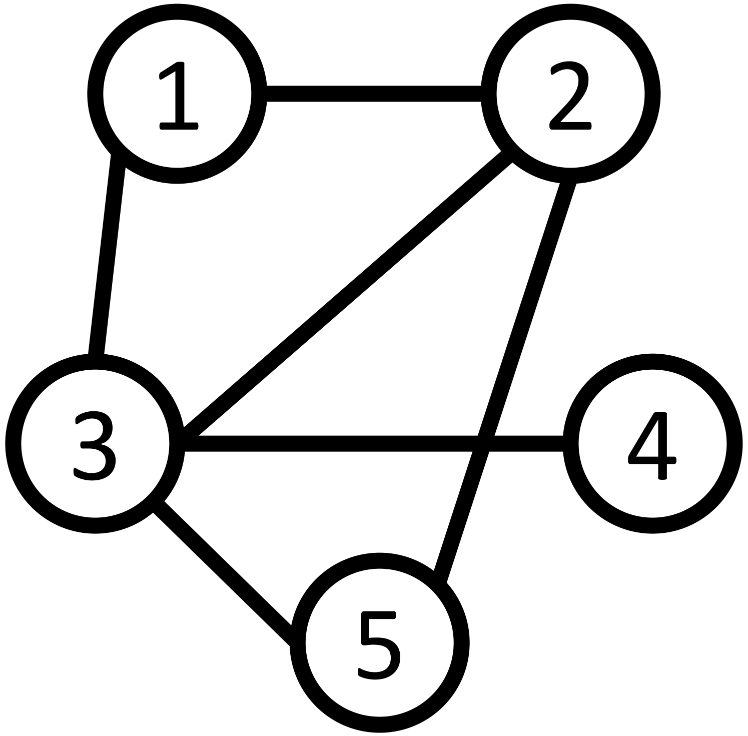


Local Binding: Example (Termination)

- All node pairs have been determined (i.e., remaining pairs $P_{\text{rem}} = \emptyset$) → The whole generation process is terminated → Final edge set $E = E_1 \cup E_2 = \{\{1,2\},\{1,3\},\{2,3\},\{2,5\},\{3,4\},\{3,5\}\}$

Node pair	Probability
$(1,2)$	$p(1,2) = 1/2$
$(1,3)$	$p(1,3) = 2/5$
$(1,4)$	$p(1,4) = 1/3$
$(1,5)$	$p(1,5) = 1/4$
$(2,3)$	$p(2,3) = 3/4$
$(2,4)$	$p(2,4) = 1/4$
$(2,5)$	$p(2,5) = 2/3$
$(3,4)$	$p(3,4) = 3/5$
$(3,5)$	$p(3,5) = 1/2$
$(4,5)$	$p(4,5) = 1/5$

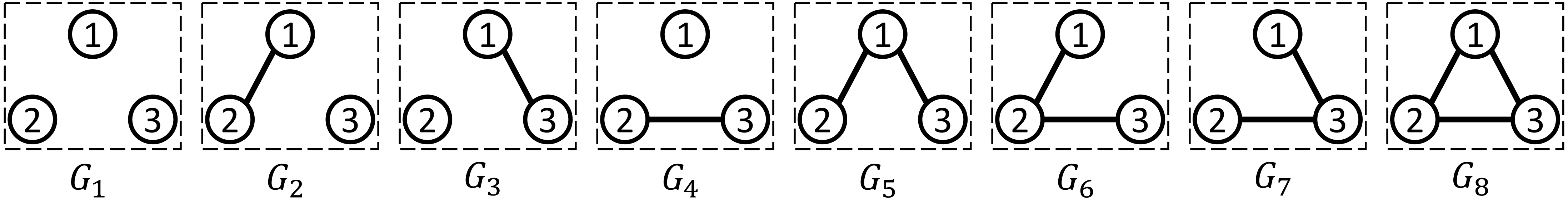
Node	Probability
1	$g(1) = 1/2$
2	$g(2) = 1/2$
3	$g(3) = 1/2$
4	$g(4) = 3/5$
5	$g(5) = 4/5$



Closed-Form Triangle Count Computation

- **Theorem:** With local binding, we are able to compute the *closed-form expected number of triangles* in a generated graph
- Linearity of expectation → Only need to compute the probability of each triangle being generated → Sum up the probabilities
- **Fact:** The probability of forming a triangle only depends on how the three node pairs are grouped

Example Revisited



Node pair	Probability
$(1,2)$	$p(1,2) = 1/2$
$(1,3)$	$p(1,3) = 1/4$
$(2,3)$	$p(2,3) = 1/2$

RGM	$\Pr[G_1]$	$\Pr[G_2]$	$\Pr[G_3]$	$\Pr[G_4]$	$\Pr[G_5]$	$\Pr[G_6]$	$\Pr[G_7]$	$\Pr[G_8]$
RGM_1	3/16	3/16	1/16	3/16	1/16	3/16	1/16	1/16
RGM_2	1/4	1/8	0	1/4	1/8	1/8	0	1/8
RGM_3	1/2	0	0	0	0	1/4	0	1/4

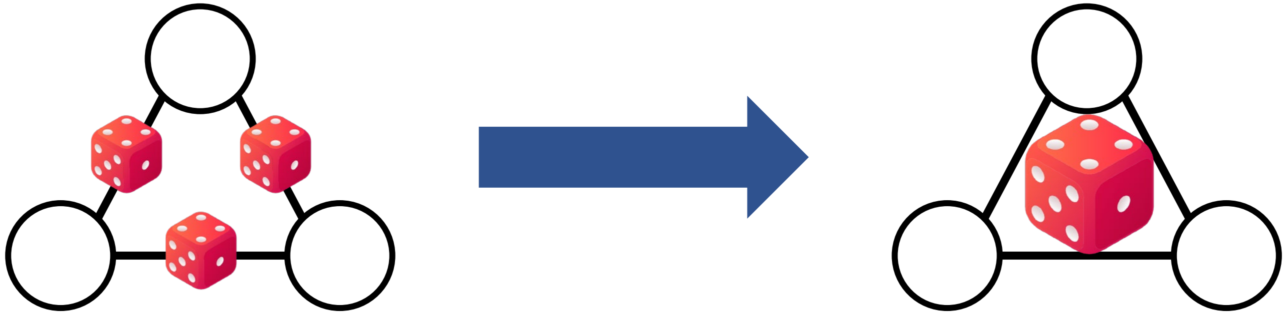
- **Fact:** The probability of forming a triangle only depends on how the three node pairs are grouped, e.g.,
  - RGM_1: All separated (three groups $\{\{1,2\}\}$ / $\{\{1,3\}\}$ / $\{\{2,3\}\}$) → $\Pr[\triangle] = 1/16$
  - RGM_2: Partially grouped (two groups $\{\{1,2\},\{1,3\}\}$ / $\{\{2,3\}\}$) → $\Pr[\triangle] = 1/8$
  - RGM_3: All grouped (a single group $\{\{1,2\},\{1,3\},\{2,3\}\}$) → $\Pr[\triangle] = 1/4$

Closed-Form Triangle Count Computation

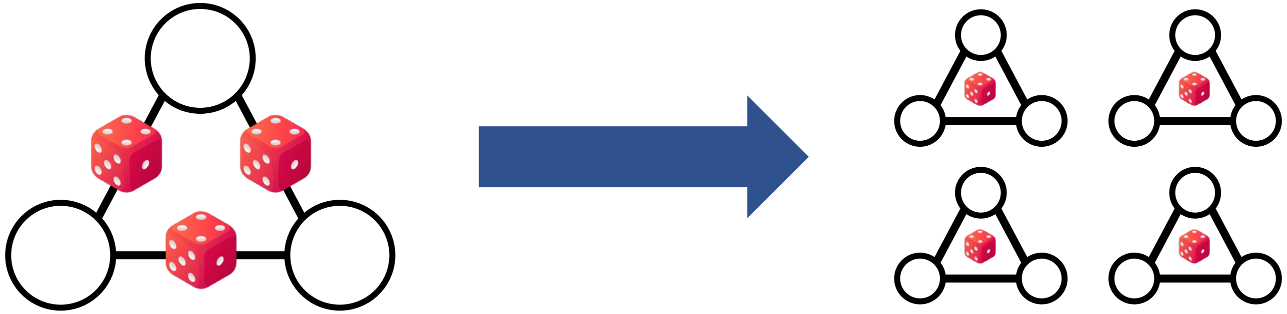
- **Theorem:** With local binding, we are able to compute the *closed-form expected number of triangles* in a generated graph
- **Fact:** The probability of forming a triangle only depends on how the three node pairs are grouped
- → Only need to compute the probability of each possible grouping
- → After getting the probability of each possible grouping
- $\Pr[\triangle] = \sum_{\text{possible grouping } \mathcal{P}} \Pr[\mathcal{P}] \Pr[\triangle | \mathcal{P}]$
    - Sum of $\Pr[\text{getting that grouping}] \cdot \Pr[\text{triangle under that grouping}]$

Parallel Binding: Easily Parallelizable Variant

- **It is non-trivial to parallelized local binding**, due to the *temporal dependency* in the generation process
  - Specifically, in a round, whether an edge is generated *depends on* whether it is grouped in a previous round
- We propose **parallel binding**, an easily-parallelizable version of binding



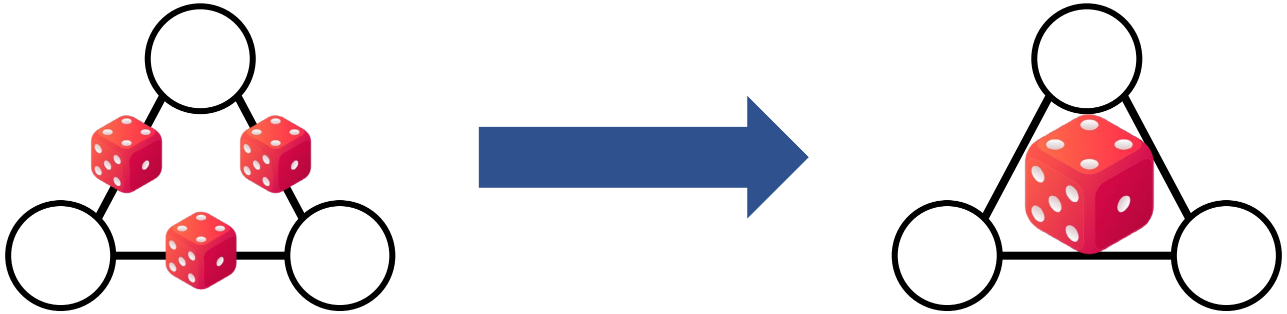
Local binding

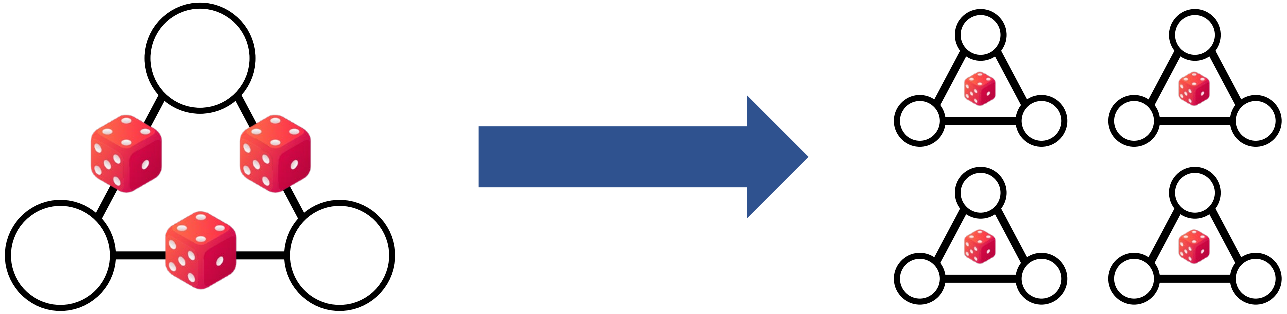


Parallel binding

Parallel Binding: Key Idea

- **Key idea:** Make the rounds *temporally independent*, so that multiple rounds can be processed in a parallel manner
- In *every round*, each pair (i, j) is possible to be grouped, and the corresponding edge is possible to be generated
- So that, the marginal edge probability $p(i, j)$ is satisfied *accumulated over the rounds*





Parallel Binding v.s. Local Binding

- Both parallel binding and local binding (1) preserve marginal edge probabilities and (2) impose edge dependency, but they are mathematically distinct and result in different random graph models
- Between the two variants, neither is always superior over the other, but we recommend parallel binding if efficiency is a major concern

Parameter Fitting

- **Parameters:** Edge probabilities p, node-sampling probabilities g, and the number of rounds R
- We assume edge probabilities p are given or obtained from some edge-probability model (e.g., Erdős-Rényi or Chung-Lu)
- We manually set the number of rounds R
- **Variables:** We only fit the node-sampling probabilities g
- **Objective:** The expected number of triangles, for which we have derived theoretical results

Parameter Fitting: Intuition

- **Q:** Why do we use the number of triangles as the objective?
- **Recall the question we had in the beginning:** Can we go beyond edge independency, breaking through the *limitations*, but still keeping the *merits* of the edge independent graph models?
- **Merits (what edge-independent models can already do well):** Heavy-tailed degree distribution and small diameter
- **Limitations (what edge-independent models cannot do well):** High clustering (e.g., higher triangle-density)
- → So we obtain the edge probabilities from those models to maintain their merits, while focusing on improving w.r.t. clustering

Experimental Settings

- **Datasets:** Real-world graphs from different domains
  - **Social networks:** Hamster and Facebook
  - **Web graphs:** Pol-blogs and Spam
  - **Biological graphs:** CE-PG and SC-HT
- **Clustering metrics:** Number of triangles (△), global clustering coefficient (GCC), and average local clustering coefficient (ALCC)

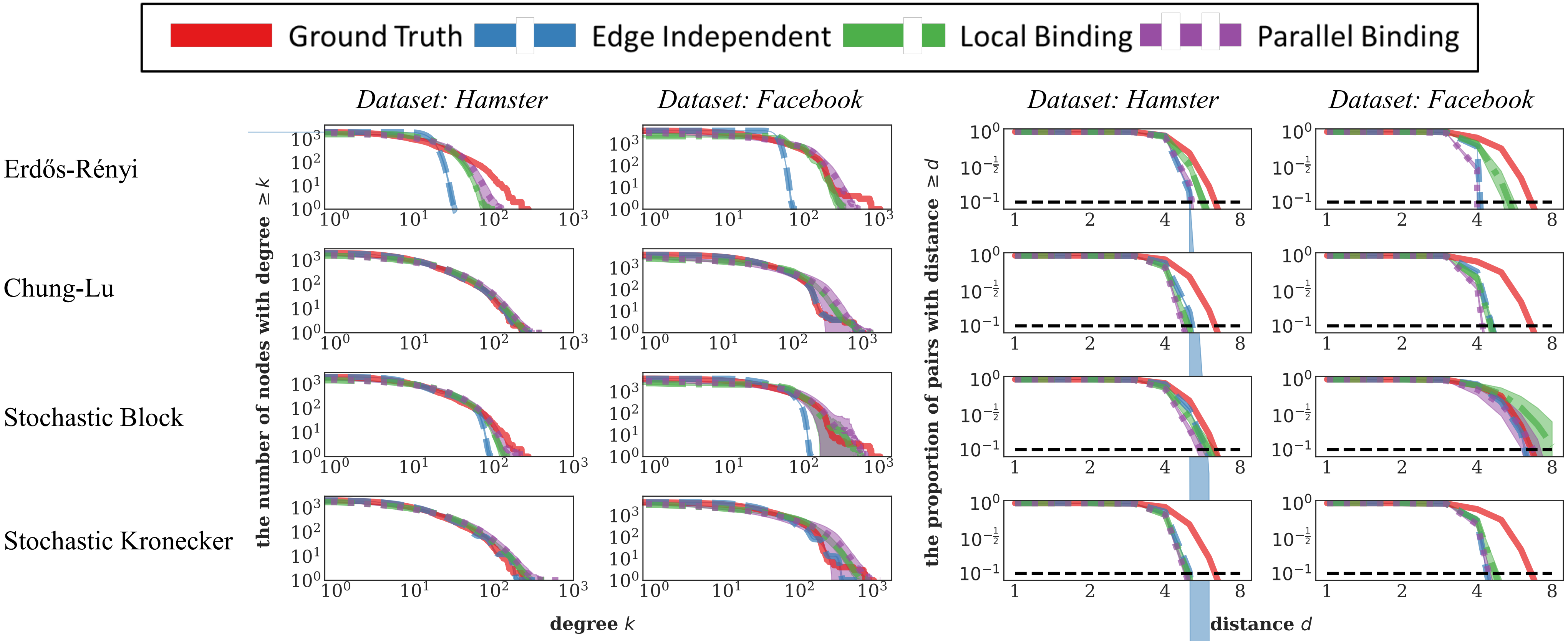
dataset	$\|V\|$	$\|E\|$	△	GCC	ALCC
Hamster	2,000	16,097	157,953	0.229	0.540
Facebook	4,039	88,234	4,836,030	0.519	0.606
Pol-blogs	1,222	16,717	303,129	0.226	0.320
Spam	4,767	37,375	387,051	0.145	0.286
CE-PG	1,692	47,309	2,353,812	0.321	0.447
SC-HT	2,077	63,023	4,192,980	0.377	0.350

Fitting and Graph Generation Processes

- **Edge-probability models:** Erdős-Rényi model, Chung-Lu model, stochastic block model, and stochastic Kronecker model
- **Edge-dependency mechanisms:** Edge independent, local binding, and parallel binding
- **Fitting 1:** Given an input graph, for each edge-probability model, we fit the parameters of the model → marginal edge probabilities p
- **Fitting 2:** Given p obtained above, we optimize node-sampling probabilities g so that the expected number of triangles in a generated graph matches the ground truth in the input graph
- **Graph generation:** We generate random graphs with binding using p and g, and we also generate graphs using the edge independent graph model with p only (i.e., $g(v) = 0$ for each node v)

Results: Binding Maintains Realistic Degrees and Distances

- **Observation:** With binding, degree and distance distributions are largely maintained as in the edge-independent models → We "inherit" the merits of edge-independent models



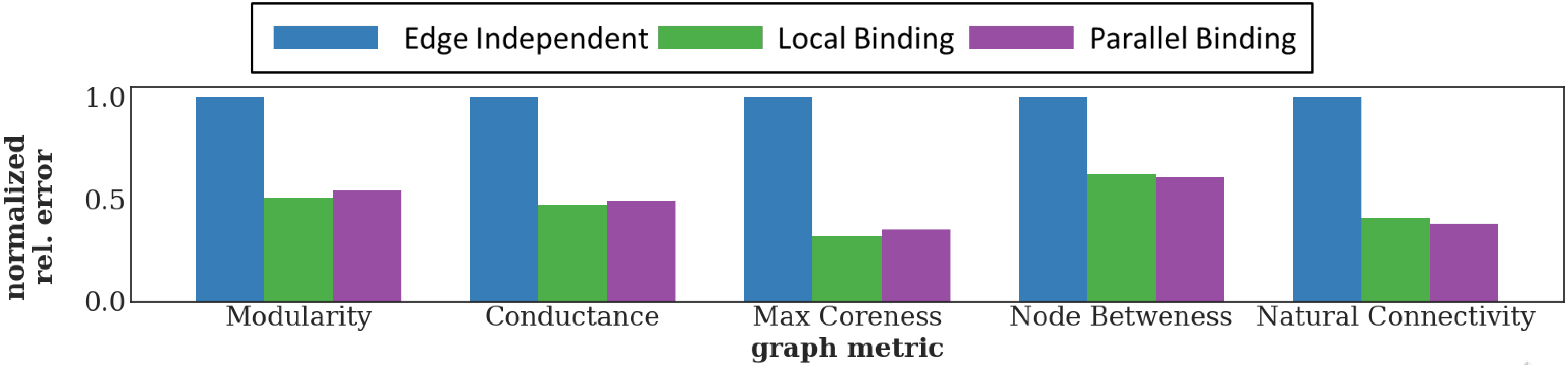
Results: Binding Improves Clustering

- **Observation:** With binding, the number of triangles (△; our fitting objective) is well fit, and both global clustering coefficient (GCC) and average local clustering coefficient (ALCC) are closer to the ground-truth values → We break the limitations of edge-independent models

dataset		*Hamster*			*Facebook*			*Pol-blogs*			*Spam*			*CE-PG*			*SC-HT*			Average Rank		
metric		△	GCC	ALCC	△	GCC	ALCC	△	GCC	ALCC	△	GCC	ALCC	△	GCC	ALCC	△	GCC	ALCC	△	GCC	ALCC
model	Ground Truth	1.00	0.23	0.54	1.00	0.52	0.61	1.00	0.23	0.32	1.00	0.14	0.29	1.00	0.32	0.45	1.00	0.38	0.35	N/A	N/A	N/A
Erdős-Rényi	Edge Independent	0.01	0.01	0.01	0.01	0.01	0.01	0.03	0.02	0.02	0.01	**0.00**	0.00	0.04	0.03	0.03	0.03	0.03	0.03	3.0	2.7	2.5
	Local Binding	**1.00**	**0.32**	0.24	1.01	0.45	0.22	0.95	**0.34**	**0.25**	0.99	0.34	**0.23**	**1.02**	**0.40**	**0.26**	1.01	**0.42**	**0.25**	1.7	**1.3**	**1.3**
	Parallel Binding	0.99	0.39	**0.64**	**1.00**	**0.57**	**0.81**	**1.02**	0.41	0.66	**0.99**	0.40	0.66	0.97	0.51	0.75	**0.99**	0.56	0.79	**1.3**	2.0	2.2
Chung-Lu	Edge Independent	0.30	0.07	0.06	0.12	0.06	0.06	0.79	0.18	0.17	0.50	0.07	0.06	0.68	0.23	0.22	0.64	0.24	**0.23**	3.0	3.0	2.5
	Local Binding	0.99	0.17	0.26	1.03	0.26	0.30	**1.00**	0.21	**0.34**	1.03	0.12	**0.26**	1.00	0.29	**0.43**	**1.04**	**0.32**	0.47	1.7	1.8	**1.5**
	Parallel Binding	**1.00**	**0.18**	**0.47**	**1.01**	**0.34**	**0.63**	1.01	**0.22**	0.47	**1.01**	**0.13**	0.44	**1.00**	**0.31**	0.58	1.14	0.29	0.61	**1.3**	**1.2**	2.0
Stochastic block	Edge Independent	0.26	0.08	0.04	0.15	0.14	0.08	0.48	0.14	0.16	0.53	0.09	0.04	0.66	0.26	0.20	0.64	0.27	0.13	3.0	3.0	3.0
	Local Binding	1.04	**0.22**	0.24	0.93	0.43	0.33	**0.99**	**0.24**	**0.35**	0.98	**0.15**	**0.22**	**0.99**	**0.32**	**0.41**	1.03	**0.35**	**0.39**	1.7	**1.2**	**1.3**
	Parallel Binding	**0.99**	0.24	**0.52**	**1.03**	**0.53**	**0.56**	1.01	0.18	0.25	**0.99**	0.16	0.36	1.05	0.33	0.36	**0.97**	0.34	0.44	**1.3**	1.8	1.7
Stochastic Kronecker	Edge Independent	0.18	0.04	0.06	0.05	0.04	0.04	0.10	0.04	0.07	0.06	0.01	0.03	0.13	0.07	0.12	0.03	0.03	0.05	3.0	3.0	3.0
	Local Binding	1.09	0.15	0.23	0.93	0.24	0.27	1.06	0.14	**0.23**	0.94	0.12	0.19	0.99	0.17	0.31	1.44	0.18	**0.28**	2.0	2.0	1.7
	Parallel Binding	**1.00**	**0.17**	**0.39**	**0.97**	**0.35**	**0.60**	**0.94**	**0.22**	0.42	**1.05**	**0.16**	**0.38**	**1.00**	**0.28**	**0.46**	**1.07**	**0.35**	0.58	**1.0**	**1.0**	**1.3**
Average Rank	Edge Independent	3.0	3.0	3.0	3.0	3.0	3.0	3.0	3.0	2.5	3.0	2.5	2.8	3.0	3.0	3.0	3.0	3.0	2.3	3.0	2.9	2.8
	Local Binding	1.8	**1.5**	2.0	2.0	2.0	2.0	**1.5**	**1.5**	**1.0**	2.0	**1.8**	**1.3**	**1.5**	**1.5**	**1.3**	1.8	**1.3**	**1.3**	1.8	1.6	**1.5**
	Parallel Binding	**1.3**	**1.5**	**1.0**	**1.0**	**1.0**	**1.0**	**1.5**	**1.5**	2.5	**1.0**	**1.8**	2.0	**1.5**	**1.5**	1.8	**1.3**	1.8	2.5	**1.3**	**1.5**	1.8

Results: Binding Improves Other Graph Metrics

- **Observation:** With binding, the overall values of various graph metrics get closer to the ground-truth values → We improve upon edge-independent models in various aspects
  - Averaged over the datasets
  - The relative error of "edge independent" is normalized as reference (1.0)



Conclusion

In this work, we…

- **Proposed new concepts:** Edge probability graph models (EPGMs) that keep marginal edge probabilities but go beyond edge dependency
- **Proposed binding framework:** A realistic and practical way to impose edge dependency by grouping nodes
- **Derived theoretical results:** Closed-form formula for the number of triangles in graphs generated using the binding framework
- **Developed efficient algorithms:** Fast parameter fitting by considering node equivalence in existing edge probability models

Appendix, Code, and Datasets: bit.ly/EPGM_ICDM25