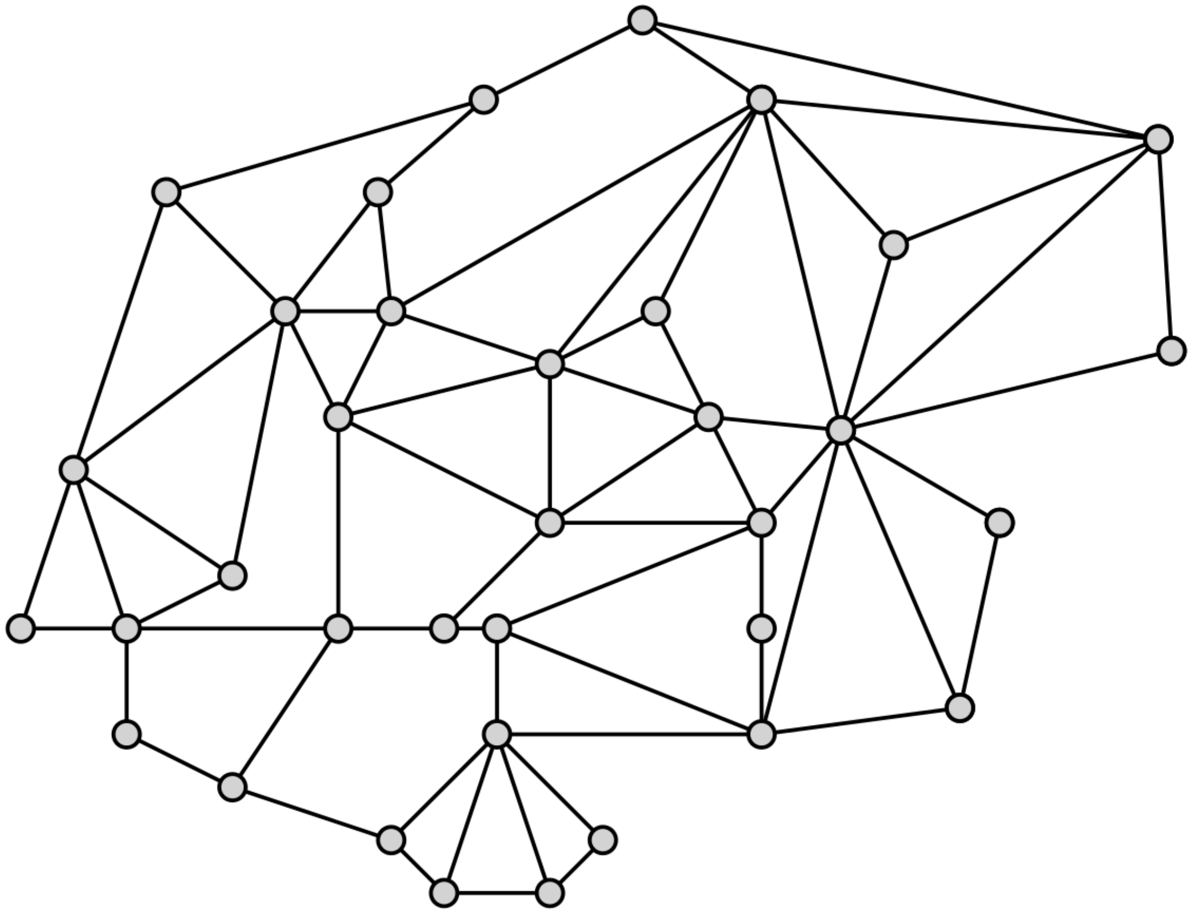
Sublogarithmic Distributed Algorithms for Lovász Local Lemma, and the Complexity Hierarchy

Manuela Fischer, Mohsen Ghaffari
ETH Zurich

LOCAL Model *Linial* [FOCS'87]

- undirected graph $G = (V, E)$,
  n nodes, maximum degree Δ
- synchronous message-passing rounds
- unbounded message size
- unbounded computation
- **Round Complexity**:
  number of rounds to solve the problem



Chang, Pettie [FOCS'17]

Completeness of Lovász Local Lemma for Sublogarithmic Problems

Chang, Pettie [FOCS'17]

Completeness of Lovász Local Lemma for Sublogarithmic Problems

Any algorithm for an LCL problem on bounded-degree graphs with round complexity $\boldsymbol{o}(\log \mathrm{n})$ can be **automatically sped up** to run in $\boldsymbol{O}(\mathrm{T}_{LLL}(\boldsymbol{n}))$ rounds.

Chang, Pettie [FOCS'17]

Completeness of Lovász Local Lemma for Sublogarithmic Problems

Any algorithm for an LCL problem on bounded-degree graphs with round complexity $\boldsymbol{o(\log n)}$ can be **automatically sped up** to run in $\boldsymbol{O(T_{LLL}(n))}$ rounds.

$\boldsymbol{T_{LLL}(n)}$ = round complexity of Lovász Local Lemma
on n events/nodes
under $pd^c < 1$ for some constant c,
for $d = O(1)$

Chang, Pettie [FOCS'17]

Completeness of Lovász Local Lemma for Sublogarithmic Problems

Any algorithm for an LCL problem on bounded-degree graphs with round complexity $\boldsymbol{o(\log n)}$ can be **automatically sped up** to run in $\boldsymbol{O(\mathrm{T}_{LLL}(n))}$ rounds.

$\boldsymbol{T_{LLL}(n)}$ = round complexity of Lovász Local Lemma
on n events/nodes
under $pd^c < 1$ for some constant c,
for $d = O(1)$

polynomial criterion

Chang, Pettie [FOCS'17]

Completeness of Lovász Local Lemma for Sublogarithmic Problems

Any algorithm for an LCL problem on bounded-degree graphs with round complexity $\boldsymbol{o}(\mathbf{log\ n})$ can be **automatically sped up** to run in $\boldsymbol{O}(\mathrm{T}_{\boldsymbol{LLL}}(\boldsymbol{n}))$ rounds.

Naor, Stockmeyer '95
Locally Checkable Labeling (LCL)
solution checkable in $O(1)$ rounds

$\boldsymbol{T_{LLL}}(\boldsymbol{n})$ = round complexity of Lovász Local Lemma
on n events/nodes
under $pd^c < 1$ for some constant c,
for $d = O(1)$

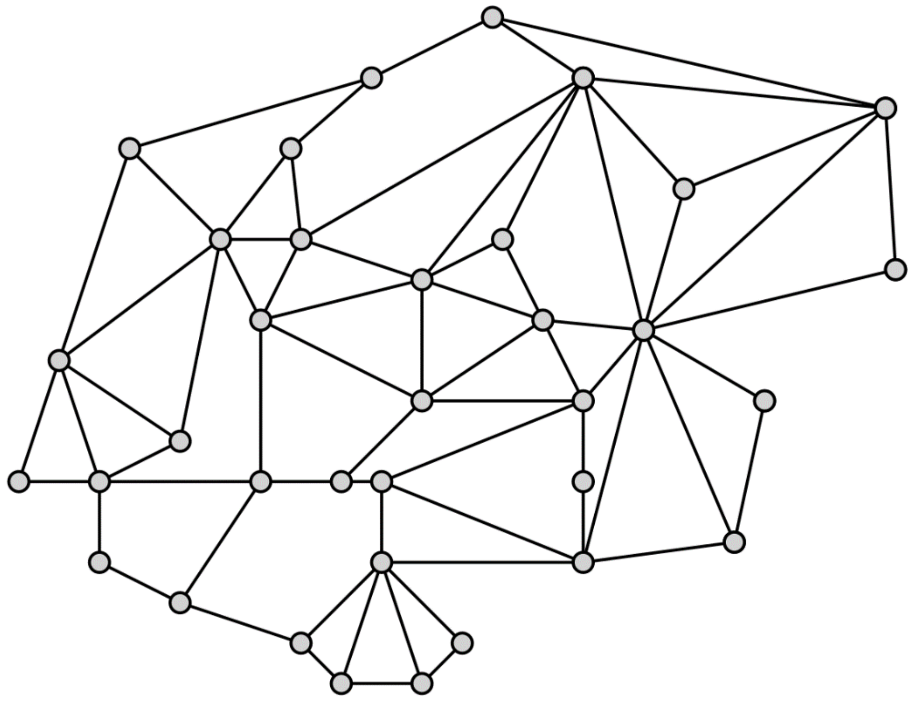
polynomial criterion

Lovász Local Lemma (LLL) *Erdős, Lovász '75*

Lovász Local Lemma (LLL)

Erdős, Lovász '75

independent variables $\mathcal{V}$ (w.l.o.g. fair coins)



Not too likely bad events:

n bad events $\mathcal{X}$ with $vbl(A) \subseteq \mathcal{V}$ for all $A \in \mathcal{X}$

$\Pr[A] \leq p$ for all $A \in \mathcal{X}$

Not too many dependencies:

dependency graph $G = (\mathcal{X}, E)$

$E = \{(A, B) : vbl(A) \cap vbl(B) \neq \emptyset\}$

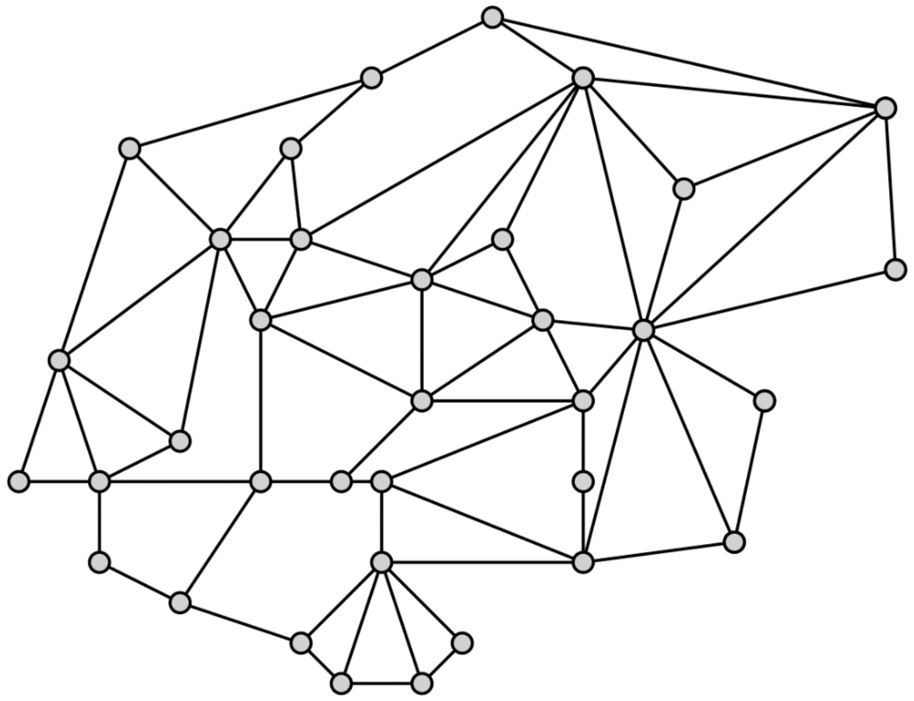
maximum degree d

If local union bound (with some slack) is satisfied, then all bad events can be avoided!

If $\mathrm{epd} \leq 1$, then $\Pr[\bigcap_{A \in \mathcal{X}} \bar{A}\,] > 0$.

Lovász Local Lemma (LLL) *Erdős, Lovász '75*

independent variables $\mathcal{V}$ (w.l.o.g. fair coins)



Not too likely bad events:

n bad events $\mathcal{X}$ with $vbl(A) \subseteq \mathcal{V}$ for all $A \in \mathcal{X}$

$\Pr[A] \leq p$ for all $A \in \mathcal{X}$

Not too many dependencies:

dependency graph $G = (\mathcal{X}, E)$

$E = \{(A, B): vbl(A) \cap vbl(B) \neq \emptyset\}$

maximum degree d

If local union bound (with some slack) is satisfied, then all bad events can be avoided!

If $\mathrm{epd} \leq 1$, then $\Pr[\bigcap_{A \in \mathcal{X}} \bar{A}\,] > 0$.

standard criterion

LOCAL Complexity of the Lovász Local Lemma

$\boldsymbol{O}(\log^2 \text{n})$ for $\boldsymbol{epd \le 1}$ **(STANDARD)** *Moser, Tardos* [JACM'10]	$\boldsymbol{O}(\log \text{n})$ for $\boldsymbol{epd^2 < 1}$ **(POLYNOMIAL)** *Chung, Pettie, Su* [PODC'14]	$\boldsymbol{O}(\log \text{n}/\text{loglog n})$ for $\boldsymbol{p2^d < 1}$ **(EXPONENTIAL)** *Chung, Pettie, Su* [PODC'14]
	CONJECTURE: $\boldsymbol{T_{LLL}(n) = O(\log \log \text{n})}$ *Chang, Pettie* [FOCS'16]	$\boldsymbol{\Omega}(\log \log \text{n})$ for $\boldsymbol{p2^d < 1}$ **(EXPONENTIAL)** *Brandt et al.* [STOC'16]

$\boldsymbol{T_{LLL}(n)}$ = round complexity of Lovász Local Lemma
on n events/nodes
under $pd^c < 1$ for some constant c,
for $d = O(1)$

Our Results

$$\boldsymbol{T_{LLL}(n) = 2^{O\left(\sqrt{\log\log n}\right)}}$$

$\boldsymbol{T_{LLL}(n)}$ = round complexity of Lovász Local Lemma
on n events/nodes
under $pd^c < 1$ for some constant c,
for $d = O(1)$

Our Results

$$T_{LLL}(n) = 2^{O(\sqrt{\log\log n})}$$

Automatic Speedup

$T_{LLL}(n)$ = round complexity of Lovász Local Lemma
on n events/nodes
under $pd^c < 1$ for some constant c,
for $d = O(1)$

Our Results

$T_{LLL}(n) = 2^{O(\sqrt{\log\log n})}$

Automatic Speedup ⇒

Gap in Distributed Complexity Hierarchy

for LCLs on bounded-degree graphs

$o(\log n) \rightarrow 2^{O(\sqrt{\log\log n})}$

$o(\log\log n) \rightarrow O(\log^* n)$

$T_{LLL}(n)$ = round complexity of Lovász Local Lemma
on n events/nodes
under $pd^c < 1$ for some constant c,
for $d = O(1)$

Other Applications

$2^{O(\sqrt{\log\log n})}$ - round Graph Coloring Algorithms

- f-defective $O\left(\frac{\Delta}{f}\right)$-coloring
- β-frugal $120\,\Delta^{1+\frac{1}{\beta}}$-coloring
- List-vertex-coloring

Previously best known $O(\log n)$ by *Chung, Pettie, Su* [PODC'14]

Algorithm for Lovász Local Lemma

BASE ALGORITHM

$\mathrm{O}(\mathrm{d}^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O\left(\sqrt{\log\log n}\right)}$ rounds

under $p(e \cdot d)^{4\lambda} < 1$

under polynomial criterion $\boldsymbol{\lambda = O(1)}$:

$\gg 2^{O\left(\sqrt{\log\log n}\right)}$ rounds

Algorithm for Lovász Local Lemma

BASE ALGORITHM

$O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O(\sqrt{\log\log n})}$ rounds

under $p(e \cdot d)^{4\lambda} < 1$

BOOTSTRAPPING:
Automatic
Speedup

under polynomial criterion $\lambda = O(1)$:
$\gg 2^{O(\sqrt{\log\log n})}$ rounds

Algorithm for Lovász Local Lemma

BASE ALGORITHM

$O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O(\sqrt{\log\log n})}$ rounds

under $p(e \cdot d)^{4\lambda} < 1$

BOOTSTRAPPING:
Automatic Speedup

$\Rightarrow$

$$T_{LLL}(n) = 2^{O(\sqrt{loglogn})}$$

$2^{O(\sqrt{\log\log n})}$ - round algorithm
under $p(e \cdot d)^{32} < 1$ (POLYNOMIAL)
for $d = O(\log^{1/5} \log n)$

under polynomial criterion $\lambda = O(1)$:
$\gg$ **$2^{O(\sqrt{\log\log n})}$ rounds**

BASE ALGORITHM $O(\mathrm{d}^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O\left(\sqrt{\log \log n}\right)}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

Shattering Technique, rooted in breakthrough LLL algorithm of Beck [RSA'91]

BASE ALGORITHM

$O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O(\sqrt{\log \log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

Shattering Technique, rooted in breakthrough LLL algorithm of Beck [RSA'91]

I. PARTIAL SAMPLING

assign values to subset of variables
such that remainder graph
1) satisifes polynomial criterion
2) consists of "small" components

in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$

BASE ALGORITHM $O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O(\sqrt{\log \log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

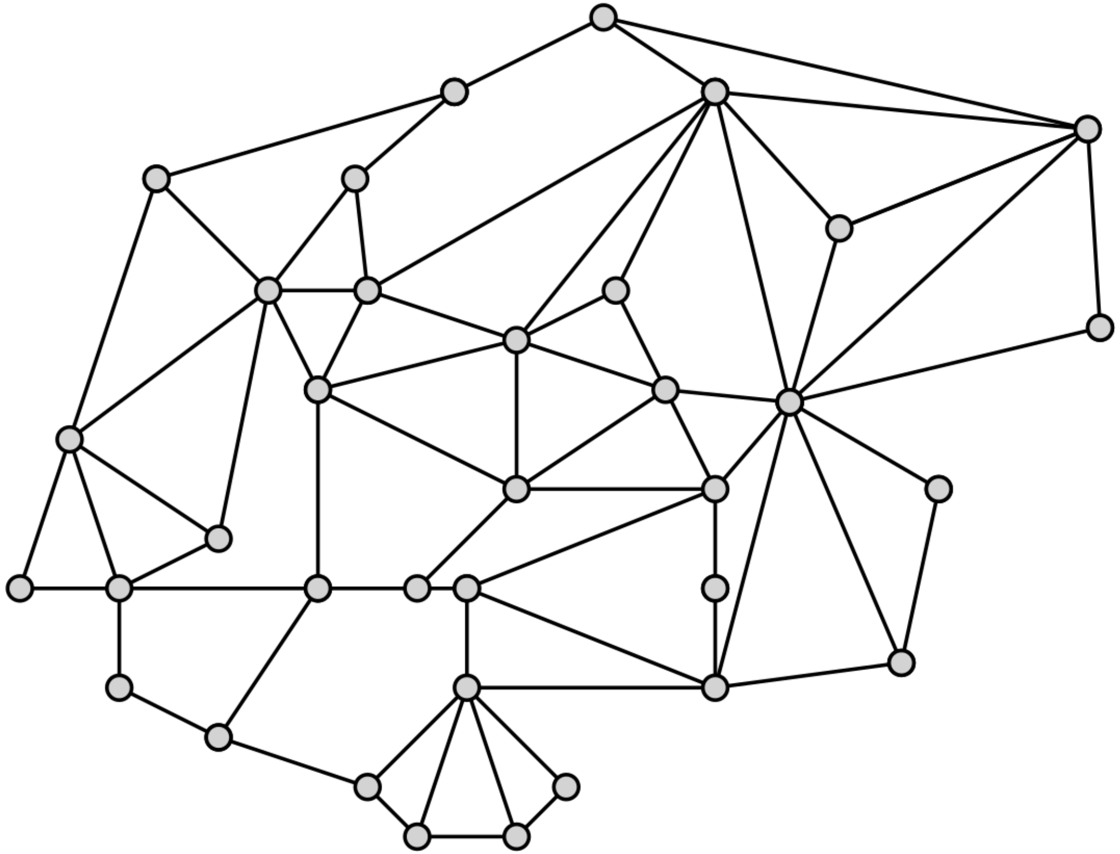
Shattering Technique, rooted in breakthrough LLL algorithm of Beck [RSA'91]

I. PARTIAL SAMPLING

assign values to subset of variables such that remainder graph

1) satisifes polynomial criterion
2) consists of "small" components

in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$



BASE ALGORITHM $O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O\left(\sqrt{\log\log n}\right)}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

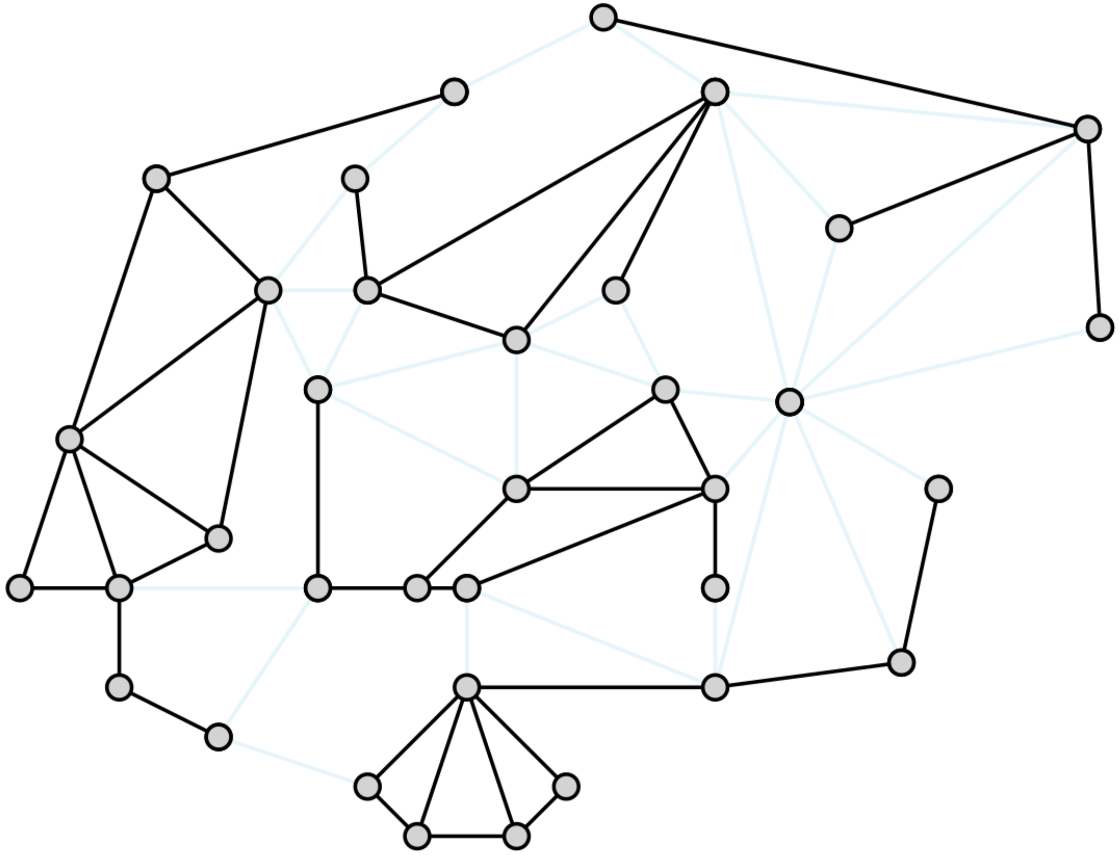
Shattering Technique, rooted in breakthrough LLL algorithm of Beck [RSA'91]

I. PARTIAL SAMPLING

assign values to subset of variables such that remainder graph

1) satisifes polynomial criterion
2) consists of "small" components

in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$



BASE ALGORITHM $O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O\left(\sqrt{\log\log n}\right)}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

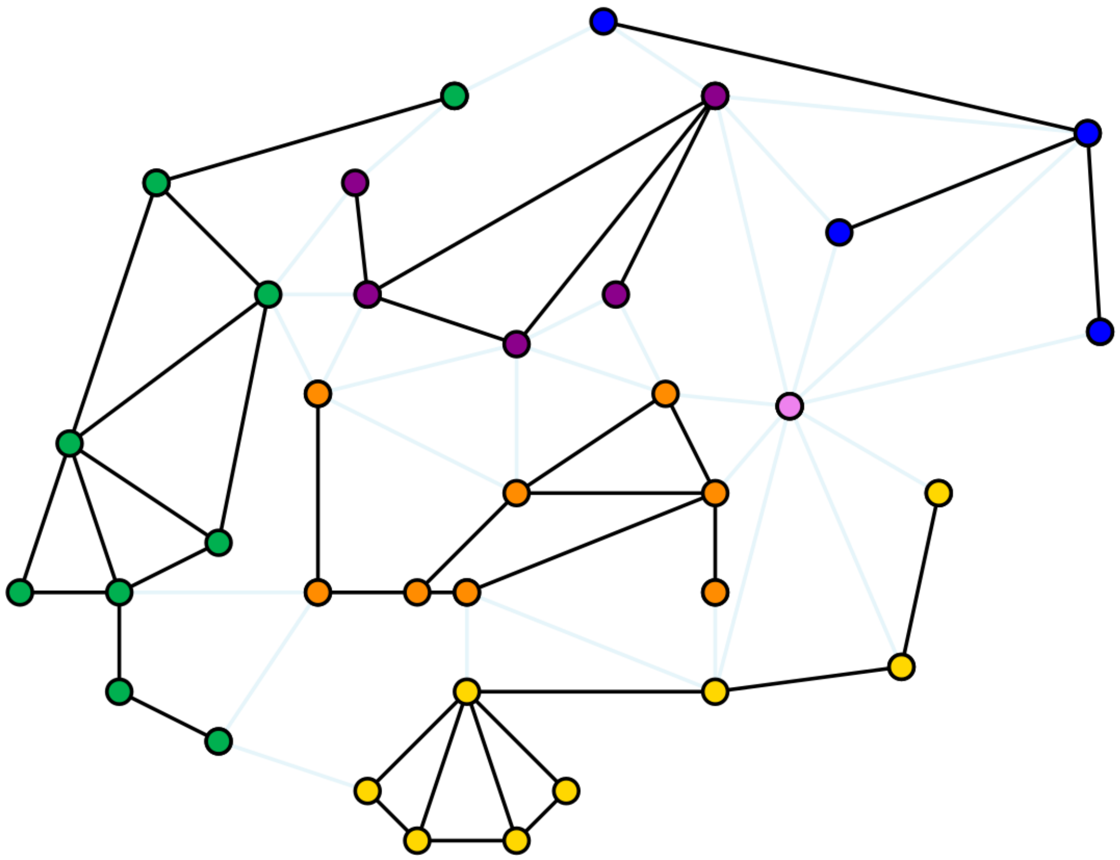
Shattering Technique, rooted in breakthrough LLL algorithm of Beck [RSA'91]

I. PARTIAL SAMPLING

assign values to subset of variables such that remainder graph

1) satisifes polynomial criterion
2) consists of "small" components

in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$



BASE ALGORITHM

$O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O(\sqrt{\log\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

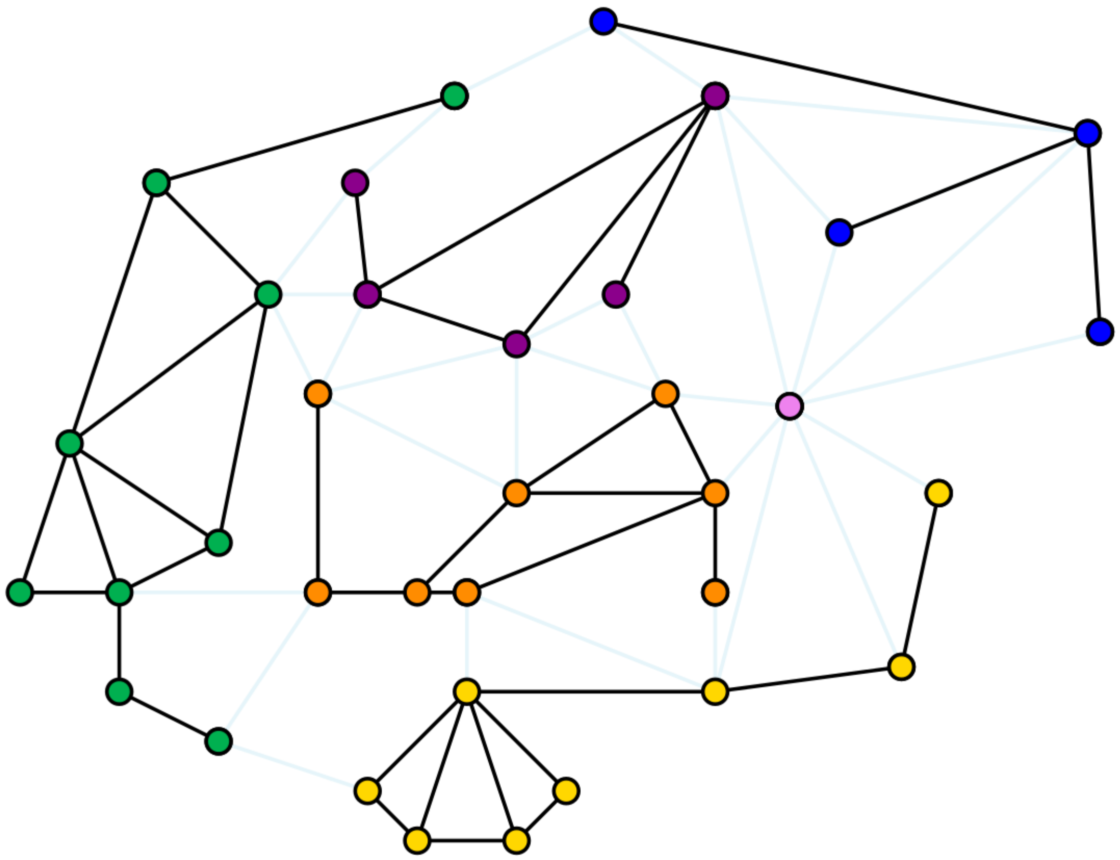
Shattering Technique, rooted in breakthrough LLL algorithm of Beck [RSA'91]

I. PARTIAL SAMPLING

assign values to subset of variables such that remainder graph

1) satisifes polynomial criterion
2) consists of "small" components

in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$



BASE ALGORITHM

$O(d^2) + \lambda \cdot \log^{\frac{1}{\lambda}} n \cdot 2^{O(\sqrt{\log\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

Shattering Technique, rooted in breakthrough LLL algorithm of Beck [RSA'91]

I. PARTIAL SAMPLING

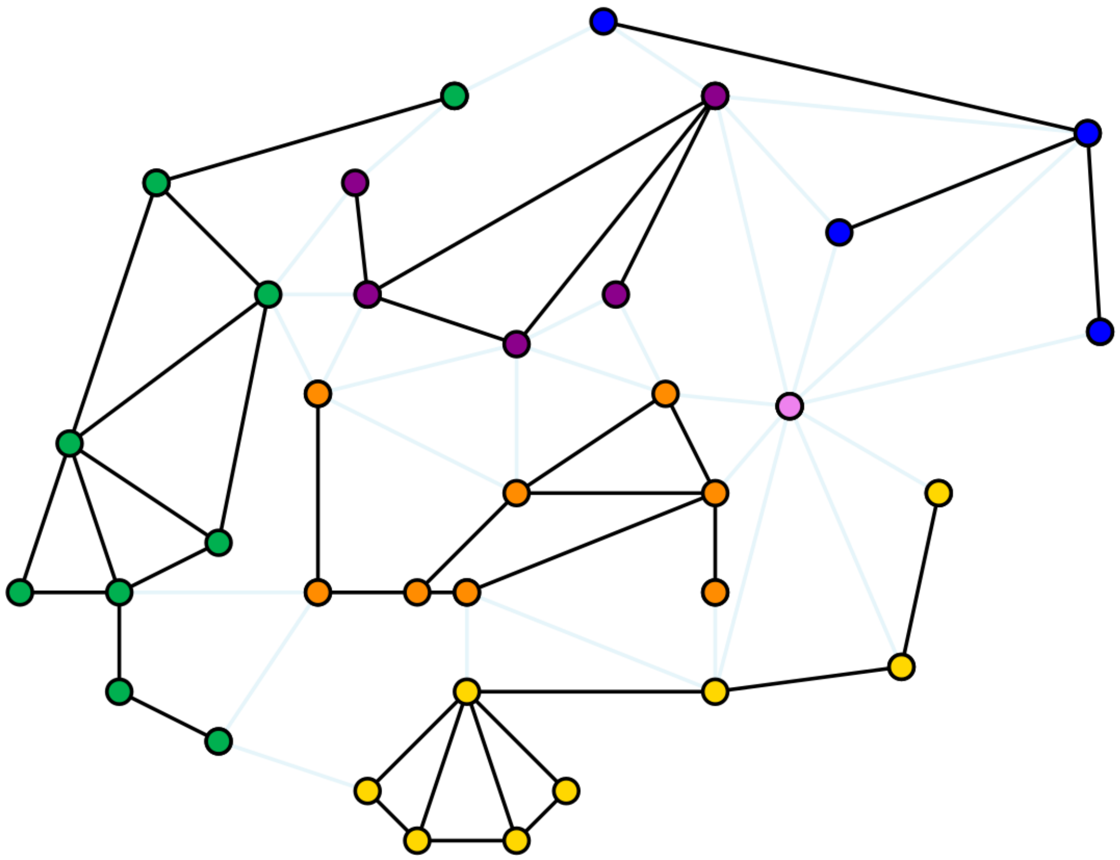
assign values to subset of variables such that remainder graph

1) satisifes polynomial criterion
2) consists of "small" components

in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$

II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ under $p(e \cdot d)^{4\lambda} < 1$



I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder
1) satisifes polynomial LLL
2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]

I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder 1) satisifes polynomial LLL 2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]

SEQUENTIAL PARTIAL SAMPLING

iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

block $\mathrm{vbl(A)}$

I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder 1) satisifes polynomial LLL 2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]

SEQUENTIAL PARTIAL SAMPLING

iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

block $\mathrm{vbl(A)}$

dangerous!

I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder 1) satisifes polynomial LLL 2) has "small" components

Inspired by Molloy and Reed's
Sequential Partial Sampling [STOC'98]

SEQUENTIAL PARTIAL SAMPLING

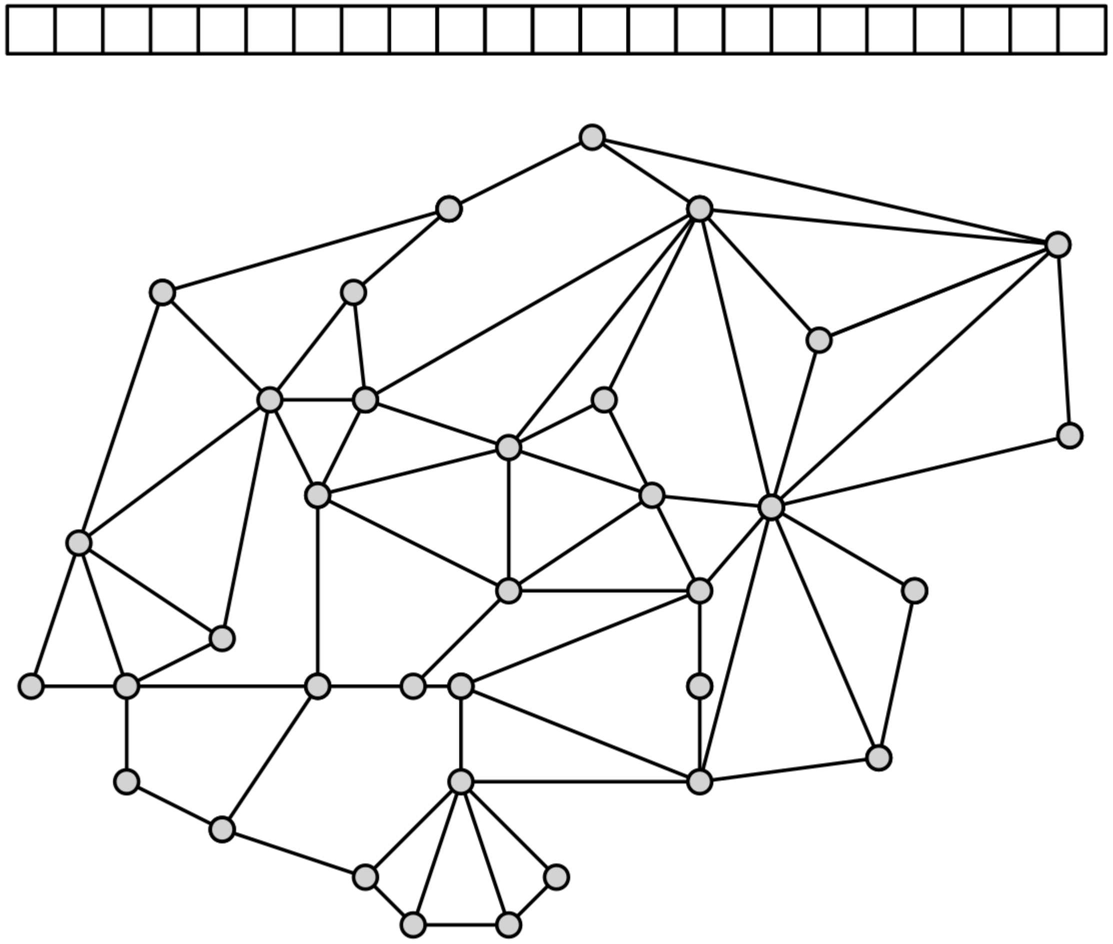
iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

if $\boldsymbol{P}[\boldsymbol{A} \mid \textbf{flipped coins}] > \frac{\sqrt{\boldsymbol{p}}}{\mathbf{2}}$

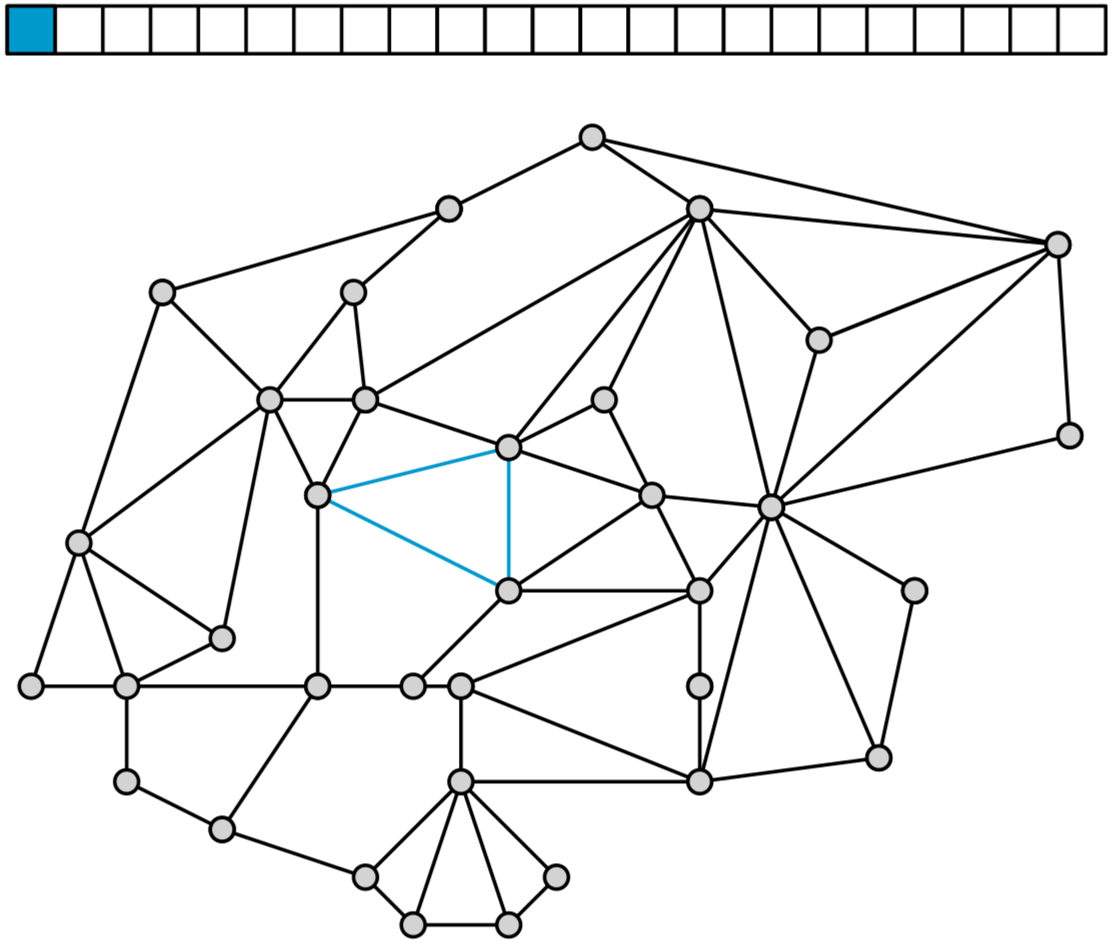
block vbl(A)

dangerous!



I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder 1) satisifes polynomial LLL 2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]



SEQUENTIAL PARTIAL SAMPLING

iterativety, for unblocked v $\in \mathcal{V}$

flip coin for v

if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

block vbl(A)

dangerous!

I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder
1) satisifes polynomial LLL
2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]

SEQUENTIAL PARTIAL SAMPLING

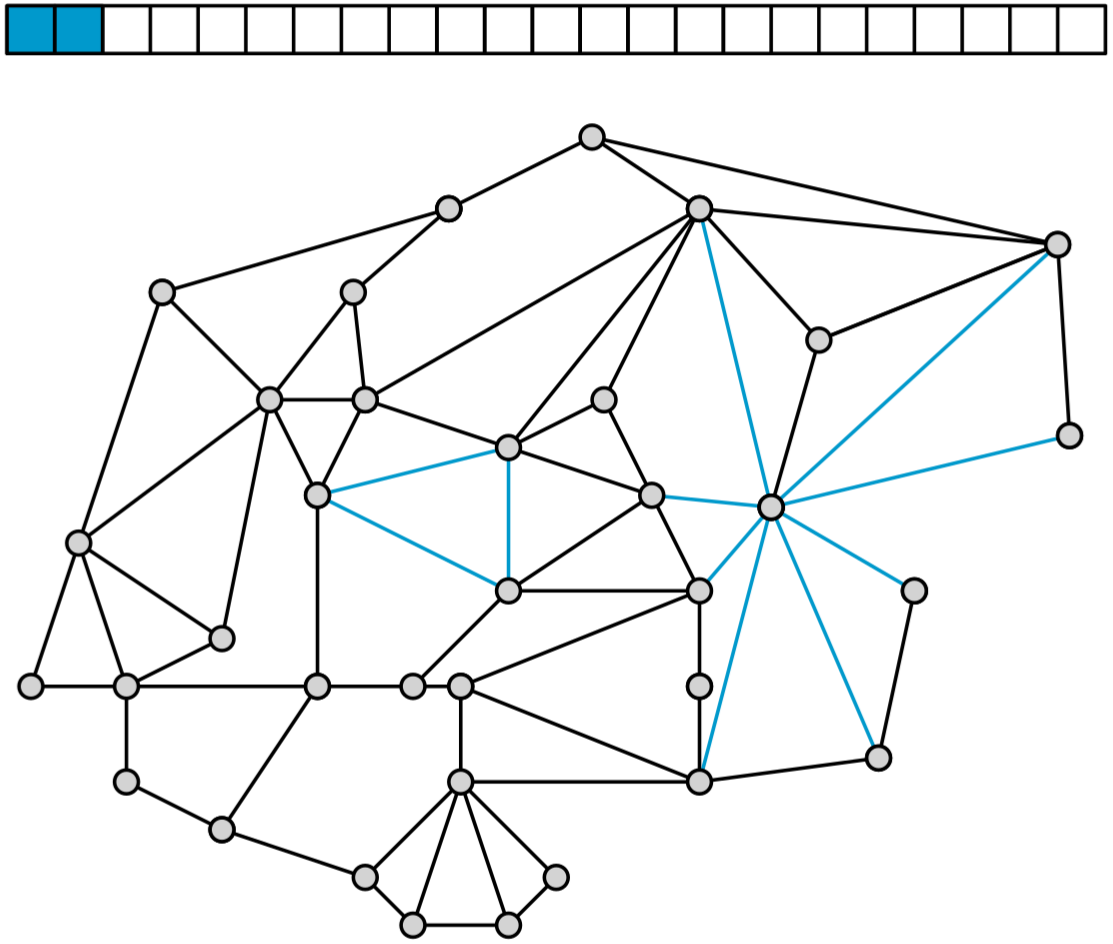
iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

if $\boldsymbol{P[A \mid \text{flipped coins}]} > \frac{\sqrt{p}}{2}$

block $\mathrm{vbl}(\mathrm{A})$

dangerous!



I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder
1) satisifes polynomial LLL
2) has "small" components

Inspired by Molloy and Reed's
Sequential Partial Sampling [STOC'98]

SEQUENTIAL PARTIAL SAMPLING

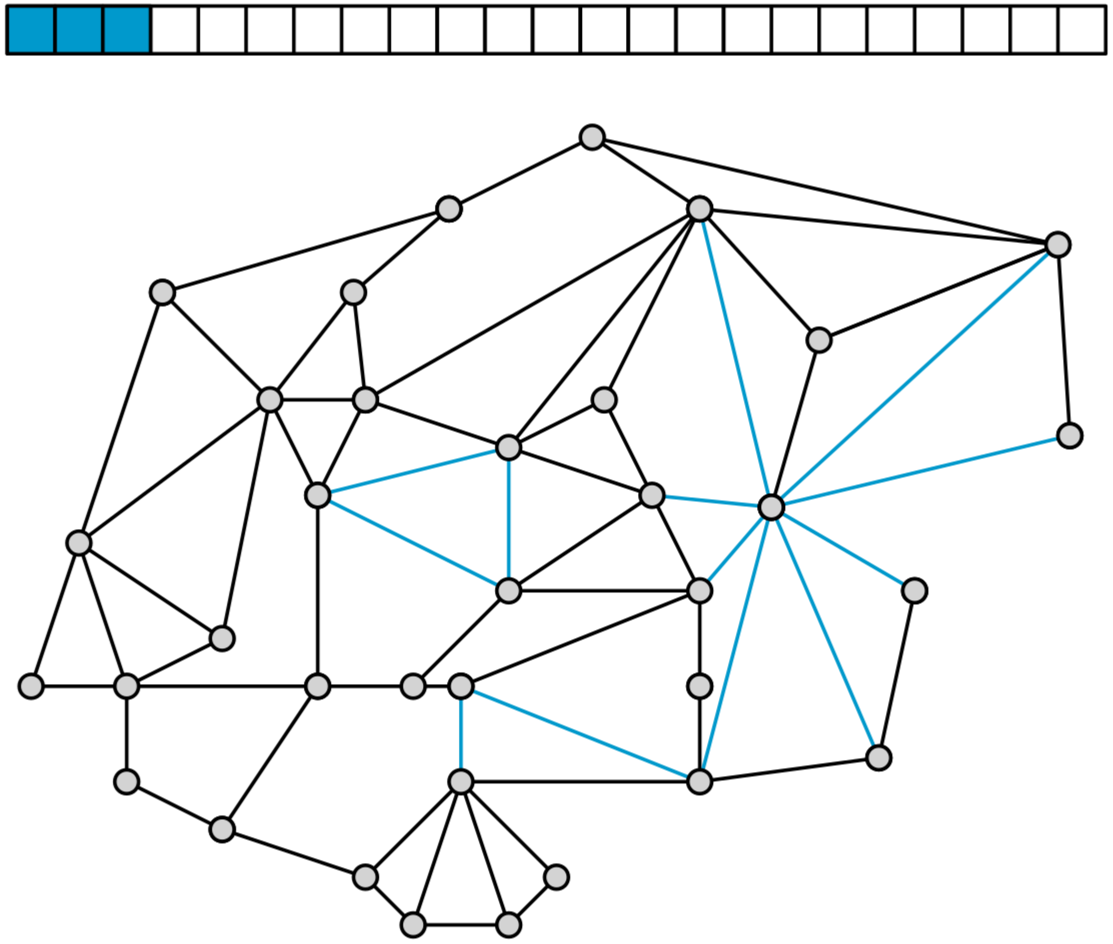
iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

block $\mathrm{vbl(A)}$

dangerous!



I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder

1) satisifes polynomial LLL
2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]

SEQUENTIAL PARTIAL SAMPLING

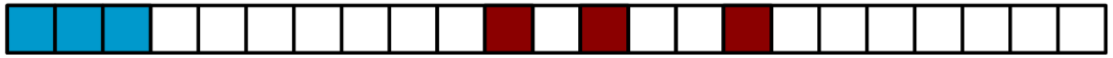
iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

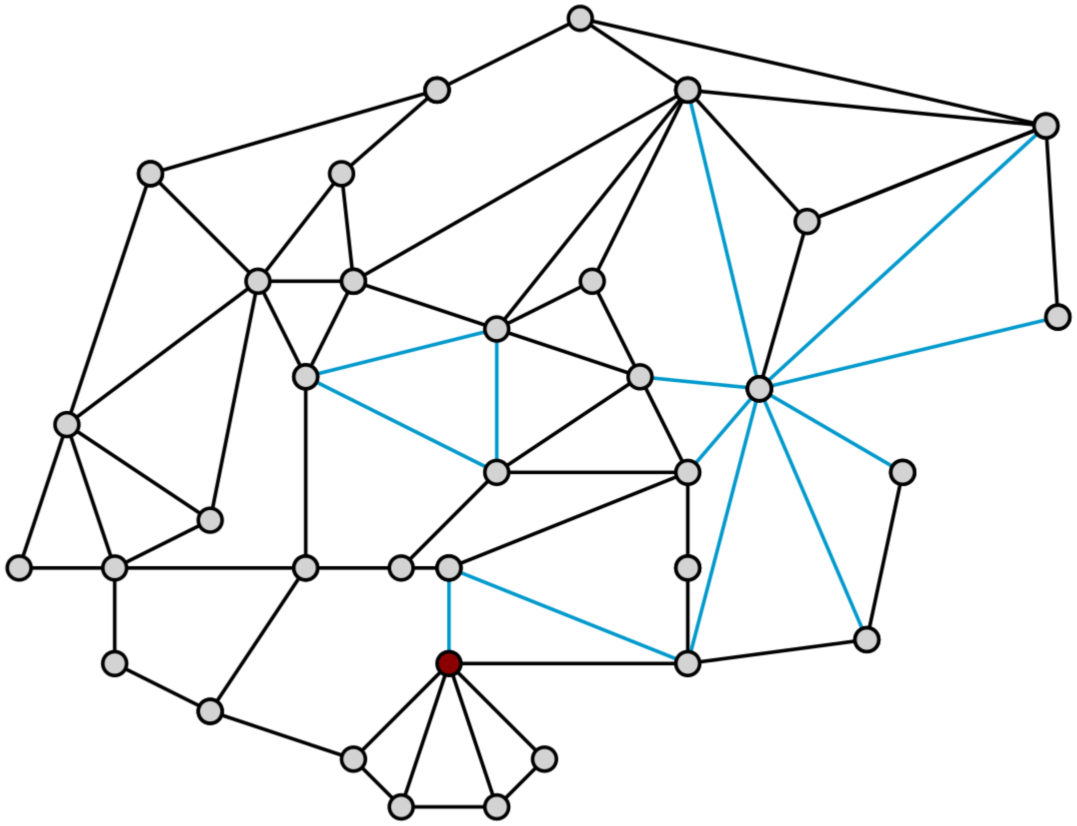
flip coin for v

if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

block $\mathrm{vbl(A)}$

dangerous!





I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder

1) satisifes polynomial LLL
2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]

SEQUENTIAL PARTIAL SAMPLING

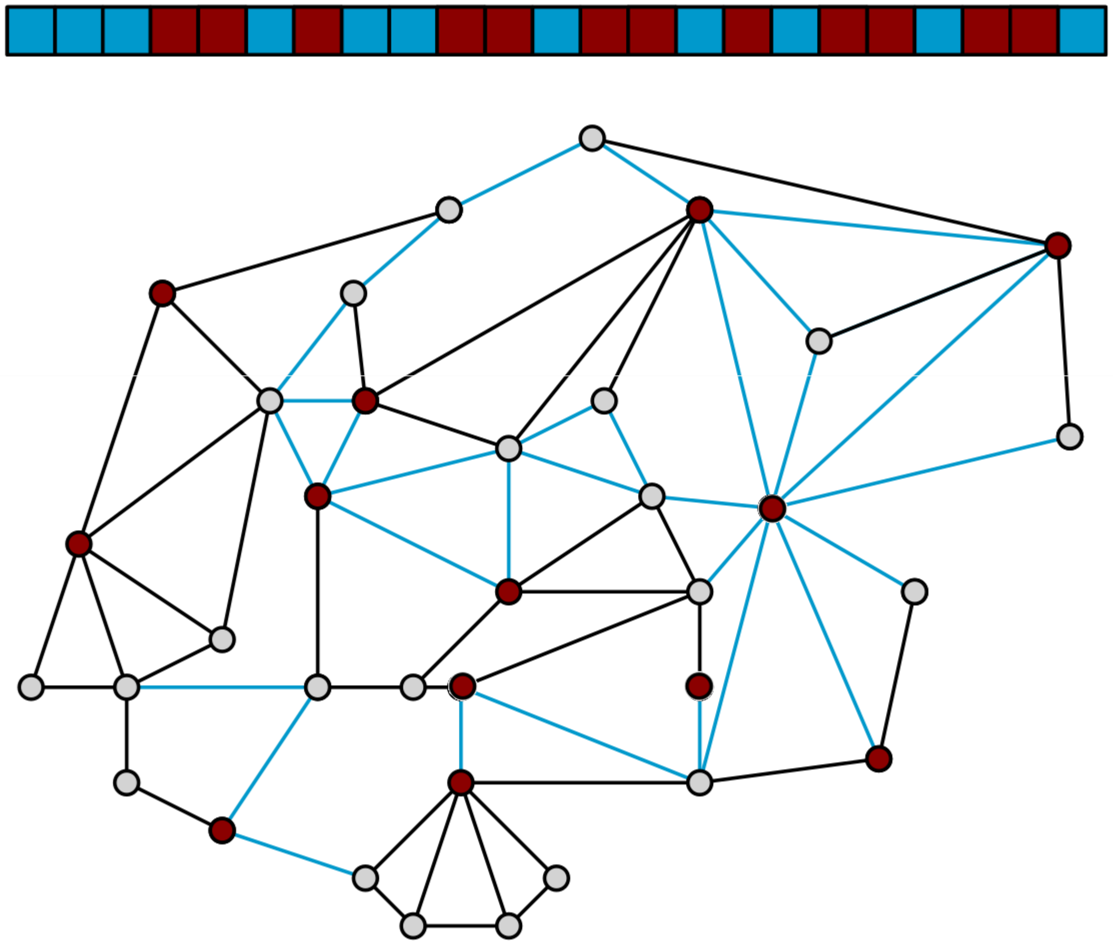
iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

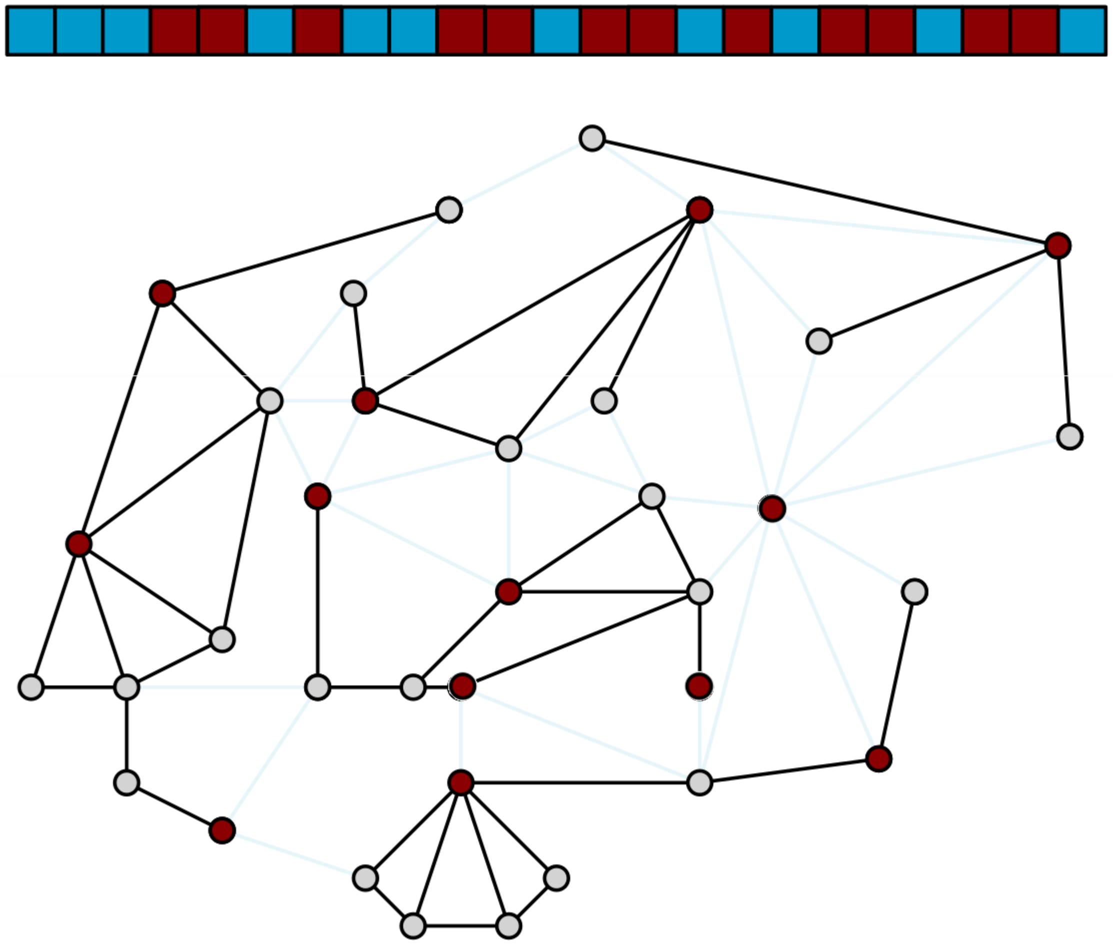
block $\mathrm{vbl(A)}$

dangerous!



I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder 1) satisifes polynomial LLL 2) has "small" components

Inspired by Molloy and Reed's
Sequential Partial Sampling [STOC'98]



SEQUENTIAL PARTIAL SAMPLING

iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

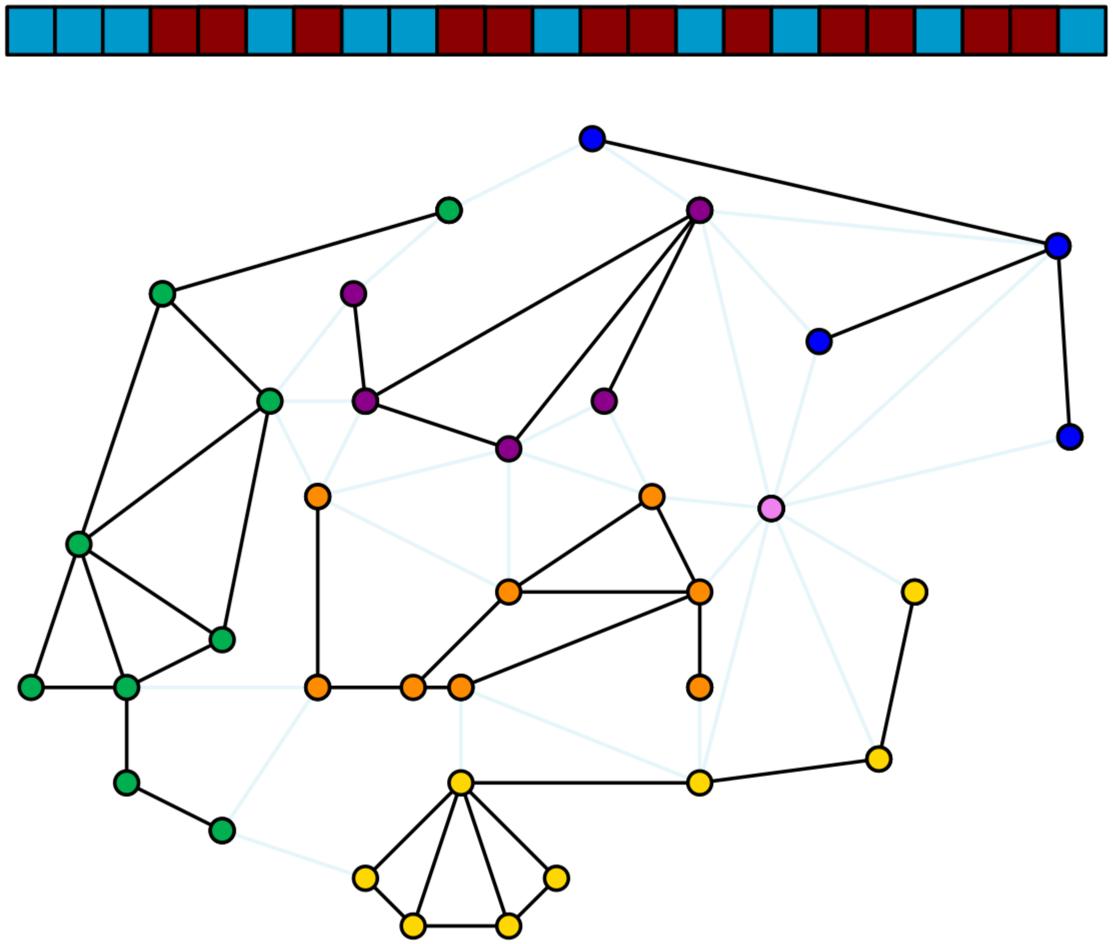
if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

block $\mathrm{vbl(A)}$

dangerous!

I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder 1) satisifes polynomial LLL 2) has “small“ components

Inspired by Molloy and Reed’s
Sequential Partial Sampling [STOC’98]



SEQUENTIAL PARTIAL SAMPLING

iteratively, for unblocked $\mathrm{v} \in \mathcal{V}$

flip coin for v

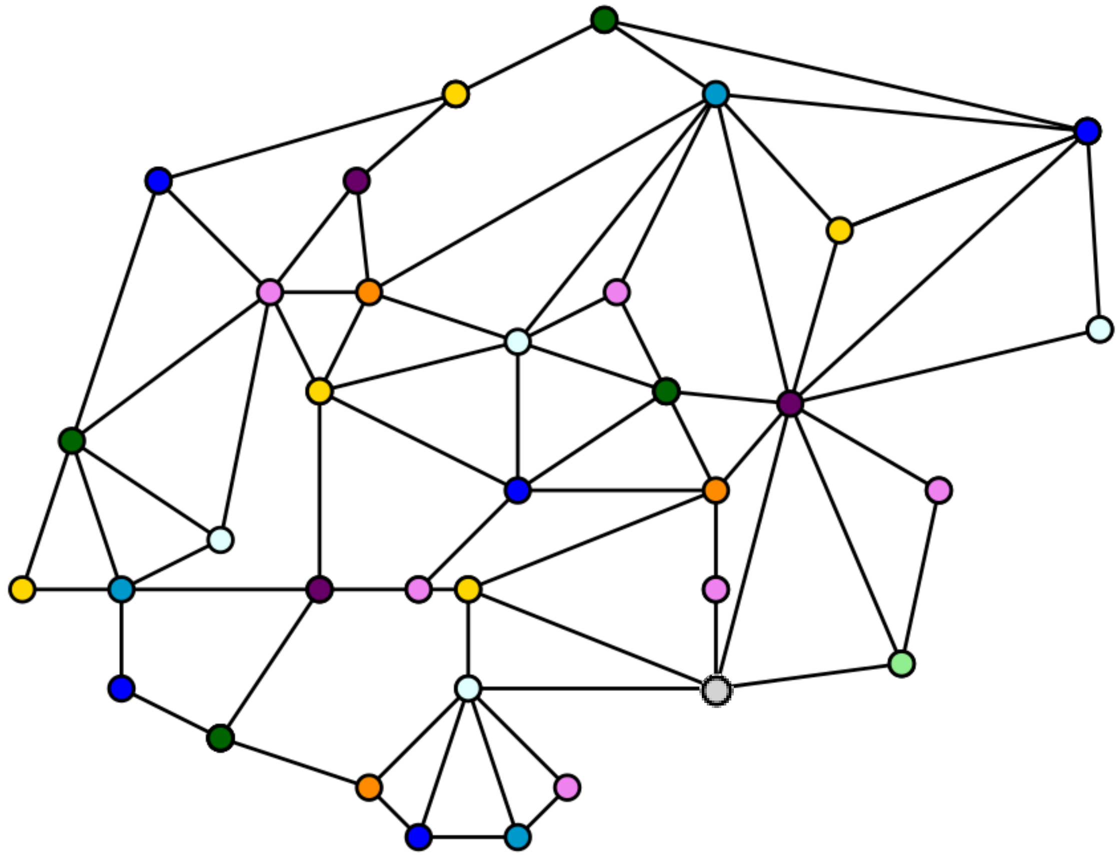
if $\boldsymbol{P[A \mid \text{flipped coins}] > \frac{\sqrt{p}}{2}}$

block $\mathrm{vbl(A)}$

dangerous!

I. PARTIAL SAMPLING in $O(d^2 + \log^* n)$ under $p(e \cdot d)^{4\lambda} < 1$ s.t. remainder
1) satisifes polynomial LLL
2) has "small" components

Inspired by Molloy and Reed's
Sequential Partial Sampling [STOC'98]



OBSERVATIONS:

- $P[A \mid \text{flipped coins}\,] \leq \sqrt{p}$

  satisfies polynomial LLL

- $P[A \text{ dangerous}\,] \leq \frac{p}{\frac{\sqrt{p}}{2}} = 2\sqrt{p}$
- $P[A \text{ remains}] = O(d)P[A \text{ dangerous}] \leq O\left(d\sqrt{p}\right) = \frac{1}{poly\ d}$
- A and B at distance > 2 remain independently

  by Shattering Lemma: small components
  (see, e.g., Barenboim, Elkin, Pettie, Schneider [FOCS'12])

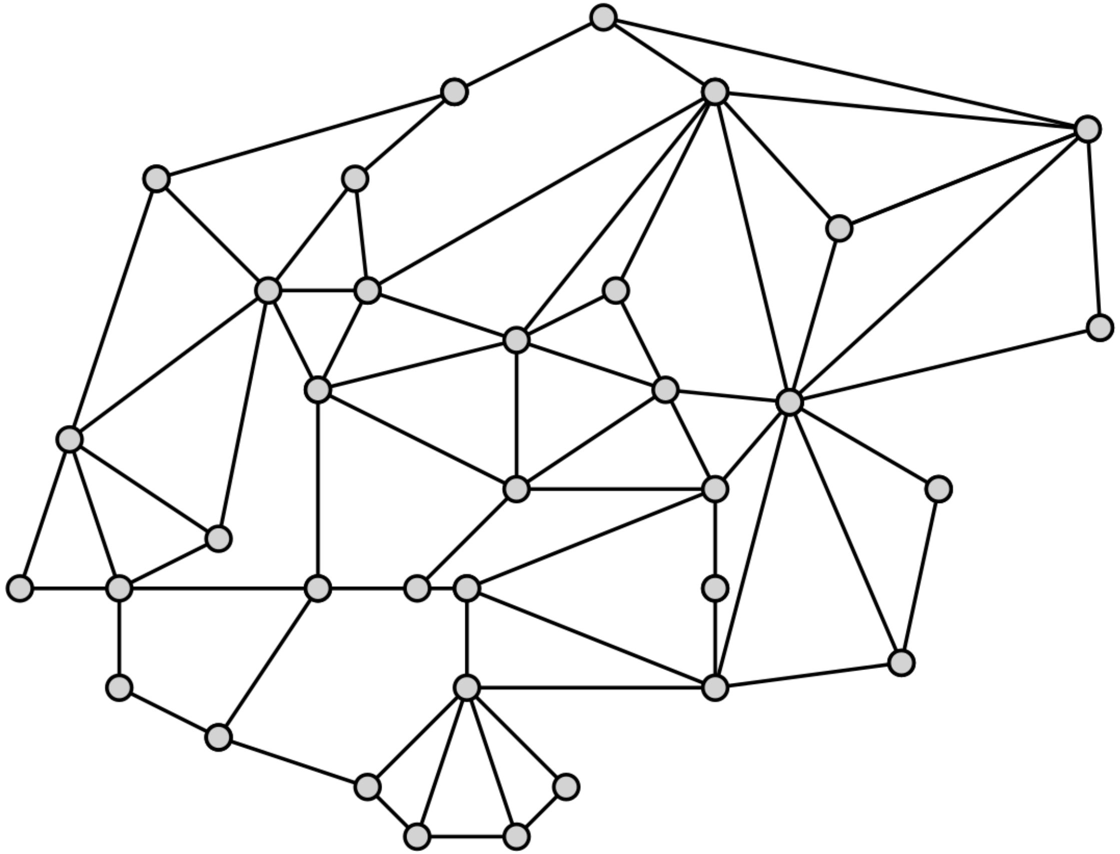
- can be parallelized using a $(\mathrm{d}^2 + 1)$ - coloring of G^2

  consistent with one sequential global order

BASE ALGORITHM

I. PARTIAL SAMPLING

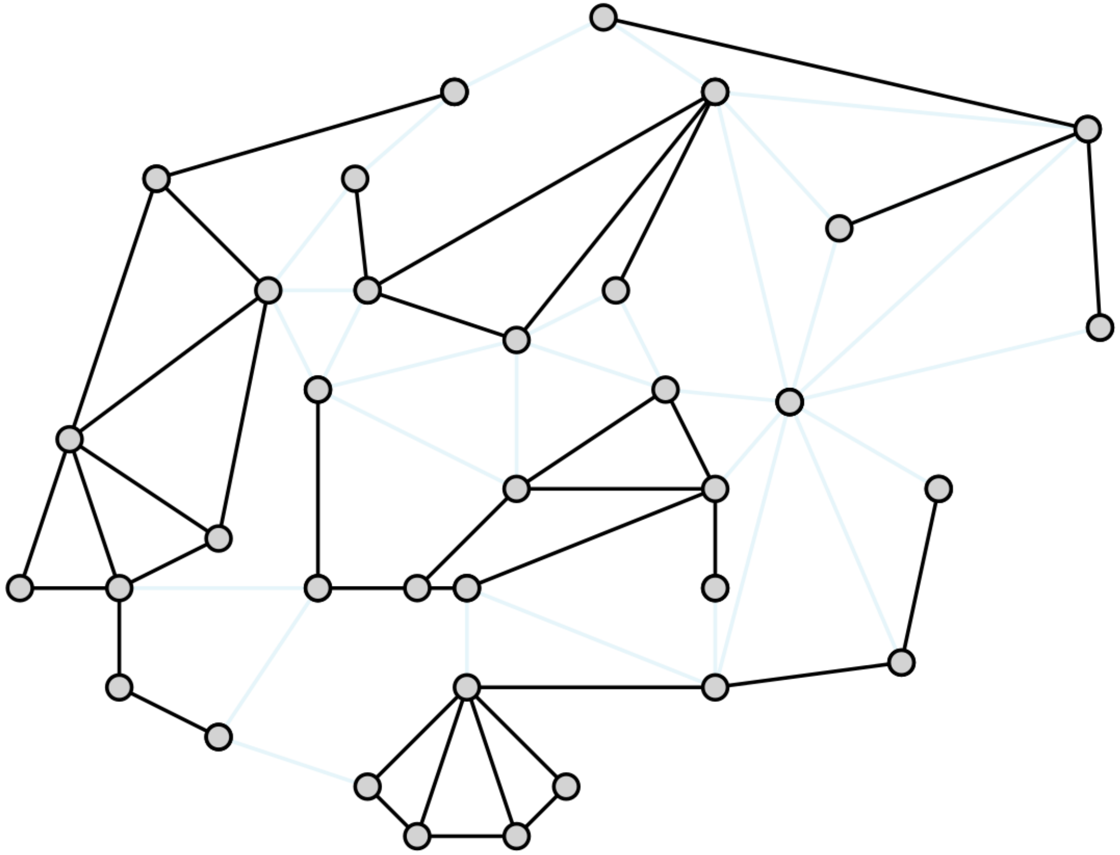
II. DETERMINISTIC LLL ALGORITHM



BASE ALGORITHM

I. PARTIAL SAMPLING

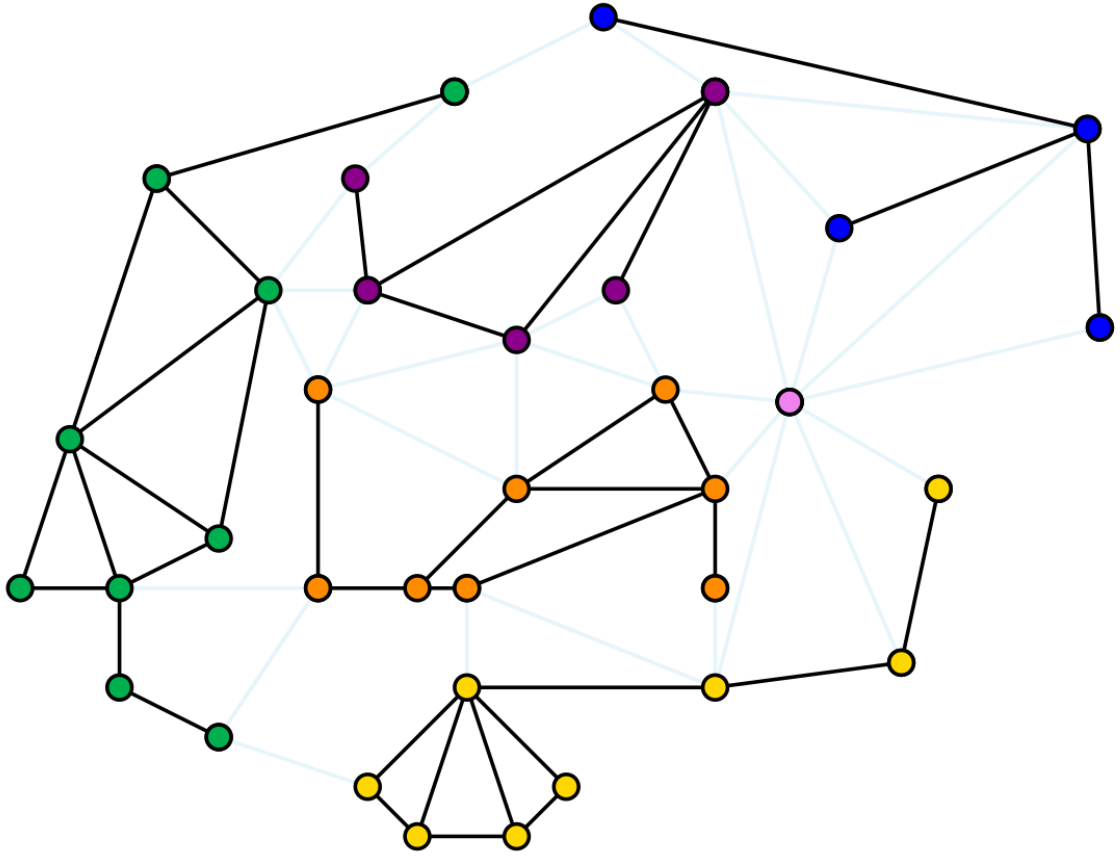
II. DETERMINISTIC LLL ALGORITHM



BASE ALGORITHM

I. PARTIAL SAMPLING

II. DETERMINISTIC LLL ALGORITHM



II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

1. $\left(\boldsymbol{\lambda}, \widetilde{\mathbf{O}}\left(\mathbf{n}^{1/\lambda}\right)\right)$**-network decomposition of** $\mathbf{G^2}$ in $\lambda \cdot \mathrm{n}^{\frac{1}{\lambda}} \cdot 2^{\mathrm{O}(\sqrt{\log \mathrm{n}})}$

combining approaches by *Awerbuch and Peleg* [FOCS'90*], Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. Iterative Assignment

Inductively, $P[\, A \mid \mathcal{V}_{\leq i} \,] \leq p(e \cdot d)^i$

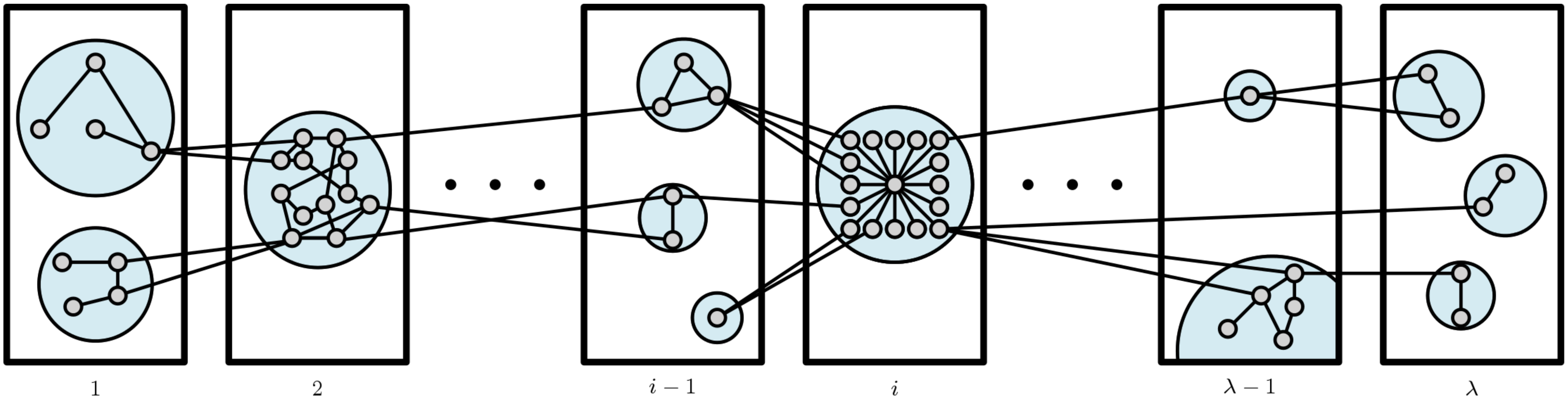
Eventually, $P[\, A \mid \mathcal{V}_{\leq \lambda} \,] = P[\, A \mid \mathcal{V} \,] \leq p(e \cdot d)^{\lambda} < 1$

II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

1. **$\left(\lambda, \tilde{O}\left(n^{1/\lambda}\right)\right)$-network decomposition of G^2** in $\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$
   combining approaches by *Awerbuch and Peleg* [FOCS'90], *Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. **Iterative Assignment**
   Inductively, $P[A \mid \mathcal{V}_{\leq i}] \leq p(e \cdot d)^i$
   Eventually, $P[A \mid \mathcal{V}_{\leq \lambda}] = P[A \mid \mathcal{V}] \leq p(e \cdot d)^{\lambda} < 1$



II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

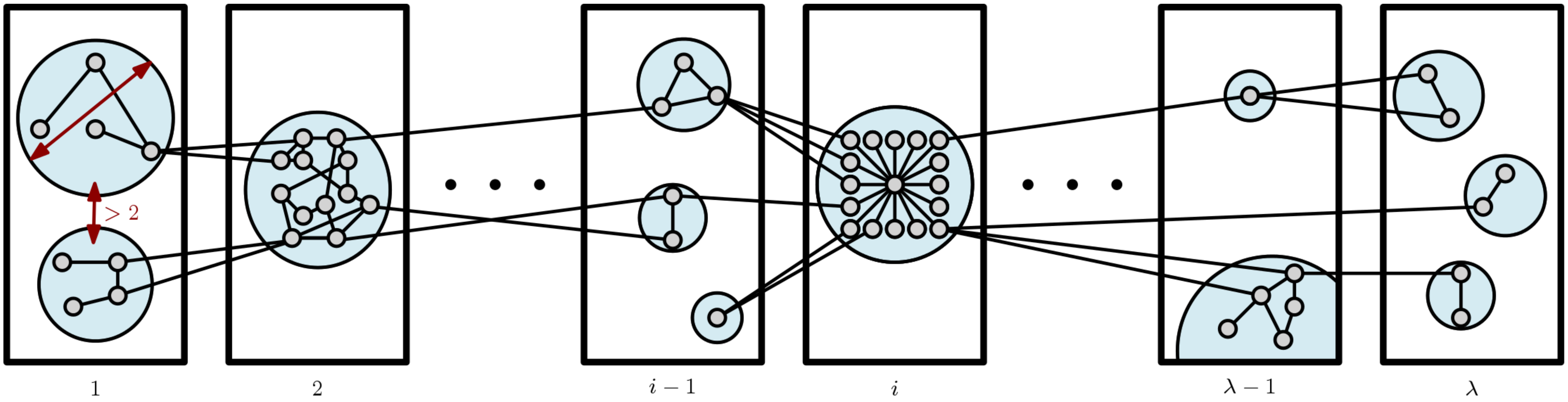
1. $\left(\boldsymbol{\lambda}, \widetilde{\mathbf{O}}\left(\mathbf{n}^{1/\lambda}\right)\right)$**-network decomposition of $\mathbf{G^2}$** in $\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$

combining approaches by *Awerbuch and Peleg* [FOCS'90], *Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. Iterative Assignment

Inductively, $P[\, A \mid \mathcal{V}_{\leq i} \,] \leq p(e \cdot d)^{i}$

Eventually, $P[\, A \mid \mathcal{V}_{\leq \lambda} \,] = P[\, A \mid \mathcal{V} \,] \leq p(e \cdot d)^{\lambda} < 1$



II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

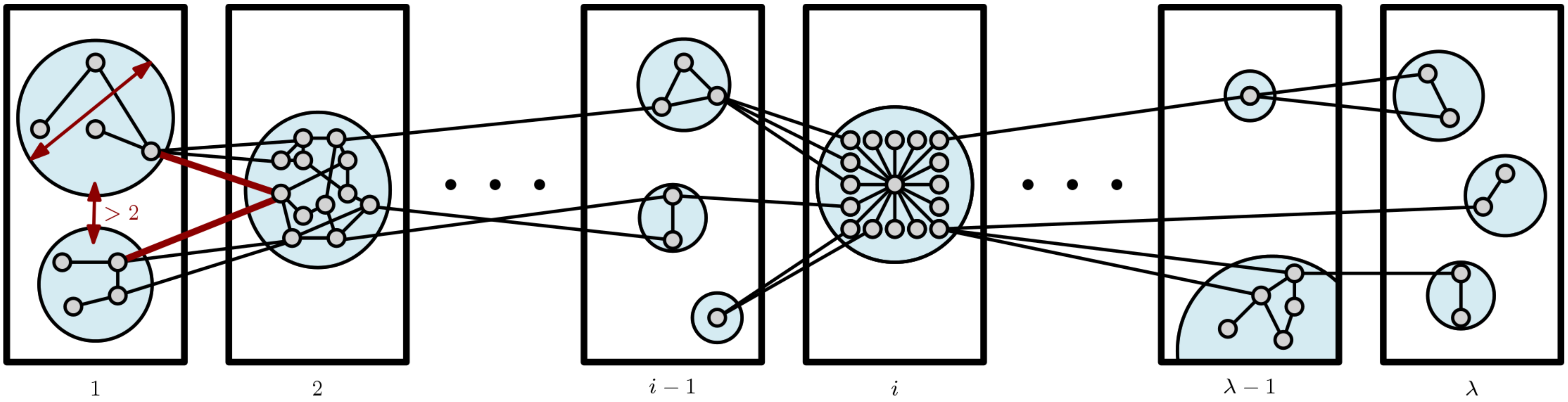
1. $\left(\lambda, \tilde{O}(n^{1/\lambda})\right)$**-network decomposition of** G^2 in $\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$

combining approaches by *Awerbuch and Peleg* [FOCS'90], *Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. Iterative Assignment

Inductively, $P[\, A \mid \mathcal{V}_{\leq i} \,] \leq p(e \cdot d)^i$

Eventually, $P[\, A \mid \mathcal{V}_{\leq \lambda} \,] = P[\, A \mid \mathcal{V} \,] \leq p(e \cdot d)^\lambda < 1$



II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

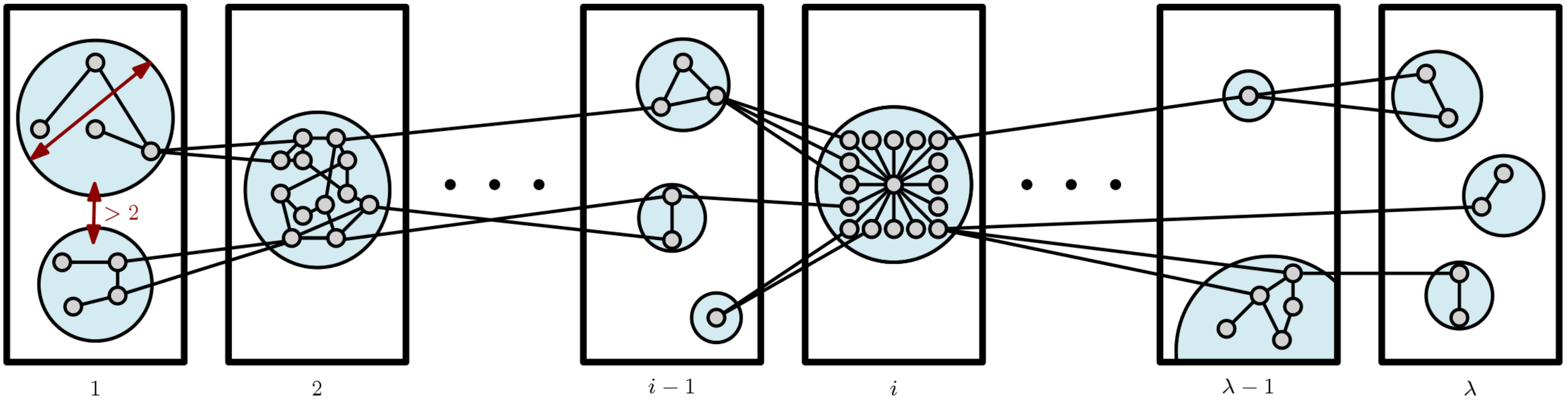
1. $\left(\boldsymbol{\lambda}, \widetilde{\mathbf{O}}\left(\mathbf{n}^{1/\lambda}\right)\right)$**-network decomposition of $\mathbf{G}^2$** in $\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$

combining approaches by *Awerbuch and Peleg* [FOCS'90], *Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. Iterative Assignment

Inductively, $P[A \mid \mathcal{V}_{\leq i}] \leq p(e \cdot d)^i$

Eventually, $P[A \mid \mathcal{V}_{\leq \lambda}] = P[A \mid \mathcal{V}] \leq p(e \cdot d)^\lambda < 1$



II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

1. $\left(\lambda, \tilde{O}(n^{1/\lambda})\right)$**-network decomposition of** G^2 in $\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$

combining approaches by *Awerbuch and Peleg* [FOCS'90], *Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. Iterative Assignment

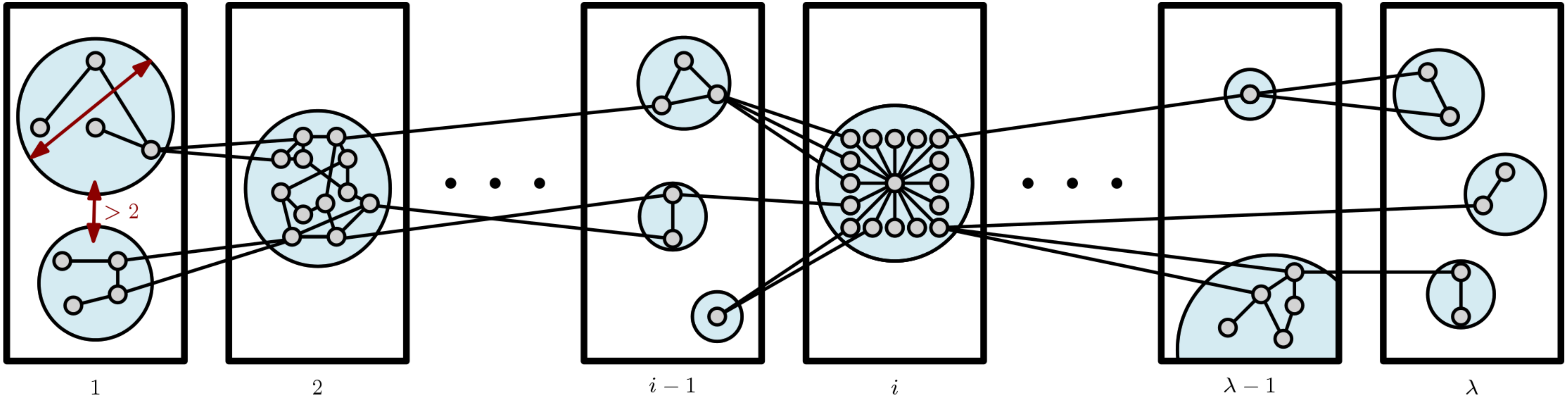
Inductively, $P[A \mid \mathcal{V}_{\le i}] \le p(e \cdot d)^i$

Eventually, $P[A \mid \mathcal{V}_{\le \lambda}] = P[A \mid \mathcal{V}] \le p(e \cdot d)^\lambda < 1$

LLL for step i

event $B_{A,i}$: $P[A \mid \mathcal{V}_{\le i}] > p(e \cdot d)^i$

probability: $\le \frac{1}{e \cdot d}$



II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

1. **$\left(\lambda, \tilde{O}(n^{1/\lambda})\right)$-network decomposition of G^2** in $\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$

   combining approaches by *Awerbuch and Peleg* [FOCS'90], *Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. **Iterative Assignment**

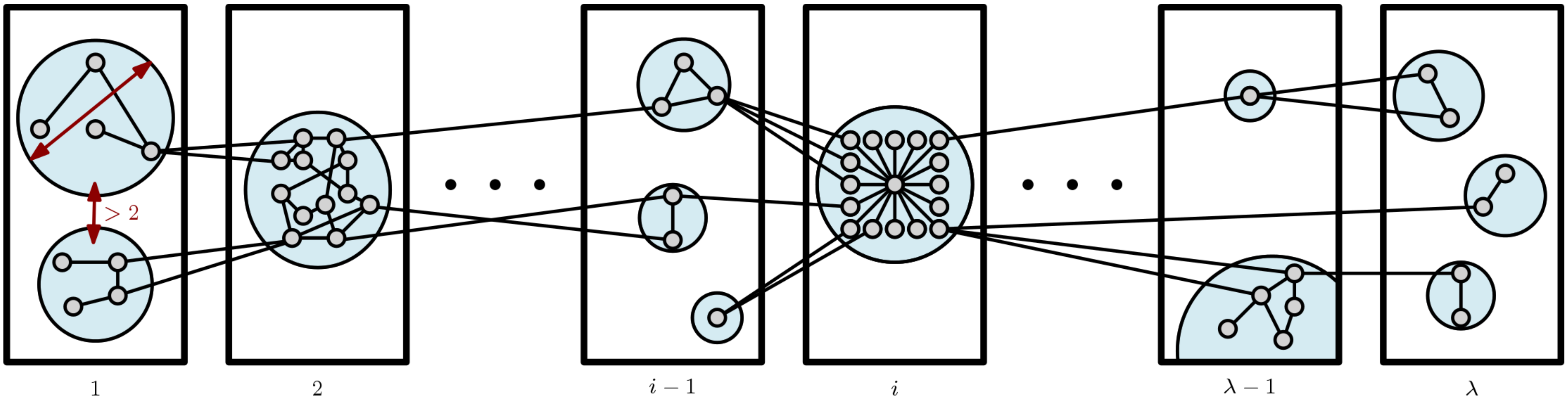
   Inductively, $P[\, A \mid \mathcal{V}_{\leq i}\,] \leq p(e \cdot d)^i$

   Eventually, $P[\, A \mid \mathcal{V}_{\leq \lambda}\,] = P[\, A \mid \mathcal{V}\,] \leq p(e \cdot d)^\lambda < 1$

LLL for step i

event $B_{A,i}$: $P[\, A \mid \mathcal{V}_{\leq i}\,] > p(e \cdot d)^i$

probability: $\leq \frac{1}{e \cdot d}$



II. DETERMINISTIC LLL ALGORITHM

$\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$ rounds under $p(e \cdot d)^{4\lambda} < 1$

1. $\left(\lambda, \tilde{O}(n^{1/\lambda})\right)$**-network decomposition of G^2** in $\lambda \cdot n^{\frac{1}{\lambda}} \cdot 2^{O(\sqrt{\log n})}$

combining approaches by *Awerbuch and Peleg* [FOCS'90], *Panconesi and Srinivasan* [STOC'92], and *Awerbuch, Luby, Goldberg, Plotkin* [FOCS'89]

2. Iterative Assignment

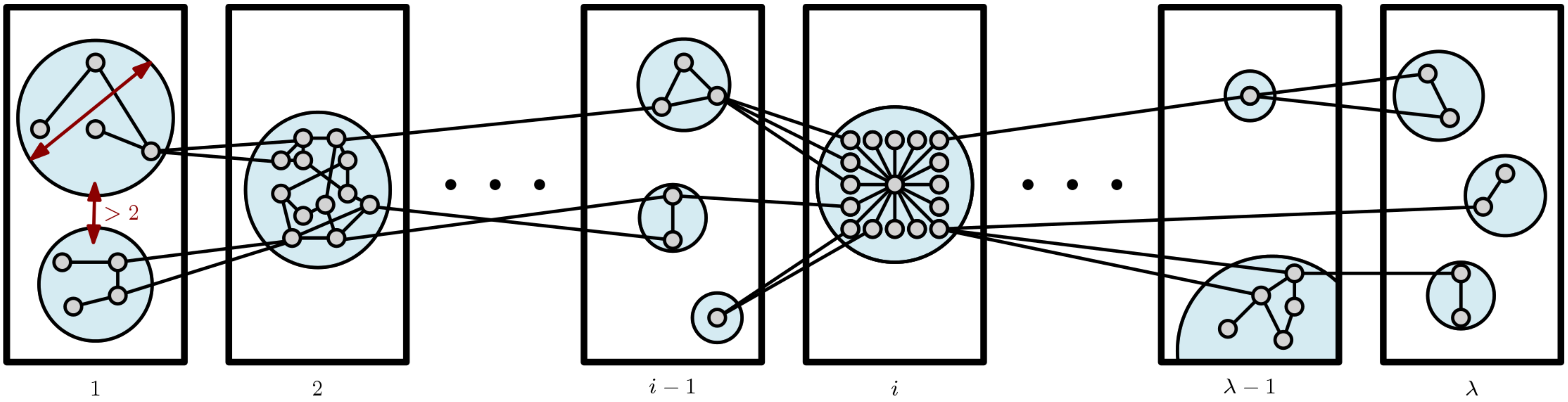
Inductively, $P[\,A \mid \mathcal{V}_{\leq i}\,] \leq p(e \cdot d)^i$

Eventually, $P[\,A \mid \mathcal{V}_{\leq \lambda}\,] = P[\,A \mid \mathcal{V}\,] \leq p(e \cdot d)^\lambda < 1$

LLL for step i satisfies standard LLL criterion

event $B_{A,i}$: $P[\,A \mid \mathcal{V}_{\leq i}\,] > p(e \cdot d)^i$

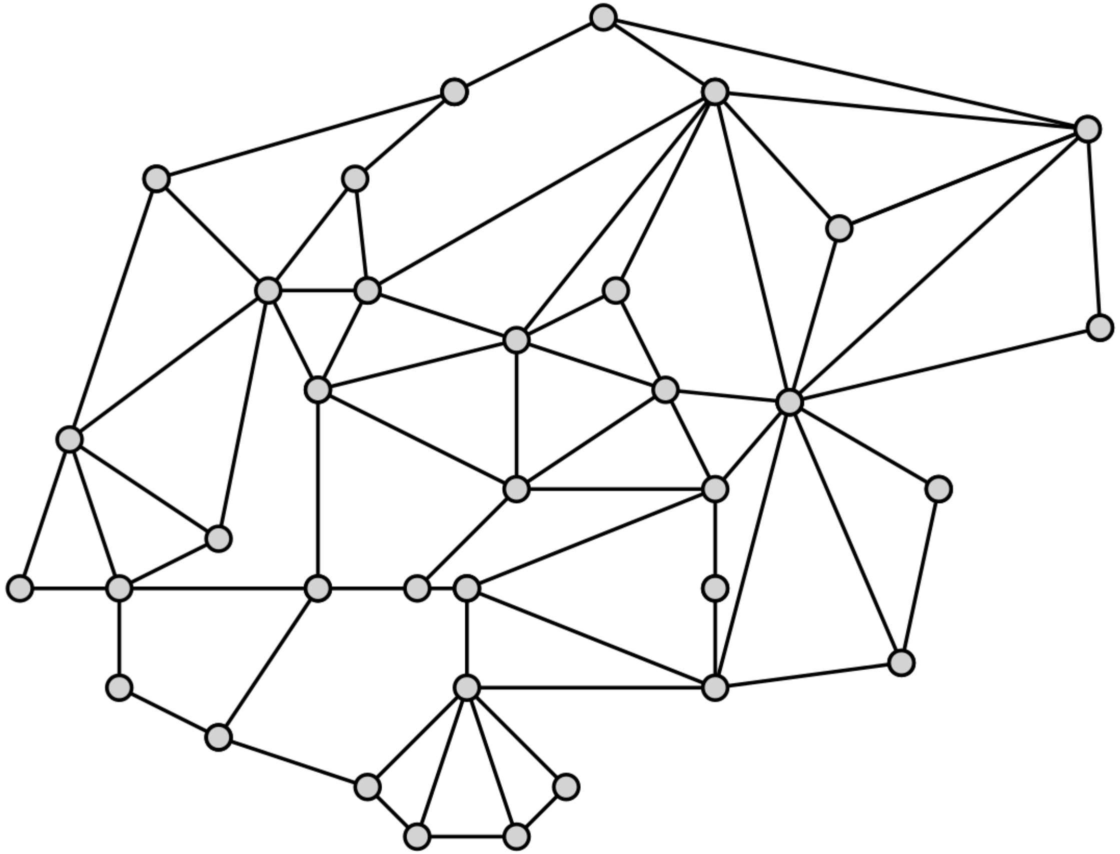
probability: $\leq \frac{1}{e \cdot d}$



BASE ALGORITHM

I. PARTIAL SAMPLING

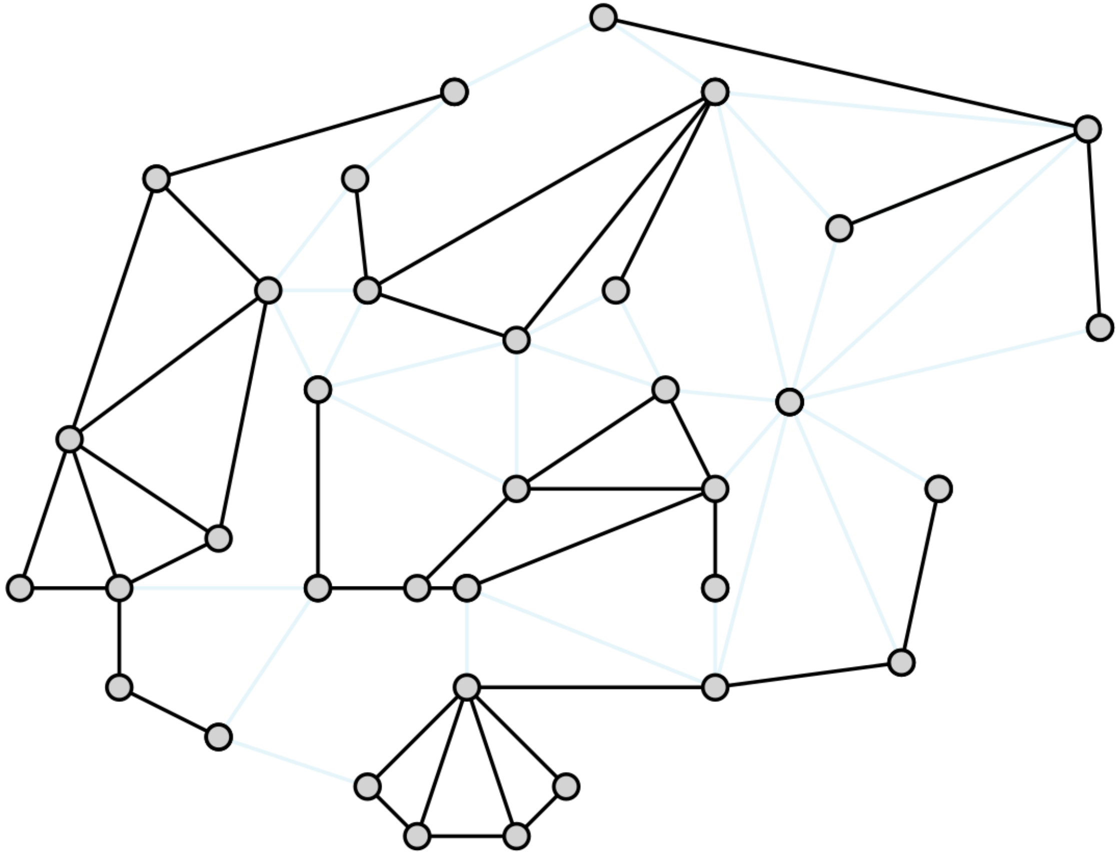
II. DETERMINISTIC LLL ALGORITHM



BASE ALGORITHM

I. PARTIAL SAMPLING

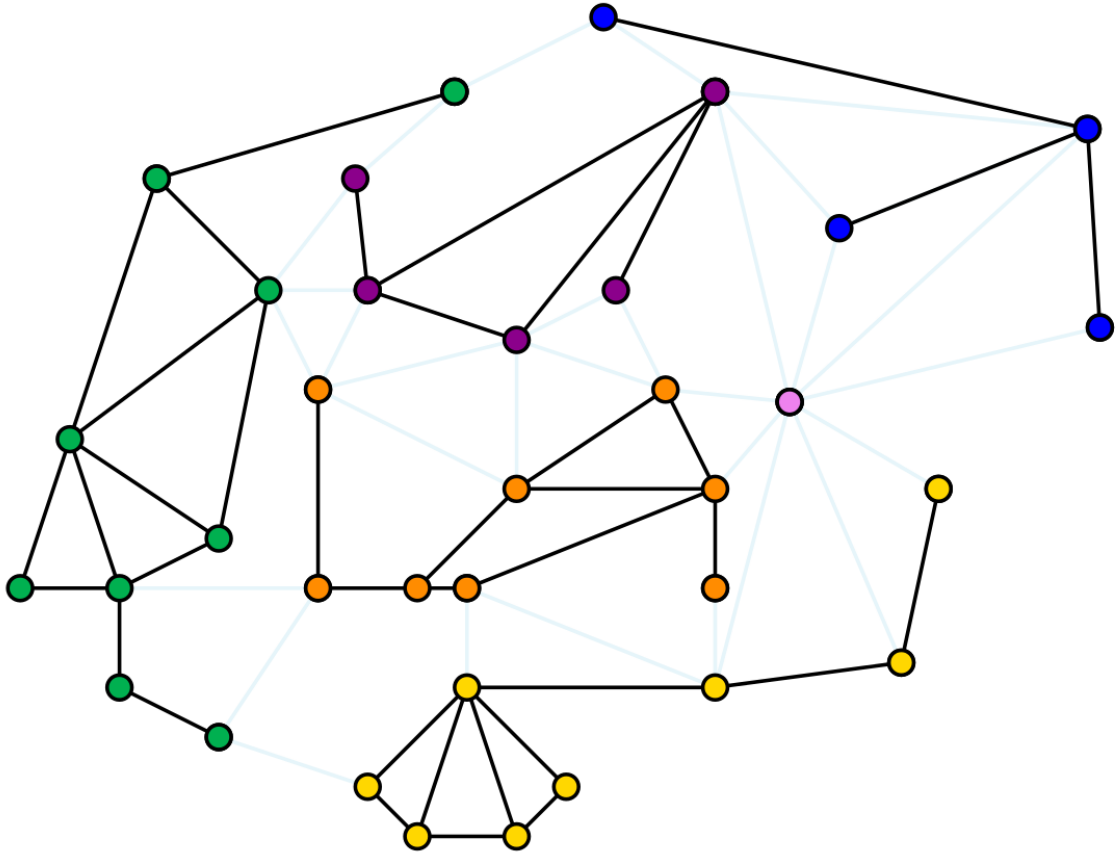
II. DETERMINISTIC LLL ALGORITHM



BASE ALGORITHM

I. PARTIAL SAMPLING

II. DETERMINISTIC LLL ALGORITHM



Summary and Open Problems

BASE ALGORITHM

BOOTSTRAPPING (Speed-Up) →

$$T_{LLL}(n) = 2^{O(\sqrt{\log\log n})}$$

Summary and Open Problems

BASE ALGORITHM

BOOTSTRAPPING (Speed-Up)

$$T_{LLL}(n) = 2^{O(\sqrt{\log\log n})}$$

CONJECTURE:

$$T_{LLL}(n) = O(\log \log n)$$

Chang, Pettie [FOCS'16]

Summary and Open Problems

BASE ALGORITHM

BOOTSTRAPPING
(Speed-Up)

→

$$T_{LLL}(n) = 2^{O(\sqrt{\log\log n})}$$

CONJECTURE:

$$T_{LLL}(n) = O(\text{poly} \log \log n)$$

Chang, Pettie [FOCS'16]

CONJECTURE:

$$T_{LLL}(n) = O(\log \log n)$$

Chang, Pettie [FOCS'16]

Summary and Open Problems

BASE ALGORITHM

BOOTSTRAPPING (Speed-Up)

$$T_{LLL}(n) = 2^{O(\sqrt{\log\log n})}$$

$(\text{poly} \log n, \text{poly} \log n)$ - network decomposition

CONJECTURE:

$$T_{LLL}(n) = O(\text{poly} \log \log n)$$

Chang, Pettie [FOCS'16]

CONJECTURE:

$$T_{LLL}(n) = O(\log \log n)$$

Chang, Pettie [FOCS'16]

Summary and Open Problems

BASE ALGORITHM

BOOTSTRAPPING (Speed-Up)

$$T_{LLL}(n) = 2^{O(\sqrt{\log\log n})}$$

$(\text{poly} \log n, \text{poly} \log n)$ - network decomposition

- **Devise a faster deterministic algorithm.**
- **Devise a faster algorithm under weaker LLL condition.**

CONJECTURE:

$$T_{LLL}(n) = O(\text{poly} \log \log n)$$

Chang, Pettie [FOCS'16]

CONJECTURE:

$$T_{LLL}(n) = O(\log \log n)$$

Chang, Pettie [FOCS'16]

Summary and Open Problems

BASE ALGORITHM

BOOTSTRAPPING (Speed-Up) →

$T_{LLL}(n) = 2^{O(\sqrt{\log \log n})}$

↓

$(\text{poly} \log n, \text{poly} \log n)$ - network decomposition

- **Devise a faster deterministic algorithm.**
- **Devise a faster algorithm under weaker LLL condition.**

CONJECTURE:
$T_{LLL}(n) = O(\text{poly} \log \log n)$
Chang, Pettie [FOCS'16]

CONJECTURE:
$T_{LLL}(n) = O(\log \log n)$
Chang, Pettie [FOCS'16]

Chang, Pettie [FOCS'17]
Completeness of Lovász Local Lemma for Sublogarithmic Bounded-Degree LCL Problems