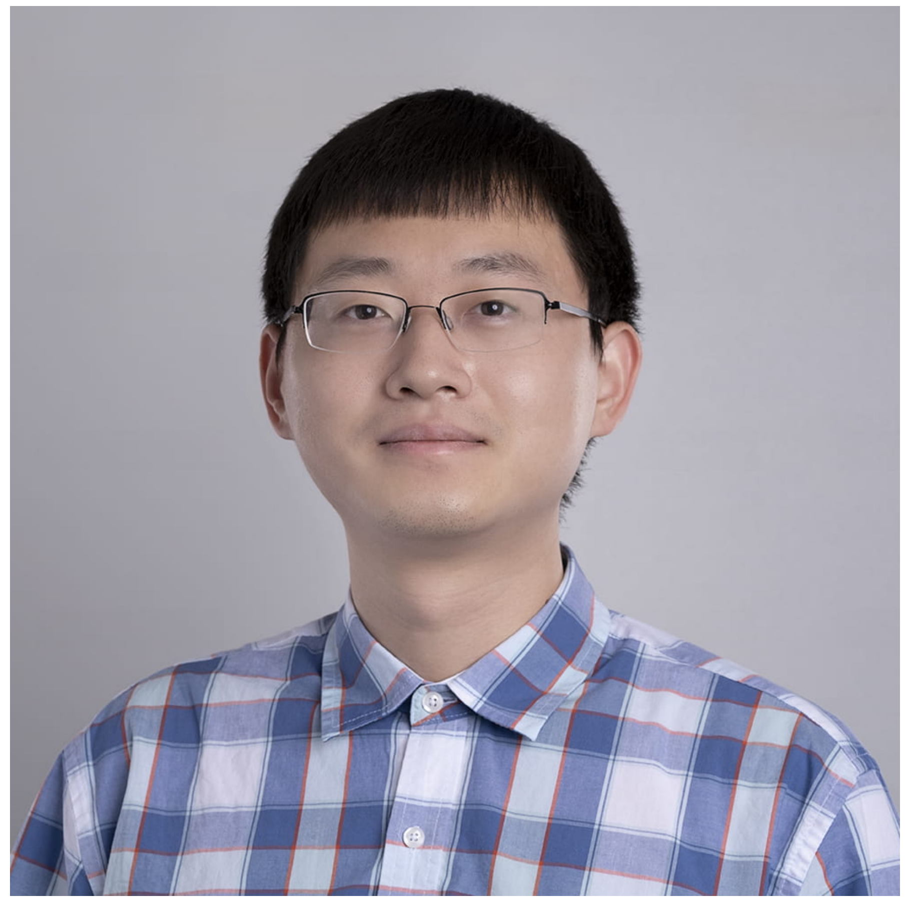
Counting Random k-SAT near the Satisfiability Threshold

Chunyang Wang (Nanjing University)

Joint work with:



Zongchen Chen
(Georgia Tech)

Aditya Lonkar
(Georgia Tech)

Kuan Yang
(SJTU)

Yitong Yin
(Nanjing)

Constraint Satisfaction Problem

$$\Phi = (V, \boldsymbol{Q}, \mathscr{C})$$

Variables: $V = \{v_1, v_2, \ldots, v_n\}$ with finite domains Q_v for each $v \in V$

Constraints: $\mathscr{C} = \{c_1, c_2, \ldots, c_m\}$ with each $c \in \mathscr{C}$ defined on $\mathsf{vbl}(c) \subseteq V$

$$c : \bigotimes_{v \in \mathsf{vbl}(c)} Q_v \to \{\text{True}, \text{False}\}$$

CSP solution: assignment $\boldsymbol{X} \in \bigotimes_{v \in V} Q_v$ s.t. all constraints evaluate to True

Constraint Satisfaction Problem

$$\Phi = (V, \boldsymbol{Q}, \mathscr{C})$$

Variables: $V = \{v_1, v_2, \ldots, v_n\}$ with finite domains Q_v for each $v \in V$

Constraints: $\mathscr{C} = \{c_1, c_2, \ldots, c_m\}$ with each $c \in \mathscr{C}$ defined on $\mathsf{vbl}(c) \subseteq V$

$$c : \bigotimes_{v \in \mathsf{vbl}(c)} Q_v \to \{\text{True}, \text{False}\}$$

CSP solution: assignment $X \in \bigotimes_{v \in V} Q_v$ s.t. all constraints evaluate to True

Random CSP: constraints generated randomly
with a fixed density $\alpha = m/n$

Constraint Satisfaction Problem

$$\Phi = (V, \boldsymbol{Q}, \mathscr{C})$$

Variables: $V = \{v_1, v_2, \ldots, v_n\}$ with finite domains Q_v for each $v \in V$

Constraints: $\mathscr{C} = \{c_1, c_2, \ldots, c_m\}$ with each $c \in \mathscr{C}$ defined on $\mathsf{vbl}(c) \subseteq V$

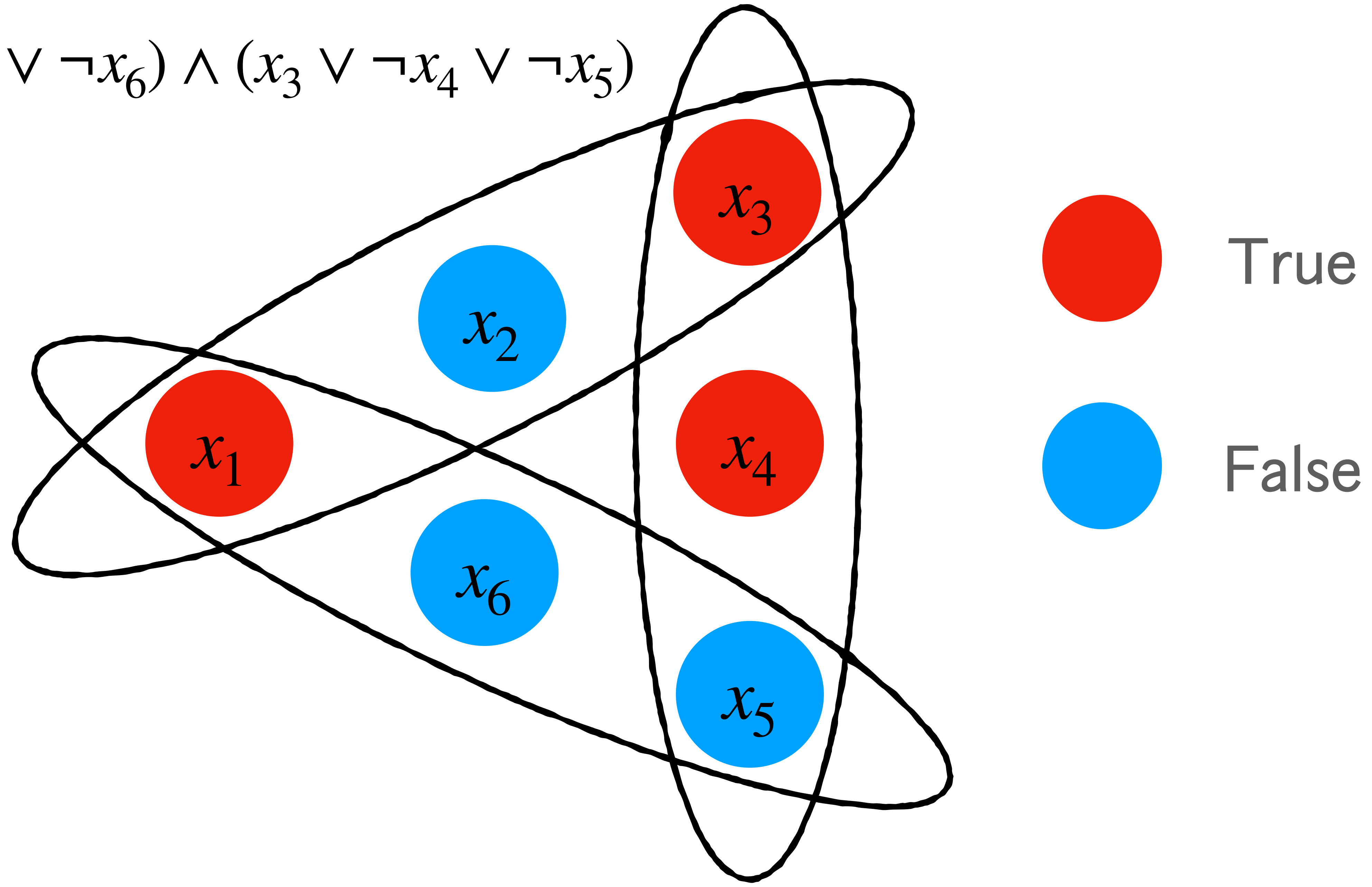
$$c : \bigotimes_{v \in \mathsf{vbl}(c)} Q_v \to \{\text{True}, \text{False}\}$$

CSP solution: assignment $X \in \bigotimes_{v \in V} Q_v$ s.t. all constraints evaluate to True

Random CSP: constraints generated randomly
with a fixed density $\alpha = m/n$

In statistical physics: dilute mean-field spin glasses

$$\Phi = (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_5 \vee \neg x_6) \wedge (x_3 \vee \neg x_4 \vee \neg x_5)$$

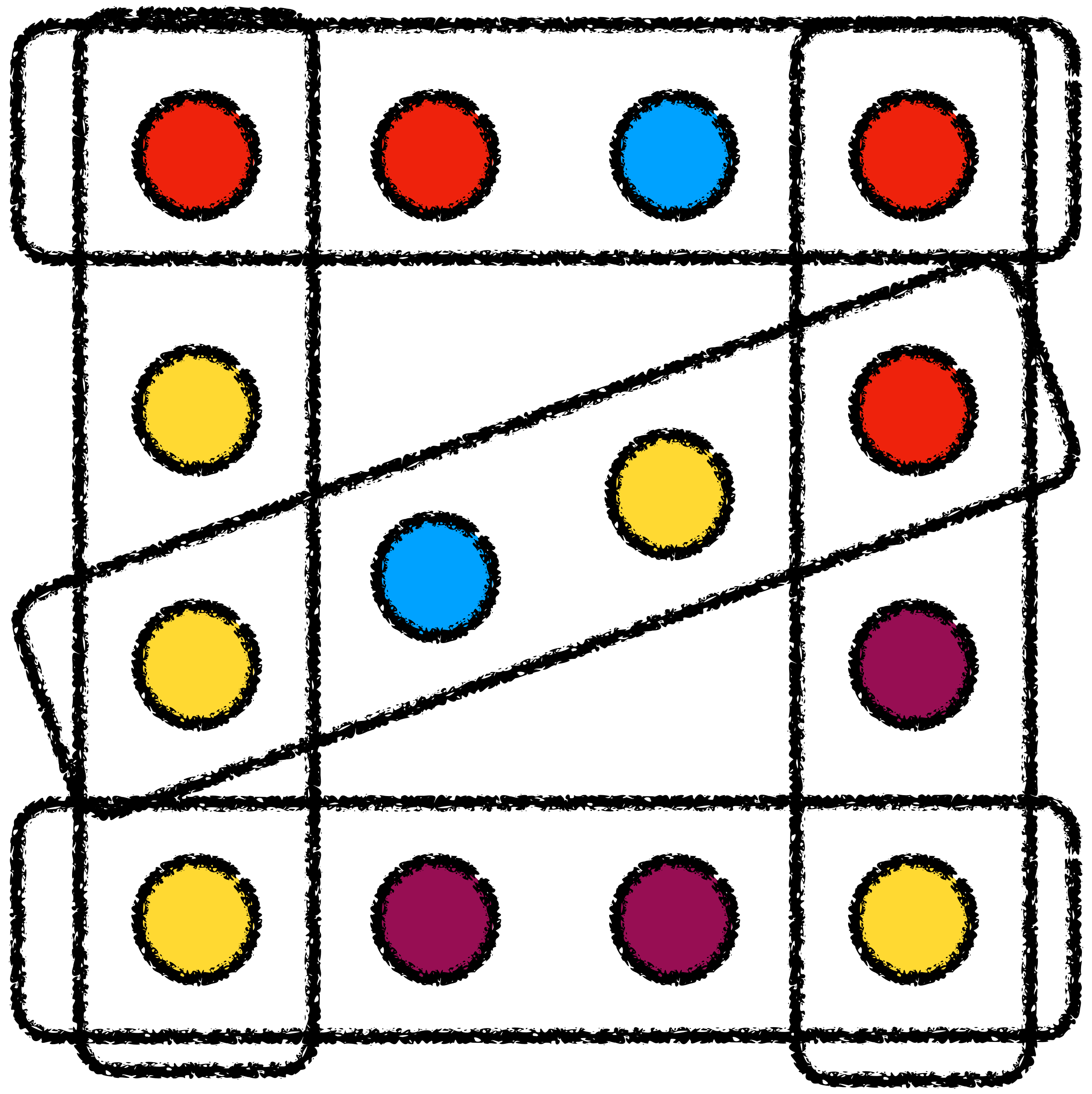


Example: hypergraph q-coloring

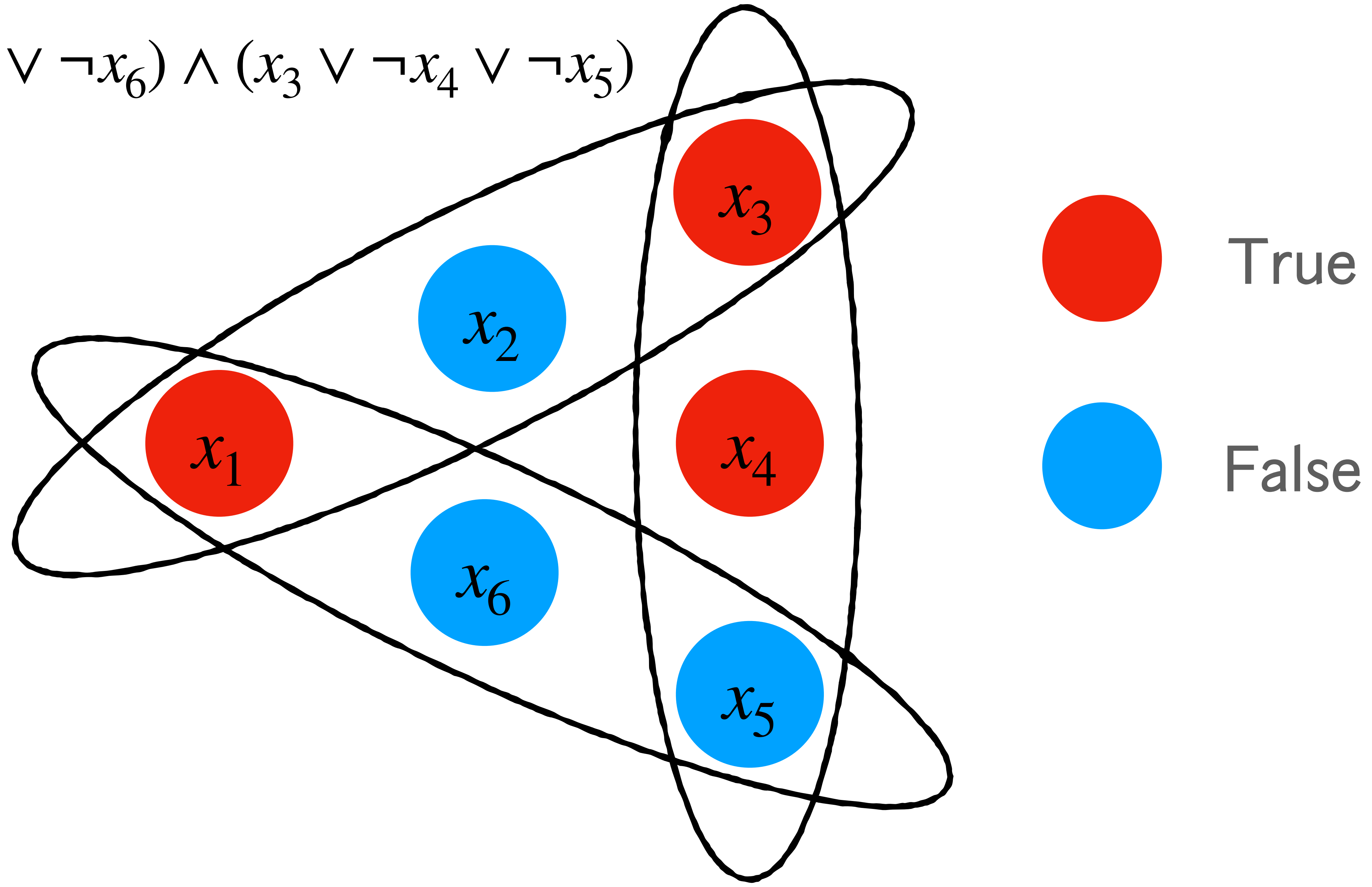
k-uniform hypergraph $H = (V, \mathscr{E})$

color set $[q]$ for each $v \in V$

Solution: an assignment such that no hyperedge (constraint) is monochromatic



$$\Phi = (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_5 \vee \neg x_6) \wedge (x_3 \vee \neg x_4 \vee \neg x_5)$$

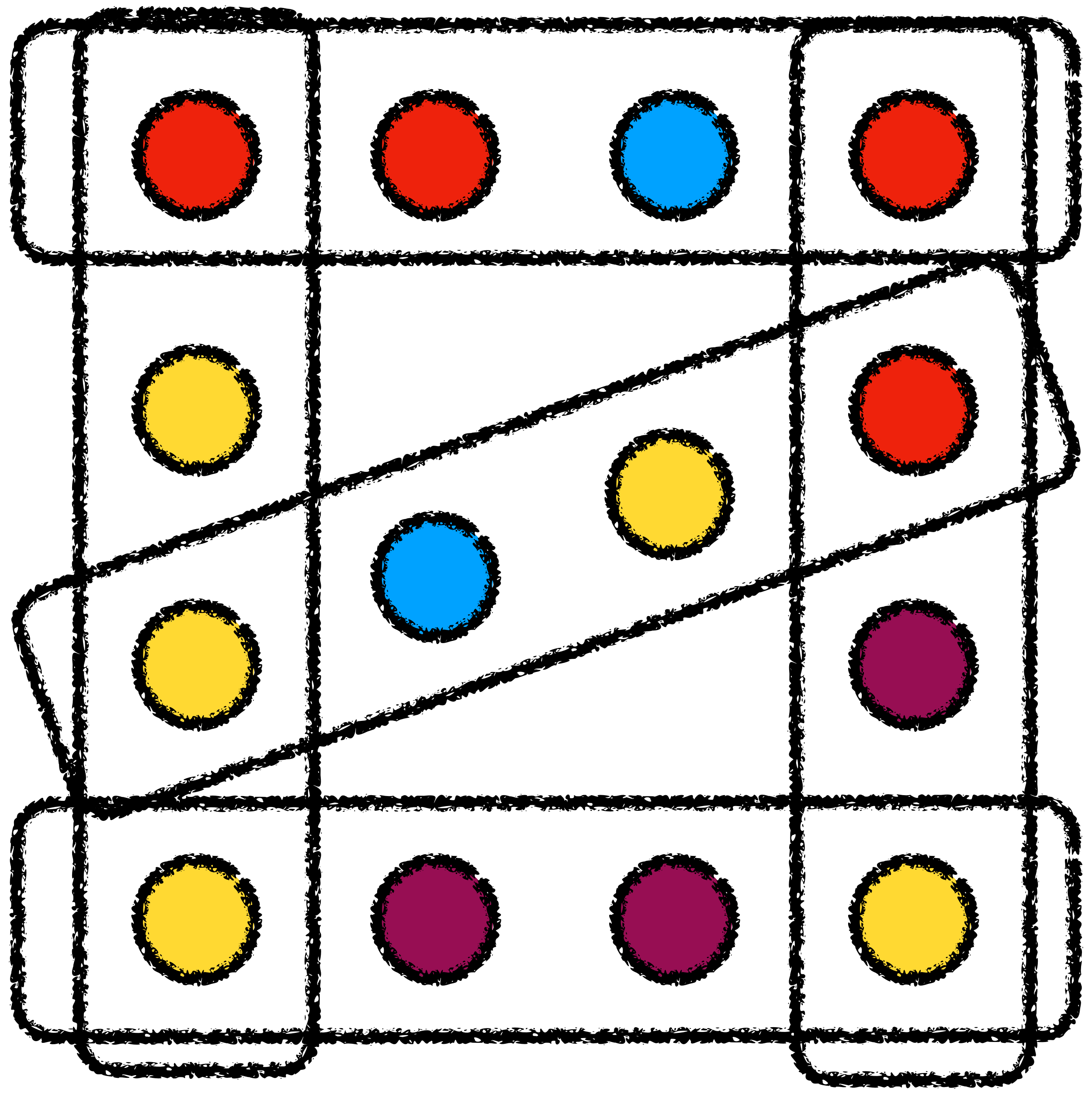


Example: random hypergraph q-coloring

Erdös-Rényi hypergraph $H(k, n, \lfloor \alpha n \rfloor)$

color set $[q]$ for each $v \in V$

Solution: an assignment such that no hyperedge (constraint) is monochromatic



The Random k-SAT

$\Phi(k, n, m = \lfloor \alpha n \rfloor)$: n variables, $m = \lfloor \alpha n \rfloor$ random clauses of size k.

Central question: how the random k-SAT behaves as α changes?

The Random k-SAT

$\Phi(k, n, m = \lfloor \alpha n \rfloor)$: n variables, $m = \lfloor \alpha n \rfloor$ random clauses of size k.

Central question: how the random k-SAT behaves as α changes?

Algorithmic aspects

- Satisfiability: when does a solution exist w.h.p?
- Algorithmic: ~ find a solution efficiently found w.h.p?
- Sampling/Counting: ~ sample/count the solutions efficiently w.h.p?

The Random k-SAT

$\Phi(k, n, m = \lfloor \alpha n \rfloor)$: n variables, $m = \lfloor \alpha n \rfloor$ random clauses of size k.

Central question: how the random k-SAT behaves as α changes?

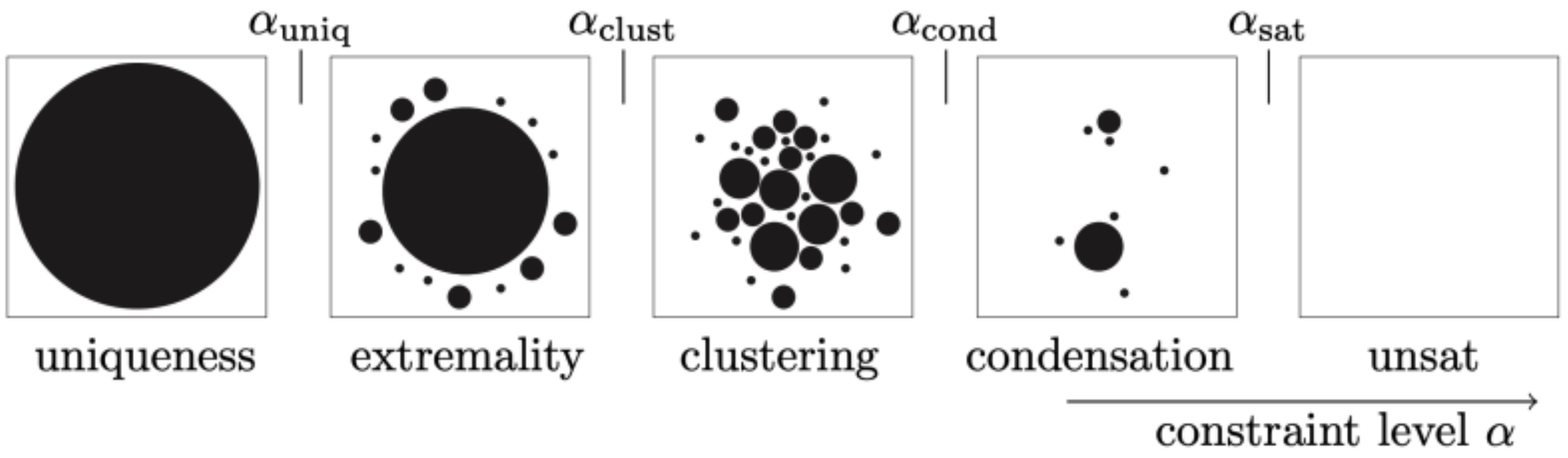
Algorithmic aspects

- Satisfiability: when does a solution exist w.h.p?
- Algorithmic: ~ find a solution efficiently found w.h.p?
- Sampling/Counting: ~ sample/count the solutions efficiently w.h.p?

Solution space geometry

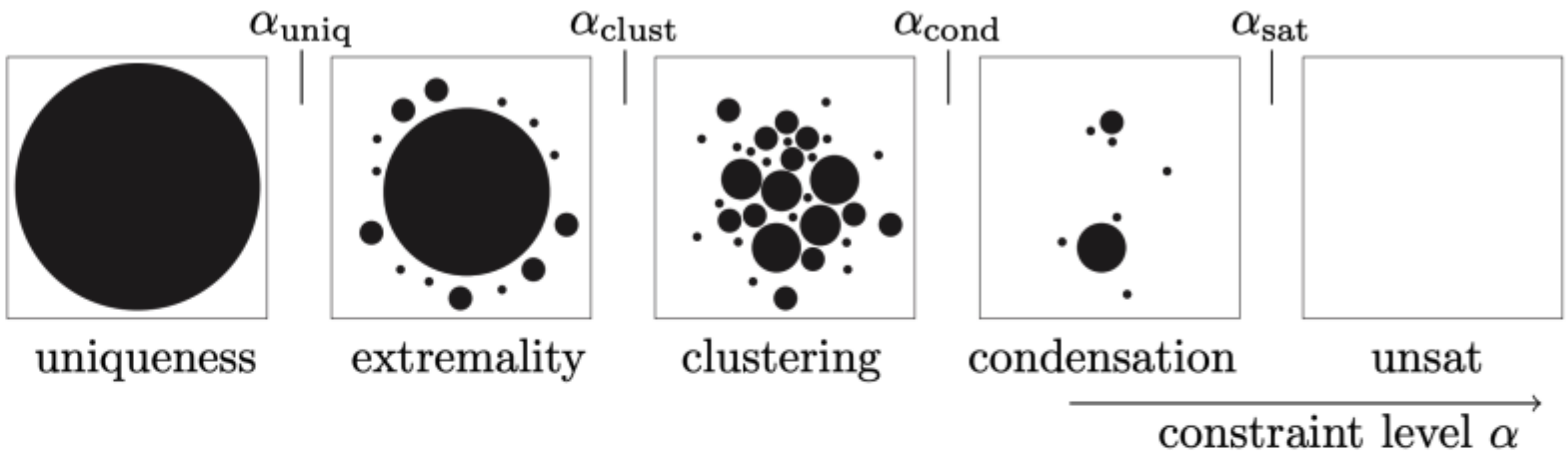
- Connectivity: How do the solution clusters behave?
- Correlation: Do long-range correlations exist?

Solution Space Geometry



Heuristic graph from [Ding, Sly, Sun, Ann. Math. 2022]

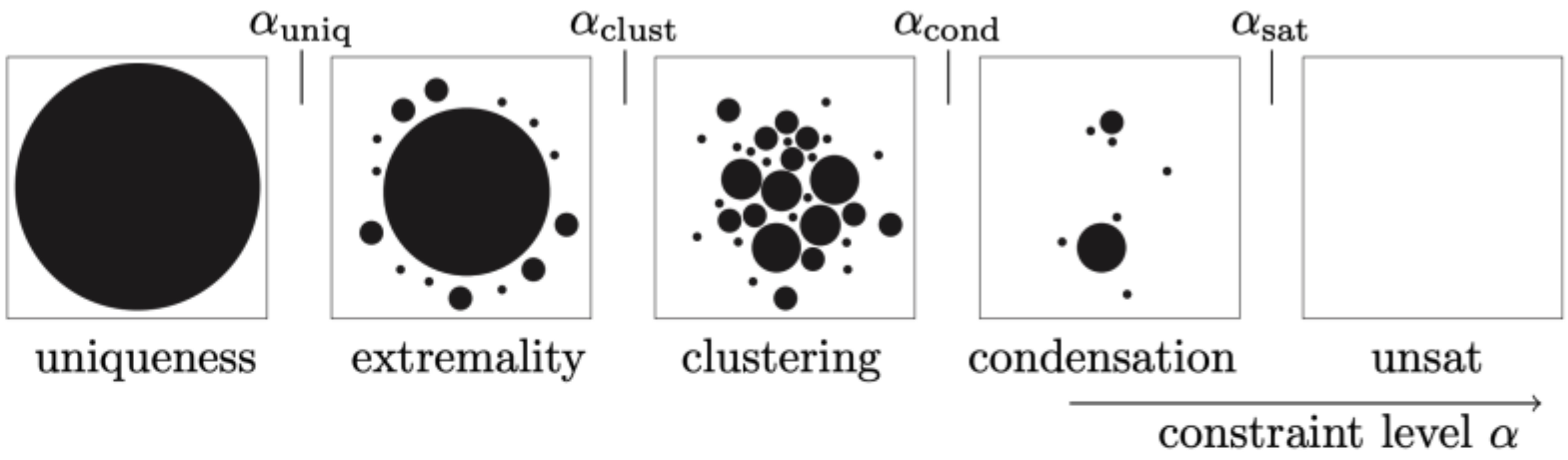
Solution Space Geometry



Heuristic graph from [Ding, Sly, Sun, Ann. Math. 2022]

[Ding, Sly, Sun' 22]: $\alpha_{\text{sat}} = 2^k \ln 2 - (1 + \ln 2)/2 + o_k(1)$

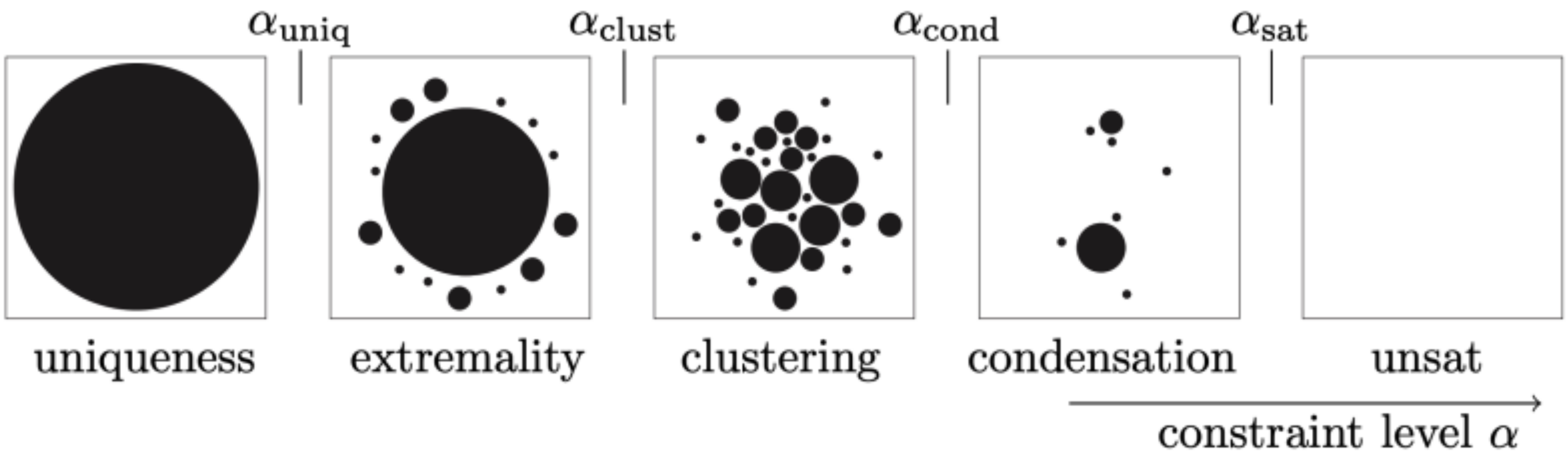
Solution Space Geometry



Heuristic graph from [Ding, Sly, Sun, Ann. Math. 2022]

[Ding, Sly, Sun' 22]: $\alpha_{\text{sat}} = 2^k \ln 2 - (1 + \ln 2)/2 + o_k(1)$ **The satisfiability threshold!**

Solution Space Geometry

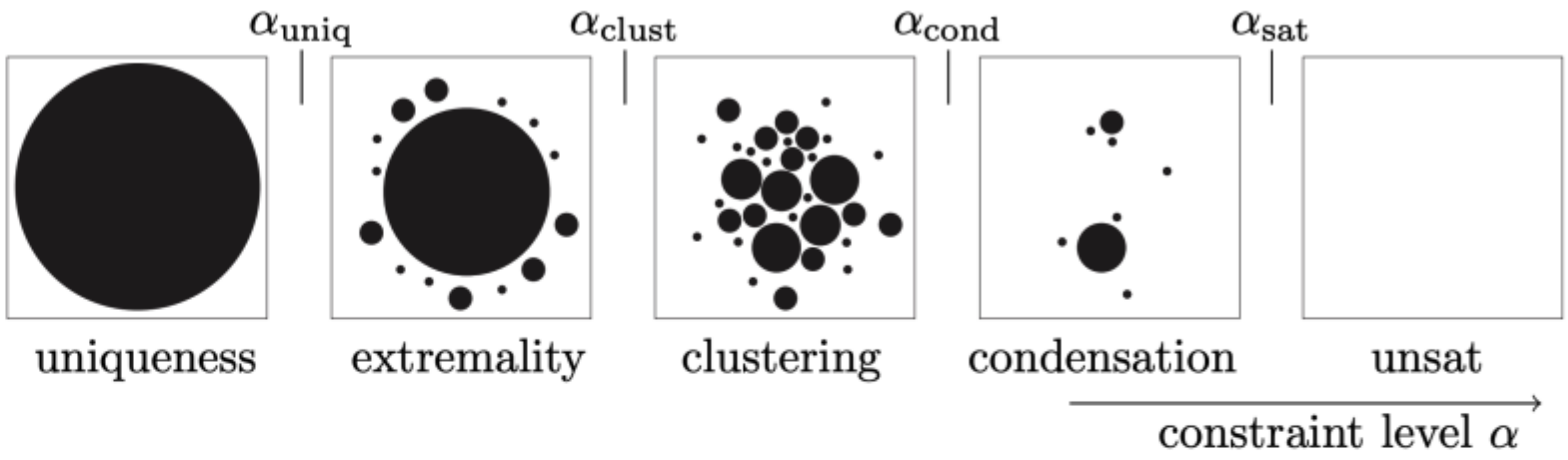


Heuristic graph from [Ding, Sly, Sun, Ann. Math. 2022]

[Ding, Sly, Sun' 22]: $\alpha_{\text{sat}} = 2^k \ln 2 - (1 + \ln 2)/2 + o_k(1)$ **The satisfiability threshold!**

[Achlioptas, Coja-Oghlan' 08]: $\alpha_{\text{cluster}} \approx 2^k (\ln k)/k$

Solution Space Geometry



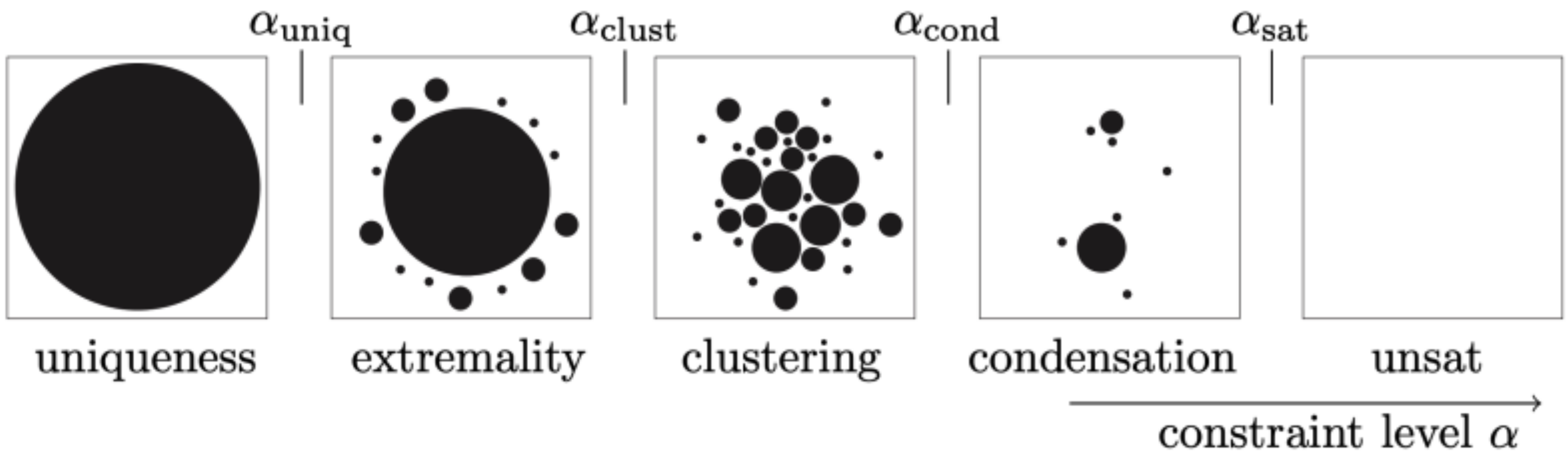
Heuristic graph from [Ding, Sly, Sun, Ann. Math. 2022]

[Ding, Sly, Sun' 22]: $\alpha_{\text{sat}} = 2^k \ln 2 - (1 + \ln 2)/2 + o_k(1)$ **The satisfiability threshold!**

[Achlioptas, Coja-Oghlan' 08]: $\alpha_{\text{cluster}} \approx 2^k (\ln k)/k$

[Coja-Oghlan' 10]: efficient search algorithm when $\alpha < (1 - o_k(1)) 2^k (\ln k)/k$

Solution Space Geometry



Heuristic graph from [Ding, Sly, Sun, Ann. Math. 2022]

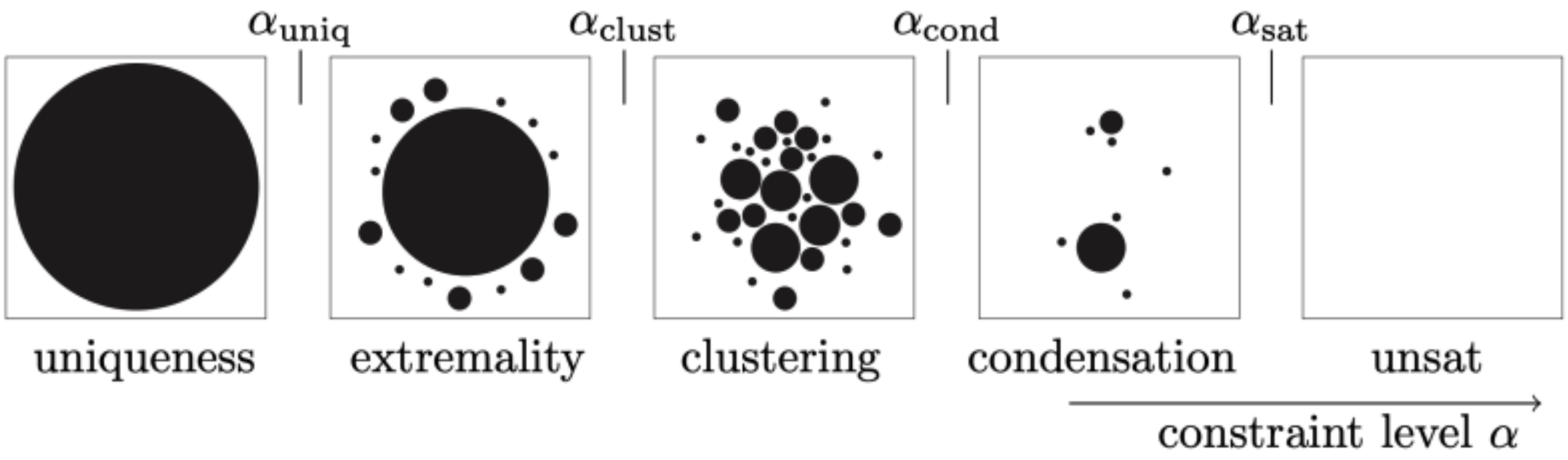
[Ding, Sly, Sun' 22]: $\alpha_{\text{sat}} = 2^k \ln 2 - (1 + \ln 2)/2 + o_k(1)$ **The satisfiability threshold!**

[Achlioptas, Coja-Oghlan' 08]: $\alpha_{\text{cluster}} \approx 2^k (\ln k)/k$

[Coja-Oghlan' 10]: efficient search algorithm when $\alpha < (1 - o_k(1))2^k (\ln k)/k$

[Bresler, Huang' 22]: low degree polynomial algorithms fail when $\alpha < 4.91(2^k \ln k/k)$

Solution Space Geometry



Heuristic graph from [Ding, Sly, Sun, Ann. Math. 2022]

[Ding, Sly, Sun' 22]: $\alpha_{\text{sat}} = 2^k \ln 2 - (1 + \ln 2)/2 + o_k(1)$ **The satisfiability threshold!**

[Achlioptas, Coja-Oghlan' 08]: $\alpha_{\text{cluster}} \approx 2^k (\ln k)/k$

[Coja-Oghlan' 10]: efficient search algorithm when $\alpha < (1 - o_k(1))2^k (\ln k)/k$

[Bresler, Huang' 22]: low degree polynomial algorithms fail when $\alpha < 4.91(2^k \ln k/k)$

The algorithmic threshold?

The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

[Galanis, Goldberg, Guo, Yang' 21]: $\alpha \lesssim 2^{k/300}$, FPTAS

The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

[Galanis, Goldberg, Guo, Yang' 21]: $\alpha \lesssim 2^{k/300}$, FPTAS

[Chen, Galanis, Goldberg, Guo, Herrera-Poyatos, Mani, Moitra '24]: $\alpha \lesssim 2^{0.039k}$, fast sampler

The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

[Galanis, Goldberg, Guo, Yang' 21]: $\alpha \lesssim 2^{k/300}$, FPTAS

[Chen, Galanis, Goldberg, Guo, Herrera-Poyatos, Mani, Moitra '24]: $\alpha \lesssim 2^{0.039k}$, fast sampler

[He, Wu, Yang '23]: $\alpha \lesssim 2^{k/3}$, fast sampler

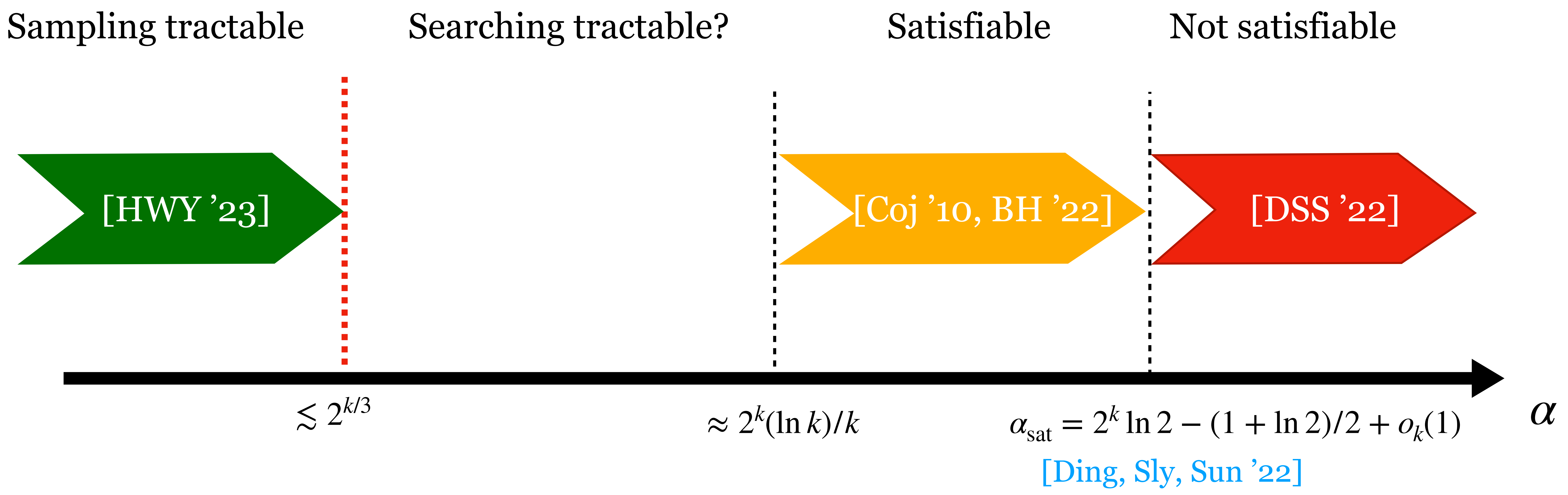
The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

[Galanis, Goldberg, Guo, Yang' 21]: $\alpha \lesssim 2^{k/300}$, FPTAS

[Chen, Galanis, Goldberg, Guo, Herrera-Poyatos, Mani, Moitra '24]: $\alpha \lesssim 2^{0.039k}$, fast sampler

[He, Wu, Yang '23]: $\alpha \lesssim 2^{k/3}$, fast sampler



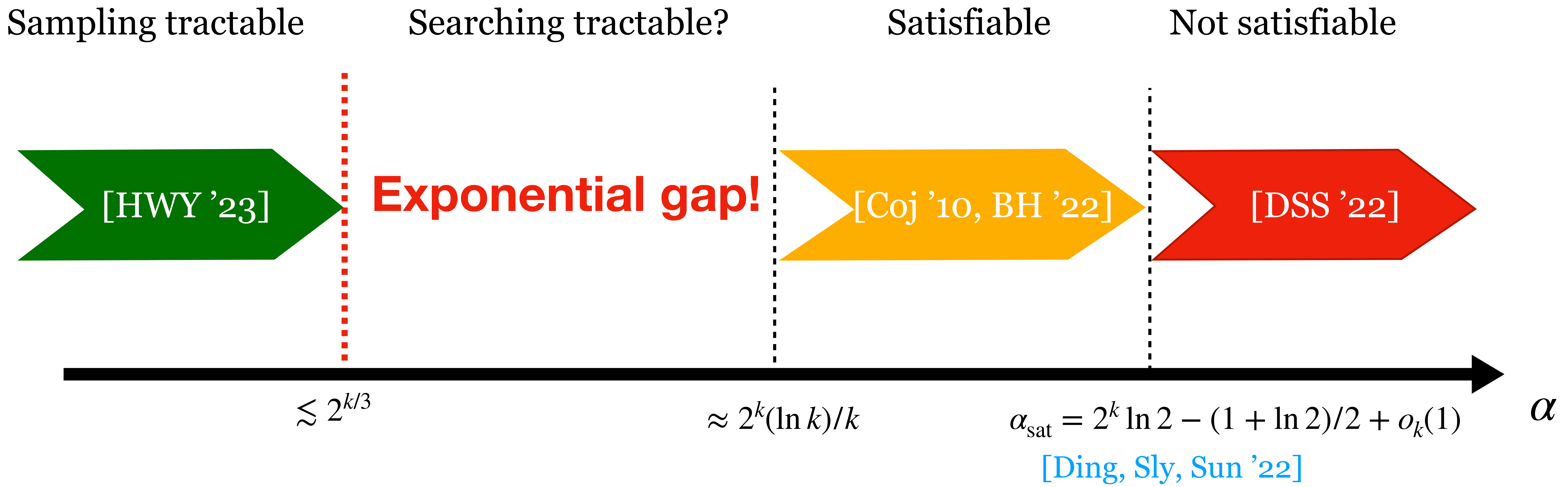
The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

[Galanis, Goldberg, Guo, Yang' 21]: $\alpha \lesssim 2^{k/300}$, FPTAS

[Chen, Galanis, Goldberg, Guo, Herrera-Poyatos, Mani, Moitra '24]: $\alpha \lesssim 2^{0.039k}$, fast sampler

[He, Wu, Yang '23]: $\alpha \lesssim 2^{k/3}$, fast sampler



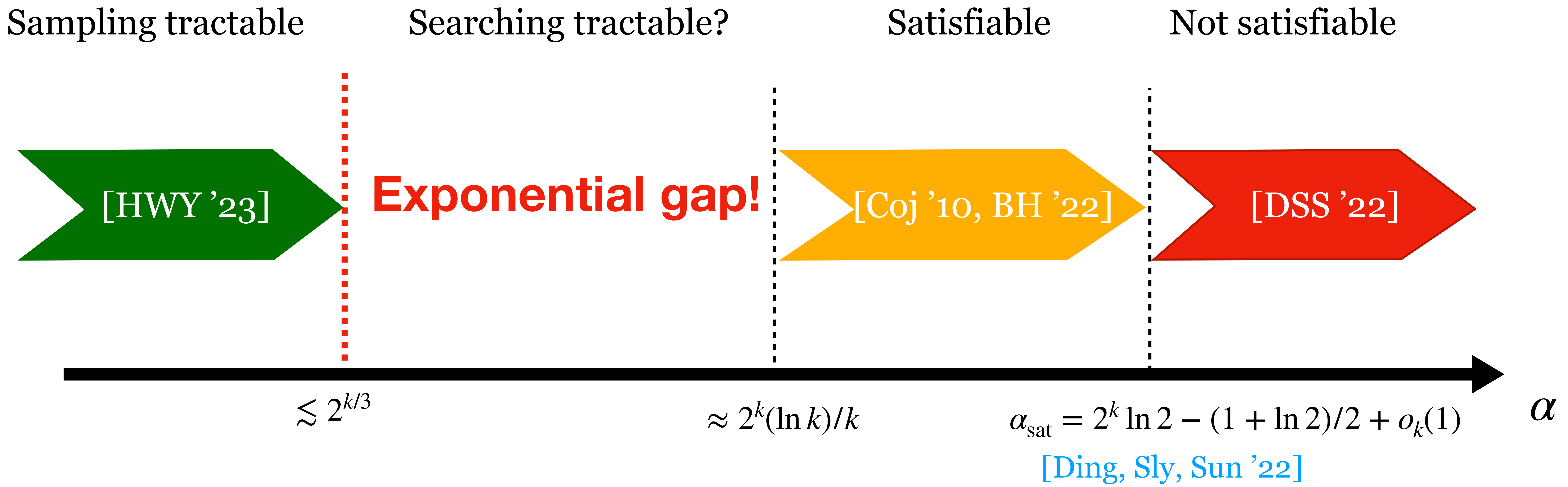
The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

[Galanis, Goldberg, Guo, Yang' 21]: $\alpha \lesssim 2^{k/300}$, FPTAS

[Chen, Galanis, Goldberg, Guo, Herrera-Poyatos, Mani, Moitra '24]: $\alpha \lesssim 2^{0.039k}$, fast sampler

[He, Wu, Yang '23]: $\alpha \lesssim 2^{k/3}$, fast sampler



Is counting/sampling tractable up to the algorithmic threshold?

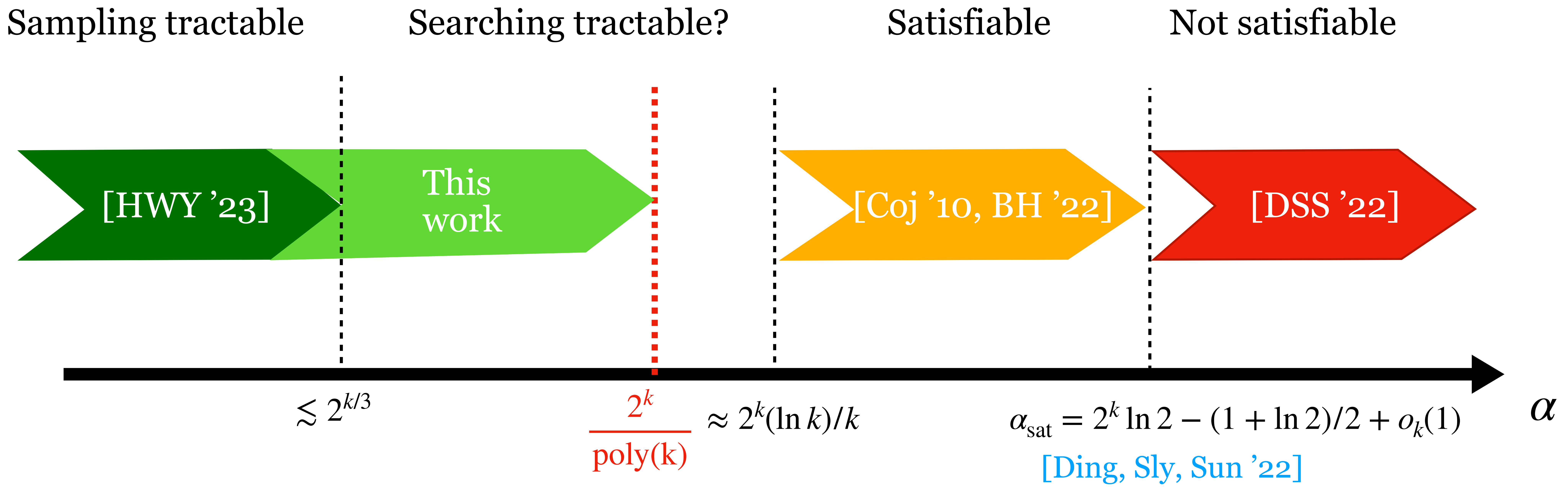
The Sampling/Counting Threshold

: α that we can efficiently sample/count solutions to $\Phi(k, n, \lfloor \alpha n \rfloor)$

[Galanis, Goldberg, Guo, Yang' 21]: $\alpha \lesssim 2^{k/300}$, FPTAS

[Chen, Galanis, Goldberg, Guo, Herrera-Poyatos, Mani, Moitra '24]: $\alpha \lesssim 2^{0.039k}$, fast sampler

[He, Wu, Yang '23]: $\alpha \lesssim 2^{k/3}$, fast sampler



Is counting/sampling tractable up to the algorithmic threshold?

Main Result

Sampling/Counting Random k-SAT near the Satisfiability Threshold

The exists a universal constant $c \geq 1$ such that if

$$0 < \alpha \leq \frac{2^k}{k^c},$$

Then the following exists w.h.p. over the choice of a random k-SAT formula $\Phi = \Phi(k, n, \lfloor \alpha n \rfloor)$.

- **Sampling algorithm**:
  draw an assignment ε-close to a uniform solution of Φ within time $(n/\varepsilon)^{\mathrm{poly}(k,\alpha)}$
- **Deterministic Counting algorithm**:
  ε-estimates the number of solutions of Φ within time $(n/\varepsilon)^{\mathrm{poly}(k,\alpha)}$.

Bounded-Degree k-SAT

random k-SAT with density $\alpha \Longrightarrow$ average degree $k\alpha$

We can compare it to k-SAT with maximum degree $d = k\alpha$

Bounded-Degree k-SAT

random k-SAT with density $\alpha \implies$ average degree $k\alpha$

We can compare it to k-SAT with maximum degree $d = k\alpha$

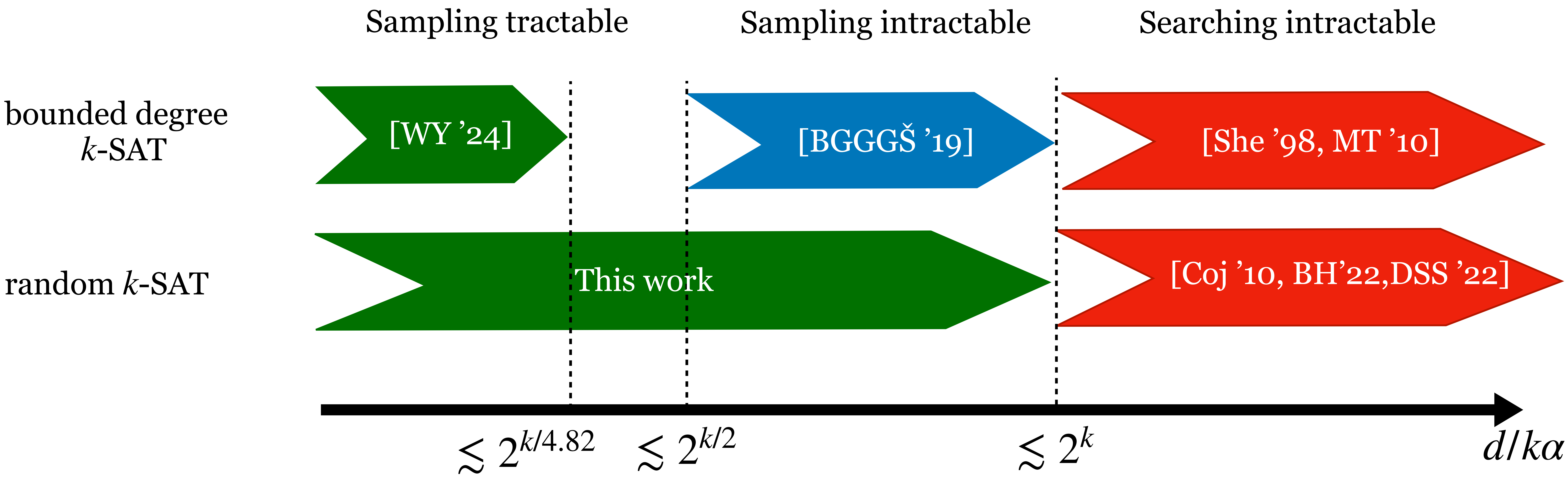
[Bezáková, Galanis, Goldberg, Guo, Štefankovič '19]: NP-hard when $d \gtrsim 2^{k/2}$!

Bounded-Degree k-SAT

random k-SAT with density $\alpha \implies$ average degree $k\alpha$

We can compare it to k-SAT with maximum degree $d = k\alpha$

[Bezáková, Galanis, Goldberg, Guo, Štefankovič '19]: NP-hard when $d \gtrsim 2^{k/2}$!

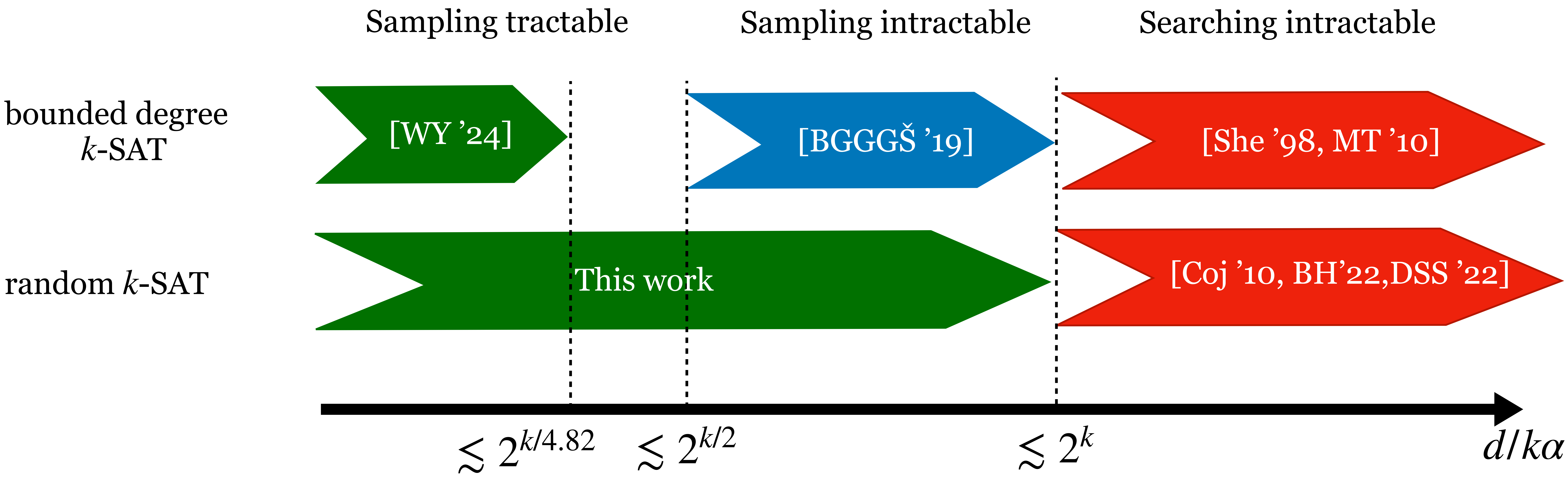


Bounded-Degree k-SAT

random k-SAT with density $\alpha \implies$ average degree $k\alpha$

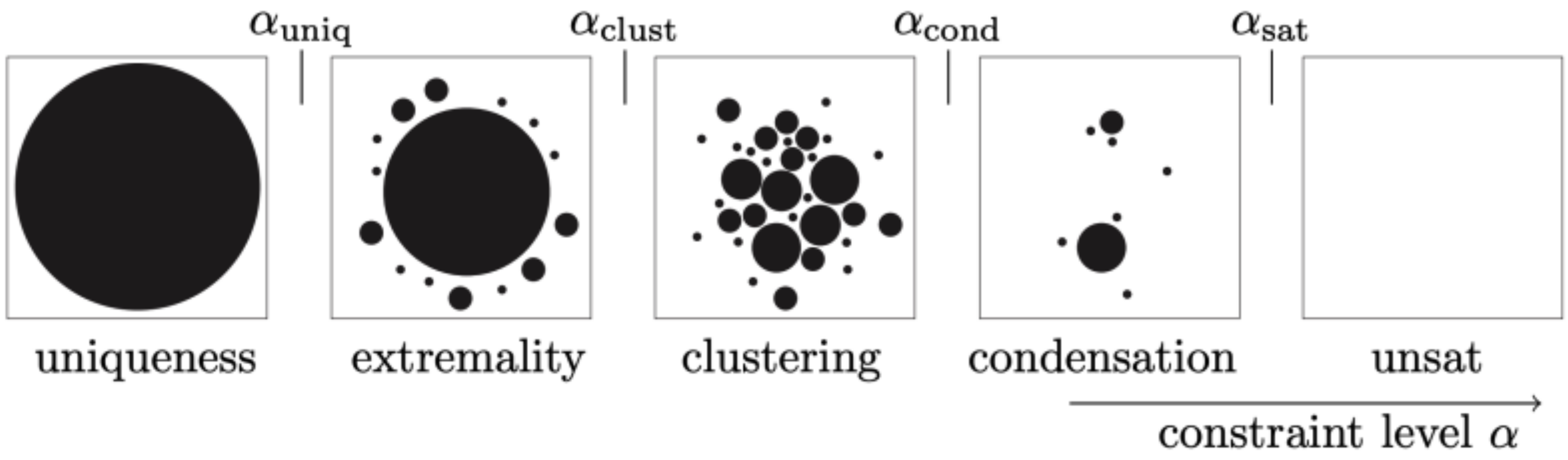
We can compare it to k-SAT with maximum degree $d = k\alpha$

[Bezáková, Galanis, Goldberg, Guo, Štefankovič ’19]: NP-hard when $d \gtrsim 2^{k/2}$!

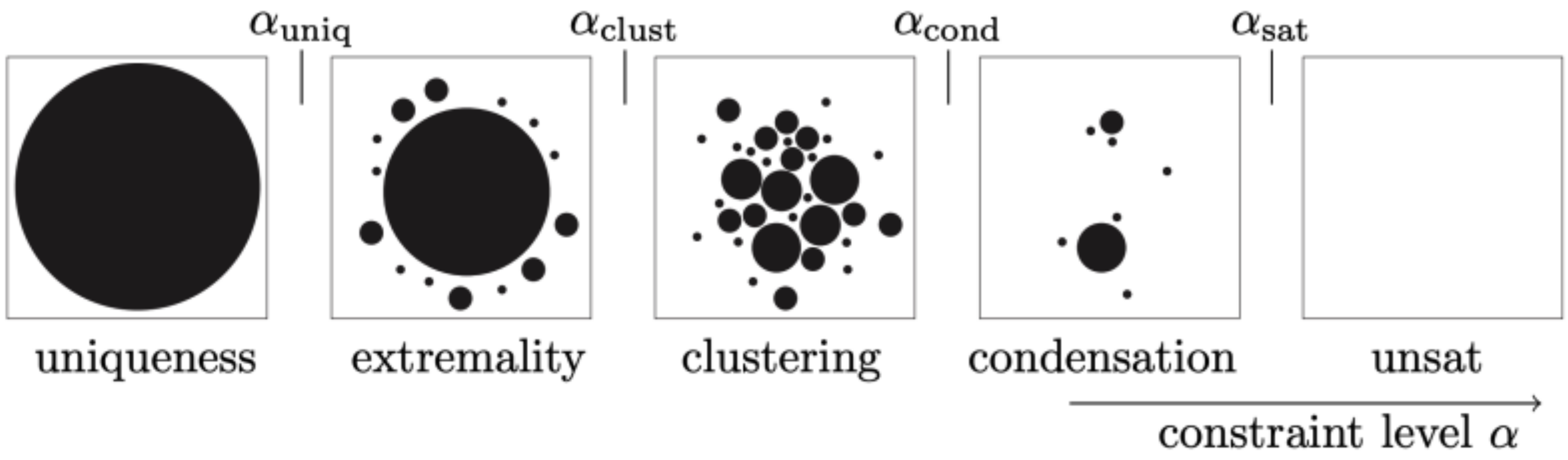


Random k-SAT is computationally easier to sample/count!

Decay of Correlation

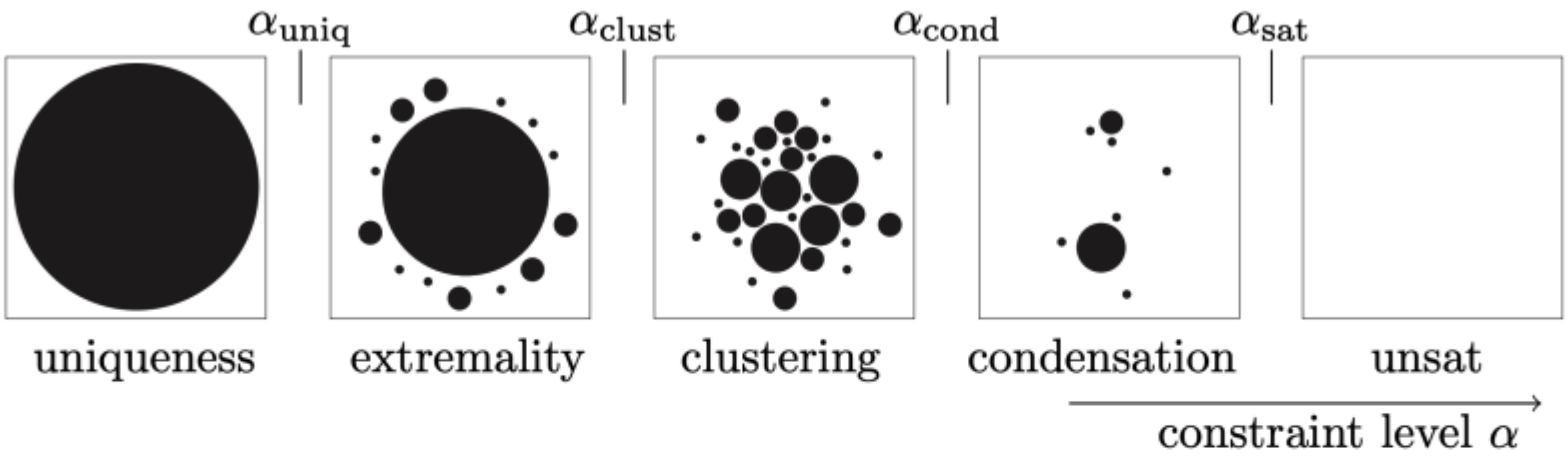


Decay of Correlation



Cavity method: studies the influence of the solution space of flipping one variable

Decay of Correlation



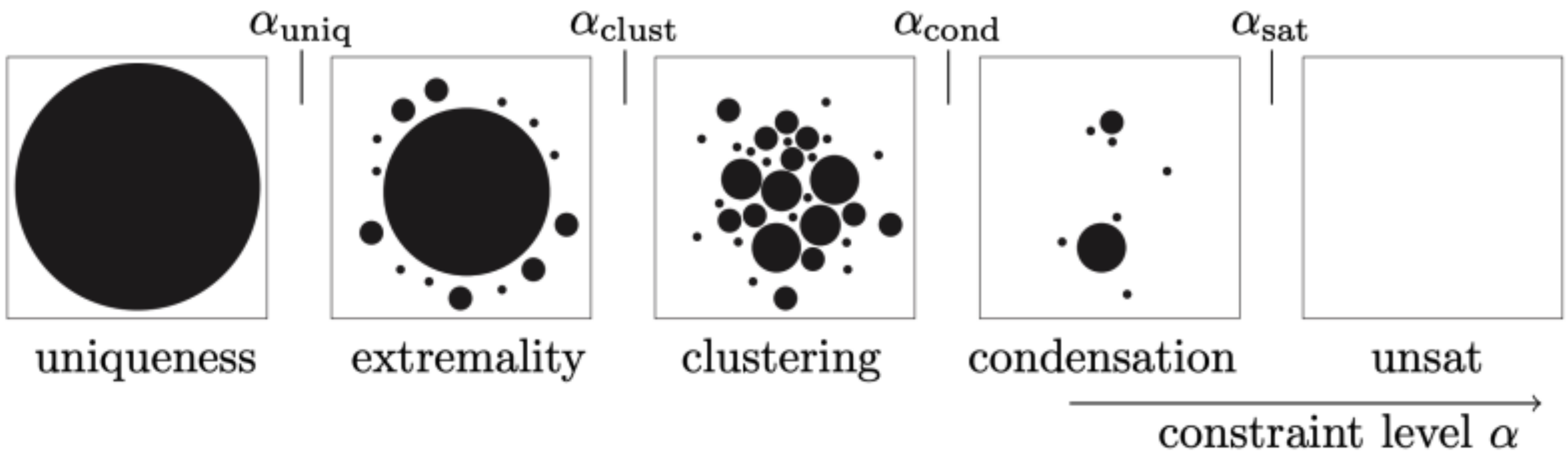
Cavity method: studies the influence of the solution space of flipping one variable

Replica symmetry

For a uniform satisfying assignment σ, and two uniform random variables $v_1, v_2 \in V$,

$$\lim_{n\to\infty} \left| \Pr[\sigma(v_1) = \sigma(v_2) = \text{True}] - \Pr[\sigma(v_1) = \text{True}]\Pr[\sigma(v_2) = \text{True}] \right| = 0.$$

Decay of Correlation



Cavity method: studies the influence of the solution space of flipping one variable

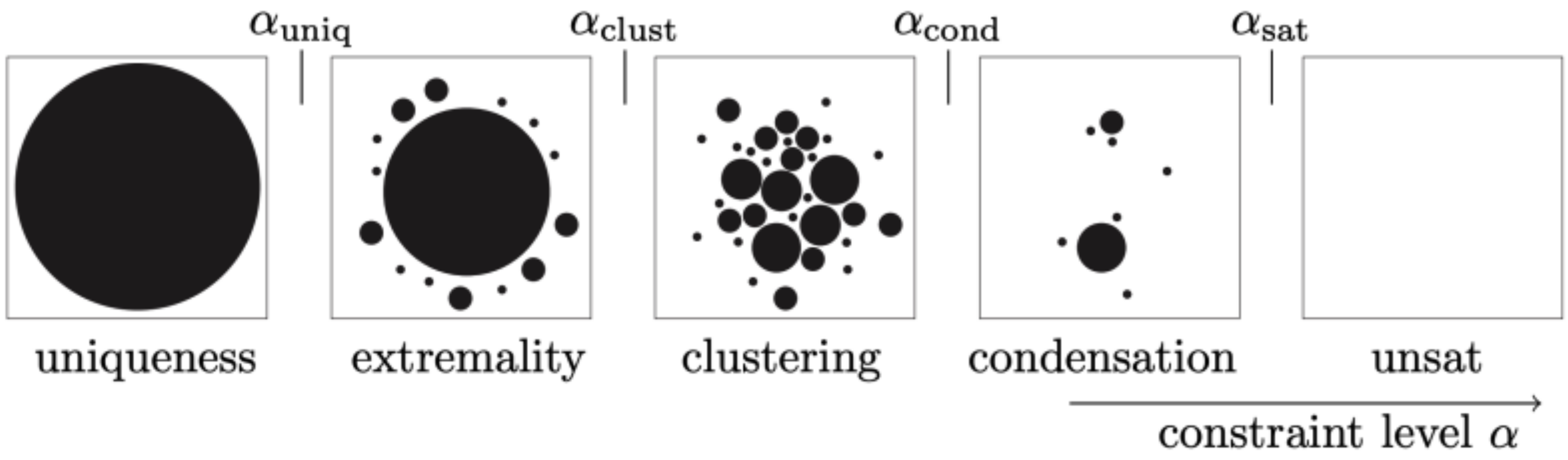
Replica symmetry

For a uniform satisfying assignment σ, and two uniform random variables $v_1, v_2 \in V$,

$$\lim_{n\to\infty} \left| \Pr[\sigma(v_1) = \sigma(v_2) = \text{True}] - \Pr[\sigma(v_1) = \text{True}]\Pr[\sigma(v_2) = \text{True}] \right| = 0.$$

Conjecture: replica symmetry holds up to α_{cond} [COKPZ '17, COEJ et. al.'18]

Decay of Correlation



Cavity method: studies the influence of the solution space of flipping one variable

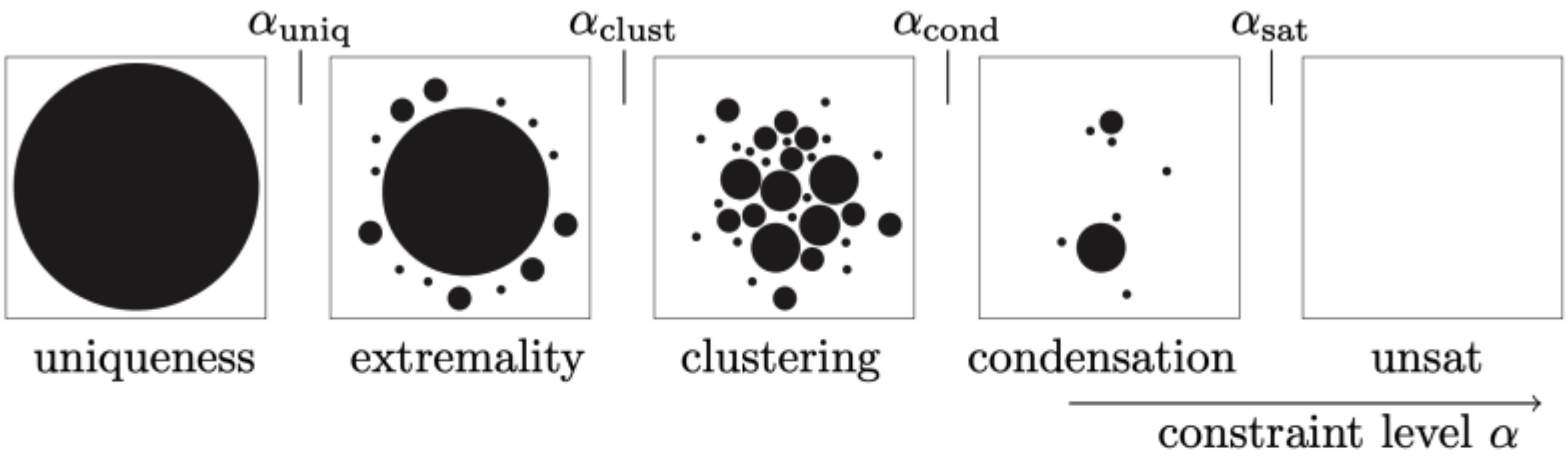
Non-reconstruction

For a uniform satisfying assignment σ, any $v \in V$, and induced hyper graph $H = H_\Phi$

$$\lim_{r\to\infty} \limsup_{n\to\infty} \mathbb{E}\left[d_{\mathrm{TV}}\left(\mu_{\{v\}\cup\bar{B}_H(v,r)}, \mu_v \otimes \mu_{\bar{B}_H(v,r)}\right)\right] = 0,$$

where $\bar{B}_{H(v,r)} \triangleq \{u \in V \mid \mathrm{dist}_H(u,v) \geq r\}$.

Decay of Correlation



Cavity method: studies the influence of the solution space of flipping one variable

Non-reconstruction

For a uniform satisfying assignment σ, any $v \in V$, and induced hyper graph $H = H_\Phi$

$$\lim_{r\to\infty} \limsup_{n\to\infty} \mathbb{E}\left[d_{\mathrm{TV}}\left(\mu_{\{v\}\cup\bar{B}_H(v,r)}, \mu_v \otimes \mu_{\bar{B}_H(v,r)}\right)\right] = 0,$$

where $\bar{B}_{H(v,r)} \triangleq \{u \in V \mid \mathrm{dist}_H(u, v) \geq r\}$.

Conjecture: non-reconstruction holds up to α_{clust} [MPZ '02, MRT '11]

Decay of Correlation

Theorem. (Decay of correlation for random k-SAT)

Let $\Phi = (V, \mathscr{C}) \sim \Phi(k, n, \lfloor \alpha n \rfloor)$. The exists a universal constant $c \geq 1$ such that if

$$0 < \alpha \leq \frac{2^k}{k^c},$$

there exists a coupling (X, Y) of $\mu_{\mathscr{C} \setminus \{c_0\}}$ and $\mu_{\mathscr{C}}$ for any $c \in \mathscr{C}$ such that

$$\mathbb{E}[d_{\mathrm{Ham}}(X, Y)] = O(\log n)\,.$$

$\mu_{\mathscr{C}}$: uniform distribution over solutions of $(V, \mathscr{C})$

$\mu_{\mathscr{C} \setminus \{c_0\}}$: uniform distribution over solutions of $(V, \mathscr{C} \setminus \{c_0\})$

Decay of Correlation

Theorem. (Decay of correlation for random k-SAT)

Let $\Phi = (V, \mathscr{C}) \sim \Phi(k, n, \lfloor \alpha n \rfloor)$. The exists a universal constant $c \geq 1$ such that if

$$0 < \alpha \leq \frac{2^k}{k^c},$$

there exists a coupling (X, Y) of $\mu_{\mathscr{C} \setminus \{c_0\}}$ and $\mu_{\mathscr{C}}$ for any $c \in \mathscr{C}$ such that

$$\mathbb{E}[d_{\mathrm{Ham}}(X, Y)] = O(\log n)\,.$$

$\mu_{\mathscr{C}}$: uniform distribution over solutions of $(V, \mathscr{C})$

$\mu_{\mathscr{C} \setminus \{c_0\}}$: uniform distribution over solutions of $(V, \mathscr{C} \setminus \{c_0\})$

Formal proofs of replica symmetry and non-reconstruction under the same density!

Decay of Correlation

Theorem. (Decay of correlation for random k-SAT)

Let $\Phi = (V, \mathscr{C}) \sim \Phi(k, n, \lfloor \alpha n \rfloor)$. The exists a universal constant $c \geq 1$ such that if

$$0 < \alpha \leq \frac{2^k}{k^c},$$

there exists a coupling (X, Y) of $\mu_{\mathscr{C} \setminus \{c_0\}}$ and $\mu_{\mathscr{C}}$ for any $c \in \mathscr{C}$ such that

$$\mathbb{E}[d_{\mathrm{Ham}}(X, Y)] = O(\log n)\,.$$

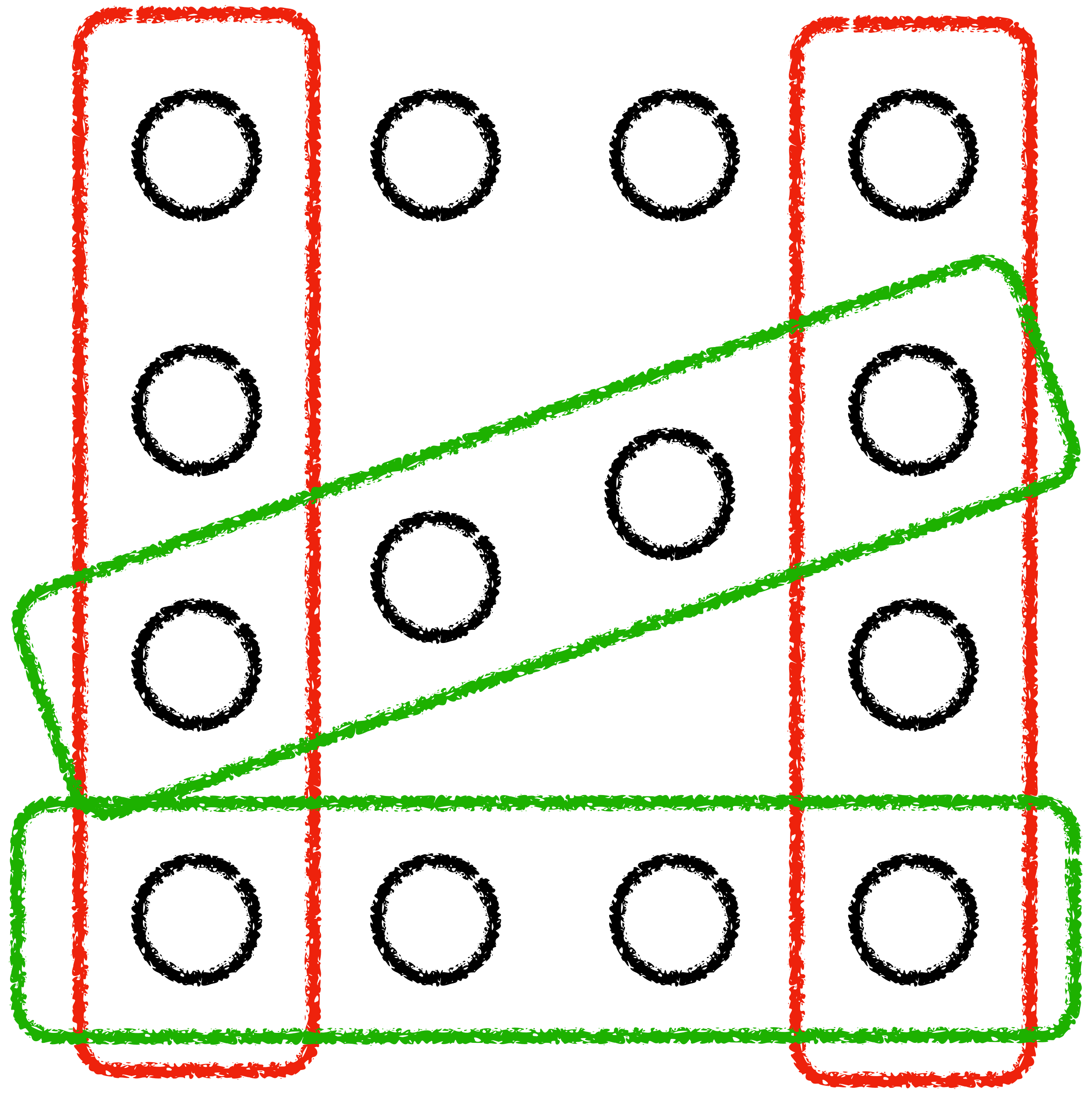
$\mu_{\mathscr{C}}$: uniform distribution over solutions of $(V, \mathscr{C})$

$\mu_{\mathscr{C} \setminus \{c_0\}}$: uniform distribution over solutions of $(V, \mathscr{C} \setminus \{c_0\})$

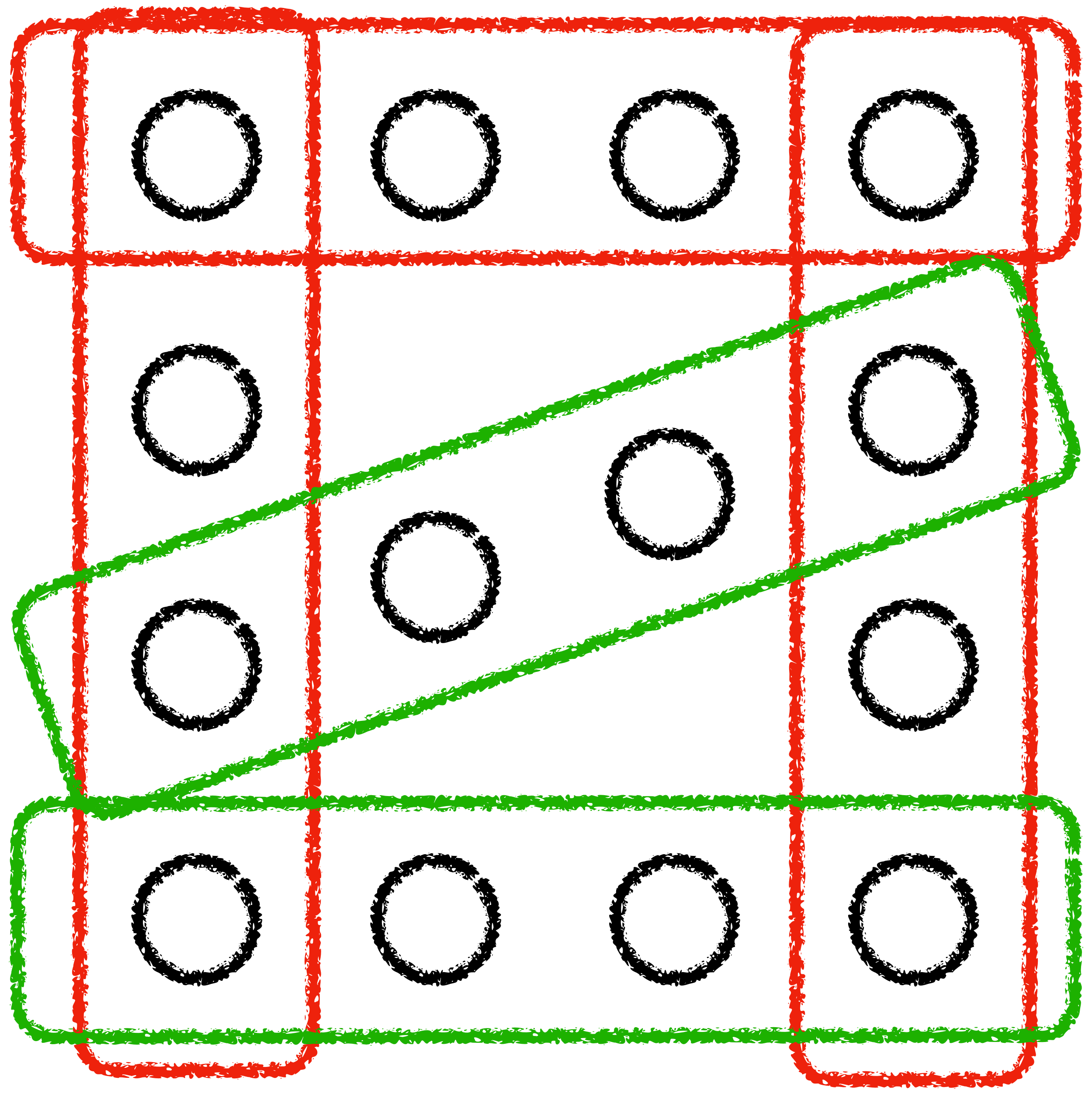
Formal proofs of replica symmetry and non-reconstruction under the same density!

Inspired by the coupling in [W., Yin '24] for bounded degree CSPs

Recursive Coupling [WY '24]



$(V, \mathscr{C} \backslash \{c_0\})$



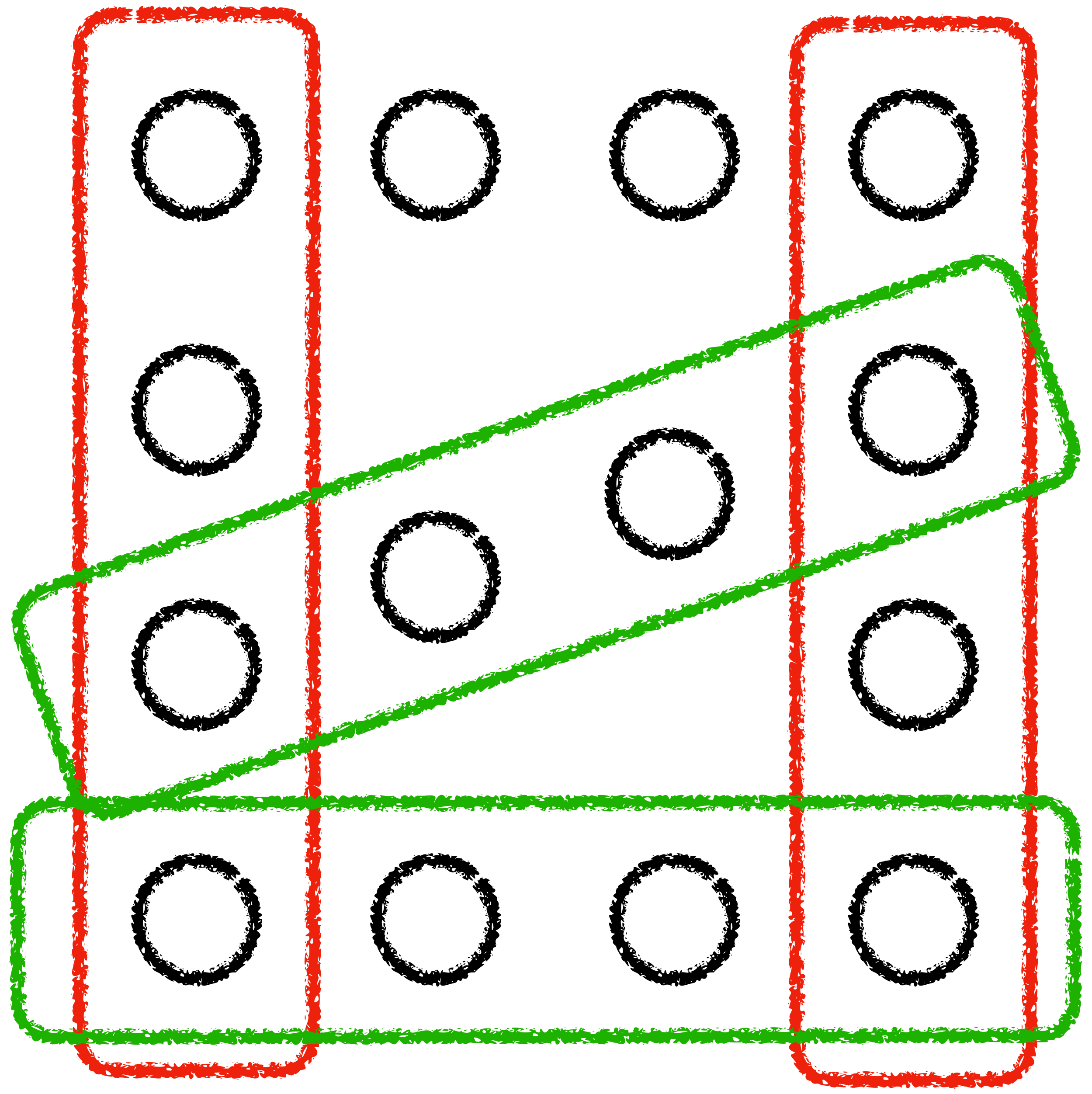
$(V, \mathscr{C})$

red clause: need at least one red variable

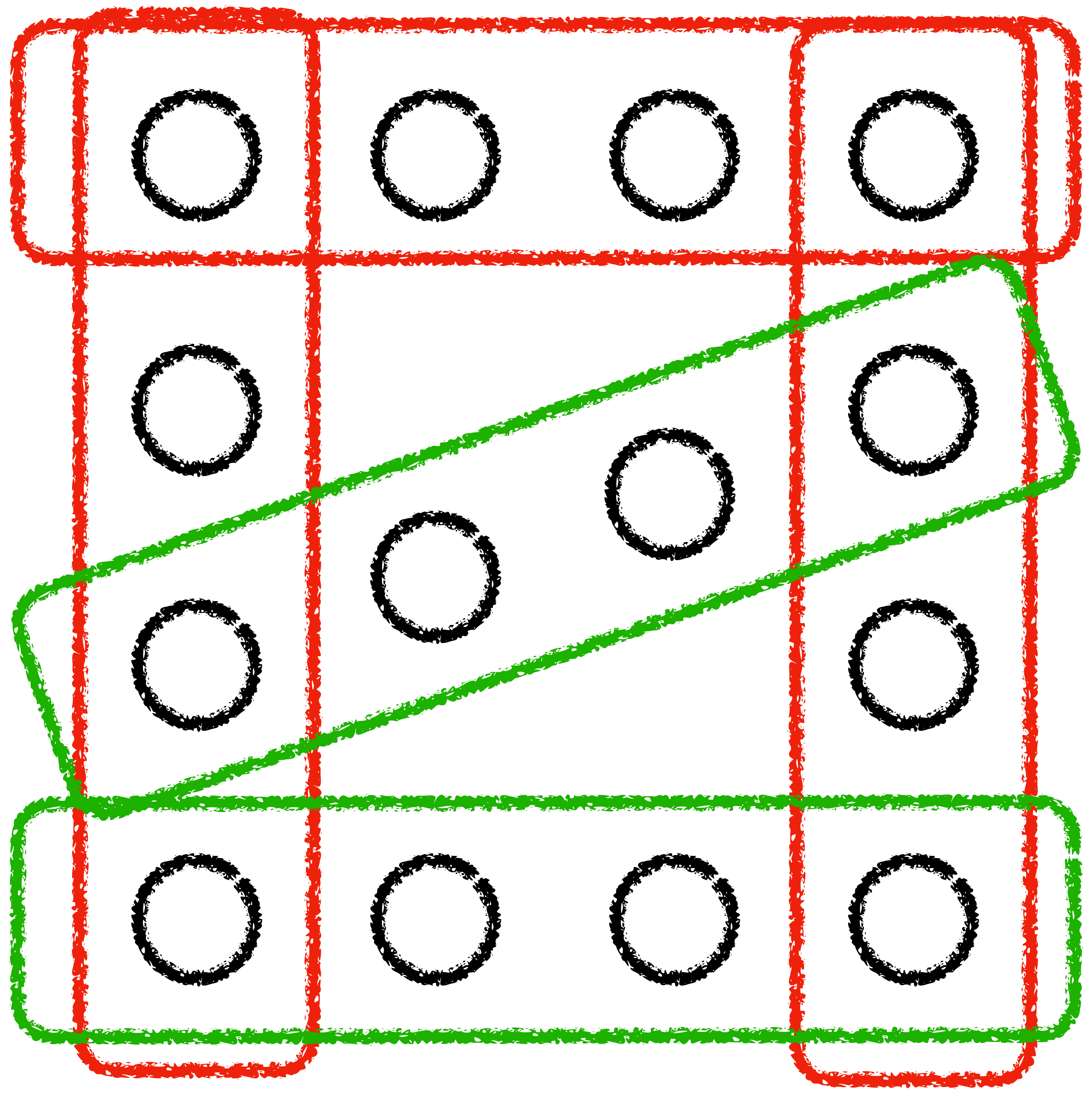
green clause: need at least one green variable

We want to couple $\mu_{\mathscr{C} \backslash \{c_0\}}$ with $\mu_{\mathscr{C}}$.

Recursive Coupling [WY '24]



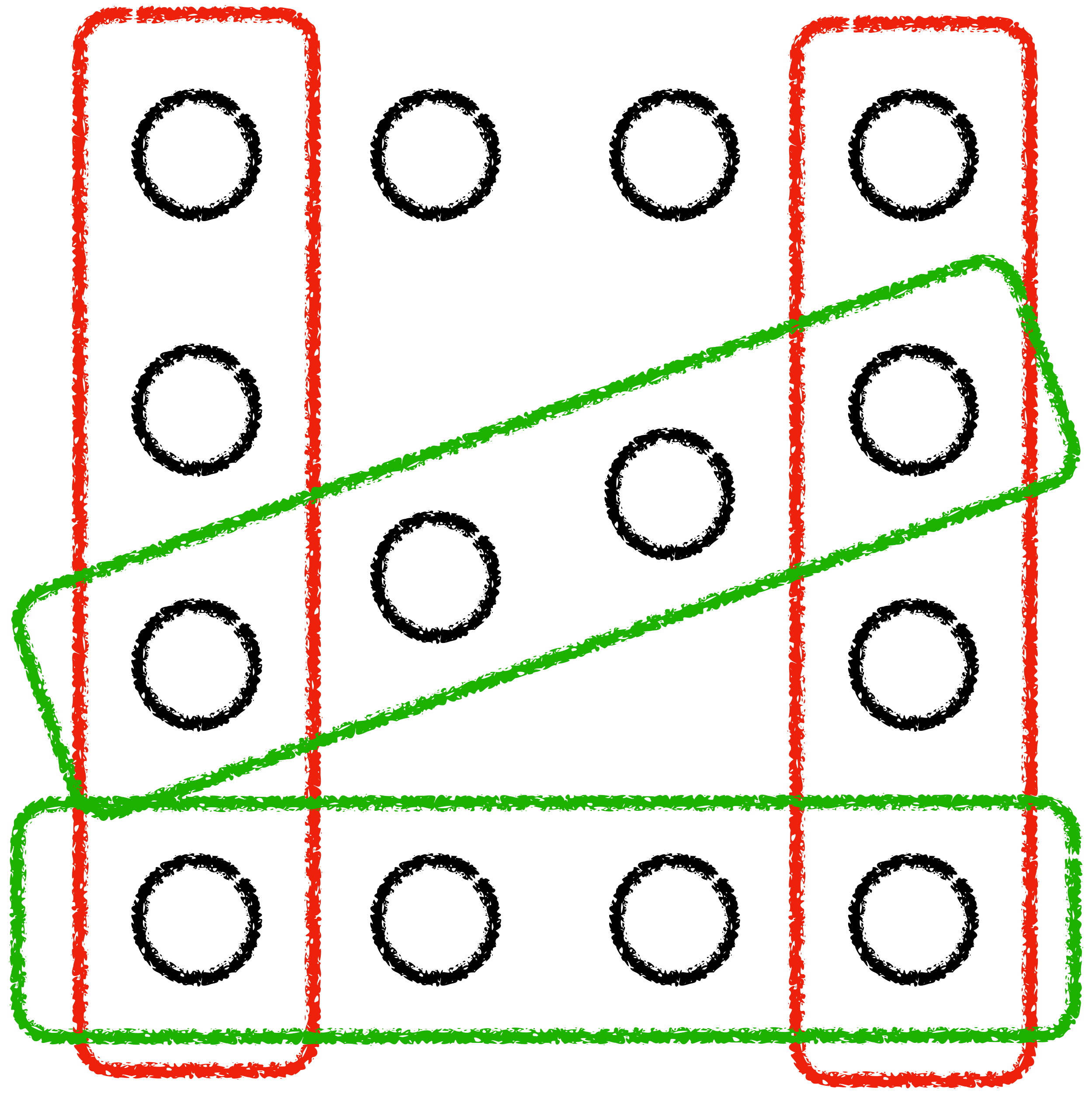
$(V, \mathscr{C} \backslash \{c_0\})$



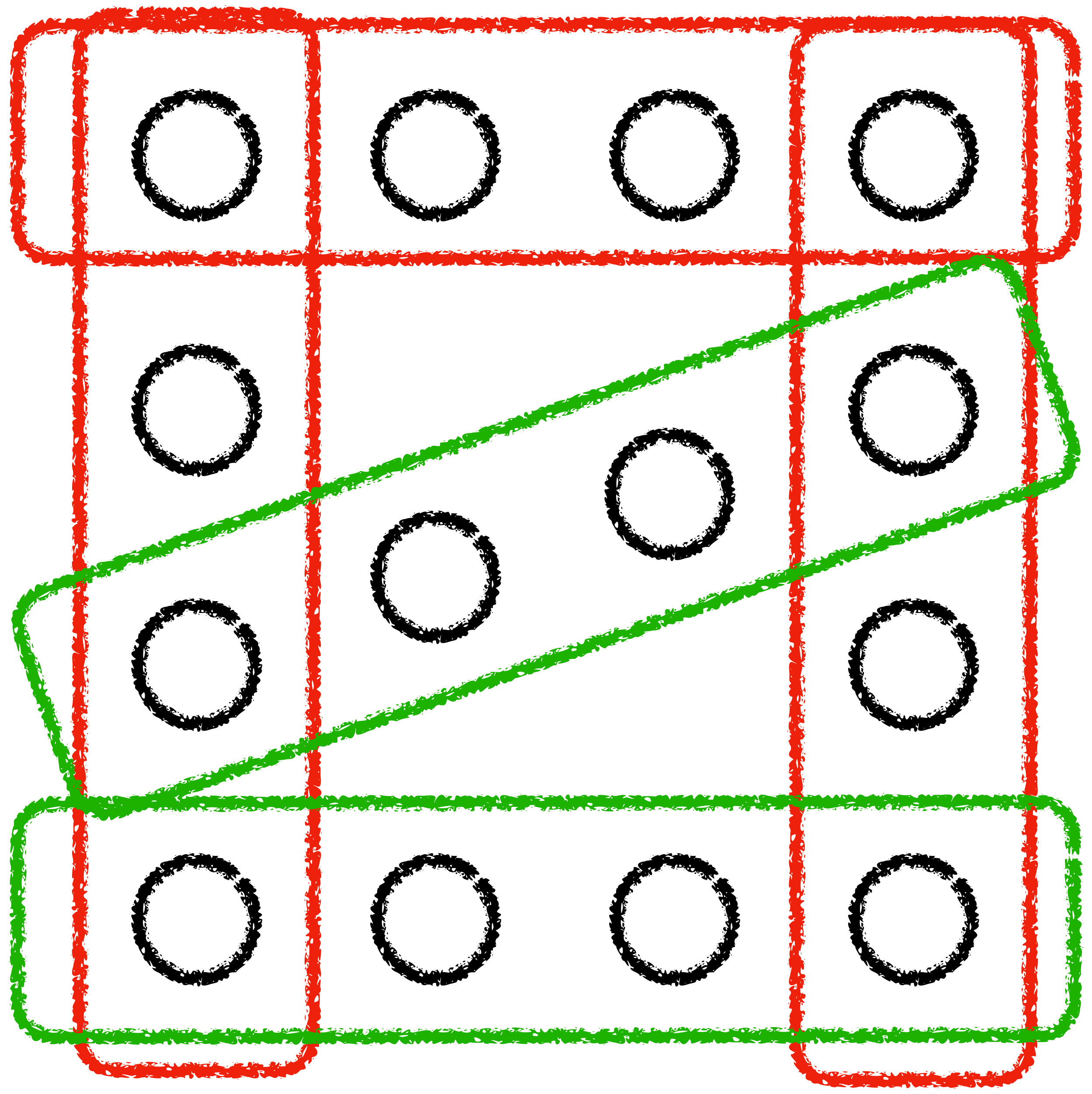
$(V, \mathscr{C})$

$$\mu_{\mathscr{C}\backslash\{c_0\}} = \mu_{\mathscr{C}\backslash\{c_0\}}(c_0) \cdot \mu_{\mathscr{C}} + \mu_{\mathscr{C}\backslash\{c_0\}}(\neg c_0) \cdot \mu_{\mathscr{C}\backslash\{c_0\}}(\cdot \mid \neg c_0)$$

Recursive Coupling [WY '24]



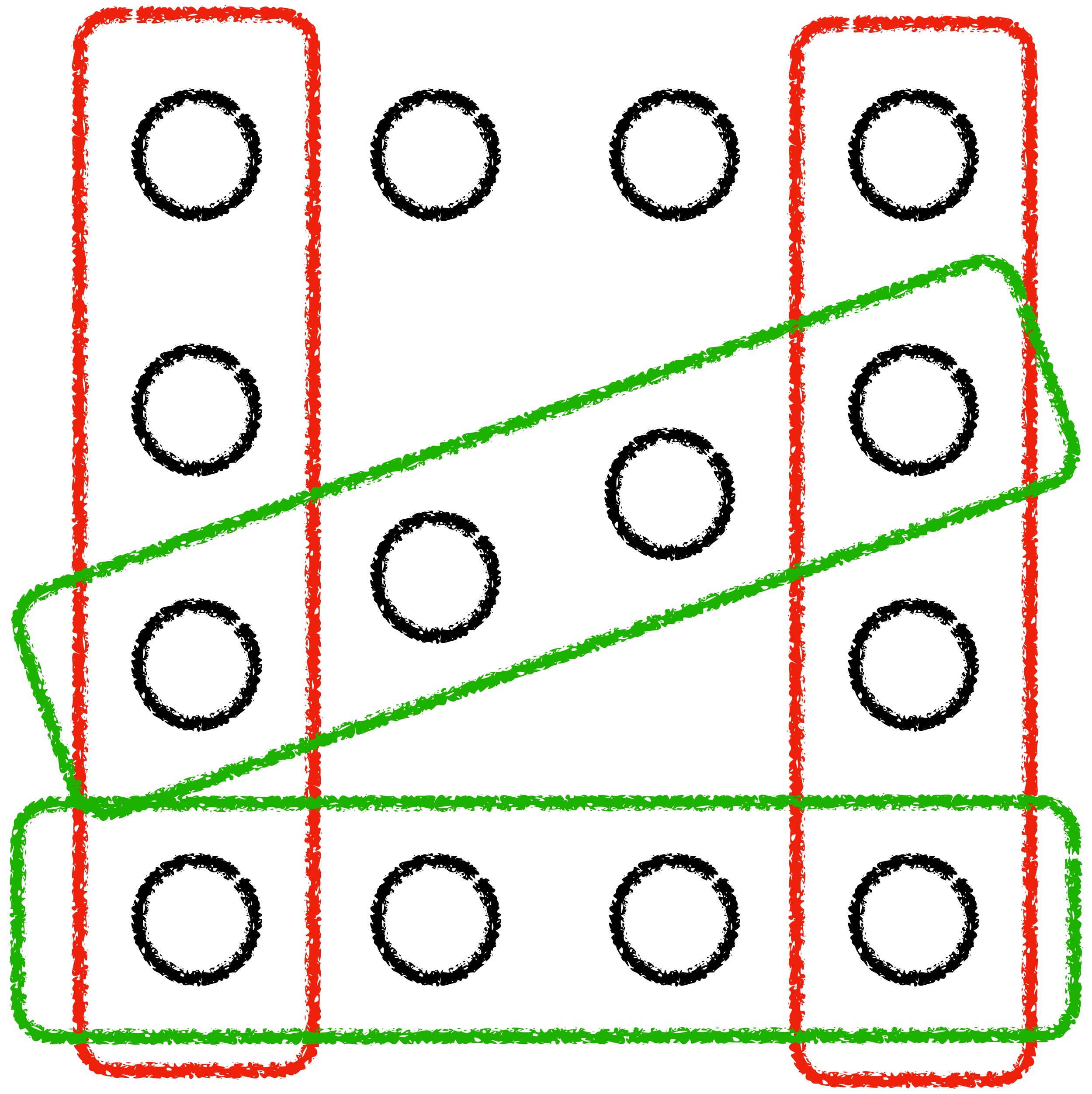
$(V, \mathscr{C} \backslash \{c_0\})$



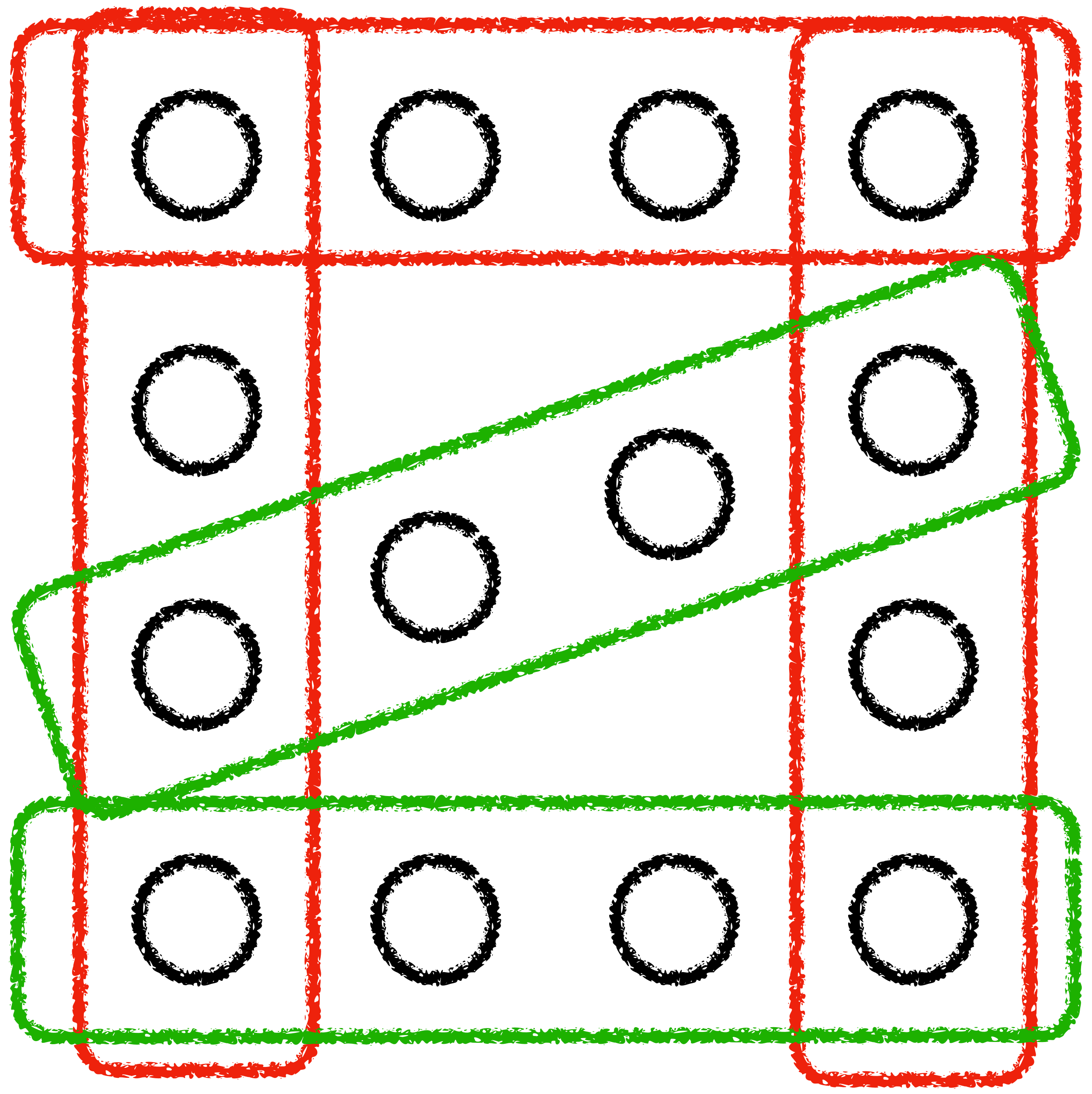
$(V, \mathscr{C})$

with prob. $\mu_{\mathscr{C}\backslash\{c_0\}}(c_0)$, couple $\mu_{\mathscr{C}}$ with $\mu_{\mathscr{C}}$;
with prob. $\mu_{\mathscr{C}\backslash\{c_0\}}(\neg c_0)$, couple $\mu_{\mathscr{C}\backslash\{c_0\}}(\cdot \mid \neg c_0)$ with $\mu_{\mathscr{C}}$.

Recursive Coupling [WY '24]



$(V, \mathscr{C} \backslash \{c_0\})$

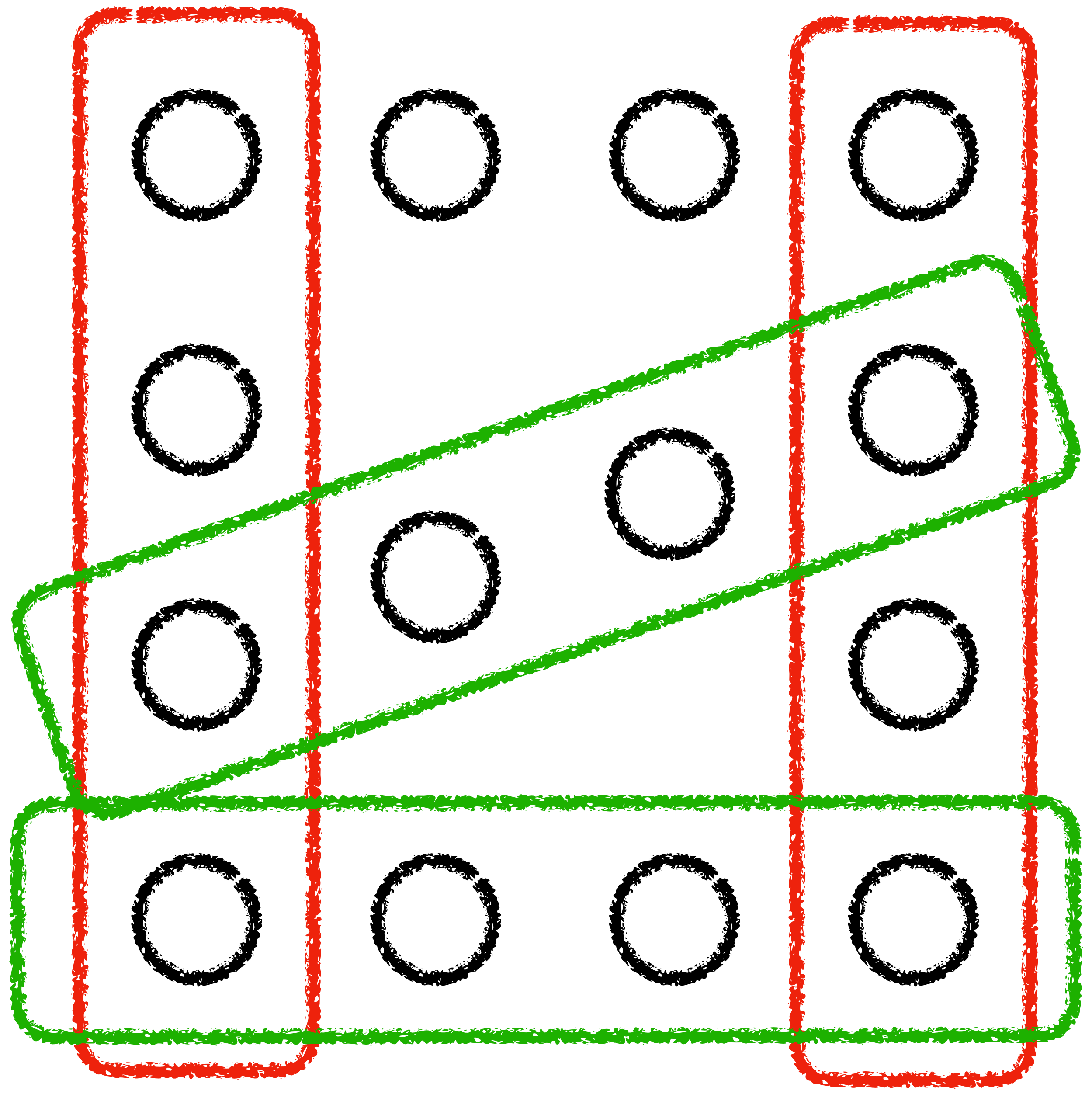


$(V, \mathscr{C})$

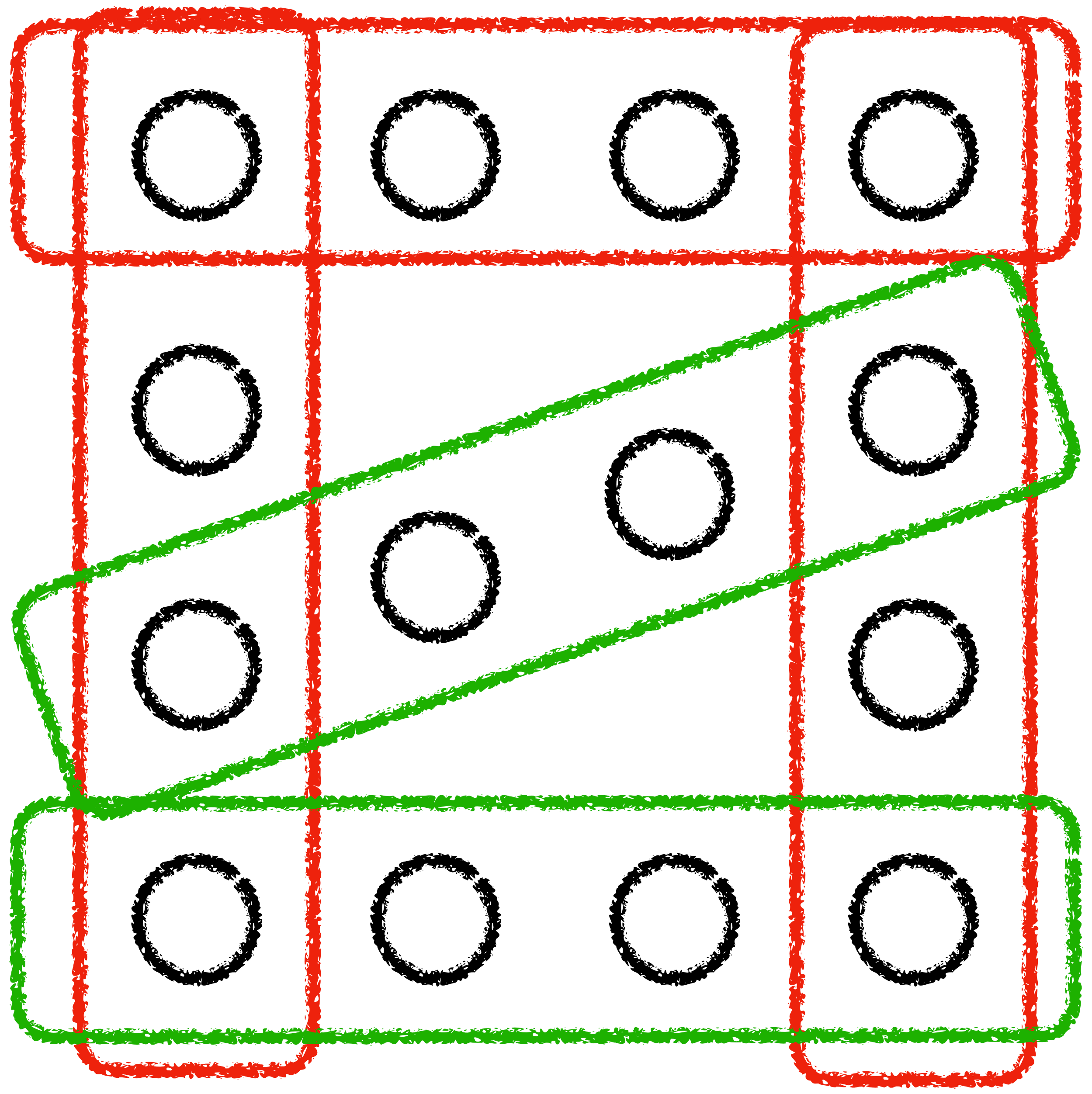
with prob. $\mu_{\mathscr{C}\backslash\{c_0\}}(c_0)$, couple $\mu_{\mathscr{C}}$ with $\mu_{\mathscr{C}}$; **can be perfectly coupled!**

with prob. $\mu_{\mathscr{C}\backslash\{c_0\}}(\neg c_0)$, couple $\mu_{\mathscr{C}\backslash\{c_0\}}(\,\cdot \mid \neg c_0)$ with $\mu_{\mathscr{C}}$.

Recursive Coupling [WY ’24]



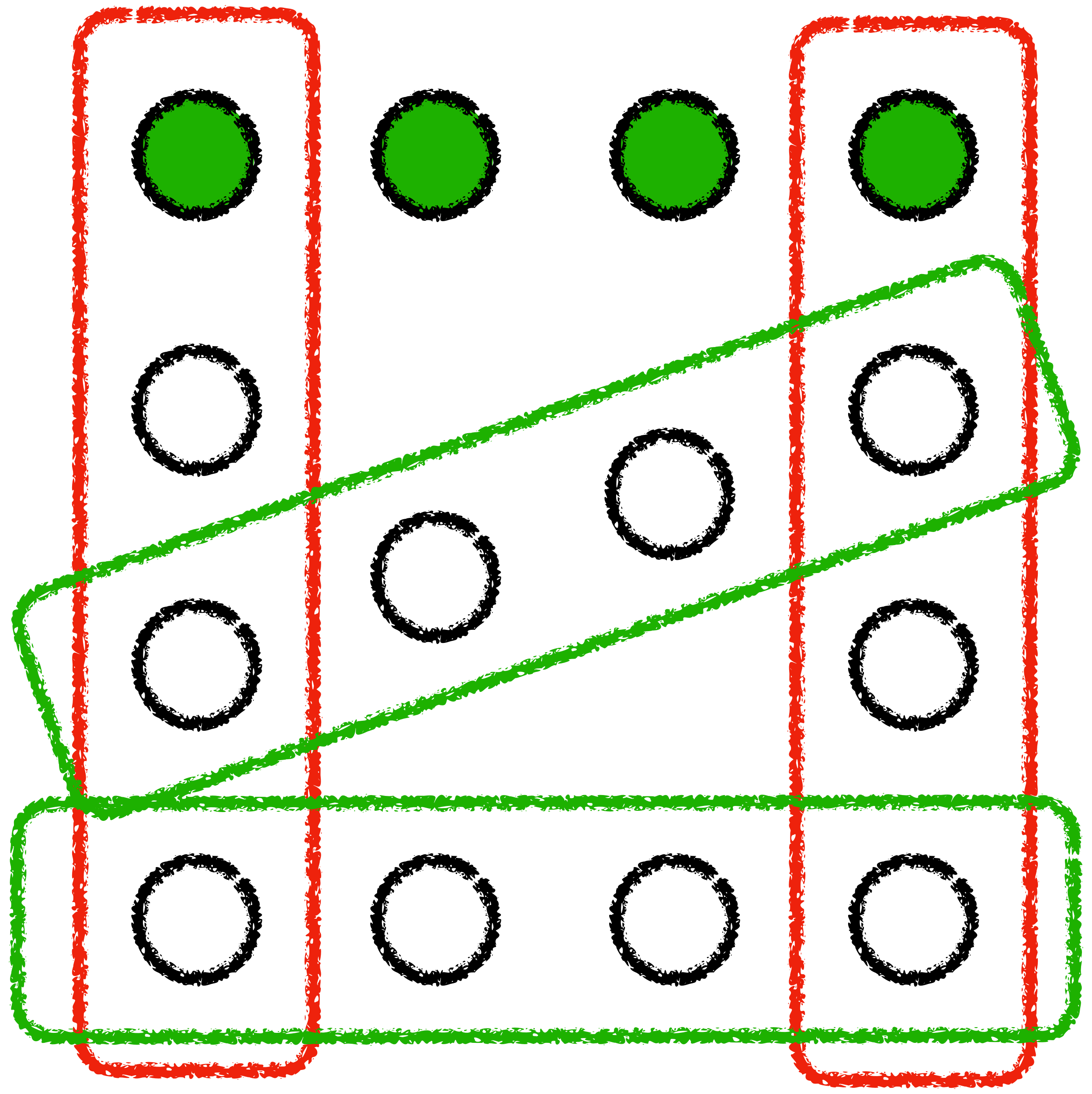
$(V, \mathscr{C} \backslash \{c_0\})$



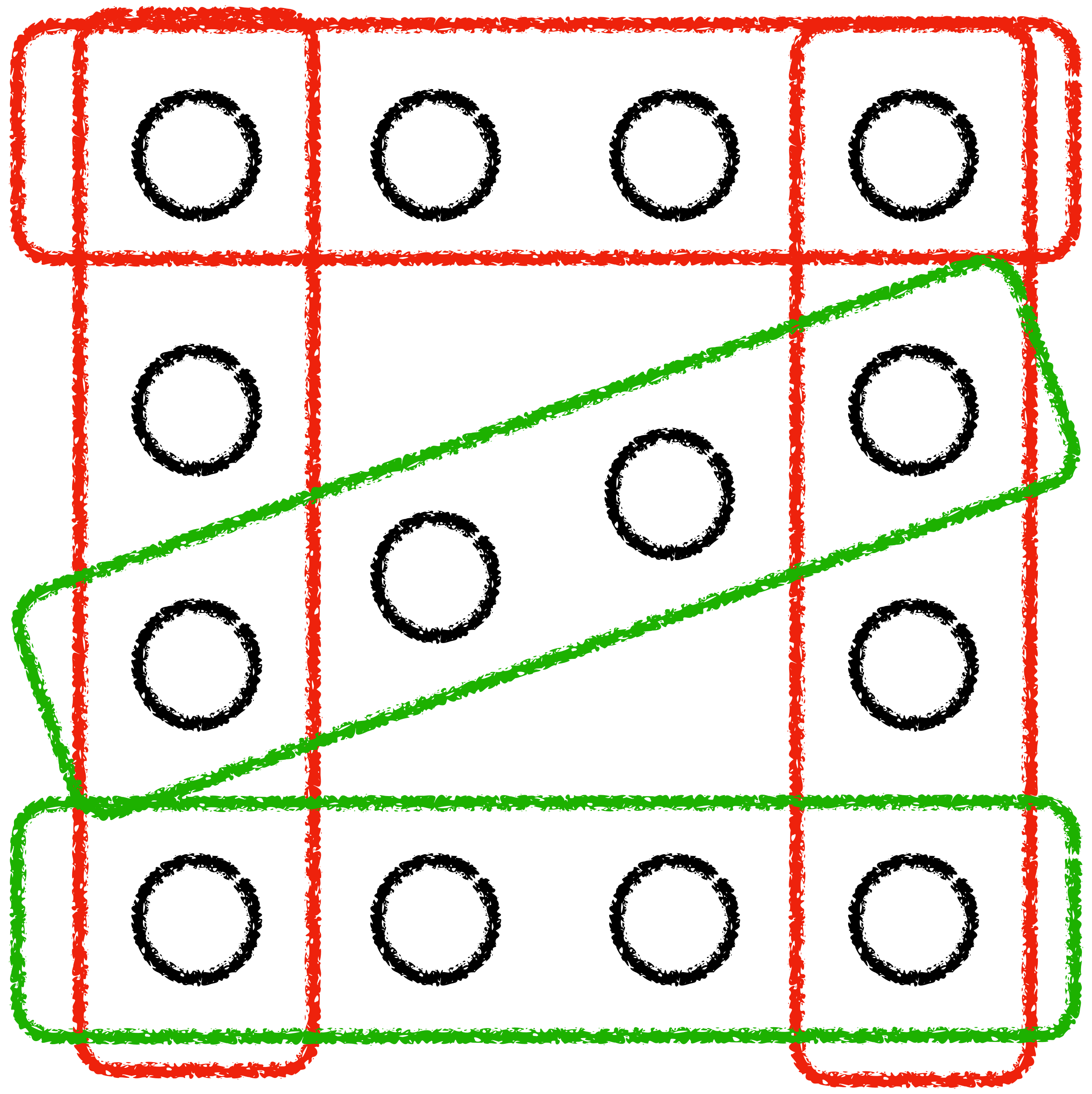
$(V, \mathscr{C})$

We now couple $\mu_{\mathscr{C} \backslash \{c_0\}}(\cdot \mid \neg c_0)$ with $\mu_{\mathscr{C}}$.

Recursive Coupling [WY '24]



$(V, \mathscr{C} \setminus \{c_0\})$



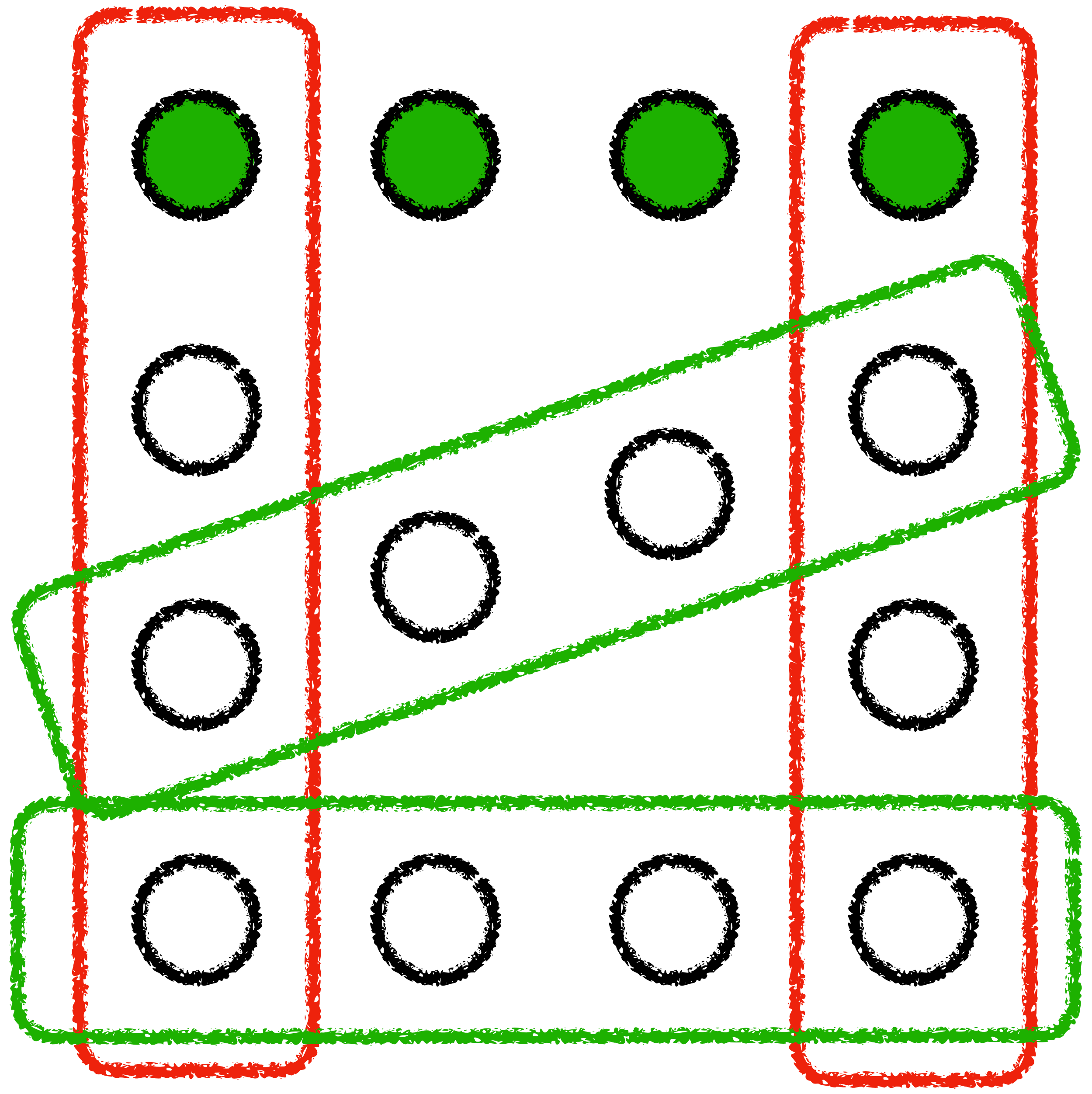
$(V, \mathscr{C})$

forced assignment !

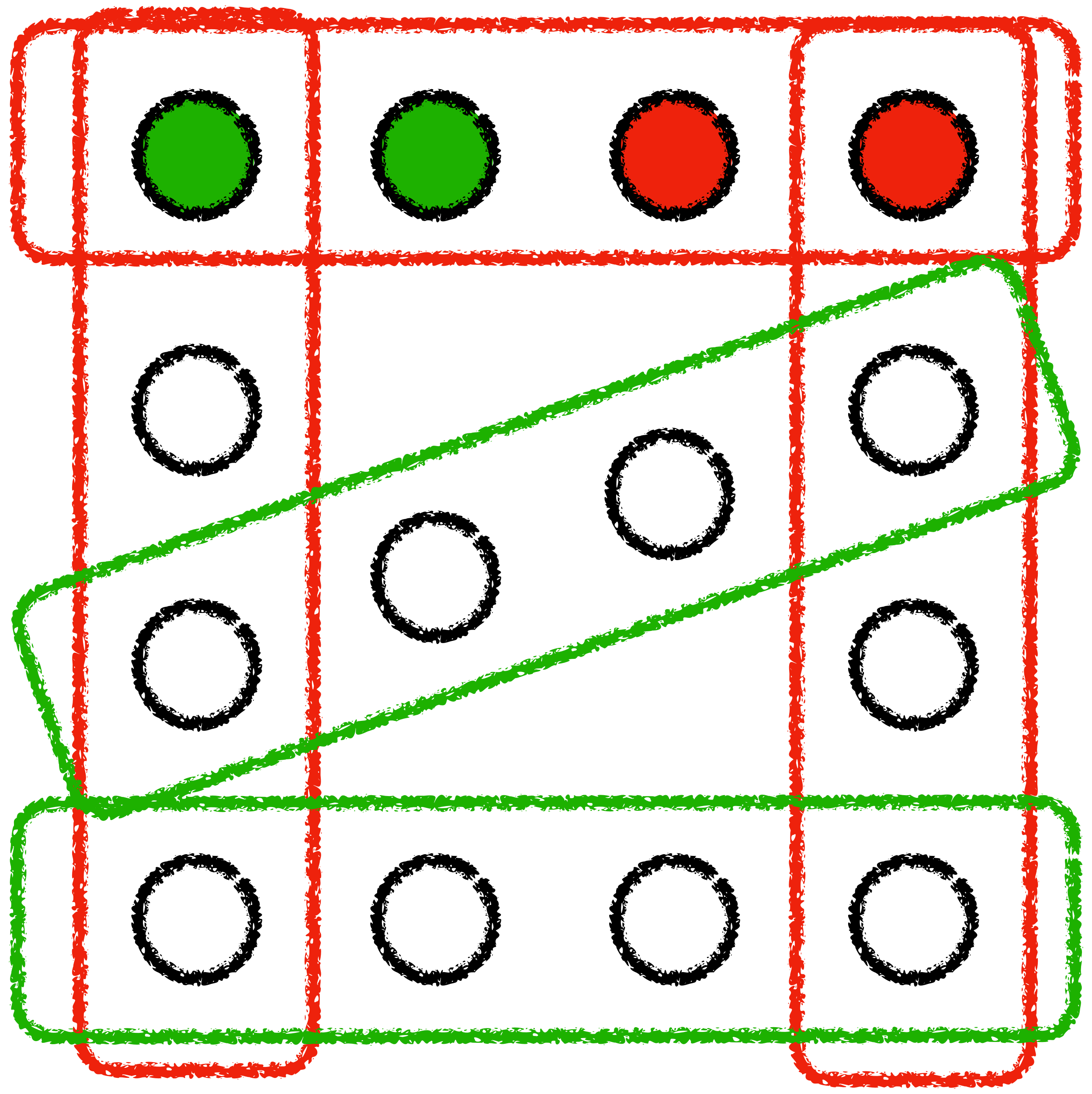
We now couple $\mu_{\mathscr{C}\setminus\{c_0\}}(\cdot \mid \neg c_0)$ with $\mu_{\mathscr{C}}$.

We further decompose $\mu_{\mathscr{C}} = \sum_{\rho \in \{R,G\}^{\text{vbl}(c_0)}} \mu_{\mathscr{C}}(\rho) \cdot \mu_{\mathscr{C}}(\cdot \mid \rho)$.

Recursive Coupling [WY '24]



$(V, \mathscr{C} \setminus \{c_0\})$

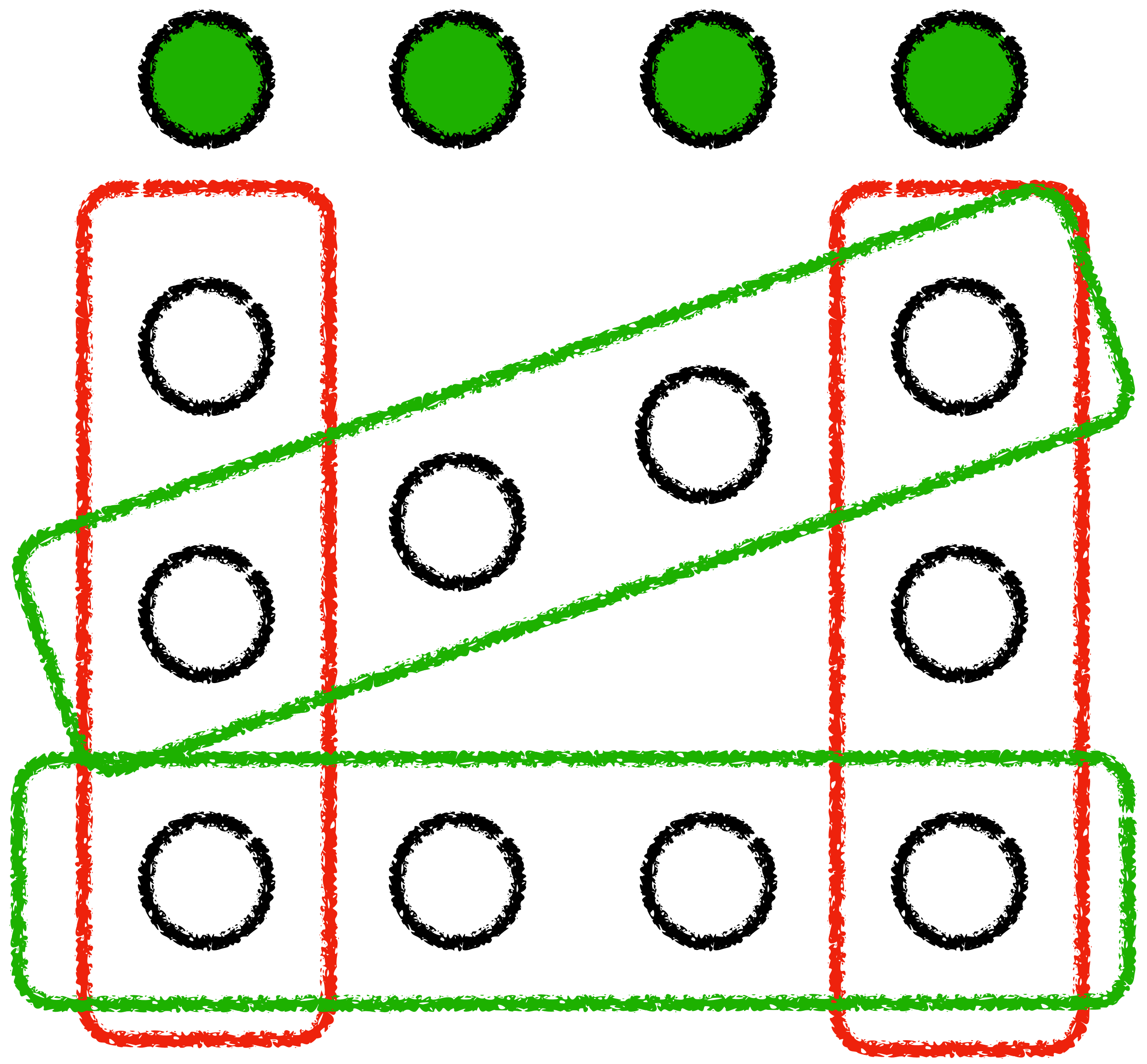


$(V, \mathscr{C})$

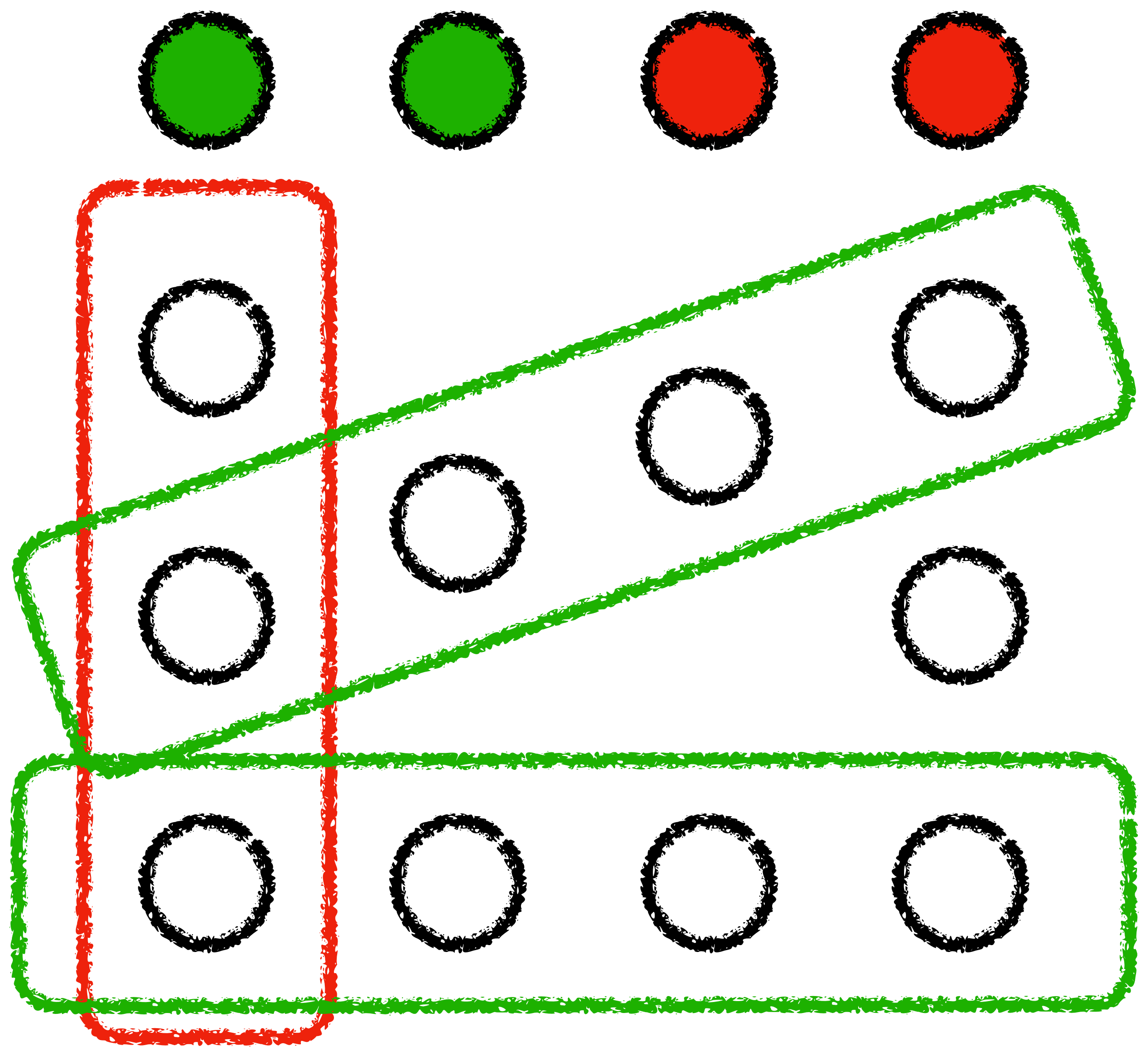
We now couple $\mu_{\mathscr{C}\setminus\{c_0\}}(\cdot \mid \neg c_0)$ with $\mu_{\mathscr{C}}$.

We further decompose $\mu_{\mathscr{C}} = \sum_{\rho \in \{R,G\}^{\mathrm{vbl}(c_0)}} \mu_{\mathscr{C}}(\rho) \cdot \mu_{\mathscr{C}}(\cdot \mid \rho)$.

Recursive Coupling [WY '24]



$(V, \mathscr{C} \setminus \{c_0\})$

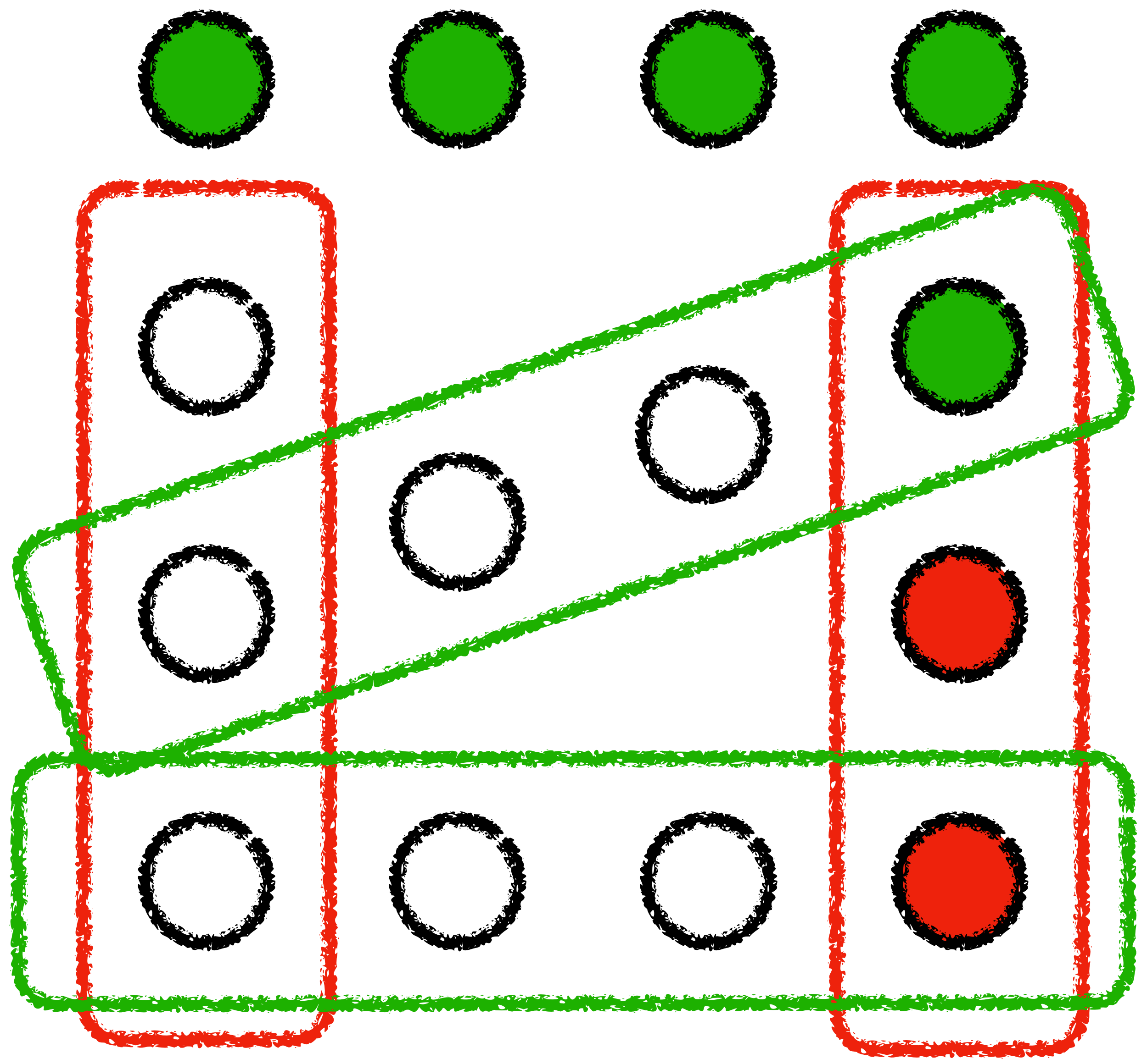


$(V, \mathscr{C})$

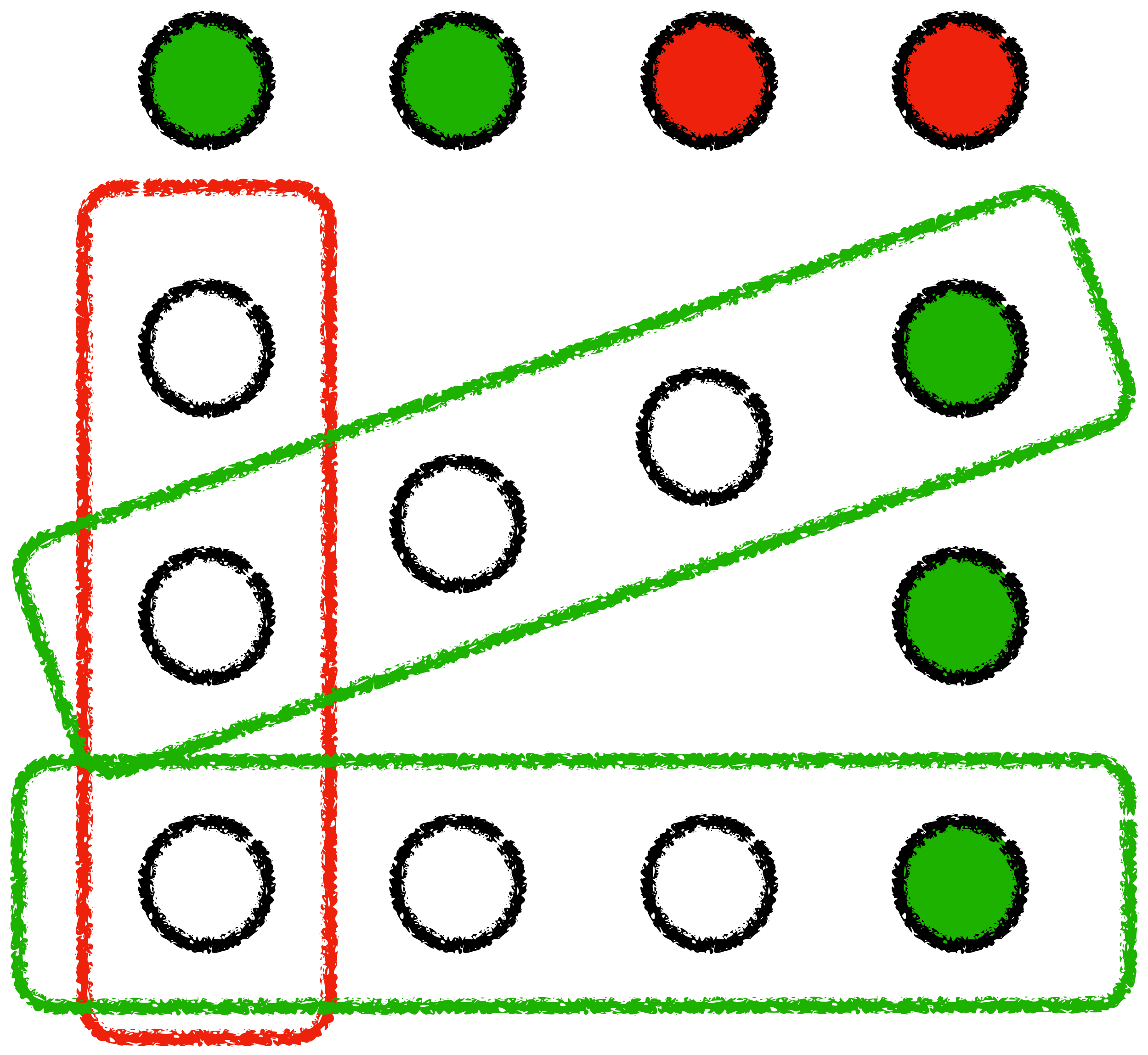
Simplify the formula, we are done if the set of clauses are the same.

Otherwise, we pick any clause in the discrepancy set and recurse!

Recursive Coupling [WY '24]



$(V, \mathscr{C} \backslash \{c_0\})$

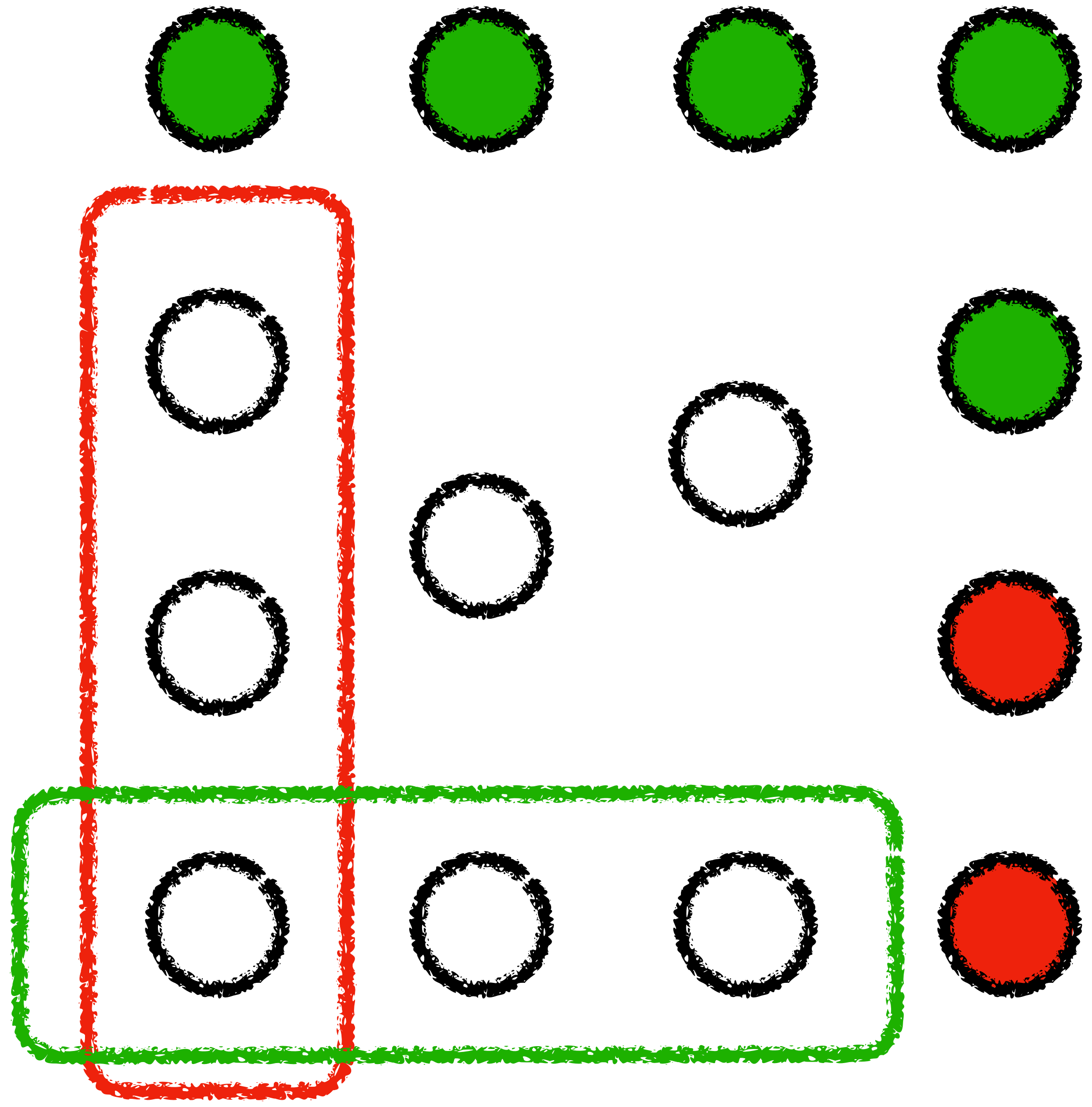


$(V, \mathscr{C})$

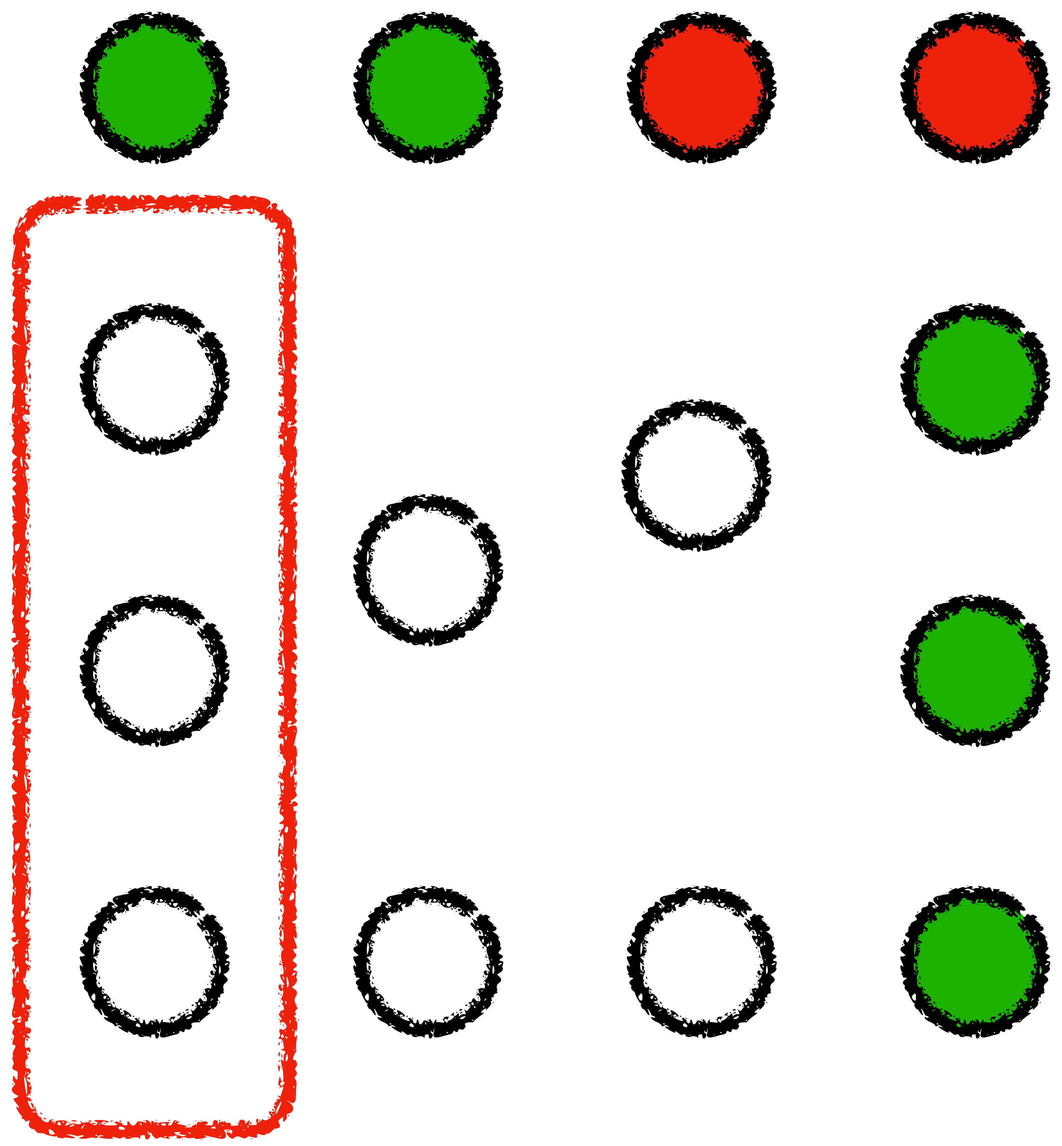
Simplify the formula, we are done if the set of clauses are the same.

Otherwise, we pick any clause in the discrepancy set and recurse!

Recursive Coupling [WY '24]



$(V, \mathscr{C} \backslash \{c_0\})$

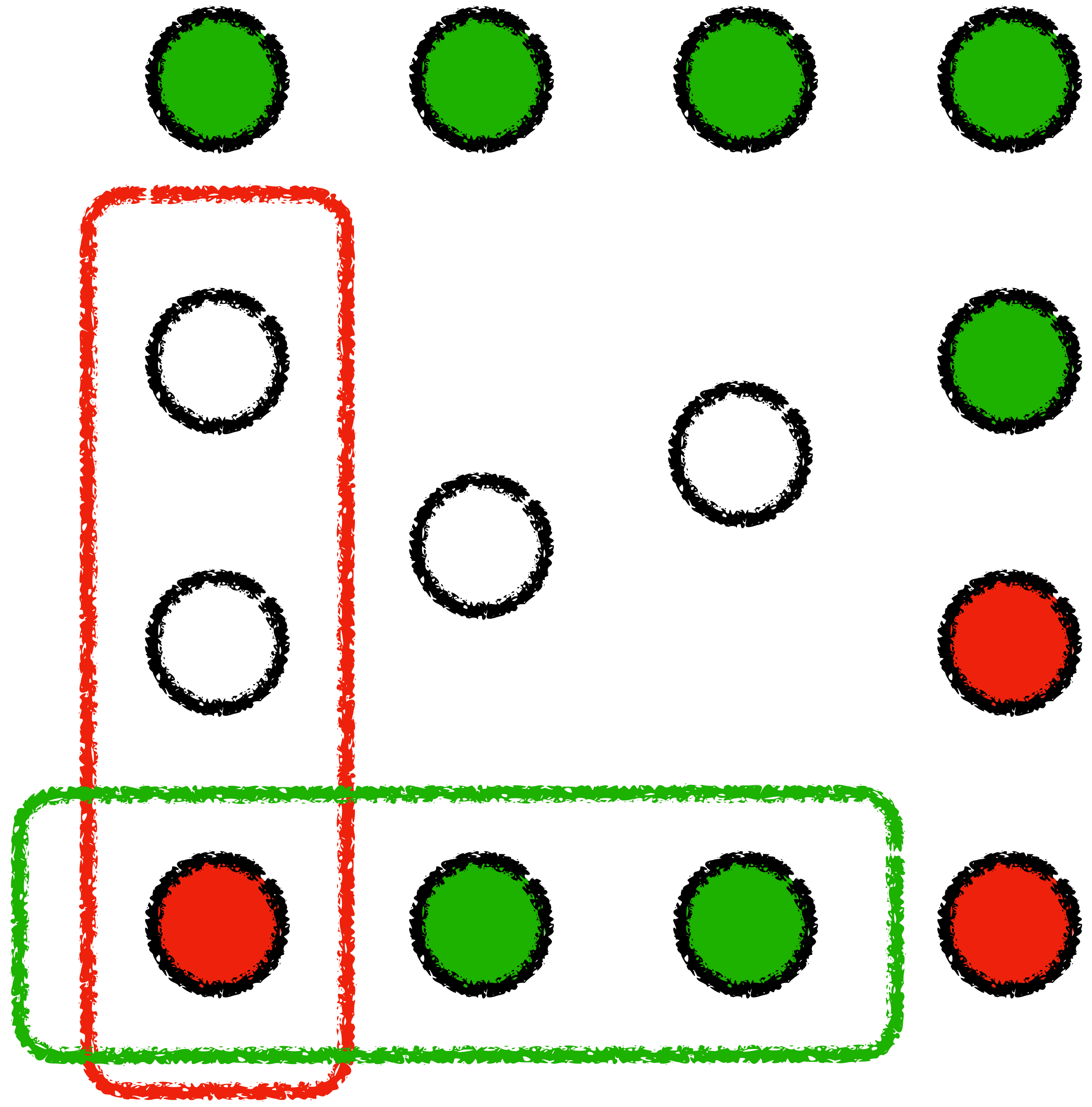


$(V, \mathscr{C})$

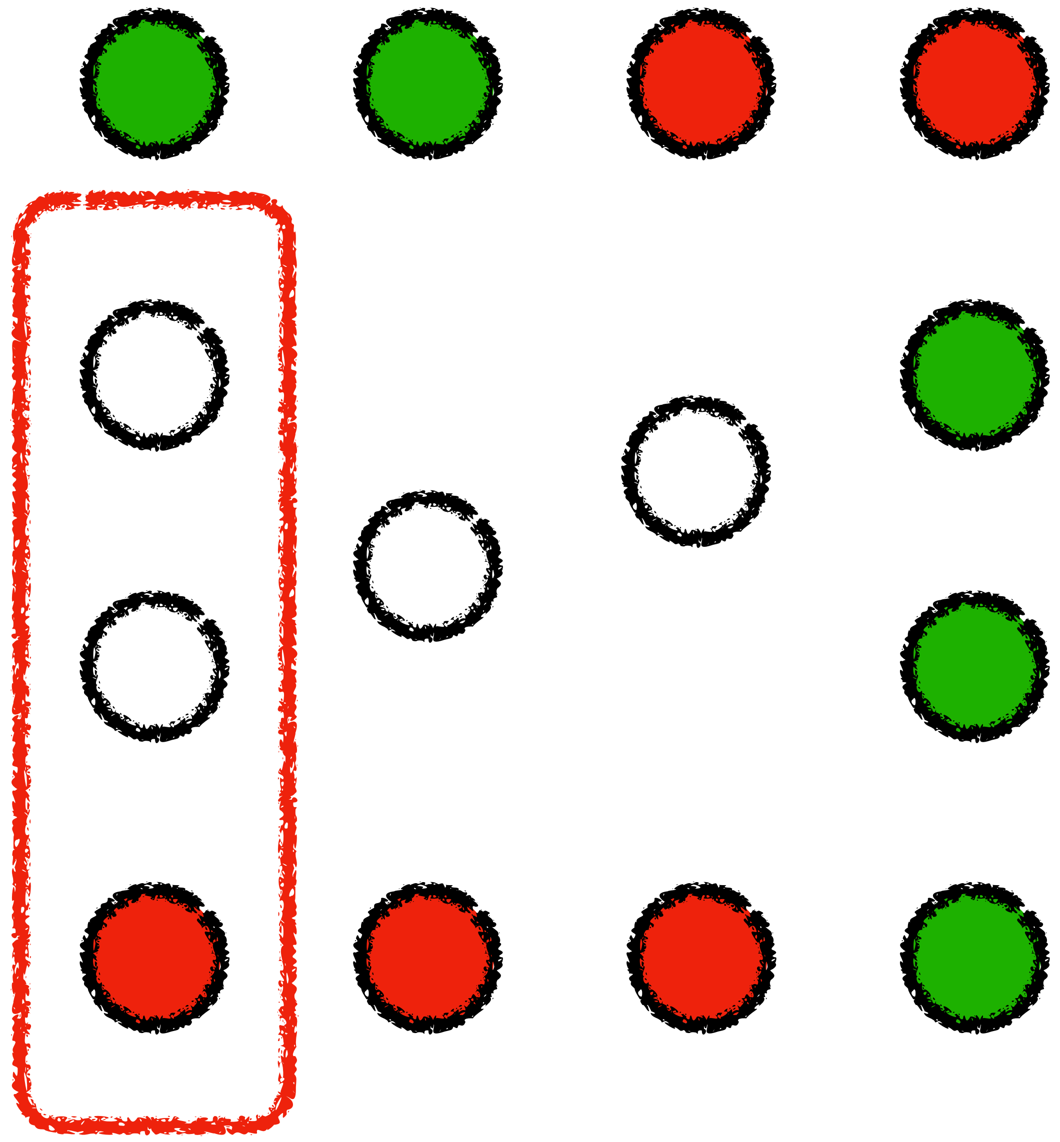
Simplify the formula, we are done if the set of clauses are the same.

Otherwise, we pick any clause in the discrepancy set and recurse!

Recursive Coupling [WY '24]



$(V, \mathscr{C} \setminus \{c_0\})$



$(V, \mathscr{C})$

Simplify the formula, we are done if the set of clauses are the same.

Otherwise, we pick any clause in the discrepancy set and recurse!

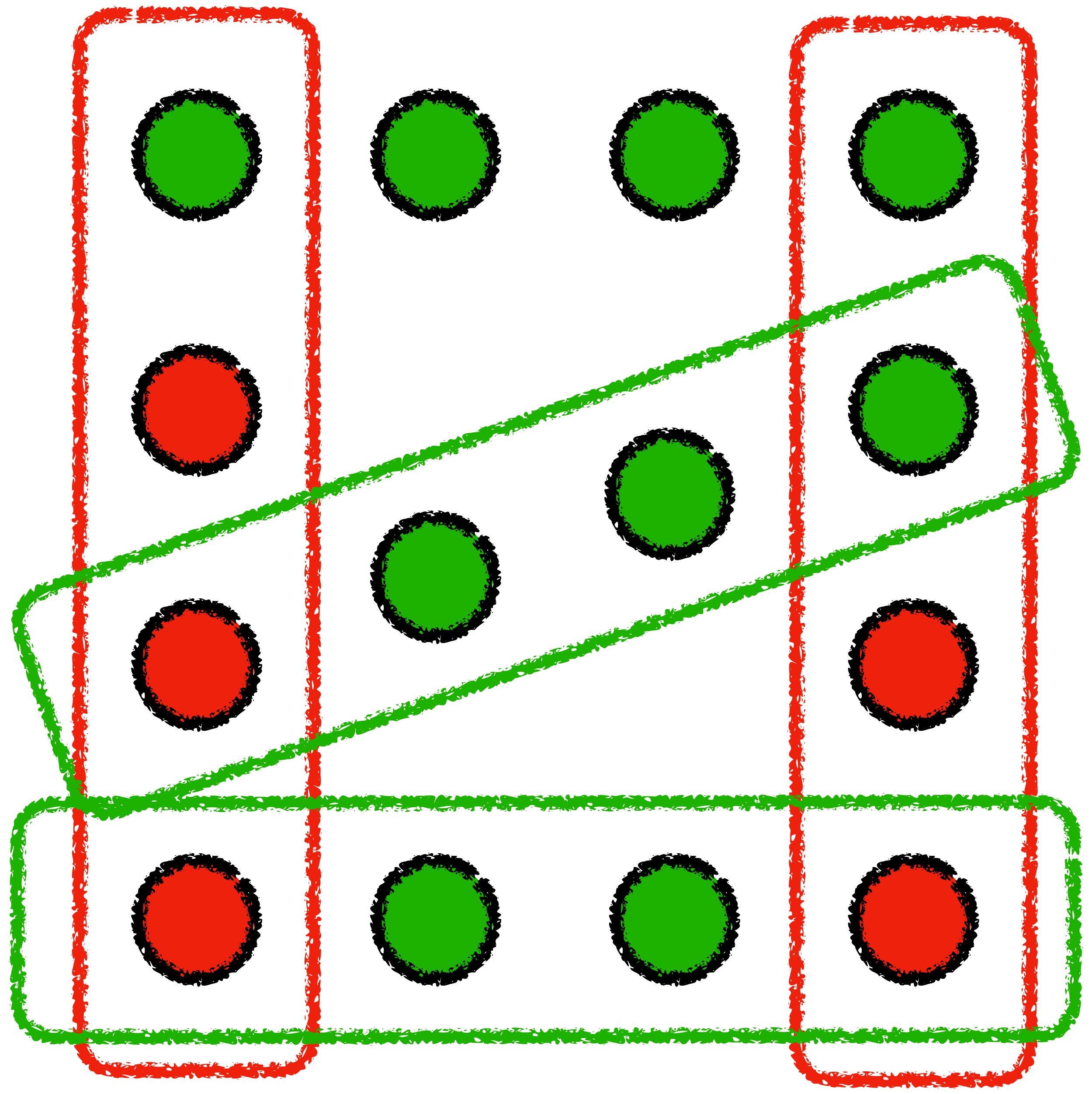
Recursive Coupling [WY '24]

$(V, \mathscr{C} \backslash \{c_0\})$ $(V, \mathscr{C})$

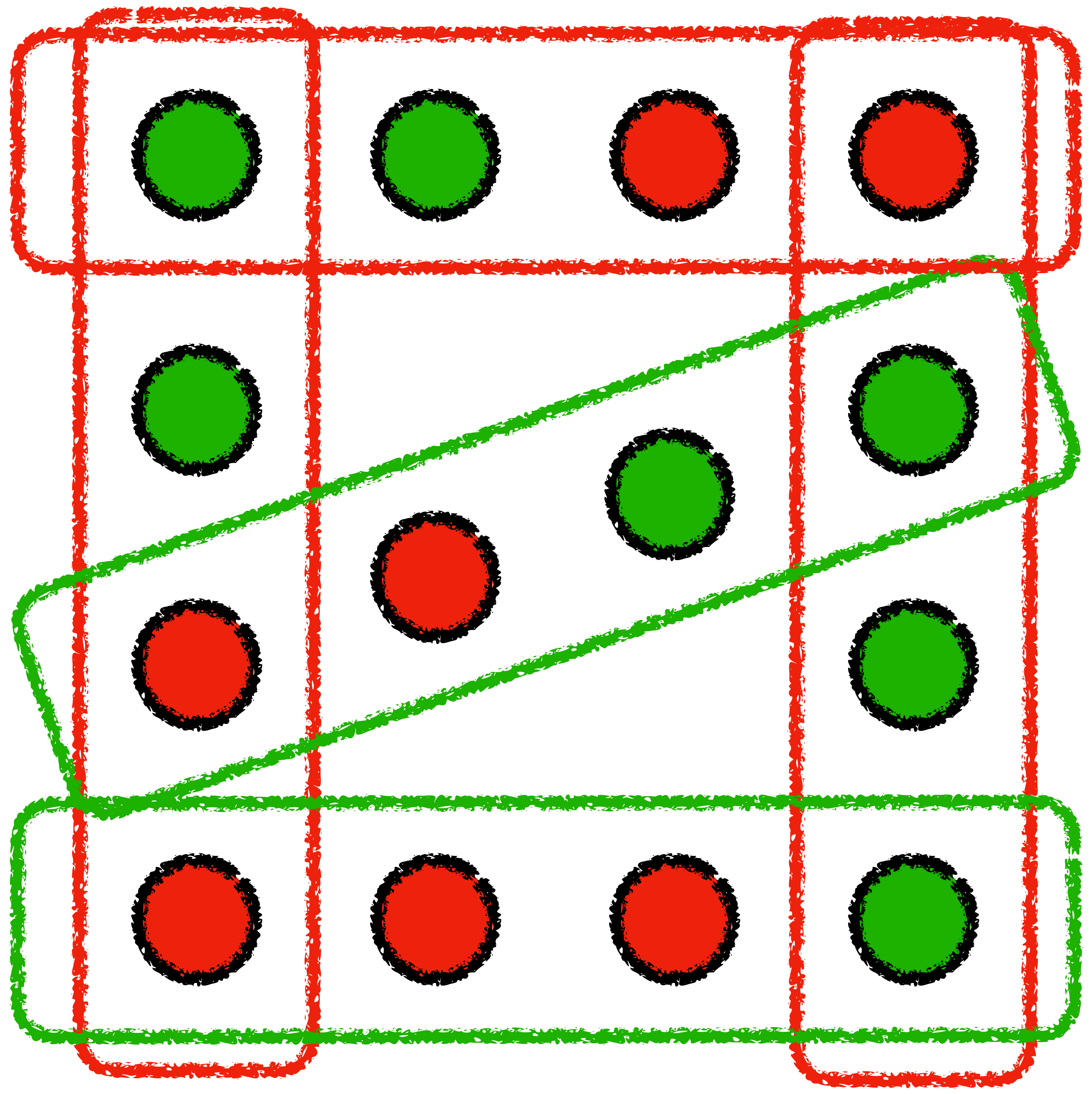
Simplify the formula, we are done if the set of clauses are the same.

Otherwise, we pick any clause in the discrepancy set and recurse!

(Original) Analysis of the Coupling



$(V, \mathscr{C} \backslash \{c_0\})$



$(V, \mathscr{C})$

All randomness by the procedure can be identified by two independent samples:

$$\mathfrak{X} \sim \mu_{\mathscr{C} \backslash \{c_0\}}, \quad \mathfrak{Y} \sim \mu_{\mathscr{C}}.$$

(Original) Analysis of the Coupling

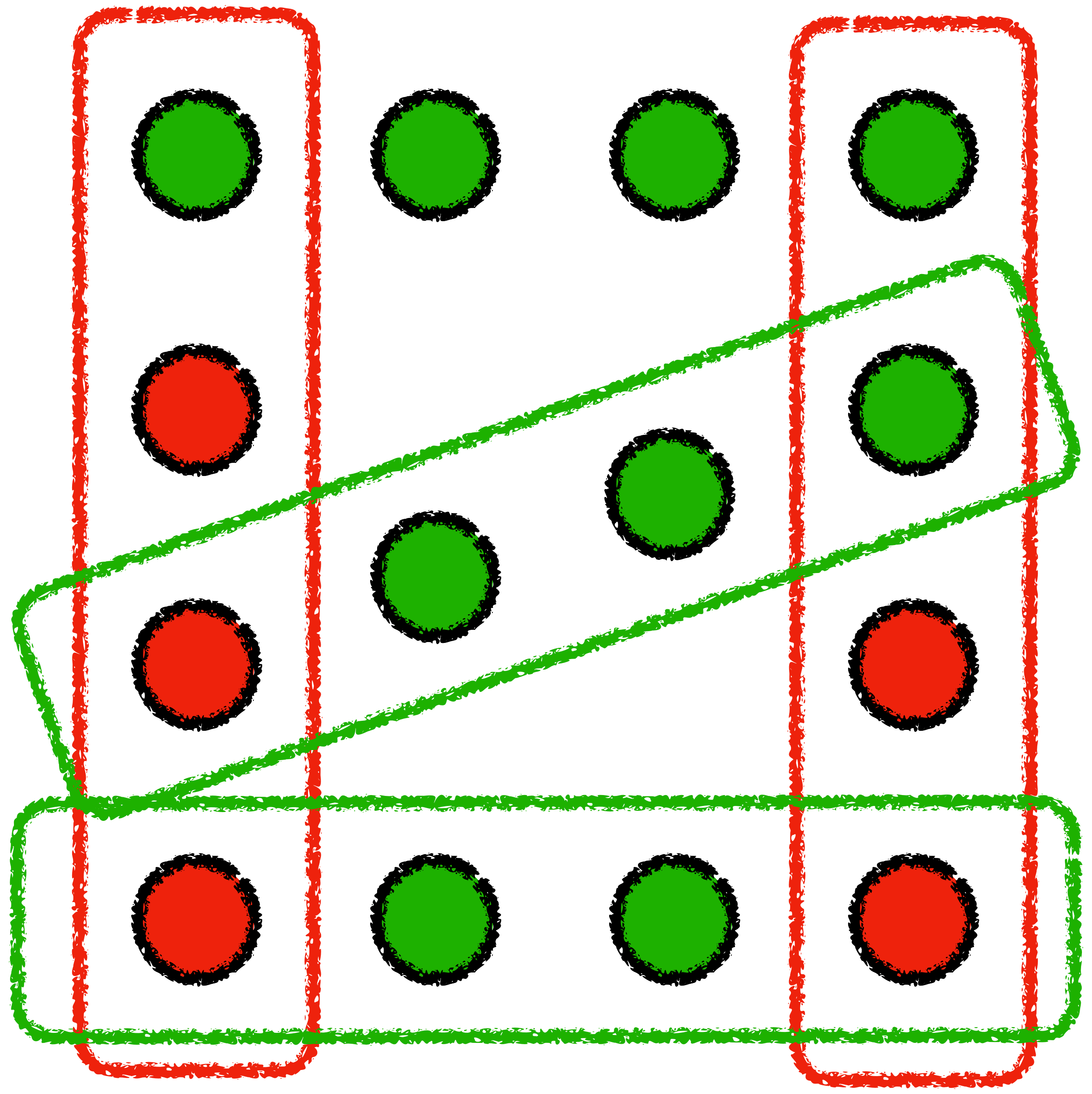
$(V, \mathscr{C} \backslash \{c_0\})$ $(V, \mathscr{C})$

All randomness by the procedure can be identified by two independent samples:

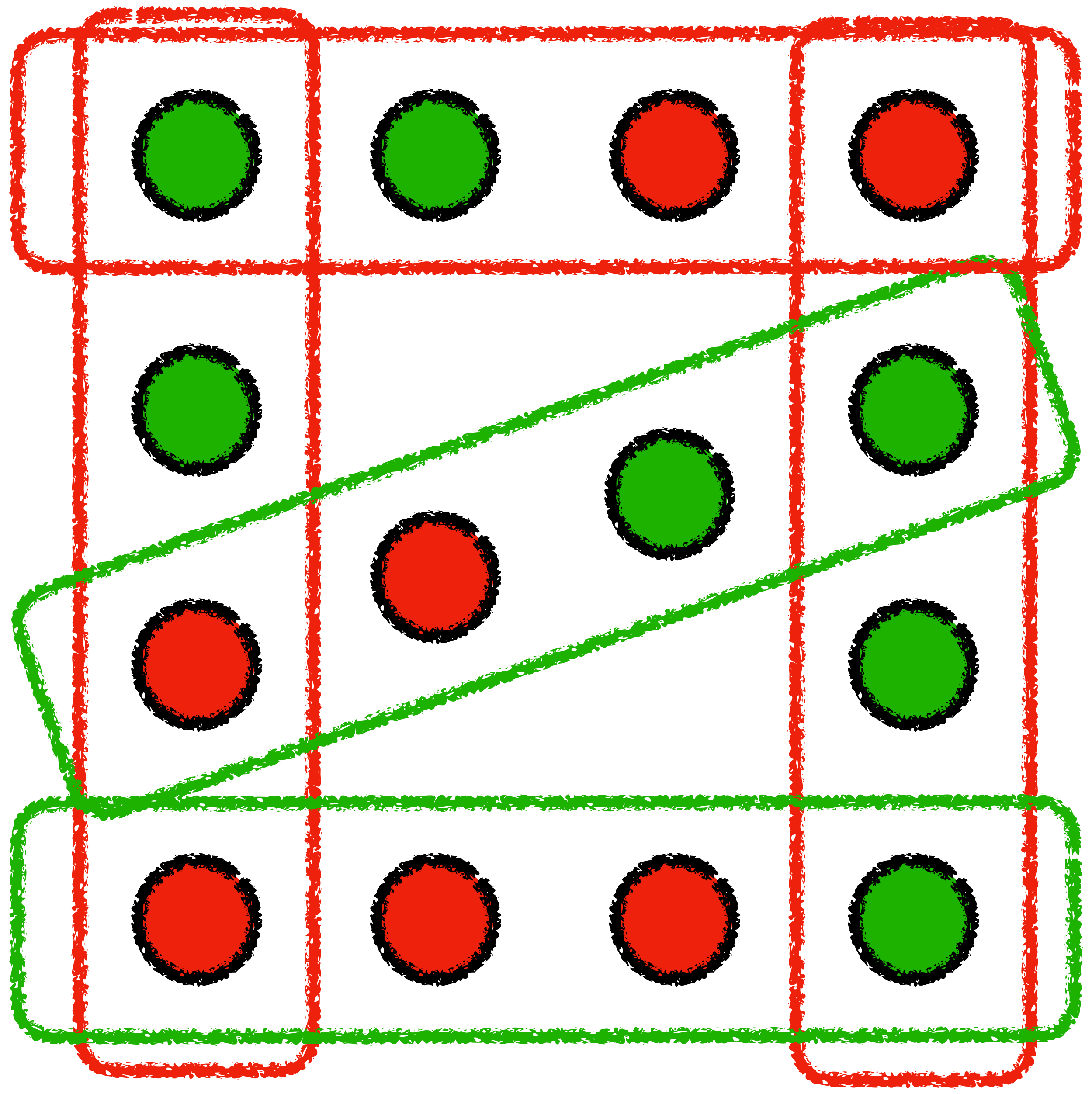
$$\mathfrak{X} \sim \mu_{\mathscr{C} \backslash \{c_0\}}, \quad \mathfrak{Y} \sim \mu_{\mathscr{C}}.$$

Sampling by marginal distribution = Revealing local information of $\mathfrak{X}$ and $\mathfrak{Y}$

(Original) Analysis of the Coupling



$(V, \mathscr{C} \backslash \{c_0\})$



$(V, \mathscr{C})$

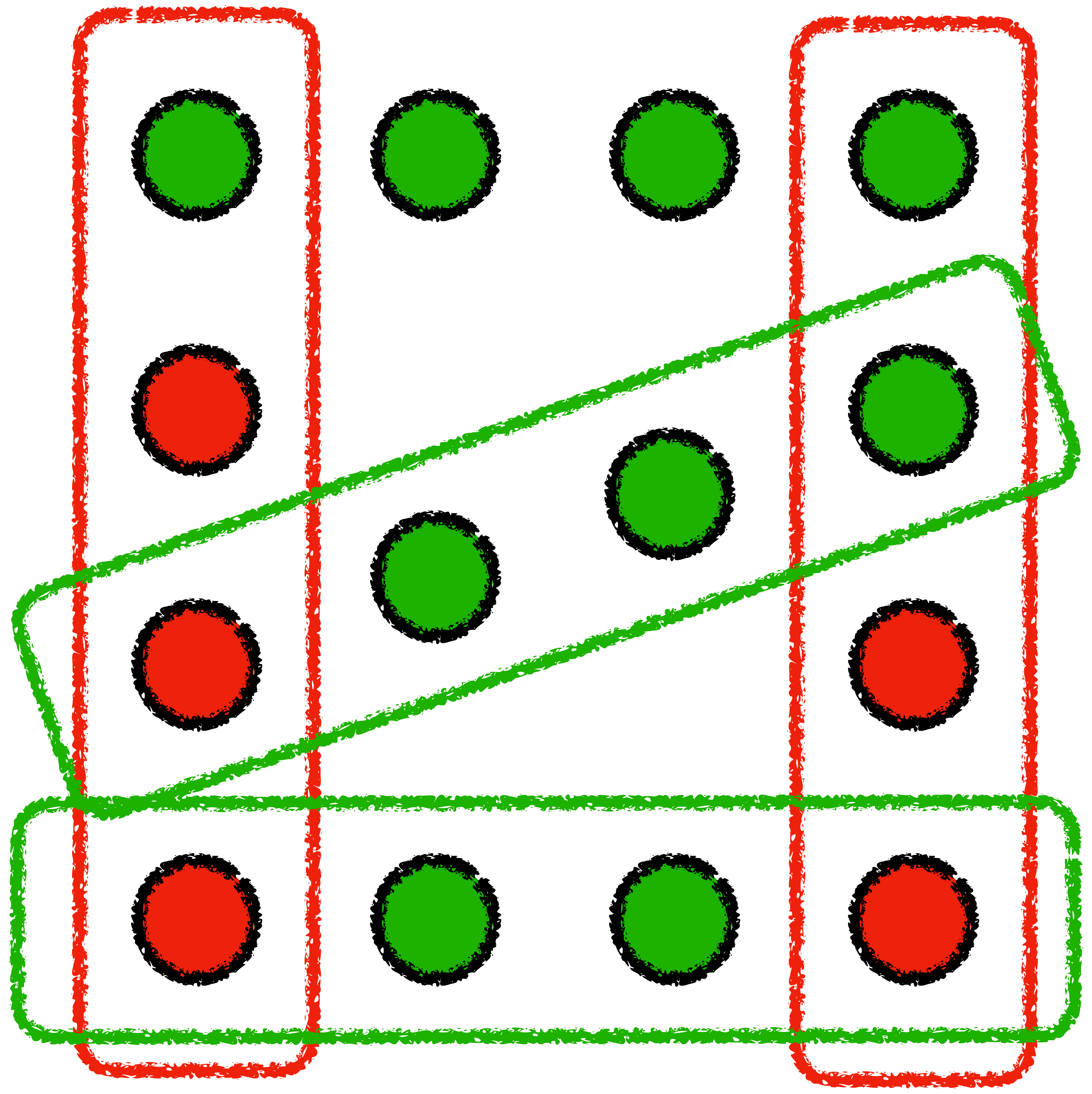
All randomness by the procedure can be identified by two independent samples:

$$\mathfrak{X} \sim \mu_{\mathscr{C} \backslash \{c_0\}}, \quad \mathfrak{Y} \sim \mu_{\mathscr{C}}\,.$$

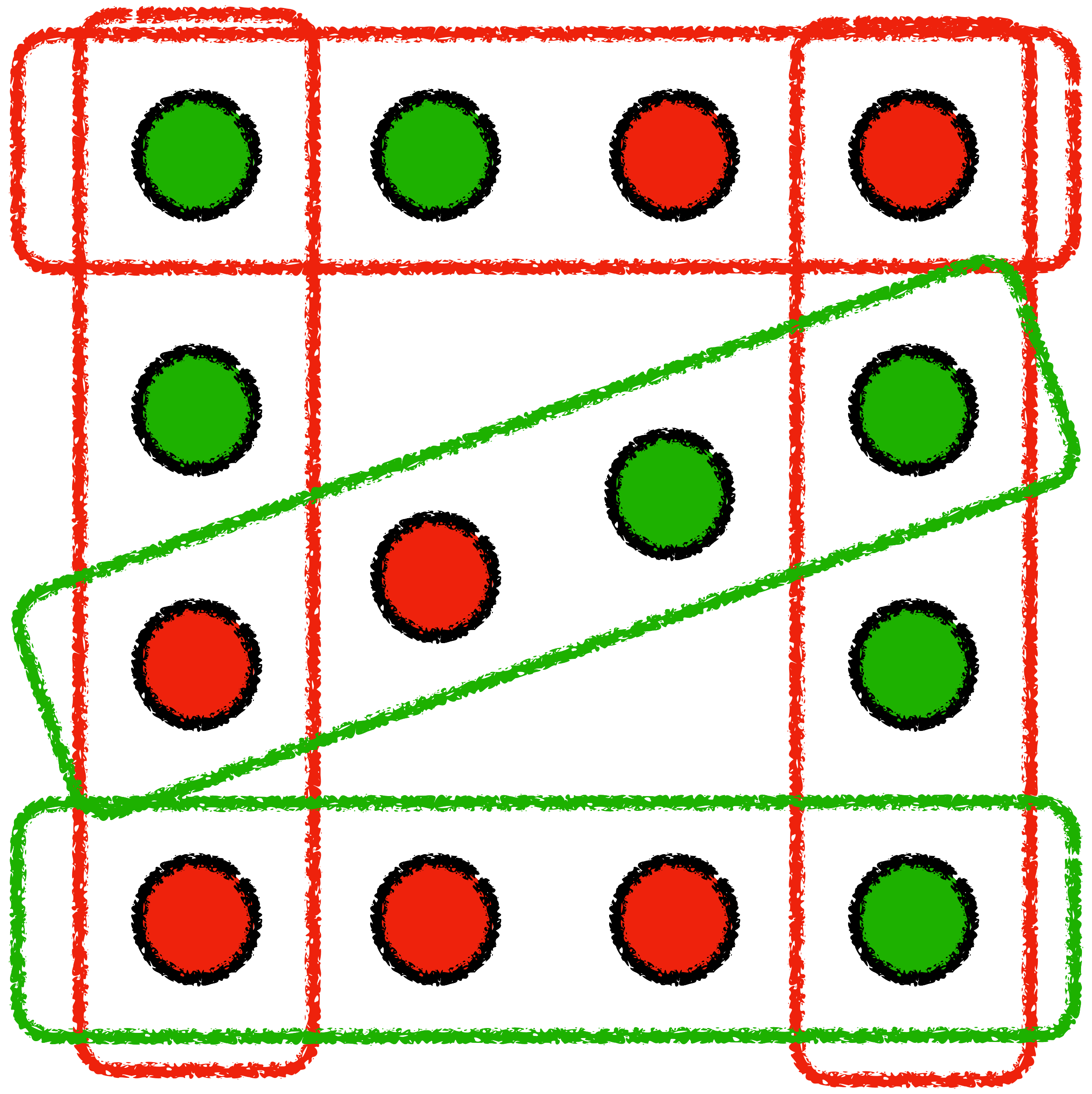
Sampling by marginal distribution = Revealing local information of $\mathfrak{X}$ and $\mathfrak{Y}$

The principle of deferred decisions!

(Original) Analysis of the Coupling



$(V, \mathscr{C} \setminus \{c_0\})$

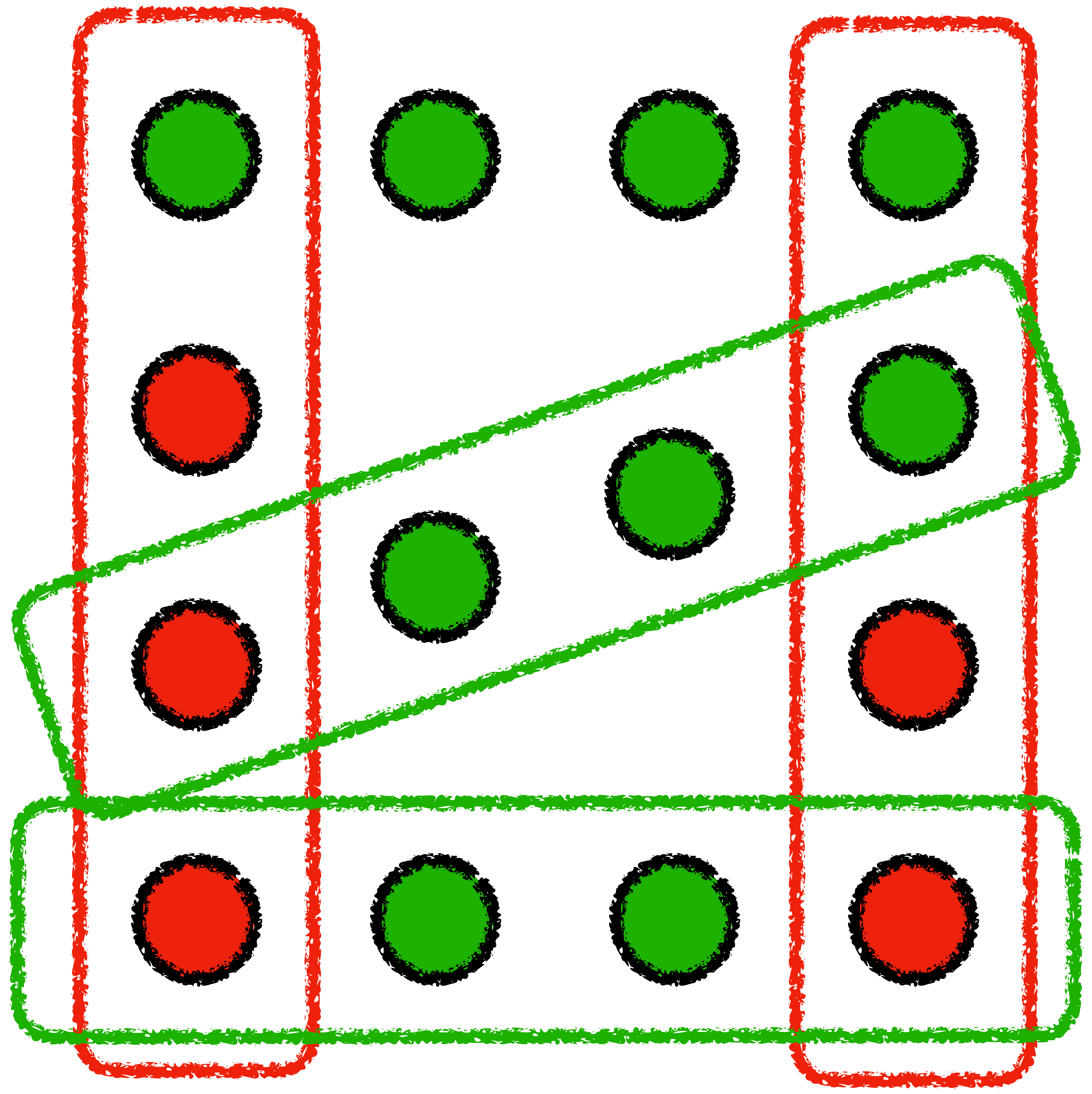


$(V, \mathscr{C})$

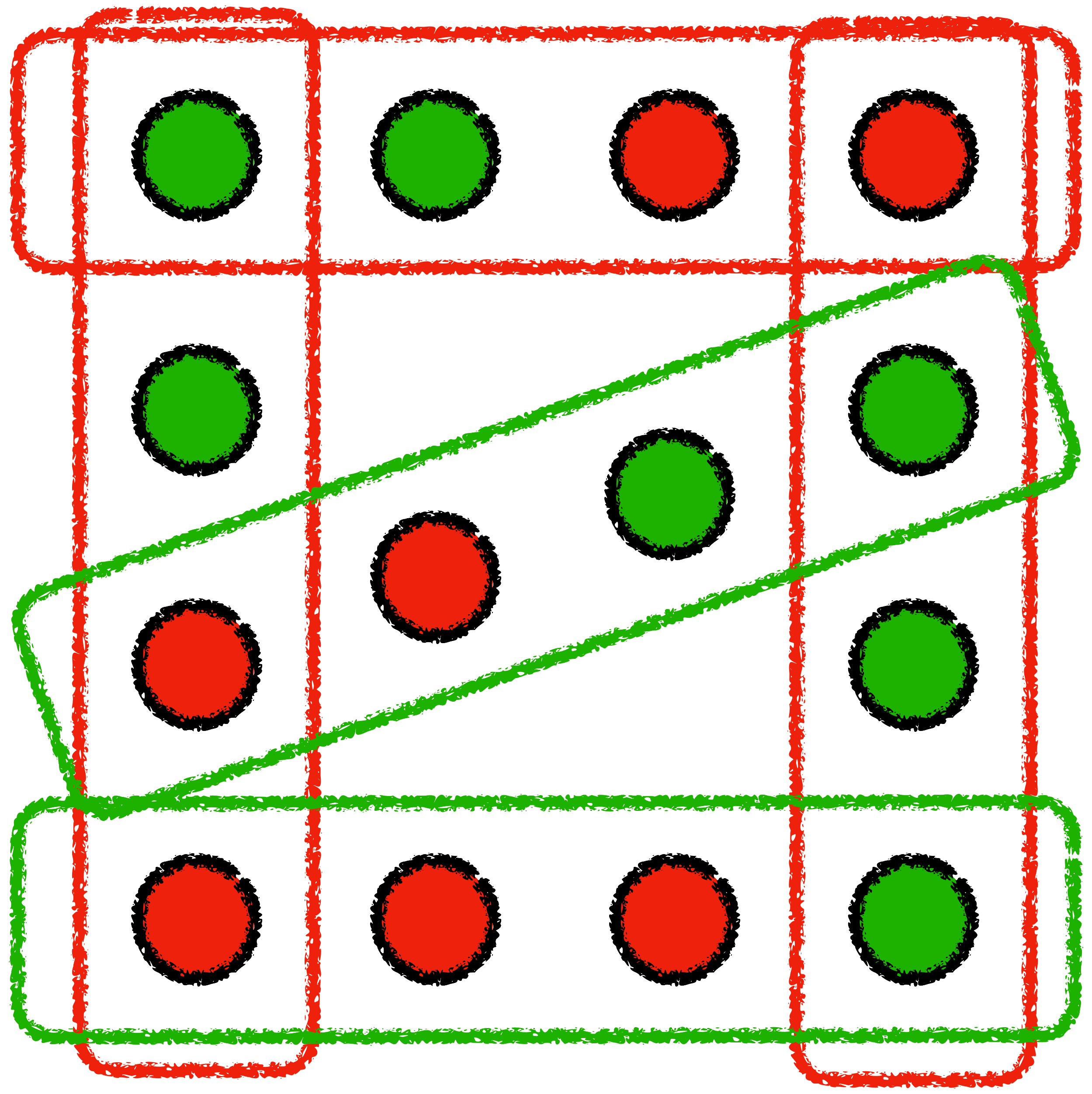
[HSS '14]: when $d < 2^k/\mathrm{e}$, a uniform random solution is locally close to uniform

witness argument: $d \lesssim 2^{k/4.82} \implies$ contraction of the coupling

Improved Analysis for Random k-SAT



$(V, \mathscr{C} \backslash \{c_0\})$

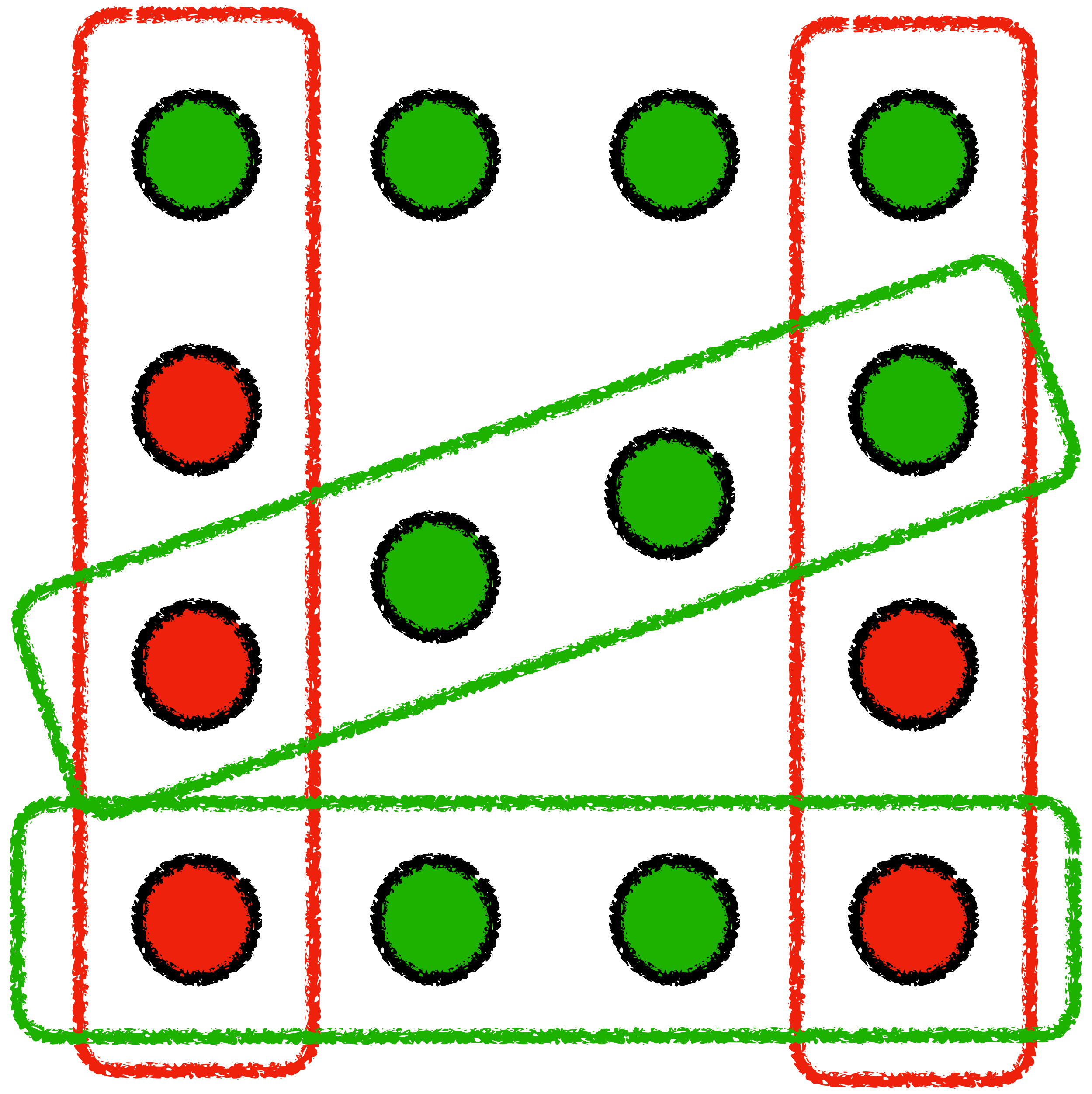


$(V, \mathscr{C})$

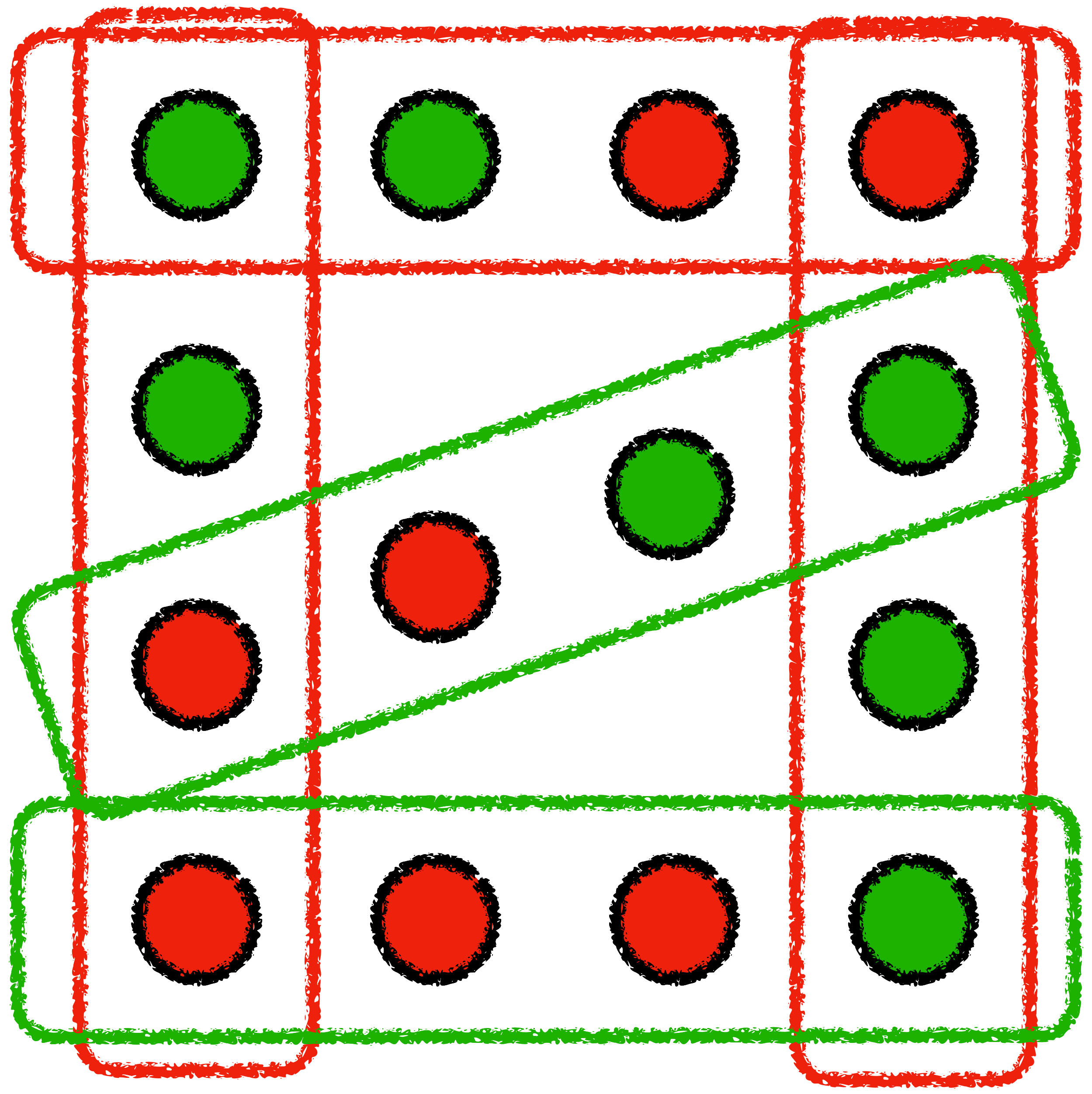
Challenges for random k-SAT:

1. Existence of high-degree variables
2. Original analysis leads to an exponent of $2 + o_q(1)$

Improved Analysis for Random k-SAT



$(V, \mathscr{C} \backslash \{c_0\})$



$(V, \mathscr{C})$

Challenges for random k-SAT:

1. Existence of high-degree variables
2. Original analysis leads to an exponent of $2 + o_q(1)$

Separating High-Degree Variables

Given degree threshold D, parameter ε and underlying hyper graph $H_\Phi = (V, \mathscr{E})$:

- Initialize $V_{\text{bad}} = \{v \in V \mid \deg(v) \geq D\}$, $\mathscr{E}_{\text{bad}} = \varnothing$;
- While $\exists e \in \mathscr{E} \backslash \mathscr{E}_{\text{bad}}$ s.t. $|e \cap V_{\text{bad}}| > (1-\varepsilon)k$:
  - update $\mathscr{E}_{\text{bad}} \leftarrow \mathscr{E}_{\text{bad}} \cup \{e\}$, $V_{\text{bad}} \leftarrow V_{\text{bad}} \cup \{e\}$

[GGGY' 21, HWY '23]: when $D = \text{poly}(k) \cdot \alpha$ and $\varepsilon = O(1/k)$, the "bad" variables are well-behaved with high probability:

- Bounded number of bad vertices: $|V_{\text{bad}}| \leq 4\varepsilon^{-1}n$
- Bounded fraction of bad hyperedges: For any connected subset of hyperedges in $\text{Lin}(H_\Phi)$ with size $\ell \geq \log n$, the number of bad hyperedges is at most $O(\ell/k)$.

Separating High-Degree Variables

Given degree threshold D, parameter ε and underlying hyper graph $H_\Phi = (V, \mathscr{E})$:

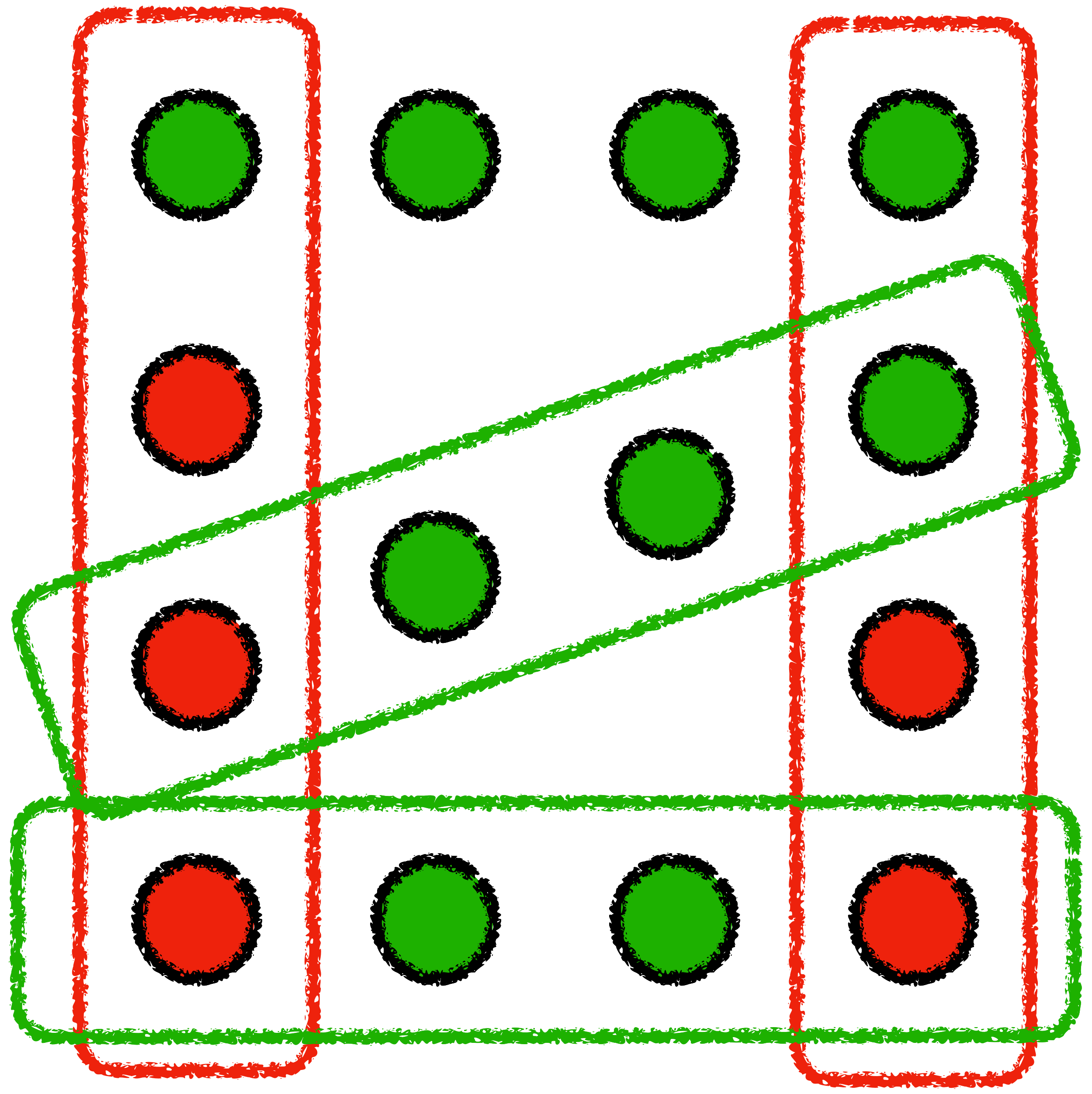
- Initialize $V_{\text{bad}} = \{v \in V \mid \deg(v) \geq D\}$, $\mathscr{E}_{\text{bad}} = \varnothing$;
- While $\exists e \in \mathscr{E} \setminus \mathscr{E}_{\text{bad}}$ s.t. $|e \cap V_{\text{bad}}| > (1 - \varepsilon)k$:
  - update $\mathscr{E}_{\text{bad}} \leftarrow \mathscr{E}_{\text{bad}} \cup \{e\}$, $V_{\text{bad}} \leftarrow V_{\text{bad}} \cup \{e\}$

[GGGY' 21, HWY '23]: when $D = \text{poly}(k) \cdot \alpha$ and $\varepsilon = O(1/k)$, the "bad" variables are well-behaved with high probability:

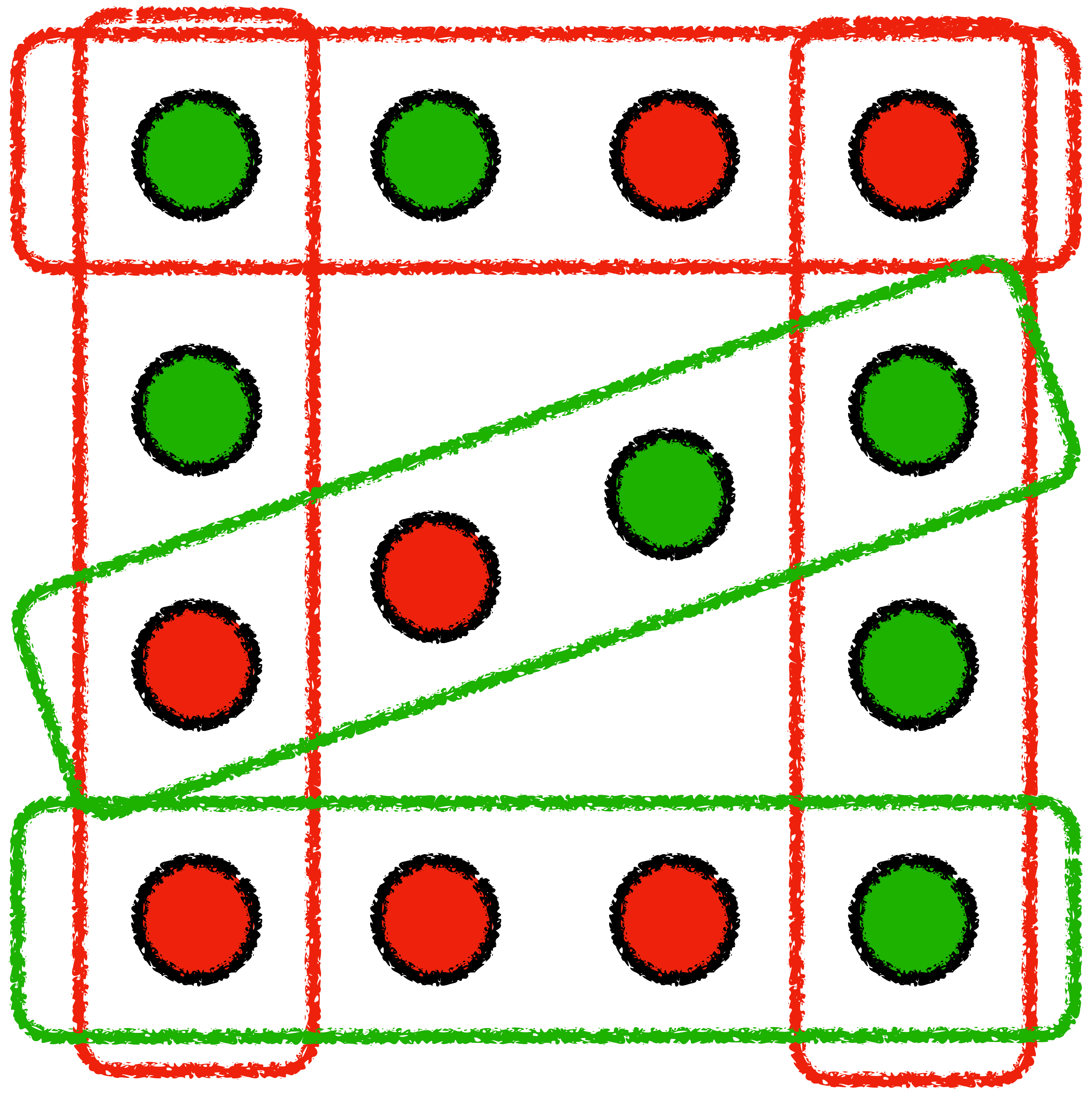
- Bounded number of bad vertices: $|V_{\text{bad}}| \leq 4\varepsilon^{-1}n$
- Bounded fraction of bad hyperedges: For any connected subset of hyperedges in $\text{Lin}(H_\Phi)$ with size $\ell \geq \log n$, the number of bad hyperedges is at most $O(\ell/k)$.

Almost reduces to the bounded-degree case!

Improved Analysis for Random k-SAT



$(V, \mathscr{C} \setminus \{c_0\})$

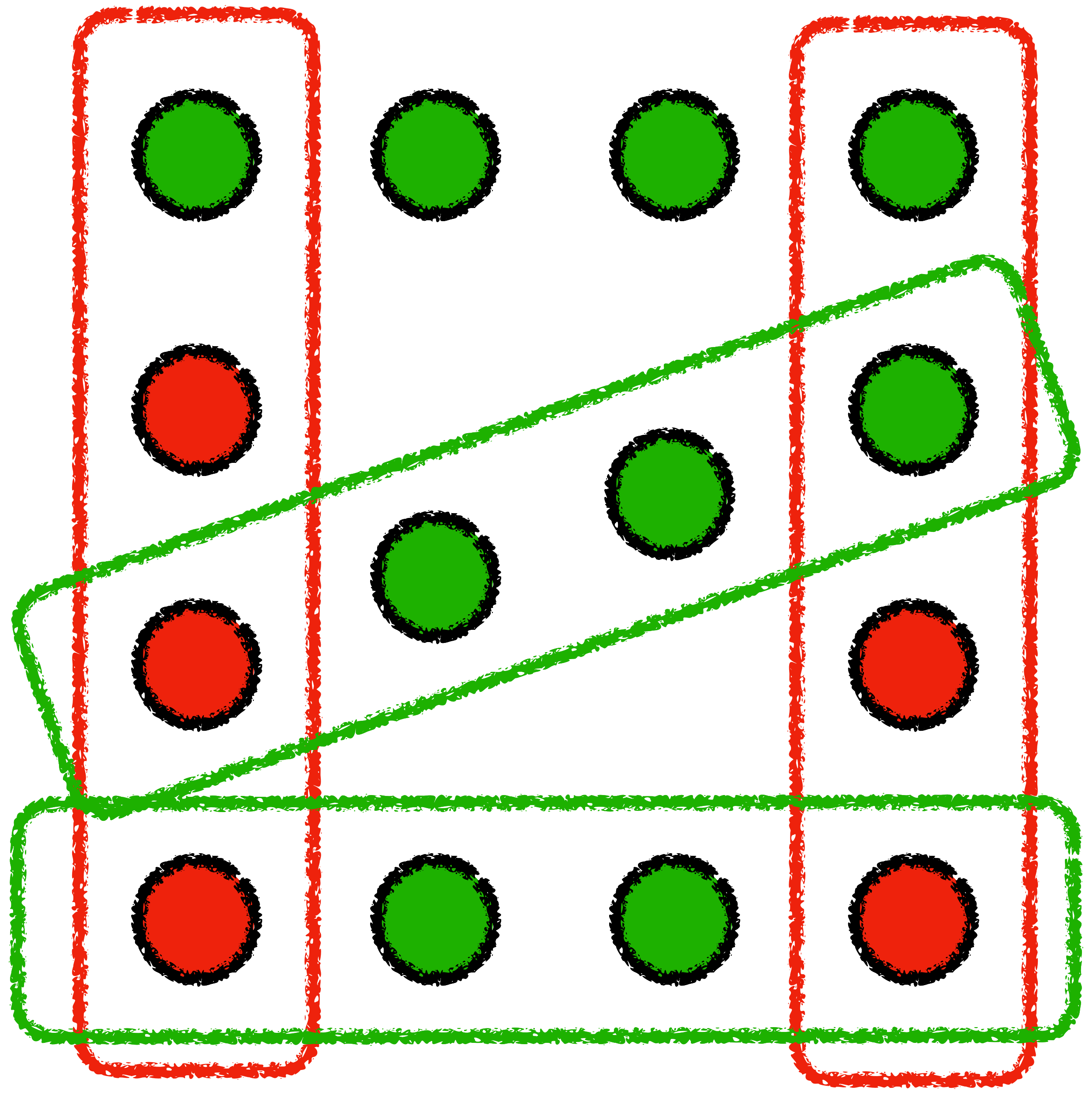


$(V, \mathscr{C})$

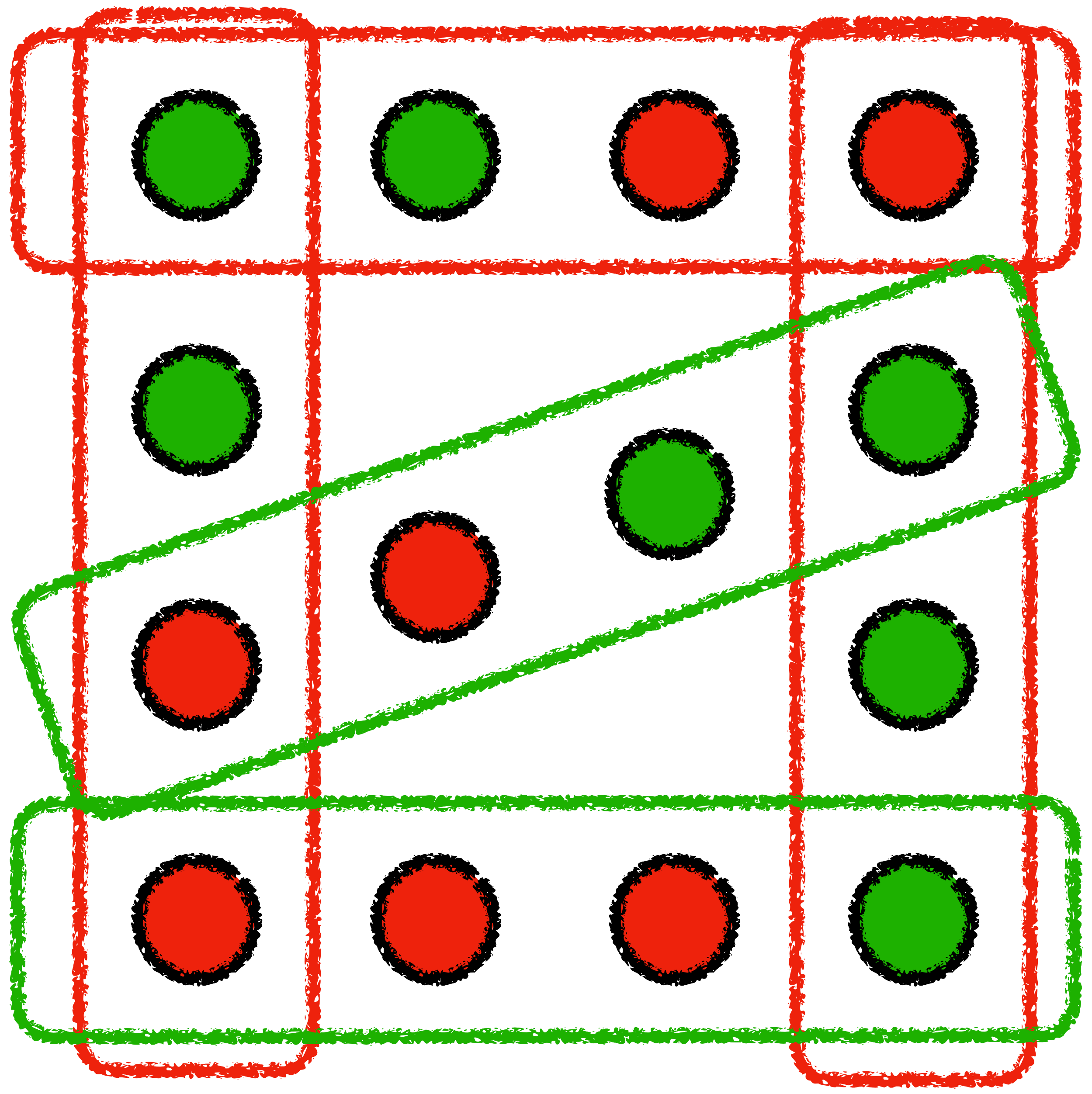
Challenges for random k-SAT:

1. Existence of high-degree variables
2. Original analysis leads to an exponent of $2 + o_q(1)$

Improved Analysis for Random k-SAT



$(V, \mathscr{C} \backslash \{c_0\})$



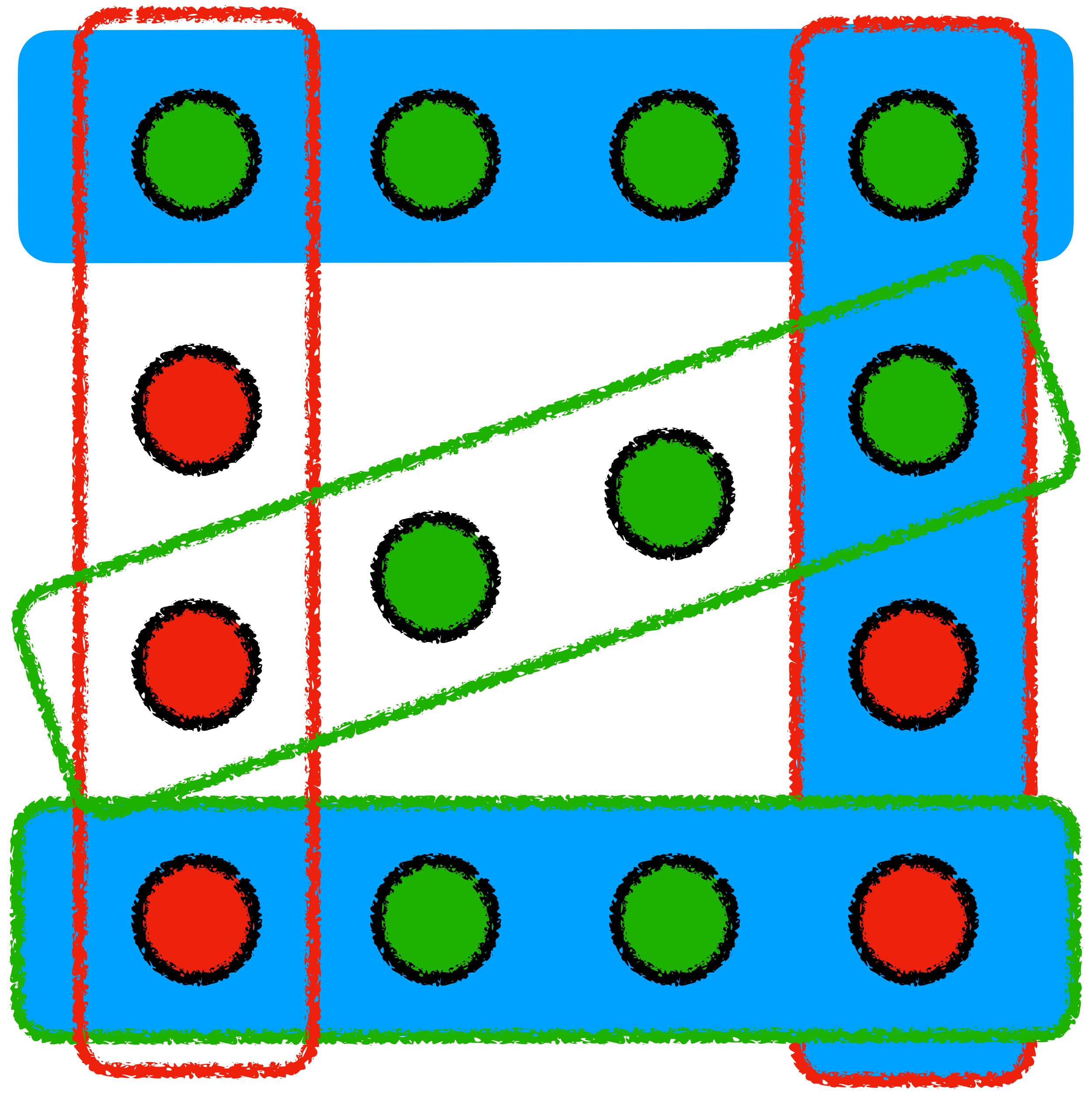
$(V, \mathscr{C})$

Challenges for random k-SAT:

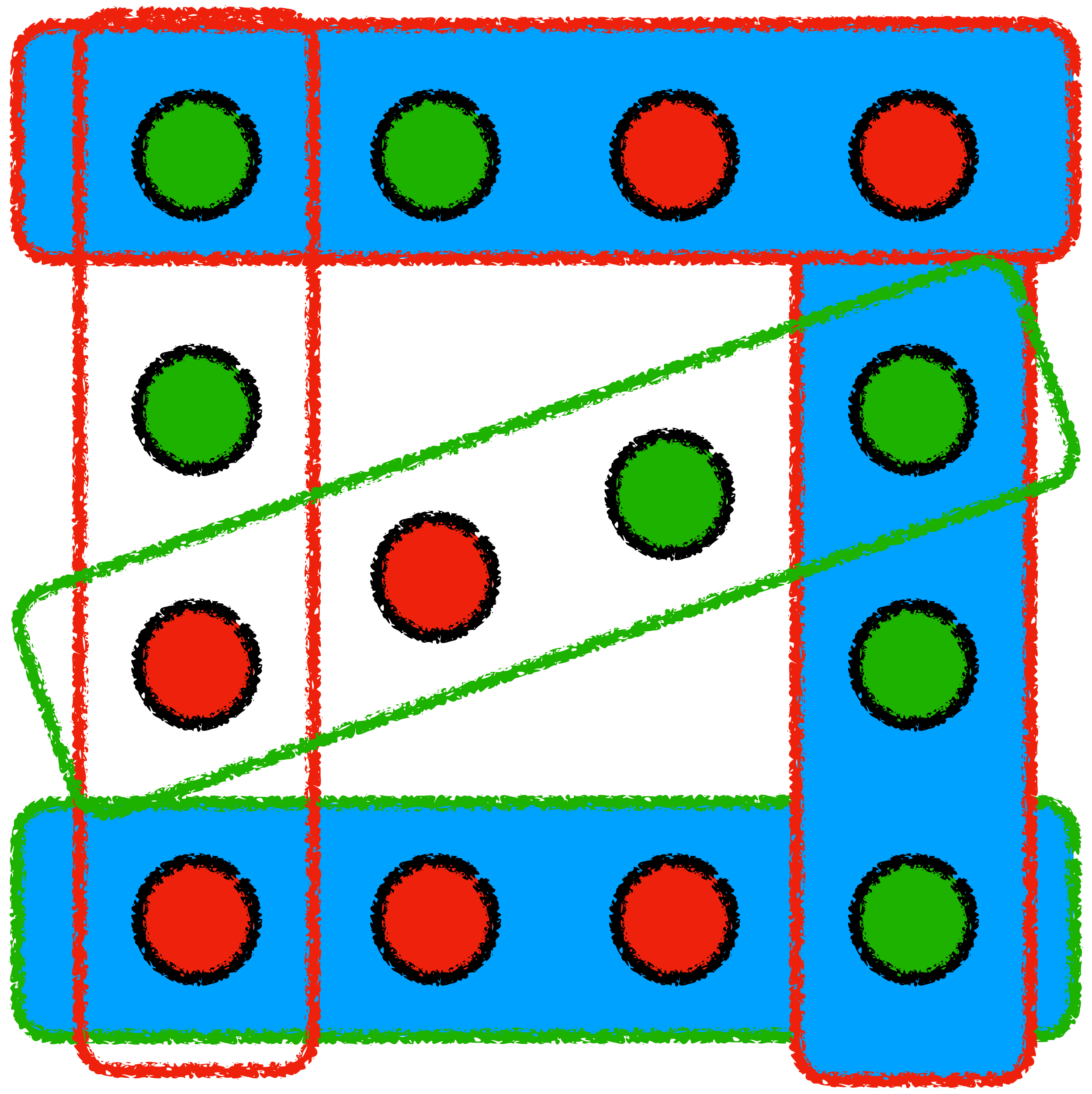
1. Existence of high-degree variables
2. Original analysis leads to an exponent of $2 + o_q(1)$

main technical contribution!

Improved Analysis for Random k-SAT



$\mathfrak{X} \sim \mu_{\mathscr{C} \setminus \{c_0\}}$



$\mathfrak{Y} \sim \mu_{\mathscr{C}}$

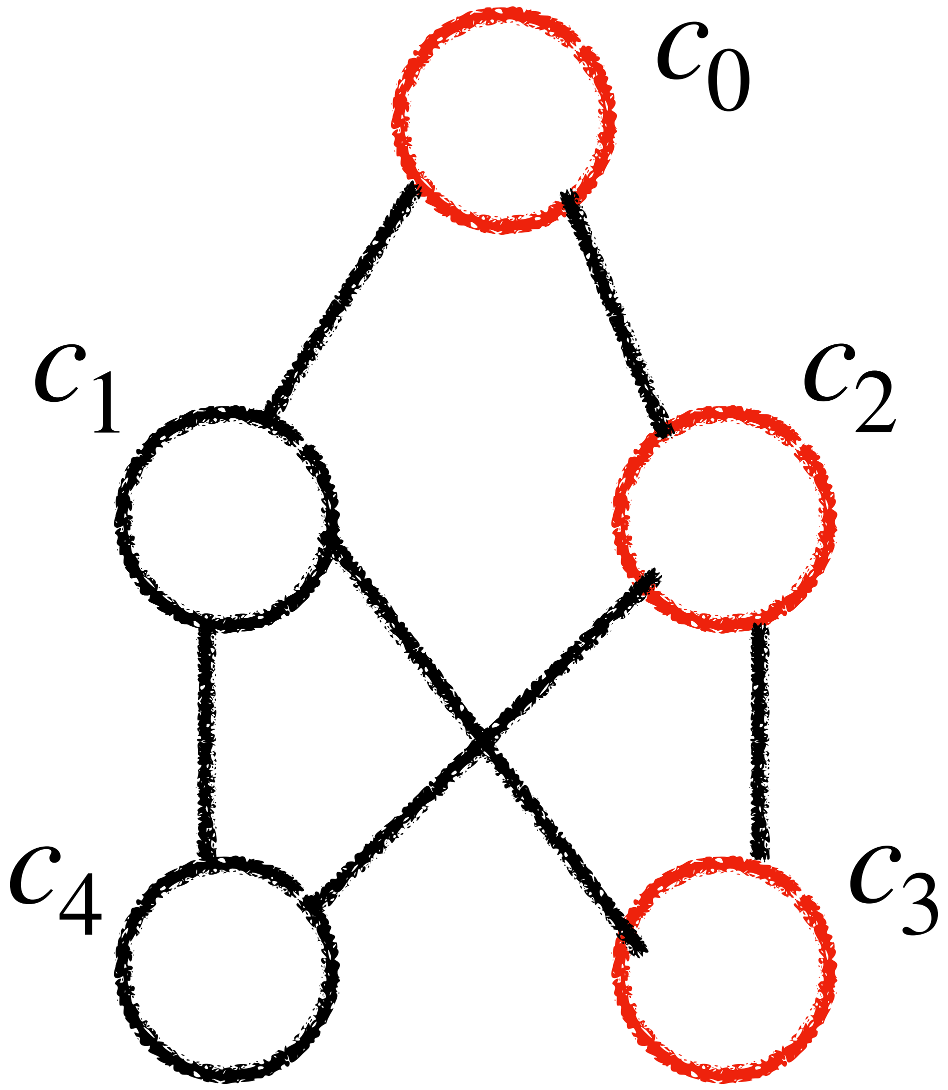
We want to bound the probability of the coupling running for too long: find a witness whose probability can be easily bounded

witness in [WY '24]: 2-tree [Alon' 91] to remove dependency

our witness: a denser witness tree [Moser, Tardos '10]

An Improved Witness

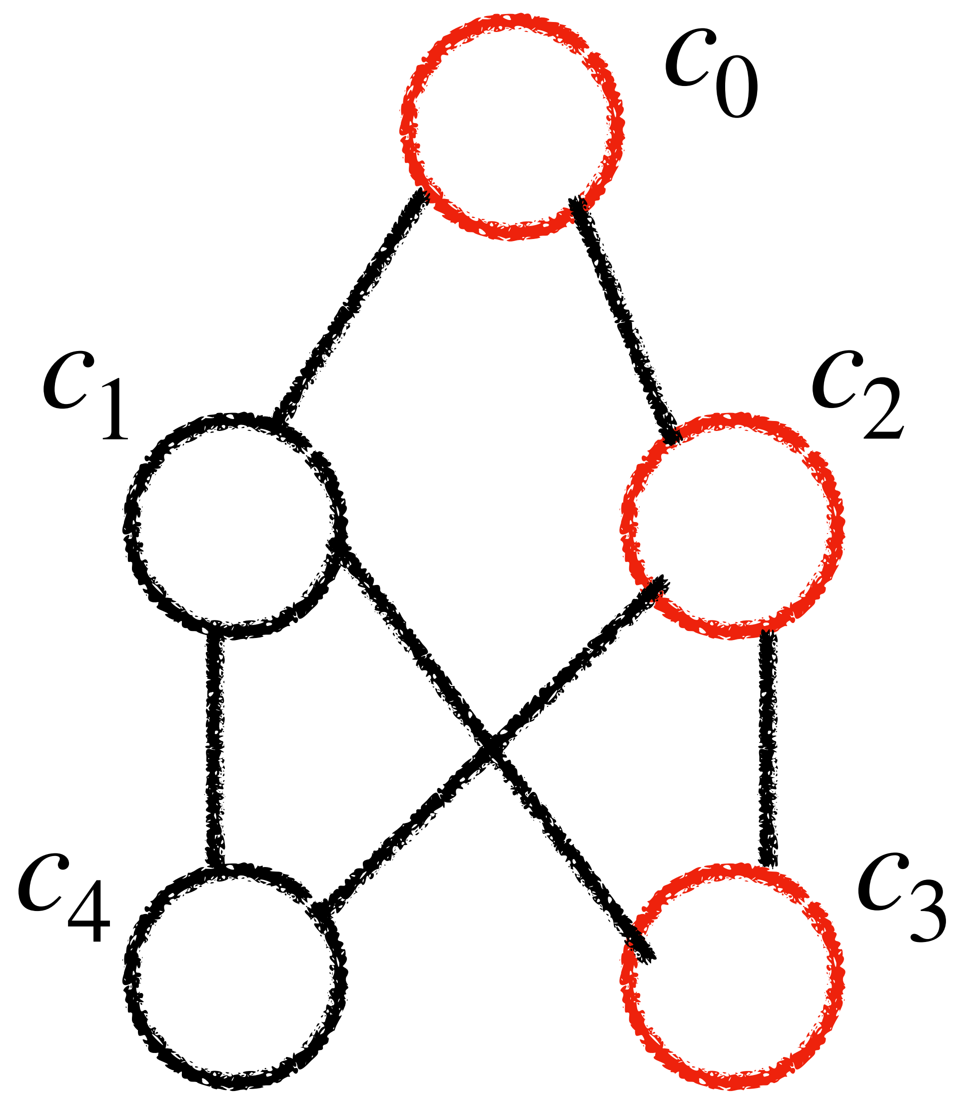
All constraints chosen in the coupling are connected in $\mathrm{Lin}(H_\Phi)$.



The coupling assigns i constraints
(connected induced subgraph)

An Improved Witness

All constraints chosen in the coupling are connected in $\mathrm{Lin}(H_\Phi)$.



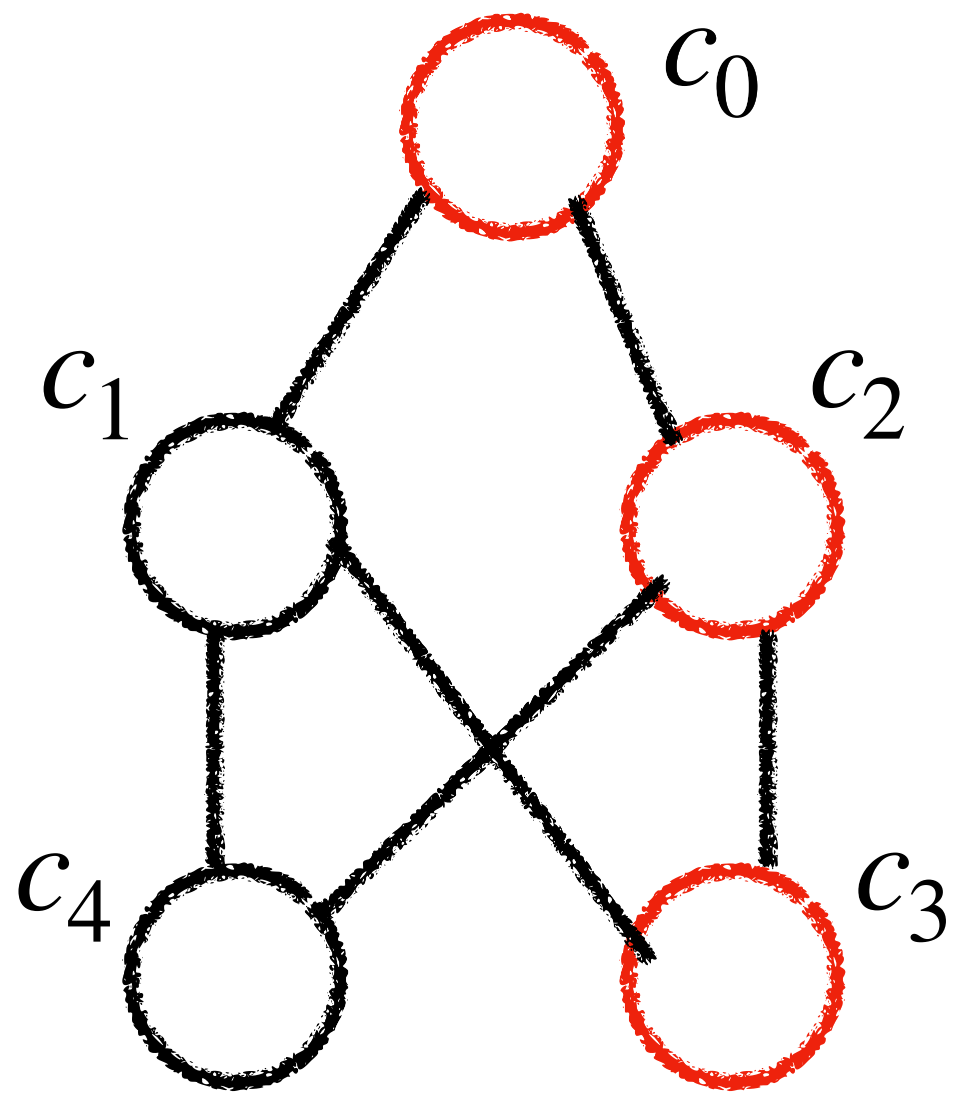
The coupling assigns i constraints
(connected induced subgraph)

2-tree: maximal independent
set connected in the square graph

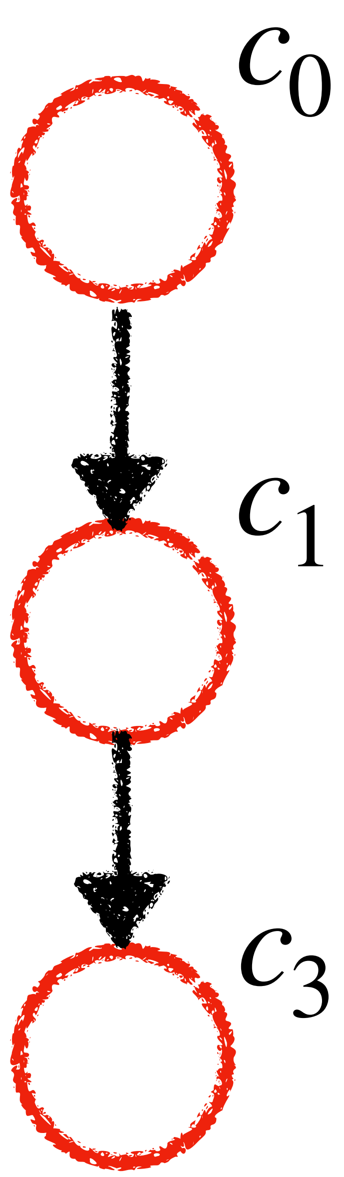
occurs with probability **at most** $(2^{-k})^{i/2}$

An Improved Witness

All constraints chosen in the coupling are connected in $\mathrm{Lin}(H_\Phi)$.



The coupling assigns i constraints
(connected induced subgraph)



2-tree: maximal independent set connected in the square graph

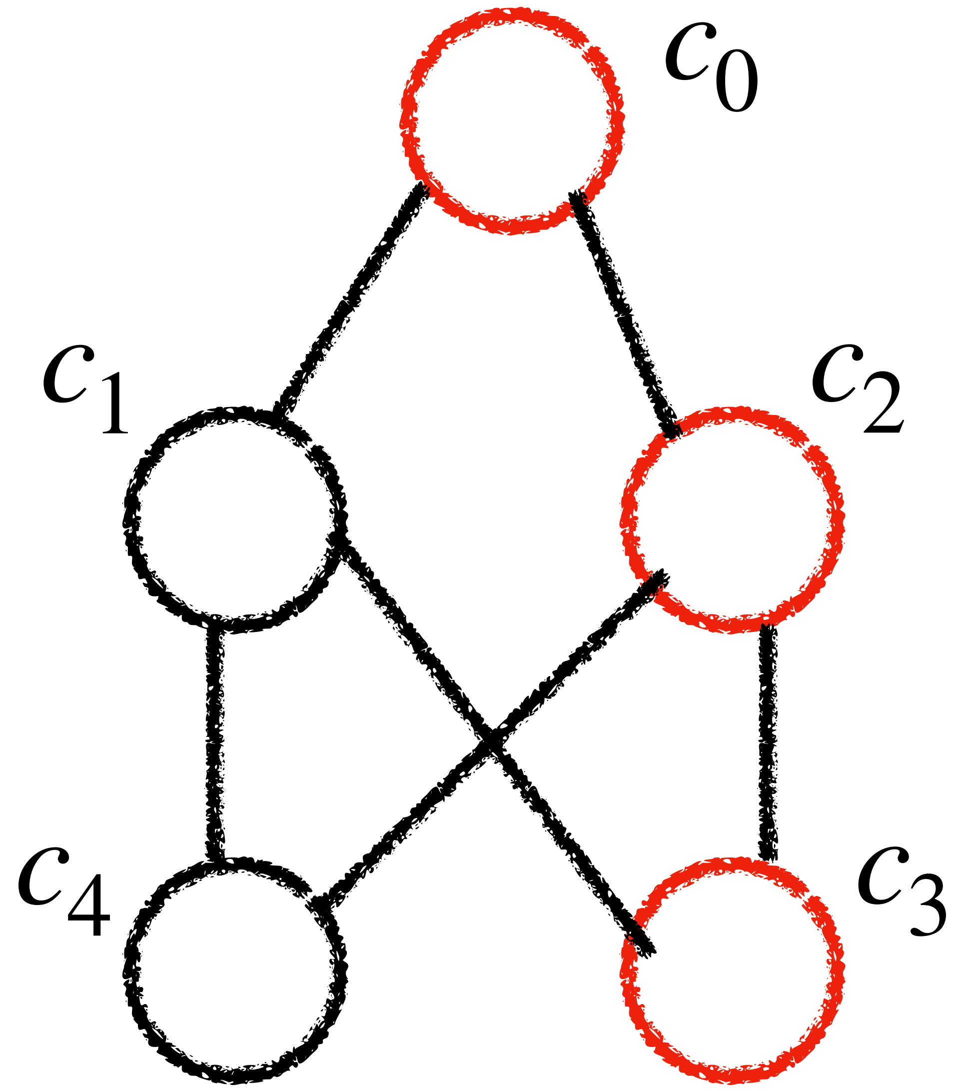
occurs with probability **at most** $(2^{-k})^{i/2}$

witness tree: a tree structure to capture A "local total ordering"

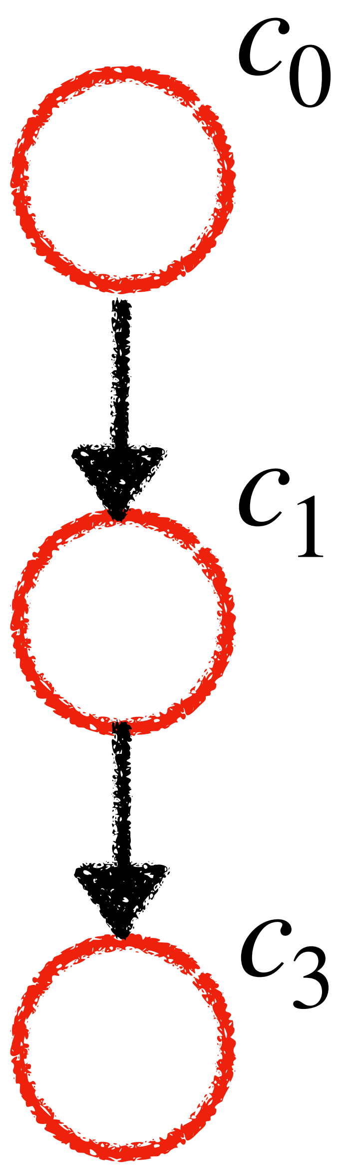
occurs with probability **near** $(2^{-k})^{i}$

An Improved Witness

All constraints chosen in the coupling are connected in $\mathrm{Lin}(H_\Phi)$.



The coupling assigns i constraints
(connected induced subgraph)



2-tree: maximal independent set connected in the square graph

occurs with probability **at most** $(2^{-k})^{i/2}$

witness tree: a tree structure to capture A "local total ordering"

occurs with probability **near** $(2^{-k})^{i}$

expansion property of random instances!

Algorithmic Implications

Algorithmic Implications

We cannot really run the coupling, but we can write down linear programs that encode coupling errors to bootstrap the marginal probability.

Algorithmic Implications

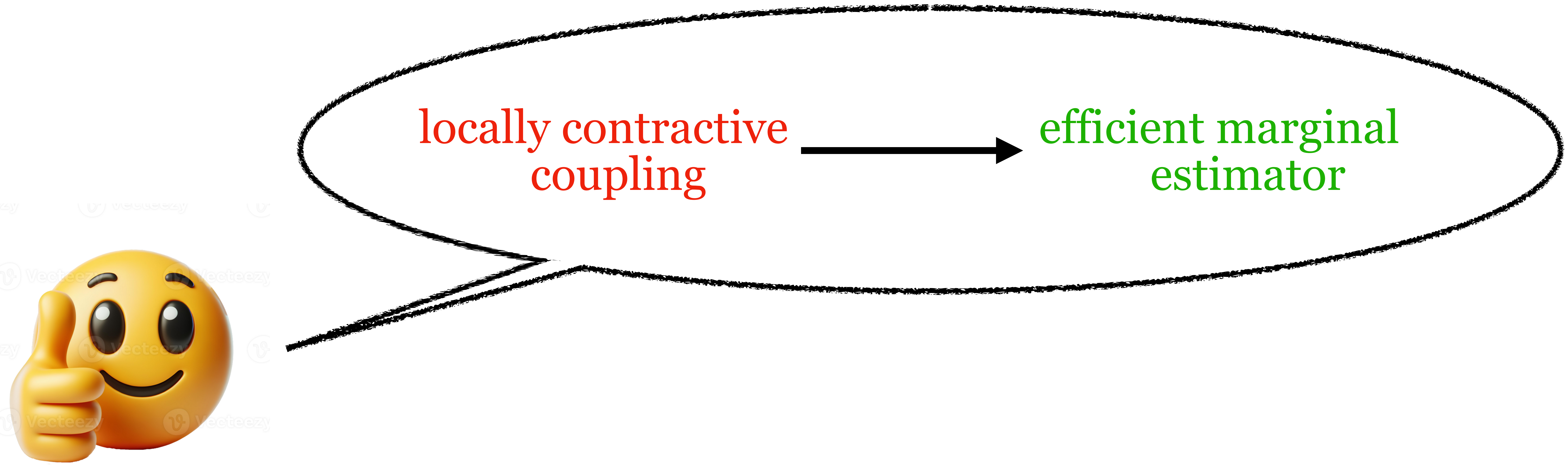
We cannot really run the coupling, but we can write down linear programs that encode coupling errors to bootstrap the marginal probability.

This method was invented by Moitra [Moi '19], applied in other works for sampling/counting bounded degree CSP solutions, [GLLZ '19, JPV '21b, WY '24], and has recently been applied to other sampling/counting settings. [HLQZ '24, CFGZZ '24]

Algorithmic Implications

We cannot really run the coupling, but we can write down linear programs that encode coupling errors to bootstrap the marginal probability.

This method was invented by Moitra [Moi '19], applied in other works for sampling/counting bounded degree CSP solutions, [GLLZ '19, JPV '21b, WY '24], and has recently been applied to other sampling/counting settings. [HLQZ '24, CFGZZ '24]



Summary

We present polynomial-time algorithms for approximate counting/almost uniform sampling random k-SAT solutions with high probability under the regime $\alpha \lesssim 2^k/\mathrm{poly}(k)$, which is near the satisfiability threshold.

Our regime bypasses the lower bound of bounded-degree k-SAT, showing that random instances are computationally easier to sample.

Our result also gives formal proofs to several correlation decay properties such as replica symmetry and non-reconstruction under the same regime.

Summary

We present polynomial-time algorithms for approximate counting/almost uniform sampling random k-SAT solutions with high probability under the regime $\alpha \lesssim 2^k/\mathrm{poly}(k)$, which is near the satisfiability threshold.

Our regime bypasses the lower bound of bounded-degree k-SAT, showing that random instances are computationally easier to sample.

Our result also gives formal proofs to several correlation decay properties such as replica symmetry and non-reconstruction under the same regime.

Open Problems

- Can we prove $O(\log n)$-connectivity under the same regime?
- What is the exact sampling threshold for random k-SAT (maybe α_{clust})?
- sampling algorithm with a faster running time? (our sampler works in $n^{\mathrm{poly}(k,\alpha,\log q)}$ time)

Summary

We present polynomial-time algorithms for approximate counting/almost uniform sampling random k-SAT solutions with high probability under the regime $\alpha \lesssim 2^k/\mathrm{poly}(k)$, which is near the satisfiability threshold.

Our regime bypasses the lower bound of bounded-degree k-SAT, showing that random instances are computationally easier to sample.

Our result also gives formal proofs to several correlation decay properties such as replica symmetry and non-reconstruction under the same regime.

Thank you!

Open Problems

- Can we prove $O(\log n)$-connectivity under the same regime?
- What is the exact sampling threshold for random k-SAT (maybe α_{clust})?
- sampling algorithm with a faster running time? (our sampler works in $n^{\mathrm{poly}(k,\alpha,\log q)}$ time)