Markov chain algorithms for bounded degree k-Sat

Heng Guo (University of Edinburgh)

Joint with Weiming Feng, Yitong Yin (Nanjing University) and Chihao Zhang (Shanghai Jiao Tong University)

LFCS lab lunch, Mar 10th, 2020

Satisfiability

One of the most important problem in computer science

Input: A formula in conjunctive normal form, like

$(x_1 \vee \overline{x_3} \vee x_5) \wedge (x_2 \vee x_3) \wedge (\overline{x_3} \vee \overline{x_4}) \wedge (x_1 \vee \overline{x_5} \vee x_6 \vee x_7) \dots$

Output: Is it satisfiable?

The first **NP**-complete problem Cook (1971) — Levin (1973)

Sampling solutions

Sometimes we are not satisfied with finding one solution. We want to generate a uniformly at random solution.

The ability of sampling solutions enables us to

- approximately count the number of solutions;
- estimate the marginal probability of individual variables;
- estimate other quantities of interest ...

And sometimes generating random instances satisfying given constraints can be useful too.

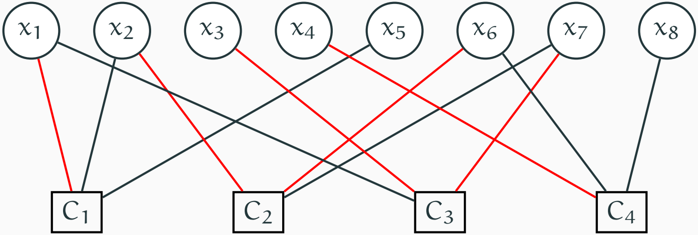
Sampling can be **NP**-hard even if finding a solution is easy (e.g. under Lovász local lemma conditions).

A natural (but not working) approach

Standard sampling approach: Glauber dynamics / Gibbs sampling

A natural (but not working) approach

Standard sampling approach: Glauber dynamics / Gibbs sampling

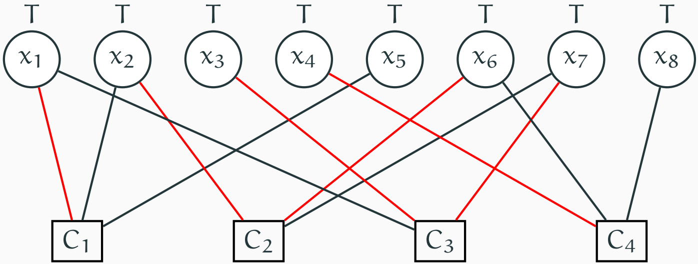


$$(\neg x_1 \vee x_2 \vee x_5) \wedge (\neg x_2 \vee \neg x_6 \vee x_7) \wedge (x_1 \vee \neg x_3 \vee \neg x_7) \wedge (\neg x_4 \vee x_6 \vee x_8)$$

Choose a random variable, sample its value conditioned on all others.

A natural (but not working) approach

Standard sampling approach: Glauber dynamics / Gibbs sampling

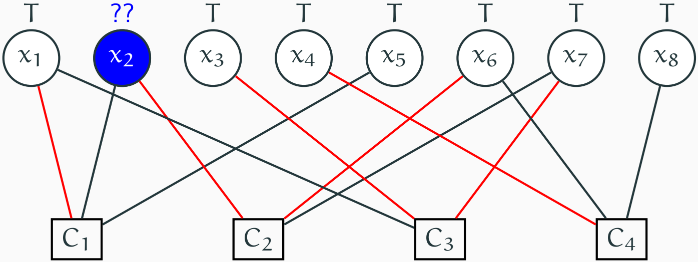


$$(\neg x_1 \vee x_2 \vee x_5) \wedge (\neg x_2 \vee \neg x_6 \vee x_7) \wedge (x_1 \vee \neg x_3 \vee \neg x_7) \wedge (\neg x_4 \vee x_6 \vee x_8)$$

Choose a random variable, sample its value conditioned on all others.

A natural (but not working) approach

Standard sampling approach: Glauber dynamics / Gibbs sampling

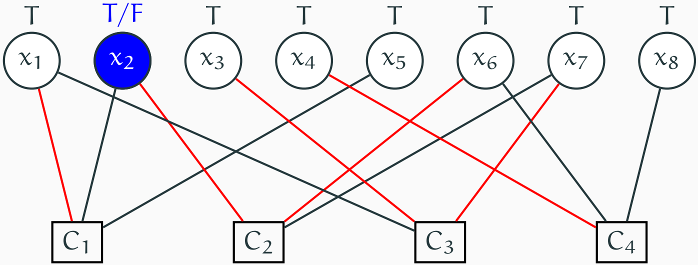


$$(\neg x_1 \vee x_2 \vee x_5) \wedge (\neg x_2 \vee \neg x_6 \vee x_7) \wedge (x_1 \vee \neg x_3 \vee \neg x_7) \wedge (\neg x_4 \vee x_6 \vee x_8)$$

Choose a random variable, sample its value conditioned on all others.

A natural (but not working) approach

Standard sampling approach: Glauber dynamics / Gibbs sampling

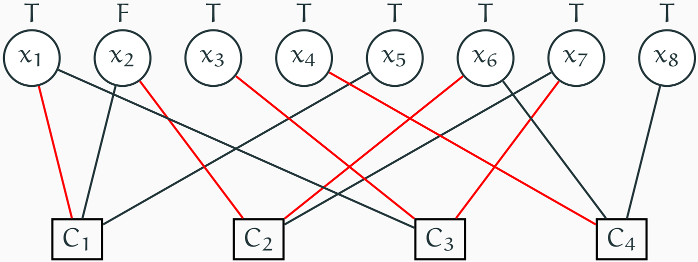


$(\neg x_1 \vee x_2 \vee x_5) \wedge (\neg x_2 \vee \neg x_6 \vee x_7) \wedge (x_1 \vee \neg x_3 \vee \neg x_7) \wedge (\neg x_4 \vee x_6 \vee x_8)$

Choose a random variable, sample its value conditioned on all others.

A natural (but not working) approach

Standard sampling approach: Glauber dynamics / Gibbs sampling

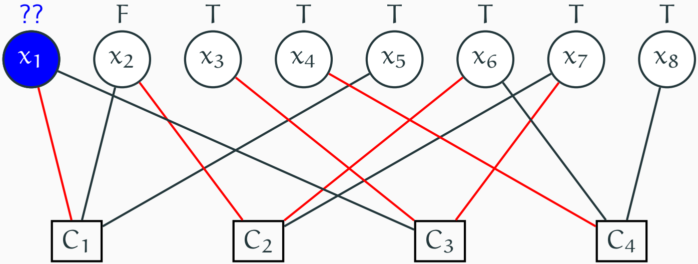


$$(\neg x_1 \vee x_2 \vee x_5) \wedge (\neg x_2 \vee \neg x_6 \vee x_7) \wedge (x_1 \vee \neg x_3 \vee \neg x_7) \wedge (\neg x_4 \vee x_6 \vee x_8)$$

Choose a random variable, sample its value conditioned on all others.

A natural (but not working) approach

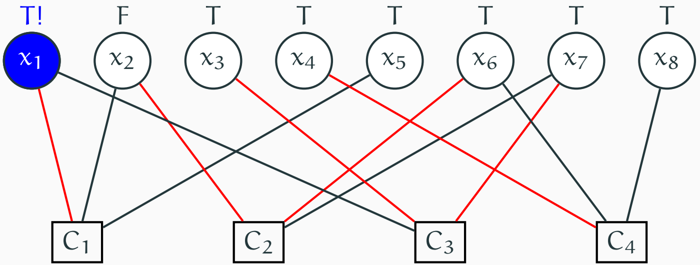
Standard sampling approach: Glauber dynamics / Gibbs sampling



$$(\neg x_1 \lor x_2 \lor x_5) \land (\neg x_2 \lor \neg x_6 \lor x_7) \land (x_1 \lor \neg x_3 \lor \neg x_7) \land (\neg x_4 \lor x_6 \lor x_8)$$

Choose a random variable, sample its value conditioned on all others.

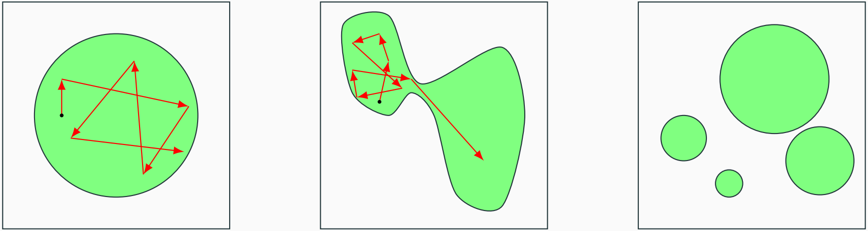
Standard sampling approach: Glauber dynamics / Gibbs sampling



$(\neg x_1 \vee x_2 \vee x_5) \wedge (\neg x_2 \vee \neg x_6 \vee x_7) \wedge (x_1 \vee \neg x_3 \vee \neg x_7) \wedge (\neg x_4 \vee x_6 \vee x_8)$

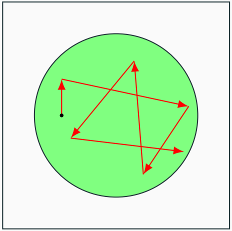
Choose a random variable, sample its value conditioned on all others.

Three scenarios for Markov chains

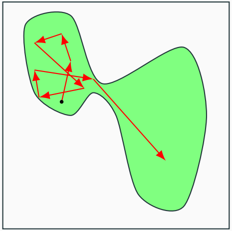


Fast mixing | Slow mixing | Not mixing!

Three scenarios for Markov chains



Fast mixing



Slow mixing

Not mixing!

Disconnectivity for k-Sat

Suppose we have k variables, and each clause contains all k variables.

$$\Phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$$

Each C_i forbids one assignment on k variables.

For example, $C_i = x_1 \vee x_2 \vee \cdots \vee x_k$ forbids the all `False` assignment.

Thus, if we forbade all assignments of Hamming weight i for some $1 \leqslant i \leqslant k-1$ (using $\binom{k}{i}$ clauses), the solution space is not connected via single variable updates.

For example, to remove Hamming weight $k-1$ assignments, we only need clauses

$$C_1 = \neg x_1 \vee x_2 \vee \dots x_k$$

$$C_2 = x_1 \vee \neg x_2 \vee \dots x_k$$

$$\vdots$$

$$C_k = x_1 \vee x_2 \vee \dots \neg x_k$$

In this example, the all `False` assignment is disconnected from the rest.

Disconnectivity for k-Sat

Suppose we have k variables, and each clause contains all k variables.

$$\Phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$$

Each C_i forbids one assignment on k variables.

For example, $C_i = x_1 \vee x_2 \vee \cdots \vee x_k$ forbids the all `False` assignment.

Thus, if we forbade all assignments of Hamming weight i for some $1 \leqslant i \leqslant k-1$ (using $\binom{k}{i}$ clauses), the solution space is not connected via single variable updates.

For example, to remove Hamming weight $k-1$ assignments, we only need clauses

$$C_1 = \neg x_1 \vee x_2 \vee \ldots x_k$$
$$C_2 = x_1 \vee \neg x_2 \vee \ldots x_k$$
$$\vdots$$
$$C_k = x_1 \vee x_2 \vee \ldots \neg x_k$$

In this example, the all `False` assignment is disconnected from the rest.

Our solution — projection

Projecting from a high dimension to a lower dimension may improve connectivity.

We will run Glauber dynamics on the projected distribution over a suitably "marked" variables.

The general problem is **NP**-hard, so we will focus on bounded degree cases.

Bounded degree k-Sat

Lovász local lemma

Theorem (Loász local lemma)

Let $E_1, \ldots, E_m$ be a set of "bad" events, such that $\Pr[E_i] \leqslant p$ for all i. Moreover, each E_i is independent from all but at most Δ events. If $ep\Delta \leqslant 1$, then

$$\Pr\left[\bigwedge_{i=1}^{m} \overline{E_i}\right] > 0.$$

In the setting of k-Sat, each clause C_i defines a bad event E_i, which is the forbidden assignment of C_i, and $p = 2^{-k}$.

If every variable appears in at most d clauses, then $\Delta \leqslant kd$.

$$\begin{aligned} ep\Delta \leqslant 1 &\Leftrightarrow e2^{-k}kd \leqslant 1 \\ &\Leftrightarrow k \geqslant \log d + \log k + C \end{aligned}$$

Theorem (Loász local lemma)

Let $E_1, \dots, E_m$ *be a set of "bad" events, such that* $\Pr[E_i] \leqslant p$ *for all* i. *Moreover, each* E_i *is independent from all but at most* Δ *events. If* $ep\Delta \leqslant 1$, *then*

$$\Pr\left[\bigwedge_{i=1}^{m} \overline{E_i}\right] > 0.$$

In the setting of k-Sat, each clause C_i defines a bad event E_i, which is the forbidden assignment of C_i, and $p = 2^{-k}$.

If every variable appears in at most d clauses, then $\Delta \leqslant kd$.

$$\begin{aligned} ep\Delta \leqslant 1 &\Leftrightarrow e2^{-k}kd \leqslant 1 \\ &\Leftrightarrow k \geqslant \log d + \log k + C \end{aligned}$$

Moser-Tardos algorithm

We consider k-CNF formula with variable degree at most d.

Theorem (Moser and Tardos, 2011)

If $k \geqslant \log d + \log k + C$, then we can always find a satisfying assignment in polynomial time.

The algorithm is extremely simple: assign variables u.a.r., then keep resample variables in violating clauses.

Unfortunately, sampling is substantially harder.

Theorem (Bezáková, Galanis, Goldberg, G. and Štefankovič, 2016)

*If $k \leqslant 2 \log d + C$, then sampling satisfying assignments is **NP**-hard, even if there is no negation in the formula (monotone case).*

We consider k-CNF formula with variable degree at most d.

Theorem (Moser and Tardos, 2011)

If $k \geqslant \log d + \log k + C$, then we can always find a satisfying assignment in polynomial time.

The algorithm is extremely simple: assign variables u.a.r., then keep resample variables in violating clauses.

Unfortunately, sampling is substantially harder.

Theorem (Bezáková, Galanis, Goldberg, G. and Štefankovič, 2016)

If $k \leqslant 2 \log d + C$, then sampling satisfying assignments is **NP***-hard, even if there is no negation in the formula (monotone case).*

Open problem:

Is there an efficient algorithm to sample satisfying assignments of k-Sat given $k \gtrsim 2 \log d + C$?

Results

Hermon, Sly and Zhang (2016) Glauber dynamics mixes in $O(n \log n)$ time if $k \geqslant 2 \log d + C$ and there is no negation (monotone formula).

G., Jerrum and Liu (2016) "Partial rejection sampling" terminates in $O(n)$ time if $k \geqslant 2 \log d + C$ and there is no small intersection.

Moitra (2016) An "exotic" deterministic algorithm in $n^{O(k^2 d^2)}$ time if $k \geqslant 60(\log d + \log k) + 300$.

Theorem (Our result)

We give a Markov chain based algorithm in $\widetilde{O}(n^{1+\delta} k^3 d^2)$ *time if* $k \geqslant 20(\log d + \log k) + \log \delta^{-1}$ *where* $\delta \leqslant 1/60$ *is an arbitrary constant.*

Results

Hermon, Sly and Zhang (2016) Glauber dynamics mixes in $O(n \log n)$ time if $k \geqslant 2 \log d + C$ and there is no negation (monotone formula).

G., Jerrum and Liu (2016) "Partial rejection sampling" terminates in $O(n)$ time if $k \geqslant 2 \log d + C$ and there is no small intersection.

Moitra (2016) An "exotic" deterministic algorithm in $n^{O(k^2 d^2)}$ time if $k \geqslant 60(\log d + \log k) + 300$.

Theorem (Our result)

We give a Markov chain based algorithm in $\widetilde{O}(n^{1+\delta} k^3 d^2)$ *time if* $k \geqslant 20(\log d + \log k) + \log \delta^{-1}$ *where* $\delta \leqslant 1/60$ *is an arbitrary constant.*

Our algorithm

The algorithm

Goal: to sample from the uniform distribution μ over satisfying assignments

1. Mark a set M of variables;
2. Run Glauber dynamics on the projected distribution μ_M for $O(n \log n)$ steps. This yields an (approximate) sample $\sigma_M \sim \mu_M$;
3. Use rejection sampling to sample from $\sigma_{V \setminus M}$;
4. Output $\sigma_M \cup \sigma_{V \setminus M}$.

A set M of variables are marked so that:

1. for any clause C_i, $|C_i \cap M| \gtrsim 0.11k$;
2. for any clause C_i, $|C_i \setminus M| \gtrsim 0.51k$;

The existence of M is guaranteed by the local lemma, and M can be found by the Moser-Tardos algorithm in linear time.

Two sides of the marking

If $|C_i \cap M|$ is large, then all components are small.

Lemma

For almost all $\sigma \in \{0,1\}^M$, $V \setminus M$ *scatters into connected components of size* $O(\mathrm{poly}(dk)\log n)$.

If $|C_i \setminus M|$ is large, then all variables are close to the uniform distribution.

Lemma

Conditioned on any assignment of M, *for any* $v \in V \setminus M$,

$$\left| \Pr_{\sigma \sim \mu_{V \setminus M}} [\sigma(v) = 1] - \frac{1}{2} \right| \leqslant \exp(-O(k)).$$

So the marking is to balance these two effects.

What to prove

1. The Glauber dynamics on the marked variables is rapidly mixing;
2. The Glauber dynamics on the marked variables can be implemented efficiently;
3. The rejection sampling step in the end terminates quickly.

Item (1) is shown using the path coupling method.

Items (2) and (3) are shown together. In particular, the Glauber dynamics is implemented using rejection sampling.

Implementing the Glauber dynamics

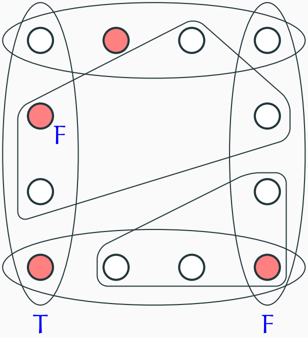
Glauber dynamics: compute the marginal probability of a variable conditioned on all other marked variables, which defines a smaller instance. (#**P**-hard in general.)

We approximately implement this by using rejection sampling on

1. all unmarked variables, and
2. the variable to be updated.

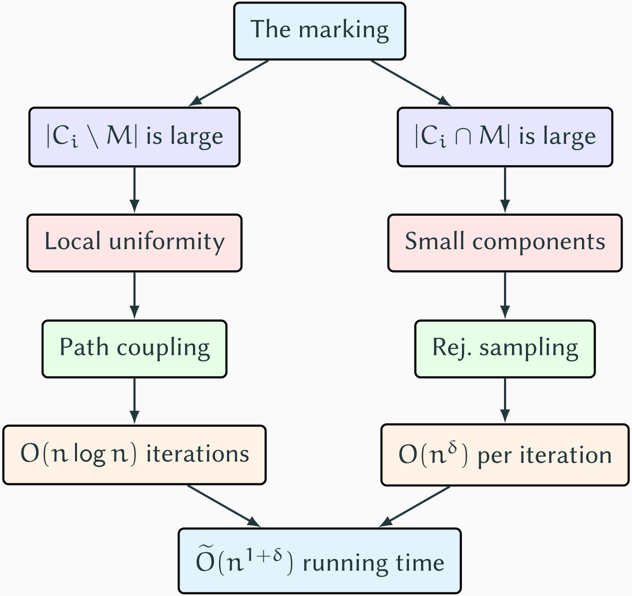
Rejection sampling terminates in $O(n^{\delta})$ steps with high probability.

$\Pr[\sigma(x_i) = T] = ??$



(Every clause here has at least 1 marked variable and 2 unmarked variable.)

Overview



The marking
$|C_i \setminus M|$ is large
$|C_i \cap M|$ is large
Local uniformity
Small components
Path coupling
Rej. sampling
$O(n \log n)$ iterations
$O(n^\delta)$ per iteration
$\widetilde{O}(n^{1+\delta})$ running time

Why rejection sampling?

Draw $\sigma_M \sim \mu_M$. For each clause, there are at least $\Omega(k)$ variables assigned. Many clauses are satisfied, and the remaining clauses scatter into connected components of size $\asymp \log n$.

However, the size is $\Omega(dk \log n)$, so a brute-force enumeration takes time $n^{\Omega(dk)}$ which is too slow to our needs.

We use the local lemma once again here to show that uniform at random assignments satisfies remaining clauses with probability at least $\Omega(n^{-\delta})$. Thus, the rejection sampling succeeds in time $\widetilde{O}(n^{\delta})$.

Path coupling

Path coupling condition: given two assignment σ_0 and σ_1 which differ on only one variable v,

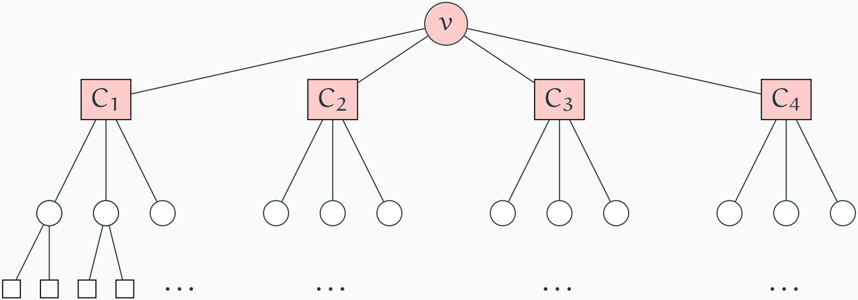
$$\sum_{u \in M,\ u \neq v} d_{TV}(\mu_u(\cdot \mid \sigma_0), \mu_u(\cdot \mid \sigma_1)) < 1.$$

Using the coupling inequality, it suffices to show that for a carefully designed coupling $\mathcal{C}$ of $\mu_{M \setminus \{v\}}(\cdot \mid \sigma_0)$ and $\mu_{M \setminus \{v\}}(\cdot \mid \sigma_1)$ such that

$$\mathop{\mathbb{E}}_{(\tau_0, \tau_1) \sim \mathcal{C}} \left| \{u \mid u \in M \setminus \{v\},\ \tau_0(u) \neq \tau_1(u)\} \right| < 1.$$

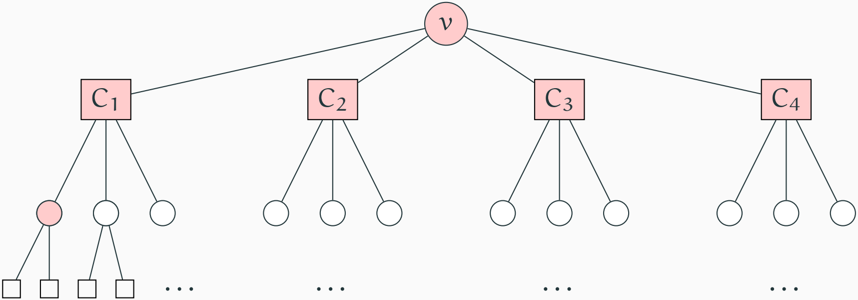
This "disagreement coupling" $\mathcal{C}$ is very similar to the one used by G., Liao, Lu and Zhang (2018), which is a refined version of Moitra (2016). However, previous analysis has a whp guarantee, and we need a new analysis to bound the expectation.

Disagreement percolation



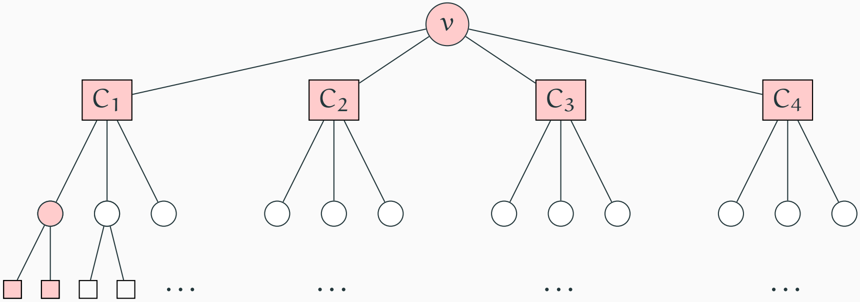
The disagreement percolation is similar to a branching process with branching factor dk, and each child survives with probability $\exp(-O(k))$ due to local uniformity.

Disagreement percolation



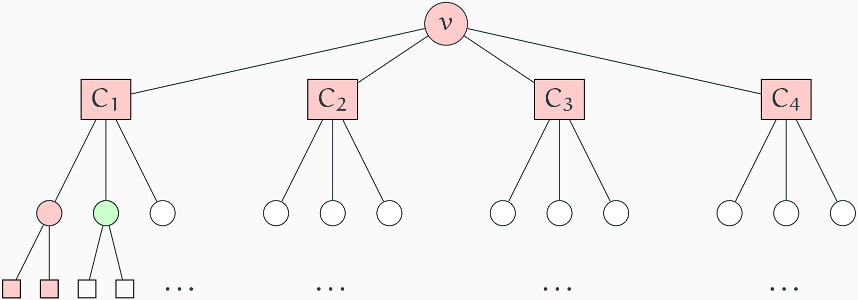
The disagreement percolation is similar to a branching process with branching factor dk, and each child survives with probability $\exp(-O(k))$ due to local uniformity.

Disagreement percolation



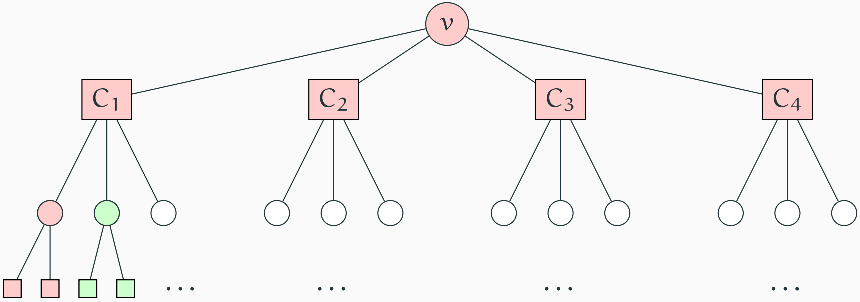
The disagreement percolation is similar to a branching process with branching factor dk, and each child survives with probability $\exp(-O(k))$ due to local uniformity.

Disagreement percolation



The disagreement percolation is similar to a branching process with branching factor dk, and each child survives with probability $\exp(-O(k))$ due to local uniformity.

Disagreement percolation



The disagreement percolation is similar to a branching process with branching factor dk, and each child survives with probability $\exp(-O(k))$ due to local uniformity.

Concluding remarks

Random k-Sat

In another recent work, Galanis, Goldberg, G. and Yang (2019) showed that there is an efficient algorithm to approximately count the number of satisfying assignment of a random k-Sat instance with high probability, if the density is at most $2^{k/300}$.

- This improves the previous best algorithm which works for density $\leqslant \frac{2 \log k}{k}$ (Montanari and Shah 2007).
- The algorithm is based on Moitra (2016), with some extra ingredients to handle $\Omega(\log n)$ degree variables.
- Nonetheless, it is not clear if the Markov chain approach works for random formulas.

Open problems

- Is the conjectured threshold correct?
    - Getting rid of the marking?

- Other CSPs, like hypergraph colouring?

- Other applications of this projection method?

Thank you!

`arXiv:1911.01319`