Fast sampling and counting k-SAT solutions in the local lemma regime

Weiming Feng
Nanjing University

Joint work with: Heng Guo (University of Edinburgh)
Yitong Yin (Nanjing University)
Chihao Zhang (Shanghai Jiao Tong University)

Conjunctive normal form (CNF)

- **$(\boldsymbol{k}, \boldsymbol{d})$-CNF formulas:**
  - each clause contains k Boolean variables.
  - each variable belongs to **at most** d clauses, e.g. **max degree $\leq \boldsymbol{d}$**.

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

example:(3,2)-CNF formula

- **SAT solutions**: an assignment of variables s.t. $\boldsymbol{\Phi} = \textbf{true}$.

- **Lovász Local Lemma (LLL):** SAT solution ***exists*** if $\boldsymbol{k} \gtrsim \boldsymbol{\log d}$.

Sampling & counting k-SAT solutions

- **Input:** a (k, d)-CNF formula $\Phi = (V, C)$ with $|V| = n$, an error bound $\epsilon > 0$.
- **Almost uniform sampling:** generate a k-SAT solution $X \in \{\text{true}, \text{false}\}^V$ s.t. the *total variation distance* $d_{TV}(X, \mu) \leq \epsilon$,

  μ: the uniform distribution of all k-SAT solutions.

- **Approximate counting:** estimate the number of k-SAT solutions, e.g. output

$$(1 - \epsilon)Z \leq \hat{\boldsymbol{Z}} \leq (1 + \epsilon)Z,$$

  Z = the number of k-SAT solutions.

Work	Regime	Running time/lower bound	Class of (k, d)-CNF formulas
Hermon et al.'19	$k \gtrsim 2 \log d$	$\mathrm{poly}(dk) n \log n$	monotone CNF
Guo et al.'17	$k \gtrsim 2 \log d$	$\mathrm{poly}(dk) n$	heavy intersection
Moitra'17	$k \gtrsim 60 \log d$	$n^{\mathrm{poly}(dk)}$	constant d, k
Bezáková et al.'15	$k \leq 2 \log d - C$	NP-hard	--

Work	Regime	Running time/lower bound	Class of (k, d)-CNF formulas
Hermon et al.'19	$k \gtrsim 2 \log d$	$\mathrm{poly}(dk) n \log n$	monotone CNF
Guo et al.'17	$k \gtrsim 2 \log d$	$\mathrm{poly}(dk) n$	heavy intersection
Moitra'17	$k \gtrsim 60 \log d$	$n^{\mathrm{poly}(dk)}$	constant d, k
Bezáková et al.'15	$k \leq 2 \log d - C$	NP-hard	--
This work	$\boldsymbol{k \gtrsim 20 \log d}$	$\boldsymbol{\tilde{O}(d^2 k^3 n^{1.000001})}$	--

Main theorem [this work]

We give a ***Markov chain*** based sampling algorithm in time $\boldsymbol{\tilde{O}(d^2 k^3 n^{1+\zeta})}$ if

$$k \geq 20 \log d + 20 \log k + 3 \log \frac{1}{\zeta}$$

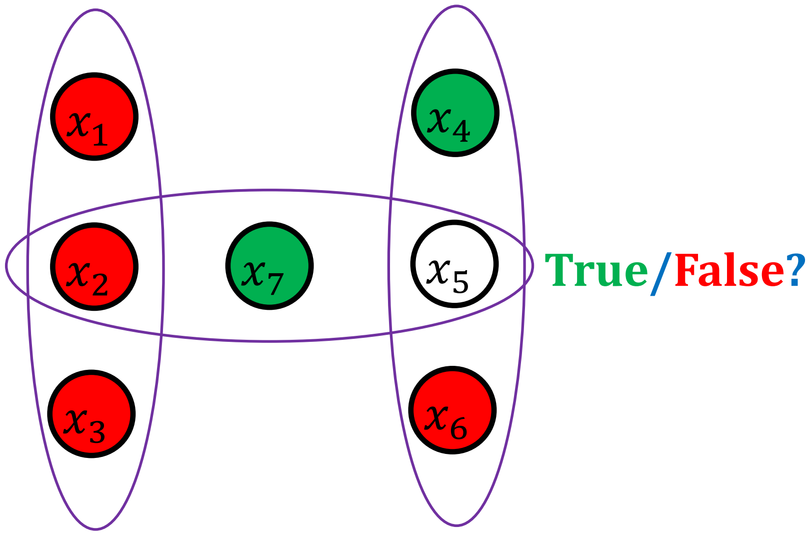
for ***arbitrary*** small $\zeta < 2^{-20}$.

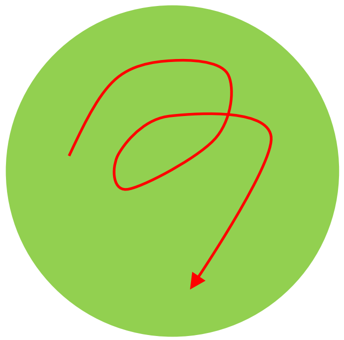
Classic Glauber dynamics (Gibbs sampling)

start from an arbitrary solution $Y \in \{T, F\}^V$;

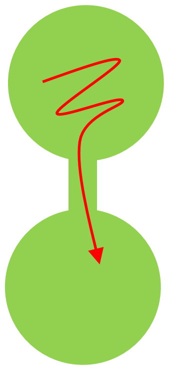
for each t from 1 to T **do**

- pick $v \in V$ uniformly at random;
- resample Y_v from conditional distribution $\mu(\cdot \mid Y_{V \setminus v})$;





rapid mixing



slow mixing

not mixing

For sampling CNF solutions,

Glauber dynamics

meets

connectivity barrier

Our technique: projection

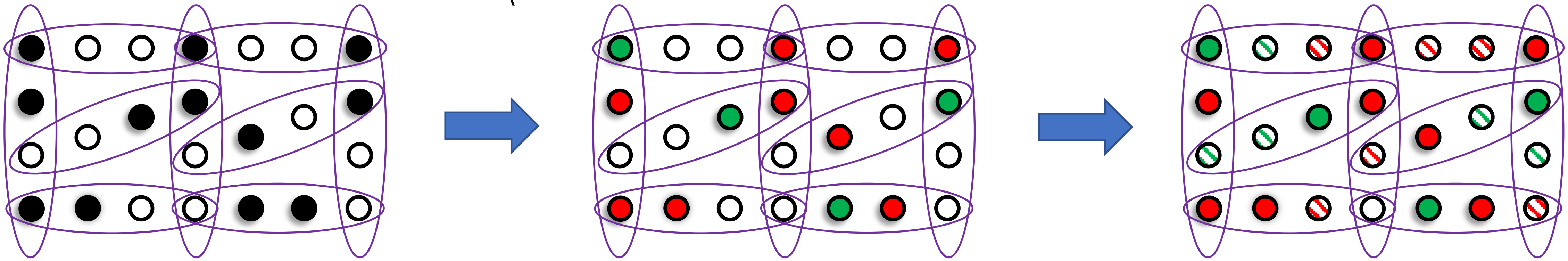
Source: https://www.shadowmatic.com/presskit/images/IMG_0650.png

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

Step 1: Construct a *good subset* of variables $M \subseteq V$

Step 2: Run *Glauber dynamics* on the *projected distribution* μ_M to draw sample $X \sim \mu_M$

Step 3: Draw sample $Y \sim \mu_{V \setminus M}(\cdot \mid X)$ from the *conditional distribution*



We are required to prove following:

- The good subset $M \subseteq V$ can ***constructed efficiently*** (use Morse-Tardos algorithm);
- The Glauber dynamics on μ_M is ***rapid mixing*** (proved via path coupling);
- The Glauber dynamics and step-3 can be ***implemented efficiently*** (via rejection sampling).

Summary

- A close to **linear time** algorithm for sampling k-SAT solutions in LLL regime.
- A close to **quadratic time** algorithm for counting k-SAT solutions in LLL regime.
- Projection + LLL technique to bypass the connectivity barrier of MCMC method.

Open problems

- Sampling & counting k-SAT solutions when $k \gtrsim 2\log d$.
- Extend the technique to more general distributions

  e.g. hyper-graph q-coloring when $q \gtrsim d^{\frac{C}{k-1}}$.

Thank you!