

# Dynamic Sampling from Graphical Models

---

Weiming Feng  
Nanjing University

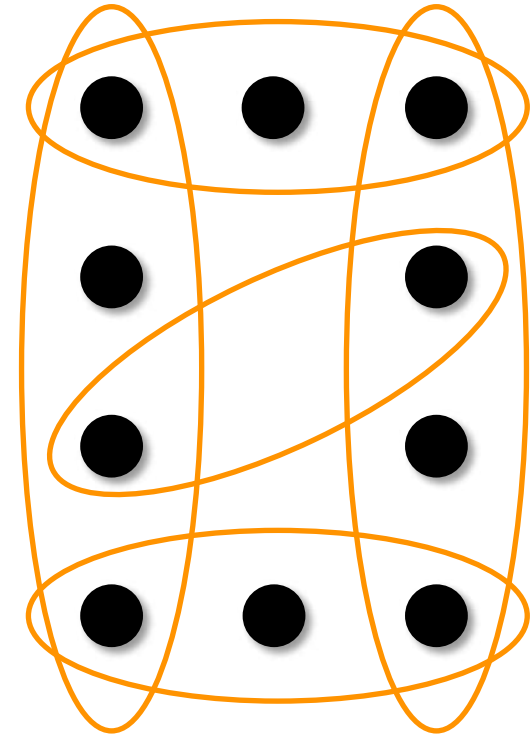
Joint work with: Nisheeth K. Vishnoi ( Yale University )  
Yitong Yin ( Nanjing University )

**STOC 2019**

# Graphical Model

- Hyper graph  $H = (V, E)$ 
  - $V$ : vertices
  - $E \subseteq 2^V$ : hyper edges.
- Vertex: **variable** with domain  $Q$ .
- Hyper edge: **constraint** on its variables.
- **Weight functions(factors)**:  $\Phi = (\phi_v)_{v \in V} \cup (\phi_e)_{e \in E}$ 
  - each variable  $\phi_v: Q \rightarrow \mathbb{R}_{\geq 0}$ ;
  - each constraint  $\phi_e: Q^e \rightarrow \mathbb{R}_{\geq 0}$ .
- Each **configuration**  $\sigma \in Q^V$ : its **weight**

$$w(\sigma) = \prod_{v \in V} \phi_v(\sigma_v) \prod_{e \in E} \phi_e(\sigma_e).$$



hyper graph  
 $H = (V, E)$

# Graphical Model

---

**Instance  $\mathcal{I} = (V, E, Q, \Phi)$**

- $V$ : **variables**
- $E$ : **constraints**
- $Q$ : **domain**
- $\Phi = (\phi_v)_{v \in V} \cup (\phi_e)_{e \in E}$ : **weight functions (factors)**

**Gibbs distribution**  $\mu$  over  $Q^V$ :

$$\forall \sigma \in Q^V: \mu(\sigma) \propto w(\sigma) = \prod_{v \in V} \phi_v(\sigma_v) \prod_{e \in E} \phi_e(\sigma_e)$$

# Hardcore Model

- Graph  $G = (V, E)$

$I(G) = \{\text{independent sets in } G\}.$

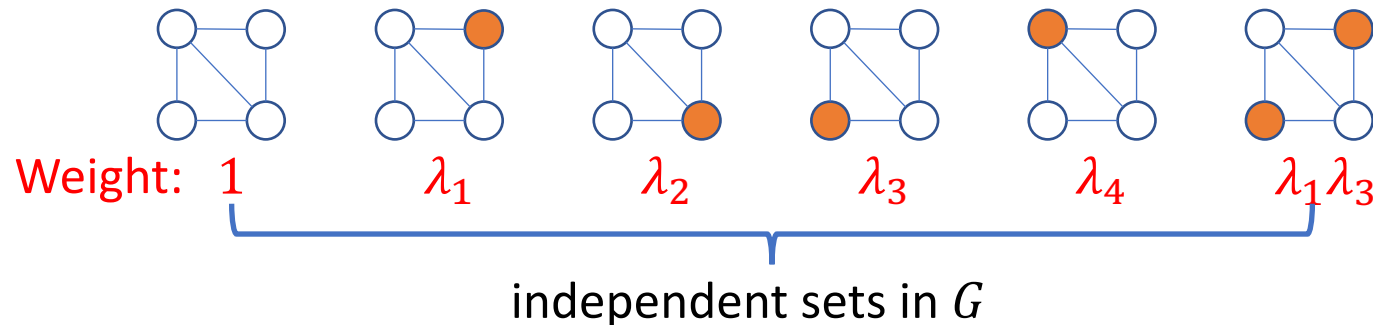
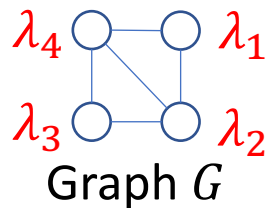
- Fugacity** of vertex  $v \in V$ :  $\lambda_v \in \mathbb{R}_{\geq 0}$ .
- Weight** of independent set  $S \in I(G)$ :

$$w(S) = \prod_{v \in S} \lambda_v.$$

product of  
vertex fugacities

- Hardcore model**: distribution  $\mu$  over  $I(G)$ , each  $S \in I(G)$ :

$$\mu(S) \propto w(S).$$



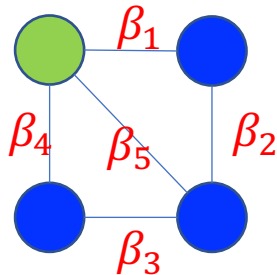
# Ising Model

- Graph  $G = (V, E)$ .
- **Inverse temperature** of edge  $e \in E$ :  $\beta_e \in \mathbb{R}_{\geq 0}$ .
- **Spin state** of vertex  $v \in V$ :  $\{-1, +1\}$ .
- **Weight** of configuration  $\sigma \in \{-1, +1\}^V$

$$w(\sigma) = \prod_{e=\{u,v\} \in E} \exp(\beta_e \sigma_u \sigma_v).$$

product of pairwise interactions

- **Ising model**: distribution  $\mu$  over  $\{-1, +1\}^V$ :  
 $\mu(\sigma) \propto w(\sigma).$



$$\text{Weight} = \exp(-\beta_1) \exp(\beta_2) \exp(\beta_3) \exp(-\beta_4) \exp(-\beta_5).$$

# Graphical Model

- **Machine Learning**  
representation, inference, learning;
- **Statistical Physics**  
Ising model, hardcore model;
- **Theoretical Computer Science**  
sampling, counting.



application: image denoising

## Sampling from Graphical Model

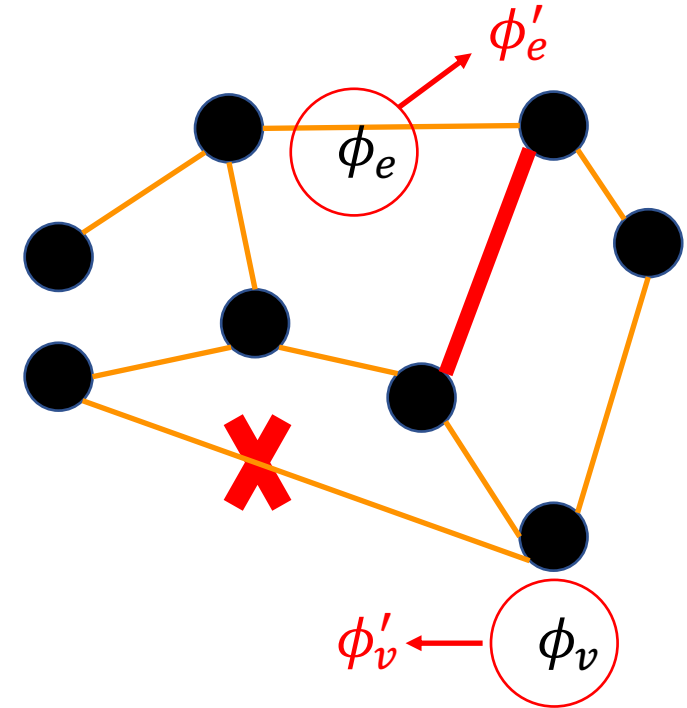
- **Input:** a graphical model  $\mathcal{I}$ ;
- **Output:** a sample  $X \sim \mu_{\mathcal{I}}$ .

# Dynamic Sampling Problem

- Graphical model  $\mathcal{I} = (V, E, Q, \Phi)$

$$\mu_{\mathcal{I}}(\sigma) \propto \prod_{v \in V} \phi_v(\sigma_v) \prod_{e \in E} \phi_e(\sigma_e).$$

- Random sample:  $\mathbf{X} \sim \mu_{\mathcal{I}}$ .
- Updates of graphical model  $\mathcal{I} \rightarrow \mathcal{I}'$ 
  - add/delete** constraints;
  - change** weight functions.



**Question:** Can we modify  $\mathbf{X}$  to  $\mathbf{X}' \sim \mu_{\mathcal{I}'}$  with a *small incremental cost*?

random sample for **updated** graphical model

Update is **represented** by a pair  $(D, \Phi_D)$

- $D \subseteq V \cup 2^V$ : **updated variables & updated constraints**;
- $\Phi_D = (\phi_a)_{a \in D}$ : **new weight functions**.

**input graphical model**

$$\mathcal{I} = (V, E, Q, \Phi)$$

**update**  $(D, \Phi_D)$



**updated graphical model**

$$\mathcal{I}' = (V, E', Q, \Phi')$$

updated constraints      updated weight functions

## Dynamic Sampling from Graphical Model

- **Input:** a graphical model  $\mathcal{I}$ ; a sample  $X \sim \mu_{\mathcal{I}}$   
an update  $(D, \Phi_D)$  that modifies  $\mathcal{I}$  to  $\mathcal{I}'$ ;
- **Output:** a sample  $X' \sim \mu_{\mathcal{I}'}$ .

**Offline adversary:** update is **independent** with the input sample  $X$ .

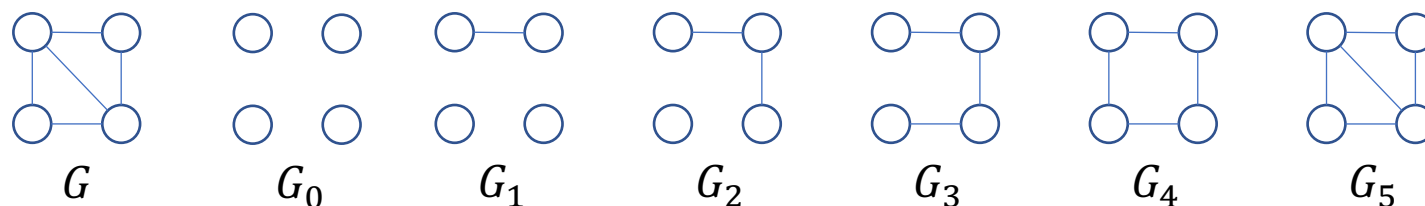


# Motivations

- Online learning with dynamic or streaming data
- Dynamic graphical models
  - Video: a sequence of closely related images.



- Approximate counting [Jerrum, Valiant, Vazirani, 1986]
  - Graph  $G = (V, E)$ , count  $\#\{\text{independent sets of } G\}$ .
  - Self reduction: a sequence of graphs  $G_0, G_1, \dots, G_{|E|}$ :



## Static Sampling

- **Input:** a graphical model  $\mathcal{J}$ ;
- **Output:** a sample  $\mathbf{X} \sim \mu_{\mathcal{J}}$ .

**Well** studied

## Dynamic Sampling

- **Input:** a graphical model  $\mathcal{J}$ ;  
a sample  $\mathbf{X} \sim \mu_{\mathcal{J}}$   
a update  $(D, \Phi_D)$
- **Output:** a sample  $\mathbf{X}' \sim \mu_{\mathcal{J}'}$ .

**Lacking** studies

## Algorithms for static sampling

- Markov chain Monte Carlo (MCMC)
  - Metropolis Hastings [Metropolis 1953]
  - Glauber Dynamics [Glauber 1963]
- Coupling from the past (CFTP) [Propp and Wilson 1996]

**Not suitable** for dynamic sampling, per se.

- **can not use** the input sample  $\mathbf{X}$ ;
- rerunning sampling algorithm on  $\mathcal{J}'$  is **wasteful**.

# Our Contribution

**New  
Algorithm**



## Dynamic Sampling Problem

- **Input:** a graphical model  $\mathcal{I}$ ;  
a sample  $\mathbf{X} \sim \mu_{\mathcal{I}}$   
a update  $(D, \Phi_D)$
- **Output:** a sample  $\mathbf{X}' \sim \mu_{\mathcal{I}'}$ .

- ***Fast***

a broad class of graphical models   $\mathbb{E}[\text{running time}] = O(|D|)$

- ***Exact Sampling***

$\mathbf{X}$  follows precisely distribution  $\mu_{\mathcal{I}'}$

- ***Las Vegas***

algorithm knows when to stop

- ***Distributed / Parallel***

each step uses only local information

# Graphical Model

---

**Instance  $\mathcal{I} = (V, E, Q, \Phi)$**

- $V$ : **variables**
- $E$ : **constraints**
- $Q$ : **domain**
- $\Phi = (\phi_v)_{v \in V} \cup (\phi_e)_{e \in E}$ : **weight functions (factors)**

**Gibbs distribution**  $\mu$  over  $Q^V$ :

$$\forall \sigma \in Q^V: \mu(\sigma) \propto w(\sigma) = \prod_{v \in V} \phi_v(\sigma_v) \prod_{e \in E} \phi_e(\sigma_e)$$

# Rejection Sampling

---

Graphical model  $\mathcal{I} = (V, E, Q, \Phi)$  with Gibbs distribution

$$\mu_{\mathcal{I}}(\sigma) \propto \prod_{v \in V} \phi_v(\sigma_v) \prod_{e \in E} \phi_e(\sigma_e).$$

**Assumption: normalized weighted functions**

- each  $\phi_v: Q \rightarrow [0,1]$  is a **distribution** over  $Q$ :  $\sum_{c \in Q} \phi_v(c) = 1$ ;
- each  $\phi_e: Q^e \rightarrow [0,1]$ .

## Rejection Sampling

- Each  $v \in V$  samples  $X_v \sim \phi_v$  ind.;
- Each  $e \in E$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ; o.w.  $e$  becomes **rejected**
- **Accept**  $X = (X_v)_{v \in V}$  if all  $e \in E$  are accepted;
- **Reject**  $X$  if otherwise.

$$\Pr[X = \sigma \wedge X \text{ is accepted}] = \prod_{v \in V} \phi_v(\sigma_v) \prod_{e \in E} \phi_e(\sigma_e).$$

generate  $X = \sigma$  all  $e \in E$  are accepted

$$\Pr[\text{all } e \in E \text{ are accepted}] = \exp(-\Omega(|E|)).$$

Rejection Sampling is **Correct** but **Slow**.

## Question

Can we obtain an efficient rejection sampling algorithm ?

- Fast
- Dynamic
- Distributed / Parallel

This problem was **partially solved** by  
Partial Rejection Sampling (**PRS**) [Guo, Jerrum, Liu, 2017].

- **Boolean** weight function  $\phi_e \rightarrow \{0,1\}$
- **Not known** to be dynamic
- **Not** distributed / parallel

# Our Contribution

---

**New  
Algorithm**



## Dynamic Sampling Problem

- **Input:** a graphical model  $\mathcal{I}$ ;  
a sample  $\mathbf{X} \sim \mu_{\mathcal{I}}$   
a update  $(D, \Phi_D)$
- **Output:** a sample  $\mathbf{X}' \sim \mu_{\mathcal{I}'}$ .

## Efficient Rejection Sampling

- Fast
- Dynamic
- Distributed/Parallel



## Rejection Sampling

- Each  $v \in V$  samples  $X_v \sim \phi_v$  ind.;
- Each  $e \in E$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ; o.w.  $e$  becomes **rejected**
- **Accept**  $X = (X_v)_{v \in V}$  if **all**  $e \in E$  are accepted;
- **Reject**  $X$  if otherwise.

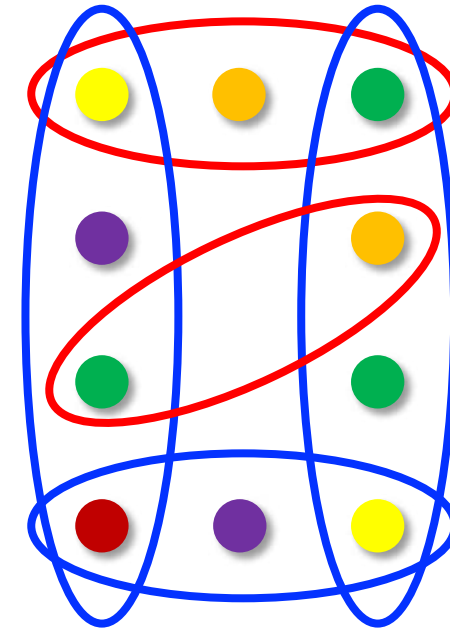
The sample  $X$  is **rejected**



Set of **Bad Variables**

$$\mathcal{R} = \bigcup_{\substack{e \in E: \\ e \text{ is rejected}}} e,$$

$\mathcal{R}$ : **variables** in **rejected constraints**



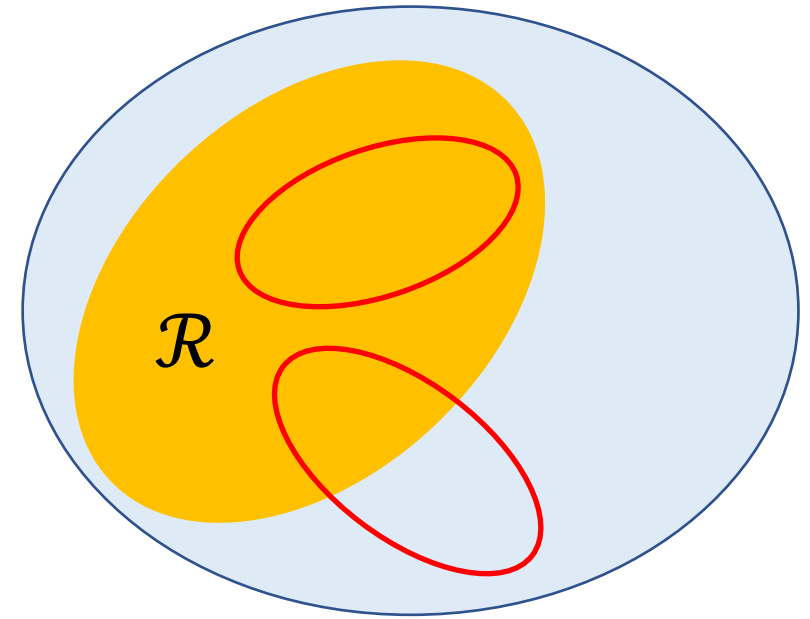
 accepted constraint  
 rejected constraint

## A “**Natural**” Resampling Algorithm

- each  $v \in \mathcal{R}$  resamples  $X_v \sim \phi_v$  ind.; o.w.  $e$  becomes **rejected**
- each  $e \in \text{ICD}(\mathcal{R})$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ;
- construct new  $\mathcal{R} \leftarrow \bigcup_{e \in E: e \text{ is rejected}} e$ .

$\text{ICD}(\mathcal{R})$ : constraints **incident** to  $\mathcal{R}$

$$\text{ICD}(\mathcal{R}) = \{e \in E \mid e \cap \mathcal{R} \neq \emptyset\}.$$



## A “**Natural**” Resampling Algorithm

- each  $v \in \mathcal{R}$  resamples  $X_v \sim \phi_v$  ind.; o.w.  $e$  becomes **rejected**
- each  $e \in \text{ICD}(\mathcal{R})$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ;
- construct new  $\mathcal{R} \leftarrow \bigcup_{e \in E: e \text{ is rejected}} e$ .

**While**( $\mathcal{R} \neq \emptyset$ )

**Update**  $(X, \mathcal{R})$  by “**Natural**” Resampling Algorithm

**Output**  $X$ .

 **Wrong  
Distribution**

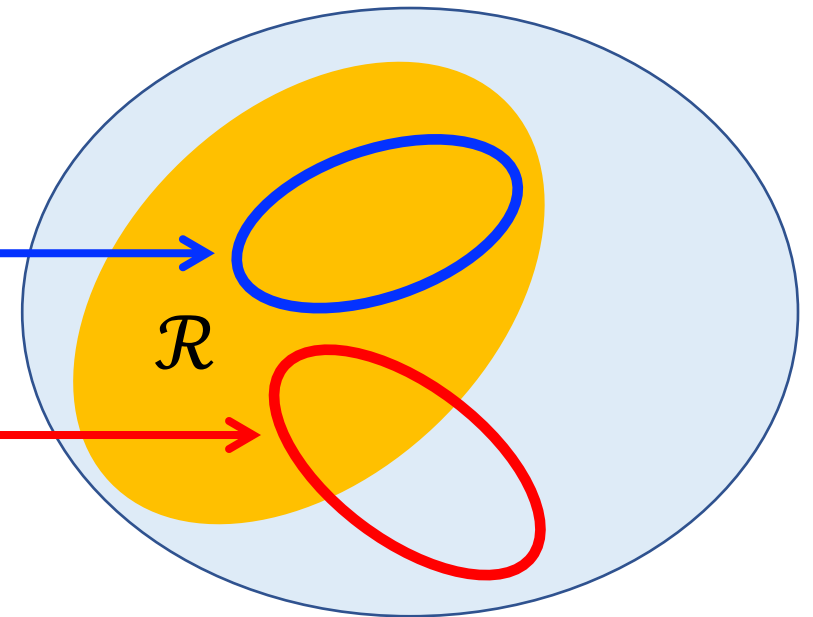
- Similar to **Morse-Tardos** for LLL. [Morse, Tardos, 2009]
- The output  $X$  does **NOT** follow the Gibbs distribution  $\mu$ .  
[Harris, Srinivasan, 2016], [Guo, Jerrum, Liu, 2017]

## A “**Natural**” Resampling Algorithm

- each  $v \in \mathcal{R}$  resamples  $X_v \sim \phi_v$  ind.;
- each  $e \in \text{ICD}(\mathcal{R})$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ;
- construct new  $\mathcal{R} \leftarrow \bigcup_{e \in E: e \text{ is rejected}} e$ .

$\text{ICD}(\mathcal{R})$ : constraints **incident** to set  $\mathcal{R}$

- $\text{ICD}(\mathcal{R})$  {
- **internal constraints**  $E(\mathcal{R}) = \{e \in E \mid e \subseteq \mathcal{R}\};$
  - **boundary constraints**  $\delta(\mathcal{R}) = \{e \in E \setminus E(\mathcal{R}) \mid e \cap \mathcal{R} \neq \emptyset\}.$



## A “**Natural**” Resampling Algorithm

- each  $v \in \mathcal{R}$  resamples  $X_v \sim \phi_v$  ind.;
- each  $e \in \text{ICD}(\mathcal{R})$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ;
- construct new  $\mathcal{R} \leftarrow \bigcup_{e \in E: e \text{ is rejected}} e$ .

## Our Algorithm: Local-Resample( $X, \mathcal{R}$ )

- each  $v \in \mathcal{R}$  resamples  $X_v \sim \phi_v$  ind.;
- [** each  $e \in E(\mathcal{R})$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ;
- [** each  $e \in \delta(\mathcal{R})$  becomes **accepted** ind. with a **modified probability**;
- construct new  $\mathcal{R} \leftarrow \bigcup_{e \in E: e \text{ is rejected}} e$ .
- return  $(X, \mathcal{R})$ ;

## Our Algorithm: Local-Resample( $X, \mathcal{R}$ )

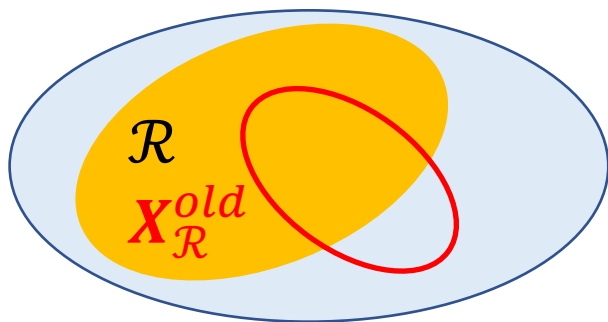
- ① each  $v \in \mathcal{R}$  resamples  $X_v \sim \phi_v$  ind.;
- ② each  $e \in E(\mathcal{R})$  becomes **accepted** ind. w.p.  $\phi_e(X_e)$ ;
- ③ each  $e \in \delta(\mathcal{R})$  becomes **accepted** ind. w.p.  $C_e \cdot \frac{\phi_e(X_e)}{\phi_e(X_e^{old})} \leq 1$ ;
- ④ construct new  $\mathcal{R} \leftarrow \bigcup_{e \in E: e \text{ is rejected}} e$ .
- ⑤ return  $(X, \mathcal{R})$ ;

**normalization factor**

$X^{old} \in Q^V$  is the old  $X$  **before** the resampling in step ①

**Normalization Factor**  $C_e = C_e(X_{\mathcal{R}}^{old})$ :

$$C_e = \min_{y \in Q^e: y_{e \cap \mathcal{R}} = X_{e \cap \mathcal{R}}^{old}} \phi_e(y).$$



**While**( $\mathcal{R} \neq \emptyset$ )

$(X, \mathcal{R}) \leftarrow \text{Local-Resample}(X, \mathcal{R})$

**Output**  $X$ .

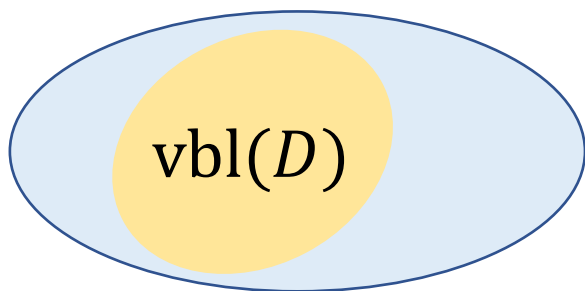
**Correct Distribution**

# Dynamic Sampler

## Dynamic Sampling from Graphical Model

- **Input:** a graphical model  $\mathcal{I}$ ; a sample  $\mathbf{X} \sim \mu_{\mathcal{I}}$   
a update  $(D, \Phi_D)$  that modifies  $\mathcal{I}$  to  $\mathcal{I}'$ ;
- **Output:** a sample  $\mathbf{X}' \sim \mu_{\mathcal{I}'}$ .

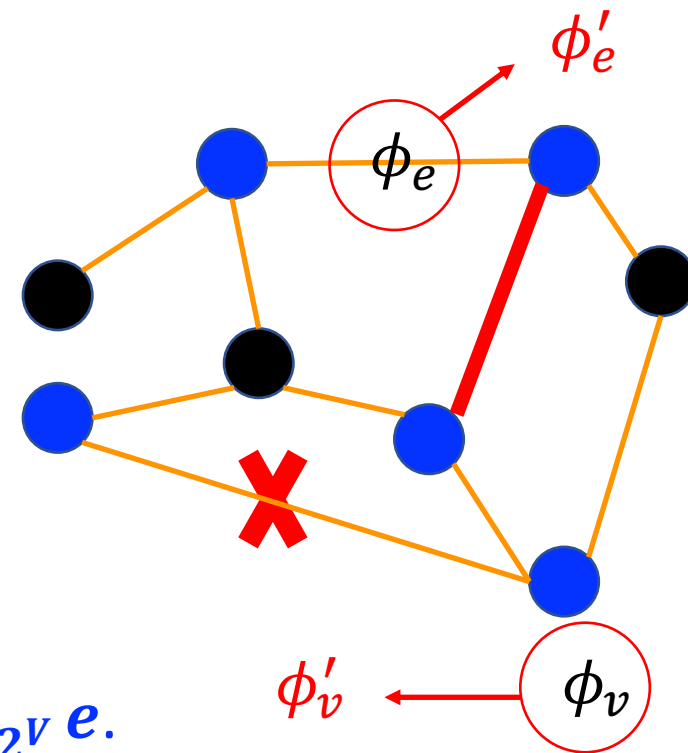
- $D$ : updated variables & updated constraints;
- $\text{vbl}(D)$ : variables **involved** by the update:
  - **updated variables:**  $D \cap V$ ;
  - **variables incident to updated constraints:**  $\bigcup_{e \in D \cap \mathcal{E}} \text{vbl}(e)$ .



Graphical models  $\mathcal{I}$  and  $\mathcal{I}'$  **differ** only on  $\text{vbl}(D)$ .



The initial **bad set**  $\mathcal{R} = \text{vbl}(D)$ .



## Dynamic Sampler

- Apply changes  $(D, \Phi_D)$  to current graphical model  $\mathcal{I}$ .
- $\mathcal{R} \leftarrow \text{vbl}(D)$ ;
- **While**( $\mathcal{R} \neq \emptyset$ )
  - $(X, \mathcal{R}) \leftarrow \text{Local-Resample}(X, \mathcal{R})$ ;
- **Return**  $X$ ;

### **Theorem: Correctness** [This Work]

Upon termination, the dynamic sampler outputs  $X \sim \mu_{\mathcal{I}'}$ .

A dynamic sampler for **general graphical model**:

- *Exact sampling*
- *Las Vegas*
- *Distributed / Parallel*



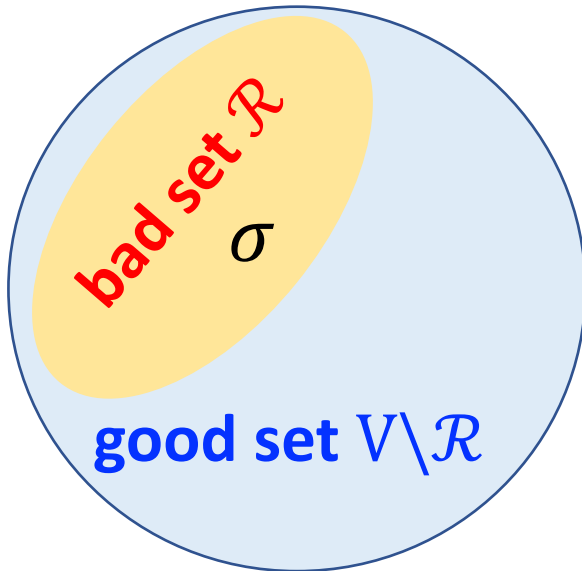
# Proof of Correctness

## Dynamic Sampler

- Apply changes  $(D, \Phi_D)$  to current graphical model  $\mathcal{J}$ .
- $\mathcal{R} \leftarrow \text{vbl}(D)$ ;
- **While**( $\mathcal{R} \neq \emptyset$ )
  - $(X, \mathcal{R}) \leftarrow \text{Local-Resample}(X, \mathcal{R})$ ;
- **Return**  $X$ ;

The algorithm maintains  $(X, \mathcal{R}) \in Q^V \times 2^V$

- $\mathcal{R}$ : **bad set**;
- $V \setminus \mathcal{R}$ : **good set**  
 $X_{V \setminus \mathcal{R}}$  follows “**correct**” distribution.



## Conditional Gibbs property w.r.t. $\mu$

Conditioning on any  $\mathcal{R} \subseteq V$  and any assignment  $\sigma \in Q^{\mathcal{R}}$  of  $X_{\mathcal{R}}$ , the distribution of  $X_{V \setminus \mathcal{R}}$  is  $\mu_{V \setminus \mathcal{R}}^{\sigma}$ .

$\mu_{V \setminus \mathcal{R}}^{\sigma}$ : marginal distribution of  $\mu$  on  $V \setminus \mathcal{R}$  conditioning on  $\sigma$ .

**Local-Resample( $X, \mathcal{R}$ )**

define



### Resampling chain

- **Markov chain** on  $\Omega = Q^V \times 2^V$
- Transition Matrix  $P \in \mathbb{R}^{\Omega \times \Omega}$   
 $P: (X, \mathcal{R}) \rightarrow (X', \mathcal{R}')$

### Equilibrium Condition

If  $(X, \mathcal{R})$  satisfies the conditionally Gibbs property w.r.t.  $\mu$ , then so does  $(X', \mathcal{R}')$ .



### Equation System for Equilibrium Condition

$\forall S, T \subseteq V, \sigma \in Q^{V \setminus S}$  and  $\tau \in Q^{V \setminus T}$ ,

$$\forall y \in Q^V, y_{V \setminus T} = \tau: \sum_{\substack{x \in Q^V \\ x_{V \setminus S} = \sigma}} \mu_S^\sigma(x_S) \cdot P((x, S), (y, T)) = \mathcal{C}(S, \sigma, T, \tau) \cdot \mu_T^\tau(y_T).$$



Our algorithm is a solution to this equation system.

## Theorem: Fast Convergence [This Work]

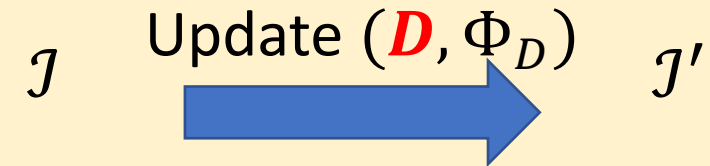
The updated graphical model satisfies  $d = O(1)$ ,  $\max_{e \in E'} |e| = O(1)$ , and

$$\forall e \in E': \min_x \phi'_e(x) > \sqrt{1 - \frac{1}{d+1}}$$

where  $d$  is the maximum degree of the dependency graph.

The **cost** of the dynamic sampler is

- $O(\log |D|)$  **iterations** in **expectation**;
- $O(|D|)$  **resamplings** in **expectation**.



**Ising Model:**  $\forall e \in E:$

$$1 - \exp(-2|\beta_e|) < \frac{1}{4\Delta}$$

**Uniqueness Regime:**  $\forall e \in E:$

$$1 - \exp(-2|\beta_e|) < \frac{2}{\Delta}$$

## Theorem: Fast Convergence [This Work]

Hardcore model and Ising model on bounded degree graph s.t.

- Hardcore model:  $\forall v \in V: \lambda_v \leq \frac{1}{\sqrt{2\Delta}-1}$ .
- Ising model:  $\forall e \in E: 1 - \exp(-2|\beta_e|) \leq \frac{1}{2.221\Delta+1}$ ,

where  $\Delta$  is the maximum degree.

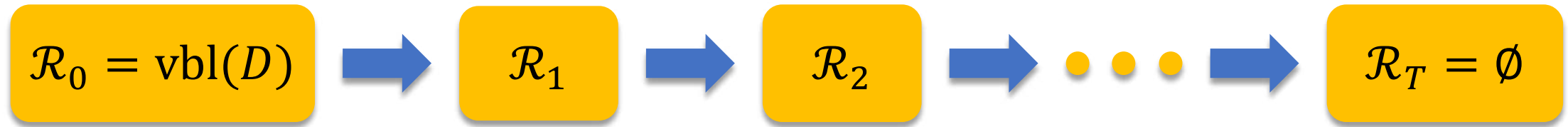
The **cost** of the dynamic sampler is

- $O(\log |D|)$  **iterations** in **expectation**;
- $O(|D|)$  **resamplings** in **expectation**;

## Uniqueness Regime:

- Hardcore model:  $\forall v \in V: \lambda_v < \frac{(\Delta-1)^{\Delta-1}}{(\Delta-2)^\Delta} \approx \frac{e}{\Delta-2}$ .  $\lambda_v = O\left(\frac{1}{\Delta}\right)$
- Ising model:  $\forall e \in E: 1 - \exp(-2|\beta_e|) < \frac{2}{\Delta}$ .  $1 - \exp(-2|\beta_e|) = O\left(\frac{1}{\Delta}\right)$

# Proof of the Fast Convergence

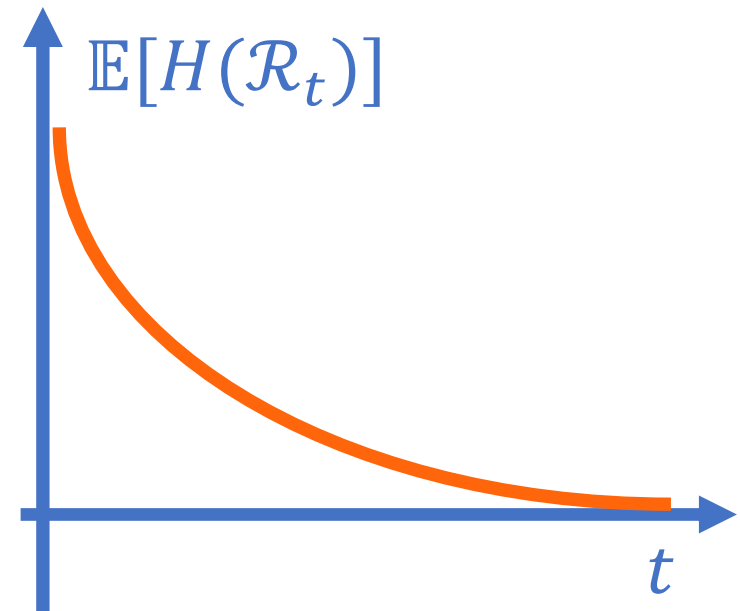


**Potential function** on **bad set**  $\mathcal{R}_t$

$$H: 2^V \rightarrow \mathbb{Z}_{\geq 0}$$

**Step-wise decay** on **expectation** of  $H(\mathcal{R}_t)$

$$\mathbb{E}[H(\mathcal{R}_t)] \leq (1 - \delta) \mathbb{E}[H(\mathcal{R}_{t-1})].$$



# Summary

- **Dynamic sampling problem.**
- **Dynamic sampler** for general graphical models  
*Exact Sampling & Las Vegas & Distributed/Parallel.*
- **Equilibrium conditions** for resampling.

## Future Work

- **Dynamic MCMC sampling** [Feng, He, Yin, Sun, arXiv:1904.11807]
- Improve the **regimes** for efficient dynamic sampling  
*correlation decay → ? efficient dynamic sampling algorithm.*
- Extend to **continuous distributions & global constraints.**

# Thank You

# See you at the poster session #131

## Dynamic Sampling from Graphical Models

Weiming Feng<sup>1</sup> Nisheeth K. Vishnoi<sup>2</sup> Yitong Yin<sup>1</sup>

<sup>1</sup>Nanjing University, China <sup>2</sup>Yale University, USA

### Abstract

We study the problem of sampling from a graphical model when the model itself is changing dynamically with time.

- We give an algorithm that can sample dynamically from a broad class of graphical models efficiently.
- We give an equilibrium condition that guarantees the correctness of the dynamic sampling.

### Graphical Model

Graphical models arise in a variety of disciplines ranging from statistical physics, machine learning, statistics, to theoretical computer science. A graphical model instance is specified by a tuple  $\mathcal{J} = (V, E, Q, \Phi)$ :

- variable set (vertex set)  $V$ ;
- constraint set (edge set)  $E \subseteq 2^V$ ;
- finite domain  $Q$ ;
- factors (weight functions)  $\Phi = \{\phi_e\}_{e \in E} \cup \{\phi_v\}_{v \in V}$ 
  - each  $\phi_v: Q^v \rightarrow \mathbb{R}_{\geq 0}$ ;
  - each  $\phi_e: Q^e \rightarrow \mathbb{R}_{\geq 0}$ ;
- Gibbs distribution  $\mu$  over  $Q^V$ :

$$\forall \sigma \in Q^V, \quad \mu(\sigma) \propto \prod_{v \in V} \phi_v(\sigma_v) \prod_{e \in E} \phi_e(\sigma_e).$$

Example: Ising model  $\mathcal{J} = (V, E, \beta)$

- graph  $G = (V, E)$ ;
- finite domain  $Q = \{-1, +1\}$ ;
- inverse temperature  $\beta = \{\beta_e\}_{e \in E}$ , each  $\beta_e \in \mathbb{R}_{\geq 0}$ ;
- Gibbs distribution  $\mu$  over  $(-1, +1)^V$ :
 
$$\forall \sigma \in \{-1, +1\}^V, \quad \mu(\sigma) \propto \prod_{v \in V} \exp(\beta_v \sigma_v) \prod_{e \in E} \exp(\beta_e \sigma_e \sigma_{e'});$$
- uniqueness condition
 
$$\forall e \in E: \exp(-2|\beta_e|) > 1 - \frac{2}{\Delta}.$$

Example: hardcore model  $\mathcal{J} = (V, E, \lambda)$ .

- graph  $G = (V, E)$ ;
- finite domain  $Q = \{0, 1\}$ ;
- fugacity  $\lambda = \{\lambda_v\}_{v \in V}$ , each  $\lambda_v \in \mathbb{R}_{\geq 0}$ ;
- Gibbs distribution  $\mu$  over  $\{0, 1\}^V$ :  $\forall \sigma \in \{0, 1\}^V$ ,
 
$$\mu(\sigma) \propto \begin{cases} \prod_{v \in V} \lambda_v & \text{if } I(\sigma) \text{ is an independent set,} \\ 0 & \text{if } I(\sigma) \text{ is not an independent set,} \end{cases}$$
 where  $I(\sigma) = \{v \in V \mid \sigma_v = 1\}$ ;
- uniqueness condition
 
$$\forall v \in V: \quad \lambda_v < \frac{(\Delta-1)^{\Delta-1}}{(\Delta-2)^{\Delta-1}} \frac{e}{\Delta-2}.$$

### Dynamic Sampling Problem

Given: dynamic graphical model and current sample.

Main question: "Can we obtain a sample from an updated graphical model with a small incremental cost?"

#### Updates of graphical model

- add/delete constraints;
- change factors  $\phi_v \rightarrow \phi'_v$ ,  $\phi_e \rightarrow \phi'_e$ ;
- add/delete independent variables.

An update of graphical model  $\mathcal{J} = (V, E, Q, \Phi)$  is represented by a pair  $(D, \Phi_D)$ :

- $D \subseteq V \cup 2^V$ : updated variables and constraints;
- $\Phi_D := \{\phi_v\}_{v \in D \cap V} \cup \{\phi_e\}_{e \in D \cap 2^V}$ : new factors.

#### Dynamic sampling from graphical model

- Input: a graphical model  $\mathcal{J}$ , a sample  $X \sim \mu_{\mathcal{J}}$  and an update  $(D, \Phi_D)$  that modifies  $\mathcal{J}$  to  $\mathcal{J}'$ .
- Output: a sample  $X' \sim \mu_{\mathcal{J}'}$ .

Offline adversary: the update  $(D, \Phi_D)$  is independent with the input random sample  $X \sim \mu_{\mathcal{J}}$ .

#### Motivation

Approximate counting [Jerrum, Valiant, Vazirani, 1986]

- Given a graph  $G = (V, E)$ , count  $\#I$  (independent sets of  $G$ ).
- Self reduction: a sequence of graphs  $G_0, G_1, \dots, G_{\ell-1}$ :

- Counting  $\implies$  Sampling uniform independent sets.

#### Inference/learning tasks

- online learning with dynamic or streaming data;
- dynamic graphical models e.g. videos.

### Dynamic Sampler

#### Notations

- Update of graphical model  $(D, \Phi_D)$ .
- $\text{vbl}(D) := (D \cap V) \cup (\bigcup_{e \in D \cap 2^V} e)$ : variables involved by the update  $(D, \Phi_D)$ :
  - updated variables;
  - variables incident to updated constraints.

- Subset of variables  $\mathcal{R} \subseteq V$ :
  - internal constraints  $E(\mathcal{R}) := \{e \in E \mid e \subseteq \mathcal{R}\}$ ;
  - boundary constraints  $\delta(\mathcal{R}) := \{e \in E \mid e \cap \mathcal{R} \neq \emptyset\}$ ;
  - incident constraints  $E^+(\mathcal{R}) := E(\mathcal{R}) \cup \delta(\mathcal{R})$ .

#### The Algorithm

Assumption: normalized factors  $\Phi = \{\phi_v\}_{v \in V} \cup \{\phi_e\}_{e \in E}$  each  $\phi_v: Q^v \rightarrow [0, 1]$  is a distribution over  $Q$ ; each  $\phi_e: Q^e \rightarrow [0, 1]$ .

#### Dynamic Sampler

Input: a graphical model  $\mathcal{J}$  and a sample  $X \sim \mu_{\mathcal{J}}$ ;

Update: an update  $(D, \Phi_D)$  that modifies  $\mathcal{J} \rightarrow \mathcal{J}'$ ;

- apply changes  $(D, \Phi_D)$  to current graphical model  $\mathcal{J}$ ;
- $\mathcal{R} \leftarrow \text{vbl}(D)$ ;
- While  $|\mathcal{R}| \neq \emptyset$ 
  - $(X, \mathcal{R}) \leftarrow \text{Local-Resample}(X, \mathcal{R})$ ;
- Return  $X$ ;

#### Local-Resample( $X, \mathcal{R}$ ):

- each  $e \in E^+(\mathcal{R})$  computes  $\kappa_e$ ;
  - each  $v \in \mathcal{R}$  resamples  $X_v \sim \phi_v$ ; then, update  $X_e$ ;
  - each  $e \in E^+(\mathcal{R})$  samples  $F_e \in [0, 1]$  independently s.t.
 
$$\Pr[F_e = 0] = \kappa_e \phi_e(X_e);$$
 (depend on both old and new samples)
- $X' \leftarrow X$  and  $\mathcal{R}' \leftarrow \bigcup_{e \in E^+(\mathcal{R})} \{v \in V \mid F_e = 1\}$ ;
- Return  $(X', \mathcal{R}')$ ;

$$\kappa_e := \frac{1}{\phi_e(X_e)} \prod_{v \in \mathcal{R} \cap e} \phi_v(X_v)$$

(with the convention  $\frac{0}{0} = 1$ ).

$\kappa_e$ : the minimum value of  $\phi_e(X_e)$  conditioning on the assignment of  $y$  on  $e \cap \mathcal{R}$  is fixed as  $X_{\mathcal{R} \cap e}$ .

#### Properties:

- for each  $e \in E(\mathcal{R})$ ,  $\kappa_e = 1$ ;
- for each  $e \in \delta(\mathcal{R})$ ,  $\kappa_e \leq 1$ .

### Our Results

#### Theorem: Correctness

The dynamic sampler outputs the correct sample  $X \sim \mu_{\mathcal{J}'}$  guaranteed by the equilibrium condition

#### Features of the Algorithm

dynamic, exact sampling, Las Vegas, distributed/parallel.

#### Theorem: Fast Convergence

- $d := \max_{e \in E} \{d(e) \mid d(e) = |\{e' \in E \mid e' \cap e \neq \emptyset\}|\}$ : the maximum degree of the dependency graph
- $\forall e \in E: \min \phi_e \geq \frac{1}{1+d}$ 
  - $\implies$  the cost of the dynamic sampler:
    - $O(\log(D))$  iterations in expectation;
    - $O(|D|)$  resamplings in expectation.

Better results on concrete graphical models:

- Ising model:  $\forall e \in E: \exp(-2|\beta_e|) \geq 1 - \frac{1}{2.218+1}$ ;
- Hardcore model:  $\forall v \in V: \lambda_v \leq \frac{1}{\Delta-2}$ .

#### Equilibrium Condition

The dynamic sampler maintains a random pair  $(X, \mathcal{R}) \in Q^V \times 2^V$ .

- $\mathcal{R}$ : current resample set that contains the problematic variables to be resampled;
- $X$ : current sample set that contains the non-problematic variables.

#### Conditional Gibbs property:

A random pair  $(X, \mathcal{R}) \in Q^V \times 2^V$  is conditionally Gibbs w.r.t.  $\mu$  if conditioning on any  $\mathcal{R} \subseteq V$  and any assignment  $\sigma \in Q^{\mathcal{R}}$  of  $X_{\mathcal{R}}$ , the distribution of  $X_{V \setminus \mathcal{R}}$  is precisely  $\mu_{\mathcal{J}'|_{\mathcal{R}}^{\sigma}}$ .

$\mu_{\mathcal{J}'|_{\mathcal{R}}^{\sigma}}$ : marginal distribution of  $\mu$  on  $V \setminus \mathcal{R}$  conditioning on  $\sigma$ .

When  $\mathcal{R} = \emptyset$ , the random sample  $X \sim \mu$ .

#### Resampling chain

The resampling chain is a Markov chain over  $Q^V \times 2^V$  with transition matrix  $P: (X, \mathcal{R}) \rightarrow (X', \mathcal{R}')$ .

#### Equilibrium condition for resampling chain:

If  $(X, \mathcal{R})$  is conditionally Gibbs w.r.t.  $\mu$ , then  $(X', \mathcal{R}')$  is also conditionally Gibbs w.r.t.  $\mu$ .

#### The condition is established by verifying equation system:

$\forall \mathcal{R} \subseteq V, \sigma \in Q^{\mathcal{R}}$  and  $e \in E(\mathcal{R})$ :

$$\forall y \in Q^{\mathcal{R} \setminus \{v\}}, \forall v \in \mathcal{R}: \sum_{\substack{y' \in Q^{\mathcal{R} \setminus \{v\}} \\ y' \cap \{v\} = y}} \mu_{\mathcal{J}'|_{\mathcal{R} \setminus \{v\}}^{\sigma}}(x, y') < (1, \sigma, \tau, \tau) \cdot \mu_{\mathcal{J}'|_{\mathcal{R}}^{\sigma}}(y, \tau).$$

Dynamic Sampling Algorithm  $\implies$  a solution to Equation System Equilibrium Condition