Dynamic MCMC Sampling

Weiming Feng† Kun He‡ Xiaoming Sun‡ Yitong Yin†

Abstract

The Markov chain Monte Carlo (MCMC) methods are the primary tools for sampling from Gibbs distributions arising by various graphical models, e.g., Markov random fields (MRF). Traditional MCMC sampling algorithms are focused on a classic static setting, where the input is fixed. In this paper we study the problem of sampling from a MRF when the graphical model itself is changing dynamically with time. The problem is well motivated by the growing volume and velocity of data in today’s applications of the MCMC methods.

For the two major MCMC approaches, respectively for the approximate and perfect sampling, namely, the Gibbs sampling and the coupling from the past (CFTP), we give dynamic versions for the respective MCMC sampling algorithms. On MRF with $n$ variables and bounded maximum degrees, these dynamic sampling algorithms can maintain approximate samples within $1/\text{Poly}(n)$ total variation errors, or perfect samples, while the MRF is dynamically changing. Furthermore, the dynamic sampling algorithms are efficient with $O(n)$ space cost, and $O(\log^2 n)$ incremental time cost upon each local update to the input MRF, as long as certain decay conditions are satisfied in each step by natural couplings of the corresponding single-site chains. These decay conditions were well known in the literature of couplings for rapid mixing of Markov chains, and now for the first time, are used to imply efficient dynamic sampling algorithms. Consequently, we have efficient dynamic (approximate or perfect) sampling algorithms with $O(n)$ space cost and $O(\log^2 n)$ incremental time cost, for the following models when the maximum degree is bounded:

- general MRF satisfying the Dobrushin-Shlosman condition (for approximate sampling);
- Ising model with temperature $\beta$ where $\exp(-2|\beta|) > 1 - \frac{2}{\Delta+1}$ (for both approximate and perfect samplings);
- hardcore model with fugacity $\lambda < \frac{-2}{\Delta-2}$ (for both approximate and perfect samplings);
- proper $q$-coloring with: $q > 2\Delta$ (for approximate sampling); or $q > 2\Delta^2 + 3\Delta$ (for perfect sampling).

These results show that the coupling of single-site Markov chains that have been widely used for implying efficient static sampling algorithms with near-linear running time, is also good for implying efficient dynamic sampling algorithms with sub-linear incremental costs.
## Contents

1 Introduction .................................................. 1

2 Dynamic Sampling Problem .................................. 3

3 Main Results .................................................. 4
   3.1 Dynamic approximate sampling .......................... 5
   3.2 Dynamic perfect sampling ............................... 6
   3.3 Implication on static CFTP ............................. 7

4 Technique Overview .......................................... 8

5 Preliminaries in Static Sampling ............................ 9
   5.1 Local neighborhood ..................................... 9
   5.2 Gibbs sampling .......................................... 9
   5.3 Coupling for mixing time ............................... 9
   5.4 Coupling from the Past (CFTP) ......................... 10

6 Dynamic Gibbs Sampling ....................................... 13
   6.1 Coupling for dynamic instances ....................... 14
   6.2 Data structure for single-site dynamics ............... 17
   6.3 The Dynamic Gibbs sampling algorithm ............... 19

7 Dynamic CFTP ................................................ 21
   7.1 CFTP expressed as single-site dynamics ............... 21
   7.2 Dynamic perfect sampling via CFTP ................. 22

8 Proofs for Dynamic Gibbs Sampling ......................... 26
   8.1 Analysis of the coupling ............................... 26
   8.2 Implementation of the algorithm ..................... 28
   8.3 Applications ............................................ 31

9 Proofs for Dynamic CFTP ..................................... 34
   9.1 Monotone systems ...................................... 34
   9.2 Anti-monotone systems ................................ 37
   9.3 Bounding chains ....................................... 47
   9.4 Applications ............................................ 50

10 Conclusion and Open Problems .............................. 55
1 Introduction

Sampling from Markov random fields (MRF) is a fundamental problem extensively studied in various fields, including: probability theory, statistical physics, machine learning, and theoretical computer science. The Markov random field is a basic graphical model that can encode exponentially sized probability distributions with succinct rules of pairwise interactions. Given a graph $G = (V, E)$, each vertex $v \in V$ is associated with a function $\phi_v : Q \to \mathbb{R}_{\geq 0}$ on a finite domain $Q = [q]$ of $q$ spin states, and each edge $e \in E$ is associated with a symmetric function $\phi_e : Q^2 \to \mathbb{R}_{\geq 0}$ which describes a pairwise interaction. Together, these induce a probability distribution $\mu$ over all configurations $\sigma \in Q^V$ as follows:

$$
\mu(\sigma) \propto \prod_{v \in V} \phi_v(\sigma_v) \prod_{e = \{u,v\} \in E} \phi_e(\sigma_u, \sigma_v).
$$

This distribution $\mu$ is known as the Gibbs distribution. It arises naturally from various physical models, statistics or learning tasks, and combinatorial problems in computer science.

The Markov chain Monte Carlo (MCMC) methods are the primary tools for sampling from Gibbs distributions. Many rich and profound theories have thus been developed on the subject (see e.g. [22, 20]). There are two major classes of MCMC algorithms for sampling from Gibbs distributions. They are: approximate sampling algorithms via rapidly mixing Markov chains, represented by the Gibbs sampling (a.k.a. heat-bath, Glauber dynamics); and algorithms for sampling exactly from the stationary distributions of Markov chains, represented by the coupling from the past (CFTP) method.\footnote{Though sampling exactly from a distribution technically is not Monte Carlo, the CFTP method is usually considered as a MCMC algorithm because it is based on mixings of Markov chains.} For both classes of algorithms, coupling is the primary technique for obtaining near-linear upper bounds on running time.

These classic MCMC sampling algorithms assume a static setting, where the input instance is not allowed to be changing dynamically. Once even a small and local change has been made to the input graphical model, the sampling algorithm has to be rerun from the beginning to guarantee the correctness of sampling.

With the advent of Big Data, the explosive growth in the volume of data and the increasing diversity of applications pose important challenges to the classic static algorithms for MCMC sampling. For example, in image processing applications [21, 1], the MRF defined by a 4k picture may have tens of millions of variables and constraints; and a 4k video may consist of a sequence of such big graphical models that could be very close to each other, where redoing the sampling from scratch seems to be a waste. Also, real systems for scalable machine learning have asked specifically for sampling from dynamically changing Gibbs distributions in their implementation (see e.g. [20, Section 5.3]).

In this paper, we study the problem of sampling from a MRF when the MRF itself is changing dynamically with time. For instance, at each time, a vertex or edge could be inserted or deleted, or a function $\phi_v$ or $\phi_e$ could change. The problem of dynamic sampling asks to maintain with low incremental cost, a sample for such dynamically changing MRF. Although one can imagine that the problem may have a lot of significances in practice, theoretical study of the problem had been largely lacking until it was initiated very recently [9]. There, a perfect sampling algorithm was given for sampling exactly from graphical models defined with dynamically changing soft constraints. The sampling algorithm was based on the idea of partial rejection sampling (PRS) [12] which is very different from the traditional MCMC methods. And the regimes for such dynamic sampling algorithm to be efficient with small incremental cost against dynamic inputs, are much more restrictive than the known regimes for efficient static sampling algorithms by rapidly mixing Markov chains.
A main quest for the studies of dynamic sampling is then to connect dynamic sampling to the well developed theories of MCMC sampling, therefore the techniques for MCMC sampling could be applied to solve dynamic sampling in a systematic way. However, it is unclear whether this is achievable. For example, in the static setting, sampling almost uniform graph coloring is among the first problems where the modern MCMC theory was developed [19, 27]. In contrast, in the dynamic setting, there is no algorithm known for efficiently sampling graph coloring from dynamically changing graphs with small incremental costs. The problem imposed a barrier for the techniques used in [9], and was considered a major open problem for dynamic sampling.

Unlike in the static setting, where the tractability of sampling is known to be captured by the temporal or spatial mixing properties [31, 23, 28, 25, 8], much less was known about the relation between dynamic sampling and mixing properties. It is unclear whether there should even exist such a way that can always transform with small cost a sample from the original probability space to a new sample from the updated probability space, as long as both spaces themselves are well connected.

Our results. In this paper, we give dynamic sampling algorithms based on the MCMC methods, for both approximate and perfect samplings. In particular, we give dynamic versions of Gibbs sampling and CFTP, which are the two major approaches respectively for sampling approximately and exactly from Gibbs distributions. We show that these dynamic sampling algorithms are efficient when the single-site coupling exhibits certain types of step-wise decay, a sufficient condition that has been used widely for implying rapid mixing of Markov chains and the efficiency of static MCMC sampling.

For dynamic Gibbs sampling, we show that there is an algorithm that maintains an approximate sample for a dynamically changing MRF with \( n \) vertices and bounded maximum degree, using \( \tilde{O}(n) \) space and \( \tilde{O}(\log^2 n) \) incremental time cost for each update, as long as the single-site coupling of Gibbs sampling is decaying in each step, and the metric in which the coupling is decaying is Lipschitz and approximately dominating the Hamming distance. This condition was established variously in existing coupling arguments in the literature. In particular, it is implied by the famous Dobrushin-Shlosman condition [4, 5, 3, 15, 6] for the decay of correlation.

For dynamic CFTP, we give the dynamic versions for all three major implementations of CFTP, namely: CFTP for monotone systems, for anti-monotone systems, and CFTP with bounding chains. For all these three implementations, we show that there is a dynamic perfect sampling algorithm with the same \( \tilde{O}(n) \) space cost and \( \tilde{O}(\log^2 n) \) incremental time cost as above, either under essentially the same condition as above for the Gibbs sampling, or under essentially the same coupling condition which was used to imply the efficiency of static CFTP.

These sufficient conditions have been established in the literature on various specific models, which directly gives us efficient dynamic sampling algorithms, approximate and perfect, with \( \tilde{O}(n) \) space cost and \( \tilde{O}(\log^2 n) \) incremental time cost on graphs with \( n \) vertices and bounded maximum degree, for the following models:

- for Ising model with temperature \( \beta \) satisfying \( e^{-2|\beta|} > 1 - \frac{2}{\Delta+1} \), which is close to the uniqueness threshold \( e^{-2|\beta_c|} = 1 - \frac{2}{\Delta} \), beyond which static sampling from anti-ferromagnetic Ising model is intractable [11, 10];

- for hardcore model with fugacity \( \lambda < \frac{2}{\Delta-2} \), which matches the best bound known for static sampling algorithm with near-linear running time on general graphs with bounded maximum degree [30, 24, 7];

- for proper \( q \)-coloring: \( q > 2\Delta \) (for dynamic approximate sampling); \( q > 2\Delta^2 + 3\Delta \) (for dynamic perfect sampling).
The above regimes for Ising and hardcore models greatly improve the previous regimes for dynamic sampling in [9]. Meanwhile, the results for proper \( q \)-coloring are the first (approximate or perfect) dynamic sampling algorithms for coloring.

Our techniques are based on: (1) couplings for dynamic instances of graphical models; and (2) dynamic data structures for representing single-site Markov chains so that the couplings can be realized algorithmically in sub-linear time. Both these techniques are of independent interests, and can be naturally extended to more general settings with multi-body interactions.

Our results also have implications on static MCMC sampling. A by-product of our analysis, is that we unify the analysis of CFTP for monotone or anti-monotone systems to the analysis of coupling of single-site chains, which greatly simplifies the analysis of CFTP. Previously such unified framework was only known for monotone systems.

**Organization of the paper.** In Section 2, we formally introduce the dynamic sampling problem. In Section 3, we formally states the main results. In Section 5, we review the preliminaries in static MCMC sampling. In Section 6 and Section 7, we present the algorithms for dynamic Gibbs sampling and dynamic CFTP, respectively. And the technical proofs of these two sections are postponed to Section 8 and Section 9, respectively. At last in Section 10, we give the conclusion and open problems.

## 2 Dynamic Sampling Problem

The dynamic sampling problem asks to maintain a sample \( X \) for the distribution \( \mu^T \) specified by the input \( T \), such that the total variation between \( \mu^T \) and the distribution of \( X \) has

\[
d_{TV}(X, \mu^T) \leq \epsilon. \tag{1}
\]

And in connection with any update of \( T \) to a \( T' \), we need to respond with the corresponding updates for \( X \), so that (1) still holds on the new sample \( X' \) for the new distribution \( \mu^{T'} \). When \( \epsilon = 0 \), the sampling is perfect; otherwise it is approximate.

In particular, we consider distributions defined by Markov random fields. An instance of Markov random field (MRF) is specified by a tuple \( T = (V, E, Q, \Phi) \), where \( G = (V, E) \) forms an undirected simple graph; \( Q \) is a domain of \( q = |Q| \) spin states (or just spins), where \( q = O(1) \); and \( \Phi = (\phi_a)_{a \in V \cup E} \) associates each vertex \( v \in V \) a function \( \phi_v : Q \to \mathbb{R}_{\geq 0} \) and each edge \( e \in E \) a symmetric function \( \phi_e : Q^2 \to \mathbb{R}_{>0} \). The functions \( (\phi_a)_{a \in V \cup E} \) are also called constraints or factors. And each vertex \( v \in V \) corresponds to a variable. The Gibbs distribution \( \mu^T \) is defined over \( Q^V \), such that for each configuration \( \sigma \in Q^V \),

\[
\mu(\sigma) \propto \prod_{v \in V} \phi_v(\sigma_v) \prod_{e = (u, v) \in E} \phi_e(\sigma_u, \sigma_v).
\]

Let \( \Omega_T \triangleq \{ \sigma \in Q^V \mid \mu^T(\sigma) > 0 \} \) denote the space of feasible configurations.

To trivialize the problem of constructing a feasible configuration, we further assume that the following natural condition always holds for the MRF instances considered in the paper:

\[
\forall v \in V, \quad \forall \tau \in Q^\Gamma_v : \quad \sum_{c \in Q} \phi_v(c) \prod_{u \in \Gamma_v} \phi_{uv}(\tau_u, c) > 0, \tag{2}
\]

where \( \Gamma_v \triangleq \{ u \in V \mid \{u, v\} \in E \} \) denotes the neighborhood of \( v \).

\footnote{This condition guarantees that the problem of constructing a feasible solution \( \sigma \in \Omega_T \) is trivial. The condition always holds for MRF with soft constraints, or hard constraints with a permissive spin, e.g. the hardcore model. And for MRF with truly repulsive constraints such as proper \( q \)-coloring, the condition may translate to the condition \( q \geq \Delta + 1 \) for the triviality of satisfiability, while the irreducible condition for coloring is \( q \geq \Delta + 2 \).}
Important examples of MRF. The followings are some well studied MRFs:

- Ising model: The domain of spins is $Q = \{-1, +1\}$. Each edge $e \in E$ is associated with a temperature $\beta \in \mathbb{R}$; and each vertex $v \in V$ is associated with a local field $h_v \in \mathbb{R}$. For each configuration $\sigma \in \{-1, +1\}^V$, $\mu_I(\sigma) \propto \exp \left( \sum_{\{u,v\} \in E} \beta \sigma_u \sigma_v + \sum_{v \in V} h_v \sigma_v \right)$.

- Hardcore model: The domain of spins is $Q = \{0, 1\}$. Each configuration $\sigma \in Q^V$ indicates an independent set in $G = (V, E)$, and $\mu_I(\sigma) \propto \lambda^{\|\sigma\|}$, where $\lambda > 0$ is a fugacity parameter.

- proper $q$-coloring: $\mu_I$ is the uniform distribution over all proper $q$-colorings of $G = (V, E)$.

Dynamic MRF. A MRF instance $\mathcal{I} = (V, E, Q, \Phi)$ may be updated to a new instance $\mathcal{I}' = (V', E', Q, \Phi')$ by one of the following update operations:

- addition/deletion of an independent variable
  - Vertex-Add$(v, \phi_v)$, which adds a new vertex $v \not\in V$ whose function is $\phi_v$;
  - Vertex-Delete$(v)$, which removes an isolated vertex $v \in V$ that has no neighbors;

- change of a constraint
  - Edge-Add$(e, \phi_e)$, which adds a new edge $e \not\in E$ whose function is $\phi_e$;
  - Edge-Delete$(e)$, which removes an existing edge $e \in E$;
  - Update$(a, \phi'_a)$, which renews the function $\phi_a$ of the vertex or edge $a \in V \cup E$ to $\phi'_a$.

For technical reason, we further require that when $a = v \in V$ there is at most one spin $c \in Q$ such that $\text{sgn}(\phi_a(c)) \neq \text{sgn}(\phi'_a(c))$ and when $a = e \in E$ there is at most one pair of spins $(c, c') \in Q^2$ such that $\text{sgn}(\phi_e(c, c')) \neq \text{sgn}(\phi'_e(c, c'))$.

Note that Update$(a, \phi'_a)$ requires that there is at most one entry between the original and the updated constraints $\phi_a$ and $\phi'_a$, whose value changing between zero and positive. General Update$(a, \phi'_a)$ with arbitrary new $\phi'_a$ can be realized without loss of generality by applying $O(1)$ such updates in a row since $q = |Q| = O(1)$. And we only consider the addition or deletion of an independent (isolated) variable: adding or deleting a variable along with incident constraints, can be realized by adding or deleting the involved variable and all constraints one by one.

Dynamic sampling from MRF. The problem of dynamic sampling for MRF asks to maintain a sample $X \in Q^V$ satisfying $d_{TV}(X, \mu_I) \leq \epsilon$ for the current MRF instance $\mathcal{I} = (V, E, Q, \Phi)$, with or without an extra data structure, such that upon each update among Vertex-Add, Vertex-Delete, Edge-Add, Edge-Delete and Update that modifies $\mathcal{I}$ to $\mathcal{I}'$, the algorithm updates the sample $X$ to a $X'$ (or equivalently, outputs the difference between $X$ and $X'$) that satisfies $d_{TV}(X', \mu_{I'}) \leq \epsilon$. We assume that the update request is determined by a non-adaptive adversary independently of current sample $X$ as well as the randomness used by the algorithm.

3 Main Results

In this section, we present our main results. Formalizing our results requires many concepts in classic MCMC sampling. Defining all these concepts in details would postpone the presentation of main discoveries too much. Therefore, instead, we defer the preliminaries to the next section while state the main results first, and refer to its formal definition when a concept is used in our statement. Readers who are not familiar with MCMC theory are advised to first go to Section 5 for preliminaries.
3.1 Dynamic approximate sampling

Our first dynamic sampling algorithm is a dynamic approximate sampler arising from Gibbs sampling (a.k.a. heat-bath, Glauber dynamics), which is a classic Markov chain for sampling from Gibbs distributions (formally defined later in Section 5.2). We show that its dynamic version exists and is efficient under the following condition.

**Condition 3.1 (mixing condition for dynamic sampling).** Let $\beta, C, K > 0$. Let $\mathcal{I} = (V, E, Q, \Phi)$ be an MRF instance with $n = |V|$, and $\Omega_\mathcal{I} \triangleq \{ \sigma \in Q^V \mid \mu_\mathcal{I}(\sigma) > 0 \}$ the feasible set. There exist a one-step local coupling (Definition 5.2) of Gibbs sampling on $\mathcal{I}$, denoted as $(X_t, Y_t)_{t \geq 0}$, and a potential function $\rho_\mathcal{I} : \Omega_\mathcal{I} \times \Omega_\mathcal{I} \to \mathbb{R}_{\geq 0}$, where $\forall \sigma, \tau \in \Omega_\mathcal{I}$, $\rho_\mathcal{I}(\sigma, \tau) = 0$ if $\sigma = \tau$ and $\rho_\mathcal{I}(\sigma, \tau) \geq 1$ if $\sigma \neq \tau$, and $\text{Diam}_\mathcal{I} \triangleq \max_{\sigma, \tau \in \Omega_\mathcal{I}} \rho_\mathcal{I}(\sigma, \tau) \leq \text{Poly}(n)$, such that

1. (step-wise decay) for the coupling $(X_t, Y_t)_{t \geq 0}$ of Gibbs sampling, it holds that

   $\forall \sigma, \tau \in \Omega_\mathcal{I} : \mathbb{E}[\rho_\mathcal{I}(X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] \leq \left(1 - \frac{3}{n}\right) \cdot \rho_\mathcal{I}(\sigma, \tau); \quad (3)$

2. (low-distortion to Hamming) for all $\sigma, \tau \in \Omega_\mathcal{I}$, $H(\sigma, \tau) \leq C \cdot \rho_\mathcal{I}(\sigma, \tau)$, where $H(\sigma, \tau)$ denotes the Hamming distance between $\sigma$ and $\tau$.

3. (Lipschitz) function $\rho_\mathcal{I}(\cdot, \cdot)$, seen as a function of $2n$ variables, is $K$-Lipschitz, that is,

   $\max_{\sigma, \sigma', \tau, \tau' \in \Omega_\mathcal{I}} |\rho_\mathcal{I}(\sigma, \tau) - \rho_\mathcal{I}(\sigma', \tau')| \leq K \cdot H(\sigma \tau, \sigma' \tau').$

The requirement of step-wise decay is just the classic condition for rapid mixing of Gibbs sampling (Proposition 5.1). In addition, Condition 3.1 further requires that the potential $\rho_\mathcal{I}$ in which the coupling is decaying, roughly dominates the Hamming distance and is Lipschitz.

**Theorem 3.2 (dynamic Gibbs sampling).** Assume that Condition 3.1 holds for the input MRF instance $\mathcal{I} = (V, E, Q, \Phi)$ and the updated instance $\mathcal{I}' = (V', E', Q, \Phi')$ with the same one-step local coupling, and the same parameters $\beta = \Omega(1)$, $C, K > 0$. Let $n = |V|$ and $\epsilon > \exp(-\text{O}(n))$.

There is a dynamic approximate sampling algorithm which maintains a sample $X \in Q^V$ for the current instance $\mathcal{I}$ satisfying $d_{TV}(X, \mu_\mathcal{I}) \leq \epsilon$, with an extra data structure of size $O(n \log \frac{n}{\epsilon})$ in memory words, each of $O(\log n)$ bits, such that when $\mathcal{I}$ is updated to $\mathcal{I}'$ by one of the operations among $\text{Vertex-Add}$, $\text{Vertex-Delete}$, $\text{Edge-Add}$, $\text{Edge-Delete}$ and $\text{Update}$, the algorithm updates the sample $X \in Q^V$ to a sample $X' \in Q^{V'}$ for $\mathcal{I}'$ satisfying $d_{TV}(X', \mu_{\mathcal{I}'}) \leq \epsilon$, with time cost

$$O\left(CK\Delta^2 \cdot \log n \cdot \log \left(\frac{n}{\epsilon}\right) \cdot \log \log \left(\frac{n}{\epsilon}\right)\right)$$

in expectation, where $\Delta = \Delta_G$ denotes the maximum degree of graph $G = (V, E)$.

The famous Dobrushin-Shlosman condition [4] (formally defined later in Definition 5.3) guarantees the step-wise decay of one-step optimal coupling in Hamming distance, which implies our Condition 3.1. Therefore we have the following corollary.

**Corollary 3.3.** Assume the Dobrushin-Shlosman condition (Definition 5.3) for the input MRF instance $\mathcal{I} = (V, E, Q, \Phi)$ and the updated instance $\mathcal{I}'$. Let $n = |V|$ and $\epsilon > \exp(-O(n))$.

There is a dynamic approximate sampling algorithm as stated in Theorem 3.2 with the same space cost $O\left(n \log \frac{n}{\epsilon}\right)$, and expected time cost $O\left(\Delta^2 \cdot \log n \cdot \log \left(\frac{n}{\epsilon}\right) \cdot \log \log \left(\frac{n}{\epsilon}\right)\right)$ for each update.
On specific models, our Condition 3.1 has been established in the literature of coupling for mixing time, which implies dynamic approximate sampling for these specific models.

In above, \( n = |V|, \epsilon > \exp(-n), \Delta = \max\{\Delta_G, \Delta_{G'}\} \) where \( G = (V, E) \) and \( G' = (V', E') \), \( \delta > 0 \) is an arbitrary constant, and the constant factors in \( O(\cdot) \) depend only on \( \delta \). In particular, the regimes for the Ising model and proper \( q \)-coloring are due to the Dobrushin-Shlosman condition (Definition 5.3). The regime for the hardcore model is obtained by verifying Condition 3.1 with parameters \( \beta = \Omega(1), C = O(1), K = O(\Delta) \), on the one-step optimal coupling and the potential function for the hardcore model due to Vigoda in [30].

### 3.2 Dynamic perfect sampling

We then give dynamic perfect sampling algorithms arising from coupling from the past (CFTP). The CFTP of Propp and Wilson [26] is one of the most important frameworks for exact sampling from Gibbs distributions (formally defined later in Section 5.4).

We need the following more restrictive notion of potential functions.

**Definition 3.4 (locally-defined potential function).** Let \( I = (V, E, Q, \Phi) \) be a MRF instance. A potential function \( \rho_I : Q^V \times Q^V \to \mathbb{R}_{\geq 0} \) is locally-defined if there is a family of local potential functions \( \rho_v : Q^{\Gamma_v^+} \times Q^{\Gamma_v^+} \to \mathbb{R}_{\geq 0} \), such that

\[
\forall \sigma, \tau \in Q^V : \quad \rho_I(\sigma, \tau) \triangleq \sum_{v \in V} \rho_v(\sigma(\Gamma_v^+), \tau(\Gamma_v^+)),
\]

where \( \Gamma_v^+ \triangleq \{ u \in V \mid \{ u, v \} \in E \} \cup \{ v \} \) denotes the inclusive neighborhood of \( v \in V \).

Such potential functions are widely used, for examples, the weighted Hamming distance, and the more complicated potential functions for hardcore model used in [30].

We also need the following operator on MRF instance.

**Definition 3.5.** Given a MRF instance \( I = (V, E, Q, \Phi) \) where \( \Phi = (\phi_u)_{u \in V \cup E} \), let \( \text{Par}(I) \) denote such an instance \( \text{Par}(I) = J = (V_J, E_J, Q, \Phi_J) \) constructed as following: \( V_J = V_1 \cup V_2 \), where \( V_1 \) and \( V_2 \) are duplicates of \( V \) such that each \( v \in V \) corresponds to a \( v_1 \in V_1 \) and a \( v_2 \in V_2 \); \( E_J = \{ \{ u_1, v_2 \}, \{ v_1, u_2 \} \mid \{ u, v \} \in E \} \); and \( \Phi_J \) assigns each \( v_1 \in V_1 \) or \( v_2 \in V_2 \) a function \( \phi_v \) and each \( \{ u_1, v_2 \}, \{ v_1, u_2 \} \in E_J \) a function \( \phi_e \) where \( e = \{ u, v \} \in E \).

Note that \( \text{Par}(I) \) does not change the maximum degree or the types of constraints of \( I \), thus \( \text{Par}(I) \) and \( I \) belong to the same class of MRF instances.

There are three major implementations of CFTP, namely: CFTP for monotone systems, CFTP for anti-monotone systems, and CFTP with bounding chains. The formal definitions of these algorithms are given later in Section 5.4. We show that dynamic versions of CFTP exist for all these implementations and the dynamic CFTP sampling algorithms are efficient either under Condition 3.1, or its slight refinement, or the known sufficient condition for the mixing of corresponding static CFTP algorithm.
Theorem 3.6 (dynamic CFTP sampling). Assume that one of the following conditions holds for the input MRF instance $\mathcal{I} = (V, E, Q, \Phi)$ and the updated instance $\mathcal{I}'$, where $n = |V|$:

- (for monotone system) $\mathcal{I}$ and $\mathcal{I}'$ are monotone with respect to a locally-defined grand coupling $g(\cdot, \cdot)$ (Definition 5.5), and Condition 3.1 holds for both $\mathcal{I}$ and $\mathcal{I}'$ with the same coupling $g(\cdot, \cdot)$ and the same parameters $\beta = \Omega(1)$, $C, K > 0$;

- (for anti-monotone system) $\mathcal{I}$ and $\mathcal{I}'$ are anti-monotone with respect to a locally-defined grand coupling $g(\cdot, \cdot)$ (Definition 5.5), and Condition 3.1 holds for both $\mathcal{J} = \operatorname{Par}(\mathcal{I})$ and $\mathcal{J}' = \operatorname{Par}(\mathcal{I}')$ with the same coupling $g(\cdot, \cdot)$ and the same parameters $\beta = \Omega(1)$, $C, K > 0$, and with locally-defined potential functions $\rho_\mathcal{J}$ and $\rho_\mathcal{J}'$;

- (for bounding chains) the mixing condition for bounding chains (Condition 5.6) holds for $\mathcal{I}$ with $\beta = \Omega(1)$ and $S = (2^Q \setminus \{\emptyset\})^V$, and also for $\mathcal{I}'$ with $\beta' = \Omega(1)$ and $S' = (2^Q \setminus \{\emptyset\})^{V'}$.

There is a dynamic perfect sampling algorithm which maintains a sample $X \sim \mu_\mathcal{I}$, with an extra data structure of size $O(n \log n)$ in memory words, each of $O(\log n)$ bits, such that when $\mathcal{I}$ is updated to $\mathcal{I}'$ by one of the operations among $\operatorname{Vertex-Add}$, $\operatorname{Vertex-Delete}$, $\operatorname{Edge-Add}$, $\operatorname{Edge-Delete}$ and $\operatorname{Update}$, the algorithm updates $X$ to $X' \sim \mu_{\mathcal{I}'}$, with time cost in expectation $O(CK\Delta^2 \cdot \log^2 n \cdot \log \log n)$ for monotone or anti-monotone systems, and $O(\Delta^2 \cdot \log^2 n \cdot \log \log n)$ for bounding chains, where $\Delta = \Delta_G$ denotes the maximum degree of graph $G = (V, E)$.

The grand coupling $g(\cdot, \cdot)$ referred in above theorem is a key notion for CFTP, and is formally defined in Definition 5.4. The one-step local coupling (formally defined in Definition 5.2) used in Condition 3.1 can be implied by such grand coupling. This is explained later in (7).

On specific models, the condition of Theorem 3.6 either has been established in the literature (for monotone/anti-monotone systems) or can be routinely verified (for bounding chains), which gives the following results for dynamic perfect sampling.

| dynamic perfect sampling | regime | space cost | incremental time cost |
|--------------------------|--------|------------|-----------------------|
| Ising model              | $e^{-2[\beta]} \geq 1 - \frac{2 - \delta}{\Delta + 1}$ | $O(n \log n)$ | $O(\Delta^2 \cdot \log^2 n \cdot \log \log n)$ |
| hardcore model           | $\lambda \leq \frac{2 - \delta}{\Delta - 2}$ | $O(n \log n)$ | $O(\Delta^3 \cdot \log^2 n \cdot \log \log n)$ |
| $q$-coloring             | $q \geq (2 + \delta)\Delta^2 + 3\Delta$ | $O(\Delta n \log n)$ | $O(\Delta^2 \cdot \log^2 n \cdot \log \log n + \Delta^3 \cdot \log n)$ |

In above, $n = |V|$, $\Delta = \max\{\Delta_G, \Delta_{G'}\}$ where $G = (V, E)$ and $G' = (V', E')$, $\delta > 0$ is an arbitrary constant, and the constant factors in $O(\cdot)$ depend only on $\delta$.

3.3 Implication on static CFTP

A by-product of our analysis of dynamic CFTP is that we unify the analysis of (static) CFTP for monotone or anti-monotone systems to the coupling of Gibbs sampling.

Theorem 3.7. Assume that one of the first two conditions of Theorem 3.6 for monotone or anti-monotone systems, holds on MRF instance $\mathcal{I} = (V, E, Q, \Phi)$ with $n = |V|$, The expected running time of the static CFTP algorithm on $\mathcal{I}$ is $O(n \log n)$.

Note that the low-distortion to Hamming and the Lipschitz condition in Condition 3.1 need not to be verified when applying this theorem, because the parameters $C$ and $K$ never show up.

Previously, such connection between mixing of CFTP and coupling of Gibbs sampling was only formalized for monotone systems [26], but not for anti-monotone systems.
4 Technique Overview

In [9], a dynamic sampling algorithm was proposed based on a local version of rejection sampling. A perfect sample $X$ for the current Gibbs distribution $\mu_I$ is maintained. When the graphical model $I$ is updated to $I'$, a new sample $Y$ for the updated Gibbs distribution $\mu_{I'}$ is obtained by resampling the variables within a region circulating the updated sites and also passing certain cleverly designed local filtration rules. This technique is restricted to exact sampling. It deviates very much from the well-developed MCMC theory. Furthermore, the technique does not work for Gibbs distributions defined by truly repulsive hard constraints, e.g. uniform proper coloring.

Alternatively, our approach in this paper is closely related to the MCMC sampling. Our results can be seen as an addition to the current MCMC theory in the dynamic setting.

Imagine that $(X_t)_{t=0}^T$ is a rapidly mixing single-site dynamics for sampling from the Gibbs distribution $\mu_I$, with an arbitrary initial configuration $X_0 \in Q^V$ and suitably large $T$, so that $X = X_T$ is a good enough approximate sample for $\mu_I$. As a single-site dynamics, at each step $t$, $X_{t-1}$ and $X_t$ may differ from each other only at a vertex $v_t$ which is picked from $V$ uniformly and independently at random. Therefore, the transition of the chain at time $t$ is fully captured by this pair $\langle v_t, X_t(v_t) \rangle$ of the vertex picked at time $t$ and its updated value.

Now suppose that instance $I$ is updated to $I'$ by some local update. We construct such a coupling between the original dynamics $(X_t)_{t=0}^T$ and the new dynamics $(Y_t)_{t=0}^T$, such that $(Y_t)_{t=0}^T$ is a faithful single-site dynamics for the updated instance $I'$ and $(Y_t)_{t=0}^T$ can be generated from $(X_t)_{t=0}^T$ by an efficient algorithm with small incremental cost, when the dynamics itself is stored by some suitable data structure. For simplicity of exposition, we first restrict ourselves to the cases where the update to the instance $I$ does not change the set of variables. Then the new dynamics $(Y_t)_{t=0}^T$ can be coupled with $(X_t)_{t=0}^T$ by using the same initial configuration $Y_0 = X_0$ and the same sequence $v_1, v_2, \ldots, v_T \in V$ of randomly picked vertices. And for $t = 1, 2, \ldots, T$, the transition $\langle v_t, Y_t(v_t) \rangle$ of the new dynamics can be constructed by using the same vertex $v_t$ as in $\langle v_t, X_t(v_t) \rangle$, and the $Y_t(v_t)$ generated according to a coupling of the marginal distributions of $X_t(v_t)$ and $Y_t(v_t)$, respectively conditioning on $X_{t-1}(\Gamma(v_t))$ and $Y_{t-1}(\Gamma'(v_t))$, where $\Gamma(v_t)$ and $\Gamma'(v_t)$ denote the respective neighborhoods of $v_t$ in instances $I$ and $I'$. Note that these two marginal distributions must be identical unless $X_{t-1}$ and $Y_{t-1}$ differ from each other over the neighborhood of $v_t$ or the $v_t$ itself is incident to where the models $I$ and $I'$ differ.

When the above coupling exhibits the decay as stated in Condition 3.1, the percolation of disagreements between $(X_t)_{t=0}^T$ and $(Y_t)_{t=0}^T$ is bounded, and we can almost always have the transitions identically coupled as $\langle v_t, X_t(v_t) \rangle = \langle v_t, Y_t(v_t) \rangle$, with only about $O(\Delta T/n)$ exceptions on average. The original dynamics $(X_t)_{t=0}^T$ can then be edited to the new dynamics $(Y_t)_{t=0}^T$ by editing these $O(\Delta T/n)$ local transitions $\langle v_t, Y_t(v_t) \rangle$ which are different from $\langle v_t, X_t(v_t) \rangle$. The single-site dynamics $(X_t)_{t=0}^T$ is represented by a data structure which literally stores the initial configuration $X_0$ and all transitions $\langle v_t, X_t(v_t) \rangle$, and can efficiently resolve all necessary queries needed to implement the above coupling, e.g. evaluating $X_t(u)$ for an arbitrary vertex $u$, or looking for the next $t$ where $X_t(v_t)$ and $Y_t(v_t)$ may not be equal. When we further consider the updates that may change the vertex set by inserting or deleting vertex, this data structure needs to be more robust to adapt to dynamically changing vertex set. This efficient dynamic data structure for single-site dynamics is of independent interests.

The approach described above gives a dynamization of the MCMC sampling. In fact, only the mixing part of the sampling is Monte Carlo, while the dynamization part is Las Vegas: the dynamic algorithm knows when it terminates and when terminates, it guarantees to produce a faithful single-site dynamics $(Y_t)_{t=0}^T$ for the new instance $I'$. By interpreting the perfect sampling strategies, the coupling from the past (CFTP) methods, as single-site dynamics, our technique can also give the dynamic version of CFTP.
5 Preliminaries in Static Sampling

5.1 Local neighborhood

Let $G = (V, E)$ be a graph. For any vertex $v \in V$, let $\Gamma_G(v) \triangleq \{ u \in V \mid \{u, v\} \in E \}$ denote the neighborhood of $v$, and $\Gamma_G^+(v) \triangleq \Gamma_G(v) \cup \{v\}$ the inclusive neighborhood of $v$. We simply write $\Gamma_v = \Gamma(v) = \Gamma_G(v)$ and $\Gamma_v^+ = \Gamma^+ (v) = \Gamma_G^+(v)$ for short when $G$ is clear in the context. We use $\Delta = \Delta_G \triangleq \max_{v \in V} |\Gamma_v|$ to denote the maximum degree of graph $G$.

A notion of local neighborhood for MRF is frequently used. Let $\mathcal{I} = (V, E, Q, \Phi)$ be a MRF instance. For $v \in V$, we denote by $\mathcal{I}_v \triangleq \mathcal{I}[\Gamma_v^+]$ the restriction of $\mathcal{I}$ on the inclusive neighborhood $\Gamma_v^+$ of $v$, i.e. $\mathcal{I}_v = (\Gamma_v^+, E_v, Q_v, \Phi_v)$, where $E_v = \{\{u, v\} \in E\}$ and $\Phi_v = (\phi_a)_{a \in \Gamma_v^+ \cup E_v}$.

5.2 Gibbs sampling

The Gibbs sampling (a.k.a. heat-bath, Glauber dynamics), is a classic Markov chain for sampling from Gibbs distributions. Let $\mathcal{I} = (V, E, Q, \Phi)$ be a MRF instance and $\mu = \mu_{\mathcal{I}}$ its Gibbs distribution. The chain of Gibbs sampling (Algorithm 1) is on the space $\Omega_{\mathcal{I}} \triangleq \{ \sigma \in Q^V \mid \mu_{\mathcal{I}}(\sigma) > 0 \}$ of feasible configurations, and has the stationary distribution $\mu_{\mathcal{I}}$ [22, Chapter 3].

| Algorithm 1: Gibbs sampling |
|----------------------------|
| **Initialization:** a feasible initial state $X_0 \in \Omega_{\mathcal{I}}$; |
| 1 for $t = 1, 2, \ldots, T$ do |
| 2 pick $v_t \in V$ uniformly at random; |
| 3 draw a random value $c \in Q$ from the marginal distribution $\mu_v(\cdot \mid X_{t-1}(\Gamma_v))$; |
| 4 $X_t(v_t) \leftarrow c$ and $X_t(u) \leftarrow X_{t-1}(u)$ for all $u \in V \setminus \{v_t\}$; |
| **Marginal distributions.** Here $\mu_v(\cdot \mid \sigma(\Gamma_v)) = \mu_{v, \mathcal{I}}(\cdot \mid \sigma(\Gamma_v))$ denotes the marginal distribution at $v \in V$ conditioning on $\sigma(\Gamma_v)$, which is computed as: |
| $\forall c \in Q : \quad \mu_v(c \mid \sigma(\Gamma_v)) = \frac{\phi_v(c) \prod_{u \in \Gamma_v} \phi_{uv}(\sigma_u, c)}{\sum_{c' \in Q} \phi_v(c') \prod_{u \in \Gamma_v} \phi_{uv}(\sigma_u, c')}$. (5) |
| Due to the assumption (2), this marginal distribution is always well defined, and its computation uses only the information of $\mathcal{I}_v$. |

5.3 Coupling for mixing time

Consider a chain $(X_t)_{t=0}^\infty$ on space $\Omega_{\mathcal{I}}$ with stationary distribution $\mu_{\mathcal{I}}$ for MRF instance $\mathcal{I}$. The mixing rate is defined as: for $\epsilon > 0$, $\tau_{\text{mix}}(\mathcal{I}, \epsilon) \triangleq \max X_0 \min \{ t \mid d_{TV}(X_t, \mu_{\mathcal{I}}) \leq \epsilon \}$, where $d_{TV}(X_t, \mu_{\mathcal{I}})$ denotes the total variation distance between $\mu_{\mathcal{I}}$ and the distribution of $X_t$.

A coupling of a Markov chain is a joint process $(X_t, Y_t)_{t \geq 0}$ such that $(X_t)$ and $(Y_t)$ marginally follow the same transition rule as the original chain. The following is well known.

**Proposition 5.1** ([19]). Let $\mathcal{I} = (V, E, Q, \Phi)$ be an MRF instance with $n = |V|$, and $\Omega_{\mathcal{I}} \subseteq Q^V$ the feasible set. If there exist a coupling $(X_t, Y_t)_{t \geq 0}$ of Gibbs sampling on $\mathcal{I}$ and a potential function $\rho_{\mathcal{I}}: \Omega_{\mathcal{I}} \times \Omega_{\mathcal{I}} \rightarrow \mathbb{R}_{\geq 0}$, where $\forall \sigma, \tau \in \Omega_{\mathcal{I}}$, $\rho_{\mathcal{I}}(\sigma, \tau) = 0$ if $\sigma = \tau$ and $\rho_{\mathcal{I}}(\sigma, \tau) \geq 1$ if $\sigma \neq \tau$, and $\text{Diam}_{\mathcal{I}} \triangleq \max_{\sigma, \tau \in \Omega_{\mathcal{I}}} \rho_{\mathcal{I}}(\sigma, \tau)$, such that

$\forall \sigma, \tau \in \Omega_{\mathcal{I}} : \quad \mathbb{E} [\rho_{\mathcal{I}}(X_t, Y_t) \mid X_{t-1} = \sigma \wedge Y_{t-1} = \tau] \leq \left(1 - \frac{\beta}{n}\right) \cdot \rho_{\mathcal{I}}(\sigma, \tau),$

then the mixing rate of Gibbs sampling on $\mathcal{I}$ is bounded as $\tau_{\text{mix}}(\mathcal{I}, \epsilon) \leq \left[ \frac{n}{\beta} \log \frac{\text{Diam}_{\mathcal{I}}}{\epsilon} \right]$. 

9
The proposition has been extensively used for analyzing the mixing time. In particular, the following type of couplings are most widely used.

**Definition 5.2 (one-step local coupling for Gibbs sampling).** A coupling \((X_t, Y_t)_{t \geq 0}\) of Gibbs sampling on a MRF instance \(\mathcal{I} = (V, E, Q, \Phi)\) is a one-step local coupling if it is constructed as follows: For \(t = 1, 2, \ldots\),

1. pick the same random \(v_t \in V\), and let \((X_t(u), Y_t(u)) \leftarrow (X_{t-1}(u), Y_{t-1}(u))\) for all \(u \neq v_t\);
2. sample \((X_t(v_t), Y_t(v_t))\) from a coupling \(D^\sigma_{\mathcal{I}, t}(\cdot, \cdot)\) of the marginal distributions \(\mu_{v_t}(\cdot | \sigma)\) and \(\mu_{v_t}(\cdot | \tau)\) where \(\sigma = X_{t-1}(\Gamma_{v_t})\) and \(\tau = Y_{t-1}(\Gamma_{v_t})\).

**Specification of a coupling.** A one-step local coupling \((X_t, Y_t)_{t \geq 0}\) is fully specified by the couplings \(D^\sigma_{\mathcal{I}, t}(\cdot, \cdot)\) of marginal distributions \(\mu_{v_t}(\cdot | \sigma)\) and \(\mu_{v_t}(\cdot | \tau)\). More precisely, a one-step local coupling rule (which may be applied to a family of instances) is identified by such an oracle \(D^\cdot\) which given access to a local specification \(\mathcal{I}_v\) of a MRF instance on a \(\Gamma_v^+\) and any two configurations \(\sigma, \tau \in Q_{\deg(v)}\), returns a joint distribution \(D^\sigma_{\mathcal{I}, t}(\cdot, \cdot)\) that is marginally distributed as \(\mu_{v_t}(\cdot | \sigma)\) and \(\mu_{v_t}(\cdot | \tau)\), which as defined in (5), are also determined by \(\mathcal{I}_v\) and \(\sigma, \tau \in Q_{\deg(v)}\).

**Dobrushin-Shlosman condition.** In particular, when \(D^\sigma_{\mathcal{I}, t}(\cdot, \cdot)\) always gives the optimal coupling of \(\mu_{v_t}(\cdot | \sigma)\) and \(\mu_{v_t}(\cdot | \tau)\) that attains the maximum \(\Pr[\mathbf{x} = \mathbf{y}]\) for all couplings \((\mathbf{x}, \mathbf{y})\) of \(\mathbf{x} \sim \mu_{v_t}(\cdot | \sigma)\) and \(\mathbf{y} \sim \mu_{v_t}(\cdot | \tau)\), the consequent coupling \((X_t, Y_t)_{t \geq 0}\) of the chains is called the one-step optimal coupling for Gibbs sampling. With such a coupling and the Hamming distance as the potential function \(p_x\), the condition in Proposition 5.1 for rapid mixing boils down to the Dobrushin-Shlosman condition [4, 5, 3, 15, 6].

**Definition 5.3 (Dobrushin-Shlosman condition).** Let \(\mathcal{I} = (V, E, Q, \Phi)\) be a MRF instance with Gibbs distribution \(\mu = \mu_{\mathcal{I}}\). Let \(A_{\mathcal{I}} \in \mathbb{R}^{V \times V}_{\geq 0}\) be the influence matrix which is defined as

\[
A_{\mathcal{I}}(u, v) \triangleq \begin{cases} 
\max_{\sigma, \tau \in \Omega} \text{TV} (\mu_v(\cdot | \sigma), \mu_v(\cdot | \tau)), & \{u, v\} \in E, \\
0, & \{u, v\} \notin E,
\end{cases}
\]

where the maximum is taken over the set \(B_{u,v}\) of all pairs \((\sigma, \tau) \in Q^V \times Q^V\) that differ only at \(u\). A MRF instance \(\mathcal{I}\) is said to satisfy the Dobrushin-Shlosman condition if there is a constant \(\delta > 0\) such that

\[
\|A_{\mathcal{I}}\|_\infty = \max_{u \in V} \sum_{v \in V} A_{\mathcal{I}}(u, v) \leq 1 - \delta.
\]

The mixing time bound \(\tau_{\text{mix}}(\mathcal{I}, \epsilon) \leq \left\lceil \frac{n}{\delta} \log \frac{n}{\epsilon} \right\rceil\) holds for Gibbs sampling under this condition.

**5.4 Coupling from the Past (CFTP)**

Consider the Gibbs sampling \((X_t)_{t=0}^\infty\) for a MRF instance \(\mathcal{I} = (V, E, Q, \Phi)\) on state space \(\Omega_{\mathcal{I}}\). Let \(P_{\mathcal{I}} : \Omega \times \Omega \to [0, 1]\) denote the transition matrix for this chain.

Let \(f_{\mathcal{I}} : \Omega_{\mathcal{I}} \times V \times [0, 1] \to \Omega_{\mathcal{I}}\) be a function such that for all \(\mathbf{x}, \mathbf{x}' \in \Omega_{\mathcal{I}}\) and \(v \in V\) and \(r \in [0, 1]\) chosen uniformly and independently at random

\[
\Pr[f_{\mathcal{I}}(\mathbf{x}, v, r) = \mathbf{x}'] = P_{\mathcal{I}}(\mathbf{x}, \mathbf{x}').
\]

Function \(f_{\mathcal{I}}\) defines a grand coupling (or complete coupling) that simultaneously couples a one-step transition of the original chain \((X_t)_{t=0}^\infty\) starting from all possible initial states \(X_0 \in \Omega_{\mathcal{I}}\).

10
A fundamental result of Propp and Wilson [26] says the following: If we run the chains starting respectively from all possible $\mathbf{X}_\tau \in \Omega_I$ (that is, from the past), coupled by the grand coupling $f_I$ as $\mathbf{X}_t = f_I(\mathbf{X}_{t-1}, v_t, r_t)$ with the same $v_t \in \mathcal{V}$ and $r_t \in [0, 1]$ chosen uniformly and independently at random, and let $T > 0$ be the time for which all these chains have reached the same state $\mathbf{X}_T$ at time 0, then the distribution of such $\mathbf{X}_0$ is precisely the stationary distribution of the original chain $(\mathbf{X}_t)_{t=0}^\infty$, which is $\mu_I$. This is the main idea for the coupling from the past (CFTP). See [22, 18] for tutorials.

For Gibbs sampling (and more generally single-site dynamics), the grand coupling $f_I$ can be defined by local informations.

**Definition 5.4 (locally-defined grand coupling for Gibbs sampling).** A grand coupling $f_I : \Omega_I \times \mathcal{V} \times [0, 1] \to \Omega_I$ of Gibbs sampling on a MRF instance $\mathcal{I} = (\mathcal{V}, \mathcal{E}, \mathcal{Q}, \Phi)$ is locally defined if for any $v \in \mathcal{V}$ and $r \in [0, 1]$, $\mathbf{X}' = f_I(\mathbf{X}, v, r)$ is constructed as: $X'(V \setminus \{v\}) = X(V \setminus \{v\})$ and $X(v) = g_{I_v}(X(\Gamma_v), r)$, where $g_{I_v} : \Omega^{\mathcal{V}} \times [0, 1] \to \Omega$ satisfies that

$$\forall c \in \mathcal{Q}, \forall \sigma \in \mathcal{Q}^{\mathcal{V}'}, \quad \Pr_{r \in [0, 1]}[g_{I_v}(\sigma, r) = c] = \mu_v(c | \sigma).$$

(6)

Recall that $\mathcal{I}_v = \mathcal{I}[\mathcal{V}_v^+]$ denotes the restriction of the MRF on $\mathcal{V}_v^+$. A locally-defined grand coupling rule (which may be applied to a family of instances) is identified by this oracle $g(\cdot, \cdot)$, which given access to local specification $\mathcal{I}_v$ of a MRF instance on $\mathcal{V}_v^+$ and any configuration $X(\Gamma_v) \in \Omega^{\mathcal{V}_v}$, maps uniform random bits $r \in [0, 1]$ to a spin in $\mathcal{Q}$. We simply call this family of functions $g_{I_v}$ a locally-defined grand coupling, or just a grand coupling.

A locally-defined grand coupling $g_{I_v}(\cdot, \cdot)$ can naturally specify a one-step local coupling as in Definition 5.2, where the coupling $D^{\tau \sigma}_{I_v}$ of marginal distributions is specified as

$$\forall \sigma, \tau \in \mathcal{Q}^{\mathcal{V}'}, \forall c, c' \in \mathcal{Q}, \quad D^{\tau \sigma}_{I_v}(c, c') = \Pr_{r \in [0, 1]}[g_{I_v}(\sigma, r) = c \land g_{I_v}(\tau, r) = c'].$$

(7)

With such grand couplings, the CFTP can sample perfectly from the stationary distribution of the chain $(\mathbf{X}_t)_{t=0}^\infty$ once the coalescence of all the coupled chains can be checked efficiently. There are two major approaches for this: the ones relying on certain monotonicities of the systems [26, 13] and the bounding chains [17].

**Monotone or anti-monotone systems.** Suppose that there is a partial order $\preceq$ defined over spins in $\mathcal{Q}$, which induces a partial order $\preceq_\mathcal{S}$ over (partial) configurations such that for $\sigma, \tau \in \mathcal{Q}^{\mathcal{S}}$ where $\mathcal{S} \subseteq \mathcal{V}$, $\sigma \preceq_\mathcal{S} \tau$ iff $\sigma(v) \leq \tau(v)$ for all $v \in \mathcal{S}$. We further assume that there are $c_{\text{min}}, c_{\text{max}} \in \mathcal{Q}$ such that $c_{\text{min}} \leq c \leq c_{\text{max}}$ for all spins $c \in \mathcal{Q}$ and it always holds that $\phi_v(c_{\text{min}}), \phi_v(c_{\text{max}}) > 0$ for the constraint function $\phi_v$ associated with any vertex $v \in \mathcal{V}$. Therefore the global minimum $X_{\text{min}} = X_{I, \text{min}}$ and maximum $X_{\text{max}} = X_{I, \text{max}}$ in $\mathcal{Q}^\mathcal{V}$ can be naturally constructed as

$$\forall v \in \mathcal{V} : \quad X_{I, \text{min}}(v) \triangleq c_{\text{min}}, \quad X_{I, \text{max}}(v) \triangleq c_{\text{max}},$$

(8)

so that $X_{\text{min}} \leq X \leq X_{\text{max}}$ for all $X \in \mathcal{Q}^\mathcal{V}$.

**Definition 5.5 (monotone/anti-monotone systems).** A MRF instance $\mathcal{I} = (\mathcal{V}, \mathcal{E}, \mathcal{Q}, \Phi)$ is said to have the **monotone** or **anti-monotone** property with respect to a grand coupling $g(\cdot, \cdot)$ if

- (monotone) $X(\Gamma_v) \preceq Y(\Gamma_v) \implies g_{I_v}(X(\Gamma_v), r) \leq g_{I_v}(Y(\Gamma_v), r)$ and $\forall c \in \mathcal{E}, \phi_v(c_{\text{min}}, c_{\text{min}}), \phi_v(c_{\text{max}}, c_{\text{max}}) > 0$;

- (anti-monotone) $X(\Gamma_v) \preceq Y(\Gamma_v) \implies g_{I_v}(X(\Gamma_v), r) \geq g_{I_v}(Y(\Gamma_v), r)$ and $\forall c \in \mathcal{E}, \phi_v(c_{\text{min}}, c_{\text{max}}) > 0.$

11
Examples for monotone system include the ferromagnet Ising model and the $q > 1$ random cluster model [26]; and examples for anti-monotone system include the hard-core model, the anti-ferromagnet Ising model, and the $q < 1$ random cluster model [13].

With such monotonicities, the CFTP can be implemented easily as Algorithm 2.

**Algorithm 2:** CFTP for monotone (or anti-monotone) systems

1. Let $v = (v_t)_{t=-\infty}^0$ and $r = (r_t)_{t=-\infty}^0$ where each $v_t \in V$ and $r_t \in [0,1]$ are chosen uniformly and independently at random;
2. $T \leftarrow 1$;
3. repeat
   4. $X^+ \leftarrow X_{\text{max}}$ and $X^- \leftarrow X_{\text{min}}$;
   5. for $t = -T + 1$ to 0 do
      6. $X^\pm(v_t) \leftarrow g_{\Gamma(v_t)}(X^\pm(\Gamma(v_t)), r_t)$; \quad (or $X^\pm(v_t) \leftarrow g_{\Gamma(v_t)}(X^\mp(\Gamma(v_t)), r_t)$);
      7. $T \leftarrow 2T$;
   8. until $X^+ = X^-$;
9. return $X^+$;

**Bounding chains.** The bounding chains introduced independently by Huber [16, 17] and Häggström and Nelander [14] provide a powerful framework for implementing CFTP in more general settings beyond monotonicity or anti-monotonicity. The bounding chain, described in Algorithm 3, is a chain on states $\mathcal{X} \in (2^Q)^V$, where for each $v \in V$, $\mathcal{X}(v) \subseteq Q$ naturally gives the subset of possible spins of vertex $v$ in all coupled chains. Though there may be exponential possible $\sigma$ in $\otimes_{u \in \Gamma_G(v_t)} \mathcal{X}(u)$, the bounding chains are designed such that $\mathcal{X}(v_t)$ in line 6 can be calculated efficiently. To simplify the analysis of time complexity, without loss of generality, we assume that $\mathcal{X}(v_t)$ can be calculated in $O(\Delta)$ time if $q = O(1)$.

**Algorithm 3:** CFTP with bounding chains

1. Let $v = (v_t)_{t=-\infty}^0$ and $r = (r_t)_{t=-\infty}^0$ where each $v_t \in V$ and $r_t \in [0,1]$ are chosen uniformly and independently at random;
2. $T \leftarrow 1$;
3. repeat
   4. $\mathcal{X} \leftarrow Q^V$;
   5. for $t = -T + 1$ to 0 do
      6. $\mathcal{X}(v_t) \leftarrow \{ g_{\Gamma_G(v_t)}(\sigma, r_t) \mid \sigma \in \otimes_{u \in \Gamma_G(v_t)} \mathcal{X}(u) \}$;
      7. $T \leftarrow 2T$;
   8. until $|\mathcal{X}(v)| = 1$ for all $v \in V$;
9. return $\mathcal{X}_0$;

**Condition 5.6 (mixing condition for bounding chains,** rephrased from [17]). Let $\beta > 0$ and $S \subseteq (2^Q)^V$. Let $(\mathcal{X}_t, \mathcal{Y}_t)_{t \geq -T}$ be the coupling of bounding chain on $I = (V, E, Q, \Phi)$, coupled with the same $v = (v_t)_{t \geq -T}$ and $r = (r_t)_{t \geq -T}$. The following holds

$$\forall \sigma \in (2^Q \setminus \emptyset)^V, \tau \in S: \quad \mathbb{E}[H(\mathcal{X}_t, \mathcal{Y}_t) \mid \mathcal{X}_{t-1} = \sigma \land \mathcal{Y}_{t-1} = \tau] \leq (1 - \frac{\beta}{n}) \cdot H(\sigma, \tau), \quad (9)$$

where $H(\sigma, \tau) \triangleq \sum_{v \in V} \mathbb{P}(\sigma(v) \neq \tau(v))$ denotes the Hamming distance.
When analyzing a bounding chain (Algorithm 3), the \( S \) in above condition is always fixed in particular as the set of all collapsed states \( S = \{X : |X(v)| = 1 \text{ for all } v \in V\} \). Thus the decay in (9) would imply the decay of \( \{v \in V : |X_t(v)| > 1\} \) with the same rate, which implies a \( O(p \log n) \) expected running time for the bounding chain.

### 6 Dynamic Gibbs Sampling

In this section, we introduce the dynamic Gibbs sampling algorithm. The method is presented for general single-site dynamics. This is not just for the possibility of generalizing our result for Gibbs sampling to other dynamics, but rather, the results for CFTP introduced later is obtained by interpreting CFTP as single-site dynamics, and will rely on the result in this section.

Let \( \mathcal{I} = (V, E, Q, \Phi) \) be an instance of Markov random field (MRF). The single-site dynamics (or single-site chains) is a class of Markov chains described abstractly as Algorithm 4. The state space of the chain is \( \Sigma^V \), where \( \Sigma \) is an alphabet, typically \( \Sigma = Q \). In each transition of the chain, a vertex \( v \in V \) is picked uniformly at random and the value of \( X(v) \) is updated to a random value \( x \in \Sigma \) according to a marginal transition distribution \( P_{\mathcal{I}_v}(\cdot \mid X(\Gamma_v)) \), where \( P_{\mathcal{I}_v}(\cdot \mid X(\Gamma_v)) \) denotes abstractly a distribution over \( \Sigma \) determined by \( X(\Gamma_v) \). The transition rule from conditions \( \sigma \in \Sigma^{\Gamma_v} \) to the marginal distributions \( P_{\mathcal{I}_v}(\cdot \mid \sigma) \) over \( \Sigma \) is fully determined by the specification of \( \mathcal{I}_v \) which is the restriction of \( \mathcal{I} \) on \( \Gamma_v^+ \).

#### Algorithm 4: single-site dynamics

**Initialization:** a feasible initial state \( X_0 \in \Sigma^V \);

1. for \( t = 1, 2, \ldots, T \) do
2.  
3.  pick \( v_t \in V \) uniformly at random;
4.  
5.  draw a random value \( x \in \Sigma \) according to the distribution \( P_{\mathcal{I}_v}(\cdot \mid X_{t-1}(\Gamma_{v_t})) \);
6.  
7.  \( X_t(v_t) \leftarrow x \) and \( X_t(u) \leftarrow X_{t-1}(u) \) for all \( u \in V \setminus \{v_t\} \);

The Gibbs sampling is a single-site dynamics where the marginal transition distribution \( P_{\mathcal{I}_v}(\cdot \mid X(\Gamma_v)) = \mu_v(\cdot \mid X(\Gamma_v)) \) is the marginal distribution of the Gibbs distribution \( \mu = \mu_\mathcal{I} \).

Let \( \epsilon \geq 0 \) be an error bound. Suppose that \( T(\mathcal{I}, \epsilon) \) is an easy-to-compute integer-valued function that upper bounds the mixing time on instance \( \mathcal{I} \), such that \( T(\mathcal{I}, \epsilon) \geq \tau_{\text{mix}}(\mathcal{I}, \epsilon) \),

where \( \tau_{\text{mix}}(\mathcal{I}, \epsilon) \) denotes the mixing rate for the Gibbs sampling chain \( (X_t)_{t \geq 0} \) on instance \( \mathcal{I} \). By Condition 3.1, it holds that \( \text{Diam}_{\mathcal{I}} \triangleq \max_{\sigma, \tau \in \Omega_{\mathcal{I}}} \rho_{\mathcal{I}}(\sigma, \tau) \leq \text{Poly}(n) \). We assume there is a constant \( d = O(1) \) such that \( \text{Diam}_{\mathcal{I}} \leq nd^t \) for any instance \( \mathcal{I} \) satisfying Condition 3.1. By Proposition 5.1, if Condition 3.1 is satisfied, we can set

\[
T(\mathcal{I}, \epsilon) = \left\lceil \frac{n}{\beta} \log \frac{nd^t}{\epsilon} \right\rceil. \tag{11}
\]

**Dynamic Gibbs sampling:** Our algorithm for dynamic Gibbs sampling maintains a random process \( (X_t)_{t=0}^T \), which is a Gibbs sampling chain on instance \( \mathcal{I}' \) of length \( T = T(\mathcal{I}, \epsilon) \), where \( T(\mathcal{I}, \epsilon) \) satisfies (10). Clearly \( X_T \) is a sample for \( \mu_{\mathcal{I}} \) with \( d_{TV}(X_T, \mu_{\mathcal{I}}) \leq \epsilon \).

When the current instance \( \mathcal{I} \) is updated to a new instance \( \mathcal{I}' \), the original process \( (X_t)_{t=0}^T \) is transformed to a new process \( (Y_t)_{t=0}^T \) such that the following holds as an invariant: \( (Y_t)_{t=0}^T \) is a Gibbs sampling chain on \( \mathcal{I}' \) with \( T' = T(\mathcal{I}', \epsilon) \). Hence \( Y_T \) is a sample for the new instance \( \mathcal{I}' \) with \( d_{TV}(X_T, \mu_{\mathcal{I}'}) \leq \epsilon \). This is achieved in two steps:
1. We construct couplings between \((X_t)_{t=0}^T\) and \((Y_t)_{t=0}^T\), so that the new process \((Y_t)_{t=0}^T\) for \(\mathcal{I}'\) can be obtained by making small changes to the original process \((X_t)_{t=0}^T\) for \(\mathcal{I}\).

2. We give a data structure which represents \((X_t)_{t=0}^T\) incrementally and supports various updates and queries to \((X_t)_{t=0}^T\) so that the above coupling can be generated efficiently.

For generality, the framework is stated for general single-site dynamics, where the Gibbs sampling is covered as a special case.

### 6.1 Coupling for dynamic instances

As a single-site dynamics, the chain \((X_t)_{t=0}^T\) can be uniquely and fully recovered from: the initial state \(X_0 \in \Sigma^V\), and the pairs \((v_t, X_t(v_t))_{t=1}^T\) that record the transitions. We call \((v_t, X_t(v_t))_{t=1}^T\) the execution-log for the chain \((X_t)_{t=0}^T\) and in particular, when \(T = T(\mathcal{I}, e)\), denote

\[
\text{Exe-Log}(\mathcal{I}, e) \triangleq \langle v_t, X_t(v_t) \rangle_{t=1}^T.
\]

The following invariants are assumed for the random execution-log with an initial state.

**Condition 6.1 (invariants for Exe-Log).** Fixed an initial state \(X_0 \in \Sigma^V\), the followings hold for the random execution-log \(\text{Exe-Log}(\mathcal{I}, e) = \langle v_t, X_t(v_t) \rangle_{t=1}^T\) for the single-site chain \((X_t)_{t=0}^T\) on instance \(\mathcal{I} = (V, E, Q, \Phi)\):

- \(T = T(\mathcal{I}, e)\) where \(T(\mathcal{I}, e)\) satisfies (10);
- each \(v_t \in V\) is uniform and independent at random;
- the random process \((X_t)_{t=0}^T\) uniquely recovered from the transitions \((v_t, X_t(v_t))_{t=1}^T\) and the initial state \(X_0\), is identically distributed as the single-site dynamics (Algorithm 4) on instance \(\mathcal{I}\) starting from initial state \(X_0\) with \(v_t\) as the vertex picked at the \(t\)-th step.

Such invariants guarantee that \(X_T\) provides a sample for \(\mu_{\mathcal{I}}\) with \(d_{TV}(X_T, \mu_{\mathcal{I}}) \leq \epsilon\).

Suppose that a new instance \(\mathcal{I}'\) is obtained from the current instance \(\mathcal{I}\) by one of the following updates defined in Section 2: Vertex-Add, Vertex-Delete, Edge-Add, Edge-Delete, and Update. We construct couplings between the execution-log \(\text{Exe-Log}(\mathcal{I}, e) = \langle v_t, X_t(v_t) \rangle_{t=1}^T\) with initial state \(X_0 \in \Sigma^V\) for \(\mathcal{I}\) and the execution-log \(\text{Exe-Log}(\mathcal{I}', e) = \langle v'_t, Y_t(v'_t) \rangle_{t=1}^{T'}\) with initial state \(Y_0 \in \Sigma^V\) for \(\mathcal{I}'\). Our goal is: assuming Condition 6.1 for \(X_0\) and \(\text{Exe-Log}(\mathcal{I}, e)\), the same condition should hold invariantly for \(Y_0\) and \(\text{Exe-Log}(\mathcal{I}', e)\).

Unlike traditional coupling of Markov chains for the analysis of mixing time, where the two chains start from arbitrarily distinct initial states but proceed by the same transition rule, here the two chains \((X_t)_{t=0}^T\) and \((Y_t)_{t=0}^T\) may start from the same state but have to obey different transition rules due to differences between instances \(\mathcal{I}\) and \(\mathcal{I}'\).

#### 6.1.1 Coupling for constraint update

We first consider the following updates for constraint: Edge-Add\((e, \phi_e)\), Edge-Delete\((e)\), and Update\((a, \phi_a)\) where \(a \in V \cup E\). These updates do not change the set of variables. Upon such an update, the new instance becomes \(\mathcal{I}' = (V, E', Q, \Phi')\). We use \(S \subset V\) to denote the set of sites (vertices) affected by the update from \(\mathcal{I}\) to \(\mathcal{I}'\):

\[
S \triangleq \begin{cases} 
\{v\} & \text{update is Update}(v, \phi_v) \text{ for some vertex } v \in V, \\
\{u, v\} & \text{update is Edge-Add}(e, \phi_e), \text{ Edge-Delete}(e), \text{ or Update}(e, \phi_e) \text{ where } e = \{u, v\}.
\end{cases}
\]
Given an initial state $X_0 \in \Sigma^V$ and $\text{Exe-Log}(I, \epsilon) = \langle v_t, X_t(v_t) \rangle_{t=1}^{T'}$ for instance $I$, we first transform it to an execution-log $\langle v_t, X_t(v_t) \rangle_{t=1}^{T'}$ with new length $T' = T(I', \epsilon)$. This is done by either truncating the chain when $T' < T$ or simulating the the chain for more steps when $T' > T$. Formally, this is described in Algorithm 6 as $\text{LengthFix}(I, X_0, \langle v_t, X_t(v_t) \rangle_{t=1}^{T}, T')$.

We then transform this pair of $X_0 \in \Sigma^V$ and $\langle v_t, X_t(v_t) \rangle_{t=1}^{T'}$ to a new pair of $Y_0 \in \Sigma^V$ and $\text{Exe-Log}(I', \epsilon) = \langle v_t, Y_t(v_t) \rangle_{t=1}^{T'}$ for $I'$ that satisfies Condition 6.1. This is achieved as following: the vertex sequence $(v_t)_{t=1}^{T'}$ is identically coupled and the chain $(X_t)_{t=0}^{T}$ is transformed to $(Y_t)_{t=0}^{T}$ by the following one-step optimal coupling between $X$ and $Y$.

**Definition 6.2 (one-step local coupling for dynamic instances).** The two chains $(X_t)_{t=0}^{\infty}$ on instance $I$ and $(Y_t)_{t=0}^{\infty}$ on instance $I'$ are coupled as:

- Initially $X_0, Y_0 \in \Sigma^V$ are two feasible states that $X_0 \oplus Y_0 \leq S$;
- for $t = 1, 2, \ldots$, the two chains $X$ and $Y$ jointly do:
  1. pick the same $v_t \in V$, and let $(X_t(u), Y_t(u)) \leftarrow (X_{t-1}(u), Y_{t-1}(u))$ for all $u \in V \setminus \{v_t\}$;
  2. sample $(X_t(v_t), Y_t(v_t))$ from a coupling $D^\sigma_{\{\cdot, \cdot\}}$ of the marginal distributions $P_{\{\cdot\}} \mid \sigma$ and $P_{\{\cdot\}} \mid \tau$ with $\sigma = X_{t-1}(\Gamma(v_t))$ and $\tau = Y_{t-1}(\Gamma(v_t))$, where $\Gamma' = (V, E')$.

For now, $D^\sigma_{\{\cdot, \cdot\}}$ is just any coupling of marginal transition distributions $P_{\{\cdot\}} \mid \sigma$ and $P_{\{\cdot\}} \mid \tau$, to be specified later.

Obviously the resulting $(Y_t)_{t=0}^{T}$ is a faithful copy of the single-site chain on instance $I'$, assuming that $(X_t)_{t=0}^{T}$ is such a chain on instance $I$.

Let $D_t$ denote the set of disagreements between $X_t$ and $Y_t$. Formally:

$$D_t \equiv \{ v \in V \mid X_t(v) \neq Y_t(v) \}. \quad (13)$$

The following observation is easy to make for the $(X_t)_{t=0}^{T}$ and $(Y_t)_{t=0}^{T}$ coupled as above.

**Observation 6.3.** For any $t \in [1, T']$, if $v_t \notin S \cup \Gamma^+_G(D_{t-1})$ then $X_t(v_t) = Y_t(v_t)$ and $D_t = D_{t-1}$.

With this observation, the new $Y_0$ and $\text{Exe-Log}(I', \epsilon) = \langle v_t, Y_t(v_t) \rangle_{t=1}^{T'}$ can be generated from $X_0$ and $\text{Exe-Log}(I, \epsilon) = \langle v_t, X_t(v_t) \rangle_{t=1}^{T'}$ as Algorithm 5.

Observation 6.3 says that the nontrivial coupling between $X_t(v_t)$ and $Y_t(v_t)$ is only needed when $v_t \in S \cup \Gamma^+_G(D_{t-1})$, which occurs rarely as long as $D_{t-1}$ remains small. This is a key to ensure the small Polylog($n$) time cost of Algorithm 5. Formally, for the $(X_t)_{t=0}^{T}$ and $(Y_t)_{t=0}^{T}$ coupled as above, for any $1 \leq t \leq T'$, let $\gamma_t$ indicate whether this bad event occurs:

$$\gamma_t \equiv 1 \left[ v_t \in S \cup \Gamma^+_G(D_{t-1}) \right], \quad (14)$$

and let $R$ denote the number of occurrences of such bad events:

$$R \equiv \sum_{t=1}^{T'} \gamma_t. \quad (15)$$

We will see that $R$ dominates the cost of Algorithm 5, once a data structure is given to encode the execution-log and resolve the updates to the data in Line 10 and various queries (in Line 3, 4, and 6) to the data.

---

3For Gibbs sampling, the assumption in (2) guarantees that given any feasible $X_0 \in Q^V$ with $\mu_T(X_0) > 0$ there always exists a feasible $Y_0 \in Q^V$ with $\mu_T(Y_0) > 0$ such that $X_0 \oplus Y_0 \subseteq S$ and $Y_0$ can be constructed locally from $X_0$ with incremental cost at most $O(\Delta)$.
This is obviously a valid coupling of marginal distributions \( \mu(v \mid \sigma) \) and \( \mu(v \mid \tau) \) because for any \( v \notin S \), we have \( I_v = I_v' \) and hence the marginal distributions \( \mu_v(I_v \mid \sigma) \) and \( \mu_v(I_v \mid \tau) \) are defined by the same law as (5) on instance \( I_v \), thus can be coupled by \( D_{\mu_v}^\sigma(\cdot, \cdot) \).

Lemma 6.4 (cost of the coupling). Assume that the single-site chain is the Gibbs sampling, and Condition 3.1 holds for both \( \mathcal{I} \) and \( \mathcal{I}' \) with the same one-step local coupling (Definition 5.2) defined by a family of couplings \( D_{\mu_v}^\sigma(\cdot, \cdot) \) of marginal distributions \( \mu_v(\cdot \mid \sigma) \) and \( \mu_v(\cdot \mid \tau) \), where the rule of coupling is determined consistently by the specification of \( I_v \), where \( I_v \) denotes the restriction of \( \mathcal{I} \) on \( \Gamma_G(v) \). Then the one-step local coupling for dynamic instances in Definition 6.2 can be defined by a natural family of couplings \( D_{\mu_v}^\sigma(\cdot, \cdot) \) of heterogenous marginal distributions \( \mu_{v,I_v}(\cdot \mid \sigma) \) and \( \mu_{v,I_v}(\cdot \mid \tau) \), constructed as following:

\[
\forall \sigma \in Q^{\Gamma_G(v)}, \tau \in Q^{\Gamma_G(v)}: \quad D_{\mu_v}^{\sigma,\tau}(\cdot, \cdot) = \begin{cases} 
D_{\mu_v}^{\sigma,\tau}(\cdot, \cdot) & \text{if } v \notin S, \\
\mu_{v,I_v}(\cdot \mid \sigma) \times \mu_{v,I_v}(\cdot \mid \tau) & \text{if } v \in S.
\end{cases} \tag{16}
\]

This is obviously a valid coupling of marginal distributions \( \mu_{v,I_v}(\cdot \mid \sigma) \) and \( \mu_{v,I_v}(\cdot \mid \tau) \) because for any \( v \notin S \), we have \( I_v = I_v' \) and hence the marginal distributions \( \mu_{v,I_v}(\cdot \mid \sigma) \) and \( \mu_{v,I_v}(\cdot \mid \tau) \) are defined by the same law as (5) on instance \( I_v \), thus can be coupled by \( D_{\mu_v}^{\sigma,\tau}(\cdot, \cdot) \).
It holds that $\mathbb{E}[R] = O\left(\frac{\Delta T^3 C K}{n^3}\right)$, where $R$ is defined as (15) by the one-step local coupling for dynamic instances constructed as (16).

### 6.1.2 Coupling for addition or deletion of an independent variable

These updates are easy to deal with by couplings, however, they may cause overheads to the data structure and algorithm for realizing the couplings.

**Addition of a variable:** The update is $\text{Vertex-Add}(v^*, \phi_{v^*})$, where $v^* \notin V$ is a newly introduced isolated vertex. The original instance $\mathcal{I} = (V, E, Q, \Phi)$ is updated to $\mathcal{I}' = (V', E, Q, \Phi')$, where $V' = V \cup \{v^*\}$ and $\Phi' = \Phi \cup (\phi_{v^*})$.

Since the new instance $\mathcal{I}'$ is the same as $\mathcal{I}$ except the isolated vertex $v^*$, we can construct $Y_0(V) = X_0$ and $Y_0(v^*) \in \Sigma$ is arbitrary, and given an $\text{Exe-Log}(\mathcal{I}, \epsilon) = \langle v_t, X_t(v_t) \rangle_{t=1}^{T'}$, the new $\text{Exe-Log}(\mathcal{I}', \epsilon) = \langle v'_t, Y_t(v'_t) \rangle_{t=1}^{T'}$ can be constructed by inserting random appearances of $v^*$ into $(v_t)_{t=1}^{T'}$, while the $Y_t(v^*)$ at the inserted steps $t$ are sampled i.i.d. from the marginal transition distribution $P_{T,v^*}$. without any condition since $v^*$ is an isolated vertex, which in the case of Gibbs sampling is just a distribution over $Q$ proportional to $\phi_{v^*}$. Formally:

1. Let $T' = T(\mathcal{I}', \epsilon)$, and $P \subseteq [T'] \triangleq \{1, 2, \ldots, T'\}$ a random subset such that each $t \in [T']$ is selected into $P$ independently with probability $1/|V'|$. Denote $m \triangleq T' - |P|$ and enumerate all elements in $[T'] \setminus P$ as $\ell_1 < \ell_2 < \cdots < \ell_m$.

2. Let $\langle v_t, X_t(v_t) \rangle_{t=1}^{m} \leftarrow \text{LengthFix}(\mathcal{I}, X_0, \langle v_t, X_t(v_t) \rangle_{t=1}^{T'}, m)$.

3. Construct $\langle v'_t, Y_t(v'_t) \rangle_{t=1}^{T'}$ as follows:

\[
\forall t \in P: \quad v'_t = v^* \quad \text{and} \quad Y_t(v'_t) \sim P_{T,v^*}(
\rangle;

\forall t = \ell_k \in [T'] \setminus P: \quad v'_t = v_k \quad \text{and} \quad Y_t(v'_t) = X_k(v'_t) = X_k(v_k).
\]

It is easy to see that $(Y_t)_{t=0}^{T}$ is a faithful copy of the single-site chain on instance $\mathcal{I}'$.

**Deletion of a variable:** The update is $\text{Vertex-Delete}(v^*)$, where $v^* \in V$ is an isolated vertex in the current instance $\mathcal{I} = (V, E, Q, \Phi)$. Upon such update, the instance is updated to $\mathcal{I}' = (V', E, Q, \Phi')$, where $V' = V \setminus \{v^*\}$ and $\Phi' = \Phi \setminus (\phi_{v^*})$.

We can simply construct $Y_0 = X_0(V')$. The new execution-log $\text{Exe-Log}(\mathcal{I}', \epsilon) = \langle v'_t, Y_t(v'_t) \rangle_{t=1}^{T'}$ can be constructed from the original $\text{Exe-Log}(\mathcal{I}, \epsilon) = \langle v_t, X_t(v_t) \rangle_{t=1}^{T}$ by simply deleting all appearances of $v^*$ in $(v_t)_{t=1}^{T}$ and the corresponding trivial transitions $X_t(v^*)$, followed by calling $\text{LengthFix}$ on instance $\mathcal{I}'$ with the new length $T' = T(\mathcal{I}', \epsilon)$ to properly truncate/append the chain to the length $T'$.

It is easy to see that $(Y_t)_{t=0}^{T}$ is a faithful copy of the single-site chain on instance $\mathcal{I}'$.

### 6.2 Data structure for single-site dynamics

We now describe an efficient data structure for single-site dynamics $(X_t)_{t=0}^{T}$. Let $\mathcal{I} = (V, E, Q, \Phi)$ be an MRF instance. The data structure should provide the following functionalities.

- **Data:** an initial state $X_0 \in \Sigma^V$ and an execution-log $\langle v_t, X_t(v_t) \rangle_{t=1}^{T} \in (V \times \Sigma)^T$ that records the $T$ transitions of the single-site dynamics $(X_t)_{t=0}^{T}$;

- **updates:**
- \text{Insert}(t, v, c)$, which inserts a transition $(v, c)$ after the $(t-1)$-th transition $(v_{t-1}, X_{t-1}(v_{t-1}))$;
- \text{Remove}(t)$, which deletes the $t$-th transition $(v_t, X_t(v_t))$;
- \text{Change}(t, c)$, which changes the $t$-th transition $(v_t, X_t(v_t))$ to $(v_t, c)$;

Note that the updates \text{Insert}(t, v, c)$ and \text{Remove}(t)$ change the length $T$ of the chain, as well as the order-numbers of all transitions after the inserted/deleted transition.

- \text{queries}:
  - \text{Eval}(t, v)$, which returns the value of $X_t(v)$ for arbitrary $t$ and $v$ (not necessarily $= v_t$);
  - \text{Succ}(t, v)$, which returns $i$ for the smallest $i > t$ such that $v_i = v$ if such $i$ exists, or returns $\bot$ if otherwise.

It is not difficult to realize that the query \text{Eval}(t, v)$ can actually be solved by a predecessor search defined symmetrically to \text{Succ}(t, v)$. This data structure problem for single-site dynamics is quite natural and is of independent interests.

**Theorem 6.5 (Data Structure for Single-Site Dynamics).** There exists a deterministic dynamic data structure which stores an arbitrary initial state $X_0 \in \Sigma^V$ and an execution-log $(v_t, X_t(v_t))_{t=1}^T \in (V \times \Sigma)^T$ for single-site dynamics using $O(T + |V|)$ memory words, each of $O(\log T + \log |V| + \log |\Sigma|)$ bits, such that each operation among \text{Insert}, \text{Delete}, \text{Change}, \text{Eval} and \text{Succ} can be resolved in time $O(\log T \log M + \log |V|)$, where $M \triangleq \max_{v \in V} |\{ t \in [T] \mid v_t = v \}|$.

**Proof.** The initial state and execution-log are stored by separate data structures.

The initial state $X_0 \in \Sigma^V$ is maintained by a deterministic dynamic dictionary, with $(v, X_0(v))$ for vertices $v \in V$ as the key-value pairs. Such a deterministic data structure answers queries of $X_0(v)$ given any $v \in V$ while $V$ is dynamically changing.

The execution-log $(v_t, X_t(v_t))_{t=1}^T \in (V \times \Sigma)^T$ is stored by $|V|$ balanced search trees $(T_v)_{v \in V}$ (e.g. red-black trees). In each tree $T_v$, each node in $T_v$ stores a distinct transition $(v_t, X_t(v_t))$ with $v_t = v$, such that the in-order tree walk of $T_v$ prints all $(v_t, X_t(v_t))$ with $v_t = v$ in the order they appear in the execution-log $(v_t, X_t(v_t))_{t=1}^T$. Altogether these trees $(T_v)_{v \in V}$ have $T$ nodes in total. Besides, these trees $(T_v)_{v \in V}$ are indexed by another deterministic dynamic dictionary with $(v, p_v)$, $v \in V$, as key-value pairs, where each $p_v$ is pointer to the root of tree $T_v$. This dictionary provides random accesses to the trees $T_v$ for all $v \in V$, while $V$ is dynamically changing.

Given any $t$, we want to answer predecessor (or successor) search for the largest $i \leq t$ (or smallest $i > t$) such that $v_i = v$. This is achieved with assistance from another data structure, an order-statistic tree (or OS-tree) $\hat{T}$ [2, Section 14]. In $\hat{T}$, each node stores the “identity” of an individual transition $(v_t, X_t(v_t))_{t=1}^T$ (which is actually a pointer to the node storing the transition $(v_t, X_t(v_t))$ in the tree $T_v$ with $v_t = v$). In particular, the in-order tree walk of $\hat{T}$ prints all $(v_t, X_t(v_t))_{t=1}^T$ in that order. Such a data structure supports two query functions: (1) \text{Select}: given any $t$, returns the identity of the $t$-th transition $(v_t, X_t(v_t))$; and (2) \text{Rank}: given the identity of any transition $(v_t, X_t(v_t))$ returns its rank $t$ in the sequence $(v_t, X_t(v_t))_{t=1}^T$. Besides, the OS-tree $\hat{T}$ also supports standard insertion (of a new transition $(v, c)$ to a given rank $t$) and deletion (of the transition $(v_t, X_t(v_t))$ at a given rank $t$). As a balanced tree, all these queries and updates for the OS-tree $\hat{T}$ can be resolved in $O(\log T)$ time.

The successor and predecessor searches mentioned above for any $v \in T$ and $t$, can then be resolved by binary searches in the balanced search tree $T_v$ while querying the OS-tree $\hat{T}$ as an oracle for ordering, which takes time at most $O(\log T \log M + \log |V|)$ in total, where $M = \max_{v \in V} |\{ t \in [T] \mid v_t = v \}|$ gives an upper bound on the size of the tree $T_v$, and the log $|V|$ cost is used for accessing the root of $T_v$ via the dynamic dictionary that indexes the trees $(T_v)_{v \in V}$. 

This solves the successor query $\text{Succ}(t, v)$ as well as the evaluation query $\text{Eval}(t, v)$ for single-site dynamics, both within time cost $O(\log T \log M + \log |V|)$, where the latter is actually solved by the predecessor search for the largest $i \leq t$ such that $v_i = v$ and returning the value of $X_i(v_i)$ recorded in the $i$-th transition $(v_i, X_i(v_i))$ or returning the value of $X_0(v)$ if no such $i$ exists.

It is also easy to verify that with the above dynamic data structures, all updates, including: $\text{Insert}(t, v, c)$, $\text{Remove}(t)$ and $\text{Change}(t, c)$, can be implemented with cost at most $O(\log T \log M + \log |V|)$, and the data structures together use $O(T + |V|)$ words in total, where each word consists of $O(\log T + \log |V| + \log |\Sigma|)$ bits. 

### 6.3 The Dynamic Gibbs sampling algorithm

With the data structure for single-site dynamics stated in Theorem 6.5, the couplings constructed in Section 6.1 can be implemented as the algorithm for dynamic Gibbs sampling.

**Lemma 6.6 (dynamic Gibbs sampling algorithm).** Let $\mathcal{I} = (V, E, Q, \Phi)$ be an MRF instance with $n = |V|$ and $\mathcal{I}' = (V', E', Q', \Phi')$ the updated instance. Let $\epsilon > 0$. Denote $T = T(\mathcal{I}, \epsilon)$, $T' = T(\mathcal{I}', \epsilon)$ and $T_{\text{max}} = \max\{T, T'\}$. Assume $T, T' \in \Omega(n \log n)$. There exists an algorithm which does the following:

- **(space cost)** The algorithm maintains an explicit copy of a sample $X \in Q^V$ for the current instance $\mathcal{I}$, and also a data structure using $O(T)$ memory words, each of $O(\log T)$ bits, for representing an initial state $X_0 \in Q^V$ and an execution-log $\text{Exe-Log}(\mathcal{I}, \epsilon) = \langle v_t, X_t(v_t) \rangle_{t=1}^T$ for the Gibbs sampling $(X_t)_{t=0}^T$ on $\mathcal{I}$ generating sample $X = X_T$.

- **(correctness)** Assuming that Condition 6.1 holds for $X_0$ and $\text{Exe-Log}(\mathcal{I}, \epsilon)$ for the Gibbs sampling on $\mathcal{I}$, upon each update that modifies $\mathcal{I}$ to $\mathcal{I}'$, the algorithm updates $X$ to an explicit copy of a sample $Y \in Q^{V'}$ for the new instance $\mathcal{I}'$, and correspondingly updates the $X_0$ and $\text{Exe-Log}(\mathcal{I}', \epsilon)$ represented by the data structure to a $Y_0 \in Q^{V'}$ and $\text{Exe-Log}(\mathcal{I}', \epsilon) = \langle v'_t, Y_t(v'_t) \rangle_{t=1}^{T'}$ for the Gibbs sampling $(Y_t)_{t=0}^{T'}$ on $\mathcal{I}'$ generating the new sample $Y = Y_T'$, where $Y_0$ and $\text{Exe-Log}(\mathcal{I}', \epsilon)$ satisfy Condition 6.1 for the Gibbs sampling on $\mathcal{I}'$, therefore, $d_{TV}(Y, \mu_{\mathcal{I}'}) \leq \epsilon$.

- **(time cost)** Assuming Condition 6.1 for $X_0$ and $\text{Exe-Log}(\mathcal{I}, \epsilon)$ for the Gibbs sampling on $\mathcal{I}$, the expected time complexity for resolving an update is:

$$O\left(\Delta \left(\frac{|T - T'|}{n} + \frac{T_{\text{max}}}{n}\right) \log T_{\text{max}} \cdot \log \frac{T_{\text{max}}}{n}\right),$$

where $\Delta$ is the maximum degree of graph $G = (V, E)$ and $R$ is defined in (15).

**Remark 6.7.** The algorithm stated in Lemma 6.6 works for general single-site dynamics described abstractly by Algorithm 4, with the same upper bounds on the costs, as long as the followings are true: (1) the alphabet $\Sigma$ for the single-site dynamics has $|\Sigma| = O(1)$; and (2) the initial state $Y_0$ for the updated instance $\mathcal{I}'$ can always be obtained by modifying $O(1)$ variables in the initial states $X_0$ for $\mathcal{I}$ with computation cost at most $O(\Delta)$.

Assume that Condition 3.1 holds for both $\mathcal{I}$ and $\mathcal{I}'$ with the same one-step local coupling and the same parameters $\beta = \Omega(1), C, K > 0$. Equation (11) gives a $O(n \log \frac{n}{\epsilon})$ upper bound on the mixing times $T, T'$ and also a $O(\log \frac{n}{\epsilon})$ upper bound on the difference $|T - T'|$. Lemma 6.4 gives a $O(CK\Delta \log \frac{n}{\epsilon})$ upper bound on $\mathbb{E}[R]$. For $\epsilon > \exp(-O(n))$, Theorem 3.2 follows consequently.
Proof of Lemma 6.6. The dynamic Gibbs sampling algorithm is implemented as follows. The algorithm uses the dynamic data structure in Theorem 6.5 to maintain the initial state $X_0$ and execution-log $\text{Exe-Log}(T, e) = \{v_t, X_t(v_t)\}_{t=1}^T$. Besides, the algorithm maintains the explicit copy of the sample $X \in Q^V$ by a deterministic dynamic dictionary, with $(v, X(v))$ for vertices $v \in V$ as the key-value pairs. The lemma is proved as follows.

**Space cost:** Note that $T = \Omega(n \log n)$. The dynamic dictionary for sample $X$ uses $O(n)$ memory words, each of $O(\log T + \log q)$ bits. Hence, the algorithm uses $O(T)$ memory words to maintain the initial state, execution-log and the random sample due to Theorem 6.5.

**Correctness:** The invariants for execution-log (Condition 6.1) are preserved by the coupling simulated by the algorithm. The correctness holds as a consequence.

**Time cost:** Consider the update that modifies $I$ to $I'$. The algorithm updates the initial state $X_0$ to $Y_0$, the execution-log $\text{Exe-Log}(I, e) = \{v_t, X_t(v_t)\}_{t=1}^T$ to $\text{Exe-Log}(I', e) = \{v'_t, Y_t(v'_t)\}_{t=1}^{T'}$, and the sample $X$ to $Y$. We define random variables $M, M'$ and $M_{\text{max}}$ as follows

$$M \triangleq \max_{v \in V} \{|t \in [T] \mid v_t = v\}, \quad M' \triangleq \max_{v \in V'} \{|t \in [T'] \mid v'_t = v\}, \quad M_{\text{max}} \triangleq \max \{M, M'\}.$$

We make the following Claim.

**Claim 6.8.** The updates of $X_0$ to $Y_0$, $\langle v_t, X_t(v_t)\rangle_{t=1}^T$ to $\langle v'_t, Y_t(v'_t)\rangle_{t=1}^{T'}$, and $X$ to $Y$, can be resolved in time

$$T_{\text{up}} \leq C \Delta(|T - T'| + \mathcal{Z}) \log T_{\text{max}} \log M_{\text{max}},$$

(17)

where $C$ is a constant and $\mathcal{Z}$ is a random variable distributed as follows:

- $\mathcal{Z} \sim B(T', 1/|V'|)$, if the update is Vertex-Add;
- $\mathcal{Z} \sim B(T, 1/|V|)$, if the update is Vertex-Delete;
- $\mathcal{Z} = R$ if the update is Edge-Add, Edge-Delete or Update, where $R$ is defined in (15).

Note that $T, T' = \Omega(n \log n)$. Let $C'$ be the constant such that $\min\{T, T'\} \geq C'n \log n$. Define the bad event $\mathcal{E}$ as

$$\mathcal{E} : \quad M_{\text{max}} \geq \frac{50T_{\text{max}}}{\min\{1, C'\}n}.$$

Note that each $v_t \in V$ is uniformly at random and each $v'_t \in V$ is uniformly at random. The random variables $M$ and $M'$ are standard balls and bins models. An easy calculation gives

$$\Pr[\mathcal{E}] \leq \frac{1}{n^{50}} \exp \left( \frac{-50T_{\text{max}}}{n} \right).$$

The expectation of $T_{\text{up}}$ can be written as

$$\mathbb{E}[T_{\text{up}}] = \Pr[\mathcal{E}] \cdot \mathbb{E}[T_{\text{up}} \mid \mathcal{E}] + \Pr[\overline{\mathcal{E}}] \cdot \mathbb{E}[T_{\text{up}} \mid \overline{\mathcal{E}}].$$

(18)

Note that $\mathcal{Z} \leq T_{\text{max}}$ and $M_{\text{max}} \leq T_{\text{max}}$, we have

$$\Pr[\mathcal{E}] \cdot \mathbb{E}[T_{\text{up}} \mid \mathcal{E}] \leq \frac{1}{n^{50}} \exp \left( \frac{-50T_{\text{max}}}{n} \right) C \Delta(|T - T'| + T_{\text{max}}) \log T_{\text{max}} \log T_{\text{max}} = O(1).$$

(19)

By the definition of $\mathcal{E}$, we have

$$\mathbb{E}[T_{\text{up}} \mid \overline{\mathcal{E}}] \leq C \Delta \left(|T - T'| + \mathbb{E}[\mathcal{Z} \mid \overline{\mathcal{E}}]\right) \log T_{\text{max}} \log \left( \frac{50T_{\text{max}}}{\min\{1, C'\}n} \right).$$

(20)
Note that $\Pr[\mathcal{E}] \geq 1 - 1/n^5$. It holds that $\mathbb{E}[Z | \mathcal{E}] \leq \frac{\mathbb{E}[Z]}{\Pr[\mathcal{E}]} \leq 2\mathbb{E}[Z]$. Combining it with (18), (19) and (20), we have

$$\mathbb{E}[T_{\text{up}}] = O \left( \Delta \left( |T - T'| + \mathbb{E}[|Z|] \right) \log T_{\max} \log \frac{T_{\max}}{n} \right).$$

For Vertex-Add and Vertex-Delete updates, it holds that $\mathbb{E}[Z] = O \left( \frac{T_{\max}}{n} \right)$. Hence

$$\mathbb{E}[T_{\text{up}}] = O \left( \Delta \left( |T - T'| + \frac{T_{\max}}{n} + \mathbb{E} \left[ R \right] \right) \log T_{\max} \log \frac{T_{\max}}{n} \right).$$

This proves the time cost of dynamic Gibbs sampling.

\section{Dynamic CFTP}

The coupling from the past (CFTP) is one of the most important frameworks for exact (or perfect) sampling from Gibbs distributions [26]. In this section, we show how to perform dynamic perfect sampling with CFTP.

The preliminaries for CFTP have been reviewed in the Section 5.4. We now formulate the CFTP algorithms as single-site dynamics, where the data structures and algorithms for dynamic sampling with CFTP.

We now formulate the CFTP (see Definition 5.4). At step $-\infty < t \leq 0$, the transition of the single-site dynamics is specified as $X_t = g_{\mathcal{L}_t}(X(\Gamma_{\mathcal{L}_t}(v_t)), r_t)$ where $v_t \in V$ and $r_t \in [0, 1]$ are uniform and independent at random and

$$g_{\mathcal{L}_t}(X(\Gamma_{\mathcal{L}_t}(v_t)), r) \triangleq \begin{cases} (g_{\mathcal{L}_t}(X^{(1)}(\Gamma_{\mathcal{L}_t}(v)), r), g_{\mathcal{L}_t}(X^{(2)}(\Gamma_{\mathcal{L}_t}(v)), r)) & \text{(for monotone systems)} \\ (g_{\mathcal{L}_t}(X^{(2)}(\Gamma_{\mathcal{L}_t}(v)), r), g_{\mathcal{L}_t}(X^{(1)}(\Gamma_{\mathcal{L}_t}(v)), r)) & \text{(for anti-monotone systems)} \\ \{g_{\mathcal{L}_t}(\sigma, r) \mid \sigma \in \otimes_{u \in \Gamma_{\mathcal{L}_t}(v)} \mathcal{X}(u)\} & \text{(for bounding chains)} \end{cases}.$$
Clearly, such transition can be formalized as a marginal transition distribution $P_{\mathcal{X}_t} (\cdot | \mathcal{X}(\Gamma_G(v_t)))$ over $\Sigma(Q)$ as in Algorithm 4, where $P_{\mathcal{X}_t} (\cdot | \mathcal{X}(\Gamma_G(v_t)))$ is precisely the distribution of the value of $g_{\mathcal{X}}(\mathcal{X}(\Gamma_G(v_t)), r)$ for uniform random $r_t \in [0, 1]$.

An execution of this single-site dynamics from the past which generates $\mathcal{X}_{-T}, \mathcal{X}_{-T+1}, \ldots, \mathcal{X}_0$ is described in Algorithms 7. We call this single-site process $(\mathcal{X}_t)_{t=-T}^0$ the CFTP chain.

**Algorithm 7: CFTP($\mathcal{I}, \mathcal{X}_{\text{init}}, T, \mathbf{v}, \mathbf{r}$)**

**Input:** a MRF instance $\mathcal{I} = (V, E, Q, \Phi)$, an initial state $\mathcal{X}_{\text{init}} \in \Sigma(Q)^V$, an integer $T \geq 1$, and random sources $\mathbf{v} = (v_t)_{t=-T+1}^0$ and $\mathbf{r} = (r_t)_{t=-T+1}^0$ where each $v_t \in V$ and $r_t \in [0, 1]$ are uniform and independent at random.

**Output:** a state $\mathcal{X}_0 \in \Sigma(Q)^V$.  

1. $\mathcal{X}_{-T} \leftarrow \mathcal{X}_{\text{init}}$; 
2. for $t = -T + 1$ to 0 do
   1. $\mathcal{X}_t(v) \leftarrow g_{\mathcal{X}_t}(\mathcal{X}_{t-1}(\Gamma_G(v_t)), r_t)$; 
   2. $\mathcal{X}_t(u) \leftarrow \mathcal{X}_{t-1}(u)$ for all $u \in V \setminus \{v\}$;
3. return $\mathcal{X}_0$;

The main CFTP algorithm then makes calls to CFTP($\mathcal{I}, \mathcal{X}_{\text{init}}, T, \mathbf{v}, \mathbf{r}$) with the consistent random sources $\mathbf{v} = (v_t)_{t=-T+1}^0$ and $\mathbf{r} = (r_t)_{t=-T+1}^0$, to search for a $T > 0$ such that when the CFTP chain starts at time $-T$ from the initial state $\mathcal{X}_{\text{init}}$ as defined in (21), the final state $\mathcal{X}_0$ collapses to a configuration $\mathcal{X} \in Q^V$.

Specifically, we say that a state $\mathcal{X} \in \Sigma(Q)^V$ collapses to a configuration $\mathcal{X} \in Q^V$ if it uniquely identifies $\mathcal{X}$, that is,

$$\forall v \in V, \mathcal{X}(v) = \begin{cases} (X(v), X(v)) & \text{(for monotone or anti-monotone systems),} \\ \{X(v)\} & \text{(for bounding chains).} \end{cases}$$  \hspace{1cm} (22)

### 7.2 Dynamic perfect sampling via CFTP

With CFTP expressed as single-site dynamics, we can similarly define its execution-log.

**Execution-log for CFTP:** Let $\mathcal{I} = (V, E, Q, \Phi)$ be the current MRF instance. Suppose that $T_{\text{couple}}(\mathcal{I})$ is an easy-to-compute integer-valued function such that

$$T \geq T_{\text{couple}}(\mathcal{I}) \implies \Pr [\text{CFTP}(\mathcal{I}, \mathcal{X}_{\text{init}}, T, \mathbf{v}, \mathbf{r}) \text{ returns a collapsed } \mathcal{X}_0 | \geq 1 - \frac{1}{n^2},$$  \hspace{1cm} (23)

where $\mathcal{X}_{\text{init}} \in \Sigma(Q)^V$ is given by (21), the meaning of $\mathcal{X}_0$ being collapsed is defined in (22), and the probability is taken over uniform random sources $\mathbf{v} = (v_t)_{t=-\infty}^0$ and $\mathbf{r} = (r_t)_{t=-\infty}^0$.

Let $(\mathcal{X}_t)_{t=-T}^0$ be a CFTP chain on $\mathcal{I}$ where $T = T_{\text{couple}}(\mathcal{I}, \epsilon)$ and $\mathcal{X}_{-T} = \mathcal{X}_{\text{init}}$. As a single-site dynamics, the process $(\mathcal{X}_t)_{t=-T}^0$ can be fully recovered by the initial state $\mathcal{X}_{-T} = \mathcal{X}_{\text{init}}$ and its execution-log which records every transition from $-T + 1$ to 0. We denote the execution-log for this CFTP chain as:

$$\text{CFTP-Log}(\mathcal{I}) \triangleq \langle v_t, \mathcal{X}_t(v_t) \rangle_{t=-T+1}^0.$$  

For CFTP, the initial state $\mathcal{X}_{\text{init}}$ can be assumed implicitly since its construction is fixed as in (21). Therefore, CFTP-Log($\mathcal{I}$) itself is sufficient to recover the whole process $(\mathcal{X}_t)_{t=-T}^0$.

The following invariants are assumed for the random execution-log for the CFTP chain.

**Condition 7.1 (invariants for CFTP-Log).** The followings hold for the random execution-log CFTP-Log($\mathcal{I}$) = $\langle v_t, \mathcal{X}_t(v_t) \rangle_{t=-T+1}^0$ for the CFTP chain $(\mathcal{X}_t)_{t=-T}^0$ on instance $\mathcal{I} = (V, E, Q, \Phi)$:
• \( T = T_{\text{couple}(\mathcal{I})} \) where \( T_{\text{couple}(\mathcal{I})} \) satisfies (23);

• each \( v_t \in V \) is uniform and independent at random;

• the random process \((\mathcal{X}_t)_{t=-\infty}^0\) uniquely specified by the initial state \( \mathcal{X}_{-T} = \mathcal{X}_{\text{init}} \) as defined in (21) and the transitions as given by \( (v_t, \mathcal{X}_t(v_t))_{t=-T+1}^0 \), is identically distributed as the random process generated by \( \text{CFTP}(\mathcal{I}, \mathcal{X}_{\text{init}}, T, v, r) \) with random sources \( v = (v_t)_{t=-T+1}^0 \) and \( r = (r_t)_{t=-T+1}^0 \) where each \( r_t \in [0, 1] \) is chosen uniformly and independently at random.

The dynamic CFTP algorithm. We now present our dynamic CFTP algorithm. As in the case of static CFTP, we assume that we have accesses to oracles for drawing independent samples from \( V \) and \([0, 1]\), with each sample returned with \( O(1) \) time cost, and function \( g(\cdot, \cdot) \) can be evaluated within \( O(\Delta) \) time cost.

**Algorithm 8: Dynamic CFTP**

**Data**: \( \text{CFTP-Log}(\mathcal{I}) = (v_t, \mathcal{X}_t(v_t))_{t=-T+1}^0 \) for current instance \( \mathcal{I} = (V, E, Q, \Phi) \) and a sample \( X \in Q^V \) if \( \mathcal{X}_0 \) does not collapse to a configuration in \( Q^V \).

**Update**: an update that modifies \( \mathcal{I} \) to \( \mathcal{I}' = (V', E', Q, \Phi') \).

1. \( T' \leftarrow T_{\text{couple}(\mathcal{I}')} \);
2. update \( \text{CFTP-Log}(\mathcal{I}) \) to \( \text{CFTP-Log}(\mathcal{I}') = (v'_t, \mathcal{X}_t(v'_t))_{t=-T'+1}^0 \);
3. if \( \mathcal{Y}_0 \) collapses to a \( Y \in Q^V \) then
   4. return;
5. sample random bits \( r' = (r'_t)_{t=-T'+1}^0 \in [0, 1]^{T'} \) conditioning on \( (v'_t, \mathcal{X}_t(v'_t))_{t=-T'+1}^0 \);
6. let \( v' = (v'_t)_{t=-T'+1}^0 \);
7. repeat
   8. construct \( v' = (v'_t)_{t=-2T'+1}^0 \) and \( r' = (r'_t)_{t=-2T'+1}^0 \), with the additional \( v'_t \in V \) and \( r'_t \in [0, 1] \) for \(-2T' + 1 \leq t \leq -T' \) chosen uniformly and independently at random;
   9. \( T' \leftarrow 2T' \);
   10. \( \mathcal{Y}_0 \leftarrow \text{CFTP}(\mathcal{I}', \mathcal{X}_{\text{init}}, T', v', r') \) where \( \mathcal{X}_{\text{init}} \) is defined as in (21) on new instance \( \mathcal{I}' \);
11. until \( \mathcal{Y}_0 \) collapses to a \( Y \in Q^V \);
12. store sample \( Y \) into the date;

The algorithm, as described in Algorithm 8, maintains an execution-log \( \text{CFTP-Log}(\mathcal{I}) \) for the CFTP chain \((\mathcal{X}_t)_{t=-T+1}^0\) on the current instance \( \mathcal{I} \), and if \( \mathcal{X}_0 \) does not collapse to a configuration in \( Q^V \), in addition, a sample \( X \in Q^V \) from \( \mu_\mathcal{I} \). When \( \mathcal{I} \) is updated to a new instance \( \mathcal{I}' = (V', E', Q, \Phi') \) due to one of the update operations described in Section 2, the algorithm updates the current \( \text{CFTP-Log}(\mathcal{I}) \) to a \( \text{CFTP-Log}(\mathcal{I}') = (v'_t, \mathcal{X}_t(v'_t))_{t=-T'+1}^0 \), where \( T' = T_{\text{couple}(\mathcal{I}')} \), and also if \( \mathcal{Y}_0 \) of the new CFTP chain \((\mathcal{X}_t)_{t=-T+1}^0\) does not collapse to a configuration in \( Q^V \), generates a sample \( Y \in Q^V \) which is stored in addition to \( \text{CFTP-Log}(\mathcal{I}') \). This is achieved in the following steps:

1. **Update the execution-log** (Line 1–4): The execution-log \( \text{CFTP-Log}(\mathcal{I}) \) is maintained and updated by the same data structure and algorithm for Gibbs sampling execution-logs in Lemma 6.6. As pointed out in Remark 6.7, the result in Lemma 6.6 holds for general single-site dynamics as long as constructing new initial state \( \mathcal{X}_{\text{init}} \) from \( \mathcal{X}_{\text{init}} \) is easy, which is true for CFTP. The couplings \( D_v^{\sigma, \tau} \) in (16) are now specified by the grand coupling \( g(\cdot, \cdot) \) as following: for any \( v \in V \), \( \sigma \in \Sigma(Q)^{\Gamma_{\mathcal{I}_v}(v)} \) and \( \tau \in \Sigma(Q)^{\Gamma_{\mathcal{I}_v}^c(v)} \):

\[
\forall c, c' \in \Sigma(Q), \quad D_v^{\sigma, \tau}(c, c') = \Pr_{r \in [0, 1]} \left[ g_{\mathcal{I}_v}(\sigma, r) = c \land g_{\mathcal{I}_v}(\tau, r) = c' \right].
\] (24)
After CFTP-Log(\mathcal{I}) updated to CFTP-Log(\mathcal{I}'), \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t=-T'+1}^0, if \mathcal{Y}_0 collapses to a configuration \mathbf{Y} \in Q^{V'} (which can be easily detected by maintaining a counter for non-collapsed vertices), the CFTP process \langle \mathbf{Y}_t \rangle_{t=-T'+1}^0 succeeds and \mathbf{Y} \in Q^{V'} gives a sample for \mu_{T'}.

2. Rarely (with probability at most \frac{1}{n_2}), the above CFTP process \langle \mathbf{Y}_t \rangle_{t=-T'+1}^0 does not successfully produce a sample \mathbf{Y} \in Q^{V'} in which case the algorithm does the followings:

(a) **Recover the random bits** (Line 5–6): Generate the random bits \mathbf{r}' = \langle r'_t \rangle_{t=-T'+1}^0 used in the CFTP process \langle \mathbf{Y}_t \rangle_{t=-T'+1}^0 with correct distribution. This can be done by reverse sampling: for each \mathbf{t} = -T'+1 to 0, repetitively sample uniform \mathbf{r} \in \{0, 1\} until \mathcal{Y}_t(v'_t) = \mathbf{g}_T(\mathcal{Y}_t-1(\Gamma_G(v_t)), \mathbf{r}) and let \mathbf{r}'_t be the first such \mathbf{r}. The random bits sampled in this way are identically distributed as the random bits \mathbf{r}' = \langle r'_t \rangle_{t=-T'+1}^0 used in CFTP(\mathcal{I}', \mathcal{Y}_{init}, T', \mathbf{v}', \mathbf{r}') conditioning on the execution-log \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t=-T'+1}^0.

(b) **Static CFTP** (Line 7–12): The algorithm then runs the static CFTP algorithm with the last \mathcal{T'} random choices set as \mathbf{v}' = \langle v'_t \rangle_{t=-T'+1}^0 and \mathbf{r}' = \langle r'_t \rangle_{t=-T'+1}^0, where \mathbf{v}' \in V^{T'} is generated by the CFTP-Log(\mathcal{I}') = \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t=-T'+1}^0 and \mathbf{r}' \in \{0, 1\}^{T'} is sampled as above. A sample \mathbf{Y} \in Q^{V'} is generated at last and stored in addition to the execution-log CFTP-Log(\mathcal{I}').

Assuming that Condition 7.1 holds for CFTP-Log(\mathcal{I}) = \langle v_t, \mathcal{X}_t(v_t) \rangle_{t=-T+1}^0. Condition 7.1 still holds invariantly for CFTP-Log(\mathcal{I}') = \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t=-T'+1}^0. This is simply guaranteed by the correctness of the couplings for single-site dynamics defined in Section 6.1, used by the algorithm in Lemma 6.6. It is also easy to verify that Algorithm 8 recovers the random bits used by the CFTP process \langle \mathbf{Y}_t \rangle_{t=-T'+1}^0 with correct distribution and hence perfectly simulates the CFTP no matter whether it succeeds in the first pass.

Altogether, Condition 7.1 holds as an invariant, and also \mathbf{Y} \sim \mu_{T'} due to the correctness of CFTP. This gives the correctness of Algorithm 8. Moreover, we have the following.

**Lemma 7.2 (dynamic CFTP algorithm).** Let \mathcal{I} = (V, E, Q, \Phi) be an MRF instance with \mathbf{n} = |V| and \mathcal{I}' = (V', E', Q, \Phi') the updated instance. Assume there is a CFTP algorithm for both \mathcal{I} and \mathcal{I}' that can be expressed as the single-site dynamics in Algorithm 8. Denote \mathcal{T} = T_{couple}(\mathcal{I}), \mathcal{T}' = T_{couple}(\mathcal{I}') and \mathcal{T}_{max} = \max\{\mathcal{T}, \mathcal{T}'\}, where T_{couple}(\cdot) is defined in (23). Assume \mathcal{T}, \mathcal{T}' \in \Omega(n \log n). There exists an algorithm which does the followings:

- **(space cost)** The algorithm maintains an explicit copy of a sample \mathbf{X} \in Q^V for the current instance \mathcal{I}, and also a data structure using \mathcal{O}(\mathcal{T}) memory words, each of \mathcal{O}(\log \mathcal{T}) bits, for representing an execution-log CFTP-Log(\mathcal{I}) = \langle v_t, \mathcal{X}_t(v_t) \rangle_{t=-T+1}^0 for the CFTP \langle \mathbf{X}_t \rangle_{t=-T}^0 on \mathcal{I}. The sample \mathbf{X} \sim \mu_{\mathcal{I}} is a perfect sample for \mathcal{I}.

- **(correctness)** Assuming that Condition 7.1 holds for CFTP-Log(\mathcal{I}) for the CFTP on \mathcal{I}, upon each update that modifies \mathcal{I} to \mathcal{I}', the algorithm updates the CFTP-Log(\mathcal{I}) represented by the data structure to CFTP-Log(\mathcal{I}') = \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t=-T'+1}^0 for the CFTP on \mathcal{I}', where CFTP-Log(\mathcal{I}') satisfies Condition 7.1 on \mathcal{I}', and the algorithm also has an explicit copy of a sample \mathbf{Y} \in Q^{V'} such that \mathbf{Y} \sim \mu_{T'}.

- **(time cost)** Assuming Condition 7.1 for CFTP-Log(\mathcal{I}) for the CFTP on \mathcal{I}, the expected time complexity for resolving an update is:

\[ O\left( \Delta \left( |\mathcal{T} - \mathcal{T}'| + \frac{T_{max}}{\mathbf{n}} + \mathcal{E} [\mathcal{R}] \right) \log T_{max} \cdot \log \frac{T_{max}}{\mathbf{n}} \right), \]
where $\Delta$ is the maximum degree of graph $G = (V, E)$ and $R$ is defined in (15) with the coupling constructed in (24).

To prove the main results for dynamic CFTP (Theorem 3.6), we need to give the functions $T_{\text{couple}}(I), T_{\text{couple}}(I')$ defined in (23), and bound the expectation of $R$ defined by the coupling constructed in (24). These are deferred to Section 9.

**Proof of Lemma 7.2.** As argued above, assuming that Condition 7.1 holds for the original log CFTP-$\text{Log}(I) = (v_t, X_t(v_t))_{t=-T+1}^0$, by the correctness of the couplings for single-site dynamics defined in Section 6.1, used by the algorithm in Lemma 6.6, Condition 7.1 still holds invariantly for CFTP-$\text{Log}(I') = (v'_t, Y_t(v'_t))_{t=-T'+1}^0$. And since Algorithm 8 recovers the random bits used by the CFTP process $(Y_t)_t^{0,T'+1}$ with correct distribution, it perfectly simulates the CFTP no matter whether it succeeds in the first pass.

Altogether, Condition 7.1 holds as an invariant, and also $Y \sim \mu_{I'}$ due to the correctness of Algorithm 8.

We then bound the space and time costs. We remark that $X_t(v_t)$ in Algorithms 7 can be calculated in $O(\Delta)$ time in our setting. It is easy to see that it can be calculated in $O(\Delta)$ time for monotone and anti-monotone systems. For bounding chains, recall that it is assumed that $X_t(v_t)$ can be calculated in $O(\Delta)$ time if $q = O(1)$.

First, we bound the cost for updating the execution-log. As pointed out by Remark 6.7, Lemma 6.6 also holds for the single-site dynamics Algorithm 7, where $|\Sigma| \leq 2^q$. Thus, we have

- **(space cost)** Algorithm 8 maintains a data structure using $O(T)$ memory words, each of $O(\log T + q)$ bits to update the execution-log.

- **(time cost)** the expected time complexity for updating CFTP-$\text{Log}(I)$ to CFTP-$\text{Log}(I')$ is

\[
O \left( \Delta \left( |T - T'| + \frac{T_{\max}}{n} + E[R] \right) \log T_{\max} \cdot \log \frac{T_{\max}}{n} \right).
\]

Second, we bound the cost for recovering the random bits. With probability at most $1/n^2$, we need to recover the random bits and this takes $O(T')$ calls to the oracles for drawing independent samples from $V$ and $[0, 1]$. This incurs no extra space cost with data structures. The expected time cost for recovering the random bits is then $O \left( \frac{T_{\max}}{n^2} \right)$.

At last, we bound the cost for static CFTP. Let random variable $T_{\min}$ be

\[
T_{\min} \triangleq \min \left\{ k : \text{CFTP}(I, X, \text{init}, k, v, r) \text{ returns a collapsed } X_0 \right\}.
\]

If $T_{\min} > T'$, we need to run the static CFTP. By the definition of $T'$, we have

\[
\Pr \left[ T_{\min} > T' \right] \leq \frac{1}{n^2}.
\]

It is well known that when bounding the stopping time, due to its memoryless property, $kT'$ rounds in Algorithm 7 can be viewed as $k$ independent running of the algorithm, each with $T'$ rounds, which means that for any integer $k > 0$ we have

\[
\Pr \left[ T_{\min} > kT' \right] \leq \frac{1}{n^{2k}}.
\]

Thus, we have $E[T_{\min}] \leq 2T'$. Similarly, $T' + T_{\min}$ rounds in Algorithm 7 can be viewed as $T'$ rounds and another $T_{\min}$ rounds independently, we have

\[
E \left[ T_{\min} \mid T_{\min} > T' \right] \leq 2T' + T' = 3T'.
\]
It is also well known for the CFTP that the total number of transitions (where each transition corresponds to one iteration in Algorithm 7) of the static CFTP algorithm (as Line 7–12 in Algorithm 8) is no more than $4T_{\text{min}}$ [26]. Therefore, conditioning on $T_{\text{min}} > T'$, the expected total number of transitions for the static CFTP algorithm is no more than $12T'$. Because $\Pr[T_{\text{min}} > T'] \leq \frac{1}{n^2}$, we have the expected total number of transitions for the static CFTP algorithm bounded as $O\left(\frac{T_{\text{max}}}{n^2}\right)$. And since each transition costs $O(\Delta)$ in time, the expected time cost for static CFTP (Line 7–12 in Algorithm 8) is at most $O\left(\frac{\Delta T_{\text{max}}}{n^2}\right)$. For this part, there is also no need to maintain extra data structure.

The lemma holds by combining above three costs.

\[ \square \]

8 Proofs for Dynamic Gibbs Sampling

8.1 Analysis of the coupling

In this section, we prove Lemma 6.4.

By the definition of $R$ in (15) and the linearity of the expectation, we have

\[ \mathbb{E}[R] = \sum_{t=1}^{T'} \mathbb{E}[\gamma_t] = \sum_{t=1}^{T'} \mathbb{E}[\mathbb{E}[\gamma_t \mid D_{t-1}]]. \]

Recall $\gamma_t = 1 \ [v_t \in S \cup \Gamma^+_G(D_{t-1})]$ and $v_t \in V$ is uniformly at random given $D_{t-1}$. Note that $|\Gamma^+_G(D_{t-1})| \leq (\Delta + 1)|D_{t-1}|$ and $|S| \leq 2$. We have

\[ \mathbb{E}[R] \leq \sum_{t=1}^{T'} \mathbb{E}\left(\frac{(\Delta + 1)|D_{t-1}| + 2}{n}\right) = \frac{(\Delta + 1)}{n} \sum_{t=1}^{T'} \mathbb{E}[|D_{t-1}|] + \frac{2T'}{n}. \] (25)

If $\mathcal{I}$ and $\mathcal{I}'$ satisfy Condition 3.1 with the same one-step local coupling and the same parameters $\beta, C, K$, we claim that

\[ \forall 0 \leq t \leq T': \ \mathbb{E}[|D_t|] = O\left(\frac{CK}{\beta}\right). \] (26)

Combining (25) and (26), we have

\[ \mathbb{E}[R] = O\left(\frac{\Delta T'CK}{n\beta}\right). \]

This proves the lemma.

We now prove (26). Recall $\Omega_\mathcal{I} \subseteq Q^V$ is the feasible set for $\mathcal{I}$ and $\Omega_{\mathcal{I}'} \subseteq Q^{V'}$ is the feasible set for $\mathcal{I}'$. Consider the updates Edge-Add, Edge-Delete, Update. Note that Update$(a, \phi'_a)$ requires that there is at most one entry between the original and the updated constraints $\phi_a$ and $\phi'_a$, whose value changing between zero and positive. It must hold that either $\Omega_\mathcal{I} \subseteq \Omega_{\mathcal{I}'}$ or $\Omega_{\mathcal{I}'} \subseteq \Omega_\mathcal{I}$. Without loss of generality, we assume

\[ \Omega_{\mathcal{I}'} \subseteq \Omega_\mathcal{I}. \]

The case $\Omega_\mathcal{I} \subseteq \Omega_{\mathcal{I}'}$ follows by the symmetry.

Let $(X'_t, Y'_t)_{t \geq 0}$ be the one-step local coupling for Gibbs sampling (Definition 5.2) defined by a family of couplings $D_{\mathcal{I},t}^\tau(\cdot, \cdot)$ of marginal distributions $\mu_{\mathcal{I},t}(\cdot \mid \sigma)$ and $\mu_{\mathcal{I},t}(\cdot \mid \tau)$. Assume $\mathcal{I}$ satisfies Condition 3.1 with the coupling $(X'_t, Y'_t)_{t \geq 0}$. By the step-wise decay property, we have

\[ \forall \sigma, \tau \in \Omega_\mathcal{I}: \ \mathbb{E}\left[\rho_\mathcal{I}(X'_t, Y'_t) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau\right] \leq \left(1 - \frac{\beta}{n}\right) \cdot \rho_\mathcal{I}(\sigma, \tau); \] (27)
Let \((X_t, Y_t)_{t \geq 0}\) be the one-step local coupling for dynamic instance (Definition 6.2), defined by a family of couplings \(D_{v}^{\sigma, \tau}(\cdot, \cdot)\) of marginal distributions \(\mu_{v, t}^{\cdot} | \sigma\) and \(\mu_{v, t}^{\cdot} | \tau\). By the definition of \(D_{v}^{\sigma, \tau}(\cdot, \cdot)\) in (16), two couplings \(D_{v}^{\sigma, \tau}(\cdot, \cdot)\) and \(D_{v}^{\sigma, \tau}(\cdot, \cdot)\) differ only if \(v \in S\). Note that \(\rho_{\tau}\) is \(K\)-Lipschitz, \(|S| \leq 2\). We claim the following result

\[
\forall \sigma \in \Omega_{T}, \tau \in \Omega_{T}; \subseteq \Omega_{T} : \quad \mathbb{E}[\rho_{\tau}(X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] \leq \left(1 - \frac{\beta}{n}\right) \cdot \rho_{\tau}(\sigma, \tau) + \frac{4K}{n}.
\]

(28)

Assume (28) holds. Taking expectation over \(X_{t-1}\) and \(Y_{t-1}\), we have

\[
\mathbb{E}[\rho_{\tau}(X_t, Y_t)] \leq \left(1 - \frac{\beta}{n}\right) \mathbb{E}[\rho_{\tau}(X_{t-1}, Y_{t-1})] + \frac{4K}{n}.
\]

(29)

By Condition 3.1, \(\rho_{\tau}(\sigma, \sigma) = 0\) for all \(\sigma \in \Omega_{T}\). Note that \(X_0 \in \Omega_{T}, Y_0 \in \Omega_{T}; \subseteq \Omega_{T}\) and the Hamming distance \(H(X_0, Y_0) \leq 2\). Since \(\rho_{\tau}\) is \(K\)-Lipschitz, we have

\[
\rho_{\tau}(X_0, Y_0) \leq \rho_{\tau}(X_0, X_0) + 2K \leq 2K.
\]

(30)

Combining (29) and (30) implies

\[
\forall 0 \leq t \leq T' : \quad \mathbb{E}[\rho_{\tau}(X_t, Y_t)] \leq 4K \left(2 + \frac{1}{\beta}\right).
\]

By the low-distortion to Hamming property, we have

\[
\forall 0 \leq t \leq T' : \quad \mathbb{E}[\|D_t\|] \leq C \mathbb{E}[\rho_{\tau}(X_t, Y_t)] = O\left(\frac{CK}{\beta}\right).
\]

This proves the claim in (26).

We finish the proof by proving the claim in (28). According to the coupling, we can rewrite the expectation in (27) as follows:

\[
\mathbb{E}[\rho_{\tau}(X_t', Y_t') \mid X_{t-1}' = \sigma \land Y_{t-1}' = \tau] = \frac{1}{n} \sum_{v \in V} \mathbb{E}\left[\rho_{\tau}\left(\sigma^{v \rightarrow C_{v}'}, \tau^{v \rightarrow C_{v}'}, \right)\right] ,
\]

(31)

where \((C_{v}', C_{v}') \sim D_{v}^{\sigma, \tau}\) and the configuration \(\sigma^{v \rightarrow C_{v}'} \in Q^V\) is defined as

\[
\sigma^{v \rightarrow C_{v}'}(u) = \begin{cases} C_{v}' & \text{if } u = v \\ \sigma(u) & \text{if } u \neq v \end{cases}
\]

and the configuration \(\tau^{v \rightarrow C_{v}'} \in Q^V\) is defined in similar way.

Similarly, we can rewrite the expectation in (28) as follows:

\[
\mathbb{E}[\rho_{\tau}(X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] = \frac{1}{n} \sum_{v \in V} \mathbb{E}\left[\rho_{\tau}\left(\sigma^{v \rightarrow C_{v}'}, \tau^{v \rightarrow C_{v}'}, \right)\right] ,
\]

(32)

where \((C_{v}', C_{v}') \sim D_{v}^{\sigma, \tau}\).

The following two properties hold for (31) and (32).

- If \(v \not\in S\), by the definition of \(D_{v}^{\sigma, \tau}(\cdot, \cdot)\) in (16), it holds that \(D_{v}^{\sigma, \tau} = D_{v}^{\sigma, \tau}\). Hence

\[
\forall v \not\in S : \quad \mathbb{E}\left[\rho_{\tau}\left(\sigma^{v \rightarrow C_{v}'}, \tau^{v \rightarrow C_{v}'}\right)\right] = \mathbb{E}\left[\rho_{\tau}\left(\sigma^{v \rightarrow C_{v}'}, \tau^{v \rightarrow C_{v}'}\right)\right].
\]
If \( v \in \mathcal{S} \), then it holds that \( H(\sigma^{v \leftarrow C_{v}^{X}}, \sigma^{v \leftarrow C_{v}^{X}}) \leq 1 \) and \( H(\tau^{v \leftarrow C_{v}^{Y}}, \tau^{v \leftarrow C_{v}^{Y}}) \leq 1 \). Since \( \Omega_{I} \subseteq \Omega_{I} \), then it holds that \( \sigma^{v \leftarrow C_{v}^{X}}, \sigma^{v \leftarrow C_{v}^{X}}, \tau^{v \leftarrow C_{v}^{Y}}, \tau^{v \leftarrow C_{v}^{Y}} \in \Omega_{I} \). Note that the function \( \rho_{I} \) is \( K \)-Lipschitz. Hence

\[
\forall v \in \mathcal{S}: \quad \mathbb{E} \left[ \rho_{I} \left( \sigma^{v \leftarrow C_{v}^{X}}, \tau^{v \leftarrow C_{v}^{Y}} \right) \right] \leq \mathbb{E} \left[ \rho_{I} \left( \sigma^{v \leftarrow C_{v}^{X}}, \tau^{v \leftarrow C_{v}^{Y}} \right) \right] + 2K.
\]

Combining above two properties with (31) and (32), we have for any \( \sigma \in \Omega_{I}, \tau \in \Omega_{I} \subseteq \Omega_{I} \),

\[
\mathbb{E} \left[ \rho_{I}(X_{t}, Y_{t}) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau \right] = \frac{1}{n} \sum_{v \in V} \mathbb{E} \left[ \rho_{I} \left( \sigma^{v \leftarrow C_{v}^{X}}, \tau^{v \leftarrow C_{v}^{Y}} \right) \right] \\
\leq \frac{1}{n} \sum_{v \in \mathcal{S}} \mathbb{E} \left[ \rho_{I} \left( \sigma^{v \leftarrow C_{v}^{X}}, \tau^{v \leftarrow C_{v}^{Y}} \right) \right] + \frac{1}{n} \sum_{v \in \mathcal{S}} \left( \mathbb{E} \left[ \rho_{I} \left( \sigma^{v \leftarrow C_{v}^{X}}, \tau^{v \leftarrow C_{v}^{Y}} \right) \right] + 2K \right) \\
(*) \leq \mathbb{E} \left[ \rho_{I}(X'_{t}, Y'_{t}) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau \right] + \frac{4K}{n} \\
\leq \left( 1 - \frac{\beta}{n} \right) \cdot \rho_{I}(\sigma, \tau) + \frac{4K}{n},
\]

where \((*)\) holds due to \(|\mathcal{S}| \leq 2\). This proves the claim in (28).

### 8.2 Implementation of the algorithm

In this section, we prove Claim 6.8. by giving the implementation of the dynamic Gibbs sampling algorithm.

Note that \( T_{\text{max}} = \Omega(n \log n) \). Hence, the time cost for each operation on the dynamic data structure in Theorem 6.5 is \( O(\log T_{\text{max}} \log M_{\text{max}}) \).

Consider the subroutine \( \text{LengthFix}(\mathcal{I}, X_{0}, \langle v_{t}, X_{t}(v_{t}) \rangle_{t=1}^{T}, T') \) that changes the length of the execution-log from \( T \) to \( T' \). If \( T' < T \), the algorithm performs \text{Delete} to delete last \( T - T' \) transitions. If \( T' > T \), the algorithm uses the operations \( \text{Eval} \) and \( \text{Insert} \) to simulate the Glauber dynamics on \( \mathcal{I} \) for \( T' - T \) more steps. By Theorem 6.5, it is easy to see the time cost is

\[
O(\Delta|T - T'| \log T_{\text{max}} \log M_{\text{max}}).
\]

Consider the update \( \text{Vertex-Add}(v^{*}, \phi_{v}^{*}) \). The algorithm does as follows:

- Sample \( N_{1} \) from the binomial distribution \( B(T', 1/|V'|) \).
- Call \( \text{LengthFix} \) on instance \( \mathcal{I} \) to change the length of the execution log to \( T' - N_{1} \).
- Update the initial state, then insert new transitions for \( N_{1} \) steps. In \( k \)-th step, sample \( t \in [T' - N_{1} + k] \) uniformly at random, sample \( c \in Q \) with probability proportional to \( \phi_{v}^{*}(c) \) and perform \( \text{Insert}(t, v^{*}, c) \).
- Update the sample \( X \) by inserting the value of \( v^{*} \) and setting \( X(v^{*}) \leftarrow \text{Eval}(T', v^{*}) \).

Note that the time cost of updating the random sample \( X \) is \( O(\log n) \). By Theorem 6.5, the update \( \text{Vertex-Add}(v^{*}, \phi_{v}^{*}) \) is resolved in time

\[
T_{\text{add}} = O \left( \Delta |T - (T' - N_{1})| \log T_{\text{max}} \log M_{\text{max}} \right) + O \left( N_{1} |T_{\text{max}} \log M_{\text{max}} \right) + O(\log n) \\
= O \left( \Delta (|T - T'| + N_{1}) \log T_{\text{max}} \log M_{\text{max}} \right),
\]

where \( N_{1} \sim B(T', 1/|V'|) \). This proves (17) for \( \text{Vertex-Add} \) update.

Consider the update \( \text{Vertex-Delete}(v^{*}) \). The algorithm does as follows:
• Perform $i \leftarrow \text{Succ}(0, v^*)$ and \text{Delete}(i)$ until $\text{Succ}(0, v^*)$ returns $\bot$.
• Update the initial state.
• Call \text{LengthFix} on instance $I'$ to change the length of the execution log to $T'$.
• Update the random sample $X$ by deleting the value of $v^*$.

The first step deletes all transitions that pick the vertex $v^*$. Let $N_2 = \{|t \in [T] \mid v_t = v^*\}$. Hence $N_2 \sim B(T, 1/|V|)$. Note that the time cost of updating the random sample $X$ is $O(\log n)$. By Theorem 6.5, the \text{Vertex-Delete} update is resolved in time

$$T_{\text{del}} = O \left( N_2 \log T_{\max} \log M_{\max} + O \left( \Delta (|T - N_2| - T') \log T_{\max} \log M_{\max} \right) + O(\log n) \right)$$

$$= O \left( \Delta (|T - T'| + N_2) \log T_{\max} \log M_{\max} \right),$$

where $N_2 \sim B(T, 1/|V|)$. This proves (17) for \text{Vertex-Delete} update.

Consider the updates \text{Edge-Add}, \text{Edge-Delete} and \text{Update}. Let $S \subset V$ denote the set of vertices affected by the update. The dynamic Gibbs sampling for these updates is the Algorithm 5.

It is easy to see the time cost of Line 1 and Line 2 is

$$T_{\text{fix}} = O(\Delta |T - T'| \log T_{\max} \log M_{\max}) + O(1).$$

Consider the while-loop in Algorithm 5. Let $(t_0, D)$ be the variables in Algorithm 5. The implementation of algorithm maintains the following invariant with respect to $(t_0, D)$: the execution log in the dynamic data structure equals

$$\left\langle Y_0, (v_t, Y_t(v_t))_{t=1}^{t_0}, (v_t, X_t(v_t))_{t=t_0+1}^{T'} \right\rangle.$$  

(34)

In addition, we need some temporary data structures that solve the following problems efficiently:

• answering queries in Line 3 and Line 4;
• retrieving the state $X_{t-1}(\Gamma^+_G(v_t))$ in Line 6.

We claim that each execution of the while-loop in Algorithm 5 can be simulated with amortized time complexity $O(\Delta \log T_{\max} \log M_{\max})$. Recall $R$ defined in (15) is the number of times that the while-loop is executed in Algorithm 5. So the time complexity for simulating while-loop is

$$T_{\text{While}} = O(R \Delta \log T_{\max} \log M_{\max}).$$

(35)

Finally, for all vertices $v \in D$, the algorithm updates the sample $X$ by setting $X(v) \leftarrow \text{Eval}(v, T')$. Initially, $|D| = 0$, and the size of $|D|$ increases at most 1 for each execution of the while-loop. Thus $|D| \leq R$. The time complexity for updating the random sample $X$ is

$$T_{\text{Sample}} = O(R \log T_{\max} \log M_{\max}).$$

(36)

Combining (33), (35) and (36), we have

$$T_{\text{chg}} = T_{\text{fix}} + T_{\text{While}} + T_{\text{Sample}}$$

$$= O \left( \Delta (|T - T'| + R) \log T_{\max} \log M_{\max} \right).$$

This proves (17) for \text{Edge-Add}, \text{Edge-Delete} and \text{Update} updates.

We give the detailed implementation to prove (35). We introduce three temporary data structures $A, B, C$. The data structure $A$ maintains a set $S_A \subseteq V$ and a map $f_A : S_A \rightarrow Q$ such that the following invariant holds

$$S_A = D \quad \text{and} \quad \forall v \in S_A, \quad f_A(v) = X_{t_0}(v).$$

(37)

The data structure $A$ supports the following operations:
• \texttt{A.Update}(v, c): which inserts $v$ to $S_A$ and set $f_A(v) = c$ if $v \notin S_A$; or updates $f_A(v) \leftarrow c$ if $v \in S_A$;
• \texttt{A.Delete}(v): which deletes $v$ from $S_A$;
• \texttt{A.Member}(v): which returns whether $v \in S_A$;
• \texttt{A.Eval}(v): which returns $f_A(v)$.

Since $|D| \leq n$, it is easy to see $A$ can be implemented by a balanced binary search tree such that the time complexity of each operation is at most $O(\log n)$.

The data structure $B$ maintains a set $S_B \subseteq [T'] \times V$, where $S_B \subseteq \{(t, v_t) \mid t \in [T']\}$, such that the following invariant holds

$$S_B \supseteq \{(t, v_t) \mid v_t \in \Gamma^+_G(D) \land t = \text{Succ}(t_0, v_t)\}. \quad (38)$$

The data structure $B$ supports the following operations:

• \texttt{B.Insert}(t, v_t): which inserts $(t, v_t)$ to set $S_B$.

• \texttt{B.Delete-Min}(): which deletes $(t, v_t)$ with smallest $t$;

• \texttt{B.Find-Min}(): which returns $(t, v_t)$ with smallest $t$ if $S_B \neq \emptyset$ or returns $\perp$ if $S_B = \emptyset$.

Since the size of $S_B$ is at most $T_{\text{max}}$, it is easy to see $B$ can be implemented by a heap such that the time complexity of each operation is at most $O(\log T_{\text{max}})$.

The data structure $C$ maintains a map $f_C : V \rightarrow \mathbb{Z}_{\geq 0}$ such that the following invariant holds

$$\forall v \in V, \quad f_C(v) = |D \cap \Gamma^+_C(v)|. \quad (39)$$

The data structure $C$ supports the following operations:

• \texttt{C.Update}(v, c): which updates $f_C(v) \leftarrow f_C(v) + c$;

• \texttt{C.Eval}(v): which returns $f_C(v)$.

It is easy to see $C$ can be implemented by a balanced binary search tree. Besides, we can implement $C$ such that $C$ only needs to store $f_C(v)$ for $f_C(v) \neq 0$. The time complexity of each operation is at most $O(\log n)$.

The implementation starting from Line 3 is given as follows.

• Initially, since $t_0 = 0$, the invariant (34) holds; since $D = \emptyset$, $A, B, C$ are initialized as empty trees, hence the invariants (37), (38) and (39) hold. Remark that the binary search tree $C$ only needs to store $f_C(v)$ for $f_C(v) \neq 0$.

• For Line 3 and Line 4, the algorithm solves the queries as follows:

1. Perform $(t, v_t) \leftarrow \texttt{B.Find-Min}()$; check the condition $t > t_0 \land v_t \in \Gamma^+_G(D)$ using \texttt{C.Eval}(v_t) (due to invariant (39), $v_t \in \Gamma^+_G(D)$ iff $f_C(v_t) > 0$); if the condition is not satisfied, then perform \texttt{B.Delete-Min}(). Repeat the above procedure until $t > t_0 \land v_t \in \Gamma^+_G(D)$ or \texttt{B.Find-Min}() = $\perp$.

2. Perform \texttt{Succ}(t_0, u) for all $u \in S$.

3. Combining above two steps to find the smallest $t > t_0$ such that $v_t \in S \cup \Gamma^+G(D)$ or declare such $t$ does not exist.
The time complexity of step 1 will be analyzed later; the time complexity of step 2 is \( O(\log T_{\text{max}} \log M_{\text{max}}) \) because \(|S| \leq 2\) and the time complexity of step 3 is \( O(1) \).

- For Line 5, the algorithm does not need to do any operation.
- For Line 6, the algorithm retrieves \( X_t(v_l) \) and \( Y_{t-1}(\Gamma^+_G(v_l)) \) by performing \( \text{Eval}(t, v_l) \) and \( \text{Eval}(t-1, u) \) for all \( u \in \Gamma^+_G(v_l) \). Note that the \( X_{t-1}(\Gamma^+_G(v_l)) \) satisfies

\[
X_{t-1}(u) = \begin{cases} 
Y_{t-1}(u) & \text{if } u \in \Gamma^+_G(v_l) \setminus D \\
X_{t-0}(u) & \text{if } u \in \Gamma^+_G(v_l) \cap D.
\end{cases}
\]

To retrieve \( X_{t-1}(u) \) for \( u \in \Gamma^+_G(v_l) \), the algorithm checks whether \( u \in D \) by \( \mathcal{A}.\text{Member}(u) \); if \( u \notin D \), retrieves \( X_{t-1}(u) \) by \( \text{Eval}(t-1, u) \); if \( u \in D \), retrieves \( X_{t-1}(u) \) by \( \mathcal{A}.\text{Eval}(u) \). After sampled \( Y_t(v_l) \), the algorithm performs \( \text{Change}(t, Y_t(v_l)) \).

The invariant (34) holds after the update. Note that \( T_{\text{max}} = \Omega(n \log n) \). The time complexity of Line 6 is \( O(\Delta \log T_{\text{max}} \log M_{\text{max}}) \).

- For Line 7, the algorithm updates the data structures \( \mathcal{A}, \mathcal{B}, \mathcal{C} \). Here are four cases depending on whether \( X_t(v_l) = Y_t(v_l) \) and whether \( v_l \in D \) before \( D \) is updated in Line 7:

1. Case \( X_t(v_l) = Y_t(v_l) \land v_l \in D \): perform \( \mathcal{A}.\text{Delete}(v_l) \); perform \( \mathcal{C}.\text{Update}(u, -1) \) for \( u \in \Gamma^+_G(v_l) \); if \( f_G(v_l) > 0 \) after the update of \( \mathcal{C} \), perform \( \mathcal{B}.\text{Insert}(\text{Succ}(t, v_l)) \) if \( \text{Succ}(t, v_l) \neq \perp \).
2. Case \( X_t(v_l) = Y_t(v_l) \land v_l \notin D \): perform \( \mathcal{B}.\text{Insert}(\text{Succ}(t, v_l), v_l) \) if \( \text{Succ}(t, v_l) \neq \perp \).
3. Case \( X_t(v_l) \neq Y_t(v_l) \land v_l \in D \): perform \( \mathcal{A}.\text{Update}(v_l, X_t(v_l)) \); perform operation \( \mathcal{B}.\text{Insert}(\text{Succ}(t, u), u) \) for \( u \in \Gamma^+_G(v_l) \) satisfying \( \text{Succ}(t, u) \neq \perp \).
4. Case \( X_t(v_l) \neq Y_t(v_l) \land v_l \notin D \): perform \( \mathcal{A}.\text{Update}(v_l, X_t(v_l)) \); perform operation \( \mathcal{B}.\text{Insert}(\text{Succ}(t, u), u) \) for \( u \in \Gamma^+_G(v_l) \) satisfying \( \text{Succ}(t, u) \neq \perp \); perform \( \mathcal{C}.\text{Update}(u, 1) \) for \( u \in \Gamma^+_G(v_l) \).

It can be verified that the invariants (37), (38) and (39) hold after the update of \( \mathcal{A}, \mathcal{B}, \mathcal{C} \). Note that \( T_{\text{max}} = \Omega(n \log n) \). The time complexity of Line 7 is \( O(\Delta \log T_{\text{max}} \log M_{\text{max}}) \).

Recall \( R \) is the number of times that the while-loop is executed in Algorithm 5. Note that for each execution of the while-loop, we at most insert \( O(\Delta) \) elements to data structure \( \mathcal{B} \) when simulating Line 7. Consider the step 1 for simulating Line 3 and Line 4. The total time complexity contributed by this step is at most \( O(\Delta R \log T_{\text{max}}) \). Combining it with the time complexities of other steps proves (35).

### 8.3 Applications

In this section, we apply Condition 3.1 on specific models and give the corresponding dynamic approximate sampling results.

**Corollary 8.1.** There exist dynamic approximate sampling algorithms as stated in Theorem 3.2 with the same space cost \( O\left( n \log \frac{n}{\epsilon} \right) \), and expected time cost \( O\left( \Delta^2 \cdot \log n \cdot \log \left( \frac{n}{\epsilon} \right) \cdot \log \log \left( \frac{n}{\epsilon} \right) \right) \) for each update, for the following models on graphs with \( n \) vertices, and \( \epsilon > \exp(-O(n)) \):

- Ising model with temperature \( \beta \) and arbitrary local fields where \( \exp(-2|\beta|) \geq 1 - \frac{2-\delta}{\Delta+1} \);
- proper \( q \)-coloring with \( q \geq (2+\delta)\Delta \);
• hardcore model with fugacity \( \lambda \leq \frac{2-\delta}{3\Delta} \), but with an alternative time cost for each update

\[
O\left(\Delta^3 \cdot \log n \cdot \log \left(\frac{n}{\epsilon}\right) \cdot \log \log \left(\frac{n}{\epsilon}\right)\right);
\]

where \( \delta > 0 \) is a constant, \( \Delta = \max\{\Delta_G, \Delta_{G'}\} \), \( \Delta_G \) denotes the maximum degree of the input graph, and \( \Delta_{G'} \) denotes the maximum degree of the updated graph.

For Ising model, proper \( q \)-coloring and hardcore model, the coupling \( D_{\Omega}^{\sigma,\tau}(\cdot, \cdot) \) in (16) is specified as the optimal coupling of \( \mu_{\nu,\Gamma}(\cdot | \sigma) \) and \( \mu_{\nu,\Gamma}(\cdot | \tau) \) that attains the maximum \( \Pr[\boldsymbol{x} = y] \) for all couplings \((\boldsymbol{x}, \boldsymbol{y})\) of \( \boldsymbol{x} \sim \mu_{\nu,\Gamma}(\cdot | \sigma) \) and \( \boldsymbol{y} \sim \mu_{\nu,\Gamma}(\cdot | \tau) \).

The regimes for the Ising model and proper \( q \)-coloring match the Dobrushin-Shlosman condition (Definition 5.3). Hence, the dynamic sampling results for Ising model and proper \( q \)-coloring are the consequences of Corollary 3.3. The regime for hard core model matches Condition 3.1 with parameters \( \beta = \Omega(1) \), \( C = O(1) \), \( K = O(\Delta) \), on the one-step optimal coupling and the potential function due to Vigoda in [30]. This is specified by the following lemma.

**Lemma 8.2.** Let \( \delta > 0 \) be a constant and \( \mathcal{I} = (V, E, Q, \Phi) \) be a hard core model on graph \( G = (V, E) \) with fugacity \( \lambda \). Let \( \Delta \) denote the maximum degree of \( G \). There exists a potential function \( \rho_{\mathcal{I}} \) such that if \( \lambda \leq \frac{2-\delta}{3\Delta} \), then \( \mathcal{I} \) satisfies Condition 3.1 with parameters \( \beta = \frac{1}{963} \), \( C = 1 \), \( K = 12\Delta \), on the one-step optimal coupling and the potential function \( \rho_{\mathcal{I}} \).

**Proof.** We give a potential function \( \rho_{\mathcal{I}} \) for the hard core instance \( \mathcal{I} \). Then we show \( \mathcal{I} \) satisfies Condition 3.1 with respect to \( \rho_{\mathcal{I}} \) and the one-step optimal coupling of the Gibbs sampling. We mainly use Vigoda’s potential function in [30]. However, we need to slightly modify Vigoda’s potential function to handle the isolated vertices.

Recall that for hard core model, \( Q = \{0, 1\} \). For any \( \sigma \in Q^V \), \( \sigma(v) = 1 \) represents \( v \) is occupied and \( \sigma(v) = 0 \) represents \( v \) is unoccupied. For each vertex \( v \in V \), we use \( \deg(v) \) to denote the degree of \( v \) in graph \( G = (V, E) \). We divide the graph \( G = (V, E) \) into two graphs \( G_1 = (V_1, E_1) \) and \( G_2 = (V_2, E_2) \) such that

\[
V_1 = \{v \in V \mid \deg(v) = 0\}, \quad E_1 = \emptyset, \\
V_2 = V \setminus V_1, \quad E_2 = E.
\]

Thus \( G_1 \) is an empty graph and \( G_2 \) contains no isolated vertex. The potential function \( \rho_{\mathcal{I}} \) is defined as

\[
\forall \sigma, \tau \in \Omega_{\mathcal{I}}: \quad \rho_{\mathcal{I}}(\sigma, \tau) = 4\rho_1(\sigma(V_1), \tau(V_1)) + 4\rho_2(\sigma(V_2), \tau(V_2)).
\]

Here, \( \rho_1 \) is the potential function on \( G_1 \), which is the Hamming distance:

\[
\rho_1(\sigma(V_1), \tau(V_1)) = \sum_{v \in V_1} 1[\sigma(v) \neq \tau(v)].
\]

And \( \rho_2(\sigma(V_2), \tau(V_2)) \) is the Vigoda’s potential function [30] on the graph \( G_2 \). Formally, let \( D = \{v \in V_2 \mid \sigma(v) \neq \tau(v)\} \). For each \( v \in V_2 \), let \( d_v = |D \cap \Gamma_{G_2}(v)| \). Let \( c = \frac{\Delta}{\Delta+2} \), where \( \Delta \) is the maximum degree of graph \( G \). Note that the maximum degree of graph \( G_2 \) is also \( \Delta \). The potential function \( \rho_2(\sigma(V_2), \tau(V_2)) \) is defined as

\[
\alpha_v = \begin{cases} 
\deg(v) & \text{if } v \in D \\
0 & \text{otherwise;}
\end{cases} \quad \beta_v = \begin{cases} 
-cd_v & \text{if } \exists w \in \Gamma_{G_2}(v) \text{ such that } \sigma(w) = \tau(w) = 1 \\
-\epsilon(d_v - 1) & \text{if there is no such } w \text{ and } d_v > 1 \\
0 & \text{otherwise;}
\end{cases}
\]

32
\[
\rho_2(\sigma(V_2), \tau(V_2)) = \sum_{v \in V_2} (\alpha_v + \beta_v).
\]

It is easy to see \(\rho_{\Omega}(\sigma, \sigma) = 0\) and \(\max_{\sigma, \tau \in \Omega_2} \rho_{\Omega}(\sigma, \tau) = \operatorname{Poly}(n)\). We then verify other properties for \(\rho_{\Omega}\).

At first, we prove the lower-distortion to Hamming property. For function \(\rho_1\), it holds that
\[
\rho_1(\sigma(V_1), \tau(V_1)) = H(\sigma(V_1), \tau(V_1)).
\]

For function \(\rho_2\), it holds that
\[
\rho_2(\sigma(V_2), \tau(V_2)) = \sum_{v \in V_2} (\alpha_v + \beta_v) = \sum_{v \in D} \alpha_v + \sum_{v \in V_2} \beta_v \geq \sum_{v \in D} \sum_{w \in \Gamma_{G_2}(v)} (1 - c),
\]
where the last inequality holds due to \(\sum_{v \in V_2} \beta_v \geq -\sum_{v \in V_2} \epsilon v = -c \sum_{v \in \Gamma_{G_2}(v)} \deg(v)\). Since graph \(G_2\) contains no isolated vertex, then \(|\Gamma_{G_2}(v)| = \deg(v) \geq 1\) for all \(v \in D\). Note \(c < 1\). Thus
\[
\rho_2(\sigma(V_2), \tau(V_2)) \geq |D|(1 - c) = |D| \frac{2}{\Delta + 2} \geq \frac{|D|}{4} = \frac{1}{4} H(\sigma(V_2), \tau(V_2)),
\]
where \(\frac{2}{\lambda + 2} \geq \frac{1}{4}\) is because \(\lambda < \frac{2}{\Delta + 2}\) and \(\Delta \geq 3\). Combining together we have
\[
\rho_{\Omega}(\sigma, \tau) = 4\rho_1(\sigma(V_1), \tau(V_1)) + 4\rho_2(\sigma(V_2), \tau(V_2)) \geq H(\sigma, \tau).
\]

This also implies \(\rho_{\Omega}(\sigma, \tau) \geq 1 [\sigma \neq \tau]\).

Next, we show the function \(\rho_{\Omega}\) is \(12\Delta\)-Lipschitz. Recall \(V_1 \cap V_2 = \emptyset\), \(V_1 \cup V_2 = V\) and
\[
\rho_{\Omega}(\sigma, \tau) = 4\rho_1(\sigma(V_1), \tau(V_1)) + 4\rho_2(\sigma(V_2), \tau(V_2)).
\]

Since \(\rho_1\) is the Hamming distance, it is easy to see \(\rho_1\) is \(1\)-Lipschitz. To give the Lipschitz constant for \(\rho_2\). We extend the function \(\rho_2\) as follows. Suppose the function \(\rho_2\) is defined over \(Q^{V_2} \times Q^{V_2}\), where \(Q = \{0, 1\}\). For any \(x, y, x', y' \in Q^{V_2}\) such that \(H(x, y, x', y') = 1\), it is easy to verify the extended function \(\rho_2\) satisfies
\[
|\rho_2(x, y) - \rho_2(x', y')| \leq 3\Delta.
\]

This implies the original function \(\rho_2\) is \(3\Delta\)-Lipschitz. Hence, the function \(\rho_{\Omega}\) is \(12\Delta\)-Lipschitz.

Finally, we prove the step-wise decay property. Let \(\left(\mathbf{X}_t^{(1)}\right)_{t \geq 0}, \left(\mathbf{Y}_t^{(1)}\right)_{t \geq 0}\) be the Gibbs sampling chains for hard core model on graph \(G_1\). Since \(G_1\) is a graph consisting of isolated vertices, then the one step optimal coupling \(\left(\mathbf{X}_t^{(1)}, \mathbf{Y}_t^{(1)}\right)_{t \geq 0}\) satisfies
\[
\forall \sigma, \tau \in \Omega : \mathbb{E} \left[\rho_1 \left(\mathbf{X}_t^{(1)}, \mathbf{Y}_t^{(1)}\right) | \mathbf{X}_{t-1}^{(1)} = \sigma(V_1) \land \mathbf{Y}_{t-1}^{(1)} = \tau(V_1)\right] \leq \left(1 - \frac{1}{|V_1|}\right) \rho_1(\sigma(V_1), \tau(V_1)).
\]

Let \(\left(\mathbf{X}_t^{(2)}\right)_{t \geq 0}, \left(\mathbf{Y}_t^{(2)}\right)_{t \geq 0}\) be the Gibbs sampling chains for hard core model on graph \(G_2\). If \(\lambda \leq \frac{2 - \delta}{\Delta - 2} = \frac{2(1 - \delta/3)}{\Delta - 2}\), then due to Vigoda’s proof\(^5\), the one step optimal coupling \(\left(\mathbf{X}_t^{(2)}, \mathbf{Y}_t^{(2)}\right)_{t \geq 0}\) satisfies:
\[
\forall \sigma, \tau \in \Omega : \mathbb{E} \left[\rho_2 \left(\mathbf{X}_t^{(2)}, \mathbf{Y}_t^{(2)}\right) | \mathbf{X}_{t-1}^{(2)} = \sigma(V_2) \land \mathbf{Y}_{t-1}^{(2)} = \tau(V_2)\right] \leq \left(1 - \frac{\delta}{96|V_2|}\right) \rho_2(\sigma(V_2), \tau(V_2)).
\]

\(^5\)It can be verified that in Vigoda’s proof [30], the Markov chain for sampling hard core is indeed the Gibbs sampling and the coupling for analysis is indeed the one step-optimal coupling for Gibbs sampling.
Let \((X_t)_{t\geq 0}, (Y_t)_{t\geq 0}\) be the Gibbs sampling chains for hard core model on graph \(G\). If \(\lambda \leq \frac{2-\delta}{\delta}\), then the one step optimal coupling \((X_t, Y_t)_{t\geq 0}\) satisfies:

\[
\forall \sigma, \tau \in \Omega_I : \quad \mathbb{E}[\rho_I (X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] = \frac{|V_1|}{n} \left( 1 - \frac{1}{|V_1|} \right) 4\rho_1(\sigma(V_1), \tau(V_1)) + 4\rho_2(\sigma(V_2), \tau(V_2)) \\
+ \frac{|V_2|}{n} \left( 4\rho_1(\sigma(V_1), \tau(V_1)) + \left( 1 - \frac{\delta}{96|V_2|} \right) 4\rho_2(\sigma(V_2), \tau(V_2)) \right) \\
\leq \left( 1 - \frac{\min\{|\delta/96, 1|\}}{n} \right) \rho_I(\sigma, \tau).
\]

Thus, the potential function \(\rho_I\) satisfies the step-wise decay property.

\[
\forall \sigma, \tau \in \Omega_I : \quad \mathbb{E}[\rho_I (X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] \leq \left( 1 - \frac{\delta/96}{n} \right) \rho_I(\sigma, \tau).
\]

Hence, \(I\) satisfies the Condition 3.1 with parameters \(\beta = \frac{\delta}{96}, C = 1\) and \(K = 12\Delta\).

9 Proofs for Dynamic CFTP

In this section, we give the proofs of our main results on CFTP. We introduce the following random variables. Consider the algorithm \(\text{CFTP}(I, X_{\text{init}}, T, v, r)\) in Algorithm 7. Define the random variable \(T_{\text{min}}(I) \in \mathbb{Z}_{\geq 0}\) as

\[
T_{\text{min}}(I) \triangleq \min \{ k : \text{CFTP}(I, X_{\text{init}}, k, v, r) \text{ returns a collapsed } X_0 \} .
\]

The random variable \(T_{\text{min}}(I)\) can be defined for monotone CFTP (if \(I\) is a monotone system), anti-monotone CFTP (if \(I\) is an anti-monotone system) and bounding chains on \(I\). We will use \(T_{\text{min}}\) to denote \(T_{\text{min}}(I)\) when \(I\) is clear in the context. Consider a CFTP algorithm (Algorithm 2 or Algorithm 3) on instance \(I\). Denote its time complexity as \(T_{\text{CFTP}}(I)\). A well-known relation [26] between \(T_{\text{min}}(I)\) and \(T_{\text{CFTP}}(I)\) is

\[
T_{\text{CFTP}}(I) \leq 4T_{\text{min}}(I) .
\]

We then prove Theorem 3.7 and Theorem 3.6 for monotone systems (Section 9.1), anti-monotone systems (Section 9.2) and bound chains (Section 9.3) respectively.

9.1 Monotone systems

For monotone systems, we prove the convergence results for static CFTP and dynamic CFTP under the following condition. This proves Theorem 3.7 and Theorem 3.6 on monotone systems.

**Condition 9.1.** Let \(I = (V, E, Q, \Phi)\) be a monotone MRF instance with respect to a locally-defined grand coupling \(g(\cdot, \cdot)\) (Definition 5.5). Instance \(I\) satisfies Condition 3.1 with the same grand coupling \(g(\cdot, \cdot)\) and parameters \(\beta = \Omega(1), C, K > 0\);

9.1.1 Static monotone CFTP

In this section, we prove Theorem 3.7 on monotone systems. The following lemma bounds \(T_{\text{min}}\) defined in (40) for monotone CFTP. Theorem 3.7 on monotone systems is a consequence of the following lemma and Inequality (41).
Lemma 9.2. Let $I = (V, E, Q, \Phi)$ be a monotone MRF instance with $n = |V|$. Assume Condition 9.1 holds on instance $I$. It holds that $E[T_{\min}] = O\left(\frac{n}{\beta} \log n\right)$ and $T_{\min} = O\left(\frac{n}{\beta} \log n\right)$ with probability at least $1 - 1/n^2$.

Proof. Suppose the instance $I$ satisfies the Condition 9.1 with the potential function $\rho_I$ and the grand coupling for Gibbs sampling $(X_t, Y_t)_{t \geq 0}$. Then, we have

$$\forall X_0, Y_0 \in \Omega_I : \Pr[X_t \neq Y_t] \leq E[\rho_I(X_t, Y_t)] \leq \left(1 - \frac{\beta}{n}\right)^t \rho_I(X_0, Y_0).$$

Consider the monotone CFTP. The chain starts from the time $-T$ such that $X_{-T} = X_{\max}, Y_{-T} = X_{\min}$. It is assumed that $\phi_v(c_{\min}), \phi_e(c_{\max}) > 0$ and $\phi_e(c_{\min}, c_{\min}), \phi_e(c_{\max}, c_{\max}) > 0$ for all constraints $\phi_v(\cdot), \phi_e(\cdot, \cdot)$ in $I$. Then it holds that $X_{\min}, X_{\max} \in \Omega_I$. Similarly, we have the following result about $\Pr[X_0 \neq Y_0]$:

$$\Pr[X_0 \neq Y_0] \leq E[\rho_I(X_0, Y_0)] \leq \left(1 - \frac{\beta}{n}\right)^T \rho_I(X_{-T}, Y_{-T}) = \left(1 - \frac{\beta}{n}\right)^T \rho_I(X_{\max}, X_{\min}).$$

Note that the chain collapses if $X_0 = Y_0$. Let $D = \max_{\sigma, \tau \in \Omega_I} \rho_I(\sigma, \tau)$. For any $T > 0$, we have

$$\Pr[T_{\min} > T] \leq \left(1 - \frac{\beta}{n}\right)^T D \leq \exp\left(-\frac{T\beta}{n}\right) D$$

(42)

Since $D = \text{Poly}(n)$, we assume $D \leq n^d$ for some constant $d = O(1)$. Define

$$T^* \triangleq \left\lceil \frac{n}{\beta} \log(n^{2+d}) \right\rceil.$$  (43)

By (42), we have

$$\Pr[T_{\min} > T^*] \leq \frac{1}{n^2}.$$

We have $T^* = O\left(\frac{n}{\beta} \log n\right)$. This implies $T_{\min} = O\left(\frac{n}{\beta} \log n\right)$ with probability at least $1 - \frac{1}{n^2}$.

We then bound the expectation of $T_{\min}$ as follows:

$$E[T_{\min}] = \sum_{t=0}^{\infty} \Pr[T_{\min} > t]$$

By (42) \quad \leq T^* + \sum_{t=T^*}^{\infty} \left(1 - \frac{\beta}{n}\right)^T D$$

By (43) \quad \leq T^* + \frac{1}{n^2} \sum_{t=0}^{\infty} \left(1 - \frac{\beta}{n}\right)^t $$

$$= O(T^*).$$

Thus, we have $E[T_{\min}] = O(\frac{n}{\beta} \log n).$ \hfill \Box

9.1.2 Dynamic monotone CFTP

In this section, we prove Theorem 3.6 on monotone systems. We need the following lemma to bound the expectation of $R$ for dynamic monotone CFTP.
Lemma 9.3. Let \( I = (V, E, Q, \Phi) \) be the input MRF instance and \( I' \) be the updated instance. Assume Condition 9.1 holds on both \( I \) and \( I' \) with the same locally-defined grand coupling \( g(\cdot, \cdot) \) and the same parameters \( \beta, C, K > 0 \). Let \( n = |V| \) and \( \Delta \) denote the maximum degree of graph \( G = (V, E) \). It holds that \( \mathbb{E}[R] = O\left(\frac{\Delta T^CCK}{n^\beta}\right) \), where \( R \) is defined as (15) by the one-step local coupling for dynamic instances constructed as (24).

We first prove Theorem 3.6 on monotone systems, then prove Lemma 9.3.

Proof of Theorem 3.6 on monotone systems. For dynamic monotone CFTP, by (43), we can set \( T_{\text{couple}}(I) = \left\lceil \frac{n}{\beta} \log((n')^{2+d}) \right\rceil \) and set \( T_{\text{couple}}(I') = \left\lceil \frac{n'}{\beta} \log((n')^{2+d}) \right\rceil \), where \( n = |V| \) is the number of variables in \( I \), \( n' = |V'| \) is the number of variables in \( I' \) and \( d = O(1) \) is a constant. Then, by combining Lemma 7.2 and Lemma 9.3, we have the result holds.

Proof of Lemma 9.3. Consider the algorithm that updates the execution-log CFTP-Log\((I) = \langle v_t, \mathcal{X}_t(v_t) \rangle_{t=-T'+1}^{0} \) to CFTP-Log\((I') = \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t=-T'+1}^{0} \). Since the value of \( T' \) is fixed, we can change the indices and assume that the algorithm updates the log CFTP-Log\((I) = \langle v_t, \mathcal{X}_t(v_t) \rangle_{t=1}^{T'} \) to CFTP-Log\((I') = \langle v_t, \mathcal{Y}_t(v_t) \rangle_{t=1}^{T'} \). The \( \mathcal{X}_0 \) and \( \mathcal{Y}_0 \) are defined in (21).

We bound the expectation of \( R \) in the same way as in (25). Thus

\[
\mathbb{E}[R] \leq \frac{(\Delta + 1)}{n} \sum_{t=1}^{T'} \mathbb{E}[|D_{t-1}|] + \frac{2T'}{n}.
\]

We claim that for dynamic monotone CFTP, if \( I \) and \( I' \) both satisfy the Condition 9.1 with the same grand coupling and the same parameters \( \beta, C, K \), then it holds that

\[
\forall 0 \leq t \leq T' : \quad \mathbb{E}[|D_t|] = O\left(\frac{CK}{\beta}\right).
\]

The lemma is proved by combining above two results.

We now prove (44). Recall that for any \( 0 \leq t \leq T' \), the set \( D_t \) is defined as

\[
D_t = \{ v \in V \mid \mathcal{X}_t(v) \neq \mathcal{Y}_t(v) \}.
\]

Recall, for any \( 0 \leq t \leq T' \) and any \( v \in V \), \( \mathcal{X}_t(v) \) and \( \mathcal{Y}_t(v) \) are defined as

\[
\mathcal{X}_t(v) = (\mathcal{X}_t^{(1)}(v), \mathcal{X}_t^{(2)}(v)),
\]

\[
\mathcal{Y}_t(v) = (\mathcal{Y}_t^{(1)}(v), \mathcal{Y}_t^{(2)}(v)).
\]

For any \( 0 \leq t \leq T' \), define the random sets \( D_t^{(1)} \) and \( D_t^{(2)} \) as

\[
D_t^{(1)} = \{ v \in V \mid \mathcal{X}_t^{(1)}(v) \neq \mathcal{Y}_t^{(1)}(v) \},
\]

\[
D_t^{(2)} = \{ v \in V \mid \mathcal{X}_t^{(2)}(v) \neq \mathcal{Y}_t^{(2)}(v) \}.
\]

It is easy to see that \( D_t = D_t^{(1)} \cup D_t^{(2)} \). Thus we have

\[
\forall 0 \leq t \leq T' : \quad \mathbb{E}[|D_t|] \leq \mathbb{E}[|D_t^{(1)}| + |D_t^{(2)}|] = \mathbb{E}[|D_t^{(1)}|] + \mathbb{E}[|D_t^{(2)}|].
\]

Note that \( (\mathcal{X}_t^{(1)})_{t=0}^{T'} \) is the Gibbs sampling chain on instance \( I \) and \( (\mathcal{Y}_t^{(1)})_{t=0}^{T'} \) is the Gibbs sampling chain on instance \( I' \). Note that the one-step local coupling for dynamic instances is constructed as (24). Then, the following properties hold.
• \( \mathbf{X}_0 = \mathbf{Y}_0 = \mathbf{X}_{\mathcal{I}, \max}, \) and \( \mathbf{X}_0^{(1)} \in \Omega_{\mathcal{I}}, \mathbf{Y}_0^{(1)} \in \Omega_{\mathcal{I}}. \)

• \( (\mathbf{X}_t^{(1)}, \mathbf{Y}_t^{(1)}), t = 0 \) is coupled by the one-step local coupling for dynamic instances (Definition 6.2) with \( D^\mathcal{I}_{\mathcal{I}} (\cdot, \cdot) \) specified as for any \( v \in \mathcal{V}, \sigma \in Q^g(v) \) and \( \tau \in Q^g(v): \)

\[
\forall c, c' \in Q, \quad D^\mathcal{I}_{\mathcal{I}} (c, c') = \Pr_{r \in [0,1]} \left[ g^\mathcal{I}_{\mathcal{I}} (\sigma, r) = c \land g^\mathcal{I}_{\mathcal{I}} (\tau, r) = c' \right].
\]

• \( \mathcal{I} \) and \( \mathcal{I}' \) both satisfy the Condition 9.1 with the same grand coupling \( g(\cdot, \cdot) \) (Definition 5.4) and the same parameters \( \beta, C, K. \)

By going through the proof of Lemma 6.4, it can be verified that

\[
\forall 0 \leq t \leq T': \quad \mathbb{E}\left[ |D_t^{(1)}| \right] = O \left( \frac{CK}{\beta} \right).
\]

(45)

Similar result holds for chains \( (\mathbf{X}_t^{(2)}, \mathbf{Y}_t^{(2)}), t = 0 \)

\[
\forall 0 \leq t \leq T': \quad \mathbb{E}\left[ |D_t^{(2)}| \right] = O \left( \frac{CK}{\beta} \right).
\]

(46)

Combining (45) and (46) proves (44).

\[ \square \]

9.2 Anti-monotone systems

For anti-monotone systems, we prove the convergence results for static CFTP and dynamic CFTP under the following condition. This proves Theorem 3.7 and Theorem 3.6 on anti-monotone systems.

**Condition 9.4.** Let \( \mathcal{I} = (V, E, Q, \Phi) \) be an anti-monotone MRF instance with respect to a locally-defined grand coupling \( g(\cdot, \cdot) \) (Definition 5.5). Instance \( \mathcal{J} = \text{Par}(\mathcal{I}) \) (Definition 3.5) satisfies Condition 3.1 with the same grand coupling \( g(\cdot, \cdot) \) and the parameters \( \beta = \Omega(1), C, K > 0, \) and with locally-defined potential function \( \rho_{\mathcal{J}} \) (Definition 3.4).

We first introduce some notations and definitions, then prove the main results on anti-monotone systems.

Let \( \mathcal{I} = (V, E, Q, \Phi) \). Recall that instance \( \text{Par}(\mathcal{I}) = \mathcal{J} = (V_{\mathcal{J}}, E_{\mathcal{J}}, Q, \Phi_{\mathcal{J}}) \) is defined on the bipartite graph \( G_{\mathcal{J}} = (V_{\mathcal{J}}, E_{\mathcal{J}}) \). The vertex set \( V_{\mathcal{J}} \) is defined as \( V_{\mathcal{J}} = V_1 \cup V_2 \), where \( V_1 \) and \( V_2 \) are duplicates of \( V \) such that each \( v \in V \) corresponds to a \( v_1 \in V_1 \) and a \( v_2 \in V_2 \); the edge set \( E_{\mathcal{J}} \) is defined as \( E_{\mathcal{J}} = \{ \{u_1, v_2\}, \{v_1, u_2\} \mid \{u, v\} \in E\} \).

By the construction of the graph \( G_{\mathcal{J}} = (V_{\mathcal{J}}, E_{\mathcal{J}}) \). For any \( v \in V \), the distance between \( v_1 \in V_1 \) and \( v_2 \in V_2 \) is least 3 in \( G_{\mathcal{J}} \). The following proposition holds.

**Observation 9.5.** For any \( v_1 \in V_1, v_2 \in V_2 \), it holds that \( \Gamma^+_G(v_1) \cap \Gamma^+_G(v_2) = \emptyset. \)

We define the following bipartite Gibbs sampling \( (\mathbf{X}_t)_{t \geq 0} \) on instance \( \mathcal{J} = \text{Par}(\mathcal{I}) \), which will be used to analyze the static and dynamic anti-monotone CFTP.

**Definition 9.6 (bipartite Gibbs sampling).** Let \( \mathcal{I} = (V, E, Q, \Phi) \) be an anti-monotone MRF instance with respect to a locally-defined grand coupling \( g(\cdot, \cdot) \). Let \( \mathcal{J} = \text{Par}(\mathcal{I}) = (V_{\mathcal{J}}, E_{\mathcal{J}}, Q, \Phi_{\mathcal{J}}) \) and \( H = G_{\mathcal{J}} = (V_{\mathcal{J}}, E_{\mathcal{J}}) \). Recall \( V_{\mathcal{J}} = V_1 \cup V_2 \). Define the bipartite Gibbs sampling \( (\mathbf{X}_t)_{t \geq 0} \) on instance \( \mathcal{J} \) as follows:
• The chain starts from an arbitrary feasible state \(X_0 \in \Omega_J\).

• For \(t = 1, 2, \ldots\), the chain does as follows:
  1. pick \(v \in V\) uniformly at random and let \(X_t(u) \leftarrow X_{t-1}(u)\) for all \(u \neq v_1 \land u \neq v_2\), where \(v_1 \in V_1\) and \(v_2 \in V_2\) are vertices corresponding to \(v\);
  2. sample \(r \in [0, 1]\) uniformly at random and set

\[
\begin{align*}
X_t(v_1) &\leftarrow g_{J_{v_1}}(X_{t-1}(\Gamma_H(v_1)), r), \\
X_t(v_2) &\leftarrow g_{J_{v_2}}(X_{t-1}(\Gamma_H(v_2)), r).
\end{align*}
\]

(47)

We then define the grand coupling for bipartite Gibbs sampling.

**Definition 9.7 (grand coupling for bipartite Gibbs sampling).** Let \(I = (V, E, Q, \Phi)\) be an anti-monotone MRF instance with respect to a locally-defined grand coupling \(g(\cdot, \cdot)\). Let \(J = \text{Par}(I) = (V_J, E_J, Q_J, \Phi_J)\) and \(H = G_J = (V_H, E_H)\). Recall \(V_J = V_1 \cup V_2\). The grand coupling \((X_t, Y_t)_{t \geq 0}\) of bipartite Gibbs sampling on \(J\) with \(g(\cdot, \cdot)\) is constructed as follows: for \(t = 1, 2, \ldots\),

• pick the same random \(v \in V\), and let \((X_t(u), Y_t(u)) \leftarrow (X_{t-1}(u), Y_{t-1}(u))\) for all \(u \neq v_1 \land u \neq v_2\), where \(v_1 \in V_1\) and \(v_2 \in V_2\) are vertices corresponding to \(v\);

• sample \(r \in [0, 1]\) uniformly at random and set

\[
\begin{align*}
X_t(v_1) &\leftarrow g_{J_{v_1}}(X_{t-1}(\Gamma_H(v_1)), r), \\
X_t(v_2) &\leftarrow g_{J_{v_2}}(X_{t-1}(\Gamma_H(v_2)), r), \\
Y_t(v_1) &\leftarrow g_{J_{v_1}}(Y_{t-1}(\Gamma_H(v_1)), r), \\
Y_t(v_2) &\leftarrow g_{J_{v_2}}(Y_{t-1}(\Gamma_H(v_2)), r).
\end{align*}
\]

We define the following condition with respect to grand coupling for bipartite Gibbs sampling, which is similar to Condition 3.1.

**Condition 9.8.** Let \(\beta, C, K > 0\), and \(I = (V, E, Q, \Phi)\) be an anti-monotone MRF instance with respect to a locally-defined grand coupling \(g(\cdot, \cdot)\). Let \(J = \text{Par}(I) = (V_J, E_J, Q_J, \Phi_J)\) with \(n = |V_J|\), and \(\Omega_J \subseteq Q^{|V_J|}\) be the feasible set. There exists a locally-defined potential function \(\rho_J : \Omega_J \times \Omega_J \rightarrow \mathbb{R}_{\geq 0}\), where \(\forall \sigma, \tau \in \Omega_J\), \(\rho_J(\sigma, \tau) = 0\) if \(\sigma = \tau\) and \(\rho_J(\sigma, \tau) \geq 1\) if \(\sigma \neq \tau\), and \(\text{Diam}_J \triangleq \max_{\sigma, \tau \in \Omega_J} \rho_J(\sigma, \tau) \leq \text{Poly}(n)\), such that

1. **(step-wise decay)** for the grand coupling of bipartite Gibbs sampling \((X_t, Y_t)_{t \geq 0}\) on \(J\) with \(g(\cdot, \cdot)\) (Definition 9.7),

\[
\forall \sigma, \tau \in \Omega_J : \quad \mathbb{E} [\rho_J(X_t, Y_t) | X_{t-1} = \sigma \land Y_{t-1} = \tau] \leq \left(1 - \frac{\beta}{n}\right) \cdot \rho_J(\sigma, \tau);
\]

2. **(low-distortion to Hamming)** for all \(\sigma, \tau \in \Omega_J\), \(H(\sigma, \tau) \leq C \cdot \rho_J(\sigma, \tau)\), where \(H(\sigma, \tau)\) denotes the Hamming distance between \(\sigma\) and \(\tau\);

3. **(Lipschitz)** function \(\rho_J(\cdot, \cdot)\), seen as a function of \(2n\) variables, is \(K\)-Lipschitz, that is,

\[
\max_{\sigma, \sigma', \tau, \tau' \in \Omega_J} |\rho_J(\sigma, \tau) - \rho_J(\sigma', \tau')| \leq K \cdot H(\sigma \tau, \sigma' \tau').
\]

The following lemma shows the relation between Condition 9.4 and Condition 9.8.
Lemma 9.9. Let \( I = (V, E, Q, \Phi) \) be an anti-monotone MRF instance and \( J = \text{Par}(I) \). If \( I \) satisfies Condition 9.4 with the locally-defined grand coupling \( g(\cdot, \cdot) \) and the parameters \( \beta, C, K \), and with the locally-defined potential function \( \rho_J \), then \( I \) satisfies Condition 9.8 with the same locally-defined grand coupling \( g(\cdot, \cdot) \) and the same parameters \( \beta, C, K \), and with the same locally-defined potential function \( \rho_J \).

Proof. The low-distortion to Hamming property and Lipschitz property must hold for the same function \( \rho_J \). We prove the step-wise decay property under the grand coupling of bipartite Gibbs sampling on \( J \).

Let \( J = \text{Par}(I) = (V_1 \cup V_2, E_J, Q, \Phi_J) \). Recall that for any vertex \( v \in V \), there are two corresponding vertices \( v_1 \in V_1 \) and \( v_2 \in V_2 \). We use \( H \) to denote the bipartite graph \( G_J = (V_1 \cup V_2, E_J) \). We use \( \Gamma(v) = \Gamma_H(v) \) to denote the neighborhood of \( v \) in \( H \).

Let \( (X_t, Y_t) \geq 0 \) be the grand coupling for bipartite Gibbs sampling (Definition 9.7) on instance \( J \). Let \( (X'_t, Y'_t) \geq 0 \) be the grand coupling for Gibbs sampling on instance \( J \). Let \( n = |V_1 \cup V_2| \).

By Condition 9.4, for any \( \sigma, \tau \in \Omega_J \), we have

\[
\mathbb{E}[\rho_J(X'_t, Y'_t) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau] \leq \left(1 - \frac{\beta}{n}\right) \rho_J(\sigma, \tau).
\]

We claim that for any \( \sigma, \tau \in \Omega_J \),

\[
\mathbb{E}[\rho_J(X_t, Y_t) - \rho_J(\sigma, \tau) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] = 2 \mathbb{E}[\rho_J(X'_t, Y'_t) - \rho_J(\sigma, \tau) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau].
\]

(48)

Hence, for any \( \sigma, \tau \in \Omega_J \),

\[
\mathbb{E}[\rho_J(X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] \leq \left(1 - \frac{2\beta}{n}\right) \rho_J(\sigma, \tau) \leq \left(1 - \frac{\beta}{n}\right) \rho_J(\sigma, \tau).
\]

This proves the step-wise decay property in Condition 9.8. The lemma is proved.

We now prove Equation (48). Fix \( \sigma, \tau \in \Omega_J \). Since \( \rho_J \) is a locally-defined potential function, then we have

\[
\rho_J(\sigma, \tau) = \sum_{w \in V_1 \cup V_2} \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)).
\]

Consider the bipartite Gibbs sampling. We define the following events

- \( \mathcal{E}_1 : X_{t-1} = \sigma \land Y_{t-1} = \tau \).
- \( \mathcal{E}_2(v_1, v_2) \) : the vertices \( v_1 \in V_1 \) and \( v_2 \in V_2 \) are picked in \( t \)-th transition.

By Observation 9.5, the set \( \Gamma^+_{v_1} \cap \Gamma^+_{v_2} = \emptyset \). Conditioning on \( \mathcal{E}_2(v_1, v_2) \), only the spin at \( v_1 \) and \( v_2 \) are updated. Hence

\[
\mathbb{E}[\rho_J(X_t, Y_t) \mid \mathcal{E}_1 \land \mathcal{E}_2(v_1, v_2)] - \rho_J(\sigma, \tau)
\]

\[
= \sum_{w \in \Gamma^+_{v_1}} \left( \mathbb{E}[\rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid \mathcal{E}_1 \land \mathcal{E}_2(v_1, v_2)] - \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)) \right)
\]

\[
+ \sum_{w \in \Gamma^+_{v_2}} \left( \mathbb{E}[\rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid \mathcal{E}_1 \land \mathcal{E}_2(v_1, v_2)] - \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)) \right).
\]

39
Note that $n = |V_1 \cup V_2|$. Since each pair $(v_1, v_2)$ is picked with probability $\frac{2}{n}$, then we have

$$
E \left[ \rho_{\mathcal{J}}(X_t, Y_t) \mid E_1 \right] - \rho_{\mathcal{J}}(\sigma, \tau) = \frac{2}{n} \sum_{v_1 \in V_1} \sum_{w \in \Gamma^+_w} \left( E \left[ \rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid E_1 \land E_2(v_1, v_2) \right] - \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)) \right) + \frac{2}{n} \sum_{v_2 \in V_2} \sum_{w \in \Gamma^+_w} \left( E \left[ \rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid E_1 \land E_2(v_1, v_2) \right] - \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)) \right).
$$

Consider the Gibbs sampling on instance $\mathcal{J}$. We define the following events:

- $E'_1 : X'_{t-1} = \sigma \land Y'_{t-1} = \tau$.
- $E'_2(v_b) :$ the vertex $v_b$ is picked in $t$-th transition, where $b \in \{1, 2\}$, $v_b \in V_1 \cup V_2$.

Conditioning on $E'_2(v_b)$, only the spin at $v_b$ are updated. Then we have

$$
E \left[ \rho_{\mathcal{J}}(X'_t, Y'_t) \mid E'_1 \land E'_2(v_b) \right] - \rho_{\mathcal{J}}(\sigma, \tau) = \sum_{w \in \Gamma^+_w} \left( E \left[ \rho_w(X'_t(\Gamma^+_w), Y'_t(\Gamma^+_w)) \mid E'_1 \land E'_2(v_b) \right] - \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)) \right).
$$

Note that $n = |V_1 \cup V_2|$. Since each $v_b$ is picked with probability $\frac{1}{n}$, we have

$$
E \left[ \rho_{\mathcal{J}}(X'_t, Y'_t) \mid E'_1 \right] - \rho_{\mathcal{J}}(\sigma, \tau) = \frac{1}{n} \sum_{v_1 \in V_1} \sum_{w \in \Gamma^+_w} \left( E \left[ \rho_w(X'_t(\Gamma^+_w), Y'_t(\Gamma^+_w)) \mid E'_1 \land E'_2(v_1) \right] - \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)) \right) + \frac{1}{n} \sum_{v_2 \in V_2} \sum_{w \in \Gamma^+_w} \left( E \left[ \rho_w(X'_t(\Gamma^+_w), Y'_t(\Gamma^+_w)) \mid E'_1 \land E'_2(v_2) \right] - \rho_w(\sigma(\Gamma^+_w), \tau(\Gamma^+_w)) \right).
$$

Consider (49). Since $\Gamma^+_w \cap \Gamma^+_w = \emptyset$, the $E \left[ \rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid E_1 \land E_2(v_1, v_2) \right]$ for $w \in \Gamma^+_v$ only take over the randomness of $X_t(v_1), Y_t(v_1)$. And the $E \left[ \rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid E_1 \land E_2(v_1, v_2) \right]$ for $w \in \Gamma^+_v$ only take over the randomness of $X_t(v_2), Y_t(v_2)$. By the definitions of the bipartite Gibbs sampling, after picked a vertex, bipartite Gibbs sampling and Gibbs sampling use the same rule to update its spin. Thus, for all $v_1 \in V_1, v_2 \in V_2$ and $w \in \Gamma^+_v$, it holds that

$$
E \left[ \rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid E_1 \land E_2(v_1, v_2) \right] = E \left[ \rho_w(X'_t(\Gamma^+_w), Y'_t(\Gamma^+_w)) \mid E'_1 \land E'_2(v_1) \right],
$$

and for all $v_1 \in V_1, v_2 \in V_2$ and $w \in \Gamma^+_v$, it holds that

$$
E \left[ \rho_w(X_t(\Gamma^+_w), Y_t(\Gamma^+_w)) \mid E_1 \land E_2(v_1, v_2) \right] = E \left[ \rho_w(X'_t(\Gamma^+_w), Y'_t(\Gamma^+_w)) \mid E'_1 \land E'_2(v_2) \right].
$$

Combining these equations with (49), (50) and the definitions of $E_1, E'_1$, we have

$$
E \left[ \rho_{\mathcal{J}}(X_t, Y_t) - \rho_{\mathcal{J}}(\sigma, \tau) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau \right] = 2 E \left[ \rho_{\mathcal{J}}(X'_t, Y'_t) - \rho_{\mathcal{J}}(\sigma, \tau) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau \right].
$$

This proves Equation (48).
9.2.1 Static anti-monotone CTFP

In this section, we prove Theorem 3.7 on anti-monotone systems. The following lemma bounds $T_{\text{min}}$ defined in (40) for anti-monotone CTFP. Theorem 3.7 on anti-monotone systems is a consequence of the following lemma and Inequality (41).

**Lemma 9.10.** Let $\mathcal{I} = (V, E, Q, \Phi)$ be an anti-monotone MRF instance with $n = |V|$. Assume Condition 9.4 holds on instance $\mathcal{I}$. It holds that $\mathbb{E}[T_{\text{min}}] = O(n^3 \log n)$ and $T_{\text{min}} = O(n^3 \log n)$ with probability at least $1 - 1/n^2$.

Lemma 9.10 is proved by the following lemma.

We introduce some notations to state the lemma. Let $\mathcal{I} = (V, E, Q, \Phi)$ be an anti-monotone MRF instance and $\mathcal{J} = \text{Par}(\mathcal{I}) = (V_{\mathcal{J}}, E_{\mathcal{J}}, Q, \Phi_{\mathcal{J}})$, where $V_{\mathcal{J}} = V_1 \cup V_2$. Let $c_{\min}, c_{\max} \in Q$ be the minimum and maximum spins for instance $\mathcal{I}$. Define the states $Z_{\text{max, min}} \in Q^{V_{\mathcal{J}}}$ and $Z_{\text{min, max}} \in Q^{V_{\mathcal{J}}}$ as follows:

\[
\forall v_1 \in V_1 : \quad Z_{\text{max, min}}(v_1) = c_{\max}, \quad Z_{\text{min, max}}(v_1) = c_{\min}; \\
\forall v_2 \in V_2 : \quad Z_{\text{max, min}}(v_2) = c_{\min}, \quad Z_{\text{min, max}}(v_2) = c_{\max}.
\]

**Lemma 9.11.** Let $\mathcal{I} = (V, E, Q, \Phi)$ be an anti-monotone MRF instance with respect to a locally-defined grand coupling $g(\cdot, \cdot)$, and $\mathcal{J} = \text{Par}(\mathcal{I}) = (V_{\mathcal{J}}, E_{\mathcal{J}}, Q, \Phi_{\mathcal{J}})$, where $V_{\mathcal{J}} = V_1 \cup V_2$. Let $(X_t, Y_t)_{t \geq 0}$ be the grand coupling of bipartite Gibbs sampling (Definition 9.7) on $\mathcal{J}$ such that $X_0 = Z_{\text{max, min}}$ and $Y_0 = Z_{\text{min, max}}$. It holds that

\[
\mathbb{P}[T_{\text{min}} > t] \leq \mathbb{P}[X_t \neq Y_t \mid X_0 = Z_{\text{max, min}} \land Y_0 = Z_{\text{min, max}}].
\]

We first prove Lemma 9.10 from Lemma 9.11. Then we prove Lemma 9.11.

**Proof of Lemma 9.10.** Suppose Condition 9.4 holds for $\mathcal{I}$ with locally-defined potential function $\rho_{\mathcal{J}}$ and parameters $\beta, C, K$. By Lemma 9.9, Condition 9.8 holds for $\mathcal{I}$ with the same locally-defined potential function $\rho_{\mathcal{J}}$ and the same parameters $\beta, C, K$.

Let $(X_t, Y_t)_{t \geq 0}$ be the grand coupling of bipartite Gibbs sampling on $\mathcal{J} = \text{Par}(\mathcal{I})$. Since $|V| = n$, then $|V_{\mathcal{J}}| = 2n$. We have

\[
\forall \sigma, \tau \in \Omega_{\mathcal{J}} : \quad \mathbb{E}[\rho_{\mathcal{J}}(X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau] \leq \left(1 - \frac{\beta}{2n}\right) \rho_{\mathcal{J}}(\sigma, \tau);
\]

It is assumed that $\phi_v(c_{\max}) > 0$, $\phi_v(c_{\min}) > 0$ and $\phi_v(c_{\min}, c_{\max}) > 0$ for all constraints $\phi_v(\cdot, \cdot)$ in $\mathcal{I}$. Then it must hold that $Z_{\text{max, min}}, Z_{\text{min, max}} \in \Omega_{\mathcal{J}}$ due to the definition of the instance $\mathcal{J} = \text{Par}(\mathcal{I})$. Note that $X_0 = Z_{\text{max, min}}$ and $Y_0 = Z_{\text{min, max}}$. It is easy to verify that if $X_{t-1}, Y_{t-1} \in \Omega_{\mathcal{J}}$, then $X_t, Y_t \in \Omega_{\mathcal{J}}$. Let $D = \max_{\sigma, \tau \in \Omega_{\mathcal{J}}} \rho_{\mathcal{J}}(\sigma, \tau)$. Hence for any $t \geq 0$, it holds that

\[
\mathbb{P}[X_t \neq Y_t \mid X_0 \land Y_0] \leq \mathbb{E}[\rho_{\mathcal{J}}(X_t, Y_t) \mid X_0 \land Y_0] \leq \left(1 - \frac{\beta}{2n}\right)^t D.
\]

By Lemma 9.11, we have

\[
\mathbb{P}[T_{\text{min}} > t] \leq \left(1 - \frac{\beta}{2n}\right)^t D.
\]

Since $D = \text{Poly}(n)$, we assume $D \leq n^d$ for some constant $d = O(1)$. By going through the proof of Lemma 9.2, we have

\[
\Pr \left[ T_{\text{min}} > \left(\frac{2n}{\beta} \log(n^2 + d)\right) \right] \leq \frac{1}{n^2} \quad \text{and} \quad \mathbb{E}[T_{\text{min}}] = O \left(\frac{n}{\beta} \log n\right). \quad (52)
\]

This proves the lemma. \(\Box\)
**Proof of Lemma 9.11.** Fix an integer $T > 0$. We prove that

$$\Pr[\mathbf{T}_{\min} > T] \leq \Pr[\mathbf{X}_T \neq \mathbf{Y}_T \mid \mathbf{X}_0 = \mathbf{Z}_{\text{max, min}} \land \mathbf{Y}_0 = \mathbf{Z}_{\text{min, max}}].$$

Let $(\mathbf{X}_t', \mathbf{Y}_t')_{t=0}^T$ denote the anti-monotone CFTP chain on instance $\mathcal{I}$ such that

$$\forall v \in V : \quad X_{-T}'(v) = c_{\max}, \quad Y_{-T}'(v) = c_{\min}. \quad (53)$$

Since $T$ is fixed, we can change the indices of two chains and write them as $(\mathbf{X}_t', \mathbf{Y}_t')_{t=0}^T$.

Let $\mathcal{J} = \text{Par}(\mathcal{I}) = (\mathcal{V}_\mathcal{J}, E_\mathcal{J}, Q, \Phi_\mathcal{J})$, where $\mathcal{V}_\mathcal{J} = V_1 \cup V_2$. Let $(\mathbf{X}_t)_t$ and $(\mathbf{Y}_t)_t$ be the bipartite Gibbs sampling chains on instance $\mathcal{J} = \text{Par}(\mathcal{I})$ such that

$$\mathbf{X}_0 = \mathbf{Z}_{\text{max, min}}, \quad \mathbf{Y}_0 = \mathbf{Z}_{\text{min, max}}. \quad (54)$$

We prove the lemma by constructing a joint process

$$(\mathbf{X}_t', \mathbf{Y}_t', \mathbf{X}_t, \mathbf{Y}_t)_{t=0}^T$$

such that the $(\mathbf{X}_t', \mathbf{Y}_t')_{t=0}^T$ is a copy of the anti-monotone CFTP chain on $\mathcal{I}$ and $(\mathbf{X}_t, \mathbf{Y}_t)_{t=0}^T$ is a copy of the grand coupling for bipartite Gibbs sampling (Definition 9.7) on $\mathcal{J}$.

For any $v \in V$, let $v_1 \in V_1$ be the corresponding vertex for $v$ in $V_1$ and $v_2 \in V_2$ be the corresponding vertex for $v$ in $V_2$. And we will prove the following invariant for this joint process:

$$\forall 0 \leq t \leq T, v \in V : \quad X_t'(v) = X_t(v_1) = Y_t(v_2) \land Y_t'(v) = X_t(v_2) = Y_t(v_1). \quad (55)$$

By (55), if $\mathbf{X}_T = \mathbf{Y}_T$, then $\mathbf{X}_T = \mathbf{Y}_T'$, which implies $\mathbf{T}_{\min} \leq T$. This proves the lemma.

We now define the joint process $(\mathbf{X}_t', \mathbf{Y}_t', \mathbf{X}_t, \mathbf{Y}_t)_{t=0}^T$. Recall the instance $\mathcal{I} = (V, E, Q, \Phi)$ and $\mathcal{J} = (\mathcal{V}_\mathcal{J}, E_\mathcal{J}, Q, \Phi_\mathcal{J})$. Denote $G = (V, E), \ H = (V_\mathcal{J}, E_\mathcal{J})$ and $V_G = V_H$. For each step $t = 1, 2, \ldots$, four chains do as follows.

- **pick a vertex** $v \in V$ u.a.r. and sample a real number $r \in [0, 1]$ u.a.r.;
- **two chains** $\mathbf{X}'$ and $\mathbf{Y}'$ set $X_t'(V \setminus \{v\}) \leftarrow X_{t-1}'(V \setminus \{v\})$ and $Y_t'(V \setminus \{v\}) \leftarrow Y_{t-1}'(V \setminus \{v\})$; and set
  $$X_t'(v) \leftarrow g_L(Y_{t-1}'(\Gamma_G(v)), r), \quad Y_t'(v) \leftarrow g_L(X_{t-1}'(\Gamma_G(v)), r). \quad (56)$$
- **two chains** $\mathbf{X}$ and $\mathbf{Y}$ does as follows:
  1. two chains $\mathbf{X}$ and $\mathbf{Y}$ set $X_t(V_H \setminus \{v_1, v_2\}) \leftarrow X_{t-1}(V_H \setminus \{v_1, v_2\})$ and $Y_t(V_H \setminus \{v_1, v_2\}) \leftarrow Y_{t-1}(V_H \setminus \{v_1, v_2\})$;
  2. two chains $\mathbf{X}$ and $\mathbf{Y}$ set
    $$X_t(v_1) \leftarrow g_{\mathcal{J}_{v_1}}(X_{t-1}(\Gamma_{H}(v_1)), r), \quad Y_t(v_1) \leftarrow g_{\mathcal{J}_{v_1}}(Y_{t-1}(\Gamma_{H}(v_1)), r),$$
    $$X_t(v_2) \leftarrow g_{\mathcal{J}_{v_2}}(X_{t-1}(\Gamma_{H}(v_2)), r), \quad Y_t(v_2) \leftarrow g_{\mathcal{J}_{v_2}}(Y_{t-1}(\Gamma_{H}(v_2)), r). \quad (57)$$

It is easy to verify in the joint process defined above, $(\mathbf{X}_t', \mathbf{Y}_t')_{t=0}^T$ is a copy of anti-monotone CFTP and $(\mathbf{X}_t, \mathbf{Y}_t)_{t=0}^T$ is a copy of the grand coupling for bipartite Gibbs sampling.

Finally, we prove the invariant (55) by the induction on $t$. The invariant holds trivially when $t = 0$ due to (53) and (54). Suppose the invariant (55) holds for $t = k - 1$. Suppose $v \in V$ is the vertex picked in $k$-th transition step. Then, it is easy to verify that

$$\forall v' \in V \setminus \{v\} : \quad X_k'(v') = X_k(v_1') = Y_k(v_2') \land Y_k'(v') = X_k(v_2') = Y_k(v_1').$$
Note that \( H \) is a bipartite graph. If \( u \in V_1 \), then \( \Gamma_H(u) \subseteq V_2 \); if \( u \in V_2 \), then \( \Gamma_H(u) \subseteq V_1 \). Recall \( v \in V \) is the vertex picked in \( k \)-th transition step. It holds that
\[
\{ X'_k(u) \mid u \in \Gamma_G(v) \} = \{ X_{k-1}(u) \mid u \in \Gamma_H(v_2) \} = \{ Y_{k-1}(u) \mid u \in \Gamma_H(v_1) \}.
\]

By the definition of the MRF instance \( J = \text{Par}(I) \), it holds that
\[
\forall v \in V, \quad g_{I_v}(\cdot, \cdot) = g_{J_{v_1}}(\cdot, \cdot) = g_{J_{v_2}}(\cdot, \cdot).
\]

Hence, we have
\[
g_{I_v}(X'_{k-1}(\Gamma_G(v)), r) = g_{J_{v_2}}(X_{k-1}(\Gamma_H(v_2)), r) = g_{J_{v_1}}(Y_{k-1}(\Gamma_H(v_1)), r).
\]

By (56) and (57), we have
\[
Y'_k(v) = X_k(v_2) = Y_k(v_1).
\]

Similarly, we can prove that
\[
X'_k(v) = X_k(v_1) = Y_k(v_2).
\]

This proves the invariant (55) for \( t = k \).

\[\square\]

### 9.2.2 Dynamic anti-monotone CFTP

In this section, we prove Theorem 3.6 on dynamic anti-monotone CFTP. We need the following lemma to bound the expectation of \( R \) for dynamic anti-monotone CFTP.

**Lemma 9.12.** Let \( \mathcal{I} = (V, E, Q, \Phi) \) be the input MRF instance and \( \mathcal{I}' \) be the updated instance. Assume Condition 9.4 holds for both \( \mathcal{I} \) and \( \mathcal{I}' \) with the same locally-defined grand coupling \( g(\cdot, \cdot) \) and the same parameters \( \beta, C, K > 0 \), and with the locally-defined potential functions \( \rho_J \) and \( \rho_{J'} \), where \( J = \text{Par}(I) \) and \( J' = \text{Par}(I') \). Let \( n = |V| \) and \( \Delta \) denote the maximum degree of graph \( G = (V, E) \). It holds that \( \mathbb{E}[R] = O\left( \frac{\Delta T' C K}{n^3} \right) \), where \( R \) is defined as (15) by the one-step local coupling for dynamic instances constructed as (24).

We first prove Theorem 3.6 on anti-monotone systems, then prove Lemma 9.12.

**Proof of Theorem 3.6 on anti-monotone systems.** For dynamic anti-monotone CFTP, by (52), we can set \( T_{\text{couple}}(\mathcal{I}) \) as \( \lceil 2n \beta \log(n^{2+d}) \rceil \) and set \( T_{\text{couple}}(\mathcal{I}') \) as \( \lceil 2n' \beta \log((n')^{2+d}) \rceil \), where \( n = |V| \) is the number of variables in \( \mathcal{I} \), \( n' = |V'| \) is the number of variables in \( \mathcal{I}' \) and \( d = O(1) \) is a constant. Then, by combining Lemma 7.2 and Lemma 9.12, we have the result holds.

**Proof of Lemma 9.12.** Consider the algorithm that updates the execution-log CFTP-Log(\( \mathcal{I} \)) \( = \langle v_t, X_t(v_t) \rangle_{t=0}^{T'} \) to CFTP-Log(\( \mathcal{I}' \)) \( = \langle v_t', Y_t(v_t') \rangle_{t=0}^{T'} \), where \( v_t = v_t' \). Since the value of \( T' \) is fixed, we can change the indices and assert that the algorithm updates CFTP-Log(\( \mathcal{I} \)) \( = \langle v_t, X_t(v_t) \rangle_{t=1}^{T'} \) to CFTP-Log(\( \mathcal{I}' \)) \( = \langle v_t, Y_t(v_t) \rangle_{t=1}^{T'} \). The \( \mathcal{X}_0 \) and \( \mathcal{Y}_0 \) are defined in (21).

The coupling of two chains \( \langle \mathcal{X}_t, \mathcal{Y}_t \rangle_{t=0}^{T'} \) is the grand coupling constructed as (24). We bound the expectation of \( R \) in the same way as in (25). Thus
\[
\mathbb{E}[R] \leq \frac{\Delta + 1}{n} \sum_{t=1}^{T'} \mathbb{E}[|D_{t-1}|] + \frac{2T'}{n}.
\]
We claim that for dynamic anti-monotone CFTP, if $\mathcal{I}$ and $\mathcal{I}'$ both satisfy the Condition 9.4 with the same grand coupling $g_\ast(\cdot, \cdot)$ and the same parameters $\beta, C, K$, then it holds that
\[
\forall 0 \leq t \leq T': \quad E[|D_t|] = O\left(\frac{CK}{\beta}\right). \tag{58}
\]

The lemma is proved by combining above two inequalities.

We now prove (58). Recall that for any $0 \leq t \leq T'$, the set $D_t$ is defined as
\[
D_t = \{ v \in V \mid X_t(v) \neq Y_t(v) \}.
\]

Recall, for any $0 \leq t \leq T'$ and any $v \in V$, $X_t(v)$ and $Y_t(v)$ are defined as
\[
X_t(v) = (X_t^{(1)}(v), X_t^{(2)}(v))
\]
\[
Y_t(v) = (Y_t^{(1)}(v), Y_t^{(2)}(v)).
\]

The process $(X_t)_{t=0}^{T'} = (X_t^{(1)}, X_t^{(2)})_{t=0}^{T'}$ is the anti-monotone CFTP on instance $\mathcal{I}$. The process $(Y_t)_{t=0}^{T'} = (Y_t^{(1)}, Y_t^{(2)})_{t=0}^{T'}$ is the anti-monotone CFTP on instance $\mathcal{I}'$. The $X_0$ and $Y_0$ is defined as
\[
X_0 = Y_0 = (X_{\min}, X_{\max}).
\]

where $X_{\min} = X_{\mathcal{I}, \min} = X_{\mathcal{I}', \min}$ and $X_{\max} = X_{\mathcal{I}, \max} = X_{\mathcal{I}', \max}$ are defined in (8).

Recall $\mathcal{I} = (V, E, Q, \Phi)$ is the current instance. Let $\mathcal{J} = \text{Par}(\mathcal{I}) = (V_\mathcal{J}, E_\mathcal{J}, Q, \Phi_\mathcal{J})$, where $V_\mathcal{J} = V_1 \cup V_2$. We denote the graph $(V_1 \cup V_2, E_\mathcal{J})$ as $H$, denote the vertex set $V_1 \cup V_2$ as $V_H$. And we use $G$ to denote the graph $(V, E)$.

Recall $\mathcal{I}' = (V, E', Q, \Phi')$ is the updated instance, which is obtained by Edge-Add, Edge-Delete or Update operations. Let $\mathcal{J}' = \text{Par}(\mathcal{I}') = (V_\mathcal{J}', E_\mathcal{J}', Q, \Phi_\mathcal{J}')$, where $V_\mathcal{J}' = V_1 \cup V_2 = V_\mathcal{J}$. We denote the graph $(V_1 \cup V_2, E_\mathcal{J}')$ as $H'$, denote the vertex set $V_1 \cup V_2$ as $V_H'$. Note that $V_H = V_H'$.

And we use $G'$ to denote the graph $(V, E')$.

Recall that, by the definitions of $\mathcal{J}$ and $\mathcal{J}'$, for any vertex $v \in V$, there are two vertices $v_1 \in V_1$ and $v_2 \in V_2$ that corresponds to vertex $v$.

Let $(X_t^\prime)_{t=0}^{T'}$ be the bipartite Gibbs sampling (Definition 9.6) on instance $\mathcal{J} = \text{Par}(\mathcal{I})$. Let $(Y_t^\prime)_{t=0}^{T'}$ be the bipartite Gibbs sampling (Definition 9.6) on instance $\mathcal{J}' = \text{Par}(\mathcal{I}')$. Define the $X_0'$ and $Y_0'$ as
\[
\forall v_1 \in V_1: \quad X_0'(v_1) = Y_0'(v_1) = c_{\max};
\]
\[
\forall v_2 \in V_2: \quad X_0'(v_2) = Y_0'(v_2) = c_{\min}. \tag{59}
\]

To prove (58), we construct a “grand coupling”, which couples four chains $(X_t, Y_t, X_t^\prime, Y_t^\prime)_{t=0}^{T'}$ as follows: For each $t = 1, 2, \ldots$:

- pick a vertex $v \in V$ u.a.r. and sample a real number $r \in [0, 1]$ u.a.r.;
- the chain $X$ sets $X_t(V \setminus \{v\}) \leftarrow X_{t-1}(V \setminus \{v\})$, then sets
  \[
  X_t^{(1)}(v) \leftarrow g_{\mathcal{I}}(X_{t-1}^{(2)}(\Gamma_\mathcal{I}(v)), r), \quad X_t^{(2)}(v) \leftarrow g_{\mathcal{I}}(X_{t-1}^{(1)}(\Gamma_\mathcal{I}(v)), r);
  \]
- the chain $Y$ sets $Y_t(V \setminus \{v\}) \leftarrow Y_{t-1}(V \setminus \{v\})$, then sets
  \[
  Y_t^{(1)}(v) \leftarrow g_{\mathcal{I}'}(Y_{t-1}^{(2)}(\Gamma_{\mathcal{I}'}(v)), r), \quad Y_t^{(2)}(v) \leftarrow g_{\mathcal{I}'}(Y_{t-1}^{(1)}(\Gamma_{\mathcal{I}'}(v)), r);
  \]
• the chain $X'$ sets $X'_t(V_H \setminus \{v_1, v_2\}) \leftarrow X'_{t-1}(V_H \setminus \{v_1, v_2\})$, then sets
\[
X'_t(v_1) \leftarrow g_{\mathcal{J}_v}(X'_{t-1}(\Gamma_H(v_1)), r), \quad X'_t(v_2) \leftarrow g_{\mathcal{J}_v}(X'_{t-1}(\Gamma_H(v_2)), r).
\] (60)
• the chain $Y'$ sets $Y'_t(V_H \setminus \{v_1, v_2\}) \leftarrow Y'_{t-1}(V_H \setminus \{v_1, v_2\})$, then sets
\[
Y'_t(v_1) = g_{\mathcal{J}_v}(Y'_{t-1}(\Gamma_H(v_1)), r), \quad Y'_t(v_2) = g_{\mathcal{J}_v}(Y'_{t-1}(\Gamma_H(v_2)), r).
\] (61)

By an induction proof (which is similar to the induction in the proof of Lemma 9.11), the following result is easy to verify
\[
\forall 0 \leq t \leq T', v \in V : \quad X_t^{(1)}(v) = X'_t(v_1) \quad \text{and} \quad X_t^{(2)}(v) = X'_t(v_2)
\] (62)
\[
\forall 0 \leq t \leq T', v \in V : \quad Y_t^{(1)}(v) = Y'_t(v_1) \quad \text{and} \quad Y_t^{(2)}(v) = Y'_t(v_2).
\]

Due to the construction of the above coupling $(X_t, Y_t, X'_t, Y'_t)_{t=0}^{T'}$, it is easy to verify that two chains $(X_t)_{t=0}^{T'}$ and $(Y_t)_{t=0}^{T'}$ are coupled in the same way as in the dynamic CFTP algorithm (the coupling in (24)).

Consider the above coupling $(X_t, Y_t, X'_t, Y'_t)_{t=0}^{T'}$. For each $0 \leq t \leq T'$, we define
\[
\mathcal{D}_t = \{u \in V_1 \cup V_2 \mid X'_t(u) \neq Y'_t(u)\}.
\]

Note that $\mathcal{X}$ and $\mathcal{Y}$ are coupled in the same way as in the dynamic CFTP algorithm. Recall the set $\mathcal{D}_t$ is defined as
\[
\mathcal{D}_t = \{v \in V \mid X_t(v) \neq Y_t(v)\} = \{v \in V \mid X_t^{(1)}(v) \neq Y_t^{(1)}(v) \lor X_t^{(2)}(v) \neq Y_t^{(2)}(v)\}.
\]
Then, by (62), we have

\[
\forall 0 \leq t \leq T' : \quad E[|\mathcal{D}_t|] \leq E[|\mathcal{D}'_t|].
\]

Hence, we can prove (58) by bounding $E[|\mathcal{D}'_t|]$.

Recall that $(X'_t)_{t=0}^{T'}$ is bipartite Gibbs sampling (Definition 9.6) on instance $\mathcal{J}$ and $(Y'_t)_{t=0}^{T'}$ is bipartite Gibbs sampling (Definition 9.6) on instance $\mathcal{J}'$. Consider the update from $\mathcal{I}$ to $\mathcal{I}'$.

• If the vertex $v$ is updated (Update$(v, \phi_v)$), then $\mathcal{J}$ and $\mathcal{J}'$ differ only at the constraints on vertices $v_1, v_2$. Define
\[
\mathcal{S}' \triangleq \{v_1, v_2\}.
\]
• If the edge $\{u, v\}$ is updated (Update$(uv, \phi_{uv})$, Edge-Add$(uv, \phi_{uv})$ or Edge-Delete$(uv)$), then $\mathcal{J}$ and $\mathcal{J}'$ differ only at the constraints on edges $\{u_1, v_2\}$, $\{u_2, v_1\}$. Let
\[
\mathcal{S}' \triangleq \{u_1, u_2, v_1, v_2\}.
\]

Note that Update$(a, \phi'_a)$ requires that there is at most one entry between the original and the updated constraints $\phi_a$ and $\phi'_a$, whose value changing between zero and positive. Then it holds that either $\Omega_\mathcal{I} \subseteq \Omega_\mathcal{I}'$ or $\Omega_\mathcal{I}' \subseteq \Omega_\mathcal{I}$. By the construction of $\mathcal{J} = \text{Par}(\mathcal{I})$ and $\mathcal{J}' = \text{Par}(\mathcal{I}')$, if $\Omega_\mathcal{I} \subseteq \Omega_\mathcal{I}'$ then $\Omega_\mathcal{J} \subseteq \Omega_\mathcal{J}'$; if $\Omega_\mathcal{I}' \subseteq \Omega_\mathcal{I}$, then $\Omega_\mathcal{J}' \subseteq \Omega_\mathcal{J}$. Without loss of generality, we assume
\[
\Omega_\mathcal{J}' \subseteq \Omega_\mathcal{J}.
\]
The case $\Omega_\mathcal{J} \subseteq \Omega_\mathcal{J}'$ follows by the symmetry.
Note that $\mathcal{I}$ satisfies Condition 9.4 with the locally-defined potential function $\rho_J$ and the parameters $\beta, C, K$. By Lemma 9.9, then $\mathcal{I}$ satisfies Condition 9.8 with the same locally-defined potential function $\rho_J$ and the same parameters $\beta, C, K$.

Let $(X_t, Y_t)_{t=0}^{T'}$ be the grand coupling of bipartite Gibbs sampling (Definition 9.7) on instance $\mathcal{J} = \text{Par}(\mathcal{I})$. By Condition 9.8, recall $|V_1 \cup V_2| = 2n$, we have for any $1 \leq t \leq T'$,

$$
\forall \sigma, \tau \in \Omega_J : \quad \mathbb{E}[\rho_J(X_t, Y_t) | X_{t-1} = \sigma \land Y_{t-1} = \tau] \leq \left(1 - \frac{\beta}{2n}\right) \cdot \rho_J(\sigma, \tau). \quad (63)
$$

We claim that $(X'_t, Y'_t)_{t=0}^{T'}$ in our coupling satisfies for any $1 \leq t \leq T'$,

$$
\forall \sigma \in \Omega_J, \tau \in \Omega_{J'} \subseteq \Omega_J :
\mathbb{E}[\rho_J(X'_t, Y'_t) | X'_{t-1} = \sigma \land Y'_{t-1} = \tau] \leq \left(1 - \frac{\beta}{2n}\right) \cdot \rho_J(\sigma, \tau) + \frac{8K}{n}. \quad (64)
$$

By the definitions of $X'_0$ and $Y'_0$ in (59), we have $X'_0 = Y'_0$. It is assumed that $\phi_v(c_{\text{max}}) > 0$, $\phi_v(c_{\text{min}}) > 0$ and $\phi_v(c_{\text{min}}, c_{\text{max}}) > 0$ for all constraints $\phi_v(\cdot, \cdot)$ in $\mathcal{I}$ and $\mathcal{I}'$. Then it must hold that $X'_0 \in \Omega_J$ and $Y'_0 \in \Omega_{J'} \subseteq \Omega_J$ due to the definition of the instances $\mathcal{J} = \text{Par}(\mathcal{I})$ and $\mathcal{J}' = \text{Par}(\mathcal{I}')$. It is easy to verify that if $X'_{t-1} \in \Omega_J$, then $X'_t \in \Omega_J$; if $Y'_{t-1} \in \Omega_{J'}$, then $Y'_t \in \Omega_{J'}$. We have

$$
\forall 1 \leq t \leq T' : \quad \mathbb{E}[\rho_J(X'_t, Y'_t)] \leq \left(1 - \frac{\beta}{2n}\right) \mathbb{E}[\rho_J(X'_{t-1}, Y'_{t-1})] + \frac{8K}{n}
$$

$$
\mathbb{E}[\rho_J(X'_0, Y'_0)] = 0.
$$

Note that $|\mathcal{D}'_t| \leq C \rho_J(X'_t, Y'_t)$. Then we have

$$
\forall 0 \leq t \leq T' : \quad \mathbb{E}[|\mathcal{D}_t|] \leq \mathbb{E}[|\mathcal{D}'_t|] \leq C \mathbb{E}[\rho_J(X'_t, Y'_t)] = O \left(\frac{CK}{\beta}\right).
$$

This proves the claim in (58).

We finish the proof by proving the claim in (64). The proof of (64) is very similar to the proof of (28). We include it here for the completeness of the proof.

According to the coupling in Definition 9.7, we can rewrite the expectation in (63) as follows:

$$
\mathbb{E}[\rho_J(X_t, Y_t) | X_{t-1} = \sigma \land Y_{t-1} = \tau] = \frac{1}{n} \sum_{v \in V} \mathbb{E}[\rho_J(\sigma(v_1, v_2) \leftrightarrow (C_{v_1}, C_{v_2}), \tau(v_1, v_2) \leftrightarrow (C'_{v_1}, C'_{v_2}))].
$$

Here $C_{v_1}, C_{v_2}, C'_{v_1}, C'_{v_2} \in Q$ are random spins defined as follows

$$
C_{v_1}^X = g_{\mathcal{J}_1}(\sigma(\Gamma_H(v_1)), r), \quad C_{v_2}^X = g_{\mathcal{J}_2}(\sigma(\Gamma_H(v_2)), r),
$$

$$
C_{v_1}^Y = g_{\mathcal{J}_1}(\tau(\Gamma_H(v_1)), r), \quad C_{v_2}^Y = g_{\mathcal{J}_2}(\tau(\Gamma_H(v_2)), r),
$$

where $r \in [0, 1]$ is uniformly at random. The state $\sigma(v_1, v_2) \leftrightarrow (C_{v_1}, C_{v_2}) \in \Omega_J$ is defined as

$$
\sigma(v_1, v_2) \leftrightarrow (C_{v_1}, C_{v_2})(u) \triangleq \begin{cases} C_{v_1}^X & \text{if } u = v_1, \\ C_{v_2}^X & \text{if } u = v_2, \\ \sigma(u) & \text{if } u \neq v_1 \land u \neq v_2. \end{cases}
$$

and the state $\tau(v_1, v_2) \leftrightarrow (C_{v_1}, C_{v_2}) \in \Omega_J$ is defined in similar way.
Similarly, by (60) and (61), we rewrite the expectation in (64) as follows:

\[
\mathbb{E}[\rho_{\mathcal{J}}(X'_t, Y'_t) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau] = \frac{1}{n} \sum_{v \in \mathcal{V}} \mathbb{E}[\rho_{\mathcal{J}}(\sigma(v_1, v_2) \leftarrow (C_{v_1}^X, C_{v_2}^X), \tau(v_1, v_2) \leftarrow (C_{v_1}^Y, C_{v_2}^Y))].
\]

Here \(C_{v_1}^X, C_{v_2}^X, C_{v_1}^Y, C_{v_2}^Y \in Q\) are random spins defined as follows

\[
\begin{align*}
C_{v_1}^X &= g_{\mathcal{J}_{v_1}}(\sigma(\Gamma_H(v_1)), r), \\
C_{v_2}^X &= g_{\mathcal{J}_{v_2}}(\sigma(\Gamma_H(v_2)), r), \\
C_{v_1}^Y &= g_{\mathcal{J}_{v_1}}(\tau(\Gamma_H(v_1)), r), \\
C_{v_2}^Y &= g_{\mathcal{J}_{v_2}}(\tau(\Gamma_H(v_2)), r),
\end{align*}
\]

where \(r \in [0, 1]\) is uniformly at random.

By the definitions of \(\mathcal{J}\) and \(\mathcal{J}'\), we have

\[
\forall u \in (V_1 \cup V_2) \setminus S': \quad \Gamma_H(u) = \Gamma_H'(u) \text{ and } \mathcal{J}_u = \mathcal{J}'_u.
\]

Hence, we have the following two properties:

- if \(u \notin S'\) and \(v \notin S'\), then
  \[
  \mathbb{E}[\rho_{\mathcal{J}}(\sigma(v_1, v_2) \leftarrow (C_{v_1}^X, C_{v_2}^X), \tau(v_1, v_2) \leftarrow (C_{v_1}^Y, C_{v_2}^Y))] = \mathbb{E}[\rho_{\mathcal{J}}(\sigma(v_1, v_2) \leftarrow (C_{v_1}^X, C_{v_2}^X), \tau(v_1, v_2) \leftarrow (C_{v_1}^Y, C_{v_2}^Y))];
  \]

- if \(u \in S'\) or \(v \in S'\), since \(\rho_{\mathcal{J}}\) is \(K\)-Lipschitz, then
  \[
  \mathbb{E}[\rho_{\mathcal{J}}(\sigma(v_1, v_2) \leftarrow (C_{v_1}^X, C_{v_2}^X), \tau(v_1, v_2) \leftarrow (C_{v_1}^Y, C_{v_2}^Y))] \leq \mathbb{E}[\rho_{\mathcal{J}}(\sigma(v_1, v_2) \leftarrow (C_{v_1}^X, C_{v_2}^X), \tau(v_1, v_2) \leftarrow (C_{v_1}^Y, C_{v_2}^Y)) + 4K.
  \]

Note that \(|S'| \leq 4\). The event \(v_1 \in S' \lor v_2 \in S'\) occurs with probability \(2/n\). Similar to the proof of (28), the claim in (64) can be proved by combining above two properties.

\[\square\]

## 9.3 Bounding chains

For bounding chains, we prove the convergence results for static CFTP and dynamic CFTP under the following condition. This proves Theorem 3.6 for bounding chains.

**Condition 9.13.** Let \(\mathcal{I} = (V, E, Q, \Phi)\) be a MRF instance such that the mixing condition for bounding chains (Condition 5.6) holds for \(\mathcal{I}\) with \(\beta = \Omega(1)\) and \(S = (2^Q \setminus \{\emptyset\})^V\).

### 9.3.1 Static bounding chains

**Lemma 9.14.** Let \(\mathcal{I} = (V, E, Q, \Phi)\) be an MRF instance with \(n = |V|\). Assume that Condition 9.13 holds on instance \(\mathcal{I}\). Then \(\mathbb{E}[T_{\min}] = O\left(\frac{n}{\beta} \log n\right)\) and \(T_{\min} = O\left(\frac{n}{\beta} \log n\right)\) with probability at least \(1 - \frac{1}{n^2}\), where \(T_{\min} = T_{\min}(\mathcal{I})\) is defined in (40). The expected running time of the static bounding chains algorithm on \(\mathcal{I}\) is \(O(n \log n)\).

**Proof.** Consider the bounding chain on \(\mathcal{I}\). The chain starts from the time \(-T\) and \(X_{-T} = Q^V\). Let \(Y_{-T} = X_{-T}\). Let \(\sigma\) be a feasible configuration of \(\mathcal{I}\) and \(Z_{-T}\) satisfies \(Z_{-T}(v) = \{\sigma(v)\}\) for any \(v \in V\). By Condition 9.13, we have

\[
\Pr[\forall t \geq 0 : Y_{-T+t} = Z_{-T+t}] \leq H(Y_{-T+t}, Z_{-T+t}) \leq \left(1 - \frac{\beta}{n}\right)^t H(Y_{-T}, Z_{-T}).
\]
Note that \((\mathcal{Y}_t, \mathcal{Z}_t)_{t \geq T}\) are coupled with the same \(v = (v_t)_{t \geq T}\) and \(r = (r_t)_{t \geq T}\). We also couple \((\mathcal{X}_t, \mathcal{Z}_t)_{t \geq T}\) with the same \(v = (v_t)_{t \geq T}\) and \(r = (r_t)_{t \geq T}\). Thus, we have that the bounding chain \((\mathcal{X}_t)_{t \geq T}\) collapses if \(\mathcal{Y}_0 = \mathcal{Z}_0\).

Let \(D = \max_{I \in \mathcal{G}} H(\sigma, \tau)\). Since \(D = \poly(n)\), we assume \(D \leq n^d\) for some constant \(d = O(1)\). By going through the proof of Lemma 9.2, we have

\[
\Pr \left[ T_{\min} > \left( \frac{n}{\beta} \log(n^{2+d}) \right) \right] \leq \frac{1}{n^2} \quad \text{and} \quad \mathbb{E}[T_{\min}] = O\left( \frac{n}{\beta} \log n \right). \tag{65}
\]

This proves the Lemma. \qed

### 9.3.2 Dynamic bounding chains

To prove Theorem 3.6 on dynamic bounding chains, we need the following lemma to bound the expectation of \(R\) for dynamic bounding chains.

**Lemma 9.15.** Consider the dynamic CFTP for bounding chains. Assume Condition 9.13 holds for both \(\mathcal{I}\) and \(\mathcal{I}'\) with the same parameters \(\beta\). It holds that \(\mathbb{E}[R] = O\left( \frac{\Delta t' \beta}{n^2} \right)\), where \(R\) is defined as (15) by the one-step local coupling for dynamic instances constructed as (24), \(n = |V|\) and \(\Delta\) denotes the maximum degree of graph \(G = (V, E)\).

We first prove Theorem 3.6 on bounding chains, then prove Lemma 9.15.

**Proof of Theorem 3.6 on bounding chains.** For dynamic bounding chains, by (65), we can set \(T_{\text{couple}}(\mathcal{I}) \geq \left( \frac{n}{\beta} \log(n^{2+d}) \right)\) and set \(T_{\text{couple}}(\mathcal{I}') \geq \left( \frac{n'}{\beta} \log((n')^{2+d}) \right)\), where \(n = |V|\) is the number of variables in \(\mathcal{I}\), \(n' = |V'|\) is the number of variables in \(\mathcal{I}'\) and \(d = O(1)\) is a constant. Then, by combining Lemma 7.2 and Lemma 9.15, we have the result holds. \qed

**Proof of Lemma 9.15.** Consider the algorithm that updates the execution-log CFTP-Log\((\mathcal{I}) = \langle v_t, \mathcal{X}_t(v_t) \rangle_{t = -T' + 1}^0 \) to CFTP-Log\((\mathcal{I}') = \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t = -T' + 1}^0\). Since the value of \(T'\) is fixed, we can change the indices and assume the algorithm updates the log CFTP-Log\((\mathcal{I}) = \langle v_t, \mathcal{X}_t(v_t) \rangle_{t = 1}^{T'}\) to CFTP-Log\((\mathcal{I}') = \langle v'_t, \mathcal{Y}_t(v'_t) \rangle_{t = 1}^{T'}\). The initial \(\mathcal{X}_0\) and \(\mathcal{Y}_0\) are defined in (21).

We bound the expectation of \(R\) in the same way as in (25). Thus

\[
\mathbb{E}[R] \leq \frac{(\Delta + 1)}{n} \sum_{t=1}^{T'} \mathbb{E}[|D_{t-1}|] + \frac{2T'}{n}.
\]

We claim that for dynamic bounding chains, if \(\mathcal{I}\) and \(\mathcal{I}'\) both satisfy the Condition 9.13 with the same parameters \(\beta\), then it holds that

\[
\forall 0 \leq t \leq T' : \quad \mathbb{E}[|D_t|] = O\left( \frac{1}{\beta} \right). \tag{66}
\]

The Lemma is proved by combining above two inequalities.

We now prove (66). Let \((\mathcal{X}_t, \mathcal{Y}_t)_{t \geq 0}\) be the grand coupling for dynamic instance (defined in (24)), where \((\mathcal{X}_t)_{t \geq 0}\) is the bound chain on \(\mathcal{I}\) and \((\mathcal{Y}_t)_{t \geq 0}\) is the bound chain on \(\mathcal{I}'\). We claim the following result

\[
\forall \sigma, \tau \in (2^Q \setminus \emptyset)^V : \quad \mathbb{E}[H(\mathcal{X}_t, \mathcal{Y}_t) \mid \mathcal{X}_{t-1} = \sigma \land \mathcal{Y}_{t-1} = \tau] \leq \left(1 - \frac{\beta}{n}\right) \cdot H(\sigma, \tau) + \frac{4}{n}; \tag{67}
\]
Assume (67) holds. Taking expectation over $X_{t-1}$ and $Y_{t-1}$, we have
\[ \mathbb{E} [ H(X_t, Y_t) ] \leq \left( 1 - \frac{\beta}{n} \right) \mathbb{E} [ H(X_{t-1}, Y_{t-1}) ] + \frac{4}{n}. \] (68)

Note that
\[ H(X_0, Y_0) = 0, \] (69)

because $X_0 = Q^V = Y_0$. Combining (68) and (69) implies
\[ \forall 0 \leq t \leq T' : \quad \mathbb{E} [ H(X_t, Y_t) ] \leq \frac{4}{\beta} + \frac{4}{n} < \frac{5}{\beta}. \]

Thus, we have
\[ \forall 0 \leq t \leq T' : \quad \mathbb{E} [ D_t ] = \mathbb{E} [ H(X_t, Y_t) ] = O \left( \frac{1}{\beta} \right). \]

This proves the claim in (66).

We finish the proof by proving the claim in (67). The proof of (67) is very similar to the proof of (28). We include it here for the completeness of the proof.

Assume $I$ satisfies Condition 9.13 with the coupling $(X'_t, Y'_t)_{t \geq 0}$, where $(X'_t, Y'_t)_{t \geq 0}$ are coupled with the same $v = (v_t)_{t \geq 0}$ and $r = (r_t)_{t \geq 0}$. Thus, we have
\[ \forall \sigma, \tau \in (2^Q \setminus \{ \emptyset \})^V : \quad \mathbb{E} \left[ H(X'_t, Y'_t) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau \right] \leq \left( 1 - \frac{\beta}{n} \right) \cdot H(\sigma, \tau); \] (70)

According to the coupling, for any given $r_t$, we can rewrite the expectation in (70) as follows:
\[ \mathbb{E} \left[ H(X'_t, Y'_t) \mid X'_{t-1} = \sigma \land Y'_{t-1} = \tau \right] = \frac{1}{n} \sum_{v \in V} \mathbb{E} \left[ H(\sigma^{v \leftarrow C'_v}, \tau^{v \leftarrow C'_v'}) \right], \] (71)

where $C'_v \leftarrow \left\{ g_{I_v}(\eta, r_t) \mid \eta \in \bigotimes_{u \in E_G(v)} \sigma(u) \right\}$, $C''_v \leftarrow \left\{ g_{I_v}(\eta, r_t) \mid \eta \in \bigotimes_{u \in E_G(v)} \tau(u) \right\}$ and the configuration $\sigma^{v \leftarrow C'_v} \in (2^Q)^V$ is defined as
\[ \sigma^{v \leftarrow C'_v} (u) = \begin{cases} C'_v & \text{if } u = v \\ \sigma(u) & \text{if } u \neq v \end{cases}. \]

and the configuration $\tau^{v \leftarrow C''_v} \in (2^Q)^V$ is defined in similar way.

Similarly, because $(X'_t, Y'_t)_{t \geq 0}$ is coupled by the grand coupling, we can rewrite the expectation in (67) for any given $r_t$ as follows:
\[ \mathbb{E} \left[ H(X_t, Y_t) \mid X_{t-1} = \sigma \land Y_{t-1} = \tau \right] = \frac{1}{n} \sum_{v \in V} \mathbb{E} \left[ H(\sigma^{v \leftarrow C'_v}, \tau^{v \leftarrow C''_v}) \right], \] (72)

where $C'_v \leftarrow \left\{ g_{I_v}(\eta, r_t) \mid \eta \in \bigotimes_{u \in E_G(v)} \sigma(u) \right\}$, $C''_v \leftarrow \left\{ g_{I_v}(\eta, r_t) \mid \eta \in \bigotimes_{u \in E_G(v)} \tau(u) \right\}$

The following two properties hold for (71) and (72).

- If $v \notin S$, for any given $r_t$, it holds that $C'_v = C'_v$, $C''_v = C''_v$. Hence
  \[ \forall v \notin S : \quad H \left( \sigma^{v \leftarrow C'_v}, \tau^{v \leftarrow C''_v} \right) = H \left( \sigma^{v \leftarrow C'_v}, \tau^{v \leftarrow C''_v} \right). \]

- If $v \in S$, then it holds that $H(\sigma^{v \leftarrow C'_v}, \sigma^{v \leftarrow C'_v}) \leq 1$ and $H(\tau^{v \leftarrow C''_v}, \tau^{v \leftarrow C''_v}) \leq 1$. Then,
  \[ \forall v \in S : \quad H \left( \sigma^{v \leftarrow C'_v}, \tau^{v \leftarrow C''_v} \right) \leq H \left( \sigma^{v \leftarrow C'_v}, \tau^{v \leftarrow C''_v} \right) + 2. \]

Note that $|S| \leq 2$. Similar to the proof of (28), the claim in (67) can be proved by combining above two properties. \(\square\)
9.4 Applications

In this section, we apply the conditions of Theorem 3.6 on specific models and give the corresponding dynamic exact sampling results.

**Corollary 9.16.** There exist dynamic perfect sampling algorithms as stated in Theorem 3.6 for the following models:

- Ising model with temperature $\beta$ and arbitrary local fields where $\exp(-2|\beta|) \geq 1 - \frac{2-\delta}{\Delta+1}$, and with space cost $O(n \log n)$ and expected time cost $O(2^2 \cdot \log^2 n \cdot \log \log n)$ for each update;
- hardcore model with fugacity $\lambda \leq \frac{2-\delta}{\Delta+2}$, and with space cost $O(n \log n)$ and expected time cost $O(\Delta^2 \cdot \log^2 n \cdot \log \log n)$ for each update;
- proper $q$-coloring with $q \geq (2+\delta)\Delta^2 + 3\Delta$, and with space cost $O(\Delta n \log n)$ and expected time cost $O(\Delta^3 \cdot \log^2 n \cdot \log \log n)$ for each update;

where $n = |V|$, $\delta > 0$ is a constant and $\Delta = \max\{\Delta_G, \Delta_{G'}\}$ where $\Delta_G$ denotes the maximum degree of the input graph, and $\Delta_{G'}$ denotes the maximum degree of the updated graph.

**Remark 9.17.** For proper $q$-coloring, the space cost and expected time cost here are different from that mentioned in Theorem 3.6. This is caused by the following reason. We assume $q = O(1)$ in Theorem 3.6 but for proper $q$-coloring in Corollary 9.16, $q = \Omega(\Delta^2)$.

The ferromagnet Ising model is a typical monotone system. For Ising model with temperature $\beta$ and arbitrary local fields such that $\exp(-|\beta|) \geq 1 - \frac{2-\delta}{\Delta+1}$, the Dobrushin-Shlosman condition (Condition 5.3) is satisfied. If an MRF instance $\mathcal{I}$ satisfies the Dobrushin-Shlosman condition, then the step-wise decay property in Condition 3.1 holds for one-step optimal coupling with $p_{\mathcal{I}}(\cdot, \cdot) = H(\cdot, \cdot)$, where $H(\cdot, \cdot)$ is the Hamming distance. This implies Condition 3.1 holds for instance $\mathcal{I}$ with parameters $\beta = \Omega(1)$, $C = K = 1$. Because for Gibbs sampling on Ising model, the grand coupling in [26] is precisely the one-step optimal coupling, we have the condition in Theorem 3.6 for monotone systems holds for Ising model.

The hard-core model is a typical anti-monotone system. For hard core model $\mathcal{I}$ with fugacity $\lambda \leq \frac{2-\delta}{\Delta+2}$, Lemma 8.2 shows that it satisfies Condition 3.1 with the one-step optimal coupling and the parameters $\beta = \Omega(1)$, $C = O(1)$, $K = O(\Delta)$. It is easy to verify that, for Gibbs sampling on hard core model, the grand coupling in [30] is precisely the one-step optimal coupling. Thus, it is easy to verify that $\mathcal{J}$ is also with fugacity $\lambda \leq \frac{2-\delta}{\Delta+2}$ and satisfies Condition 3.1 with the grand coupling and the parameters $\beta = \Omega(1)$, $C = O(1)$, $K = O(\Delta)$. Meanwhile, the potential function used in Lemma 8.2 is obviously locally-defined. Thus, the condition in Theorem 3.6 for anti-monotone systems holds for hard core model.

Thus, we have the first two results of Corollary 9.16 hold as a consequence of Theorem 3.6. In the following, we prove the third result.

Our dynamic algorithm for sampling graph coloring is based on the bounding chain with CFTP for coloring in [17], which calls Algorithm 9 in each round. To simplify the analysis, we give Algorithm 10, a modified version of Algorithm 9. If more than $\Delta$ colors have been picked and added to $\mathcal{R}$ in Line 5 of Algorithm 10, then there must be a feasible color of $v_t$ in $\mathcal{R}$ for any $\sigma \in \bigotimes_{v \in \Gamma_G(v_t)} X_{t-1}(v)$. Thus, the correctness of Algorithm 10 follows. Algorithm 11 is an equivalent implement of Algorithm 10. Our analysis concentrates on Algorithm 11.

We first prove the following two lemmas to bound the expected running time of the static bounding chain for coloring and the expectation of $R$ for the dynamic bounding chain for coloring.

**Lemma 9.18.** Let $\mathcal{I}$ be a coloring model on graph $G = (V, E)$ with $n = |V|$ and color set $Q$. If $|Q| \geq 2\Delta^2 + 3\Delta$ where $\Delta$ denotes the maximum degree of graph $G$, then $\mathbb{E}[T_{\min}] = O(n \log n)$ and $T_{\min} = O(n \log n)$ with probability at least $1 - \frac{1}{n^2}$, where $T_{\min} = T_{\min}(\mathcal{I})$ is defined in (40).
Algorithm 9: round $t$ of bounding chain for coloring

1. $X_t \leftarrow X_{t-1}$;
2. Choose $v_t \in V$, let $X_t(v_t) \leftarrow \emptyset$;
3. repeat
   4. Choose $c \in U \{1, \cdots, q\}$;
   5. if no neighbor $w$ of $v_t$ has $X_{t-1}(w) = \{c\}$ then
      6. Let $X_t(v_t) \leftarrow X_{t-1}(v_t) \cup \{c\}$
3. until $c \not\in \bigcup_{w \in \Gamma(v_t)} X_{t-1}(w)$ or $|X_t(v_t)| > \Delta$

Algorithm 10: round $t$ of modified bounding chain for coloring

1. $X_t \leftarrow X_{t-1}$;
2. Choose $v_t \in V$, let $X_t(v_t) \leftarrow \emptyset$, $R \leftarrow \emptyset$;
3. repeat
   4. Choose $c \in U \{1, \cdots, q\} \setminus R$;
   5. Let $R \leftarrow R \cup \{c\}$;
   6. if no neighbor $w$ of $v_t$ has $X_{t-1}(w) = \{c\}$ then
      7. Let $X_t(v_t) \leftarrow X_{t-1}(v_t) \cup \{c\}$
3. until $c \not\in \bigcup_{w \in \Gamma(v_t)} X_{t}(w)$ or $|R| > \Delta$

Lemma 9.19. Consider the dynamic bounding chain for coloring. Under the condition of Corollary 9.16 on coloring, we have $E[R] = O\left(\frac{\Delta T^*}{n\delta}\right)$, where $R$ is defined as (15) by the one-step local coupling for dynamic instances constructed as (24), $n = |V|$ and $\Delta$ denotes the maximum degree of graph $G = (V, E)$.

Similar to Lemma 7.2, we also have the following lemma.

Lemma 9.20. Let $\mathcal{I}$ be a coloring instance on graph $G = (V, E)$ with color set $Q$. Let $\mathcal{I}'$ be the updated instance on graph $G' = (V', E')$. Consider the bounding chain algorithm for coloring calling Algorithm 11. Denote $T = T_{\text{couple}}(\mathcal{I})$, $T' = T_{\text{couple}}(\mathcal{I}')$ and $T_{\text{max}} = \max\{T, T'\}$, where $T_{\text{couple}}()$ is defined in (23). Assume $T, T' \in \Omega(\log n)$. Then, there exists a dynamic bounding chain which does the followings:

- **(space cost)** The dynamic bounding chain maintains an explicit copy of a sample $X \in Q^V$ for the current instance $\mathcal{I}$, and also a data structure using $O(T)$ memory words, each of $O(\log T + \Delta \log |Q|)$ bits, for representing an execution-log $\text{CFTP-Log}(\mathcal{I}) = \langle v_t, X_t(v_t) \rangle_{t=-T+1}^0$ for the bounding chain $(\mathcal{X}_t)_{t=-T+1}^0$ on $\mathcal{I}$. The sample $X \sim \mu_{\mathcal{I}}$ is a perfect sample for $\mathcal{I}$.
- **(correctness)** Assuming that Condition 7.1 holds for $\text{CFTP-Log}(\mathcal{I})$ for the bounding chain on $\mathcal{I}$, upon each update that modifies $\mathcal{I}$ to $\mathcal{I}'$, the algorithm updates the $\text{CFTP-Log}(\mathcal{I})$ represented by the data structure to $\text{CFTP-Log}(\mathcal{I}') = \langle v'_t, Y_t(v'_t) \rangle_{t=-T'+1}^0$ for the bounding chain on $\mathcal{I}'$, where $\text{CFTP-Log}(\mathcal{I}')$ satisfies Condition 7.1 on $\mathcal{I}'$, and the algorithm also has an explicit copy of a sample $Y \in Q^{V'}$ such that $Y \sim \mu_{\mathcal{I}'}$.
- **(time cost)** Assuming Condition 7.1 for $\text{CFTP-Log}(\mathcal{I})$ for the bounding chain on $\mathcal{I}$, the expected time complexity for resolving an update is:

$$O\left(\left(\left|T - T'\right| + \frac{T_{\text{max}}}{n} + E[R]\right) \cdot \left(\Delta \log T_{\text{max}} \cdot \log \frac{T_{\text{max}}}{n} + \Delta^2\right)\right),$$
Algorithm 11: round \( t \) of modified bounding chain for coloring

\[
\begin{align*}
1 & \quad \mathcal{X}_t \leftarrow \mathcal{X}_{t-1}; \\
2 & \quad \text{Choose } v_t \in U \ V, \text{ let } \mathcal{X}_t(v_t) \leftarrow \emptyset, \mathcal{R} \leftarrow \emptyset, i \leftarrow 1; \\
3 & \quad \textbf{repeat} \\
4 & \quad \quad \text{Choose } c_i \in U \ \{1, \cdots, q\} \ \setminus \mathcal{R}; \\
5 & \quad \quad i \leftarrow i + 1; \\
6 & \quad \quad \text{Let } \mathcal{R} \leftarrow \mathcal{R} \cup \{c_i\}; \\
7 & \quad \quad \textbf{until } |\mathcal{R}| > \Delta; \\
8 & \quad i \leftarrow 1; \\
9 & \quad \textbf{repeat} \\
10 & \quad \quad \text{if no neighbor } w \text{ of } v_t \text{ has } \mathcal{X}_{t-1}(w) = \{c_i\} \text{ then} \\
11 & \quad \quad \quad \text{Let } \mathcal{X}_t(v_t) \leftarrow \mathcal{X}_{t-1}(v_t) \cup \{c_i\}; \\
12 & \quad \quad i \leftarrow i + 1; \\
13 & \quad \quad \textbf{until } c_i \not\in \bigcup_{w \in I(v_t)} \mathcal{X}_t(w) \text{ or } i > \Delta;
\end{align*}
\]

where \( \Delta \) is the maximum degree of graph \( G = (V, E) \) and \( \mathcal{R} \) is defined in (15) with the coupling constructed in (24).

**Proof of Corollary 9.16 on coloring.** The result of Corollary 9.16 on coloring is immediate by combining lemmas 9.18, 9.19 and 9.20. \( \square \)

Lemma 9.18 is immediate by following the analysis of Algorithm 9 in [17]. In the following, we give the proof of Lemma 9.19.

**Proof of Lemma 9.19.** As in Lemma 9.15, consider the updates \text{Edge-Add, Edge-Delete, Update}. Then, we have \( V = V' \). Let \((\mathcal{X}_t, \mathcal{Y}_t)_{t \geq -T'}\) be the grand coupling for dynamic instance (defined in (24)) where \( \mathcal{X}_{t-T} \) is on \( I \), \( \mathcal{Y}_{t-T} \) is on \( I' \) and \( \mathcal{X}_{-T} = \mathcal{Y}_{-T} = Q^V \). We claim the following result

\[
\forall \sigma, \tau \in (2^Q \setminus \{\emptyset\})^V : \quad \mathbb{E}[H(\mathcal{X}_t, \mathcal{Y}_t) \mid \mathcal{X}_{t-1} = \sigma \land \mathcal{Y}_{t-1} = \tau] \leq \left(1 - \frac{\beta}{n}\right) \cdot H(\sigma, \tau) + \frac{4}{n}; \quad (73)
\]

Assume (73) holds. Then, by following the proofs in Section 9.3, it is easy to verify this lemma.

In the following, we finish the proof by proving the claim in (73).

It is easy to verify that the follow properties holds for Algorithm 11.

1. For any \( t \geq -T \) and \( w \in V \), either \( \mathcal{X}_t(w) = Q \) or \( |\mathcal{X}_t(w)| \leq \Delta + 1 \).

2. Because \((\mathcal{X}_t, \mathcal{Y}_t)_{t \geq -T'}\) are coupled with grand coupling, the generated colors and set \( \mathcal{R} \) are public for these two chains. Thus, we have \( |\mathcal{X}_t(v_t) \cup \mathcal{Y}_t(v_t)| \leq |\mathcal{R}| \leq \Delta + 1 \).

3. For any \( w \neq v_t \) where \( \mathcal{X}_t(w) \neq \mathcal{Y}_t(w) \), the Markov chain in Algorithm 11 must has transitioned on \( w \). Then by the last property, we have \( |\mathcal{X}_t(w) \cup \mathcal{Y}_t(w)| \leq |\mathcal{R}| \leq \Delta + 1 \).

Recall that

\[
\mathcal{S} \triangleq \begin{cases} 
\{v\} & \text{update is } \text{Update}(v, \phi_v) \text{ for some vertex } v \in V, \\
\{u, v\} & \text{update is } \text{Edge-Add}(e, \phi_e), \text{Edge-Delete}(e), \text{ or } \text{Update}(e, \phi_e) \text{ where } e = \{u, v\}.
\end{cases}
\]

In the following, we discuss two different cases.
• \( v_t \notin S \) and \( \mathcal{X}_{i-1}(w) \neq Q \) for all \( w \in \Gamma(v_t) \). Recall that \( \mathcal{D}_t \triangleq \{ v \in V \mid \mathcal{X}_i(v) \neq \mathcal{Y}_i(v) \} \). Let

\[
\mathcal{T}_1(v_t) \triangleq \bigcup_{w \in \Gamma(v_t) \cap \mathcal{D}_{i-1}} (\mathcal{X}_{i-1}(w) \cup \mathcal{Y}_{i-1}(w)).
\]

(74)

Then, by property 3 we have

\[
|\mathcal{T}_1(v_t)| \leq (\Delta + 1)|\Gamma(v_t) \cap \mathcal{D}_{i-1}|.
\]

(75)

Let

\[
\mathcal{T}_2 \triangleq \left( \bigcup_{w \in \Gamma(v_t)} \mathcal{X}_{i-1}(w) \right) \cap \left( \bigcup_{w \in \Gamma(v_t)} \mathcal{Y}_{i-1}(w) \right).
\]

(76)

By property 1, we have

\[
|\mathcal{T}_2| \leq \left| \bigcup_{w \in \Gamma(v_t)} \mathcal{X}_{i-1}(w) \right| \leq \Delta(\Delta + 1).
\]

(77)

We declare that \( v_t \in \mathcal{D}_t \) only if there exists some \( c_i \) such that \( c_1 \in \mathcal{T}_2, \ldots, c_{i-1} \in \mathcal{T}_2 \) and \( c_i \in \mathcal{T}_1 \). Otherwise, we have there exists some \( c_j \) such that \( c_1 \notin \mathcal{T}_1, \ldots, c_{j-1} \notin \mathcal{T}_1 \) and \( c_j \notin \mathcal{T}_1 \cup \mathcal{T}_2 \). Then, we have \( \mathcal{X}_i(v_t) = \mathcal{Y}_i(v_t) \) and \( v_t \notin \mathcal{D}_t \). This is because

- \( c_k \notin \mathcal{T}_1 \) for any \( k < j \). Then we have either \( c_k \notin \bigcup_{w \in \Gamma(v_t)} (\mathcal{X}_i(w) \cup \mathcal{Y}_i(w)) \), or \( c_k \notin \bigcup_{w \in \Gamma(v_t)} (\mathcal{X}_i(w) \cup \mathcal{Y}_i(w)) \). By induction, it can be proved that either \( c_k \) is added to both \( \mathcal{X}_i(v_t) \) and \( \mathcal{Y}_i(v_t) \), or \( c_k \) is not added to \( \mathcal{X}_i(v_t) \) or \( \mathcal{Y}_i(v_t) \) in Algorithm 11.

- \( c_j \notin \mathcal{T}_1 \) and \( c_j \notin \mathcal{T}_2 \). We have \( c_j \notin \bigcup_{w \in \Gamma(v_t)} (\mathcal{X}_i(w) \cup \mathcal{Y}_i(w)) \). Then, either \( c_j \) is added to both \( \mathcal{X}_i(v_t) \) and \( \mathcal{Y}_i(v_t) \), or \( c_j \) is not added to \( \mathcal{X}_i(v_t) \) or \( \mathcal{Y}_i(v_t) \) in Algorithm 11. And the following \( c_{j+1}, c_{j+2}, \ldots \) are not added to \( \mathcal{X}_i(v_t) \) and \( \mathcal{Y}_i(v_t) \).

Note that

\[
\Pr(c_1, \ldots, c_{i-1} \in \mathcal{T}_2, c_i \in \mathcal{T}_1) \leq \frac{|\mathcal{T}_1|}{|\mathcal{T}_2|} \left( \frac{|\mathcal{T}_2|}{|\mathcal{T}_1|} \right)^{i-1} = \frac{|\mathcal{T}_1|}{q \Delta} \left( \frac{|\mathcal{T}_2|}{q \Delta} \right)^{i-1}.
\]

Thus, we have

\[
\Pr(v_t \in \mathcal{D}_t) \leq \sum_{i=1}^{\Delta+1} \frac{|\mathcal{T}_1|}{q \Delta} \left( \frac{|\mathcal{T}_2|}{q \Delta} \right)^{i-1}
\]

\[
\leq \sum_{i=0}^{\Delta} \frac{(\Delta + 1)|\Gamma(v_t) \cap \mathcal{D}_{i-1}|}{q \Delta} \left( \frac{(\Delta + 1) \Delta}{q \Delta} \right)^i.
\]

Thus, if \( q \geq (2 + \delta)\Delta^2 + 3\Delta \), we have \( q \geq (1 + \frac{\delta}{4})(2\Delta^2 + 2\Delta) + \Delta \). Let \( \delta' = \frac{\delta}{4} \). Then, we have

\[
\Pr(v_t \in \mathcal{D}_t) \leq \sum_{i=0}^{\Delta} \frac{(\Delta + 1)|\Gamma(v_t) \cap \mathcal{D}_{i-1}|}{q - \Delta} \left( \frac{(\Delta + 1) \Delta}{q - \Delta} \right)^i
\]

\[
\leq \sum_{i=0}^{\Delta} \frac{|\Gamma(v_t) \cap \mathcal{D}_{i-1}|}{2(1 + \delta') \Delta} \frac{1}{\Delta}
\]

\[
\leq \frac{|\Gamma(v_t) \cap \mathcal{D}_{i-1}|}{(1 + \delta') \Delta}.
\]

(78)
• $v_t \not\in S$ and there is a $u \in \Gamma(v_t)$ where $\mathcal{X}_t(u) = Q$. Then, the Markov chain in Algorithm 11 has not transitioned on $w$. Thus, we have $\mathcal{Y}_t(u) = Q$. In this case, we have $v_t \in D_t$ only if there exists some $c_i \in T_3$, where $T_3$ is defined as

$$T_3(v_t) \triangleq \bigcup_{w \in \Gamma(v_t) \cap D_{t-1}} \{ c : \mathcal{X}_i(w) = \{ c \} \vee \mathcal{Y}_i(w) = \{ c \} \}.$$ 

Because for any $c \not\in T_3(v_t)$, $c$ will be added to both $\mathcal{X}_t(v_t)$ and $\mathcal{Y}_t(v_t)$ or not added to $\mathcal{X}_t(v_t)$ or $\mathcal{Y}_t(v_t)$ under the condition $\mathcal{X}_i(u) = \mathcal{Y}_i(u) = Q$. Thus, by union bound, we have

$$\Pr(v_t \in D_t) \leq \sum_{i=1}^{\Delta+1} \Pr(c_i \not\in T_3(v_t)) \leq \frac{(\Delta + 1)|T_3(v_t)|}{q - \Delta}. \quad (79)$$

Note that

$$|T_3(v_t)| \leq 2|\Gamma(v_t) \cap D_{t-1}|. \quad (80)$$

By (79) and (80), we have

$$\Pr(v_t \in D_t) \leq \frac{(\Delta + 1)|T_3(v_t)|}{q - \Delta} \leq \frac{2(\Delta + 1)|\Gamma(v_t) \cap D_{t-1}|}{q - \Delta} \leq \frac{|\Gamma(v_t) \cap D_{t-1}|}{(1 + \delta')\Delta}. \quad (81)$$

The last inequality is by $q \geq (1 + \delta')(2\Delta^2 + 2\Delta) + \Delta$.

Combining (78) with (81), we have

$$\Pr(v_t \in D_t) = \frac{1}{n} \sum_{v \in V} \Pr(v_t \in D_t | v_t = v)$$

$$= \frac{1}{n} \sum_{v \in S} \Pr(v_t \in D_t | v_t = v) + \frac{1}{n} \sum_{v \in S} \Pr(v_t \in D_t | v_t = v)$$

$$\leq \frac{2}{n} \sum_{v \in S} \frac{|\Gamma(v) \cap D_{t-1}|}{(1 + \delta')\Delta}$$

$$\leq \frac{2}{n} \sum_{v \in S} \frac{|D_{t-1}|}{(1 + \delta')n}.$$ 

The last inequality is by $\sum_v |\Gamma(v) \cap D_{t-1}| \leq \Delta|D_{t-1}|$.

Thus, we have

$$\mathbb{E}[|D_t|] = |D_{t-1}| + \mathbb{E}[\mathbf{1} [v_t \in D_t]] - \mathbb{E}[\mathbf{1} [v_t \in D_{t-1}]]$$

$$= |D_{t-1}| + \Pr(v_t \in D_t) - \frac{|D_{t-1}|}{n}$$

$$\leq \frac{2}{n} + |D_{t-1}||1 - \frac{1}{n} + \frac{1}{(1 + \delta')n}|$$

$$\leq \frac{2}{n} + |D_{t-1}||1 - \frac{\delta'}{(1 + \delta')n}|.$$ 

Note that $|D_t| = H(\mathcal{X}_t, \mathcal{Y}_t)$, $|D_{t-1}| = H(\mathcal{X}_{t-1}, \mathcal{Y}_{t-1})$ and our proof holds for any $\sigma, \tau$ under the condition $\mathcal{X}_{t-1} = \sigma \wedge \mathcal{Y}_{t-1} = \tau$. Let $\beta = \frac{\delta'}{1 + \delta'}$, we have (73) holds.

At last, we give the proof of Lemma 9.20.

Proof of Lemma 9.20. By following the proof of Lemma 7.2, the result on correctness is obvious. Note that $(\Delta + 1) \log |Q|$ bits of each word is for representing the $\Delta + 1$ colors of each $\mathcal{X}_t(v_t)$. And with an auxiliary array of size $|Q|$ for hashing, in each round $t$ Algorithm 11 can be finished with $O(\Delta^2)$ time. Then, by following the proof of Lemma 6.6, the results on space cost and time cost are also immediate.
10 Conclusion and Open Problems

This paper studies dynamic sampling algorithms that can maintain a (approximate or perfect) sample with low incremental cost, when the underlying graphical model is dynamically changing.

The discoveries of the paper basically show that the step-wise decay of single-site coupling, a primary tool for rapid mixing of Markov chains and the corresponding near-linear running time of MCMC sampling in the static setting, can be used to imply efficient dynamic sampling algorithms with near-linear space cost and poly-logarithmic incremental time cost. Although some restrictions need to be imposed on the metric in which the coupling is decaying, such restrictions are mostly satisfied implicitly in the existing analysis of coupling in the literature.

Therefore, an important message sent by the paper is: the coupling of single-site Markov chains, which have been widely used in supporting efficient MCMC sampling in the classic static setting, can also imply efficient dynamic MCMC sampling algorithms.

Two major concrete open problems for dynamic sampling are to sample: (1) matchings in dynamically changing graphs; and (2) from dynamically changing hardcore models up to the uniqueness threshold. Dynamical samplers with poly-logarithmic incremental costs for these problem on graphs with bounded maximum degrees would improve the state-of-the-arts for the respective problems in the static setting, in particular: (1) for sampling matchings, it would improve the time bound of the classic Jerrum-Sinclair chain \[20\] on graphs with bounded maximum degree; and (2) for the hardcore model, it would improve the best known time upper bounds for sampling from the hardcore models \[31, 8, 30\].

The above open problems arise from a more profound background: the general relation between dynamic sampling and spatial mixing properties \[32, 31\]. It is natural to foresee that some decay of correlation properties must have played some roles to affect how much a new sample \(Y \sim \mu_{\mathcal{I}'}\) has to deviate from the old sample \(X \sim \mu_\mathcal{I}\) where \(\mathcal{I}'\) differs from \(\mathcal{I}\) by local updates. Indeed, the result of this paper shows that the Dobrushin-Shlosman condition for the decay of correlation is sufficient to imply such small discrepancy between the samples before and after the local update to the graphical model, by explicitly giving a dynamic sampling algorithm. It is then a fundamental open problem to properly characterize the spatial mixing condition such that the samples from the graphical models that differ locally can always be coupled with small discrepancy, and furthermore such existence of coupling with small discrepancy can be transformed to dynamic sampling algorithms.

In this paper we give systematic approaches for transforming classic MCMC sampling algorithms to dynamic sampling algorithms with near-linear space costs and sub-linear incremental time costs for each update. For practical reasons, it is much more desirable to have both space and time costs sub-linear in the size of the input.\(^6\) In this aspect, the dynamic perfect sampler in \[9\] performs better, but only in more restrictive regimes. It is then an important open problem to obtain dynamic sampling algorithms with sub-linear space and time costs in the current or even broader regimes, or to prove information-theoretical lower bounds showing that this cannot be achieved by any dynamic data structures. Such lower bounds would separate dynamic sampling (with sub-linear time and space costs) from static sampling.

In a high level, this line of researches on dynamic sampling may extend the horizon of classic topics in computer science such as dynamic data structures and dynamic graphs to the world of sampling. Unlike the classic scenarios, where usually the answers (e.g. connectivity) or solutions (e.g. MST) are dynamically maintained, here what we dynamically maintain has to be a “typical” solution, or more properly phrased, the statistical information of the solution space, which is important for contemporary applications, in e.g. machine learning.

\(^6\)By space cost, we mean the extra space used by the algorithm in addition to storing the sample itself.
References

[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2009.

[3] RL Dobrushin and SB Shlosman. Completely analytical interactions: constructive description. *Journal of Statistical Physics*, 46(5-6):983–1014, 1987.

[4] Roland L Dobrushin and Senya B Shlosman. Completely analytical gibbs fields. In *Statistical physics and dynamical systems*, pages 371–403. Springer, 1985.

[5] Roland Lvovich Dobrushin and Senya B Shlosman. Constructive criterion for the uniqueness of gibbs field. In *Statistical physics and dynamical systems*, pages 347–370. Springer, 1985.

[6] Martin Dyer, Leslie Ann Goldberg, and Mark Jerrum. Dobrushin conditions and systematic scan. *Combinatorics, Probability and Computing*, 17(6):761–779, 2008.

[7] Martin Dyer and Catherine Greenhill. On markov chains for independent sets. *Journal of Algorithms*, 35(1):17–49, 2000.

[8] Charilaos Efthymiou, Thomas P Hayes, Daniel Štefankovic, Eric Vigoda, and Yitong Yin. Convergence of MCMC and loopy BP in the tree uniqueness region for the hard-core model. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 704–713, 2016.

[9] Weiming Feng, Nisheeth Vishnoi, and Yitong Yin. Dynamic sampling from graphical models. To appear in STOC’19.

[10] Andreas Galanis, Daniel Štefanković, and Eric Vigoda. Inapproximability for antiferromagnetic spin systems in the tree nonuniqueness region. *Journal of the ACM (JACM)*, 62(6):50, 2015.

[11] Andreas Galanis, Daniel Štefanković, and Eric Vigoda. Inapproximability of the partition function for the antiferromagnetic Ising and hard-core models. *Combinatorics, Probability and Computing*, 25(04):500–559, 2016.

[12] Heng Guo, Mark Jerrum, and Jingcheng Liu. Uniform sampling through the Lovász local lemma. In *Proceedings of the 49th Annual ACM Symposium on Theory of Computing (STOC)*, pages 342–355, 2017.

[13] Olle Häggström and Karin Nelander. Exact sampling from anti-monotone systems. *Statistica Neerlandica*, 52(3):360–380, 1998.

[14] Olle Haggstrom and Karin Nelander. On exact simulation of markov random fields using coupling from the past. *Scandinavian Journal of Statistics*, 26(3):395–411, 1999.

[15] Thomas P Hayes. A simple condition implying rapid mixing of single-site dynamics on spin systems. In *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 39–46, 2006.

[16] Mark Huber. Exact sampling and approximate counting techniques. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 31–40. ACM, 1998.
[17] Mark Huber. Perfect sampling using bounding chains. *The Annals of Applied Probability*, 14(2):734–753, 2004.

[18] Mark L Huber. *Perfect simulation*. Chapman and Hall/CRC, 2016.

[19] Mark Jerrum. A very simple algorithm for estimating the number of $k$-colorings of a low-degree graph. *Random Structures & Algorithms*, 7(2):157–165, 1995.

[20] Mark Jerrum. *Counting, sampling and integrating: algorithms and complexity*. Springer Science & Business Media, 2003.

[21] Daphne Koller, Nir Friedman, and Francis Bach. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[22] David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

[23] Liang Li, Pinyan Lu, and Yitong Yin. Correlation decay up to uniqueness in spin systems. In *Proceedings of the 24th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 67–84, 2013.

[24] Michael Luby and Eric Vigoda. Fast convergence of the glauber dynamics for sampling independent sets. *Random Structures & Algorithms*, 15(3-4):229–241, 1999.

[25] Elchanan Mossel and Allan Sly. Exact thresholds for Ising–Gibbs samplers on general graphs. *The Annals of Probability*, 41(1):294–328, 2013.

[26] James Gary Propp and David Bruce Wilson. Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures & Algorithms*, 9(1-2):223–252, 1996.

[27] Jesús Salas and Alan D Sokal. Absence of phase transition for antiferromagnetic Potts models via the Dobrushin uniqueness theorem. *Journal of Statistical Physics*, 86(1):551–579, 1997.

[28] Allan Sly and Nike Sun. The computational hardness of counting in two-spin models on $d$-regular graphs. In *Proceedings of the 53rd IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 361–369, 2012.

[29] Padhraic Smyth, Max Welling, and Arthur U Asuncion. Asynchronous distributed learning of topic models. In *Proceedings of the 22nd Advances in Neural Information Processing Systems (NIPS)*, pages 81–88, 2009.

[30] Eric Vigoda. Fast convergence of the glauber dynamics for sampling independent sets: Part II. Technical Report TR-99-003, International Computer Science Institute, 1999.

[31] Dror Weitz. Counting independent sets up to the tree threshold. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, pages 140–149, 2006.

[32] Dror Weitz and Alistair Sinclair. *Mixing in time and space for discrete spin systems*. University of California, Berkeley, 2004.