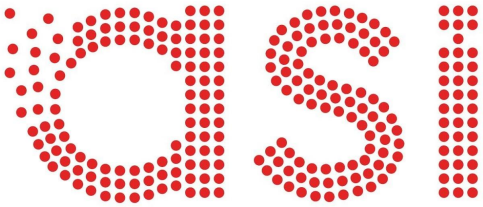

Conditional Independence Coupling

Authors: Kenton White, Guichong Li and Nathalie Japkowicz

Advanced Symbolics Inc
1 Rideau Street,
Suite 700
Ottawa, K1N 8S7
(613) 518-1644

2012 IEEE 12th International Conference on Data Mining Workshops

Sampling Online Social Networks Using Coupling From The Past

Kenton White
Girih
Ottawa, Canada
kenton.white@girih.com

Guichong Li
Computer Science,
University of Ottawa,
Ottawa, Canada
jli136@site.uottawa.ca

Nathalie Japkowicz
Computer Science,
University of Ottawa,
Ottawa, Canada
nat@site.uottawa.ca

***Abstract*—Recent research has focused on sampling online social networks (OSNs) using traditional Markov Chain Monte Carlo (MCMC) techniques such as the Metropolis-Hastings algorithm (MH). While these methods have exhibited some success, the techniques suffer from slow mixing rates by themselves, and the resulting sample is usually approximate. An appealing solution is to apply the state of the art MCMC technique, Coupling From The Past (CFTP), for perfect sampling of OSNs. In this initial research, we explore theoretical and methodological issues such as customizing the update function and generating small sets of non-trivial states to adapt CFTP for sampling OSNs. Our research proposes the possibility of achieving perfect samples from large and complex OSNs using CFTP.**

Keywords-Sampling; Online Social Networks; Markov Chain Monte Carlo, Coupling From The Past.

I. INTRODUCTION

Analyzing or Mining Online Social Networks (OSN) has become one of the most pressing problems of modern-day data mining. This is due to the exponential growth of these networks, which are becoming increasingly popular (e.g., Facebook, Twitter, Linkedln and cellular networks) and include more and more information that can be useful for commercial purposes, management issues, and even for defense and security applications.

However, the task for Social Network Analysis has been limited because the size of online networks is overwhelming. For example, the Facebook network takes up hundreds of terabytes of memory storage. The volume of information is expanding on a daily basis as more and more people join the service or post information. Processing it remains a daunting task. The only way that any kind of analysis can be made possible is by sampling from the huge network and working on this sample. Recent research has shown that this can be achieved by crawling OSNs to find a relatively small representative sample suitable for studying properties and testing algorithms on OSNs [8][9].

A number of existing techniques for crawling include Breadth First Search (BFS) [9] and Random Walk (RW). While such techniques usually yield a bias toward the most highly connected nodes [8][9], the main result is that crawling using the traditional Metropolis-Hastings algorithm (MH), which is a typical Monte Carlo Markov Chain (MCMC) technique, can create unbiased samples suitable for the problem of Social Network Analysis [8].

However, MCMC techniques such as the MH algorithm come with significant challenges: significant burn-in lengths (the number of steps of a chain to reach stationarity) and correlation with the initial node choice. This usually leads to slow mixing, i.e., a large $T_{mix}(\varepsilon) = min\{t: d(t) \leq \varepsilon\}$, where $d(t)$ is the distance from stationarity. For example, recent research has shown that the Metropolis-Hastings Random Walk (MHRW) algorithm can produce unbiased samples of Facebook by randomly requesting 84k samples for convergence after discarding the burn-in length 6k. On the other hand, various convergence diagnostic methods [1][2][7][12] such as Geweke Diagnostic [6] cannot guarantee the chain has converged to a sample value from the desired distribution. Therefore, the sample, e.g., 78k, obtained by such MCMC algorithm is usually approximate.

These issues behind the MH algorithm can be naturally overcome by using the state of the art Coupling From The Past (CFTP). In CFTP convergence is achieved by coalescence to a single state, which turns out to be a perfect sample from the stationary distribution. Therefore, issues of selecting proper convergence diagnostics are discarded [11][15][18]. While this technique is advanced, to the best of our knowledge, it has not applied in sampling OSNs.

Our research shows that there are at least two crucial issues for this application. Firstly, the update function in CFTP is a random map, which defines how a state moves to a new state. Defining an effective update function is not trivial. An improper update function might lead to failure to coalesce to a single state [15][19]. Secondly, the state space is usually unavailable for CFTP in advance of sampling. One has only a few initial nodes at the beginning for crawling. Even if the whole state space of OSNs is available, it is usually too huge to be used in CFTP.

We investigate new techniques for sampling OSNs by using CFTP. We develop a new CFTP by designing a proper update function and generating small sets of non-trivial states (rather than the whole state space). We reveal theoretical and methodological issues in this new technique. Our research result suggests that the standard CFTP technique can be successfully applied in sampling OSNs for perfect sampling. It is superior to other MCMC techniques such as the MHRW for sampling on large and complex networks, and the resulting sample is more suitable for social network analysis than other MCMC techniques.

978-0-7695-4925-5/12 $26.00 © 2012 IEEE
DOI 10.1109/ICDMW.2012.126

II. Related Work

A. Techniques for Sampling Online Social Networks

Sampling Online Social Networks (OSNs) can be achieved by crawling online social networks, producing a representative sample of users from the network suitable for conducting the analysis. In general, the crawling starts from an initial node and proceeds iteratively to visit all its neighbors. Several recently proposed methods for sampling OSNs are described as follows.

The Breadth First Search (BFS) method, which is regarded as a graph traversal technique, explores the next node assuming the traditional Breadth First Search algorithm. It has been used practically for sampling OSNs in past research [9]. Recent research also shows that the method might densely cover only some specific region of the graph due to incomplete search. Further, this bias can be corrected by deriving an unbiased estimator of the original node degree distribution [9].

The Random Walk (RW) method chooses the next state w uniformly at random among the neighbors of the current node v. The transition probability can be defined as

$$P(v,w) = \begin{cases} \frac{1}{k_v}, w \text{ is a neighbor of } v \\ 0, \quad otherwise \end{cases} \tag{1}$$

where k_v is the degree of v.

Because the probability of the RW at the particular node v converges to $\pi_v^{rw} \sim k_v$, the RW sample nodes are biased towards high degree nodes. This bias may be corrected by an appropriate re-weighting of the measured value such as the Hansen-Hurwitz estimator [8].

The Metropolis-Hastings Random Walk (MHRW) method [8], shown as below, appropriately modifies the transition probabilities so that it converges to the desired uniform distribution. The Metropolis-Hastings algorithm is a typical Markov Chain Monte Carlo (MCMC) technique for sampling from a probability distribution μ. For sampling from the uniform distribution $\mu_v = \frac{1}{|V|}$, the transition probability can be defined as

$$P(v,w) = \begin{cases} \frac{1}{k_v} \cdot min\left(1, \frac{k_v}{k_w}\right) & if\ w \text{ is a neighbor of } v, \\ 1 - \sum_{y \neq v} P(v,y) & if\ w = v \\ 0 & otherwise \end{cases} \tag{2}$$

Techniques for crawling using random walks are based on traditional Markov Chain Monte Carlo (MCMC) methods. Typically, the chain is started from an initial state, and it is run for some burn-in time long enough for the chain to have converged. The generated samples are assumed to be truly samples from the stationary distribution. Although various diagnostics such as Geweke Diagnostic and Gelman-Rubin Diagnostic [8] can be used for assessing convergence, none of them guarantees that the chain has exactly converged. As a result, the samples are usually only approximate. It has been shown that the MHRW requires a large number of rejections during the initial sampling process, and the method is subject to slow mixing [8].

MHRW(v)
// v: initial node
while not converged **do**
  $w \sim neighbors(v))$, chosen uniformly at random
  $p \sim U(0,1)$
  if $p \leq \frac{k_v}{k_w}$ **then**
    $v \leftarrow w$
  else
    $v \leftarrow v$
  end

B. Coupling From The Past

The CFTP algorithm, which was developed by Propp and Wilson [18], allows for perfect (exact) sampling from a desired distribution. It determines the burn-in time by ascertaining coalescence to a single value. Therefore, the issue of convergence diagnostic is ruled out. The fundamentals of the method can be described as follows.

Let P be a transition probability that defines an ergodic Markov chain on state space Ω. The transition probability is associated with a random function representation,

$$Pr(\Phi(x) = y) = P(x,y), \tag{3}$$

where $x, y \in \Omega$. That is, the probability of Φ mapping x to y is equal to the transition probability $P(x,y)$ in the Markov chain.

Assume that t_1 and t_2 are two time steps from the Markov chains. The composite map $F_{t_1}^{t_2}$, which describes evolution of Markov chains from t_1 to t_2 given any initial state x, can be defined as

$$\begin{aligned} F_{t_1}^{t_2}(x) &= \left(\Phi_{t_2-1} \circ \Phi_{t_2-2} \circ \cdots \circ \Phi_{t_1}\right)(x) \\ &= \left(\Phi_{t_2-1}\left(\Phi_{t_2-2}\left(\cdots \Phi_{t_1}(x) \cdots\right)\right)\right), \forall x \in \Omega. \end{aligned} \tag{4}$$

Therefore, F_0^t and F_{-t}^0, where $t \to \infty$, define forward coupling and backward coupling Markov chains, respectively. It has been observed [18] that the forward coupling does not necessarily produce a sample from the stationary distribution, and is subject to a bias due to changed coalescence time while the backward coupling produces an exact sample without any bias due to a fixed time for coalescence detection [4][18][19].

CFTP [18], shown as below, is a backward coupling technique for exact sampling. It assumes an ergodic Markov chain with discrete and finite state space Ω of size N. CFTP runs N copies of the chain from the past, where each copy corresponds to a different initial state. The chains will eventually coalesce to a steady state by time $t = 0$. Therefore, the effect of initial states in the Markov chains is

ruled out. This steady state is a perfect sample from the stationary distribution.

For simulation of coupling from the infinite past, CFTP runs the N chains starting at $T = -1$ and are checked for coalescence at $t = 0$. If the coalescence occurs, the single state of the chains at $t = 0$ is accepted as an effective sample from the stationary distribution π. Otherwise, the starting time is moved back to $T = 2T$ and the procedure is repeated until coalescence occurs.

Other algorithms for perfect sampling have been proposed in the literature. Fill's perfect rejection sampling algorithm [5] is based on rejection sampling, and can be interrupted at any time during the simulation without introducing any bias to the generated samples. The output of the algorithm is independent of the running time. The state space, however, is assumed to be finite and partially ordered with minimal and maximal states. Beyond finite state spaces, Murdoch and Green [15][17] have developed the standard CFTP into a continuous state space.

Instead of a perfect sample by singly running CFTP, a collection of sufficient number of independent perfect samples can be obtained by independently running the standard CFTP. This requires a large computational effort, and obtains a single sample in each CFTP. However, independent random streams with different random seeds could be a problem. Read-Once CFTP [20] can produce perfect samples by running the standard CFTP with a read-once random sequence stream and no re-use of the previously generated random numbers. Alternative methods such as Repeated CFTP [3][16] make use of the generated values using the CFTP for inference on exact samples with some degree of dependency.

CFTP algorithm
Input Ω: state space
Output X^0: singleton state
begin
  $T = -1, T_0 = 0$
  repeat
    $R^{T+1}, R^{-T+2}, \dots, R^{T_0} \sim U(0,1)$
    $t = T, X^t = \Omega$
    while $t < 0$
      $X^{t+1} = \Phi(X^t, R^{t+1})$
      $t = t + 1$
    $T_0 = T,\ T = 2T$
  until $|X^0| = 1$
end

To the best of our knowledge, CFTP has not been used for sampling OSNs in practice. We are motivated to apply CFTP for perfect sampling from online OSNs.

There are several crucial issues for success of practical applications of CFTP. The first is how to define the update function Φ_t for evolution of Markov chains when only the local transition probability P is available. The update function is also called a random map, which is just a random function representation of P as described above. An invalid Φ_t might lead to failure of coalescence in CFTP. Secondly, those $\Phi_t, t = -\infty, \dots, 0$, should be i.i.d generated for building the composite map F_t^0. This can be achieved by defining a random vector $R^t, t = -T, -T+1, \dots -1, 0$, generated uniformly at random from $U(0,1)$. Moreover, random numbers R^t generated in the previous iterations must be re-used. Thirdly, instead of the whole state space Ω, it is desired to identify a small state space Ω', which would not be trivial when sampling from large online social networks.

III. Sampling Online Social Networks Using CFTP

Two key issues for defining the update function and generating small sets of the whole state space will be discussed in this section. We, thus, develop a new algorithm using CFTP for sampling OSNs.

A. Update Function

The update function Φ is defined as follows:

$$X^{t+1} = \Phi(X^t, R^{t+1}), \tag{5}$$

where $R^{t+1} \sim U(0,1)$ and $X^t, X^{t+1} \subseteq \Omega$.

In more details, the update function maps a set of nodes X^t to a new set of nodes X^{t+1}. Each node in X^t is mapped to a new adjacent node based on the probability of the transition. For a graph, the probability of transition is typically a global property. Which adjacent node to select is determined by the random parameter R^{t+1}. The update function $\Phi(,)$ can be written in terms of a range $(R_{lower}, R_{upper}]$ for which the transition will occur. As discussed in Section II(B), the probability of Φ mapping x to one of its neighbors using the update function is equal to the transition probability. The key property of the update function $\Phi(X^t, R^{t+1})$ is that it be deterministic in R^{t+1}. Given a set of initial states X^{-T} and a Markov Chain R^{-T}, $R^{-T+1}, \dots, R^{-1}, R^0$, where $T > 0$, the set of states, nodes, and paths is completely deterministic.

For OSNs it is possible to estimate the update function $\Phi(X^t, R^{t+1})$ for each node $x \in X^t$ based on the adjacent nodes. There are two common methods for calculating the update function $\Phi(,)$ in OSNs: Random Walk and Metropolis-Hastings [13].

For each node x_i with degree k_i, we denote the set of adjacent nodes as $\{x_{i,j} | j \in [0, k_i - 1]\}$. In the Random Walk (RW), where each adjacent node is equally likely, the probability of transitioning to a node is

$$P(x_i, x_j) = \frac{1}{k_i} \tag{6}$$

and the update function can be given by

$$\Phi(x_i, R^{t+1}) = x_{i,j}, if\ R^{t+1} \in \left(\frac{j}{k_i}, \frac{j+1}{k_i}\right]. \tag{7}$$

Metropolis-Hastings (MH) modifies the probability $P(x_i, x_{i,j})$ to be

$$P(x_i, x_{i,j}) = min\left(\frac{1}{k_i}, \frac{1}{k_{i,j}}\right), \tag{8}$$

where $k_{i,j}$ is the degree of the adjacent node $x_{i,j}$. The MH allows for self transition with the probability being

$$P(x_i, x_{i,j}) = 1 - \sum_j P(x_i, x_{i,j}). \tag{9}$$

Then the update function is given by

$$\Phi(x_i, R^{t+1}) = \begin{cases} x_{i,j} \; if \; R^{t+1} \in \left(\sum_{k=0}^{j-1} P(x_i, x_{i,k}), \sum_{k=0}^{j} P(x_i, x_{i,k})\right] \\ x_i \; if \; R^{t+1} \in \left(\sum_j P(x_i, x_{i,j}), 1\right] \end{cases} \tag{10}$$

Example 1. Given a toy example used in the previous research [4][19], as shown in Figure. 1, we show how to define the update function using the MH method for CFTP.

The toy example describes a directed graph network. Instead of estimating the transition probabilities using the degrees of the node and its adjacent nodes, it typically has defined the global transition probabilities of the chain. Therefore, given the current state 1 the update function is actually estimated by using (10) as follows.

$$\Phi_{\mathrm{MH}}(1) = \begin{cases} 0, \text{if } \mathrm{R}^{\mathrm{t+1}} \in (0, 0.4] \\ 2, \text{if } \mathrm{R}^{\mathrm{t+1}} \in (0.4, 0.8] \\ 1, \text{if } \mathrm{R}^{\mathrm{t+1}} \in (0.8, 1] \end{cases}$$

The update function for other states can be estimated in the same way using the MH. The results are omitted due to the simplicity.

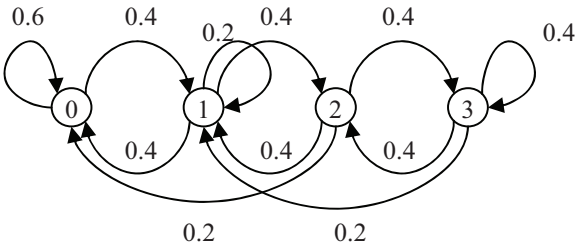

Figure 1. States of the Markov chain in the toy examples.

B. Non-trivial State Space

The standard CFTP constructs a global coupling with full copies of the chain from each state in Ω for producing a perfect sample. Given a large state space, this global coupling is prohibitive at the cost of computational time and space.

According to the definition the stationary distribution π,

$$\forall y \in \Omega, \qquad \pi(y) = \sum_{x \in \Omega} \pi(x) P(x, y).$$

Given a large state space Ω and a target state $y \in \Omega$, it is believed that a number of states x are independent of y such that $\exists \Omega' \subset \Omega$,

$$\pi(y) = \sum_{x \in \Omega'} \pi(x) P(x, y).$$

This implies that the stationary value $\pi(y)$ might only be associated with a few states from a small non-trivial state space Ω'. Coupling chains from only these initial states in Ω' is sufficient to coalesce to a perfect sample.

The idea behind a small state space has been introduced in the literature [14]. A set $\Omega' \subset \Omega$ is called a *small set* of order m if there exists an $m > 0$, and a non-trivial measure v_m on Ω, such that for all $x \in \Omega'$ and $\forall A \subseteq \Omega$, $P^m(x, \mathrm{A}) \geq v_m(\mathrm{A})$. The central result is that small sets exist for any Markov chain, and then a collection of small sets can be used for exploring the whole state space Ω. We discuss this kind of small sets in the context of coupling techniques as follows.

The update function, as shown in (5), defines a random map, which is a deterministic function,

$$\Phi_{\mathrm{t}}(x): \Omega \to \Omega \tag{11}$$

In essence, Φ_{t} and $F_{\mathrm{t}_1}^{\mathrm{t}_2}$ can define a limited random map as

$$\Phi_{\mathrm{t}}(x), F_{t_1}^{t_2}(x): \Omega' \to \Omega \tag{12}$$

This non-trivial state space Ω' can be formally defined as follows.

Definition 1. Given $\Phi(\Omega, R^t)$, denoted as $\Phi_{\mathrm{t}}(\Omega)$, and the composite function

$$F_{\mathrm{t}_1}^{\mathrm{t}_2}(x) = \left(\Phi_{\mathrm{t}_{2-1}} \circ \Phi_{\mathrm{t}_{2-2}} \circ \cdots \circ \Phi_{\mathrm{t}_1}\right)(x),$$

where $t_1 < t_2$ and $\forall x \in \Omega$, if $t_2 - t_1$ is sufficient large, s.t., $F_{\mathrm{t}_1}^{\mathrm{t}_2}(x) \subseteq \Omega' \subseteq \Omega$, then Ω' is called a *non-trivial state space* with respect to t_2.

A not-trivial state space Ω' is a small set of Ω. Simply, Ω is also a non-trivial state space by itself. One may expect that $F_{-M}^{0}(\Omega')$ can coalesce to an exact sample from π given M. This can be justified by the following theorem.

Theorem 1. $F_{-M}^{0}(\Omega')$ has the same distribution as π.

Proof: For $t_1 < t$, we have

$$F_{-t}^{0} = F_{-t_1}^{0} \circ F_{-t}^{-t_1-1}.$$

If $F_{-M}^{0}(\Omega')$ is a constant function, then for all sufficient large $t > M$, from the assumption that Ω' is a non-trivial state space for $-M-1$,

$$F_{-t}^{-M-1}(\Omega) \subseteq \Omega'. \tag{14}$$

and

$$F_{-t}^{0}(\Omega) = F_{-M}^{0} \circ F_{-t}^{-M-1}(\Omega) = F_{-M}^{0}(\Omega').$$

This shows that output of $F_{-M}^{0}(\Omega')$ is the same as the output of $F_{-\infty}^{0}(\Omega)$ as $t \to \infty$. Note that

$$\lim_{t\to\infty} P(F_{-t}^{0}(x) = y) = \pi(y).$$

Therefore, the theorem is proven. ■

Consider (14) as $t \to \infty$, it is equivalent to say that the backward coupling from the state space Ω "coalesces" to a non-trivial state space Ω' instead of a single state. This lets us design an algorithm to generate a non-trivial state space given initial states X_0.

We propose the Non-Trivial State Space algorithm (NTSS) to generate a non-trivial state space for some initial states X_0. Note that the standard CFTP produces a perfect sample by obtaining the coalescence of global coupling. Instead of only a single perfect sample, the idea behind NTSS is to adapt the CFTP algorithm to search for non-trivial states given X_0. The additional parameter τ is specified as the fixed step size for backward coupling rather than the doubling scheme used in the standard CFTP; it can be used to describe the distance between non-trivial states with a fixed time interval. The larger the length, the less their relationship is.

In essence, the generated Ω' is a customized small set covering X_0, which usually contains a single state seed in practice. The purpose is to use the generated non-trivial state space for perfect sampling rather than those initial seeds X_0.

Non-Trivial State Space algorithm
Input N: the size of non-trivial state space; X_0: initial states; τ: the fixed step size for backward coupling, e.g., $\tau = -5$;
Output Ω': a non-trivial state space w.r.t. X_0
begin
  $\Omega' = X_0$
  $T = \tau, T_0 = 0$
  repeat
    $R^{T+1}, \dots, R^{T_0} \sim U(0,1)$
    $t = T, X^t = \Omega'$
    while $t - T_0 < 0$
      $X^{t+1} = \Phi(X^t, R^{t+1})$
      $t = t + 1$
    $\Omega' = \Omega' \cup X^0$
    $T_0 = T,\ T = T + \tau$
  until $|\Omega'| \geq N$
end

Example 2. Given the directed graph network and the states and the transition probabilities of the Markov chain, as shown in Example 1, there are only two initial states, 0 and 1, covered by the dashed double circles. The NTSS is run with N = 4 and $\tau = -2$. We show how the NTSS produces a non-trivial state space of 0 and 1, as shown in Figure 2.

In the first iteration of the while loop in the NTSS, the generated random vector is $[R^{-1}, R^0] = [0.7, 0.3]$. $\Omega' = \{0,1\}$ is not changed because no new state is found at $t = 0$.

In the second iteration of the while loop, the random vector is $[R^{-3}, R^{-2}] = [0.1, 0.5]$, and $\Omega' = \{0,1,2\}$.

The process continues until the non-trivial state space Ω' with the size N = 4 is obtained. This can take place as long as some random vector, e.g., $[R^t, R^{t+1}] = [0.5, 0.7]$, is drawn in later iterations of the while loop, at which point, $\Phi(2, 0.7) = 3, R^t \in (0.6,1]$.

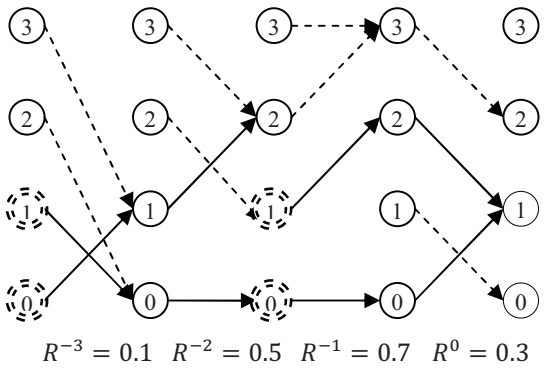

Figure 2. Generating a non-trivial state space of 0 and 1 using the NTSS on the toy example.

C. Online CFTP Algorithm

According to the previous discussion, we propose the Online CFTP for perfect sampling on OSNs. The algorithm first generates a non-trivial state space Ω' from given initial states X_0 using the proposed NTSS. We assume that X_0 only contains a single initial state without any loss of generality. Then the algorithm runs the standard CFTP with the customized update function defined in (7) or (10) for perfect sampling from Ω'. More details about the Online CFTP will be discussed in Section IV.

Online CFTP algorithm
Input N: the total number of non-trivial states; X_0: initial states consisting of some nodes as sample seeds from a given OSN, e.g., Facebook, etc;
Output a single value in X^0
begin
  $\Omega' = NTSS(N, X_0, \tau)$, e.g., $\tau = -5$
  $T = -1, T_0 = 0$
  repeat
    $R^{T+1}, R^{T+2}, \dots, R^{T_0} \sim U(0,1)$
    $t = T, X^t = \Omega'$
    while $t < 0$
      $X^{t+1} = \Phi(X^t, R^{t+1})$, defined in (7) or (10)
      $t = t + 1$
    $T_0 = T,\ T = 2T$
  until $|X^0| = 1$
end

IV. DISCUSSION

As discussed above, to implement the Online CFTP algorithm for sampling OSNs, one of the crucial issues is to effectively design the update function in the Online CFTP. We further discuss some related important issues as follows.

A. Comparison of Two Updation Functions

Two common methods: RW and MH for probability transition exhibit quite different performance. The uniform method allows for a Markov chain evolving from the current state to the next new state with equal probability. The MH method heuristically estimates the probability distribution for transition of states in a Markov chain by estimating local density of nodes. Therefore, the MH can be more powerful than the uniform method in complex networks.

B. Update Function for Self Transition

The proposed update function is a function of a random variable and is govern by the transition probability of the node. Its probability is equal to the transition probability.

However, it is observed that all Markov chains in the Online CFTP may stay at the current state at some time step t when the random number is large, e.g., $R^t \simeq 1$, from (10). This can be regarded as a failed update for coupling when the update function is used. One may want to avoid this kind of failure for fast coalescence in the Online CFTP.

In practice, the backward coupling with the update function in (10) can be simulated using the Metropolis-Hastings (MH) algorithm. However, failed updates might occur as well since the MH algorithm usually has a low acceptance rate. As a result, many of chains might not go fast forward.

To this end, one can introduce a different random number $\hat{R}_i = \hat{R}(x_i)$ which is associated with each state x_i. $\hat{R}(x_i)$ is used for updating R^t by

$$R^t = \begin{cases} R^t + \hat{R}_i, if\ R^t + \hat{R}_i \leq 1 \\ R^t + \hat{R}_i - 1, otherwise \end{cases}$$

Therefore, this can be re-written as

$$R^t = \left(R^t + \hat{R}_i\right) \% \, 1.0 \tag{15}$$

Because R^t in the right side of Equation (15) is a uniform random number on the interval (0, 1) and $\hat{R}_i$ is fixed for each x_i, the resulting R^t in the left side of Equation (15) is also uniformly distributed on the interval (0,1) due to the liner transformation. This avoids self transitions occurring at the same time steps in all coupling chains when the update function in (10) is used. In essence, the resulting R^t is intended be different between states, and is still uniformly distributed for each state.

C. Initialization of Random Sequences

In Online CFTP, the random sequence R^t is generated uniformly at random from $U(0,1)$. With a different random seed, the used random generator for R^t is initialized at the beginning of Online CFTP for a different random vector. As a result, the algorithm can produce different independent perfect samples with different random seeds. Online CFTP can be repeatedly run with each of the previously obtained perfect samples to generate as many independent perfect samples as needed.

The method for independent perfect samples, as described above, is comparable with the Read-Once CFTP for independent perfect samples with a few random seeds. It is also superior to other algorithms such as Repeated CFTP (RCFTP) [3] for inferring perfect samples that might be dependent between them.

D. Bounding Mixing time

Social networks consists of nodes, which are indexed with their identifiers, denoted by integers of k-digits, e.g., $k = 12$ on Twitter. A Markov chain on the state space Ω consisting of all nodes of a social network is constructed as follows: at time t, pick any neighbor y of x_t uniformly at random and a bit $b \in \{0,1\}$ uniformly at random; if $b = 0$, $x_{t+1} = x_t$; otherwise $x_{t+1} = y$.

Because the indexes of nodes of social networks can be represented by n-bit binary string, the chain described above is equivalent to a simple random walk on the hypercube $\{0,1\}^n$: pick a coordinate $i \in \{1, \dots, n\}$ uniformly at random and a bit $b \in \{0,1\}$ uniformly at random; set $x_i = b$.

Bounding the mixing time for a random walk on the hypercube can be analyzed by coupling [10],

$$\tau(\varepsilon) = \mathrm{O}\big(nlog(n/\varepsilon)\big).$$

Similarly, the mixing time of our method can be bounded with a number related to the length of integers for indexes. Empirically, we show that Online CFTP can be efficiently implemented for sampling OSNs.

V. OBSERVED MIXING TIME ON TWITTER

A rapid mixing time is crucial for sampling OSNs. This requires that the coalescence time should be bounded within an acceptable amount. We conducted the first experiment to show empirically the mixing time of Online CFTP on Twitter.

We ran two chains starting from two different initial states, which were generated using NTSS on Twitter, and are diverse by setting the fixed step size $\tau = -5$. We ran 7 experiments by using different random sequences in NTSS for generating different small sets given the same initial state seed. We observed that the simulation steps, as shown in Table 1, are usually more than two thousand steps with the elapsed time over 6hs on average for a perfect sample.

For comparison, people have adopted the MHRW algorithm to run 28 chains separately for sampling Facebook [8]. The Geweke diagnostic is applied in each of the 28 chains. The convergence is detected when all 28 values fall in the [-1,1] interval. In general, after a long run, e.g., 500-2000 iterations, a z-score falls within the interval.

For analyzing all the 28 chains, the Gelman-Rubin diagnostic is used, and an R score is computed. It has been observed that after a long run, e.g., 3000 iterations, all the R scores drop below 1.02. Because the convergence detection is not precisely implemented, the resulting sample is obtained approximately by discarding the initial sample nodes before the mixing time.

As we can see, the observed mixing time of Online CFTP from our first experiment on Twitter approximates to that of the MHRW algorithm on Facebook. On average, Online CFTP can exhibit the mixing time less than the MHRW by 2798 vs. 3000.

Table 1. Simulation steps(observed mixing time<0) and elapsed time(s) for Online CFTP on Twitter

Simulation steps	Elapsed time
-2048	9657.72
-4096	40139.10
-4096	17556.56
-1024	8350.87
-4096	9765.21
-128	281.93
-4096	50772.72
-2798	19503.44

VI. CONLCUSION AND FUTURE WORK

In this research, we develop a new technique for sampling OSNs using CFTP. We discuss and analyze theoretical and methodological issues for this new technique. Two key issues are tackled. The update function is set by selecting either the RW or the MH methods, which are defined according to transition probabilities; the update function with the MH method is realized to be more robust than the update function with the RW method for sampling on complex networks. A small set of states from the whole state space, called non-trivial state space, can be generated by using the proposed Non-Trivial State Space algorithm. This is achieved by modifying the standard CFTP to search for non-trivial states with respect to some given initial state seeds. Finally, we propose the Online CFTP algorithm for sampling OSNs. Related issues such as producing independent perfect samples and bounding the mixing time using Online CFTP are discussed. The initial research results suggest that the state of the art CFTP can be successfully applied in sampling OSNs. The results from our first experiment show that the proposed technique has an observed mixing time less than the previously proposed MCMC techniques such as the MHRW for sampling OSNs. There is still potential improvement in the new technique. Our future work will further explore the characteristics of the proposed approach, and show the quality of the resulting sample for various data mining tasks.

REFERENCES

[1] S. Brooks and G. Roberts, “Assessing convergence of Markov chain Monte Carlo algorithms,” Statist. Comput., pp. 319–335, 1999.

[2] M. Cowles and B. Carlin, “Markov chain Monte Carlo convergence diagnostics: A comparative study,” J. Amer. Statist. Assoc., vol. 91, pp. 883–904, 1996.

[3] X. Dimakos, "A guide to exact simulation," Int. Statist. Rev., 69, 27−48, 2001.

[4] P. M. Djurić, Y. Huang, and T. Ghirmai, “Perferct sampling: A review and applications to signal processing,” IEEE Trans. Signal Processing, vol. 50, no. 2, pp. 345-356, 2002.

[5] J. Fill : “An interruptible algorithm for perfect sampling via Markov chains,” The Annals of Applied Probability, 8 (1998), 131−162.

[6] J. Geweke, “Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments,” in Bayesian Statistics, 1992, pp. 169–193.

[7] A. Gelman and D. Rubin, “Inference from iterative simulation using multiple sequences,” in Statistical science, vol. 7, no. 4, 1992, pp. 457– 472.

[8] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou, "Walking in Facebook: A Case Study of Unbiased Sampling of OSNs," INFOCOM, 2010 Proceedings IEEE, 2010.

[9] M. Kurant, A. Markopoulou, and P. Thiran, “On the bias of BFS (Breadth First Search),” in Proc. 22nd Int. Teletraffic Congr., also in arXiv:1004.1729, 2010.

[10] David A. Levin, Yuval Peres, and Elizabeth L. Wilmer, "Markov Chains and Mixing Times," AMS, 2009.

[11] X. L. Meng, "Towards a more general Propp–Wilson algorithm: multistage backward coupling," Monte Carlo Methods–Fields. Inst. Commun. 26, 85–93, 2000.

[12] K. Mengersen, C. Robert, and C. Guihenneuc-Jouyaux, “MCMC convergence diagnostics: A review,” in Bayesian Statistics 6, J. Bernardo, J. Berger, A. Dawid, and A. Smith, Eds. Oxford, U.K.: Oxford Univ. Press, 1999.

[13] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, "Equations of State Calculations by Fast Computing Machines", Journal of Chemical Physics 21 (6): 1087-1092, 1953.

[14] S. P. Meyn and R. L. Tweedie, "Markov Chains and Stochastic Stability," Springer-Verlag, New York, 1993.

[15] D. Murdoch, "Exact sampling for Bayesian inference: unbounded state spaces," Monte Carlo Methods—Fields Inst. Commun. 26, 111–121, 2000.

[16] D. J. Murdoch and J. S. Rosenthal, "Efficient use of exact samples," Statistics and Computing, 10, 237-243, 2000.

[17] D. J. Murdoch and P. J. Green, "Exact sampling from a continuous state space," Scandinavian Journal of Statistics 25 (1998), 483-501.

[18] J. G. Propp and D. B. Wilson, "Exact sampling with coupled markov chains and applications to statistical mechanics," Random Structures and Algorithms 9 (1996), 223-252.

[19] E. Thönnes, "A primer on perfect simulation," Statistical Physics and Spatial Statistics, (K.R. Mecke and D. Stoyan, eds.), Springer Lecture Notes in Physics, vol. 554, Springer-Verlag, 2000, pp. 349-378.

[20] D. B. Wilson, "How to couple from the past using a read-once source of randomness," Proceedings of the Ninth International Conference Random Structures and Algorithms," vol. 16, Random Structures and Algorithms, no. 1, 2000, pp. 85-113.