Uniform and Ergodic Sampling in Unstructured Peer-to-Peer Systems with Malicious Nodes

Emmanuelle Anceaume, Yann Busnel, Sebastien Gambs

Uniform and Ergodic Sampling in Unstructured Peer-to-Peer Systems with Malicious Nodes

Emmanuelle Anceaume[1], Yann Busnel[2], and Sébastien Gambs[3]

[1] IRISA / CNRS Rennes (France), emmanuelle.anceaume@irisa.fr
[2] LINA / Université de Nantes (France), Yann.Busnel@univ-nantes.fr
[3] IRISA / Université de Rennes 1 - INRIA (France), sebastien.gambs@irisa.fr

Abstract. We consider the problem of uniform sampling in large scale open systems. Uniform sampling is a fundamental primitive that guarantees that any individual in a population has the same probability to be selected as sample. An important issue that seriously hampers the feasibility of uniform sampling in open and large scale systems is the unavoidable presence of malicious nodes. In this paper we show that restricting the number of requests that malicious nodes can issue and allowing for a full knowledge of the composition of the system is a necessary and sufficient condition to guarantee uniform and ergodic sampling. In a nutshell, a uniform and ergodic sampling guarantees that any node in the system is equally likely to appear as a sample at any non malicious node in the system and that infinitely often any node has a non-null probability to appear as a sample of honest nodes.

Keywords: Uniform sampling, unstructured peer-to-peer systems, ergodicity, Byzantine adversary.

1 Introduction

We consider the problem of uniform sampling in large scale open systems with adversarial (Byzantine) nodes. Uniform sampling is a fundamental primitive guaranteeing that any individual in a population has the same probability to be selected as sample. This property is of utmost importance in systems in which the population is continuously evolving and where it is impossible to capture the full complexity of the network through global snapshots. By collecting random subsets of information over the network, one can infer at almost no cost some global characteristic of the whole population (such as its size, its topological organization, its resources, ...). Therefore uniform sampling finds its root in many problems such as data collection, dissemination, load balancing, and data-caching [1–4].

Providing unbiased (*i.e.*, uniform) sampling in these open systems is a challenging issue. First, this primitive must cope with the continuous change of the network structure caused by nodes departures and arrivals. Nevertheless, it has been shown through simulations [1, 5] and analytic studies [6–8] that simply maintaining a partial and small local view of node identifiers (ids) is sufficient to

provide near uniform sampling. This can be achieved through gossip-based algorithms [1, 9, 10] or through random walks [5, 11–13]. Gossip-based algorithms mainly consist, for each node v in the system, in periodically selecting some other node w in v's local view and exchanging information. Information can either be pushed to other nodes or pulled from other nodes. Over time, information spreads over the system in an epidemic fashion allowing each node to continuously update its local view with fresh node ids. On the other hand, a random walk on a network (which can be represented as a graph) is a sequential process, starting from an initial node v, which consists in visiting a node in v's neighborhood according to some randomized order. In its simpler form, the next node is chosen uniformly at random among the neighbors, while more sophisticated choices are implemented to cope with the bias introduced towards high degree nodes (for instance, through the Metropolis-Hastings algorithm [14]).

An important issue that seriously hampers the feasibility of uniform sampling in open and large scale systems is the unavoidable presence of malicious nodes. Malicious (or Byzantine) nodes typically try to manipulate the system by exhibiting undesirable behaviors [15]. In our context, they try to subvert the system by launching targeting attacks against nodes in the aim of biasing uniformity by isolating honest nodes within the system. This is quickly achieved by poisoning local views of honest nodes with malicious node ids. For instance in unstructured graphs, a number of push operations logarithmic in the size of local views is sufficient to fully eclipse honest nodes from the local view of a node [16], while in structured graphs, a linear number of join operations is required [17]. Recent works have been proposed to detect and exclude these adversarial behaviors [18–20] by observing that malicious nodes try to get an in-degree much higher than honest nodes in order to isolate them. Extensive simulations [18] have shown that this approach is only highly effective for a very small number of malicious nodes (*i.e.*, in $\mathcal{O}(\log |\mathcal{S}|)$ where $|\mathcal{S}|$ is the size of the network $\mathcal{S}$), otherwise detection mechanisms may boil down to false positive detection (*i.e.*, detection of honest nodes).

On the other hand, when the system is harmed by a large number of malicious peers (*i.e.*, a linear proportion of the nodes of the system), which is definitively a realistic assumption in peer-to-peer systems [15, 21], additional mechanisms are required to prevent targeted attacks from succeeding. Specifically, in structured peer-to-peer systems, analytical studies have shown that applying the "induced churn" principle allows to defend the system against adversarial behaviors, either through competitive induced churn strategies [22], or through global induced churn [23]. Briefly, this principle states that, by forcing nodes to periodically change their position in the graph, malicious peers cannot predict the evolution of the state of the system after a given sequence of join and leave operations. By taking advantage of the properties of structured graphs, the authors of both papers have shown that, with high probability, any node is equally likely to appear in the local view of each other honest node in a number of rounds polynomial in the size of the system. Unfortunately, in unstructured peer-to-peer systems, nodes cannot rely on the topological nature of structured graphs to reject new

node ids that do not conform to the imposed distance function (contrary to structured networks [22, 23]). To circumvent this issue, Bortnikov *et al.* [16] rely on the properties of min-wise independent permutations, which are fed by the streams of gossiped node ids, to eventually converge towards uniform sampling on the node ids. More precisely, these authors have derived an upper bound on the expected time T_s to converge towards unbiased (uniform) samples. However, by construction, this convergence is definitive in the sense that once a random sample has been locally observed it is kept as a local sample forever. As a consequence, beyond the time limit T_s, no other node ids received in the input stream can ever appear in the random sample. The property of a sampler to guarantee that each received node id infinitely often has a non-null probability to locally appear as a sample is called the *ergodic sampling* property (this property is formally defined later in the paper).

Intuitively, this lack of adaptivity seems to be the only defense against adversarial behavior when considering bounded resources (memory and bandwidth). This paper is devoted to the formal analysis of the conditions under which uniform and ergodic sampling is feasible or not. More precisely, *the main contribution of this paper* is to show necessary and sufficient conditions under which uniform and ergodic sampling is achievable in unstructured peer-to-peer systems potentially populated with a large proportion of Byzantine nodes. Specifically, let $\mathcal{S}$ represent the wide collection of nodes in the system, and $k < 1$ the proportion of malicious nodes in $\mathcal{S}$. Let δ_m be the number of (not necessarily unique) malicious node ids gossiped by malicious nodes during a time interval T_s, and Γ denote the local memory of any honest node u in $\mathcal{S}$. In this context, we prove the following assertions:

- If the number δ_m of (non-unique) malicious ids received at node u during a given period of time T_s is strictly greater than $T_s - |\mathcal{S}|(1-k)$ then, neither uniform sampling nor ergodic sampling can be achieved;
- If $\delta_m \leq T_s - |\mathcal{S}|(1-k)$ and the size of the memory Γ is greater than or equal to $|\mathcal{S}|$ then, both uniform and ergodic sampling can be achieved;
- If $\delta_m \leq T_s - |\mathcal{S}|(1-k)$, and $|\Gamma| < |\mathcal{S}|$ then, uniform and ergodic sampling cannot be achieved.

Briefly, these conditions show that if the system cannot provide the means to limit the number of messages an adversary can periodically send, then solving either uniform sampling or ergodic sampling is impossible. On the other hand, if this assumption holds and if all honest nodes in the system have access to a very large memory (in the size of the network) then, the problem becomes trivially solvable. Unfortunately, as will be shown, both conditions are necessary and sufficient to solve the uniform and ergodic sampling problem. Clearly, these strong conditions highlight the damage that adversarial behavior can cause in large-scale unstructured systems.

To the best of our knowledge, we are not aware of any previous work that has specified the conditions for which uniform and ergodic sampling is reachable in presence of adversarial behaviors.

The outline of this paper is the following. In the next section, we describe the model of the system and how it is vulnerable to malicious nodes. Afterwards in Section 3, we define uniform and ergodic sampling, while in Section 4, related work is presented. Finally, Section 5 identifies the two conditions for which uniform and ergodic sampling is achievable, before concluding in Section 6.

2 System Model

An overlay network is a logical network built on top of a physical network. We consider an overlay network $\mathcal{S}$ populated with nodes labelled through a system wide identifier. We assume that a unique and permanent identifier is assigned to each node. In the following, nodes identifiers are abbreviated by node ids. Nodes communicate among each other along the edges of the overlay by using the communication primitives provided by the underlying network (*e.g.*, IP network service). Nodes are free to join and leave the overlay at any time. The particular algorithms use by nodes to choose their neighbors and to route messages induce the resulting overlay topology. In particular, the topology of unstructured overlays conforms with that of random graphs (*i.e.* relationships among nodes are mostly set according to a random process).

2.1 Adversary

A fundamental issue faced by any practical open system is the inevitable presence of nodes that try to manipulate the system by exhibiting undesirable behaviors [15]. Such nodes are called malicious or Byzantine nodes. Malicious nodes can simply display behaviors such as simply dropping or re-routing messages towards other malicious nodes, or they can devise more complex strategies such as mounting eclipse attacks (also called routing-table poisoning [15, 24]) by having honest nodes redirecting outgoing links towards malicious ones. Moreover, they can magnify the impact of their attacks by colluding and coordinating their actions. In our work, we do not consider Sybil attacks [21], which mainly consist in flooding the system with numerous fake identifiers. We assume the existence of some external mechanism for solving this problem (for instance an off-line certification authority, *cf.* Section 2.2). We model malicious behaviors through a strong adversary that fully controls these malicious nodes. The adversary has the ability to inspect the whole overlay and strategizes on the time at which malicious nodes operations must be issued. We assume that the adversary cannot control more than a fraction $k < 1$ of malicious nodes in the overlay. A node which always follows the prescribed protocols is called *honest*. Note that honest nodes cannot *a priori* distinguish honest nodes from malicious ones, which would otherwise render the problem trivial.

2.2 Security Mechanisms

We assume the availability of a signature scheme that enables to verify the validity of a signature on a message (i.e. the authenticity and integrity of this

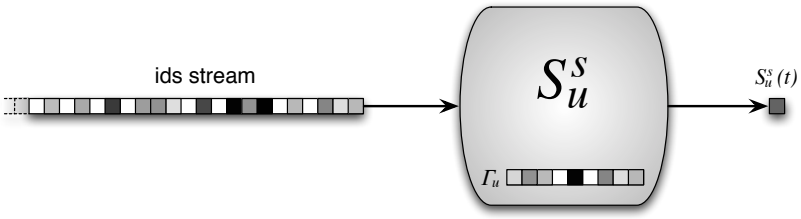

Fig. 1. Sampling component of node $u \in \mathcal{N}$.

message with respect to a particular node). Recipients of a message ignore any message that is not signed properly. Nodes ids and keys (private and public) are acquired via a registration authority [24] and it is assumed that honest nodes never reveal their private keys to other nodes. We also assume the existence of private channels (obtained through cryptographics means) between each pair of nodes preventing an adversary from eavesdropping and unnoticeably tampering with the content of a message exchanged between two honest nodes through this channel. However of course, a malicious node has complete control over the messages it sends and receives.

3 Uniform and Ergodic Sampling

In this section, we describe the terminology and assumptions used in this paper and then define uniform and ergodic sampling.

3.1 Assumptions and Terminology

Similarly to Bortnikov *et al.* [16], we consider the following assumptions. There exists a time T_0 such that from time T_0 onwards, the churn of the system ceases. This assumption is necessary to make the notion of uniform sample meaningful. Thus from T_0 onwards, the population of the system $\mathcal{S}$ is composed of $|\mathcal{S}|$ nodes, such that at least $(1-k)|\mathcal{S}|$ of them are honest and no more than $k|\mathcal{S}|$ of them are controlled by the adversary (for $k < 1$). The subset of honest nodes in the overlay is denoted by $\mathcal{N}$ and we assume that all the nodes in $\mathcal{N}$ are weakly connected from time T_0 onwards.

Each node $u \in \mathcal{N}$ has locally access to a *sampling component*[4] as presented in Figure 1. The sampling component implements a *strategy* s and has uniquely access to a data structure Γ_u, referred to as the *sampling memory*. The size of the sampling memory Γ_u is bounded and is denoted by $|\Gamma_u|$. The sampling

[4] Although malicious nodes have also access to a sampling component, we cannot impose any assumptions on how they feed it or use it as their behavior can be totally arbitrary.

component S_u^s is fed with (non unique) node ids that correspond to the node ids periodically received (either through gossip algorithms or through random walks). This stream of node ids may contain repetition of the same node id, which can be particularly frequent for malicious node ids, as discussed later. At each time t, the following three steps are executed: the first element of the stream, say node id v, is given as input to the sampler component. The sampling component S_u^s reads v, and removes it from the stream. According to its strategy s, S_u^s may store or not v in Γ_u (for example, the strategy s may consist in storing v if Γ_u is not full or in substituting v for a randomly chosen node id in Γ_u), and outputs at most one node id v'. The output at time t is denoted $S_u^s(t)$. The produced node id v' is chosen among the node ids in Γ_u according to the strategy s (for instance, strategy s may choose the smallest node id in Γ_u or the smallest node id under a given min-wise permutation [16]). Note that these three steps are atomically done. The maximum finite hitting time needed for the sampling component S_u^s to reach a uniform sample is denoted by T_s. Clearly T_s depends on the strategy s implemented by the sampling component and also on the stream of node ids the sampling component has access to. Finally, δ_m represents the number of malicious node ids received (possibly multiple times) in the stream of node ids at node u during the time interval T_s.

3.2 Sampling Properties

We consider the problem of achieving an unbiased (uniform) and ergodic sampling in large scale unstructured peer-to-peer systems subject to adversarial attacks. A strategy s that solves this problem has to meet the following two properties: *i)* Uniformity, which states that any node in the overlay should have the same probability to appear in the sample of honest nodes in the overlay, and *ii)* Ergodicity, which states that any node should have a non-null probability to appear infinitely often in the sample of any honest nodes in the overlay. More formally, strategy s should guarantee:

Property 1 (Uniformity). Let $\mathcal{N}$ be a weakly connected graph from time T_0 onwards, then for any time $t \geq T_s$, for any node $u \in \mathcal{S}$, and for any node $v \in \mathcal{N}$,

$$\mathbb{P}[u \in S_v^s(t)] = \frac{1}{|\mathcal{S}|}.$$

Property 2 (Ergodicity). Let $\mathcal{N}$ be a weakly connected graph from time T_0 onwards, then for any time $t \geq T_0$, for any node $u \in \mathcal{S}$, and for any node $v \in \mathcal{N}$,

$$\mathbb{P}\left[\{t' | t' > t \wedge u \in S_v^s(t')\} = \emptyset\right] = 0,$$

where $\emptyset$ represents the empty set. In the following, Properties 1 and 2 are respectively denoted $\mathcal{U}$ and $\mathcal{E}$.

Remark 1. Uniformity by itself does not imply ergodicity and conversely, ergodicity by itself does not imply uniformity. Indeed, Property 1 guarantees that any

node (honest or not) has an equal probability to be sampled by any honest node in the system. Nonetheless, once convergence to a random sample locally holds, this property does not say that this sample must change over time to provide a fresh and random node id (this is definitively important for data-caching applications which continuously require fresh node id). Guaranteeing this dynamicity is formalized by Property 2 which states that each node has a non-null probability to be selected as a sample at any time, guaranteeing the access of new sample graphs.

4 Related Work

In the literature, different approaches have been proposed to deal with malicious behaviors, each one focusing on a particular adversarial strategy.

With respect to eclipse attacks, a very common technique, called *constrained routing table*, relies on the uniqueness and impossibility of forging nodes identifiers. It consists in selecting as neighbors only the nodes whose identifiers are closer to some particular points in the identifier space [24]. Such an approach has been successfully implemented into several overlays (*e.g.*, CAN, Chord, Pastry). More generally, to prevent messages from being misrouted or dropped, the seminal works of Castro *et al.* [24] and Sit and Morris [15] on distributed hash tables based overlays combine routing failure tests and redundant routing as a solution to ensure robust routing. Their approach has then been successfully implemented in different structured-based overlays (*e.g.*, [25–27]). In all these previous works, it is assumed that at any time, and anywhere in the overlay, the proportion of compromised nodes is bounded and known, allowing powerful building blocks such as Byzantine tolerant agreement protocols to be used among peers subsets [26, 27]. When such an assumption fails, additional mechanisms are needed. For instance, Awerbuch *et al.* [22] propose the *Cuckoo&flip* strategy, which consists in introducing local induced churn (*i.e.*, forcing a subset of nodes to leave the overlay) upon each join and leave operation. This strategy prevents malicious nodes from predicting what is going to be the state of the overlay after a given sequence of join and leave operations. Subsequently to this theoretical work, experiments have been conducted to verify the practical feasibility of global induced churn, which consists in having all the nodes of the overlay periodically leaving their positions. These experiments assume that the overlay is populated by no more than $k = 25\%$ of compromised nodes [28]. Authors of [23] have analyzed several adversarial strategies, and show that an adversary can very quickly subvert DHT-based overlays (DHT for Distributed Hash Tables) by simply never triggering leave operations. They also show that when all nodes (honest and malicious ones) are imposed a limited lifetime, the system eventually reaches a stationary regime where the ratio of corrupted clusters is bounded, independently from the initial amount of corruption in the system.

Jesi *et al.* [18] propose a random sampling algorithm that deals with malicious nodes. Their solution assumes that the ultimate goal of the malicious nodes is

to mutate the random graph in a hub-based graph, hub for which malicious nodes gain the lead. Once this goal is reached, malicious nodes can very quickly and easily subvert the whole overlay by performing denial-of-service attacks. Conducting a hub attack mainly consists for malicious nodes in increasing their in-degree. Jesi *et al.* [18] propose to detect highly popular nodes by extending classic node sampling services with a module that identifies and blacklists nodes that have an in-degree much higher than the other nodes of the overlay. This approach, also adopted in several structured based overlays [19] through auditing mechanisms, or in sensor networks [20], is effective only if the number of malicious nodes is very small with respect to the size of the overlay, typically of $\mathcal{O}(\log |\mathcal{S}|)$.

Recently, Bortnikov *et al.* [16] have proposed a uniform sampling algorithm that tolerates up to a linear number of malicious nodes. Their sampling mechanism exploits the properties offered by min-wise permutations. Specifically, the sampling component is fed with the stream of node ids periodically gossiped by nodes, and outputs the node id whose image value under the randomly chosen permutation is the smallest value ever encountered. Thus eventually, by the property of min-wise permutation, the sampler converges towards a random sample. By limiting the number of requests malicious nodes can periodically issue, their solution requires a single node id to be stored in the local memory. Nevertheless, their solution does not satisfy the ergodicity property as convergence toward a random sample is permanent. It is worth noting that our results complement two previous results [6, 7], in which both papers propose an analysis of the class of uniform and ergodic sampling protocols. Each paper provides a complete analytical proof of a gossip-based protocol that reaches both $\mathcal{U}$ and $\mathcal{E}$. However, in contrast to the present work, adversarial behaviors were not considered.

Finally, taking a completely different approach from the previously mentioned papers, which are based on gossip algorithms or on distance function properties, the techniques presented in [29, 30] rely on social network topologies to guard against Sybil attacks. Both protocols take advantage of the fact that Sybil attacks try to alter the fast mixing property of social networks to defend against these attacks. However, in presence of malicious nodes with a high degree, performance of both protocols degrade drastically.

Note that the analysis presented in this paper is independent from the way the stream of node ids at each node u has been generated. That is, it may result from the propagation of node ids through gossip-based algorithms (namely through push, pull or push-pull mechanisms initiated by u and its neighbors), from the node ids received during random walks initiated at u, or even from the induced churn imposed in structured-based overlays.

5 Characterization of the Uniform and Ergodic Sampling Problem

We start our characterization by showing that the adversary can bias the input stream in such a way that neither uniform nor ergodic properties can be met. This is achieved by flooding the input stream with sufficiently many malicious

node ids. Specifically, Lemma 1 states that for any strategy s, if the number δ_m of non unique malicious node ids that appear in the input stream of node $u \in \mathcal{N}$ during T_s time units exceeds a given threshold then it is impossible for any node in the overlay to equally likely appear as a sample of node u, and this holds forever. Let $(\mathcal{C}_1)$ be a condition on δ_m value

$$\delta_m \leq T_s - (1-k)|\mathcal{S}|. \qquad (\mathcal{C}_1)$$

Condition $(\mathcal{C}_1)$ characterizes the fact that for any honest node $v \in \mathcal{N}$, during the time interval T_s, v has a non-null probability to appear in the input stream. We have

Lemma 1.

$$\neg(\mathcal{C}_1) \Longrightarrow \neg\mathcal{U} \wedge \neg\mathcal{E}.$$

Proof. Let $v \in \mathcal{N}$. Suppose that Condition $(\mathcal{C}_1)$ does not hold, namely it exists an adversarial behavior such that

$$\delta_m > T_s - (1-k)|\mathcal{S}|.$$

In this case, the number of honest node ids in the input stream at v (*i.e.*, $T_s - \delta_m$) is strictly lower than $(1-k)|\mathcal{S}|$, which means formally that

$$T_s - \delta_m < (1-k)|\mathcal{S}|.$$

By assumption (*cf.* Section 3.1) the overlay is populated by $(1-k)|\mathcal{S}|$ honest nodes. Thus, as the adversary manages to flood the input stream at v, there exists at least one node id $u \in \mathcal{S}$ that will never appear in the stream. Therefore, whatever the strategy s, v's sampling component can never output u. Thus,

$$\forall t > T_0,\ \mathbb{P}[u \in S_v(t)] = 0, \qquad (1)$$

which clearly violates Property $\mathcal{U}$.

Equation (1) can be rewritten as $\exists t > T_0, \exists u \in \mathcal{S}, \forall t' > t, \mathbb{P}[u \in S_v(t')] = 0$, which has for consequence that the set of instants t' for which u can be sampled by v is empty. Formally,

$$\mathbb{P}[\{t' | t' > T_0 \wedge u \in S_v(t')\} = \emptyset] = 1,$$

which violates Property $\mathcal{E}$, and completes the proof of the lemma. □

We now assume that Condition $(\mathcal{C}_1)$ holds. The second lemma states that if the size of the sampling memory is large enough, then whatever the constrained adversarial behavior, the sampling component succeeds in exhibiting uniform and ergodic samples. This makes a sufficient condition to solve our problem. Specifically, let $(\mathcal{C}_2)$ be defined as follows

$$|\Gamma| < |\mathcal{S}|. \qquad (\mathcal{C}_2)$$

Condition $(\mathcal{C}_2)$ characterizes the fact that nodes cannot maintain the full knowledge of the population overlay (essentially for scalability reasons). Then,

Lemma 2.

$$(\mathcal{C}_1) \wedge \neg(\mathcal{C}_2) \Longrightarrow \mathcal{U} \wedge \mathcal{E}.$$

Proof. Proof of the lemma is straightforward. By Condition $(\mathcal{C}_1)$, any node $u \in \mathcal{S}$ has a non-null probability to appear in the input stream of any node $v \in \mathcal{N}$. By assumption of the lemma, $|\Gamma_v| \geq |\mathcal{S}|$. Consider the basic strategy s of v's sampling component that consists in storing into Γ_v, any new id read from the input stream. Then eventually, all the node ids will be present into Γ_v, and thus any node u is equally likely to be chosen in Γ_v, which guarantees Property $\mathcal{U}$.

Moreover, v has the possibility to return infinitely often any node id u present in Γ_v. Thus for any time t, the set of instants t', with $t' > t$, such that u is chosen has a zero probability to be empty, which provides Property $\mathcal{E}$ and completes the proof. □

The following Lemma completes the characterization of the problem, specifically:

Lemma 3.

$$(\mathcal{C}_1) \wedge (\mathcal{C}_2) \Longrightarrow \neg(\mathcal{U} \wedge \mathcal{E}).$$

Proof. Suppose that both Conditions $(\mathcal{C}_1)$ and $(\mathcal{C}_2)$ hold. Proving that $\neg(\mathcal{U} \wedge \mathcal{E})$ is equivalent to showing that $(\neg\mathcal{E} \vee \neg\mathcal{U})$ holds, and thus, that $(\mathcal{E} \Longrightarrow \neg\mathcal{U})$ holds. Suppose that $(\mathcal{C}_1) \wedge (\mathcal{C}_2) \wedge \mathcal{E}$ is met, we now show that $\mathcal{U}$ cannot hold.

Consider any node $v \in \mathcal{N}$ (the set of honest nodes) and let $\Gamma_v(t)$ denote the content of v's sampling memory at the instant t. From Condition $(\mathcal{C}_2)$,

$$\forall t' \geq T_0, \exists u \in \mathcal{S}, \quad u \notin \Gamma_v(t'). \tag{2}$$

In particular, Equation (2) is true for $t' = T_s$. Let node $w \in \mathcal{S}$ be such that $w \notin \Gamma_v(T_s)$, then by assumption, Property $\mathcal{E}$ holds. Thus

$$\exists t > T_s, \quad w \notin \Gamma_v(T_s) \wedge w \in \Gamma_v(t). \tag{3}$$

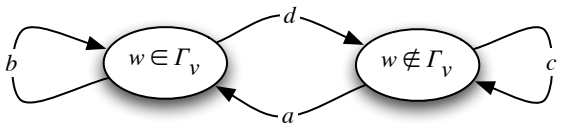

Fig. 2. Markov chain that represents the evolution of w's presence in the sampling memory Γ_v of node $v \in \mathcal{N}$.

The presence of a node id in the local memory of the sampling component can be represented by a Markov chain. Figure 2 depicts the evolution of $w \in \Gamma_v$ as a function of the time. Labels a, b, c and d on the edges represent the probability of transitions from both states. We have $a + c = b + d = 1$. From Equation (3), we have $a > 0$ and thus, $c < 1$. We prove by contradiction that $d > 0$.

Suppose that $d = 0$, then $\forall t'' \geq t,\ w \in \Gamma_v(t'')$, the state $(w \in \Gamma_v)$ is absorbing. Suppose that the overlay contains only two nodes, v and w. By assumption, at least one of the two nodes is honest ($k < 1$). Let us assume that v is honest (the proof is similar for w). Then, by Condition $(\mathcal{C}_2)$, we have $|\Gamma_v| = 1$ (the case $|\Gamma_v| = 0$ trivially leads to impossibility). By assumption, we have $\forall t'' \geq t,\ w \in \Gamma_v(t'')$ and as $|\Gamma_v| = 1$, we also have $\forall t'' \geq t,\ \Gamma_v(t'') = \{w\}$. As a consequence, whatever the strategy s implemented in v's sampling component,

$$\forall t'' \geq t,\ \mathbb{P}[v \in S_v^s(t'')] = 0 \quad \Longrightarrow \quad \mathbb{P}[\{t''|t'' > t \wedge v \in S_v^s(t'')\} = \emptyset] > 0,$$

contradicting $\mathcal{E}$, and thus contradicting the assumption of the lemma. Thus $d > 0$ and, *a fortiori*, $b < 1$, and no state is absorbing.

Suppose now that $\mathcal{U}$ holds. We prove the lemma by contradiction. Consider again the case where the overlay is populated by only two nodes, v and w. As above suppose that node v is honest and that $|\Gamma_v| = 1$. The evolution of the sampling memory at node v can be modeled by a Markov chain as represented in Figure 3. By assumption, $\mathcal{E}$ holds, thus infinitely often, and successively, both v and w appear in Γ_v. Moreover also by assumption, $\mathcal{U}$ holds, that is, $\forall t \geq T_s, \mathbb{P}[w \in S_v^s(t)] = \mathbb{P}[v \in S_v^s(t)] = \frac{1}{2}$. As a consequence, w has the same probability as v to be in Γ_v, whatever the number of times w and v appear in the stream before time T_s.

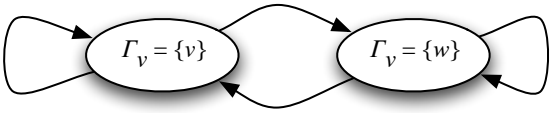

Fig. 3. Markov chain that represents the state of the local memory Γ_v of v.

Suppose now that node w is malicious. By Condition $(\mathcal{C}_1)$, node id w can appear in v's stream no more than $T_s - 1$ times during any sliding window of T_s time units. As $|\Gamma_v| = 1$, a single node id can be stored, and beyond this node id, no other additional information can be stored. We show that whatever the strategies s implemented by v's sampling component, they all lead to a contradiction.

Blind replacement. At any time t, the sampling component reads the first node id in the stream, and stores it in Γ_v in place of the previous one. By construction, any strategy has to select its output among the elements stored in Γ_v, thus the output of the sampling component follows the same probability distribution as the one observed in the stream. As the adversary can flood the stream with up to $T_s - 1$ malicious node ids, this means that property $\mathcal{U}$ cannot be met.

No replacement. Similarly to the blind replacement strategy, node ids are read from the stream, and stored in Γ_v up to time t, where t is the first

time at which a specific node id is read. From time t onwards, this specific node id is kept in Γ_v, independently from the node ids read from the stream after t, leading to an absorbing state of the Markov chain. For instance, this specific node id can be the smallest image value under a random min-wise independent function, such as the min-wise permutation [16]. Clearly, this strategy violates property $\mathcal{E}$.

Probabilistic replacement. This strategy consists in substituting the current node id in Γ_v with the next one read from the stream according to a given probability law. To guarantee that $\forall t, \mathbb{P}[w \in S_v^s(t)] = \mathbb{P}[v \in S_v^s(t)] = \frac{1}{2}$, then either both v and w have an equal probability to appear in the stream or the sampling component must be able to remember the node ids it has seen in the past to guarantee that, at any time t, each node id has the same probability to be chosen as sample. The former case does not hold as by assumption, the adversary can flood the stream with up to $T_s - 1$ malicious ids. Moreover, the latter case is impossible as by assumption $|\Gamma_v = 1|$, and thus a single information can be stored which prevents to store more than a single piece of information (*e.g.*, it is impossible to store both a node id and a counter), therefore property $\mathcal{U}$ cannot hold.

Thus $(\mathcal{C}_1) \wedge (\mathcal{C}_2) \Longrightarrow \neg(\mathcal{U} \wedge \mathcal{E})$, which concludes the proof of the lemma. □

The last lemma reformulates the necessary condition of the problem characterization by combining Lemmata 1 and 3.

Lemma 4.

$$\mathcal{U} \wedge \mathcal{E} \Longrightarrow (\mathcal{C}_1) \wedge \neg(\mathcal{C}_2).$$

Proof. The contrapositive form of writing Lemma 3 is $\mathcal{U} \wedge \mathcal{E} \Longrightarrow \neg((\mathcal{C}_1) \wedge (\mathcal{C}_2))$, and thus, by distributivity,

$$\mathcal{U} \wedge \mathcal{E} \Longrightarrow \neg(\mathcal{C}_1) \vee \neg(\mathcal{C}_2). \tag{4}$$

On the other hand, the contraposition of Lemma 1 leads to $\mathcal{U} \vee \mathcal{E} \Longrightarrow (\mathcal{C}_1)$. As $(\mathcal{U} \wedge \mathcal{E} \Rightarrow \mathcal{U} \vee \mathcal{E})$, we have

$$\mathcal{U} \wedge \mathcal{E} \Longrightarrow (\mathcal{C}_1). \tag{5}$$

By combining Equations 4 and 5, the following holds

$$\mathcal{U} \wedge \mathcal{E} \Longrightarrow (\mathcal{C}_1) \wedge (\neg(\mathcal{C}_1) \vee \neg(\mathcal{C}_2)).$$

Thus,

$$\mathcal{U} \wedge \mathcal{E} \Longrightarrow ((\mathcal{C}_1) \wedge \neg(\mathcal{C}_1)) \vee ((\mathcal{C}_1) \wedge \neg(\mathcal{C}_2)).$$

Due to the principle of contradiction, $(\mathcal{C}_1) \wedge \neg(\mathcal{C}_1)$ cannot hold, leading to

$$\mathcal{U} \wedge \mathcal{E} \Longrightarrow (\mathcal{C}_1) \wedge \neg(\mathcal{C}_2),$$

which completes the proof. □

The Uniform and Ergodic Sampling Problem defined in Sections 2 and 3 is completely characterized by the following theorem:

Theorem 1. $(\mathcal{C}_1) \wedge \neg(\mathcal{C}_2)$ *is a necessary and sufficient condition for Uniform and Ergodic Sampling Problem to hold.*

Proof. This result follows directly from the statements of Lemma 2 and 4. □

6 Conclusion

In this paper, we have investigated the sampling problem of large-scale unstructured peer-to-peer systems in adversarial environments. We have first shown that, if the system cannot provide the means to limit resources of an adversary, then solving either uniform sampling or ergodic sampling is impossible. We have then demonstrated that, if this assumption holds and if all honest nodes in the system have access to a very large memory (in the size of the system) then, the problem becomes trivially solvable but not yet realistic. Unfortunately, we have shown that both conditions are necessary and sufficient ingredients to solve the uniform and ergodic sampling problem in potentially adversarial environments. Clearly, these strong conditions highlight the damage that adversarial behavior can cause in large-scale unstructured systems.

As future work, first we intend to study to which extent the adversary model needs to be weaken to achieve uniform and ergodic sampling in a setting where the nodes themselves have limited ressources (for instance in terms of memory). Second, we plan to investigate an approximate version of the sampling primitive to achieve near uniform and/or near ergodic sampling despite the presence of a strong adversary. Both studies should have a positive impact for applications exhibiting different requirements in terms of resources (*i.e.* memory, computational power and communication complexity) and for settings in which probabilistic guarantees on samples are sufficient.

Acknowledgements

We are very grateful to the anonymous reviewers for their constructive comments that have help us to improve the quality of this paper.

References

1. Jelasity, M., Voulgaris, S., Guerraoui, R., Kermarrec, A.M., van Steen, M.: Gossip-based Peer Sampling. ACM Transaction on Computer System **25**(3) (2007)
2. Bertier, M., Busnel, Y., Kermarrec, A.M.: On Gossip and Populations. In: Proceedings of the 16th International Colloquium on Structural Information and Communication Complexity (SIROCCO). (2009)
3. Karger, D.R., Ruhl, M.: Simple Efficient Load Balancing Algorithms for Peer-to-Peer. In: Proceedings of the 3rd International Workshop on Peer-to-Peer Systems (IPTPS). (2004)
4. Lv, Q., Cao, P., Cohen, E., Li, K., Shenker, S.: Search and Replication in Unstructured Peer-to-Peer Networks. In: Proceedings of the International Conference on Supercomputing (ICS). (2002) 84–95

5. Massoulié, L., Merrer, E.L., Kermarrec, A.M., Ganesh, A.: Peer Counting and Sampling in Overlay Networks: Random Walk Methods. In: Proceedings of the 25th Annual Symposium on Principles of Distributed Computing (PODC), ACM Press (2006) 123–132
6. Busnel, Y., Beraldi, R., Baldoni, R.: A Formal Characterization of Uniform Peer Sampling Based on View Shuffling. In: Proceedings of the International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT), IEEE Computer Society (2009) 360–365
7. Gurevich, M., Keidar, I.: Correcness of Gossip-Based Membership under Message Loss. In: Proceedings of the 28th annual Symposium on Principles of distributed computing (PODC), Calgary, AL, Canada, ACM Press (2009)
8. Bakhshi, R., Gavidia, D., Fokkink, W., van Steen, M.: An Analytical Model of Information Dissemination for a Gossip-based Protocol. Computer Networks **53**(13) (2009) 2288–2303
9. Karp, R., Schindelhauer, C., Shenker, S., Vocking, B.: Randomized Rumor Spreading. In: the 41st Annual Symposium on Foundations of Computer Science (FOCS), IEEE Computer Society (2000) 565
10. Voulgaris, S., Gavidia, D., van Steen, M.: CYCLON: Inexpensive Membership Management for Unstructured P2P Overlays. Journal of Network System Management **13**(2) (2005) 197–217
11. Stutzbach, D., Rejaie, R., Duffield, N., Sen, S., Willinger, W.: On Unbiased Sampling for Unstructured Peer-to-Peer Networks. IEEE/ACM Transactions on Networking **17**(02) (2009) 377–390
12. Bollobás, B.: Random Graphs – 2nd Edition. Cambridge University Press (2001)
13. Zhong, M., Shen, K., Seiferas, J.: Non-uniform Random Membership Management in Peer-to-Peer Networks. In: Proceedings of the 24th Annual Joint Conference of the Computer and Communications Societies (INFOCOM), IEEE Press (2005)
14. Awan, A., Ferreira, R.A., Jagannathan, S., Grama, A.: Distributed Uniform Sampling in Unstructured Peer-to-Peer Networks. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS). (2006)
15. Sit, E., Morris, R.: Security Considerations for Peer-to-Peer Distributed Hash Tables. In: Proceedings of the 1rst International Workshop on Peer-to-Peer Systems (IPTPS), Springer-Verlag (2002) 261–269
16. Bortnikov, E., Gurevich, M., Keidar, I., Kliot, G., Shraer, A.: Brahms: Byzantine Resilient Random Membership Sampling. Computer Networks **53** (2009) 2340–2359 A former version appeared in the 27th ACM Symposium on Principles of Distributed Computing (PODC), 2008.
17. Awerbuch, B., Scheideler, C.: Group Spreading: A Protocol for Provably Secure Distributed Name Service. In: Proceedings of the 31rst International Colloquium on Automata, Languages and Programming (ICALP). (2004) 183–195
18. Jesi, G.P., Montresor, A., van Steen, M.: Secure Peer Sampling. Computer Networks (To appear) (2010)
19. Singh, A., Ngan, T.W., Druschel, P., Wallach, D.S.: Eclipse Attacks on Overlay Networks: Threats and Defenses. In: Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM). (2006)
20. Liu, D., Ning, P., Du, W.: Detecting Malicious Beacon Nodes for Secure Location Discovery in Wireless Sensor Networks. In: Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS). (2005)
21. Douceur, J., Donath, J.S.: The Sybil Attack. In: Proceedings of the 1rst International Workshop on Peer-to-Peer Systems (IPTPS). (2002) 251–260

22. Awerbuch, B., Scheideler, C.: Towards a Scalable and Robust Overlay Network. In: Proceedings of the 6th International Workshop on Peer-to-Peer Systems (IPTPS). (2007)
23. Anceaume, E., Brasileiro, F.V., Ludinard, R., Sericola, B., Tronel, F.: Analytical Study of Adversarial Strategies in Cluster-based Overlays. In: Proceedings of the International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT). (2009) 293–298
24. Castro, M., Druschel, P., Ganesh, A., Rowstron, A., Wallach, D.S.: Secure Routing for Structured Peer-to-peer Overlay Networks. In: Proceedings of the 5th Symposium on Operating Systems Design and Implementation (OSDI), ACM (2002) 299–314
25. Hildrum, K., Kubiatowicz, J.: Asymptotically Efficient Approaches to Fault-tolerance in Peer-to-Peer Networks. In: Proceedings of the International Symposium on Distributed Computing (DISC). (2003) 321–336
26. Fiat, A., Saia, J., Young, M.: Making Chord Robust to Byzantine Attacks. In: Proceedings of the Annual European Symposium on Algorithms. (2005) 803–814
27. Anceaume, E., Brasileiro, F., Ludinard, R., Ravoaja, A.: PeerCube: an Hypercube-based P2P Overlay Robust against Collusion and Churn. In: Proceedings of the IEEE International Conference on Self-Adaptive and Self-Organizing Systems. (2008) 15–24
28. Condie, T., Kacholia, V., Sank, S., Hellerstein, J.M., Maniatis, P.: Induced Churn as Shelter from Routing-Table Poisoning. In: Proceedings of the International Network and Distributed System Security Symposium (NDSS). (2006)
29. Yu, H., Kaminsky, M., Gibbons, P.B., Flaxman, A.: SybilGuard: Defending against Sybil Attacks via Social Networks. In: Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM). (2006) 267–278
30. Yu, H., Gibbons, P.B., Kaminsky, M., Xiao, F.: SybilLimit: A Near-Optimal Social Network Defense against Sybil Attacks. In: Proceedings of the IEEE Symposium on Security and Privacy (SP). (2008) 3–17