Analytical Study of Adversarial Strategies in Cluster-based Overlays

E. Anceaume*, R. Ludinard†, B. Sericola†, F. Tronel‡ and F. Brasileiro§
* *CNRS/IRISA, France*
† *INRIA Rennes Bretagne-Atlantique, France*
‡ *Supelec, France*
§ *Universidade Federal de Campina Grande, LSD Laboratory, Brazil*

***Abstract*—Scheideler has shown that peer-to-peer overlays networks can only survive Byzantine attacks if malicious nodes are not able to predict what will be the topology of the network for a given sequence of join and leave operations. In this paper we investigate adversarial strategies by following specific protocols. Our analysis demonstrates first that an adversary can very quickly subvert DHT-based overlays by simply never triggering leave operations. We then show that when all nodes (honest and malicious ones) are imposed on a limited lifetime, the system eventually reaches a stationary regime where the ratio of polluted clusters is bounded, independently from the initial amount of corruption in the system.**

***Keywords*-Clusterized P2P Overlays, Adversary, Churn, Collusion, Markov chain.**

I. Introduction

The adoption of peer-to-peer overlay networks as a building block for architecting Internet scale systems has raised the attention of making these overlays resilient not only to benign crashes, but also to more malicious failure models for the peers [1], [2], [3]. As a result, Byzantine-resilient overlays have been proposed (e.g., [4], [5], [6]). Awerbuch and Scheideler [7] have shown that peer-to-peer overlays networks can only survive Byzantine attacks if malicious nodes are not able to predict what will be the topology of the network for a given sequence of join and leave operations. A prerequisite for this condition to hold is to guarantee that nodes identifiers randomness is continuously preserved. However targeted join/leave attacks may quickly endanger the relevance of such an assumption [8]. Inducing churn has been shown to be the other fundamental ingredient to preserve randomness. Several strategies based on these principles have been proposed. Most of them are based on locally induced churn. However either they have been proven incorrect or they involve a too high level of complexity to be practically acceptable [7]. The other ones, based on globally induced churn, enforce limited lifetime for each node in the system. However, these solutions keep the system in an unnecessary hyper-activity, and thus need to impose strict restrictions on nodes joining rate which clearly limit their applicability to open systems.

In this paper we propose to leverage the power of clustering to design a practically usable solution that preserves randomness under an adaptive adversary. Our solution relies on the clusterized version of peer-to-peer overlays combined with a mechanism that allows the enforcement of limited nodes lifetime [9]. Clusterized versions of structured-based overlays are such that nodes at the vertices of the graph are substituted by clusters of nodes. Cluster-based overlays have revealed to be well adapted for efficiently reducing the impact of churn on the system and/ or in greatly reducing the damage caused by failures—assuming that failures assumptions hold anywhere and at any time in the system [6], [10], [4]. We first investigate adversarial strategies by following specific protocols. Our analysis demonstrates that an adversary can very quickly subvert cluster-based overlays by simply never triggering leave operations. We then show that when all nodes are imposed on a limited lifetime, the system eventually reaches a stationary regime where the ratio of polluted clusters is bounded.

II. Cluster-based DHT Overlays in a Nutshell

In this section we first present the common features of cluster-based overlays and then present the different join/leave strategies whose long term behaviors are analysed in Section III.

Clusterized versions of structured-based overlays are such that clusters of nodes substitute nodes at the vertices of the graph. Nodes are uniquely identified with some m-bit string randomly chosen from an ID-space. Identifiers (IDs) are derived by using standard collision-resistant one-way hash functions (e.g. MD5, SHA1). Each graph vertex is composed of a set of nodes self-organised within a cluster according to some distance metrics (e.g., logical or geographical). Clusters in the system are uniquely labelled. Size of each cluster is lower (resp. upper) bounded. The lower bound, named S_{min} in the following, usually satisfies some constraint based on the assumed failure model. For instance $S_{min} \geq 4$ allows Byzantine tolerant agreement protocols to be run among these S_{min} nodes [11]. The upper bound, that we call S_{max}, is typically in $\mathcal{O}(logN)$, where N is the current number of nodes in the system, to meet scalability requirements. When a cluster size reaches these bounds, cluster-based overlays react by respectively splitting that cluster into two smallest clusters or by merging it with its closest cluster neighbours. Finally for most of the cluster-based overlays, operations (`join`, `leave`, `merge`,

and `split`) are poly-logarithmic in the number of nodes in the system.

In the present work we assume that at cluster level, nodes are organised as core and spare members. Members of the core set are primarily responsible for handling messages routing and clusters operations. Management of the core set is such that its size is maintained to constant S_{min}. Spare members are the complement number of nodes in the cluster. In contrast to core members, they are not involved in any of the overlay operations. Rationale of this classification is two-fold: first it allows to introduce the unpredictability required to deal with Byzantine attacks through a randomized core set generation algorithm. Second it limits the management overhead caused by the natural churn present in typical overlay networks through the spare set management.

Specifically we consider the following join and leave operations:

- `join`(p): when peer p `joins` a cluster, it joins it as a spare member.
- `leave`(p): When a peer p `leaves` a cluster either p belongs to the spare set or to the core set. In the former case, core members simply update their spare view to reflect p's departure, while in the latter case, the core view maintenance procedure is triggered. Two different maintenance policies are implemented. The first one, referred to as *policy 1* in the following, simply consists in replacing the left core member by one randomly chosen spare member. The second one, referred to as *policy 2*, consists in refreshing the whole core set by choosing S_{min} random peers within the cluster.

For space reasons we do not give any detail regarding the localization of a cluster nor its creation/split/merge process. Description of these operations are not necessary for the understanding of our work. The interested reader is invited to read them in the original papers (e.g. [6], [10], [4]).

III. Modelling the Adversarial Strategy as a Protocol

In this section, we investigate the two previously described policies (policies 1 and 2). We model adversarial behavior by focusing on specific protocols. Both protocols intend to prevent the adversary from elaborating deterministic strategies to win. These protocols are played in the following context. There is a potentially infinite number of balls in a bag, with a proportion μ of red balls and a proportion $1-\mu$ of white balls, μ being a constant in $(0,1)$. White (resp. red) balls are indistinguishable. Red balls are owned by the adversary. In addition to the bag, there are two urns, named $\mathcal{C}$ and $\mathcal{S}$. Initially, $c+s$ balls are drawn from the bag such that c of them are thrown into urn $\mathcal{C}$, and the other s ones are thrown into urn $\mathcal{S}$. We denote by C_r (resp. S_r) the number of red balls in $\mathcal{C}$ (resp. $\mathcal{S}$). It is easily checked that C_r and S_r are independent and have a binomial distribution, i.e. for

```
/* First protocol */
/* stage 1 */
   draw ball b_0 from C ∪ S
/* stage 2 */
   if b_0 was in S then throw b_0 into the bag,
      draw ball b_2 from the bag, and throw it into S
   else
      throw b_0 into the bag
      draw ball b_1 from S and throw it into C
      draw ball b_2 from the bag and throw it into S

/* Second protocol */
/* stage 1 */
   draw ball b_0 from C ∪ S
/* stage 2 */
   if b_0 was in S then throw b_0 into the bag,
      draw ball b_2 from the bag, and throw it into S
   else
      throw b_0 into the bag
      draw c balls from S ∪ C and throw these c balls into C
      draw one ball b_2 from the bag and throw it in S
```

Figure 1: Rule of the first and second protocol.

$x = 0, \ldots, c$ and $y = 0, \ldots, s$, we have

$$\begin{aligned} \alpha(x,y) &= \mathbb{P}\{C_r = x, S_r = y\} = \mathbb{P}\{C_r = x\}\mathbb{P}\{S_r = y\} \\ &= \binom{c}{x}\mu^x(1-\mu)^{c-x}\binom{s}{y}\mu^y(1-\mu)^{s-y}. \end{aligned} \quad (1)$$

This joint distribution represents the initial distribution of the process detailed below. Each protocol is a succession of rounds $r_1, r_2, \ldots$ during which the protocol rules described in Figure 1 are applied. Rules are oblivious to the colour of the balls, that is, they cannot distinguish between the white and the red balls.

The goal of the adversary is to get a quorum Q of red balls in both urns $\mathcal{C}$ and $\mathcal{S}$ so that the number of red balls in $\mathcal{C}$ is bound to continuously exceed $\lfloor (c-1)/3 \rfloor$ [11]. The adversary may at any time inspect both urns and bag to elaborate adversarial strategies to win the protocol. In particular it may not follow the rule of the protocols by preventing its red balls from being extracted from both urns. Specifically, at stage 1 of both protocols, if the drawn ball b_0 is red then the adversary puts back the ball into the urn from which it has been drawn. Stage 2 is not applied, and a new round is triggered. Clearly this strategy ensures that the number of red balls in $\mathcal{C} \cup \mathcal{S}$ is monotonically non decreasing.

We model the effects of these rounds using a homogeneous Markov chain denoted by $X = \{X_n, n \geq 0\}$ representing the evolution of the number of red balls in both urns $\mathcal{C}$ and $\mathcal{S}$. More formally, the state space S of X is defined by $S = \{(x,y) \mid 0 \leq x \leq c,\ 0 \leq y \leq s\}$, and, for $n \geq 1$, the event $X_n = (x,y)$ means that, after the n-th transition or n-th round, the number of red balls in urn $\mathcal{C}$ is equal to x and the number of red balls in urn $\mathcal{S}$ is equal to y. The transition probability matrix P of X depends on the rule of the given protocol and on the adversarial behaviours. This matrix is detailed in each of the following subsections. In

all the cases, the initial state X_0 is given by $X_0 = (C_r, S_r)$ and its probability distribution is given by relation (1).

We define a state as *polluted* if in this state urn $\mathcal{C}$ contains more than $\lfloor (c-1)/3 \rfloor$ balls. In the following, we denote by c' the value $\lfloor (c-1)/3 \rfloor$. Conversely, a state that is not polluted is said to be *safe*. The subset of safe states, denoted by A, is defined as: $A = \{(x,y) \mid 0 \leq x \leq c',\ 0 \leq y \leq s\}$, while the set of polluted states, denoted by B, is the subset $S - A$, i.e. $B = \{(x,y) \mid c'+1 \leq x \leq c,\ 0 \leq y \leq s\}$. We partition matrix P in a manner conformant to the decomposition of $S = A \cup B$, by writing

$$P = \begin{pmatrix} P_A & P_{AB} \\ P_{BA} & P_B \end{pmatrix},$$

where P_A (resp. P_B) is the sub-matrix of dimension $|A| \times |A|$ (resp. $|B| \times |B|$), containing the transitions between states of A (resp. B). In the same way, P_{AB} (resp. P_{BA}) is the sub-matrix of dimension $|A| \times |B|$ (resp. $|B| \times |A|$), containing the transitions from states of A (resp. B) to states of B (resp. A). We also partition the initial probability distribution α according to the decomposition $S = A \cup B$, by writing $\alpha = (\alpha_A \quad \alpha_B)$, where sub-vector α_A (resp. α_B) contains the initial probabilities of states of A (resp. B).

A. First protocol

We can easily derive the transition probability matrix P of the Markov chain X chain associated to this protocol. For all $x \in \{0, \ldots, c\}$ and for all $y \in \{0, \ldots, s\}$, we have

$$\begin{aligned}
p_{(x,y),(x,y)} &= \left(\frac{c}{c+s}\right)\left(\frac{x}{c} + \left(\frac{c-x}{c}\right)\left(\frac{s-y}{s}\right)(1-\mu)\right) \\
&\quad + \left(\frac{s}{s+c}\right)\left(\frac{y}{s}\mu + 1 - \mu\right) \\
p_{(x,y),(x,y+1)} &= \left(\left(\frac{c-x}{c+s}\right) + \left(\frac{s-y}{s+c}\right)\right)\mu \quad \text{for } y \leq s-1 \\
p_{(x,y),(x+1,y-1)} &= \left(\frac{c-x}{c+s}\right)\frac{y}{s}(1-\mu) \quad \text{for } x \leq c-1 \text{ and } y \geq 1 \\
p_{(x,y),(x+1,y)} &= \left(\frac{c-x}{c+s}\right)\frac{y}{s}\mu \quad \text{for } x \leq c-1.
\end{aligned}$$

In all other cases, transition probabilities are null. Clearly, the adversary wins the protocol when the process X reaches the subset of states B from which it cannot exit. Thus quorum $Q = \{(x,y) \mid (x,y) \in B\}$ with B the set of polluted states. By the rules of the protocol, one can never escape from these states to switch to safe states since the number of red balls in $\mathcal{C}$ is non decreasing. Thus there is a finite random time T after which the process X is absorbed within B. Thus we have $P_{BA} = 0$. The Markov chain X is reducible and the states of A are transient, which means that matrix $I - P_A$ is invertible, where I is the identity matrix of the right dimension which is $|A|$ here. Specifically T, the time needed to reach subset B, is defined as $T = \inf\{n \geq 0 \mid X_n \in B\}$. The cumulative distribution function of T is easily derived as

$$\mathbb{P}\{T \leq k\} = 1 - \alpha_A (P_A)^k \mathbb{1}, \tag{2}$$

where $\mathbb{1}$ is the column vector with all components equal to 1. The expectation of T is given by

$$E(T) = \alpha_A (I - P_A)^{-1} \mathbb{1}, \tag{3}$$

B. Second protocol

By proceeding similarly as above, we can derive the transitions of process X associated to the second protocol. For space reasons, we omit their description from the paper.

Briefly, when the protocol starts in state (x,y) at round r, it remains in state (x,y) during the round if either ball b_0 is red or b_0 is white, and has been drawn from $\mathcal{S}$, and b_2 is white. It changes to state $(x, y+1)$ if b_0 is white, it has been drawn from $\mathcal{S}$, and b_2 is red. Finally the protocol switches to state $(k, x+y-k+\ell)$, where k is an integer $k = 0, \ldots, c'$ and $\ell = 0$ or 1 if b_0 is white, it has been drawn from $\mathcal{C}$, and the renewal process leads to the choice of k red balls. For all $x \in \{0, \ldots, c\}$ and $y \in \{0, \ldots, s\}$, we have

$$\begin{aligned}
p_{(x,y),(x,y)} &= \frac{x}{c+s}\mu q(x, x+y-1) \\
&\quad + \frac{c-x}{c+s}(1-\mu) q(x, x+y) \\
&\quad + \left(\frac{s}{c+s}\right)\left(\frac{s-y}{s}(1-\mu) + \frac{y}{s}\mu\right) \\
p_{(x,y),(x,y+1)} &= \frac{x}{c+s}\mu q(x, x+y) \\
&\quad + \left(\frac{s}{c+s}\right)\left(\frac{s-x}{c}\mu\right) \text{ for } y \leq s-1 \\
p_{(x,y),(x,y-1)} &= \frac{x}{c+s}(1-\mu) q(x, x+y-1) \\
&\quad + \left(\frac{s}{c+s}\right)\left(\frac{s-x}{c}\mu\right) \text{ for } y \geq 1 \\
p_{(x,y),(k,x+y-k)} &= \left(\frac{c-x}{c+s}\right)(1-\mu)\, q(k, x+y) \\
&\quad \text{for } \max(0, x+y-s) \leq k \leq \min(c, x+y) \\
&\quad \text{and } k \neq x \\
p_{(x,y),(k,x+y-k+1)} &= \left(\frac{c}{c+s}\right)\left(\frac{c-x}{c}\right)\mu q(k, x+y) \\
&\quad \text{for } k \geq \max(0, x+y+1-s) \\
&\quad \text{and } k \leq \min(c, x+y+1) \\
&\quad \text{and } k \neq x
\end{aligned}$$

where

$$q(x, x+y) = \frac{\binom{x+y}{x}\binom{c+s-1-(x+y)}{c-x}}{\binom{c+s-1}{c}}$$

is the probability of getting x red balls when c balls are drawn, without replacement, in an urn containing $x+y$ red balls and $c+s-1-(x+y)$ white balls, referred to as the hypergeometric distribution. In all other cases, transition probabilities are null.

In contrast to the first protocol, this protocol alternates between safe and polluted states. After a random number of these alternations the process ends by entering a set of closed polluted states. Indeed, by the rule of the protocol, one can escape finitely often from polluted state (x,y) to switch back to a safe state as long as (x,y) satisfies $c'+1 \leq x+y \leq s+c'$ (there are still sufficiently many white balls in both $\mathcal{C}$ and

$\mathcal{S}$ so as to successfully withdrawing c balls such that $\mathcal{C}$ can be reverted to a safe state). However, there is a time T_D when state (x, y), with $x + y \geq s + c' + 1$, is entered. From T_D onwards, going back to safe states is impossible. Thus at time T_D the adversary wins the protocol. Hence an interesting metrics to be evaluated is the total time spent by the process in safe states before being definitely absorbed in polluted states. Formally, we need to decompose the set B of polluted states into two subsets C and D defined by $C = \{(x, y) \mid c' + 1 \leq x + y \leq s + c',\ c' + 1 \leq x \leq c,\ 0 \leq y \leq s\}$, and $D = \{(x, y) \mid x + y \geq s + c' + 1,\ 0 \leq y \leq s\}$. Subsets A and C are transient and subset D is a closed subset. We partition matrix P and initial probability vector α accordingly.

We are interested in the random variable T_A which counts the total time spent in subset A before reaching subset D. Following the result obtained in [12], we have, for every $k \geq 0$,

$$\mathbb{P}\{T_A \leq k\} = 1 - vG^k\mathbb{1}, \tag{4}$$

where $v = \alpha_A + \alpha_C(I - P_C)^{-1}P_{CA}$ and $G = P_A + P_{AC}(I - P_C)^{-1}P_{CA}$. The expected total time spent in A is given by

$$E(T_A) = v(I - G)^{-1}\mathbb{1}. \tag{5}$$

Figure 2(a) compares the expectation of the time spent in safe states for both protocols. In accordance with the intuition, increasing the size of the urns augments the expected time spent in safe states of both protocols, i.e., $E(T)$ and $E(T_A)$, independently from the ratio of red balls in the bag. Similarly, for a given cluster size, increasing the ratio of red balls in the bag drastically decreases both $E(T)$ and $E(T_A)$. However surprisingly enough, increasing the level of randomness (protocol 2 vs. protocol 1) does not increase the resilience to the adversary behavior since the first protocol always overpasses the second one in expectation. It is even more true when $\mathcal{S}$ size is large with respect to $\mathcal{C}$ one. The intuition behind this fact is as follows: when $\mathcal{S}$ size is equal to 1, both protocols are equivalent as illustrated in Figure 2(a) for $s = 1$. Now, consider the case where the size of $\mathcal{S}$ is large with respect to $\mathcal{C}$ one. First of all, note that the probability to draw a ball from $\mathcal{S}$ tends to 1, and because the adversary never withdraw its red balls from any urns, the ratio of red balls within $\mathcal{S}$ is monotonically non decreasing. Hence, the ratio of red balls in $\mathcal{S}$ tends also to 1. With small probability, a ball from $\mathcal{C}$ is drawn. In the first protocol it is replaced with high probability by a red ball drawn from $\mathcal{S}$. Hence to reach a polluted state, at least c' white balls have to be replaced by red ones. While in the second protocol with high probability, the renewal of $\mathcal{C}$ reaches a polluted state in a single step. From this crude reasoning we can derive that the ratio of $E(T)$ over $E(T_A)$ tends to c'.

IV. Constraining the adversary

Our next step is to evaluate the benefit of constraining the adversary by limiting the sojourn time of its balls in both urns, so that randomness among red and white balls is continuously preserved. In the model we propose, we assume that the adversary cannot prevent red balls from being withdrawn for both urns.

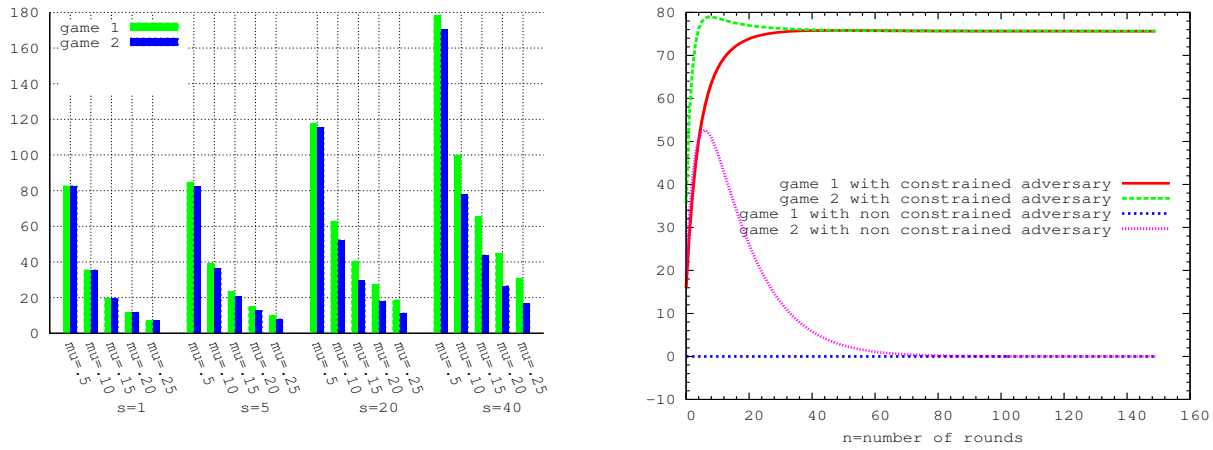
By proceeding exactly as for the first protocol, we can easily derive the transition probability matrix P for both protocols. For all $x \in \{0, \ldots, c\}$ and $y \in \{0, \ldots, s\}$, the entries of P are given, for the first protocol, by

$$\begin{aligned}
p_{(x,y),(x,y)} &= \frac{xy + (c(s-y) - xs)(1-\mu)}{(c+s)s}) \\
&\quad + \frac{y\mu + (s-y)(1-\mu)}{c+s} \\
p_{(x,y),(x,y-1)} &= \frac{(x+s)y}{(c+s)s}(1-\mu) \text{ for } y \geq 1 \\
p_{(x,y),(x,y+1)} &= \left(\frac{c-x+s}{c+s}\right)\left(\frac{s-y}{s}\right)\mu \text{ for } y \leq s-1 \\
p_{(x,y),(x+1,y-1)} &= \frac{(c-x)y}{(c+s)s}(1-\mu) \text{ for } x \leq c-1 \text{ and } y \geq 1 \\
p_{(x,y),(x+1,y)} &= \frac{(c-x)y}{(c+s)s}\mu \text{ for } x \leq c-1 \\
p_{(x,y),(x-1,y)} &= \frac{x(s-y)}{(c+s)s}(1-\mu) \text{ for } x \geq 1 \\
p_{(x,y),(x-1,y+1)} &= \frac{x(s-y)}{(c+s)s}\mu \text{ for } x \geq 1 \text{ and } y \leq s-1.
\end{aligned}$$

In all other cases, transition probabilities are null. Similarly for second protocol , for all $x \in \{0, \ldots, c\}$ and $y \in \{0, \ldots, s\}$, we have

$$\begin{aligned}
p_{(x,y),(x,y)} &= \frac{xq(x, x+y-1)\mu}{c+s} \\
&\quad + \frac{(c-x)q(x, x+y)(1-\mu)}{c+s} \\
&\quad + \frac{y\mu + (s-y)(1-\mu)}{c+s} \\
p_{(x,y),(x,y-1)} &= \frac{x}{c+s}q(x, x+y-1)(1-\mu) \\
&\quad + \frac{y}{c+s}(1-\mu) \text{ for } y \geq 1 \\
p_{(x,y),(x,y+1)} &= \frac{c-x}{c+s}q(x, x+y)\mu + \frac{s-y}{c+s}\mu \text{ for } y \leq s-1 \\
p_{(x,y),(k,x+y-k-1)} &= \frac{x}{c+s}q(k, x+y-1)(1-\mu) \\
&\quad \text{for } k \geq \max(0, x+y-1-s) \\
&\quad \text{and } k \leq \min(c, x+y-1) \\
&\quad \text{and } k \neq x \\
p_{(x,y),(k,x+y-k)} &= \frac{x}{c+s}q(k, x+y-1)\mu \\
&\quad + \frac{c-x}{c+s}q(k, x+y)(1-\mu) \\
&\quad \text{for } k \geq \max(0, x+y-s) \\
&\quad \text{and } k \leq \min(c, x+y-1) \\
&\quad \text{and } k \neq x \\
p_{(x,y),(k,x+y-k+1)} &= \frac{c-x}{c+s}q(k, x+y)\mu \\
&\quad \text{for } k \geq \max(0, x+y+1-s) \\
&\quad \text{and } k \leq \min(c, x+y) \\
&\quad \text{and } k \neq x,
\end{aligned}$$

where we set $q(u, v) = 0$ when $u > v$. In all other cases, transition probabilities are null. It is not difficult to see that



(a) (b)

Figure 2: (a) Expectation of the number of rounds spent in safe states for protocols 1 and 2 function of $\mathcal{S}$ size and the ratio of malicious nodes μ as resp. given by relations (3) and (5). (b) Mean number of safe clusters $E(N_n)$ (relation (8)) in function of the rounds number n for both protocols and both kind of adversaries. There are l=100 clusters, and the ratio of red balls in the bag is equal to .25 and $c = 7$. Note that the initial number of safe clusters is equal to 16.

none of the protocols exhibit an absorbing class of states (i.e., both protocols never ends). We have $P_{BA} \neq 0$ and the process X is irreducible and aperiodic since at least one state has a transition to itself. The distribution of the time T needed to reach subset B is given, for every $k \geq 0$, by

$$\mathbb{P}\{T \leq k\} = 1 - \alpha_A (P_A)^k \mathbb{1}. \tag{6}$$

We denote by π the stationary distribution of the Markov chain X. The row vector π is thus the solution to the linear system

$$\pi = \pi P \text{ and } \pi \mathbb{1} = 1.$$

As we did for row vector α, we partition π according to the decomposition $S = A \cup B$, by writing $\pi = (\pi_A \ \ \pi_B)$, where sub-vector π_A (resp. π_B) contains the stationary probabilities of states of A (resp. B).

Theorem *1:* For both protocols 1 and 2, the stationary distribution π is equal to α, i.e. for all $x = 0, \ldots, c$ and $y = 0, \ldots, s$, we have

$$\lim_{n \longrightarrow \infty} \mathbb{P}\{X_n = (x, y)\} = \alpha(x, y) \text{ as given by relation (1).}$$

Proof: For space reasons, we omit the proof of the theorem. The reader is invited to read it in [9]. ■

Theorem 1 is quite interesting as it shows that the stationary distribution π is exactly the same for both protocols, and that this distribution is equal to the initial distribution α. At a first glance, we could guess that this phenomenon is due to the fact that the Markov chain X is the tensor product of two independent Markov chains. However, this is clearly not the case as the behavior of red balls in $\mathcal{C}$ depends on the behavior of red balls in $\mathcal{S}$. This holds for both protocols. The stationary availability of the system defined by the long run probability to be in safe states is denoted by P_{safe} and is given by

$$P_{\text{safe}} = \pi_A \mathbb{1} = \sum_{x=0}^{c'} \binom{c}{x} \mu^x (1 - \mu)^{c-x}.$$

This probability can also be interpreted as the long run proportion of time spent in safe states. Note that the stationary distribution does not depend on the size of $\mathcal{S}$.

We conclude this paper by showing that by inducing global churn, we can preserve the safety of the system. We consider that we have ℓ identical and independent Markov chains $X^{(1)}, \ldots, X^{(\ell)}$ on the same state space $S = A \cup \{S \setminus A\}$, with initial probability distribution β and transition probability matrix P. Each Markov chain models a particular cluster of nodes and, for $n \geq 0$, N_n represents the number of safe clusters after the n-th round, i.e. the number of Markov chains being in subset A after the n-th transition has been triggered, defined by

$$N_n = \sum_{j=1}^{\ell} 1_{\{X_n^{(j)} \in A\}}.$$

The ℓ Markov chains being identical and independent, N_n has a binomial distribution, that is, for $k = 0, \ldots, \ell$

$$\begin{aligned} \mathbb{P}\{N_n = k\} &= \binom{\ell}{k} \left(\mathbb{P}\{X_n^{(1)} \in A\}\right)^k \\ &\quad \times \left(1 - \mathbb{P}\{X_n^{(1)} \in A\}\right)^{\ell - k} \\ &= \binom{\ell}{k} \left(\beta P^n \mathbb{1}_A\right)^k \left(1 - \beta P^n \mathbb{1}_A\right)^{\ell - k} \end{aligned} \tag{7}$$

and

$$E(N_n) = \ell\beta P^n \mathbb{1}_A, \tag{8}$$

where $\mathbb{1}_A$ is the column vector with the i-th entry equal to 1 if $i \in A$ and equal to 0 otherwise. If N denotes the stationary number of safe clusters, we have, for

$$\begin{aligned} E(N) &= \ell\pi_A\mathbb{1} && \text{for a constrained adversary} \\ &= 0 && \text{for a non constrained adversary} \end{aligned}$$

These results are illustrated in Figure 2(b). We can observe that with a constrained adversary, the ratio of safe clusters tends to the same limit for both protocols, whatever the amount of initially safe clusters (less than a 1/4), while with a non constrained adversary eventually all the clusters get polluted.

V. Conclusion

In this paper, we have proposed a mechanism that enables the enforcement of limited nodes lifetime compliant with DHT-based overlays specificities. We have investigated the long run behavior of several adversarial strategies. Our analysis has demonstrated that an adversary can easily subvert a cluster-based overlay by simply never triggering leave operations. We have shown that when nodes have to regularly leave the system, a stationary regime where the ratio of malicious nodes is bounded is eventually reached.

For future work, we plan to implement this limited node lifetime mechanism in the cluster-based DHT overlay PeerCube [6] to study its impact on the induced churn and its management overhead. We are convinced that this additional churn will be efficiently amortised thanks to the organisation of nodes in core and spare sets.

References

[1] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach, "Secure routing for structured peer-to-peer overlay networks," in *Proceedings of the Symposium on Operating Systems Design and Implementation*, 2002.

[2] A. Singh, T. Ngan, P. Drushel, and D. Wallach, "Eclipse attacks on overlay networks: Threats and defenses," in *Proceedings of the Conference on Computer Communications*, 2006.

[3] E. Sit and R. Morris, "Security considerations for peer-to-peer distributed hash tables," in *Proceedings of the Int'l Workshop on Peer-to-Peer Systems*, 2002.

[4] A. Fiat, J. Saia, and M. Young, "Making chord robust to byzantine attacks," in *Proceedings of the Annual European Symposium on Algorithms*, 2005.

[5] I. Baumgart and S. Mies, "S/kademlia: A practicable approach towards secure key-based routing," in *Procs of the Int'l Conference on Parallel and Distributed Systems'*, 2007.

[6] E. Anceaume, F. Brasileiro, R. Ludinard, and A. Ravoaja, "Peercube: an hypercube-based p2p overlay robust against collusion and churn," in *Procs of the IEEE Int'l Conference on Self-Adaptive and Self-Organizing Systems*, 2008.

[7] B. Awerbuch and C. Scheideler, "Towards scalable and robust overay networks," in *Proceedings of the Int'l Workshop on Peer-to-Peer Systems*, 2007.

[8] ——, "Group spreading: A protocol for provably secure distributed name service," in *Procs of the Int'l Colloquium on Automata, Languages and Programming*, 2004.

[9] E. Anceaume, F. Brasileiro, R. Ludinard, B. Sericola, and F. Tronel, "Analytical study of adversarial strategies in cluster-based overlays," http://hal.archives-ouvertes.fr/hal-00408871/en.

[10] T. Locher, S. Schmid, and R. Wattenhofer, "equus: A provably robust and locality-aware peer-to-peer system," in *Proceedings of the Int'l Conference on Peer-to-Peer Computing*, 2006.

[11] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, 1982.

[12] B. Sericola, "Closed form solution for the distribution of the total time spent in a subset of states of a Markov process during a finite observation period," *Journal of Applied Probability*, vol. 27, 1990.