On the Impossibility of Efficient Self-Stabilization in Virtual Overlays with Churn

Stefanie Roos, Thorsten Strufe
TU Dresden, Privacy and IT Security
{stefanie.roos,thorsten.strufe}@tu-dresden.de

***Abstract*—Virtual overlays generate topologies for greedy routing, like rings or hypercubes, on connectivity restricted networks. They have been proposed to achieve efficient content discovery in the Darknet mode of Freenet, for instance, which provides a private and secure communication platform for dissidents and whistle-blowers. Virtual overlays create tunnels between nodes with neighboring addresses in the topology. The routing performance hence is directly related to the length of the tunnels, which have to be set up and maintained at the cost of communication overhead in the absence of an underlying routing protocol.**

In this paper, we show the impossibility to efficiently maintain sufficiently short tunnels. Specifically, we prove that in a dynamic network either the maintenance or the routing eventually exceeds polylog cost in the number of participants. Our simulations additionally show that the length of the tunnels increases fast if standard maintenance protocols are applied. Thus, we show that virtual overlays can only offer efficient routing at the price of high maintenance costs.

I. Introduction

Virtual overlays have been proposed to improve the routing performance in dynamic and connectivity restricted environments [1]–[4]. They create structures allowing for greedy routing in networks for which the distribution of topological information, as it is used for routing in fixed networks like the Internet, is impossible or undesired.

Darknets, restricting connections to devices of users sharing a real-world trust relationship to protect user privacy, such as Freenet [5] or GNUnet [6] are but one example for which virtual overlays have been proposed, primarily to enhance their unsatisfying routing performance[1]. These systems aim at providing privacy-preserving communication platforms for dissidents, whistle-blowers, or for privacy-aware social networking in general. The concept of virtual overlays has also found prominent application in wireless sensor networks, in which connectivity is restricted by the radio range, and moving nodes frequently discover new and disconnect from old neighbors.

Given arbitrary, yet static node addresses (also called identifier or IDs), virtual overlays establish paths through the underlying network between nodes that, according to their identifiers, share a connection in the routing structure. These tunnels are configured locally: all nodes store the target, predecessor, and successor for all tunnels they participate in. Each tunnel represents an overlay hop of the provisioned end-to-end routing, and the entirety of tunnels represents the virtual, greedily routable overlay.

Establishing and maintaining tunnels for virtual overlays comes at a cost: the neighboring nodes need to be identified and the tunnels then either set up from scratch or created by augmenting existing tunnels that span parts of the intended path. The length of the tunnels then determines the overall performance of the end-to-end routing.

Virtual overlays need to self-stabilize: Churn, the joining or leaving of nodes, or other topological changes require the frequent update of disrupted tunnels as well as the establishment of additional tunnels from and to newly arrived nodes. Flooding [2] is one extreme solution to find tunnels. Although it guarantees the discovery of shortest paths between neighbors in the virtual overlay, the communication overhead increases at least linearly with the number of nodes. This cost renders it insufficient for any realistic deployment. Other solutions have been suggested [1], [4], but not sufficiently analyzed.

In this paper, we pose the question if efficient self-stabilization in virtual overlays can be achieved at all. Specifically, we are interested if both the expected length of the tunnels as well as the overhead for their maintenance can be within polylogarithmic complexity. To this end, we model the tunnel length distribution as a discrete stochastic process, where each step corresponds to an added or removed tunnel. In our theoretical analysis, we distinguish two types of routable topologies. Different conventional overlays create either routing tables with unique entries per target address range (cf. Chord [7]), or routing tables with bins of entries (cf. Kademlia [8]). We show that, applied to virtual overlays, neither approach can achieve an expected polylog tunnel length at polylog cost over time. For the former, we also prove that the polylog tunnel length is exceeded after at most $\mathcal{O}(n\ polylog(n))$ tunnels are added or removed. To put these asymptotic results into context, we perform a simplified simulation study under idealized conditions. The results clearly indicate that the tunnels degrade quickly, even for small networks, within a very short period of operation, using a churn model based on real-world Freenet data.

In Section II, we introduce background and related work on virtual overlays. Afterwards, in Section III and IV, our system model is designed and its basic properties are given, respectively. Section V presents our asymptotic analysis of virtual overlays with uniquely defined routing tables, whereas Section VI considers less strictly defined overlay structures. A

[1]https://emu.freenetproject.org/pipermail/devl/2013-January/036765.html

simulation study completes our evaluation in Section VII.

II. Background and Related Work

Establishing multi-hop tunnels between nodes in an overlay using social graphs has been considered for various reasons, e.g. Pisces [9] aims to provide anonymity in a distributed hash table, whereas Whanau [10] has been suggested to increase the resilience to Sybil attacks. However, in both cases, the tunnels are set up randomly with randomly selected endpoints.

Virtual overlays, in contrast, use tunnels to establish a structure on connectivity restricted networks, which allows for recursive, greedy routing based on the local knowledge of each node. The majority of approaches suggests a ring routing (cf. [1], [3], [4]), only Vasserman et al. suggest a Kademlia-like hypercube [2]. Though these choices have an impact on the routing performance, the decisive conceptual difference is the approach to their tunnel setup and maintenance.

The straightforward solution of flooding, as suggested in [2], provides the shortest possible tunnel length. The authors only provide an experimental evaluation of the routing algorithm's resilience to concurrent node failures, the maintenance overhead is not considered. However, flooding the network is bound to be highly inefficient and not scalable, especially in high churn scenarios such as Darknets, for which sessions of less than 10 minutes are common [11].

Other approaches aim to leverage the overlay algorithm to discover and repair tunnels, to circumvent the inefficiency of flooding. [1] arranges the nodes on a virtual ring. A joining node first establishes an initial tunnel to the closest node in the overlay by relaying a message to its own ID through a random, physically connected neighbor. This initial tunnel to its new successor in the overlay is then used to establish another tunnel to the predecessor, thus reconnecting the ring through the new node. Tunnels to additional nodes can now be set up by routing for the respective IDs. Upon tunnel disruption due to churn, two possibilities are suggested for repair: Either the first hop in the tunnel may set up an entirely novel tunnel. Or the last hop before the failed node repairs the tunnel by locally routing towards the tunnel endpoint and concatenating the remaining first part of the original tunnel and the new tunnel. The approach is analyzed for rather small deployments of sensor networks with up to 200 mobile nodes, and an improvement over various prior solutions is demonstrated. [3] extends the above work by changing the routable topology from a simple ring to a Chord. Due to the low network size and the absence of churn, the results do not give any insights on the scalability of the maintenance costs.

Mittal et al. [4] analyze the protocol from [1] in the context of Darknets. Finding that the tunnel length indeed increases over time, they extend it by two additional maintenance algorithms. First, after a new node has joined and established its tunnels, its underlay neighbors consider all tunnels they are contained in whose length exceeds a certain threshold, and try to find alternative shorter tunnels via the newly joined node. Secondly, if the number of tunnels a node participates in exceeds a second threshold, it contacts the first hop of the tunnel to look for an alternative tunnel by routing via a different underlay neighbor. The simulative evaluation using networks of several ten thousands nodes shows that these enhancements provide a decreased mean tunnel length in comparison to the basic approach. However, the system behavior over time is only analyzed by simulating sequential joins, not leaves. Leaving nodes are only considered in terms of the routing success under concurrent failures. So, the impact of realistic churn on the tunnel length is disregarded entirely.

With most studies focusing on the engineering of novel, better solutions, none actually analyzes the general benefits and limitations of the concept of virtual overlays. We hence set out to analyze if efficient maintenance and efficient routing is actually possible in virtual overlays in general.

III. Model

In this section, we formally model the tunnel length over time. We start by clarifying some notation about random variables, before describing our model of a (static) virtual overlay. Then, we define the stochastic process used to characterize the system's development over time.

A. Notation

We start by clarifying our notation regarding random variables and random processes, distinguishing means over all nodes and expectations over time. We formally distinguish the two as follows. Denote by $\Pi(\mathbb{Z}_n)$ the set of all probability mass functions with values in $\mathbb{Z}_n = \{0, \dots, n-1\}$. For any $x \in \Pi(\mathbb{Z}_n)$, we denote the probability that x has value i by $x(i)$. The *mean* of x is denoted by $mean(x) = \sum_{i=0}^{n-1} ix(i)$. For a random process $(Y_t)_{t\in\mathbb{N}}$ with $Y_t \in M$ for an arbitrary set M, we denote the probability that Y_t has value y by $P(Y_t = y)$. Now, let $(X_t)_{t\in\mathbb{N}}$ be a random process in $\Pi(\mathbb{Z}_n)$, i.e. each realization $x_0, x_1, \dots$ of $(X_t)_{t\in\mathbb{N}}$ is a sequence of probability mass functions. Furthermore, X_t takes only a finite number of values [2]. Using the above terminology, the *expectation* at time t of any function $f\colon \Pi(\mathbb{Z}_n) \to \mathbb{R}$ on the random variable X_t is defined as

$$\mathbb{E}(f(X_t)) = \sum_{x\in\Pi(\mathbb{Z}_n)} f(x)P(X_t = x).$$

In particular, the expected mean of the random process $(X_t)_{t\in\mathbb{N}}$ is given by

$$\mathbb{E}(mean(X_t)) = \sum_{x\in\Pi(\mathbb{Z}_n)} mean(x)P(X_t = x).$$

Therefore, we can use the above terminology to model the evolution of multi-scalar graph properties. Furthermore, for all function $g\colon \mathbb{R} \to \mathbb{R}$ and $x \in \Pi(\mathbb{Z}_n)$, we define

$$mean(g(x)) = \sum_{i=0}^{n-1} g(i)x(i), \tag{1}$$

[2]The condition assures that the following terminology is well-defined.

and for any random variable X_t with values in $\Pi(\mathbb{Z}_n)$

$$\mathbb{E}(mean(g(X_t)) = \sum_{x \in \Pi(\mathbb{Z}_n)} P(X_t = x) \sum_{i=0}^{n-1} g(i)x(i). \quad (2)$$

The expectations $\mathbb{E}(Y)$ and $\mathbb{E}(g(Y))$ of a real-valued random variable are defined by their Lebesgue integral. We denote the complement of an event H by $H^{\perp}$.

B. System Description

Note that all our algorithms are distributed and local, meaning that each node bases its actions purely on its partial view of the network.

A *virtual overlay* is a 7-tuple $O = (V, E, W, ID, F, S, A)$, so that

- (V, E) is a graph with node set V and edge set $E \subset V \times V$.
- W is a finite set (called *ID space*) with a distance function $dist$.
- $ID\colon V \to W$ maps each node $v \in V$ to an element in W (called *ID* or *identifier*). IDs are chosen independently of the underlying graph (V, E), so there is no relation between the distance of two node's IDs and their topological distance. In practice, $ID(v)$ is commonly the hash of v's IP address.
- $F \subset V^*$ is the tunnel set, consisting of vectors $p = (v_1, \ldots, v_l)$ for some $l \in \mathbb{Z}_n$ with $(v_i, v_{i+1}) \in E$ for $i = 1, \ldots, l-1$.
- S is a local *routing algorithm* that, given an arbitrary identifier $w \in W$ and source node $s \in V$, finds a path from s to $t \in V$ such that $dist(w, ID(t))$ is minimized.
- A is a local *tunnel discovery algorithm* that, given a source node v_0 finds a tunnel $p = (v_1, \ldots, v_l)$ to a virtual overlay neighbor v_l [3].

The distance function $dist$ is defined for elements of W. We extend the definition to nodes $v, u \in V$, so that $dist(v, u) = dist(ID(u), ID(v))$ denotes the distance of v and u's identifiers. In the following, we refer to the first and the last hop of a tunnel p as *startpoint* $s(p)$ and *endpoint* $e(p)$, respectively. We define the length $len(p)$ of a tunnel p as the number of nodes on the tunnel and assume tunnels to be acyclic. A node v is said to be *contained* in a tunnel $p = (v_1, \ldots, v_l)$ if $v \in p$. The tunnel length distribution $L \in \Pi(\mathbb{Z}_n)$ gives the fractions of tunnels of length i for all $i \in \mathbb{Z}_n$.

Routing Tables: Each node $v \in V$ in a virtual overlay O keeps a *neighbor set* $NT(v)$

$$NT(v) = \{w \in V, (v, w) \in E\},$$

a *routing table* $RT(v)$ of tunnels with startpoint v

$$RT(v) = \{(id_z, v_2) \in W \times V\colon \exists f = (v, v_2, \ldots, v_l) \in F, ID(v_l) = id_z\}$$

and a *tunnel table* $FT(v)$ of tunnels v is contained in, but not the startpoint

$$FT(v) = \{(id_s, id_z, v_-, v_+) \in W \times W \times V \times V\colon \exists f = (v_1, \ldots, v_-, v, v_+, \ldots, v_l) \in F, ID(v_0) = id_s, ID(v_l) = id_z\}.$$

Routing and Tunnel Discovery: For both the routing algorithm S as well as the tunnel discovery algorithm A, a node v contacts a set $next \subset NT(v)$ of its neighbors based on $RT(v)$ and $FT(v)$. If all nodes of an old tunnel $p \in F$ are contained in a newly constructed tunnel p', we say that p is *contained* in p'. We say that a newly constructed tunnel $p' = (v_0, \ldots, v_l)$ contains a *shortcut* if p' cannot be represented as a concatenation of old tunnels, such that all tunnels are contained in p'. Note that we assume that no underlying routing protocol similar to IP or geographic coordinates in provided, which can be used to discover tunnels.

C. The Tunnel Length as a Random Process

In this section, we model the evolution of a virtual overlay or, more precisely, of the tunnel length in a virtual overlay, as a discrete random process. Each step of the random process corresponds to either establishing or removing one tunnel. Since we assume the network size to remain largely constant, both are equally likely. In practice, each topology change leads to the removal and construction of several tunnels. The number of removed and newly constructed tunnels per change depends highly on the nature of the topology changes: Both nodes and edges can be added or removed. Note that the average number of tunnels per node/edge are not necessarily equal to the expected number of tunnels that are affected by a removal. Long-lived nodes/edge are expected to be contained in a higher number of tunnels. Therefore, the expected number of removed tunnels is bound to be lower than the average number of tunnels a node is contained in. The exact relation between the two quantities depends on the failure model. In order to overcome this dependency, we abstract the process as a sequence of tunnel failures and constructions rather than node joins and leaves. If the failure model is known, the number of topology changes can be related to the number of tunnel changes in the model.

The state of the virtual overlay at time t is denoted by $O_t = (V_t, E_t, W, ID, F_t, S, A)$. Note that the ID space W, the ID assignment $ID\colon \bigcup_{t \in \mathbb{N}} V_t \to W$, the routing, and the tunnel discovery algorithm remain the same for all t. The neighbor set, routing table, and tunnel table at step t are called $NT_t(v)$, $RT_t(v)$, and $FT_t(v)$, respectively. The evolution of the tunnel length distribution is modeled as a random process $(L_t^A)_{t \in \mathbb{N}}$, with $L_t^A \in \Pi(\mathbb{Z}_n)$ being the tunnel length distribution after t tunnels have been changed. In agreement with our assumption that the network size remains largely constant, a tunnel is equally likely to be removed or constructed. We denote by N_t^A the length of a newly constructed tunnel at step t, and by R_t^A the length of a removed tunnel.

In terms of the introduced model, we obtain the following result in this paper: Fix an arbitrary real number $r > 0$. We

[3] To emphasize the generality of our result, we do not restrict the routing and tunnel discovery algorithm apart from their decentralized nature. Commonly, a greedy algorithm which selects the locally known tunnel(s) with endpoints closest to the target is applied.

prove that there exists t_r, such that the expected mean tunnel length after t_r changes is bound from below by $C \log^r n$, i.e.

$$\forall r \in \mathbb{R} \exists t_r \in \mathbb{N} \colon \mathbb{E}(mean(L^A_{t_r})) = \Omega(\log^r n).$$

The expected mean tunnel length is hence not polylog for $t \to \infty$ because it can be bound from below by any polylog term.

Having introduced all required concepts, we are now able to express basic results and our assumptions in formal terms.

IV. Basic Properties and Assumptions

In this section, we establish the common ground needed for the remainder of the paper. More precisely, we i) define the distribution of R^A_t, the length of a removed tunnel, ii) obtain an upper bound on the probability of a tunnel to contain a shortcut, and iii) state assumptions the rest of the paper is based upon.

A. Distribution of R^A_t

We now express the distribution of R^A_t, the length of a removed tunnel, in terms of current tunnel length distribution L^A_t. The probability of a tunnel to be destroyed by a leaving node is proportional to the length of the tunnel. Let l^A_t be a realization of the corresponding tunnel length distribution. For any tunnel p, denote by $\delta(p)$ the event that the tunnel is destroyed by one leaving node and recall that $len(p)$ is the length of p. A tunnel of length i is destroyed if any of its i nodes leave, i.e. $P(\delta(p)|len(p) = i \cap L^A_t = l^A_t) = \frac{i}{n}$. The probability that a removed tunnel p has length i given the realization l^A_t is

$$\begin{aligned} &P\left(len(p) = i | \delta(p) \cap L^A_t = l^A_t\right) = \\ &\frac{P\left(\delta(p)|len(p) = i \cap L^A_t = l^A_t\right) P(len(p) = i | L^A_t = l^A_t)}{P\left(\delta(p)|len(p) = i \cap L^A_t = l^A_t\right)} \\ &= \frac{i \cdot l^A_t(i)}{n \sum_{j=0}^{n} \frac{j \cdot l^A_t(j)}{n}} = \frac{i \cdot l^A_t(i)}{mean(l^A_t)}. \end{aligned}$$

applying Bayes' rule in the first step. We thus define the probability that a removed tunnel at time t has length i as the expectation of $P\left(len(p) = i | \delta(p) \cap L^A_t = l^A_t\right)$ over all possible realizations l^A_t of L^A_t

$$\begin{aligned} &P(R^A_t = i) = \mathbb{E}\left(\frac{i L^A_t(i)}{mean(L^A_t)}\right) \qquad (3) \\ &= \sum_{l^A_t \in \Pi(\mathbb{Z}_n)} P(len(p) = i | \delta(p) \cap L^A_t = l^A_t) P(L^A_t = l^A_t). \end{aligned}$$

B. Probability to Shortcut

In this section, we prove a Lemma needed in future sections. An endpoint $e(p')$ of a newly constructed tunnel p' is either found when a tunnel leading to $e(p')$ is fully contained in p' or if p' contains a shortcut to $e(p')$. We obtain an upper bound on the probability of the latter. In the following sections, we will then show that with high probability a new tunnel is a concatenation of old tunnels and hence likely to be longer than existing tunnels.

Lemma IV.1. *Let $O_t = (V_t, E_t, W, ID, F_t, S, A)$ be a virtual overlay. Set $n = |V_0|$ and assume $|F_t| = \mathcal{O}(n \log^\alpha n)$. Furthermore, let M_t be the number of messages exchanged during a tunnel discovery. The probability of the event H that the newly constructed tunnel contains a shortcut to at least one of Z_t nodes is bound from above by*

$$P(H) = \mathcal{O}\left(\frac{\mathbb{E}(M_t)\left(\mathbb{E}(mean(L^A_t)) \log^\alpha n + 2\mathbb{E}\left(\frac{E_t}{V_t}\right)\right)\mathbb{E}(Z_t)}{n}\right). \qquad (4)$$

Proof: We first show a version of Jensen's inequality for random variables in $\Pi(\mathbb{Z}_n)$. In the main part of the proof, we then obtain the probability that one node does not have a shortcut to any of the Z_t nodes. The claim follows by determining the probability that M_t nodes do not have such a shortcut.

We show the following version of Jensen's inequality for discrete random variables X with finitely many values in $\Pi(\mathbb{Z}_n)$: For any convex function $g \colon \mathbb{R} \to \mathbb{R}$, it holds that

$$\mathbb{E}(mean(g(X))) \geq g(\mathbb{E}(mean(X)). \qquad (5)$$

For the proof, let $x \in \Pi(\mathbb{Z}_n)$. By the definition of mean and Jensen's inequality for real-valued random variables, we have $mean(g(x)) \geq g(mean(x))$. Furthermore, by Eq. 2

$$\begin{aligned} \mathbb{E}(mean(g(X))) &= \sum_{x \in \Pi(\mathbb{Z}_n)} P(X = x) mean(g(x)) \\ &\geq \sum_{x \in \Pi(\mathbb{Z}_n)} P(X = x) g(mean(x)). \end{aligned}$$

Consider a real-valued random variable $\tilde{X}$ with $P(\tilde{X} = mean(x)) = P(X = x)$ for all $x \in \Pi(\mathbb{Z}_n)$. Then

$$\begin{aligned} &\sum_{x \in \Pi(\mathbb{Z}_n)} P\left(\tilde{X} = mean(x)\right) g(mean(x)) = \mathbb{E}\left(g(\tilde{X})\right) \\ &\geq g\left(E(\tilde{X})\right) = g\left(\mathbb{E}(mean(X))\right). \end{aligned}$$

The second last step follows from Jensen's inequality, the last step from the definitions of $\tilde{X}$ and $\mathbb{E}(mean(X))$. This completes the proof of Eq. 5.

Now, we prove the Eq. 4. A node v is aware of its neighbors in the underlay, as well as the startpoints and endpoints of the tunnels it is contained in. Denote by H_1 the event that a node v does not have a shortcut to any of the Z_t potential endpoints. If Z_t takes value z and v has d neighbors and is contained in y tunnels, the probability that v is not aware of any possible endpoint is at most $(1 - \frac{z}{n})^{d+y}$. The mean number of tunnel table entries Y_t per node is at most $\mathbb{E}(Y_t) = \mathcal{O}\left(\mathbb{E}(mean(L^A_t) \log^\alpha n\right)$ because there are $|F_t| = \mathcal{O}(n \log^\alpha n)$ tunnels. We apply Jensen's inequality and Eq. 5 for the convex functions $f_1(z) = 1 - \frac{z}{n}$ and

$f_2(x) = \left(1 - \frac{mean(Z)}{n}\right)^x$ to obtain

$$P(H_1) \geq \mathbb{E}\left(\left(1 - \frac{Z_t}{n}\right)^{|N_t(v)|+Y_t}\right) \geq \left(1 - \frac{\mathbb{E}(Z_t)}{n}\right)^{2\mathbb{E}\left(\frac{E_t}{V_t}\right)+\mathbb{E}(mean(L_t^A))\log^\alpha n}.$$

Hence, the probability $P(H)$ can be bound as the complement of the event that none of at most M_t nodes contacted during the tunnel discovery are aware of a potential endpoint. Again, we apply Jensen's inequality to the function

$$g(x) = \left(\left(1 - \frac{\mathbb{E}(Z_t)}{n}\right)^{2\mathbb{E}\left(\frac{E_t}{V_t}\right)+\mathbb{E}(mean(L_t^A))\log^\alpha n}\right)^x,$$

resulting in

$$P(H) \leq 1 - \mathbb{E}(g(X_t)) \leq 1 - g(\mathbb{E}(M_t)) = 1 - \left(1 - \frac{\mathbb{E}(Z_t)}{n}\right)^{\left(2\mathbb{E}\left(\frac{E_t}{V_t}\right)+\mathbb{E}(mean(L_t^A))\log^\alpha n\right)\mathbb{E}(M_t)} = \mathcal{O}\left(\frac{\mathbb{E}(M_t)\left(\mathbb{E}(mean(L_t^A)\log^\alpha n + 2\mathbb{E}\left(\frac{E_t}{V_t}\right)\right)\mathbb{E}(Z_t)}{n}\right)$$

as claimed. ∎

C. Assumptions

In the remainder of the paper, we set $n = |V_0|$. The following assumptions are made:

1) The average degree $2\frac{|E_t|}{|V_t|} \leq K$ is bound by a constant K.
2) $|F_t| = \theta\left(n \log^\alpha n\right)$ for some $\alpha \in \mathbb{R}$, i.e. the average routing table size is polylog.
3) During the execution of the tunnel discovery algorithm A at most $\mathcal{O}\left(\log^\beta n\right)$ messages are exchanged.

Assumption 1) of a constant average degree can be replaced by the assumption of a polylog average degree at the price of some additional case distinctions. For simplicity, we choose the above assumption, which is commonly used for various network types such as trust topologies for Darknets. Assumption 2) holds for $\alpha = 1$ for most common structured overlays such as Chord and Kademlia. Assumption 3) states that we only consider algorithms with a polylog overhead.

After introducing our model and its basic properties, we now present our analysis.

V. Fully Determined Virtual Overlays

In this section, we consider *fully determined* virtual overlays, for which the tunnel start- and endpoints are uniquely determined by the ID assignment ID. For example, a virtual overlay based on Chord is fully determined. We show that for all $r > 0$ the expected mean tunnel length for all is at least of order $\log^r n$ after $n \log^{3r+1+\alpha} n$ steps of the random process described in Section III .

Theorem V.1. *Let $O_t = (V_t, E_t, W, ID, F_t, S, A)$ be a fully determined virtual overlay. For any $r > 0$, the expected mean tunnel length is bound from below by*

$$\mathbb{E}\left(mean(L_t^A)\right) = \Omega(\log^r n) \text{ for all } t = \Omega(n \log^{3r+1+\alpha} n).$$

Proof: Let $\lambda_t(q)$ be the q-quantile of L_t^A for some $q = \frac{1}{\log^k n}$ where $k > 1$ is determined during the proof. In the following, we bound the number of steps until $\mathbb{E}(\lambda_t(q)) = \Omega(\log^r n)$. Then the expected mean tunnel length is at least of order $\log^r n$ as well because for $n \geq 4$, $\mathbb{E}\left(mean(L_t^A)\right) \geq (1-q)\mathbb{E}(\lambda_t(q)) > \frac{\mathbb{E}(\lambda_t(q))}{2} = \Omega(\log^r n)$. Let C_t be the number of tunnels that are at least of length $\lambda_t(q) + 1$. We show that

$$\mathbb{E}(C_t - C_{t-1}) = \Omega\left(\frac{q}{\log^r n}\right), \tag{6}$$

independently of t. Based on Eq. 6, we can determine the number of steps needed to increase the q-quantile by 1. The expected number of tunnels with length at least $\lambda_t(q) + 1$ increases by $\Omega\left(\frac{q}{\log^r n}\right)$ for one tunnel addition or removal. As a consequence, in x changes, the number of such tunnels increases by $\Omega\left(x\frac{q}{\log^r n}\right)$. Hence, there are $(1-q)|F_t|$ tunnels of length at least $\lambda_t(q) + 1$ after $\mathcal{O}\left((1-q)\frac{\log^r n}{q}|F_t|\right) = \mathcal{O}\left(\frac{\log^r n}{q}|F_t|\right)$ changes, i.e.

$$\forall t_0 \colon \mathbb{E}(\lambda_{t_0+t}(q)) = \Omega\left(\mathbb{E}(\lambda_{t_0}(q)) + 1\right) \text{ for all } t = \Omega\left(\frac{\log^r n}{q}|F_{t_0}|\right).$$

An upper bound on the number of steps to increase the mean tunnel length by $\log^r n$ follows directly. It is

$$\mathbb{E}\left(mean(L_t^A)\right) = \Omega\left(\mathbb{E}(\lambda_t(q))\right) = \Omega\left(\mathbb{E}(\lambda_0(q)) + \log^r n\right) = \Omega\left(\log^r n\right) \tag{7}$$
$$\text{for all } t = \Omega\left(\log^r n \frac{\log^r n}{q}|F_0|\right) = \Omega\left(n\frac{\log^{2r+\alpha} n}{q}\right).$$

It remains to prove Eq. 6. If a new tunnel of length longer than $\lambda_t(q)$ is constructed, the number of such tunnels increases by 1, and decreases by 1 if such a tunnel is removed. Removal and construction are equally likely, so that $\mathbb{E}(C_t - C_{t-1}) = \frac{1}{2}\left(P\left(N_t^A > \lambda_t(q)\right) - P\left(R_t^A > \lambda_t(q)\right)\right)$.

We assume $\mathbb{E}(mean(L_t^A)) = \mathcal{O}(\log^r n)$, otherwise the claim holds. Each removed tunnel is of length at least 1, so that by Eq. 3

$$P(R_t^A \leq \lambda_t(q)) \geq \frac{q}{\mathbb{E}(mean(L_t^A))} = \Omega\left(\frac{q}{\log^r n}\right). \tag{8}$$

In order to bound the tunnel length of a newly constructed tunnel, consider the event H that the discovered tunnel p contains less than two old tunnels. Otherwise, the new tunnel can only be of length at most $\lambda_t(q)$ if two tunnels are shorter than $\lambda_t(q)$, i.e.

$$P(N_t^A \leq \lambda_t(q)) = P(N_t^A \leq \lambda_t(q)|H)P(H) + P(N_t^A \leq \lambda_t(q)|H^\perp)(1 - P(H)) \leq P(H) + P(N_t^A \leq \lambda_t(q)|H^\perp) \leq P(H) + q^2. \tag{9}$$

The probability $P(H)$ in Eq. 9 can be bound by Lemma IV.1. We determine an upper bound on the number of nodes Z_t, so that p can only contain less than two tunnels if it contains a shortcut to any of those Z_t nodes. Let u be the uniquely determined endpoint of the new tunnel. Then the Z_t nodes consist of all nodes on a tunnel to u or to any of u's overlay neighbors.

We hence define $ON_t(u) = \{v \in V\colon \exists(ID(u), v_2) \in RT(v)\}$ to be the set of nodes with routing table entries with endpoint u. Furthermore, let $FN_t(u) = \{v \in V\colon \exists(id_S, ID(u), v_+, v_-) \in FT(v)\}$ be the set of nodes that have a trail table entry with endpoint u. If p contains less than two tunnels, p has to contain a shortcut to a node in

$$Z(u) = \{u\} \cup NT_t(u) \cup ON_t(u) \cup FN_t(u) \bigcup_{v \in ON_t(u)} FN_t(v).$$

On average, a node is the endpoint of $\mathbb{E}\left(|ON_t(u)|\right) = \mathbb{E}\left(|F_t(u)|/n\right) = \mathcal{O}\left(\log^{\alpha} n\right)$ tunnels. By assumption, each tunnel is on average of length at most $\mathcal{O}(\log^{r} n)$, so that the number of tunnel table entries with endpoint v is $\mathbb{E}\left(|FN_t(v)|\right) = \mathcal{O}\left(\log^{r} n\, |ON_t(v)|\right) = \mathcal{O}\left(\log^{r+\alpha} n\right)$. Hence $\mathbb{E}(Z_t) = \mathbb{E}\left(|Z(u)|\right) \leq 1 + E\Bigg(|NT_t(u)| + |ON_t(u)| + |FN_t(u)| + \sum_{v \in ON_t(u)} |FN_t(v)|\Bigg) = \mathcal{O}\left(|FN_t(u)|\, |ON_t(u)|\right) = \mathcal{O}\left(\log^{r+2\alpha} n\right)$.

The number of nodes contacted during the tunnel discovery by is at most $\mathbb{E}(M_t) = \log^{\beta} n$ by assumption, and the expected degree is constant. We apply Lemma IV.1 to determine the upper bound

$$P(H) = \mathcal{O}\left(\frac{\log^{\beta} n \log^{\alpha} n \log^{r} n \log^{r+2\alpha-2} n}{n}\right) = \mathcal{O}\left(\frac{polylog(n)}{n}\right).$$

Therefore, Eq. 9 is dominated by the term q^2 for $q = \frac{1}{\log^{k} n}$ for a constant k, so that

$$P(N_t^A \leq \lambda_t(q)) = \mathcal{O}\left(q^2\right). \tag{10}$$

We can determine $\mathbb{E}(C_t - C_{t-1})$ in Eq. 6 by Eqs. 8 and 10 $\mathbb{E}(C_t - C_{t-1}) = \Omega\left(1 - q^2 - 1 + \frac{q}{\log^{r} n}\right) = \Omega\left(\frac{q}{\log^{r} n} - q^2\right)$. We set $q = \frac{1}{\log^{r+1} n}$, so that $\mathbb{E}(C_t - C_{t-1}) = \Omega\left(\frac{1}{\log^{2r+1} n}\right)$. By Eq. 7, for all $t = \Omega(n \log^{2r+\alpha} n/q) = \Omega(n \log^{3r+\alpha+1})$, we indeed have $\mathbb{E}\left(mean(L_t^A)\right) = \Omega\left(\log^{r} n\right)$. ∎

We have shown in Theorem V.1 that for any $r > 0$, there exists t_r, such that $\mathbb{E}\left(mean(L_t^A)\right) = \Omega\left(\log^{r} n\right)$ for all $t > t_r$. For all $\epsilon > 0$ and $t = \Omega(n^{1+\epsilon})$, the expected mean tunnel length is $\mathbb{E}\left(mean(L_t^A)\right) = \omega\left(\log^{r} n\right)$ for all $r > 0$ and not polylog.

VI. Partially Determined Virtual Overlays

In this section, we consider *partially determined* virtual overlays, such that a link in the virtual overlay can potentially have several endpoints. For example, a virtual overlay constructing Kademlia is partially determined because all nodes with a certain prefix are potential endpoints. Here, we restrict our analysis to a wide class of virtual overlays, for which the routing terminates in a logarithmic number of virtual overlay hops. We show that the hops in the underlay cannot be polylog under the assumptions of polylog maintenance costs.

In order to simplify notation, we assume $|W| = |V|$, i.e. there is a bijective mapping from IDs to nodes. The results hold regardless of the above assumption. Let $B_d(u) = \{w \in V : dist(u, w) \leq d\}$ denote the set of nodes within distance d of u. We say that a node u is a k-closest node of v if $u \in B_d(v)$ for some d such that $|B_d(v)| \leq k$. We define a partially determined virtual overlay with a $1/B_d$ distance distribution by the following three conditions:

1) The set of tunnels between 2^{i+1}-closest but not 2^i-closest nodes makes up approximately a logarithmic fraction of all tunnels for $i = 0 \ldots \log n$.
2) Furthermore, we assume that $|B_d(w)| = \theta(d^{\mu})$ for some $\mu \in \mathbb{N}$, i.e. the ID space is essentially a μ-dimensional lattice.
3) When setting up a tunnel, the the endpoint $e(p)$ has a 2^{i+1}-closest but not within the 2^i-closest node to the startpoint for a fixed i.

Examples satisfying condition 1) and 2) are e.g. Kleinberg's small world model, with the probability of two nodes being neighbors decreasing in proportion to the number of nodes with at most that distance [12], as well as Kademlia and its variations, which select roughly the same number of neighbors for each common prefix length. Routable topologies satisfying condition 2) are efficient in the number of overlay hops if and only if condition 1) holds [12]. So, the definition considers basically all routable topologies that achieve routing in a polylog number of virtual overlay hops, which is a necessary but not sufficient condition for achieving a polylog hop count in the underlay. In the following, we show that indeed such topologies do not provide efficient routing in the underlay, at least not at polylog maintenance cost. We make use of the following fact:

Lemma VI.1. *Let T be a set of nodes. We say that a tunnel p improves by a factor f if it leads from a startpoint $u \in B_{d_1}(T) \setminus B_{d_1-1}(T)$ to an endpoint $v \in B_{d_2}(T) \setminus B_{d_2-1}(T)$, such that $f = |B_{d_1}(T)|/|B_{d_2}(T)|$. For a virtual overlay O_t with a $1/B_d$ distance distribution, the probability that a tunnel improves by a factor $f > 1$ is at most*

$$\mathcal{O}\left(\frac{1}{f \log n}\right). \tag{11}$$

The proof is presented for Kleinberg's small world model in [13], so that we exclude it due to space constraints.

Theorem VI.2. *Let $O_t = (V_t, E_t, W, ID, F_t, S, A)$ be a virtual overlay with a $1/B_d$ distance distribution and $|F_t| = \Omega(n)$. For any $r \in \mathbb{N}_0$, there exists t such that*

$$\mathbb{E}\left(mean(L_t^A)\right) = \omega(\log^{r} n).$$

Proof: We focus on a set of tunnels with the minimal number of potential endpoints. Because they make up a logarithmic fraction of all tunnels due to the $1/B_d$ distance distribution, it suffices to show that their length exceeds $\omega(\log^{r+1} n)$ to show the claim. The main idea of the proof is to show that with high probability the newly constructed tunnel contains $\Omega\left(\frac{\log n}{\log\log n}\right)$ tunnels, each improving by at most 2^m for $m = k \log\log n$ to be determined during the proof. In other words, the fraction of nodes closer to the target decreases by at most a factor 2^m while following one tunnel. We then bound the expected tunnel length for $t \to \infty$ from below by $\omega(\log^r n)$ based on that result.

We start by introducing some notation. Let $F_{t,i}$ denote the set of tunnels such that the endpoint is a $2^{(i+1)m}$-closest node but not a 2^{im}-closest node. We have $|F_{t,i}| = \theta\left(\frac{k \log\log n}{\log n}|F_t|\right)$ because O_t has a $1/B_d$ distance distribution (Condition 1). Denote the tunnel length distribution of $F_{t,i}$ by $L^A_{t,i}$. In particular,

$$\mathbb{E}\left(mean(L^A_t)\right) = \Omega\left(\frac{\log\log n}{\log n}\mathbb{E}\left(mean(L^A_{t,i})\right)\right). \quad (12)$$

We only consider tunnels for which the number of potential endpoints is at most $\sqrt{n} = 2^{\log n/2}$. For this purpose, fix $i_0 = \lceil \frac{\log n}{2k \log\log n} \rceil$, the highest index i of interest. In the following, we divide the set $F_{t,i}$ into subsets $S_{t,i}$ and $S^\perp_{t,i}$. $S_{t,i} = S^0_{t,i} \cup S^1_{t,i} \cup S^2_{t,i}$ contains tunnels that are potentially shorter:

1) $S^0_{t,i}$: all remaining initially present tunnels,
2) $S^1_{t,i}$: newly constructed tunnels that contain a shortcut to any of the $n^{0.5}$-closest nodes to T
3) $S^2_{t,i}$: newly constructed tunnels not in $S^1_{t,i}$, but for which at least one of the contained tunnels is an element of $S_{t,j}$ for some $i_0 \geq j > i$

The tunnel length distribution of tunnels in $S^\perp_{t,i}$ is denoted by $\Lambda^A_{t,i}$ in the following.

Having introduced the necessary notation, we now give a short overview of the necessary steps of the proof. The actual proof is then rather technical, employing a variety of techniques from probability theory and calculus. We first show that $S^\perp_{t,0}$ makes up a non-negligible fraction of $F_{t,0}$. For this purpose, we derive an recursive formula of $\limsup_{t\to\infty} \mathbb{E}\left(|S_{t,i}|\right)$ and solve the recursion using the case $i = i_0$ as the recursion anchor. Secondly, we determine a bound on $\mathbb{E}\left(\Lambda^A_{t,i}\right)$ for $i = 0$. We condition on the event that all contained tunnels improve by at most a factor 2^m and again derive an recursive relation expressing $\mathbb{E}\left(mean(\Lambda^A_{t,i})\right)$ in terms of $\sum_{j=i+1}^{i_0} \mathbb{E}\left(mean(\Lambda^A_{t,j})\right)$. In summary, we prove the claim by showing

$$\begin{aligned}\mathbb{E}\left(mean(L^A_{t,0})\right) &\geq \mathbb{E}\left(mean(\Lambda^A_{t,0})\right)\mathbb{E}\left(\frac{|S^\perp_{t,0}|}{|F_{t,i}|}\right)\\ &= \Omega(\log^{r+1} n). \end{aligned} \quad (13)$$

We assume that $\mathbb{E}\left(mean(L^A_{t,i})\right) = \mathcal{O}\left(\log^{r+1} n\right)$, otherwise there is nothing to show.

In the first part, we prove that there exists t_A, so that for $t \geq t_A$, $\mathbb{E}\left(\frac{|S^\perp_{t,0}|}{|F_{t,0}|}\right) = \Omega(1)$. We proof the above bound by expressing the probability to remove and to construct a tunnel in $S_{t,i}$ in terms of $\mathbb{E}\left(|S_{t,i}|\right)$. Let $E^R_{t,i}$ and $E^C_{t,i}$ denote the event that a tunnel in $F_{t,i}$ is removed and constructed, respectively, and p_t be the tunnel removed or constructed in step t. Then the expected size of $S_{t,i}$ is recursively expressed as

$$\begin{aligned}&\mathbb{E}\left(|S_{t,i}|\right) = \mathbb{E}\left(|S_{t-1,i}|\right) + \\ &P(p_t \in S_{t,i}|E^C_{t,i})P(E^C_{t,i}) - P(p_t \in S_{t-1,i}|E^R_{t,i})P(E^R_{t,i}),\end{aligned} \quad (14)$$

the expected size in the step before plus the expected change in size. Since the probabilities of removal and construction are equal and the $1/B_d$ distance distribution is preserved, we have $P(E^R_{t,i}) = P(E^C_{t,i})$. We derive an upper bound on

$$\gamma_i = \limsup_{t\to\infty} \mathbb{E}\left(|S_{t,i}|\right).$$

Note that the above bound is well defined since $\frac{|S_{t,i}|}{|F_{t,i}|} \leq 1$ and $|F_{t,i}| = \mathcal{O}\left(\log^\alpha n\right)$. By the definition of $\limsup$, there exists t_1, such that for all $t > t_1$ and all $i = 0, \ldots, i_0$, $\mathbb{E}\left(|S_{t,i}|\right) < \gamma_i + \frac{2}{n\,P(E^R_{t,i})}$. In particular, if $|\mathbb{E}\left(|S_{t,i}|\right) - \gamma_i| \leq 1/n$

$$P(p_t \in S_{t,i}|E^C_{t,i}) - P(p_t \in S_{t-1,i}|E^R_{t,i}) \leq \frac{1}{n} \quad (15)$$

applying Eq. 14 for $t > t_1$. An upper bound γ_i can be derived as the maximal value of $|S_{t,i}|$, such that Eq. 15 holds. By Eq. 3, the probability of removing a tunnel in $S_{t-1,i}$ is bound by

$$\begin{aligned}P(p_t \in S_{t-1,i}|E^R_{t,i}) &= \mathbb{E}\left(\frac{|S_{t-1,i}|}{|F_{t-1,i}|mean(L^A_{t,i})}\right)\\ &= \Omega\left(\frac{\mathbb{E}(|S(t,i)|)}{n\log^{\alpha+r+1} n}\right).\end{aligned} \quad (16)$$

For the construction, we consider the sets $S^0_{t,i}$, $S^1_{t,i}$, and $S^2_{t,i}$ individually. By definition, $S^0_{t,i}$ only consists of initially existing tunnels, so $P(p_t \in S^0_{t,i}|E^C_{t,i}) = 0$. For $S^1_{t,i}$, the probability that p_t contains a shortcut to a node within the $2^{i_0 k \log\log n}$-closest nodes of any $u \in T$ is bound by Lemma IV.1. The number of exchanged messages, tunnels, and routing hops are bound by $\mathcal{O}(\log^\beta n)$, $\mathcal{O}(\log^\alpha n)$ and $\mathcal{O}(\log^{r+1} n)$, respectively. by assumption. The number of potential target nodes Z_t is at most $2^{i_0 k \log\log n}|T| = 2^{\lceil \frac{\log n}{2k\log\log n}\rceil k \log\log n} 2^m = \mathcal{O}(n^{0.75})$, so that

$$\begin{aligned}P(p_t \in S^1_{t,i}|E^C_{t,i}) &= \mathcal{O}\left(\frac{\log^\beta n \log^\alpha n \log^{r+1} n\, n^{0.75}}{n}\right)\\ &= \mathcal{O}\left(\frac{1}{n^{0.2}}\right).\end{aligned} \quad (17)$$

The last step holds for n so large that $n^{0.05} \geq \log^{\beta+\alpha+r+1} n$. We assume that that most $\log^\beta n$ nodes are involved in the tunnel discovery, hence the new tunnel can consist of at most $\log^\beta n$ old tunnels. Furthermore, $|F_{t,i}| = \Omega\left(n\log^{\alpha-1} n\right)$. The

probability that any of $\log^\beta n$ tunnels is an element of $S_{t,j}$ is obtained by a union bound

$$P(p_t \in S^2_{t,i}|E^C_{t,i}) = \mathcal{O}\left(\log^\beta n \max_{j:i_0\geq j>i}\left\{\frac{\mathbb{E}(|S_{t-1,j}|)}{n\log^{\alpha-1} n}\right\}\right). \tag{18}$$

The fraction of initial paths in the network converges to 0. So, there exists t_ϵ such that $P(S^0_{t,i}) \leq n^{-0.2}$ for all $i = 0, \ldots, i_0$. It follows from Eqs. 17 and 18 that

$$P(p_t \in S_{t,i}|E^C_{t,i}) = \mathcal{O}\left(\frac{1}{n^{0.2}} + \max_{j:i_0\geq j>i}\left\{\frac{\mathbb{E}(|S_{t-1,j}|)}{n\log^{\alpha-\beta-1} n}\right\}\right)$$

By Eqs. 15 and Eq. 16, we have for the limit γ_i

$$\frac{1}{n^{0.2}} + \max_{j:i_0\geq j>i}\left\{\frac{\gamma_{i+1}}{n\log^{\alpha-\beta-1} n}\right\} - \frac{\gamma_i}{n\log^{\alpha+r+1} n} = \mathcal{O}\left(\frac{1}{n}\right).$$

The set $\{j : i_0 \geq j > i\}$ is empty for $i = i_0$. The upper bound on γ_{i_0} is hence

$$\gamma_{i_0} = \mathcal{O}\left(\frac{n\log^{\alpha+r+1} n}{n^{0.2}}\right) = \mathcal{O}\left(n^{0.8}\log^{\alpha+r+1} n\right).$$

For $i < i_0$, we obtain the recursive relation $\gamma_i = \mathcal{O}\left(\log^{\beta+r+2} n\gamma_{i+1}\right)$. So,

$$\begin{aligned}\gamma_0 &= \mathcal{O}\left((\log^{\beta+r+2})^{\frac{\log n}{2k\log\log n}}\gamma_{i_0}\right)\\ &= \mathcal{O}\left(2^{(\beta+r+2)\log\log n\frac{\log n}{2k\log\log n}}\gamma_{i_0}\right)\\ &= \mathcal{O}\left(n^{\frac{\beta+r+2}{2k}}\gamma_{i_0}\right) = \mathcal{O}\left(n^{\frac{\beta+r+2}{2k}}n^{0.8}\log^{\alpha+r+1} n\right)\end{aligned}$$

is an upper bound on γ_0. In order to show $\mathbb{E}\left(|S^\perp_{t,0}|/|F_{t,0}|\right) = \Omega(1)$, consider that $|F_{t,0}| = \Omega(n\log^{\alpha-1} n)$ and $\gamma_0 = \mathcal{O}\left(n^{(\beta+r+2)/(2k)}n^{0.8}\log^{\alpha+r+1} n\right) = \mathcal{O}(n^{0.9}\log^{\alpha+r+1} n)$ for $k \geq 5(\beta + r + 2)$. Hence, there exists t_2 such that for $t > t_A = \max\{t_1, t_2\}$ indeed

$$\mathbb{E}\left(\frac{|S^\perp_{t,0}|}{|F_{t,0}|}\right) = \Omega\left(1 - \frac{\gamma_0}{|F_{t,0}|}\right) = \Omega(1) \tag{19}$$

because $\frac{\gamma_0}{|F_{t,0}|} = \mathcal{O}\left(\frac{n^{0.9}\log^{\alpha+r+1} n}{n\log^{\alpha-1} n}\right)$ and $n^{0.9}\log^{\alpha+r+1} n < 1/2n\log^{\alpha-1} n$ for n big enough. This completes the first part of the proof.

In order to determine a lower bound on $\mathbb{E}\left(mean(\Lambda^A_{t,i})\right)$ for t large enough, we determine a recursive relation for

$$\eta_i = \liminf_{t\to\infty}\mathbb{E}\left(mean(\Lambda^A_{t,i})\right), i \leq i_0. \tag{20}$$

Trivially, $\eta_{i_0} = \Omega(1)$. Denote by H the event that the improvement is at most 2^m for all tunnels contained in p_t after the first node within the closest $n^{0.5}$ nodes to T has been reached. If H does not hold, the length of the new tunnel is at least 1, otherwise there is at least one tunnel from each $F_{t,j}$ contained in p_t for the maximal improvement of 2^m. Therefore, the expected length $\mathbb{E}(len(p_t))$ of a new trail in $S^\perp_{t,i}$ is

$$\mathbb{E}\left(len(p_t)\right) > 1 + P(H^\perp)\sum_{j=i+1}^{i_0}\mathbb{E}\left(mean(\Lambda^A_{t,j})\right), \tag{21}$$

and hence asymptotically $\eta_i \geq 1 + P(H^\perp)\sum_{j=i+1}^{i_0}\eta_j$

holds as well since η_i is bound by n. Now, we determine $P(H)$. We condition on the fact that tunnels in $S^\perp_{t,i}$ do not contain shortcuts, so tunnel table entries are not of interest. Furthermore, the probability that one tunnel improves by 2^m is $\Omega\left(\frac{1}{2^m\log n}\right)$ according to Lemma VI.1. Recall that the tunnel discovery algorithm A considers at most $\log^\beta n$ nodes with on average $\log^\alpha n$ routing table entries. The probability of an improvement by at least a factor 2^m for $m = k\log\log n$ is hence

$$P(H^\perp) = 1 - \left(1 - \frac{1}{2^m\log n}\right)^{\log^{\alpha+\beta} n} = \Omega\left(\frac{\log^{\alpha+\beta} n}{\log^{k-1} n}\right).$$

Thus, $P(H^\perp) \geq 1/2$ for any $k > \alpha - 1 + \beta$ and n large enough. We get for η_i that

$$\begin{aligned}\eta_i &\geq 1 + 1/2\sum_{j=i+1}^{i_0}\eta_j \geq 1 + 1/2\sum_{j=i+2}^{i_0}\eta_j + 1/2\eta_{i+1}\\ &\geq 1 + \eta_{i+1} - 1 + 1/2\eta_{i+1} = 1.5\eta_{i+1}.\end{aligned} \tag{22}$$

By Eq. 22, η_0 is recursively determined as $\eta_0 = \Omega\left(1.5^{\log n/(2k\log\log n)}\right) = \Omega\left(2^{\log 1.5\log n/(2k\log\log n)}\right) = \omega\left(2^{(r+1)\log\log n}\right) = \omega\left(\log^{r+1} n\right)$ and hence for $t > t_B$ for some t_B, $\mathbb{E}\left(mean(\Lambda^A_{t,0})\right) = \omega\left(\log^{r+1} n\right)$. Setting $k = \max\{5(\beta+r+2), \alpha+\beta\}$, the expected mean tunnel length in $F_{t,0}$ is at least $\omega(\log^{r+1} n)$ by Eq. 13 for $t \geq \max\{t_A, t_B\}$. The claim $\mathbb{E}\left(mean(L^A_t)\right) = \omega(\log^r n)$ follows from Eq. 12. ∎

We have now shown that eventually the expected mean tunnel exceeds $\log^r n$ for all $r > 0$. Hence, virtual overlays cannot provide polylog tunnels and hence routing length in combination with polylog maintenance cost.

VII. Simulation

The analytical results being of asymptotic nature, we additionally performed a simplified simulation study to assess the behavior of virtual overlays in Darknet specific environments. We deliberately chose idealized conditions and simplified churn to highlight the extent of the problems virtual overlay approaches are facing, even at moderate network sizes.

For the experiment we implemented a Chord-like virtual overlay: using a b-bit ID space, each node establishes tunnels to their predecessor and successor, as well as to the nodes with an ID succeeding $ID(v) + 2^i \ mod \ 2^b$ for $i = 1 \ldots b-1$. The tunnel discovery was implemented as a greedy routing through the virtual overlay, along existing tunnels or direct neighbors, towards the target ID, chosen to complete the routing table. Disruption of the ring, when a set of nodes that connected two components of the network failed, and reconnection of disconnected components were handled according to [1]: Each ring is identified, known to all nodes participating in it, and upon discovery of a "superior" ring (with a lower ID, by definition), nodes release tunnels to their previous neighbors, informing them of the new ring ID, and establish tunnels to the neighbors in the newly joined ring.

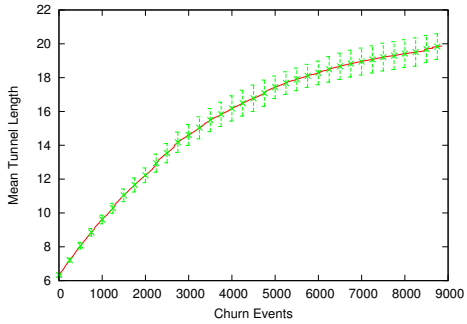

Fig. 1: Mean tunnel length, ∼12k online nodes ([11], [14])

To be conservative in giving a low estimate on the minimum increase of tunnel lengths, we apply two simplifications. First, we treat churn as atomic events: Upon arrival or departure of a node, all corresponding tunnels are established or torn down in an instant, before the next churn event is executed. Second, we reconstruct each disrupted tunnel at its initial node instead of the last hop prior to the departing node. This allows for the discovery of new short tunnels, as opposed to the simple stitching of existing tunnels, which is commonly suggested.

To approximate realistic assumptions, we chose the regional Facebook network of 63371 nodes from [14] as an underlying trust graph to connect the nodes, since recent studies [11] indicate that this reflects the size of the current Freenet deployment well. We also use the measurements from [11] for choosing the join and leave events according to the observed session and intersession length distributions, which yielded that around 12.000 nodes were concurrently online throughout the simulations.

As the initial setup, again according to [11], we chose 19% randomly sampled nodes to be online at the beginning, which initially were connected by shortest path tunnels with global knowledge. All results are averaged over 15 simulation runs, and presented with 95%-confidence intervals computed using the student-t distribution.

The results denote a steep incline of the length of tunnels (which each represent a single hop in the end-to-end overlay routing), even within the first 10.000 churn events. This corresponds to about ten minutes in the life of Freenet, according to [11], and represents the arrival or departure of only a fraction of the entire node set (cmp. Fig. 1). While the increase subsequently slows down, the experiments indicate that the asymptotic results apply already to networks of rather small size, and that virtual overlay approaches can not provide efficient routing even under idealized conditions, and suffer from degraded routing already after a very short period of operation.

VIII. Conclusion

Virtual overlays have been proposed to achieve polylog routing in networks without an underlying routing protocol, such as sensor networks or Darknets. We have shown that virtual overlays cannot achieve both polylog routing and maintenance cost under churn. As a consequence, it is only possible to design maintenance algorithms for virtual overlays that exceed polylog cost to yield polylog expected routing length. Such designs are still bound to achieve a high performance in specific environments characterized by a vast majority of routing over maintenance incidents, e.g. due to very low dynamics and churn. Generally, it is questionable if the high maintenance costs needed to provide efficient routing can be compensated in large-scale networks, without producing unacceptable congestion, overload, and delays.

However, the tunnel maintenance is only provable inefficient under the assumption that the overlay does not provide an underlying routing protocol such as IP. Efficient virtual overlays can hence potentially be achieved by establishing such a routing protocol using for example network embeddings such as [15]. Network embeddings adapt node identifiers to provide greedy routing directly on the original network. Since their performance in the face of churn as well as their impact on privacy insofar has not been satisfyingly analyzed, we will investigate if embeddings can yield polylog routing, and if they can be achieved at polylog cost in our future work.

Acknowledgments

The authors thank Sascha Grau for his helpful comments and advice. The presented work was partially funded by the German Research Foundation (DFG) Grant STR 1131/2-1.

References

[1] Matthew Caesar, Miguel Castro, Edmund B Nightingale, Greg O'Shea, and Antony Rowstron. Virtual ring routing: network routing inspired by dhts. In *SIGCOMM*, 2006.

[2] Eugene Vasserman, Rob Jansen, James Tyra, Nicholas Hopper, and Yongdae Kim. Membership-concealing overlay networks. In *CCS*, 2009.

[3] Abdalkarim Awad, Reinhard German, and Falko Dressler. Exploiting virtual coordinates for improved routing performance in sensor networks. *Mobile Computing, IEEE Transactions on*, 10(9):1214–1226, 2011.

[4] Prateek Mittal, Matthew Caesar, and Nikita Borisov. X-vine: Secure and pseudonymous routing using social networks. *arXiv preprint arXiv:1109.0971*, 2011.

[5] Ian Clarke, Oskar Sandberg, Matthew Toseland, and Vilhelm Verendel. Private communication through a network of trusted connections: The dark freenet. http://freenetproject.org/papers.html, 2010.

[6] Nathan S. Evans and Christian Grothoff. R5N: Randomized recursive routing for restricted-route networks. In *NSS*, 2011.

[7] Ion Stoica, Robert Morris, David Karger, M Frans Kaashoek, and Hari Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *SIGCOMM*, 2001.

[8] Petar Maymounkov and David Mazieres. Kademlia: A peer-to-peer information system based on the xor metric. In *Peer-to-Peer Systems*, 2002.

[9] Prateek Mittal, Matthew Wright, and Nikita Borisov. Pisces: Anonymous Communication using Social Networks. *arXiv preprint arXiv:1208.6326*, 2012.

[10] Chris Lesniewski-Lass and M Frans Kaashoek. Whanau: A Sybil-Proof Distributed Hash Table. NSDI, 2010.

[11] Stefanie Roos, Benjamin Schiller, Stefan Hacker, and Thorsten Strufe. Measuring Freenet in the Wild - Censorship-resilience under Observation. In *PETs*, 2014.

[12] Jon Kleinberg. The small-world phenomenon: an algorithm perspective. In *STOC*, 2000.

[13] Chip Martel and Van Nguyen. Analyzing Kleinberg's (and other) Small-World Models. In *PODC*, 2004.

[14] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P. Gummadi. On the evolution of user interaction in Facebook. In *WOSN*, 2009.

[15] Robert Kleinberg. Geographic routing using hyperbolic space. In *INFOCOM*, 2007.