Expansion and Flooding in Dynamic Random Networks with Node Churn

Luca Becchetti
Sapienza Università di Roma
Rome, Italy
becchetti@dis.uniroma1.it

Andrea Clementi
Università di Roma Tor Vergata
Rome, Italy
clementi@mat.uniroma2.it

Francesco Pasquale
Università di Roma Tor Vergata
Rome, Italy
pasquale@mat.uniroma2.it

Luca Trevisan
Università Bocconi
Milan, Italy
l.trevisan@unibocconi.it

Isabella Ziccardi
Università dell'Aquila
L'Aquila, Italy
isabella.ziccardi@graduate.univaq.it

2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS) | 978-1-6654-4513-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICDCS51616.2021.00097

***Abstract*—We study expansion and information diffusion properties of dynamic networks, i.e., networks whose topologies evolve over time as nodes enter or leave the system and edges are continuously created or destroyed. In this scenario, we investigate *flooding* as a basic information diffusion mechanism.**

We are interested in models that are likely to result in *sparse* networks, i.e., in networks containing $\mathcal{O}(n)$ edges, with n the number of nodes that are present at any given time of interest, with a focus on models in which edges are created randomly according to simple probabilistic mechanisms, rather than according to carefully designed distributed algorithms. In this perspective, in all models we consider, upon joining the network, a node connects to $d = \mathcal{O}(1)$ random nodes currently in the system. On the other hand, an edge remains alive as long as both its endpoints are.

For the case in which edges that fail (because one endpoint left the network) are not replaced, we show that, although the network is likely to contain $\Omega_d(n)$ isolated nodes, flooding still informs a fraction $1-\exp(-\Omega(d))$ of the nodes in time $\mathcal{O}(\log n)$ with large, constant probability. Moreover, we are able to show, that at any given time, the graph exhibits a "large-set expansion" property.

We further investigate models that exhibit *edge regeneration*, meaning that, whenever an edge (v, w) established by v fails because w leaves the network, it is replaced by a new random edge (v, z). We show that models with edge regeneration result in evolving networks that, at any given time, are vertex expanders with high probability, so that flooding takes $\mathcal{O}(\log n)$ time.

The above results hold both for a simplfied streaming model of node churn and in a more realistic, continuous-time setting, in which the interval between two consecutive node arrivals follows a Poisson distribution, while nodes' lifetimes follow an exponential distribution.

Previous work considered models in which either the vertex set is fixed or edges are established according to more or less sophisticated algorithms. Our motivation for studying models with simple and random edge creation mechanisms is to move one step further towards models that may eventually capture key aspects of the formation of social or peer-to-peer networks.

LT's work on this project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 834861). LB's work on this project was partially supported by the ERC Advanced Grant 788893 AMDROMA, the EC H2020RIA project "SoBigData++" (871042), the MIUR PRIN project ALGADIMAR. AC and FP's work on this project was partially supported by the ALBLOTECH project n. E89C20000620005 under the Tor Vergata University Programme "Beyond Borders".

I. Introduction

In this paper, we investigate *information diffusion* in *dynamic networks*, with a focus on *flooding*, the basic mechanism whereby each informed node in turn relays the information received to each of its neighbors. Flooding represents the fastest protocol for *broadcast*, a fundamental communication primitive in distributed systems.

We use the term *dynamic network* to denote communication networks that change over time, as nodes enter or leave the system and links between nodes are created or destroyed. Several important cases of information diffusion occur in networks that evolve over time, such as social or peer-to-peer networks.

Information diffusion in dynamic networks has been the focus of extensive previous work, surveyed in Section II. We are interested in models that i) exhibit *node churn* (that is, in which nodes enter and leave the network over time) and ii) in which edges are created randomly, rather than according to sophisticated distributed algorithms. Our motivation is that a satisfactory modeling of network formation in social networks and peer-to-peer networks will have to meet both requirements. To the best of our knowledge, information diffusion in dynamic networks with node churn and with random, uniform edge creation was not studied before.

We kept all other modeling choices as simple as possible, defining models with as few parameters as possible, in order to identify qualitative features that we believe might prove robust to different modeling choices. If, as a result, our models are too simplistic to reflect all properties of realistic networks, one of the models we consider (the Poisson model with edge regeneration defined below) bears a certain resemblance to the way peer-to-peer networks such as bitcoin are formed [20], [24].

We study models in which the network is *sparse*, meaning that it has $\mathcal{O}(n)$ edges, where n is the number of nodes at any given time under consideration. Specifically, when a node

2575-8411/21/$31.00 ©2021 IEEE
DOI 10.1109/ICDCS51616.2021.00097

is "born," it connects to $d = \mathcal{O}(1)$ random nodes. We show that dynamic random graphs created in this fashion maintain interesting expansion properties over time, with flooding informing all or most nodes (depending on details of the model) in $\mathcal{O}(\log n)$ time.

A. Modeling dynamic networks

A dynamic network model is specified by describing i) how nodes enter and leave the network and ii) how edges are generated and destroyed.

a) Modeling node churn: We initially study an unrealistic, yet very simple and insightful model of node churn: at each discrete time unit, one node enters the network and each node is alive for exactly n time units. We refer to this as a *streaming* model of node churn. After the first n time units, the network always has exactly n nodes, and precisely one node is born and one node dies in each time unit. We then study a more realistic continuous-time model, in which the number of nodes' births within each time unit follows a Poisson distribution with mean λ, while the lifetime of each node is independently distributed as an exponential distribution with parameter μ, so that the average lifetime of a node is $1/\mu$ and the average number of nodes in the network at any given time is λ/μ. In order to reduce the number of parameters, we assume that the time it takes to deliver a message along an edge is the same, or the same order as, the typical time between consecutive node births, which is 1 in the streaming model and λ in the Poisson model. In order to use a consistent notation for the two models, we choose time units in the continuous model so that $\lambda = 1$, and we set $n = 1/\mu$. With these conventions, the results we prove in the streaming model extend to the continuous time model (despite proofs becoming considerably more technical in some cases), suggesting that the results exhibit a certain robustness and that the streaming model, despite its simplicity, might have some predictive power on the behavior of more realistic models.

b) Modeling edge creation and destruction: When a node enters the network, we assume that it connects to $d = \mathcal{O}(1)$ nodes chosen uniformly at random among those currently in the network. Once an edge (u, v) is created, it remains active as long as both u and v are alive. We consider two models: in the one *without edge regeneration* edges are created only when a new node joins the network. In the model *with edge regeneration*, a node creates its outgoing edges not only upon joining the network, but also every time it looses an outgoing edge because one of its neighbors left the network, so that the out-degree of every node is always d.

Remark. Although the assumption that a node can pick its neighbors uniformly at random among all nodes of the network is unrealistic in many scenarios, the edge creation and regeneration processes in our models are reminiscent of the way some unstructured peer-to-peer networks maintain a "random" topology. For example, each *full-node* of the Bitcoin network running the Bitcoin Core implementation has a "target out-degree value" and a "maximum in-degree value" (respectively 8 and 125, in the default configuration) and it locally stores a large list of (ip addresses of) "active" nodes. This list is initialized with nodes' addresses received in response to queries to some DNS seeds. Whenever its number of current neighbors falls below the configured target value, a full-node tries to create new connections with nodes sampled from its stored list of active nodes. The list stored by a full-node is periodically advertised to and updated with the lists advertised from its neighbors. Hence, in the long run, each full-node samples its out-neighbors from a list containing a "sufficiently random" subset of all nodes in the network [9], [24].

B. Results and techniques

In the next two subsections, we describe our positive and negative results, summarized in Table I, for all models we consider.

1) Models without edge regeneration: For models without edge regeneration, we prove that, at any given time, *with high probability*[1] (for short, *w.h.p.*), there are $\Omega_d(n)$ isolated nodes in the network. A node v becomes isolated if i) all d edges it created at birth were to destinations that died in the interim and ii) v was never chosen as a neighbor by a younger node.

Because of the presence of such isolated nodes, we can show the following two statements for a flooding process initiated at time t, for sufficiently large t: (i) there is a constant probability that the flooding process dies out after informing only $\mathcal{O}(1)$ nodes; ii) W.h.p. the *flooding time* is $\Omega_d(n)$ in the streaming model and $\Omega_d(n \log n)$ in the Poisson model, where, as usual, the flooding time is defined as the time required by flooding to broadcast the source message to all nodes. We remark here that flooding is the most resilient and fastest broadcast protocol: hence, the statements above hold for any broadcast protocol as well.

On the other hand, we show that, with large constant probability (in detail, $1 - \exp(-\Omega(d))$, thus tending to 1 as $d \to \infty$), flooding will broadcast the source message to, say, 90% (in general, a fraction $1 - \exp(-\Omega(d))$ of the nodes, thus again tending to 1 as $d \to \infty$) in $\mathcal{O}(\log n)$ time.

To prove such fast convergence we establish two results. One is that, in $\mathcal{O}(\log n)$ time, flooding reaches at least, say, $n/10$ nodes. To prove this, we argue that, as long as the number of informed nodes is below $n/10$, there is a sufficiently high probability that the number of informed nodes grows by a constant factor in each step (the probability that the above condition fails after exactly t steps decreases exponentially with t, so that we can take a union bound over all t). The basic idea of this proof is to apply the principle of deferred decisions to the d edges chosen by each node, assuming that those edges are chosen upon informing the destination node itself, so that the "frontier" of newly informed nodes keeps growing. Unfortunately, two problems arise when implementing this approach. One is that, in the generic step of diffusion, older nodes are more likely to have chosen neighbors that have

[1] As usual, we say that an event E holds with high probability if a constant $\gamma > 0$ exists such that $\mathbf{P}(E) \geqslant 1 - n^{-\gamma}$.

		Poisson/Streaming dynamic graphs	
		without Edge Regeneration	with Edge Regeneration
Expansion properties	Negative results	Constant fraction of isolated nodes (w.h.p.)	—
	Positive results	$\Omega(1)$-expansion of big-size node subsets (w.h.p.)	$\Omega(1)$-expansion (w.h.p.)
Flooding	Negative results	Flooding may not complete, with probability $\Omega(1)$	—
	Positive results	Flooding informs a fraction $1-\exp(-\Omega(d))$ of the nodes in $O(\log n)$ time, with probability $1-(\exp(-\Omega(d)))$	Flooding time is $O(\log n)$ (w.h.p.)

TABLE I
SUMMARY OF OUR RESULTS.

meanwhile died, so that they are unlikely to significantly contribute to the number of newly informed nodes in the next step. The second difficulty is that a node might become informed by a message received from one of the d neighbors it chose at birth, so that we cannot really apply deferred decision in the way we would like.

To sidestep these difficulties, we consider a modified diffusion process, which underestimates the number of truly informed nodes in every step, but at the same time removes dependencies that would be otherwise hard to tackle. Specifically, we only consider paths of diffusion that alternate between "young" nodes, whose ages are below the median age, and "old" nodes, whose ages are above the median. Moreover, we arbitrarily split the d edges established by each node at birth into two subsets of $d/2$ "type-A" edges and $d/2$ "type-B" edges, so that only edges of one type are used to move back and forth between young and old nodes in each step of alternating path construction. This way, some cumbersome dependencies are removed and we able to apply deferred decisions again.

Using the above approach, we are able to show that at least $n/10$ nodes are informed within $\mathcal{O}(\log n)$ steps. To complete the argument, we show that, if d is a sufficiently large constant, all node sets of cardinality at least $n/10$ have constant node expansion, which leads to informing at least $.9n$ nodes within $\mathcal{O}(1)$ further steps. In the argument sketched above, $1/10$ can be replaced by $\exp(-\Omega(d))$. This tradeoff is best possible since, as argued above, $\Omega_d(n)$ nodes will be isolated and thus will not be reachable by the flooding process.

2) Models with edge regeneration: In the model with edge regeneration we show that, in each time step, w.h.p., the graph has constant node expansion. This in turn implies that the flooding protocol informs all nodes within $\mathcal{O}(\log n)$ steps, despite the presence of node churn.

In the streaming model, since the number of nodes is fixed in each step, the proof of node expansion proceeds along the lines generally used to prove expansion in static random graphs: we bound the probability that a fixed set of k nodes fails to have constant node expansion, then we take a union bound by multiplying by $\binom{n}{k}$ and then by summing over k. The main difficulty is in characterizing the probability that an edge exists between a pair of nodes u, v, because it is a non-trivial function of the ages of u and v. On the other hand, since node churn is limited and deterministic in the streaming model, the result on node expansion easily lends a logarithmic bound on flooding time.

The analysis becomes considerably more technical in the Poisson model. The main difficulty is that, in order to compute the probability that an edge (u, v) exists, we need to argue about the ages of u and v, which brings to considering a union bound over all possible demographics of the nodes (i.e., age distributions). In the Poisson model however, there is constant probability that nodes of age up to $n \log n$ exist. The point is that most of the $\binom{n}{k}$ ways of choosing k nodes of all possible ages involve the choice of many old nodes, which are unlikely to have all survived. In order to carefully account for the "demographics" of all possible sets of edges in our union bound, we consider the logarithm of the probability that a certain set fails to expand, interpret it is as the KL divergence of two appropriately defined distributions, and then leverage properties of the KL divergence itself. Finally, derivation of a bound on flooding time from node expansion requires more technical care in the Poisson model, since we need to account for the presence of a random number of node insertions/deletions during each 1-hop message transmission.

C. Roadmap

Due to (strict) page limits, the rest of this extended abstract is devoted to two technical contributions, which we believe are the most innovative and interesting ones. The first one is the analysis of the flooding process via the alternating path argument (the so called onion-skin process) in the model without edge regeneration: In Section III, we describe this analysis in the (simpler) streaming model, since it essentially proposes the main ingredients for both the streaming and Poisson model, without some complications of the latter. In Section IV, we describe our main idea to establish node expansion for the model with edge regeneration, using a notion of edge subset "demographics", which is quantified using the KL divergence. In order to discuss some important technical issues of this analysis and our approach to tackle them, we present the analysis for the (more complex) Poisson model.

Remark. The full version of our paper, including all technical statements and their proofs, is available in the ArXiv Repository [5].[2]

II. Related Work

A first, rough classification of dynamic graphs follows a key feature: whether or not the set of nodes is static or evolves over

[2]Available at https://arxiv.org/abs/2007.14681

time. In the former case, we have an *edge-dynamic graph*, in which a topology dynamics defines the way the edges spanning a fixed set V of nodes change over time. For this class of dynamic graphs, several models [6], [7], [15], [16], [18], have been introduced and rigorously analyzed.

In contrast, the case in which the set of nodes changes over time received less attention in the past. This class of dynamic graphs $\{G_t = (V_t, E_t), t \geqslant 0\}$ are often called *dynamic networks with churn* [1]: in this framework, a specific graph dynamics describes both the node arrival/departure rule for the time sequence V_t and the edge updating rule for the time sequence E_t. The number of nodes that can join or leave the network in every round is called *churn rate*. For brevity's sake, in the remainder we only discuss previous analytical results on dynamic networks with churn that are more closely related to the models we investigate in this paper. In particular, we mostly focus on previous works that rigorously analyzed connectivity properties of dynamic networks with churn.

As remarked in the introduction, to the best of our knowledge, previous analytical studies focus on distributed algorithms that are suitably designed to maintain topologies with desired connectivity properties.

In this line of work, Pandurangan et al. [22] introduced a partially-distributed protocol that constructs and maintains a bounded-degree graph which relies on a centralized cache of a constant number of nodes. In more detail, their protocol ensures the network is connected, has logarithmic diameter, and has always bounded degree. The protocol manages a central cache which maintains a subset of the current set of nodes. When joining the network, a new node chooses a constant number of nodes in the cache. The centralized cache update procedures follow rather complex rules that require $\mathcal{O}(\log n)$ overhead and delays, w.h.p.

In [13], Duchon et al presented ad-hoc protocols that maintain a given distribution of random graphs under an arbitrary sequence of node insertions and deletions. More in detail, given that the graph G_t is random uniform over the set of k-out-degree graphs with n nodes, they provide suitable distributed randomized protocols that can insert (respectively delete) a node such that the graph G_{t+1} at round t is again random uniform over the set of k-out-degree graphs with $n+1$ (respectively, $n-1$) nodes. They do not assume a centralized knowledge of the whole graph but, instead, their protocol relies on some random primitives to sample arbitrary-sized subsets of nodes uniformly at random. For instance, once a new node u is inserted, a random subset of nodes is selected (thanks to one of such centralized primitives). Then, each of the sampled nodes in turn deletes one of its links deterministically connects to u. The basic versions of their insertion/deletion procedures require each node to communicate with nodes at distance 2, while more sophisticated variants (achieving optimal performance) require communications over longer paths.

An important and effective approach to ensure good expansion properties in dynamic graphs with churn relies on the use of ID random walks. Roughly speaking, in this approach, every participating node starts k independent random walks to circulate tokens containing its ID. All other nodes cooperate to perform these random walks long enough, so that each token is well-mixed over the network. Once a token is mature, it can be used by any node that, in that step, needs a new edge by simply connecting to the corresponding ID. The probabilistic analysis then typically shows two main, correlated invariants: on one hand, the edge set resulting from the above process corresponds to a random graph exhibiting good expansion properties. On the other hand, random walks are well-mixed within a small number of steps. In this line of work, Cooper et al [8] consider two deterministic churn processes: in the first one, at every round, a new node is inserted while no nodes leave the network, while, in the second process, the size n of the graph never changes since, at every round, a new node is inserted and the oldest node leaves the graph (this is in fact the streaming model we study in this paper). They provide a protocol where each node v starts $c \cdot m$ independent random walks (containing the ID-label of v) until they are picked up, m at a time, by new nodes joining the network. The new node connects to the m peers that contributed the tokens it got. The resulting dynamic topology is shown to have diameter $\mathcal{O}(\log n)$, and to be fault-tolerant against adversarial deletion of both edges and nodes. We remark that the tokens in the graph must be constantly circulated in order to ensure that they are well-mixed. Moreover, the rate at which new nodes can join the system is limited, as they must wait while the existing tokens mix before they can use them.

Law and Siu [17] provide a distributed algorithm for maintaining a regular expander in the presence of a limited number of insertions/deletions. The algorithm is based on a complex procedure that is able to sample uniformly at random from the space of all possible $2d$-regular graphs formed by d Hamiltonian circuits over the current set of alive nodes. They present possible distributed implementations of this sample procedure, the best of which, based on random walks, have $\mathcal{O}(\log n)$ overhead and time delay. Such solutions cannot manage frequent node churn.

Further distributed algorithms with different approaches achieving $\mathcal{O}(\log n)$ overhead and time delay in the case of slow node churn are proposed in [4], [14], [19], [23].

In [2], Augustine et al present an efficient randomized distributed protocol that guarantees the maintenance of a bounded degree topology that, with high probability, contains an expander subgraph whose set of nodes has size $n - o(n)$, where n is the stable network size. This property is preserved despite the presence of a large oblivious adversarial churn rate — up to $\mathcal{O}(n/\mathrm{polylog}(n))$. In more detail, considering the node churn adopted in [3], i.e., an oblivious churn adversary that: can remove any set of nodes up to the churn limit in every round, and, at the same time, it should add (an equal amount of) nodes to the network with the following constraints. A new node should be connected to at least one existing node and the number of new nodes added to an existing node should not exceed a fixed constant (thus, all nodes have constant bounded degree).

The expander maintenance protocol is efficient even though it is rather complex and the local overhead for maintaining the topology is polylogarithmic in n. A complication of the protocol follows from the fact that, in order to prevent the emergence of large clusters of nodes outside the expander subgraph, specific criteria to "refresh" the links of some nodes are needed, even when these were not involved in any edge deletions following from node churn.

Recently, flooding has been analytically investigated in dynamic graph models with churn in [1], [3]. Here, the authors consider the model analysed in [2], that we discussed above. Using the expansion property proved in [2], they show that, for any fixed churn rate $C(n) \leqslant n/\mathrm{polylog}(n)$ managed by an oblivious worst-case adversary, there is a set S of size $n - \mathcal{O}(C(n))$ of nodes such that, if a source node in S starts the flooding in round t, then all except $\mathcal{O}(C(n)$ nodes get informed within round $t + \mathcal{O}(\log(n/C(n)) \log n)$, w.h.p.

Our models are inspired by the way some unstructured P2P networks maintain a "well-connected" topology despite node arrivals and departures, in an almost fully decentralized fashion and typically with small average degree. For example, after an initial bootstrap in which they rely on DNS seeds for node discovery, full-nodes of the Bitcoin network [20] running the Bitcoin Core implementation turn to a fully-decentralized policy to regenerate their neighbors when their degree drops below a configured threshold [9]. This allows them to sample new neighbors essentially at random among all nodes of the network [24]. In this respect, it should be noted that the real topology of the Bitcoin network is hidden by the network formation protocol and discovering the real network structure has been recently an active research area [11], [21].

III. The Streaming Model Without Edge Regeneration

A discrete-time *dynamic graph* $\mathcal{G}$ is a sequence of graphs $\mathcal{G} = \{G_t = (N_t, E_t) : t \in \mathbb{N}\}$ in which the sets of nodes and edges can vary in each discrete round. If they can change randomly, we call the corresponding random process a *dynamic random graph*. We call G_t the *snapshot* of the dynamic graph at round t. For a set of nodes $S \subseteq N_t$, we denote by $\partial^t_{out}(S)$ the outer boundary S in snapshot G_t, i.e., $\partial_{out}(S) = \{v \in N \setminus S : \{u, v\} \in E \text{ for some } u \in S\}$; we omit superscript t when it is clear from context. In this section, we study a dynamic random graph model in which nodes join and leave the network according to a *deterministic streaming* process, while edges are created randomly by the nodes.

Definition III.1 (Streaming node churn)**.** *The set of nodes N_t evolves as follows: It starts with $N_0 = \emptyset$; At each round $t \geqslant 1$ a new node joins the network and it stays in the network for exactly n rounds (i.e., node joining at round t stays up to round $t + n - 1$), then it leaves the network (or* die*). We say that a node has* age *k at round t if it joined the network at round $t - k$. We say that a node u is* older *(respectively,* younger*) than a node v if u joined the network before (respectively, after) v.*

In the streaming model SDG edges are created only when a new node joins the network, more precisely:

Definition III.2. *A* Streaming Dynamic Graph without edge regeneration *(for short, SDG) $\mathcal{G}(n, d)$ is a dynamic random graph $\{G_t = (N_t, E_t) : t \in \mathbb{N}\}$ where the set of nodes N_t evolves according to Definition III.1, while the set of edges E_t obeys the following topology dynamics: i) When a new node appears, it establishes d independent connections, each one with a node chosen uniformly at random among nodes currently in the network[3]. ii) When a node dies, all its incident edges disappear.*

A remark on terminology. To the purpose of communication, the graphs we consider are always undirected. However, to the purpose of the analysis that follows in this section and in Section IV, it is useful to associate a direction to an edge, depending on which of its endpoints was responsible for its creation. More precisely, considered any active node v, its *out-edges* are the edges that were established by v towards other nodes (for this reason, we sometimes use the term *request* when discussing the event of an edge being established), while v's *in-edges* are edges incident to v that were established by other nodes with v as destination. Moreover, here and in Section IV, the terms *birth* and *arrival* are used interchangeably to refer to the event of a node joining the network, while the terms *death* and *departure* are used to denote the event of a node leaving the network.

A. Partial broadcast in the SDG model

We are interested in estimating the time a message sent by a node takes to reach a large fraction of the nodes. To this end, we formalize the *flooding process* over a discrete-time dynamic (random) graphs.

Definition III.3 (Flooding)**.** *Let $\mathcal{G} = \{G_t = (N_t, E_t) : t \in \mathbb{N}\}$ be a dynamic (random) graph. The flooding process over $\mathcal{G}$ starting at time t_0 from the source node $v_0 \in N_{t_0}$ is the sequence of (random) sets of nodes $\{I_t : t \in \mathbb{N}\}$ where, $I_t = \emptyset$ for all $t < t_0$, $I_{t_0} = \{v_0\}$[4] and, for every $t \geqslant t_0$, I_t contains all nodes in N_t that were neighbors of some node in I_{t-1} in the snapshot G_{t-1}, i.e.,*

$$I_t = (I_{t-1} \cup \partial^{t-1}_{out}(I_{t-1})) \cap N_t \,.$$

We say that I_t is the subset of informed *nodes at round t.*

We next show that there is a large constant probability that the flooding process will reach a large fraction of nodes within $\mathcal{O}(\log n)$ rounds.

Theorem III.4 (Flooding completes for a large fraction of nodes)**.** *For every constant[5] $d > 200$, for every suf-*

[3]To the sake of simplicity, in all our models, we allow the presence of multi-edges.

[4]In this paper we will assume that I_0 contains the node joining the network at round t_0.

[5]Our probabilistic analysis definitely does not aim at optimizing absolute constants. Moreover, though it is an absolute constant, in the upper bound on τ that follows we leave the exact, asymptotic dependence on d for the sake of completeness.

ficiently large n and for every fixed $t_0 \geqslant n$, there is a $\tau = \mathcal{O}(\log n/\log d + d)$, such that the flooding process over an SDG sampled from $\mathcal{G}(n,d)$ starting at t_0 satisfies the following:

$$\mathbf{Pr}\left(|I_{t_0+\tau}| \geqslant (1-e^{-d/10})n\right) \geqslant 1 - 4e^{-d/100} - o(1)\,.$$

We now give a detailed outline of the proof of the above result, since this allows us to present, in a simplified setting, some key technical ingredients that are also used for the Poisson model.

Proof of Theorem III.4: The proof consists of two parts. Assuming the source node s joined the network in round t_0, we first show (Lemma III.5) that, with probability at least $1 - 4e^{-d/100}$, a restriction of the true topology dynamics establishes a bipartite graph which i) contains s, ii) only connects nodes with ages in the interval $\{1,\ldots,n/2\}$ to nodes with ages in the interval $\{n/2+1,\ldots,n-\log n\}$, iii) has diameter $\mathcal{O}(\log n/\log d)$, iv) includes at least $2n/d$ nodes. This is enough to prove that, with probability $1-4e^{-d/100}$, $2n/d$ nodes are informed at time $t_0+\tau_1$, where $\tau_1 = \mathcal{O}(\log n)$.

The second part in turns consists of two steps: i) we first show that every sufficiently large subset of nodes has high expansion, w.h.p. ii) Thanks to this expansion property of large subsets, once $2n/d$ nodes have been informed, at least $(1-e^{-d/10})n$ nodes will become informed within a constant number $\tau_2 = \Theta(d)$ of additional steps, w.h.p.

Overall, the above prove that within time $t_0+\tau_1+\tau_2$, at least $(1-e^{-d/10})n$, nodes have been informed, with probability at least $1-4e^{-d/100}-o(1)$.

For the sake of space, we only describe the first part of the proof, corresponding to the following lemma.

Lemma III.5 (Flooding completes for a large fraction of nodes, phase 1)**.** *Under the hypotheses of Theorem III.4, there is a $\tau_1 = \mathcal{O}(\log n/\log d)$ such that*

$$\mathbf{Pr}\left(|I_{t_0+\tau_1}| \geqslant \frac{2n}{d}\right) \geqslant 1 - 4e^{-\frac{d}{100}}\,. \tag{1}$$

Proof. We begin by defining the following subsets of N_{t_0}:

- the set of the *young nodes*: $Y = \{v \in N_{t_0} \mid v \text{ has life } l \text{ with } 2 \leqslant l < \frac{n}{2}\}$.
- the set of the *old nodes*: $O = \{v \in N_{t_0} \mid v \text{ has life } l \text{ with } \frac{n}{2} \leqslant l \leqslant n - \log n\}$.
- the set of the *very old nodes*: $\hat{O} = N_{t_0} - (Y \cup O) = \{v \in N_{t_0} \,|\, v \text{ has life } l \text{ with } n - \log n < l \leqslant n\}$.

To prove (1) we show that $G_{t_0} = (N_{t_0}, E_{t_0})$ contains a bipartite subgraph with logarithmic diameter, containing the informed node s and such that i) links are established only between nodes in Y and in O and ii) it contains no very old node. The graph in question is the result of the *onion-skin* process described below.

a) The onion-skin process.: The iterative process we consider operates in phases, each consisting of two steps. Starting from s, the onion-skin process builds a connected, bipartite graph, corresponding to alternating paths in which young nodes only connect to old ones. In particular, each realization of this process generates a subset of the edges generated by the original topology dynamics. Moreover, each iteration of the process corresponds to a partial flooding in the original graph, in which a new layer of informed nodes is added to the subset of already informed ones, hence the term onion-skin. Flooding is partial since i) the network uses a subset of the edges that would be present in the original graph.

In the following, we denote by $Y_k \subseteq Y$ and $O_k \subseteq O$ the subsets of young and old nodes that are informed by the end of phase k, respectively. In the remainder, we let $O_{-1} = \emptyset$ for notational convenience.

Onion-skin process

Phase **0**: $Y_0 = \{s\}$; O_0 is obtained as follows: s establishes d links. We let $O_0 \subset O$ denote the subset of old nodes that are destinations of these links. Links with endpoints in Y or $\hat{O}$ are discarded;

Phase $\mathbf{k \geqslant 1}$: Y_k and O_k are iteratively obtained as follows:

Step 1. Each node in $Y - Y_{k-1}$ establishes $d/2$ links. More precisely:

$$Y_k \setminus Y_{k-1} = \{v \in Y \setminus Y_{k-1} \mid v \text{ connects to } O_{k-1} \text{ by a request } i \in \{\tfrac{d}{2}+1,\ldots,d\}\}$$

Links to nodes not belonging to O are discarded;

Step 2. Each node in $Y_k \setminus Y_{k-1}$ establishes $d/2$ links to nodes in $O \setminus O_{k-1}$. More precisely:

$$O_k \setminus O_{k-1} = \{v \in O\backslash O_{k-1} \mid \text{some } w \in Y_k \text{ connects to } v \text{ by a request } i \in \{1,\ldots,\tfrac{d}{2}\}\}$$

Links to nodes not belonging to O are discarded.

A couple remarks are in order. It is clear that the links in E_{t_0} can be established in any order, as long as they are created from younger nodes towards older ones. As a consequence, each realization of the onion-skin process produces a subset of E_{t_0}. In particular, i) nodes in O and $\hat{O}$ do not create any links, though they can still be the targets of links originating from Y; ii) a node $v \in Y$ created at time $\hat{t}$ ($\leqslant t_0$) generates d links, with possible destinations the nodes created in the interval $[\hat{t}, t_0]$, but only links with destinations in O are retained, the others are discarded.

The next claim states that, in each step, the sets of informed nodes $Y_k \subseteq Y$ and $O_k \subseteq O$ grow by a constant factor $d/20$. The proof proceeds by analyzing Phases 0 and the generic Phase k separately. Concentration bounds are derived using the method of bounded differences (see for instance [12]).

Claim III.6. *In Phase 0, it holds* $\mathbf{Pr}\left(|O_0| \geqslant d/20\right) \geqslant 1 - e^{-d/100}$. *In the generic phase $k \geqslant 1$, if $|Y_{k-1}| \leqslant n/d$, $|O_{k-1}| \leqslant n/d$, $y \leqslant 10n/d$ and $x \leqslant 20(n/2 - \log n)/d$ it holds:* $\mathbf{Pr}\left(|Y_k \setminus Y_{k-1}| > \frac{d}{20}y \mid |O_{k-1} \setminus O_{k-2}| \geqslant y\right) \geqslant 1 - e^{-yd/100}$,

and
$\mathbf{Pr}\left(|O_k \setminus O_{k-1}| \geqslant \frac{d}{20}x \mid |Y_k \setminus Y_{k-1}| \geqslant x\right) \geqslant 1 - e^{-dx/100}$.

Using the above claim and the chain rule, we obtain for each $k \geqslant 0$,

$$\mathbf{Pr}\left(|O_k \setminus O_{k-1}| \geqslant a_{2k+1}\right) \geqslant \prod_{i=0}^{2k}\left(1 - e^{-a_i(d/100)}\right)$$

$$\mathbf{Pr}\left(|Y_k \setminus Y_{k-1}| \geqslant a_{2k}\right) \geqslant \prod_{i=0}^{2k}\left(1 - e^{-a_i(d/100)}\right),$$

where $a_k = \left(\frac{d}{20}\right)^k$ and as long as a_{2k} and a_{2k+1} are smaller than n/d. Then, after some $\tau_1 = \log n/\log d$ rounds, we get $|Y_{t_0+\tau_1}| \geqslant n/d$ and $|O_{t_0+\tau_1}| \geqslant n/d$, with probability at least

$$c = \prod_{i=0}^{\infty}\left(1 - e^{-a_i(d/100)}\right).$$

By using standard calculus, we can then prove the following claim, which concludes the proof.

Claim III.7. *For each $d > 200$, if $a_i = (d/20)^i$,*

$$c = \prod_{i=0}^{\infty}\left(1 - e^{-a_i(d/100)}\right) \geqslant 1 - 4e^{-\frac{d}{100}}.$$

□

IV. The Poisson Model With Edge Generation

A *continuous dynamic graph* $\mathcal{G}$ is a continuous family of graphs $\mathcal{G} = \{G_t = (N_t, E_t) : t \in \mathbb{R}^+\}$ in which the sets of nodes and edges can change at any time $t \in \mathbb{R}^+$. As in the discrete case, we call G_t the *snapshot* of the dynamic graph at time t and, for a set of nodes $S \subseteq N_t$, we denote with $\partial^t_{out}(S)$ the outer boundary of S (see Definition IV.8) in snapshot G_t and we omit subscript t when it is clear from context. We consider continuous-time dynamic graph models in which nodes' arrivals follow a Poisson process and their lifetimes obey an exponential distribution.

Definition IV.1 (Poisson node churn)**.** *Initially $N_0 = \emptyset$. Node arrivals in N_t follow a Poisson process with mean λ. Moreover, once a node joins the network, its lifetime has exponential distribution with parameter μ.*

We first discuss useful properties of Poisson dynamic graphs that only depend on the random node churn process. We remark that, according to Definition IV.1 above, the time interval between two consecutive node arrivals is an exponential random variable of parameter λ, while the number of nodes joining the network in a time interval of duration τ is a Poisson random variable with expectation $\tau \cdot \lambda$. We finally note that the stochastic continuous process $\{N_t : t \in \mathbb{R}^+\}$ is clearly a continuous Markov Process. A first important fact our analysis relies on is that we can bound the number of active nodes at every time. In particular, it is easy to show that $\mathbf{E}[|N_t|] \to \lambda/\mu$. Moreover, we have the following bound in concentration.

Lemma IV.2 (Pandurangan et al. [22] - Number of nodes in the network)**.** *For every pair of parameters λ and μ such that $n = \lambda/\mu$ is sufficiently large, the following holds for the Poisson node churn $\{N_t : t \in \mathbb{R}^+\}$ defined in Definition IV.1: for every fixed real $t \geqslant 3n$, $|N_t| = \Theta(n)$ w.h.p. More precisely,*

$$\mathbf{Pr}\left(0.9n \leqslant |N_t| \leqslant 1.1n\right) \geqslant 1 - 2e^{-\sqrt{n}}.$$

Leveraging Lemma IV.2, our analysis of the Poisson considers the setting $\lambda = 1$ without loss of generality. In the remainder, we define the key parameter $n = \frac{1}{\mu}$ representing the "expected" size of the network. Moreover, since the probability that two or more churn events occur at the same time is zero, the points of change of the dynamic graph yield a discrete-time sequence of *events*. In particular, we can observe and prove properties of the dynamic graph by restricting ourselves to time instants in which one event changing the graph occurs, namely, the arrival of a new node or the departure of an existing one.

Definition IV.3. *Let $\{N_t : t \in \mathbb{R}^+\}$ be the Poisson node churn process from Definition IV.1. We define the infinite sequence of random* steps *(also called* rounds*) $\{T_r : r \in \mathbb{N}\}$ (with parameters λ and μ) as follows: $T_0 = 0$, and $T_{r+1} = \inf\{t > T_r : N_t \neq N_{T_r}\}$, for $r = 0, 1, \ldots$*

It is worth noting that, since the Poisson stochastic process $\{N_t : t \in \mathbb{R}^+\}$ is a countinuous Markov process, the stochastic process $\{N_{T_r} : r \in \mathbb{N}\}$, with the T_r's defined above, consistently is a discrete Markov chain. It is then possible to derive the law of the random variables that define the time steps at which new events occur and get the following results.

Lemma IV.4. *For every sufficiently large n, consider the Markov chain $\{N_{T_r}, r \in \mathbb{N}\}$ in Definition IV.3 with parameters $\lambda = 1$ and $\mu = 1/n$. Then, for every fixed integer $r \geqslant n \log n$,*

$$0.47 \leqslant \mathbf{Pr}\left(|N_{T_{r+1}}| = |N_{T_r}| - 1\right) \leqslant 0.53$$
$$0.47 \leqslant \mathbf{Pr}\left(|N_{T_{r+1}}| = |N_{T_r}| + 1\right) \leqslant 0.53.$$

Moreover, if $v \in N_{T_r}$,

$$\frac{0.45}{n} \leqslant \mathbf{Pr}\left(v \notin N_{T_{r+1}} \mid v \in N_{T_r}\right) \leqslant \frac{0.56}{n}. \qquad (2)$$

The next lemma provides a useful bound on the lifetime of any node in the network.

Lemma IV.5 (Lifetime of the nodes)**.** *For every sufficiently large n consider the Markov chain $\{N_{T_r}, r \in \mathbb{N}\}$ in Definition IV.3 with parameters $\lambda = 1$ and $\mu = 1/n$. Then, for every fixed integer $r \geqslant 7n \log n$, with probability at least $1 - 2/n^{2.1}$, each node in N_{T_r} was born after step $T_{r-7n\log n}$.*

A. Node expansion of Poisson graphs with edge regeneration

We next consider graph dynamics in which active nodes replace out-edges that fail as their endpoints leave the network.

Definition IV.6. *A* Poisson Dynamic Graph with edge Regeneration $\mathcal{G}(\lambda, \mu, d)$ *(for short, PDGR) is a continuous dynamic*

random graph $\{G_t = (N_t, E_t) : t \in \mathbb{R}^+\}$ where the set of nodes N_t evolves according to Definition IV.1, while the set of edges E_t evolves according to the following topology dynamics: i) When a new node appears, it creates d independent connections, each of them with a node chosen uniformly at random among the nodes currently in the network. ii) When a node dies, all its incident edges disappear. iii) When one of the d out-edges of a node v fails,[6] v establishes a new connection with a node chosen uniformly at random among all nodes currently in the network.

Our first technical step is to provide an upper bound on the probability that a fixed node chooses any other active node in the network as destination of one of its d requests/out-edges. A major technical issue to cope with here is the presence of "very old" nodes (i.e. nodes having ages $\omega(n)$). In fact, a very old nodes can be selected as destination of a link request from a younger node with probability $\omega(1/n)$. The next lemma formalizes this fact as function of the age of the nodes.

Lemma IV.7. *For every constant $d \geqslant 20$ and for every sufficiently large n, let $\{G_t = (N_t, E_t) : t \in \mathbb{R}^+\}$ be a PDGR sampled from $\mathcal{G}(\lambda, \mu, d)$ with $\lambda = 1$ and $\mu = 1/n$. Then, for every fixed integer $r \geqslant 7n \log n$, consider the snapshot G_{T_r}. Let $u \in N_{T_r}$ be a node born in round T_{r-i} for some integer $i \leqslant r$. Then, if another node $v \in N_{T_r}$ was born before u, the probability that a single, specific request of u has v as its destination is at most*

$$\frac{1.3}{n}\left(1 + \frac{0.6i}{n}\right). \tag{3}$$

While, if v was born after u, the probability that a single request of u has destination v is always $\leqslant \frac{1.3}{n}$.

Proof. We define the following events:

$$A_{u,v} = \{\text{a specific request of } u \text{ has destination } v \text{ at time } T_r\}.$$

Notice that we avoid to index a specific request in the above definition, since the considered graph process is perfectly symmetric w.r.t. the d random requests made by every node. We first bound the probability that a fixed, single request of u has destination v when v is younger than u. To this purpose, we define L_r as the event

$$L_r = \{\text{each node in } N_{T_r} \text{ was born after time } T_{r-7n\log n}\} \cap \{|N_{T_i} \in [0.9n, 1.1n] \text{ with } i = r - 7n\log n, \ldots, r\}.$$

Note that the event L_r implies that, when each node in N_{T_r} joined the network, the network consisted of at least $0.9n$ and at most $1.1n$ nodes. Now, Lemma IV.2 and Lemma IV.5 together imply $\mathbf{Pr}(L_r) \geqslant 1 - 1/n^2$. So, $\mathbf{Pr}(A_{u,v} \mid L_r) \leqslant 1/0.9n$ since, being v younger than u, u can choose v only after the death of one of its neighbors. From the law of total probability, we then have that $\mathbf{Pr}(A_{u,v}) \leqslant 1/(0.8n)$.

[6]I.e., the other endpoint of the edge left the network.

We next analyze the case in which v is older than u, while u was born in step T_{r-i}. Again from the law of total probability we have:

$$\mathbf{Pr}(A_{u,v}) \leqslant \mathbf{Pr}(A_{u,v} \mid L_r) + \frac{1}{n^2}. \tag{4}$$

The next step is to evaluate $\mathbf{Pr}(A_{u,v} \mid L_r)$. To this aim, for each $k \geqslant 1$ and $w \in N_{T_k}$, we define the event

$$D_{w,k} = \{w \text{ dies at time } T_k\}.$$

To bound $\mathbf{Pr}(D_{w,k} \mid L_r)$, for each $k = r-i, \ldots, r$ and $w \in N_{T_k}$, we use Lemma IV.4 to get $\mathbf{Pr}(D_{w,k}) \leqslant 1/(1.8n)$, and, hence, from Bayes' rule,

$$\mathbf{Pr}(D_{w,k} \mid L_r) = \frac{\mathbf{Pr}(D_{w,k} \cap L_r)}{\mathbf{Pr}(L_r)} \leqslant \frac{\mathbf{Pr}(D_{w,k})}{1 - 1/n^2} \leqslant \frac{0.6}{n} \tag{5}$$

Now, for each $j = r-i, \ldots, r$, we define the following events

$$A^j_{u,v} = \{\text{a specific request of } u \text{ connects to } v \text{ at time } T_j\},$$

and write $A_{u,v} = \cup_{j=r-i}^{r} A^j_{u,v}$. Notice that there is some difference between the probability distribution of $A^{r-i}_{u,v}$ and that of $A^j_{u,v}$ for each $j > r - i$. Indeed, the following holds

$$\mathbf{Pr}\left(A^{r-i}_{u,v} \mid L_r\right) \leqslant \frac{1.2}{n}, \tag{6}$$

since this is the probability that the request of u has destination v at the time of u's arrival (since v is older than u). On the other hand, for each $j = r - i + 1, \ldots, r$, thanks to the memoryless property of the exponential distribution,

$$\mathbf{Pr}\left(A^j_{u,v} \mid L_r\right) \leqslant 1 \cdot \frac{0.6}{n} \cdot \frac{1.2}{n}. \tag{7}$$

The above bound holds since any specific request of u can choose v as destination at round T_j only if, at round T_{j-1}, u was not already connected to v. So, the first factor 1 in the r.h.s. of (7) is an upper bound on the probability that, at time T_{j-1}, u is not connected to v. The second factor, $1/(1.7n)$, is the upper bound on the probability (conditional to L_r from (5)) that the node to which u is connected dies at time T_j. Moreover, $1/(0.9n)$ is the probability, conditional to L_r, that the request of u connects to v at time T_j, if its neighbour is died at time T_j.

So, recalling that $A_{u,v} = \cup_{j=r-i}^{r} A^j_{u,v}$, from (6) and (7),

$$\mathbf{Pr}(A_{u,v} \mid L_r) \leqslant \sum_{j=r-i}^{r} \mathbf{Pr}\left(A^j_{u,v} \mid L_r\right) \leqslant \frac{1.2}{n}\left(1 + \frac{0.6i}{n}\right). \tag{8}$$

Finally, since conditional to L_r we have that $i \leqslant 7n \log n$, using (8) into (4), the proof is completed. □

We now recall the notion of node expansion of a graph.

Definition IV.8 (Node expansion)**.** *The* node isoperimetric number $h_{out}(G)$ *of a graph $G = (N, E)$ is*

$$h_{out}(G) = \min_{0 \leqslant |S| \leqslant |N|/2} \frac{|\partial_{out}(S)|}{|S|},$$

where we used $\partial_{out}(S)$ for the outer boundary of S

$$\partial_{out}(S) = \{v \in N \setminus S : \{u, v\} \in E \text{ for some } u \in S\}.$$

Given a constant $\varepsilon > 0$, a graph G is an (node) ε-expander *if $h_{out}(G) \geqslant \varepsilon$.*

As for the Poisson model with edge regeneration, we can prove the following expansion property.

Theorem IV.9 (Expansion)**.** *For every constant $d \geqslant 35$ and for every sufficiently large n, let $\{G_t = (N_t, E_t) : t \in \mathbb{R}^+\}$ be a PDGR sampled from $\mathcal{G}(\lambda, \mu, d)$ with $\lambda = 1$ and $\mu = 1/n$. Then, for every fixed integer $r \geqslant 7n \log n$, w.h.p. the snapshot G_{T_r} is an ε-expander with parameter $\varepsilon \geqslant 0.1$.*

The proof proceeds analyzing three different size ranges of the node subset $S \subseteq N_{T_r}$, the expansion of which has to be shown. We here only provide the proof for subsets of middle sizes, which represents one of the key technical contributions of this paper.

a) Expansion of middle-size subsets: This case deals with subsets of size in the range $n/\log^2 n \leqslant |S| \leqslant n/14$. The presence of a large number of subsets in this range does not allow use of any rough worst-case counting argument: for instance, assuming that all nodes in the considered subset S have ages $\mathcal{O}(n \log n)$ and applying the corresponding edge-probability bound given by (3) would lead to a useless, overestimate of the probability of non-expansion for some subset S.

In a nutshell, we address this technical issue by classifying and partitioning subsets S according to their *age profiles*. We first define a sequence of $\Theta(\log n)$ *slices* for possible nodes' ages and then we provide an effective age profile of each subset S (and T), depending on how large its intersection is with each of these slices. Thanks to the properties of the exponential distributions of node lifetimes in the Poisson model (see (2) in Lemma IV.4), we show that the existence of a given subset in a given time has a probability that essentially depends on its profile. Informally, the larger the number of old nodes in S, the smaller the probability of S being a subset of N_{T_r}.

Combining age profiling with a more careful use of the parameterized bound on the edge probability of (3), we compute a bound (see (25)) that, in turn, we show to be dominated by the KL divergence of two suitably defined probability distributions. Finally, our target probability bound, stated in the next lemma, is obtained by the standard KL divergence inequality (see [10] pag. 658-659, Theorem 17.1.7). In the end, the arguments outlined above allow us to prove the following result:

Lemma IV.10 (Expansion of middle-size subsets)**.** *Under the hypothesis of Theorem IV.9, for subsets S of N_{T_r}, with probability of at least $1 - 2/n^2$,*

$$\min_{n/\log^2 n \leqslant |S| \leqslant n/14} \frac{|\partial_{out}(S)|}{|S|} \geqslant 0.1.$$

Proof. From Lemma IV.5, all nodes in N_{T_r} were born after time $T_{r-7n\log n}$ with probability at least $1 - 1/n^2$. So, if we define the event

$$L_r = \{\text{each node in } N_{T_r} \text{ was born after time } T_{r-7n\log n}\},$$

we obtain $\mathbf{Pr}(L_r) \geqslant 1 - 1/n^2$. We will condition on this event throughout the rest of this proof. In the remainder, we also use i to denote the node that joined the network at round T_{r-i+1} (i.e. the node whose age is i in terms of discrete rounds). So, conditioning on L_r implies

$$N_{T_r} \subseteq \{1, 2, 3, \ldots, 7n \log n\}.$$

We next show that (conditioning on L_r) any two disjoint sets $S, T \subseteq \{1, 2, \ldots, 7n \log n\}$, such that $n/\log^2 n \leqslant |S| \leqslant n/14$, $|T| = 0.1|S|$, $S, T \subseteq N_{T_r}$, and $\partial_{out}(S) \subseteq T$, may only exist with negligible probability. To this aim, we define the event

$$A_{S,T} = \{\partial_{out}(S) \subseteq T\} \cap \{S, T \subseteq N_{T_r}\}.$$

From the law of total probability,

$$\mathbf{Pr}\left(\min_{n/\log^2 n \leqslant |S| \leqslant n/14} \frac{|\partial_{out}(S)|}{|S|} \leqslant 0.1\right) \leqslant \sum_{\substack{n/\log^2 n \leqslant |S| \leqslant n/2, |T| = 0.1|S| \\ S,T \subseteq \{1,2,\ldots,7n\log n\}}} \mathbf{Pr}(A_{S,T} \mid L_r) + \frac{1}{n^2}. \quad (9)$$

Hence, our next goal is to upper bound the quantity $\mathbf{Pr}(A_{S,T} \mid L_r)$. For each $i \in S$, let B_i denote the event

$$B_i = \{\text{Each of the } d \text{ requestsof node } i \text{ has destination in } S \cup T\}.$$

Then, we can write

$$A_{S,T} = \cap_{i \in S} B_i \cap \{S, T \subseteq N_{T_r}\},$$

and from Bayes' rule

$$\mathbf{Pr}(A_{S,T} \mid L_r) = \mathbf{Pr}(\cap_{i \in S} B_i \mid S, T \subseteq N_{T_r}, L_r)\, \mathbf{Pr}(S, T \subseteq N_{T_r} \mid L_r). \quad (10)$$

From Lemma IV.7, conditioned on the event $\{S, T \subseteq N_{T_r}\}$, we get

$$\mathbf{Pr}(B_i \mid S, T \subseteq N_{T_r}, L_r) = \left[\frac{1.3|S \cup T|}{n}\left(1 + \frac{0.6i}{n}\right)\right]^d. \quad (11)$$

Since we use (10) to bound $\mathbf{Pr}(A_{S,T} \mid L_r)$, we need an upper bound on $\mathbf{Pr}(S, T \subseteq N_{T_r})$. To this aim, we use the bound (2) from Lemma IV.4. However, the notion of *round* given in Definition IV.3 implies that nodes' deaths/departures are not independent. In fact, if we know that, in a given round, node v died, we also know that no other event occurred in that round, including departures of other nodes. By the same token, knowing that one node did not die in a given round can only increase the probability of other nodes to die in that round. To address this issue, we consider the probability that

a fixed set of node survives in any given round. From Lemma IV.4, for an arbitrary set of k nodes we have

$$\begin{aligned}&\mathbf{Pr}\left(v_1,\dots,v_k \in N_{T_r} \mid v_1,\dots,v_k \in N_{T_{r-1}}, L_r\right)\\ &\leqslant 1-\frac{0.4k}{n} \leqslant \left(1-\frac{0.4}{n}\right)^k,\end{aligned} \tag{12}$$

This is the probability that the next step does not see the death of any of the k nodes under consideration. The last inequality in (12) follows from the binomial inequality. So, thanks to (12) and to the memoryless property of the exponential distribution,

$$\mathbf{Pr}\left(S,T \subseteq N_{T_r} \mid L_r\right) \leqslant \prod_{i\in S\cup T} e^{-0.4i/n}, \tag{13}$$

where, in (13) we used the fact that, from (12), each node contributes in the product with a factor $1-0.4/n$ for each round of its life and that, for each $i \geqslant 1$, $(1-0.4/n)^i \leqslant e^{-0.4i/n}$. Since each node chooses the destination of its out-edges independently of the other nodes, we can place (11) and (13) into (10), and obtain

$$\begin{aligned}&\mathbf{Pr}\left(A_{S,T} \mid L_r\right)\\ &\leqslant \prod_{i\in S\cup T} e^{-0.4i/n}\cdot\prod_{i\in S}\min\left\{1,\left[\frac{1.3|S\cup T|}{n}\left(1+\frac{0.6i}{n}\right)\right]^d\right\}.\end{aligned} \tag{14}$$

For each set $R \subseteq N_{T_r}$, we define the sequence $(K_1^R,\dots,K_L^R)$ (where $L = 7\log n$), whose goal is to classify the nodes of the set according to their *age profile*:

$$\begin{aligned}&K_1^R = |R\cap\{1,2,\dots,n\}|,\ K_2^R = |R\cap\{n+1,\dots 2n\}|\\ &\cdots K_L^R = |R\cap\{(L-1)n+1,\dots,Ln\}|.\end{aligned}$$

Notice that, if $|R| = r$ and $K_1^R = r_1,\dots,K_L^R = r_L$, then $\sum_{m=1}^L r_m = r$. For each set $R \subseteq N_{T_r}$, we denote the vector of random variables $(K_1^R,\dots,K_L^R)$ as $\mathbf{K}^R$. According to this definition, by setting $\mathbf{k} = (k_1,\dots,k_L)$ and $\mathbf{h} = (h_1,\dots,h_L)$, we can rewrite (9) as follows:

$$\begin{aligned}&\mathbf{Pr}\left(\min_{n/\log^2 n\leqslant|S|\leqslant n/14}\frac{|\partial_{out}(S)|}{|S|}\leqslant 0.1\right)\\ &\leqslant \sum_{k=n/\log^2 n}^{n/14}\sum_{\substack{k_1+\cdots+k_L=k\\ h_1+\cdots+h_L=0.1k}}\sum_{\substack{S,T:\ \mathbf{K}^S=\mathbf{k}\\ \mathbf{K}^T=\mathbf{h}}}\mathbf{Pr}\left(A_{S,T}\mid L_r\right)+\frac{1}{n^2}.\end{aligned} \tag{15}$$

Here, we have to sum over all the possible size $k = n/\log^2 n,\dots,n/14$ of the set S, all the possible vectors $\mathbf{k}$ and $\mathbf{h}$ whose sum of the elements is equal to k and $0.1k$, respectively (i.e. the characterization of the age profiles of S and T with $|S| = k$ and $|T| = 0.1|S| = 0.1k$), and, finally, over all the possible sets S,T characterized by $\mathbf{K}^S = \mathbf{k}$ and $\mathbf{K}^T = \mathbf{h}$, respectively. From (14), we get

$$\begin{aligned}&\mathbf{Pr}\left(A_{S,T}\text{ s.t. }\mathbf{K}^S=\mathbf{k},\ \mathbf{K}^T=\mathbf{h}\mid L_r\right)\leqslant p(\mathbf{k},\mathbf{h})\\ &=\prod_{m=1,\dots,L}e^{-0.4(m-1)k_m}\prod_{m=1,\dots,L}e^{-0.4(m-1)h_m}\\ &\cdot\prod_{m=1,\dots,L}\min\left\{1,\left[\frac{|S\cup T|}{0.8n}(1+0.6m)\right]^{dk_m}\right\}.\end{aligned} \tag{16}$$

The number of subsets $S,T \subseteq \{1,2,\dots,7n\log n\}$ such that $(K_1^S,\dots,K_L^S) = (k_1,\dots,k_L)$ and $(K_1^T,\dots,K_L^S) = (h_1,\dots,h_L)$ is bounded by

$$n(\mathbf{k},\mathbf{h}) = \binom{n}{k_1}\cdot\binom{n}{h_1}\cdots\binom{n}{k_2}\cdot\binom{n}{k_2}\cdots\binom{n}{k_L}\cdot\binom{n}{h_L}. \tag{17}$$

So, we introduce the quantity $s(\mathbf{k},\mathbf{h})$ and get the following bound from (16) and (17):

$$\begin{aligned}s(\mathbf{k},\mathbf{h}) &= \sum_{\substack{S,T:\mathbf{K}^S=\mathbf{k}\\ \mathbf{K}^T=\mathbf{h}}}\mathbf{Pr}\left(A_{S,T}\text{ s.t. }\mathbf{K}^S=\mathbf{k},\ \mathbf{K}^T=\mathbf{h}\right)\\ &\leqslant n(\mathbf{k},\mathbf{h})\cdot p(\mathbf{k},\mathbf{h}).\end{aligned} \tag{18}$$

We plug (17) and (16) into (18) and, since $|S\cup T| = 1.1k$, if we define

$$s_1(\mathbf{k},\mathbf{h}) = \prod_{m=1}^L\binom{n}{h_m}e^{-0.4(m-1)h_m}\quad\text{and}\quad s_2(\mathbf{k},\mathbf{h}) = \prod_{m=1}^L\binom{n}{k_m}e^{-0.4(m-1)k_m}\cdot\min\left\{1,\left(\frac{1.5k(1+0.6m)}{n}\right)^{dk_m}\right\},$$

we have $s(\mathbf{k},\mathbf{h}) \leqslant s_1(\mathbf{k},\mathbf{h})\cdot s_2(\mathbf{k},\mathbf{h})$. The next step is to prove that $s(\mathbf{k},\mathbf{h}) \leqslant 2^{-0.15k}$. To this aim, we provide separate upper bounds for $\log(s_1(\mathbf{k},\mathbf{h}))$ and $\log(s_2(\mathbf{k},\mathbf{h}))$. In particular, we want to show that

$$\log(s(\mathbf{k},\mathbf{h})) \leqslant -0.15k, \tag{19}$$

which implies $s(\mathbf{k},\mathbf{h}) \leqslant 2^{-0.15k}$. We start by bounding $\log(s_1(\mathbf{k},\mathbf{h}))$. Using $\binom{n}{k} \leqslant \left(\frac{n\cdot e}{k}\right)^k$,

$$\log(s_1(\mathbf{k},\mathbf{h})) \leqslant \sum_{m=1}^L h_m\log\left(\frac{n}{h_m}e^{-0.4m+1.4}\right). \tag{20}$$

Since $\log(x)$ is a concave function, we can apply Jensen's inequality and, recalling that $\sum_{m=1}^L h_m = 0.1k$, we obtain

$$\begin{aligned}&\sum_{m=1}^L h_m\log\left(\frac{n}{h_m}e^{-0.4m+1.4}\right)\Big/\left(\sum_{m=1}^L h_m\right)\\ &\leqslant \log\left(\frac{n\sum_{m=1}^L e^{-0.4m+1.4}}{0.1k}\right).\end{aligned} \tag{21}$$

Since $\sum_{m=1}^L e^{-0.4m+1.4} \leqslant 7$, combining (20), (21) and since $k \leqslant n/14$, we get

$$\log(s_1(\mathbf{k},\mathbf{h})) \leqslant 0.1k\log\left(\frac{7n}{0.1k}\right) \leqslant k\log\left(\frac{n}{7k}\right), \tag{22}$$

where the last inequality follows from a simple calculation. As for $\log(s_2(\mathbf{k},\mathbf{h}))$,

$$\log(s_2(\mathbf{k},\mathbf{h})) \leqslant \sum_{m=1}^{L} k_m \log\left(\frac{n}{7k}\cdot\frac{n\cdot e}{k_m}e^{-0.4(m-1)}\cdot\right.$$
$$\left.\cdot\left(\min\left\{1,\frac{1.5k(0.6m+1)}{n}\right\}\right)^{d}\right) - k\log\left(\frac{n}{7k}\right) . \quad (23)$$

Moreover, since $\log(s(\mathbf{k},\mathbf{h})) = \log(s_1(\mathbf{k},\mathbf{h}))+\log(s_2(\mathbf{k},\mathbf{h}))$, (22) and (23) imply,

$$\log(s(\mathbf{k},\mathbf{h})) \leqslant$$
$$\sum_{m=1}^{L} k_m \log\left(\frac{0.6n^2}{k\cdot k_m}e^{-0.4m}\left(\min\left\{1,\frac{1.5k(0.6m+1)}{n}\right\}\right)^{d}\right).$$

So, from the above inequality,

$$-\frac{\log(s(\mathbf{k},\mathbf{h}))}{k} \geqslant \sum_{m=1}^{L}\frac{k_m}{k}\log\left(\frac{k_m}{k}\frac{9}{10}\cdot\frac{k^2}{0.6n^2}e^{0.4m}\right.$$
$$\left.\left(\min\left\{1,\frac{1.5k(0.6m+1)}{n}\right\}\right)^{-d}\right) + \log\left(\frac{10}{9}\right) . \quad (24)$$

Now, notice that, if we prove that

$$\sum_{m=1}^{L}\frac{k_m}{k}\log\left(\frac{k_m}{k}\frac{9}{10}\cdot\frac{k^2}{0.6n^2}e^{0.4m}\right.$$
$$\left.\left(\min\left\{1,\frac{1.5k(0.6m+1)}{n}\right\}\right)^{-d}\right) \geqslant 0 , \quad (25)$$

then (24) implies (19), since $\log(10/9) \geqslant 0.15$.

Hence, our next step is proving (25). To this purpose, thanks to the *KL divergence inequality* (see [10] pag. 658-659, Theorem 17.1.7), it is sufficient to show that the following are density mass functions over $\{1,2,\ldots,L\}$:

$$p_m = \frac{k_m}{k} \quad \text{and}$$
$$q_m = \frac{10}{9}\cdot\frac{0.6n^2}{k^2}e^{-0.4m}\min\left\{1,\left(\frac{1.5k(0.6m+1)}{n}\right)^{d}\right\} .$$

Notice that $\sum_{m=1}^{L} p_m = 1$, and $\sum_{m=1}^{L} q_m \leqslant 1$ if d is large enough ($d \geqslant 30$) and $k \leqslant \frac{n}{14}$. So, we have proved that q_m and p_m are density mass functions over $\{1,2,\ldots,L\}$ and thus, thanks to KL divergence inequality, (25) holds, implying that $s(\mathbf{k},\mathbf{h}) \leqslant 2^{-0.15k}$.

By plugging (18) into (15) and using $s(\mathbf{k},\mathbf{h}) \leqslant 2^{-0.15k}$,

$$\mathbf{Pr}\left(\min_{n/\log^2\leqslant|S|\leqslant n/14}\frac{|\partial_{out}(S)|}{|S|}\leqslant 0.1\right)$$
$$\leqslant \sum_{k=n/\log^2 n}^{n/14}\sum_{\substack{k_1+\cdots+k_L=k\\ h_1+\cdots+h_L=0.1k}} s(\mathbf{k},\mathbf{h}) + \frac{1}{n^2} \leqslant \frac{2}{n^2} ,$$

where the last inequality holds since the number of integral sequences $k_1,\ldots,k_L$ that sum up k is bounded by $\binom{k+L}{L}$ (and the same holds for h_m), and from simple calculations. □

REFERENCES

[1] J. Augustine, G. Pandurangan, and P. Robinson. Distributed algorithmic foundations of dynamic networks. *ACM SIGACT News*, 47(1):69–98, 2016.

[2] J. Augustine, G. Pandurangan, P. Robinson, S. Roche, and E. Upfal. Enabling robust and efficient distributed computation in dynamic peer-to-peer networks. In *IEEE FOCS'15*, pages 350–369, 2015.

[3] J. Augustine, G. Pandurangan, P. Robinson, and E. Upfal. Towards robust and efficient computation in dynamic peer-to-peer networks. In *ACM-SIAM SODA'12*, page 551–569, 2012.

[4] B. Awerbuch and C. Scheideler. The hyperring: A low-congestion deterministic data structure for distributed environments. In *ACM-SIAM SODA'04*, page 318–327, 2004.

[5] L. Becchetti, A. Clementi, F. Pasquale, L. Trevisan, and I. Ziccardi. Expansion and flooding in dynamic random networks with node churn, 2020.

[6] A. Clementi, C. Macci, A. Monti, F. Pasquale, and R. Silvestri. Flooding time of edge-markovian evolving graphs. *SIAM Journal on Discrete Mathematics*, 24(4):1694–1712, 2010.

[7] A. Clementi, A. Monti, Francesco Pasquale, and R. Silvestri. Information spreading in stationary markovian evolving graphs. *IEEE Trans. Parallel Distrib. Syst.*, 22(9):1425–1432, 2011.

[8] C. Cooper, M. Dyer, and C. Greenhill. Sampling regular graphs and a peer-to-peer network. *Combinatorics, Probability and Computing*, 16(4):557–593, 2007.

[9] Bitcoin Core. Bitcoin Core 0.11 (ch 4): P2P Network.

[10] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, USA, 2006.

[11] S. Delgado-Segura, S. Bakshi, C. Pérez-Solà, J. Litton, A. Pachulski, A. Miller, and B. Bhattacharjee. Txprobe: Discovering bitcoin's network topology using orphan transactions. In *Int. Conf. on Financial Cryptography and Data Security*, pages 550–566, 2019.

[12] D. Dubhashi and A. Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. 1st edition, 2009.

[13] P. Duchon and R. Duvignau. Local update algorithms for random graphs. In A. Pardo and A. Viola, editors, *LATIN'14*, volume 8392 of *LNCS*, pages 367–378, 2014.

[14] R. Jacob, A. Richa, C. Scheideler, S. Schmid, and H. Täubig. A distributed polylogarithmic time algorithm for self-stabilizing skip graphs. In *ACM PODC'09*, page 131–140, 2009.

[15] F. Kuhn, N. Lynch, and R. Oshman. Distributed computation in dynamic networks. In *ACM STOC'10*, page 513–522, 2010.

[16] F. Kuhn and R. Oshman. Dynamic networks: Models and algorithms. *SIGACT News*, 42(1):82–96, March 2011.

[17] C. Law and K-Y Siu. Distributed construction of random expander networks. In *IEEE INFOCOM'03)*, volume 3, pages 2133–2143, 2003.

[18] Othon M. An introduction to temporal graphs: An algorithmic perspective. *Internet Mathematics*, 12(4):239–280, 2016.

[19] P. Mahlmann and C. Schindelhauer. Peer-to-peer networks based on random transformations of connected regular undirected graphs. In *ACM SPAA'15*, page 155–164, 2005.

[20] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. urlhttps://bitcoin.org/bitcoin.pdf, 2008.

[21] T. Neudecker, P. Andelfinger, and H. Hartenstein. Timing analysis for inferring the topology of the bitcoin peer-to-peer network. In *IEEE Conf. on Ubiquitous Intelligence & Computing,....*, pages 358–367, 2016.

[22] G. Pandurangan, P. Raghavan, and E. Upfal. Building low-diameter peer-to-peer networks. *IEEE Journal on selected areas in communications*, 21(6):995–1002, 2003. Prel. vers. in FOCS'01.

[23] G. Pandurangan and A. Trehan. Xheal: A localized self-healing algorithm using expanders. *Distrib. Comput.*, 27(1):39–54, February 2014.

[24] A. Yeow. Global Bitcoin Nodes Distribution. urlhttps://bitnodes.io/.