Rumor Spreading in Random Evolving Graphs*

A. Clementi[1], P. Crescenzi[2], C. Doerr[3], P. Fraigniaud[4], M. Isopi[5], A. Panconesi[5], F. Pasquale[5], and R. Silvestri[5]

[1] Università Tor Vergata di Roma
[2] Università di Firenze
[3] Université Paris Diderot and Max Planck Institute Saarbrücken
[4] CNRS and Université Paris Diderot
[5] Sapienza Università di Roma

Abstract. In this paper, we aim at analyzing the classical information spreading `Push` protocol in *dynamic* networks. We consider the *edge-Markovian* evolving graph model which captures natural temporal dependencies between the structure of the network at time t, and the one at time $t+1$. Precisely, a non-edge appears with probability p, while an existing edge dies with probability q. In order to fit with real-world traces, we mostly concentrate our study on the case where $p = \Omega(\frac{1}{n})$ and q is constant. We prove that, in this realistic scenario, the `Push` protocol does perform well, completing information spreading in $O(\log n)$ time steps, w.h.p., even when the network is, w.h.p., disconnected at every time step (e.g., when $p \ll \frac{\log n}{n}$). The bound is tight. We also address other ranges of parameters p and q (e.g., $p+q=1$ with arbitrary p and q, and $p = \Theta\left(\frac{1}{n}\right)$ with arbitrary q). Although they do not precisely fit with the measures performed on real-world traces, they can be of independent interest for other settings. The results in these cases confirm the positive impact of dynamism.

1 Introduction

Context and Objective. *Rumor spreading* is a well-known gossip-based distributed algorithm for disseminating information in large networks. According to the synchronous `Push` version of this algorithm, an arbitrary source node is initially informed, and, at each time step (a.k.a. round), each informed node u chooses one of its neighbors v uniformly at random, and this node becomes informed at the next time step.

Rumor spreading (originally called *rumor mongering*) was first introduced by [13], in the context of replicated databases, as a solution to the problem of

* A. Clementi is supported by Italian MIUR under the COFIN 2010-11 project *ARS TechnoMedia*. C. Doerr is supported by a F. Lynen postdoctoral research fellowship of the A. von Humboldt Foundation and by ANR project CRYQ. P. Fraigniaud is supported by ANR project DISPLEXITY, and INRIA project GANG. A. Panconesi is supported by a Google Faculty Research Award and EU FET project MULTIPLEX 317532. F. Pasquale is supported by EU FET project MULTIPLEX 317532.

distributing updates and driving replicas towards consistency. Successively, it has been proposed in several other application areas (for a nice survey of gossip-based algorithm applications, see also [31]). Rumor spreading has also been deeply analyzed from a theoretical and mathematical point of view. Indeed, as already observed in [13], rumor spreading is just an example of an epidemic process: hence, its analysis "benefits greatly from the existing mathematical theory of epidemiology".

In particular, the *completion time* of rumor spreading, that is, the number of steps required in order to have all nodes informed with high probability[1] (w.h.p.), has been investigated in the case of several network topologies [6, 14, 17, 20–22, 30, 34], to mention just a few. Further works also derive deep connections between the completion time itself and some classic measures of graph spectral theory [7, 8, 23, 24, 35]. Recently, rumor spreading has been also analysed in the presence of transmission failures of the protocol [19, 15].

It is important to observe that the techniques and the arguments adopted in these studies strongly rely on the fact that the underlying graph is *static* and does not change over time. For instance, most of these analyses exploit the crucial fact that the degree of every node (no matter whether this is a random variable or a deterministic value) never changes during the entire execution of the rumor spreading algorithm. This paper addresses the speed of rumor spreading in the case of *dynamic* networks, where nodes and edges can appear and disappear over time (several emerging networking technologies such as ad hoc wireless, sensor, mobile networks, and peer-to-peer networks are indeed inherently dynamic).

In order to investigate the behavior of distributed protocols in the case of dynamic networks, the concept of evolving graph has been introduced in the literature. An *evolving graph* is a sequence of graphs $(G_t)_{t \geq 0}$ where $t \in \mathbb{N}$ (to indicate that we consider the graph *snapshots* at discrete time steps t, although it may evolve in a continuous manner) with the same set of n nodes.[2] This concept is rater general, ranging from *adversarial* evolving graphs [11, 32] to *random* evolving graphs [4]. In the case of *random* evolving graphs, at each time step, the graph G_t is chosen randomly according to some probability distribution over a specified family of graphs. One very well-known and deeply studied example of such a family is the set $\mathcal{G}_{n,p}$ of *Erdős-Rényi* random graphs [1, 16, 25]. In the evolving graph setting, at every time step t, each possible edge exists with probability p (independently of the previous graphs $G_{t'}$, $t' < t$, and independently of the other edges in G_t).

Random evolving graphs can exhibit communication properties which are much stronger than static networks having the same expected edge density (for a recent survey on computing over dynamic networks, see [33]). This has been proved in the case of the simplest communication protocol that implements the broadcast operation, that is, the `Flooding` protocol. It has been shown [3, 10, 12] that the `Flooding` completion time may be very fast (typically poly-logarithmic in the number of nodes) even when the network topology is, w.h.p., sparse, or

[1] An event holds w.h.p. if it holds with probability $1 - O(1/n^c)$ for some $c > 0$.
[2] As far as we know, this has been formally introduced for the first time in [18].

even highly disconnected at every time step. Therefore, such previous results provide analytical evidences of the fact that random network dynamics not only do not hurt, but can actually help data communication, which is of the utmost importance in several contexts, such as, e.g., delay-tolerant networking [37, 38].

The same observation has been made when the model includes some sort of *temporal* dependency, as it is in the case of the random *edge-Markovian* model. According to this model, at every time step t,

- if an edge does not exist in G_t, then it appears in G_{t+1} with probability p;
- if an edge exists in G_t, then it disappears in G_{t+1} with probability q.

For every initial graph G_0, and $0 < p, q < 1$, an edge-Markovian evolving graph will eventually converge to a (random) graph in $\mathcal{G}_{n,\tilde{p}}$ with stationary edge-probability $\tilde{p} = \frac{p}{p+q}$. However, there is a Markovian dependence between graphs at two consecutive time steps, hence, given G_t, the next graph G_{t+1} is not necessarily a random graph in $\mathcal{G}_{n,\tilde{p}}$. Interestingly enough, the edge-Markovian model has been recently subject to experimental validations, in the context of sparse opportunistic mobile networks [38], and of dynamic peer-to-peer systems [37]. These validations demonstrate a good fitting of the model with some real-world data traces.

The completion time of the `Flooding` protocol has been recently analyzed in the edge-Markovian model, for all possible values of $\tilde{p}$ (see [3, 12]). The `Flooding` protocol however generates high message complexity. Moreover, although its completion time is an interesting analog for dynamic graphs of the diameter for static graphs, it is not reflecting the kinds of gossip protocols mentioned at the beginning of this introduction, used for practical applications. Hence the main objective of this paper is to analyze the more practical `Push` protocol, in edge-Markovian evolving graphs.

Framework. We focus our attention on dynamic network topologies yielded by the edge-Markovian evolving graphs for parameters p (*b*irth) and q (*d*eath) that correspond to a good fitting with real-world data traces, as observed in [37, 38]. These traces describe networks with relatively high dynamics, for which the death probability q is at least one order of magnitude greater than the birth probability p. In order to set parameters p and q fitting with these observations, let us consider the expected number of edges $\bar{m}$, and the expected node-degree $\bar{d}$ at the stationary regime, governed by $\tilde{p} = \frac{p}{p+q}$. We have $\bar{m} = \frac{p}{p+q}\binom{n}{2}$, and $\bar{d} = \frac{2\bar{m}}{n} = (n-1)\frac{p}{p+q}$. Thus, at the stationary regime, the expected number of edges ν that switch their state (from non existing to existing, or vice versa) in one time step satisfies

$$\nu = \bar{m}q + (\binom{n}{2} - \bar{m})p = \frac{n(n-1)}{2}\left(\frac{pq}{p+q} + \left(1 - \frac{p}{p+q}\right)p\right) = n(n-1)\frac{pq}{p+q} = nq\bar{d}.$$

Hence, in order to fit with the high dynamics observed in real-world data traces, we set q constant, so that a constant fraction of the edges disappear at every step, while a fraction p of the non-existing edges appear. We consider an arbitrary range for p, with the unique assumption that $p \geq \frac{1}{n}$. (For smaller p's, the completion time of any communication protocol is subject to the expected time $\frac{1}{np} \gg 1$ required for a node to acquire just one link connected to another node). To sum up, we essentially focus on the following range of parameters:

$$\frac{1}{n} \leqslant p < 1 \quad \text{and} \quad q = \Omega(1). \tag{1}$$

This range includes network topologies for a wide interval of expected edge density (from very sparse and disconnected graphs, to almost-complete ones), and with an expected number of switching edges per time step equal to some constant fraction of the expected total number of edges. Other ranges are also analyzed in the paper (e.g., $p + q = 1$ with arbitrary p and q, and $p = \Theta\left(\frac{1}{n}\right)$ with arbitrary q), but the range in Eq. (1) appears to be the most realistic one, according to the current measurements on dynamic networks.

Our results. For the parameter range in Eq. (1), we show that, w.h.p., starting from any n-node graph G_0, the `Push` protocol informs all n nodes in $\Theta(\log n)$ time steps. Hence, in particular, even if the graph G_t is w.h.p. disconnected at every time step (this is the case for $p \ll \frac{\log n}{n}$), the completion time of the `Push` protocol is as small as it could be (the `Push` protocol cannot perform faster than $\Omega(\log n)$ steps in any static or dynamic graph since the number of informed nodes can at most double at every step).

We also address other ranges of parameters p and q. One such case is the sequence of independent $\mathcal{G}_{n,p}$ graphs, that is, the case where $p + q = 1$. Actually, the analysis of this special case will allow us to focus on the first important probabilistic issue that needs to be solved: spatial dependencies. Indeed, even in this case, the `Push` protocol induces a positive correlation among some crucial events that determine the number of new informed nodes at the next time step. This holds despite the fact that every edge is set independently from the others. For a sequence of independent $\mathcal{G}_{n,p}$ graphs, we prove that for every p (i.e., also for $p = o(\frac{1}{n})$) and $q = 1 - p$ the completion time of the `Push` protocol is, w.h.p., $\mathcal{O}(\log n/(\hat{p}n))$, where $\hat{p} = \min\{p, 1/n\}$. By comparing the lower bound for `Flooding` in [12], it turns out that this bound is tight, even for very sparse graphs.

Finally, we show that the logarithmic bound for the `Push` protocol holds for more "static" network topologies as well, e.g., for the range $p = \frac{c}{n}$ where $c > 0$ is a constant, and q is arbitrary. This parameter range includes edge-Markovian graphs with a small expected number of switching edges (this happens when $q = o(1)$). In this case, too, `Push` completes, w.h.p., in $O(\log n)$ rounds. This gives yet another evidence that dynamism helps.

Due to lack of space several proofs are omitted. We refer the interested reader to the full version of the paper [9].

2 Preliminaries

The number of vertices in the graph will always be denoted by n. We abbreviate $[n] := \{1, \ldots, n\}$ and $\binom{[n]}{2} := \{\{i, j\} \mid i, j \in [n]\}$. For any subset $E \subseteq \binom{[n]}{2}$

and any two subsets $A, B \subseteq [n]$, define $E(A) = \{\text{edges of } E \text{ incident to } A\}$ and $E(A,B) = \{\{u,v\} \in E \mid u \in A, v \in B\}$. We consider the edge-Markovian evolving graph model $\mathcal{G}(n,p,q;E_0)$ where E_0 is the starting set of edges.

The Push Protocol over $\mathcal{G}(n,p,q;E_0)$ can be represented as a random process over the set $\mathcal{S}$ of all possible pairs (E,I) where E is a subset of edges and I is a subset of nodes. In particular, the combined Markov process works as follows

$$\ldots \to \ (E_t, I_t) \ \stackrel{\text{edge-Markovian}}{\longrightarrow} \ (E_{t+1}, I_t) \stackrel{\texttt{Push} \text{ protocol}}{\longrightarrow} (E_{t+1}, I_{t+1}) \ \stackrel{\text{edge-Markovian}}{\longrightarrow} \ \ldots$$

where E_t and I_t represent the set of existing edges and the set of informed nodes at time t, respectively. All events, probabilities and random variables are defined over the above random process. Given a graph $G = ([n], E)$, a node $v \in [n]$, and a subset of nodes $A \subseteq [n]$ we define $\deg_G(v, A) = |\{(v,a) \in E \mid a \in A\}|$. When we have a sequence of graphs $\{G_t = ([n], E_t) : t \in \mathbb{N}\}$ we write $\deg_t(v, A)$ instead of $\deg_{G_t}(v, A)$.

Given a graph G and an informed node $u \in I$, we define $\delta_G(u)$ as the random variable indicating the node selected by u in graph G according to the Push protocol. When G and/or t are clear from the context, they will be omitted.

Remark. It is worth noticing that analyzing the Push protocol in edge-Markovian graphs is not only subject to temporal dependencies, but also to *spatial* dependencies. To see why, consider a time step of the Push protocol. For an informed node u and a non-informed one v it is not hard to calculate the probability that $\delta(u) = v$ by conditioning on the degree of u. However, if u_1, u_2 are two informed nodes and v_1, v_2 are two non-informed ones, events "$\delta(u_1) = v_1$" and "$\delta(u_2) = v_2$" are not independent. Indeed, since the underlying graph is random, event "$\delta(u_1) = v_1$" *decreases* the probability of existence of an edge between u_1 and u_2, and so it affects the value of the random variable $\delta(u_2)$.

3 Warm up: the time-independent case

In this section we analyze the special case of a sequence of independent $G_{n,p}$ (observe that a sequence of independent $G_{n,p}$ is edge-Markovian with $q = 1-p$). We show that the completion time of the Push protocol is $\mathcal{O}(\log n/(\hat{p}n))$ w.h.p., where $\hat{p} = \min\{p, 1/n\}$. In Theorem 1 we prove the result for $p \geqslant 1/n$ and in Theorem 2 for $p \leqslant 1/n$. From the lower bound on the flooding time for edge-Markovian graphs [12], it turns out that our bound is optimal.

As mentioned in Section 2, even though in this case there is no time-dependency in the sequence of graphs, the Push protocol introduces a kind of dependence that has to be carefully handled. The key challenge is to evaluate the probability that v receives the information from at least one of the informed nodes; i.e., $1 - \mathbf{P}\left(\cap_{u \in I}\{\delta(u) \neq v\}\right)$. We consider the Push operation on a *modified* random graph where we prove that the above events become independent and the number of new informed nodes in the original random graph is at least as large as in the modified version.

Definition 1. *Let $G = ([n], E)$ be a graph, let $I \subseteq [n]$ be a set of nodes, and let $b \in [n]$ be a positive integer. The (I, b)-*modified graph *G is the graph $H = ([n] \cup \{v_1, \ldots, v_b\})$, where $\{v_1, \ldots, v_b\}$ is a set of extra* virtual *nodes, obtained from G by the following operations: 1. For every node $u \in I$ with $\deg_G(u) > b$, remove all edges incident to u; 2. For every node $u \in I$ with $\deg_G(u) \leqslant b$, add all edges $\{u, v_1\}, \ldots, \{u, v_b\}$ between u and the virtual nodes; 3. Remove all edges between any pair of nodes that are both in I.*

Let I be the set of informed nodes performing a `Push` operation on a $G_{n,p}$ random graph. As previously observed, if $v \in [n] \setminus I$ is a non-informed node, then the events $\{\{\delta_G(u) = v\} \,:\, u \in I\}$ are not independent, but the events $\{\{\delta_H(u) = v\} \,:\, u \in I\}$ on the (I, b)-modified graph H are independent because of Operation 3 in Definition 1. In the next lemma we prove that, if the informed nodes perform a `Push` operation both in a graph and in its modified version, then the number of new informed nodes in the original graph is (stochastically) larger than the number of informed nodes in the modified one. We will then apply this result to $G_{n,p}$ random graphs.

Lemma 1. *Let $G([n], E)$ be a graph and let b an integer such that $1 \leqslant b \leqslant n$. Let $I \subseteq [n]$ be a set of nodes performing a `Push` operation in graphs G and H, where H is the (I, b)-modified G according to Definition 1. Let X and Y be the random variables counting the numbers of new informed nodes in G and H respectively. Then for every $h \in [0, n]$ it holds that $\mathbf{P}(X \leqslant h) \leqslant \mathbf{P}(Y \leqslant h)$.*

Proof. Consider the following coupling: Let $u \in I$ be an informed node such that $\deg_G(u) \leqslant b$ and let h and k be the number of informed and non-informed neighbors of u respectively. Choose $\delta_H(u)$ u.a.r. among the neighbors of u in H. As for $\delta_G(u)$, we do the following: If $\delta_H(u) \in [n] \setminus I$ then choose $\delta_G(u) = \delta_H(u)$; otherwise (i.e., when $\delta_H(u)$ is a virtual node) with probability $1 - x$ choose $\delta_G(u)$ u.a.r. among the informed neighbors of u in G, and with probability x choose $\delta_G(u)$ u.a.r. among the non-informed ones, where $x = \frac{k(b-h)}{(h+k)b}$. Every informed node u with $\deg_G(u) > b$ instead performs a `Push` operation in G independently. By construction we have that the set of new (non-virtual) informed nodes in H is a subset of the set of new informed nodes in G. Moreover, it is easy to check that, for every informed node u in I, $\delta_G(u)$ is u.a.r. among neighbors of u. $\square$

In the next lemma we give a lower bound on the probability that a non-informed node gets informed in the modified $G_{n,p}$.

Lemma 2. *Let $I \subseteq [n]$ be the set of informed nodes performing the `Push` operation in a $G_{n,p}$ random graph and let X be the random variable counting the number of non-informed nodes that get informed after the `Push` operation. It holds that $\mathbf{P}(X \geqslant \lambda \cdot \min\{|I|,\, n - |I|\}) \geqslant \lambda$, where λ is a positive constant.*

Proof. Let I be the set of currently informed nodes, let $G = ([n], E)$ be the random graph at the next time step and let H be its $(I, 3np)$-modified version. Now we show that the number of nodes that gets informed in H is at least $\lambda \cdot \min\{|I|, n - |I|\}$ with probability at least λ, for a suitable constant λ.

Let $u \in I$ be an informed node and let $v \in [n] \setminus I$ be a non-informed one. Observe that by the definition of H, u cannot choose v in H if the edge $\{u, v\} \notin E$ or if the degree of u in G is larger than $3np$ (see Operation 3 in Definition 1). Thus the probability $\mathbf{P}(\delta_H(u) = v)$ that node u chooses node v in random graph H according to the `Push` protocol is equal to

$$\mathbf{P}(\delta_H(u) = v \mid \{u, v\} \in E \wedge \deg_G(u) \leqslant 3np) \cdot \mathbf{P}(\{u, v\} \in G \wedge \deg_G(u) \leqslant 3np) . \tag{2}$$

If $\deg_G(u) \leqslant 3np$ then node u in H has exactly $3np$ virtual neighbors plus at most other $3np$ non-informed neighbors. It follows that

$$\mathbf{P}(\delta_H(u) = v \mid \{u, v\} \in E \wedge \deg_G(u) \leqslant 3np) \geqslant 1/(6np). \tag{3}$$

We also have that

$$\begin{aligned} \mathbf{P}(\{u, v\} \in E, \deg_G(u) \leqslant 3np) &= \mathbf{P}(\{u, v\} \in E)\, \mathbf{P}(\deg_G(u) \leqslant 3np \mid \{u, v\} \in E) \\ &= p \cdot \mathbf{P}(\deg_G(u) \leqslant 3np \mid \{u, v\} \in E) . \end{aligned}$$

Since $\mathbf{E}[\deg_G(u) \mid \{u, v\} \in E] \leqslant np + 1$ with $np \geqslant 1$, from the Chernoff bound we can choose a positive constant c and then a positive constant $\beta < 1$ such that

$$\begin{aligned} \mathbf{P}(\deg_G(u) > 3np \mid \{u, v\} \in E) &\leqslant \mathbf{P}(\deg_G(u) > 2np + 1 \mid \{u, v\} \in E) \\ &\leqslant e^{-cnp} = \beta < 1. \end{aligned} \tag{4}$$

By replacing Eq.s 3 and 4 into Eq. 2 we get $\mathbf{P}(\delta_H(u) = v) \geqslant \frac{\alpha}{n}$, for some constant $\alpha > 0$. Since the events $\{\{\delta_H(u) = v\}, u \in I\}$ are independent, the probability that node v is not informed in H is thus $\mathbf{P}(\cap_{u \in I} \delta_H(u) \neq v) \leqslant (1 - \alpha/n)^{|I|} \leqslant e^{-\alpha|I|/n}$. Let Y be the random variable counting the number of new informed nodes in H. The expectation of Y is $\mathbf{E}[Y] \geqslant (n - |I|)\left(1 - e^{-\alpha|I|/n}\right) \geqslant (\alpha/2)(n - |I|)|I|/n$. Hence we get

$$\mathbf{E}[Y] \geqslant \begin{cases} (\alpha/4)|I| & \text{if } |I| \leqslant n/2 , \\ (\alpha/4)(n - |I|) & \text{if } |I| \geqslant n/2 . \end{cases}$$

Since $Y \leqslant \min\{|I|, n - |I|\}$, it follows that $\mathbf{P}(Y \geqslant (\alpha/8) \cdot \min\{|I|, n - |I|\}) \geqslant \alpha/8$. Finally we get the thesis by applying Lemma 1. □

We can now derive the upper bound on the completion time of the `Push` protocol on $G_{n,p}$ random graphs.

Theorem 1. *Let $\mathcal{G} = \{G_t : t \in \mathbb{N}\}$ be a sequence of independent $G_{n,p}$ with $p \geqslant 1/n$. The completion time of the `Push` protocol over $\mathcal{G}$ is $\mathcal{O}(\log n)$ w.h.p.*

Proof. Consider a generic time step t of the execution of the `Push` protocol where $I_t \subseteq [n]$ is the set of informed nodes and $m_t = |I_t|$ is its size. For any t such that $m_t \leqslant n/2$, Lemma 2 implies that $\mathbf{P}(m_{t+1} \geqslant (1 + \lambda)m_t) \geqslant \lambda$, where λ is a positive constant. Let us define event $\mathcal{E}_t = \{m_t \geqslant (1 + \lambda)m_{t-1}\} \vee \{m_{t-1} \geqslant n/2\}$

and let $Y_t = Y_t((E_1, I_1), \ldots, (E_t, I_t))$ be the indicator random variable of that event. Observe that if $t = \frac{\log n}{\log(1+\lambda)}$ then $(1+\lambda)^t \geqslant n/2$. Hence, if we set $T_1 = \frac{2}{\lambda}\frac{\log n}{\log(1+\lambda)}$, we get $\mathbf{P}(m_{T_1} \leqslant n/2) \leqslant \mathbf{P}\left(\sum_{t=1}^{T_1} Y_t \leqslant (\lambda/2)T_1\right)$. This probability is at most as large as the probability that in a sequence of T_1 independent coin tosses, each one giving `head` with probability λ, we see less than $(\lambda/2)T_1$ `heads` (see e.g. Lemma 3.1 in [2]). A direct application of the Chernoff bound shows that this probability is smaller than $e^{-(1/4)\lambda T_1} \leqslant n^{-c}$, for a suitable constant $c > 0$. We can thus state that, after $\mathcal{O}(\log n)$ time steps, there at least $n/2$ informed nodes w.h.p. If $m_{T_1} \geqslant n/2$, then, for every $t \geqslant T_1$, Lemma 2 implies that $\mathbf{P}(n - m_{t+1} \leqslant (1-\lambda)(n - m_t)) \geqslant \lambda$. Observe that if $t = \frac{\log n}{\lambda}$ then $(1-\lambda)^t \leqslant 1/n$, so that for $T_2 = \frac{2}{\lambda} \cdot \frac{\log n}{\lambda} + T_1$ the probability that the `Push` protocol has not completed at time T_2 is $\mathbf{P}(m_{T_2} < n) \leqslant \mathbf{P}\left(m_{T_2} < n \,|\, m_{T_1} \geqslant \frac{n}{2}\right) + \mathbf{P}\left(m_{T_1} < \frac{n}{2}\right)$. As we argued in the analysis of the spreading till $n/2$, the probability $\mathbf{P}\left(m_{T_2} < n \,|\, m_{T_1} \geqslant \frac{n}{2}\right)$ is not larger than the probability that in a sequence of $\frac{2}{\lambda} \cdot \frac{\log n}{\lambda}$ independent coin tosses, each one giving `head` with probability λ, there are less than $\frac{\log n}{\lambda}$ `heads`. Again, by applying the Chernoff bound, the latter is not larger than n^{-c} for a suitable positive constant c. □

In order to prove the bound for $p \leqslant 1/n$, we first show that one single `Push` operation over the union of a sequence of graphs informs (stochastically) less nodes than the sequence of `Push` operations performed in every single graph (this fact will also be used later in Section 4 to analyse the edge-MEG).

Lemma 3. *Let $\{G_t = ([n], E_t) : t = 1, \ldots, T\}$ be a finite sequence of graphs with the same set of nodes $[n]$. Let $I \subseteq [n]$ be the set of informed nodes in the initial graph G_1. Suppose that at every time step every informed node performs a `Push` operation, and let X be the random variable counting the number of informed nodes at time step T. Let $H = ([n], F)$ be such that $F = \cup_{t=1}^{T} E_t$ and let Y be the random variable counting the number of informed nodes when the nodes in I perform one single `Push` operation in graph H. Then for every $\ell = 0, 1, \ldots, n$ it holds that $\mathbf{P}(X \leqslant \ell) \leqslant \mathbf{P}(Y \leqslant \ell)$.*

Observe that if we look at a sequence of independent $G_{n,p}$ with $p \leqslant 1/n$ for a time-window of approximately $1/(np)$ time steps, then every edge appears at least once in the sequence with probability at least $1/n$. The above lemma thus allows us to reduce the case $p \leqslant 1/n$ to the case $p \geqslant 1/n$.

Theorem 2. *Let $\mathcal{G} = \{G_t : t \in \mathbb{N}\}$ be a sequence of independent $G_{n,p}$ with $p \leqslant 1/n$ and let $s \in [n]$. The `Push` protocol with source s over $\mathcal{G}$ completes the broadcast in $\mathcal{O}(\log n/(np))$ time steps w.h.p.*

4 Edge-Markovian graphs with high dynamics

In this section we prove that the `Push` protocol over an edge-Markovian graph $\mathcal{G}(n, p, q; E_0)$ with $p \geqslant 1/n$ and $q = \Omega(1)$ has completion time $\mathcal{O}(\log n)$ w.h.p.

As observed in the Introduction, the stationary random graph is an Erdős-Rényi $G_{n,\tilde{p}}$ where $\tilde{p} = \frac{p}{p+q}$ and the mixing time of the edge Markov chain is $\Theta(\frac{1}{p+q})$. Thus, if p and q fall into the range defined in Eq. 1, we get that the stationary random graph can be sparse and disconnected (when $p = o(\frac{\log n}{n})$) and that the mixing time of the edge Markov chain is $O(1)$. Thus, we can omit the term E_0 and assume it is random according to the stationary distribution.

The time-dependency between consecutive snapshots of the dynamic graph does not allow us to obtain directly the *increasing rate* of the number of informed nodes that we got for the independent-$G_{n,p}$ model. In order to get a result like Lemma 2 for the edge-Markovian case, we need in fact a *bounded-degree* condition on the current set of informed nodes (see Definition 2) that does not apply when the number of informed nodes is *small* (i.e., smaller than $\log n$). However, in order to reach a state where at least $\log n$ nodes are informed, we can use a different ad-hoc technique that analyzes the spreading rate yielded by the source only.

Lemma 4. *Let $\mathcal{G} = \mathcal{G}(n, p, q)$ be an edge-Markovian graph with $p \geqslant 1/n$ and $q = \Omega(1)$, and consider the* `Push` *protocol in $\mathcal{G}$ starting with one informed node. For any positive constant γ, after $\mathcal{O}(\log n)$ time steps there are at least $\gamma \log n$ informed nodes w.h.p.*

We can now start the second part of our analysis where the `Push` operation of all informed nodes (forming the subset I) will be considered and, thanks to the bootstrap, we can assume that $|I| = \Omega(\log n)$. As mentioned at the beginning of the section, we need to introduce the concept of *bounded-degree state* (E, I) of the Markovian process describing the information-spreading process over the dynamic graph, where E is the set of edges and I is the set of informed nodes.

Definition 2. *A state (E, I) such that $|E(I)| \leqslant (8/q)n\tilde{p}|I|$ (with $\tilde{p} = \frac{p}{p+q}$ the stationary edge probability) will be called a* bounded-degree *state.*

In the next lemma we show that, if I is the set of informed nodes with $|I| \geqslant \log n$, if in the starting random graph G_0 every edge exists with probability approximately $(1 \pm \varepsilon)p$, and if it evolves according to the edge-Markovian model and the informed nodes perform the `Push` protocol, then for a long sequence of time steps the random process is in a bounded-degree state. We will use this property in Theorem 3 by observing that, for every initial state, after $\mathcal{O}(\log n)$ time steps an edge-Markovian graph with $p \geqslant 1/n$ and $q \in \Omega(1)$ is in a state where every edge $\{u, v\}$ exists with probability $p_{\{u,v\}} \in [(1-\varepsilon)\tilde{p},\, (1+\varepsilon)\tilde{p}]$.

Lemma 5. *Let $\mathcal{G} = \mathcal{G}(n, p, q, E_0)$ be an edge-Markovian graph starting with G_0 and consider the* `Push` *protocol in $\mathcal{G}$ where I_0 is the set of informed nodes at time $t = 0$. Then, for any constant $c > 0$, for a sequence of $c \log n$ time steps every state is a bounded-degree one w.h.p.*

Now we can bound the *increasing rate* of the number of informed nodes in an edge-Markovian graph. The proof of the following lemma combines the analysis adopted in the proof of Lemma 2 with some further ingredients required to manage the time-dependency of the edge-Markovian model.

Lemma 6. *Let (E, I) be a bounded-degree state and let X be the random variable counting the number of non-informed nodes that get informed after two steps of the* `Push` *operation in the edge-Markovian graph model. It holds that* $\mathbf{P}\left(X \geqslant \varepsilon \cdot \min\{|I|,\, n-|I|\}\right) \geqslant \lambda$*, where ε and λ are positive constants.*

Now we can prove that in $\mathcal{O}(\log n)$ time steps the `Push` protocol informs all nodes in an edge-Markovian graph, w.h.p.

Theorem 3. *Let $\mathcal{G} = \mathcal{G}(n, p, q, E_0)$ be an edge-Markovian graph with $p \geqslant 1/n$ and $q = \Omega(1)$ and let $s \in [n]$ be a node. The* `Push` *protocol with source s completes the broadcast over $\mathcal{G}$ in $\mathcal{O}(\log n)$ time steps w.h.p.*

Proof. Lemma 4 implies that after $\mathcal{O}(\log n)$ time steps there are $\Omega(\log n)$ informed nodes w.h.p. From Lemma 5, it follows that, after further $\mathcal{O}(\log n)$ time steps, the edge-Markovian graph reaches a bounded-degree state and remains so for further $\Omega(\log n)$ time steps. Let us rename $t = 0$ the time step where there are $\Omega(\log n)$ informed nodes and every edge $e \in \binom{[n]}{2}$ exists with probability $p_e \in [(1-\varepsilon)\tilde{p}\,,\ (1+\varepsilon)\tilde{p}]$. We again abbreviate $m_t := |I_t|$. Observe that if recurrence $m_{2(t+1)} \geqslant (1+\varepsilon)m_{2t}$ holds $\log n/\log(1+\varepsilon)$ times, then there are $n/2$ informed nodes. Let us thus name $T = \frac{2}{\lambda}\frac{\log n}{\log(1+\varepsilon)}$. If at time $2T$ there are less than $n/2$ informed nodes, then recurrence $m_{2(t+1)} \geqslant (1+\varepsilon)m_{2t}$ held less than $\lambda T/2$ times. Since, at each time step, the recurrence holds with probability at least λ (there are less than $n/2$ informed nodes and the state is a bounded-degree one w.h.p.), the above probability is at most as large as the probability that in a sequence of T independent coin tosses, each one giving `head` with probability λ, we see less than $(\lambda/2)T$ `heads` (see, e.g., Lemma 3.1 in [2]). By the Chernoff bound such a probability is smaller than $e^{-\gamma\lambda T}$, for a suitable positive constant γ. Since γ and λ are constants and $T = \Theta(\log n)$ we have that

$$\mathbf{P}\left(m_{2T} \leqslant n/2\right) \leqslant n^{-\delta} \tag{5}$$

for a suitable positive constant δ. When m_t is larger than $n/2$ and the edge-Markovian graph is in a bounded-degree state, from Lemma 6 it follows that recurrence $n - m_{t+1} \leqslant (1-\varepsilon)(n - m_t)$ holds with probability at least λ. If this recurrence holds $\log n/\log\left(1/(1-\varepsilon)\right)$ times then the number of informed nodes cannot be smaller than n. Hence, if we name $\tilde{T} := (2/\lambda)\log n/\log\left(1/(1-\varepsilon)\right)$, with the same argument we used to get Eq. 5, we obtain that after $2T + 2\tilde{T}$ time steps all nodes are informed w.h.p. □

5 Edge-Markovian graphs with slow dynamics

We have also considered "more static" sparse dynamic graphs. In particular, we can provide a logarithmic bound on the completion time of the `Push` protocol over the $\mathcal{G}(n, p, q)$ model even for $p = \Theta(1/n)$ and for $q = o(1)$. The proof of the following result combines some new coupling arguments with a previous analysis of the `Push` protocol for static random graphs given in [17].

Theorem 4. *Let $p = \frac{d}{n}$ for some absolute constant $d \in \mathbb{N}$ and let $q = q(n)$ be such that $q(n) = o(1)$. The* `Push` *protocol over edge-Markovian graphs in $\mathcal{G}(n, p, q)$ completes in $O(\log n)$ time, w.h.p.*

6 Conclusion

Completing the whole figure, i.e., for every $(p, q) \in [0, 1]^2$, is of intellectual interest. Our results obtained for the most realistic cases are however already sufficient to measure the positive impact of a certain form of network dynamics on information spreading. To go one step further, we think that the most challenging question is to analyze rumor spreading over more general classes of evolving graphs where edges may not be independent. For instance, it would be interesting to analyze the `Push` protocol over geometric models of mobile networks [12, 28].

References

1. C. Avin, M. Koucky, and Z. Lotker. How to explore a fast-changing world. In *Proc. of 35th ICALP*, volume 5125 of *LNCS*, pages 121–132. Springer-Verlag, 2008.
2. Y. Azar, A.Z. Broder, A.R. Karlin, and E. Upfal. Balanced allocations. *SIAM J. on Computing*, 29(1):180–200, 1999.
3. H. Baumann, P. Crescenzi, and P. Fraigniaud. Parsimonious flooding in dynamic graphs. *Distributed Computing*, 24(1):31–44, 2011.
4. B. Bollobás. *Random Graphs.* Cambridge University Press, 2001.
5. S. Boyd, G. Arpita, P. Balaji, and S. Devavrat. Gossip algorithms: Design, analysis and applications. In *Proc. of 24th INFOCOM*, pages 1653–1664. IEEE, 2005.
6. F. Chierichetti, S. Lattanzi, and A. Panconesi. Rumor Spreading in Social Networks. In *Proc. of 36th ICALP*, volume 5556 of *LNCS*, pages 375–386, 2009.
7. F. Chierichetti, S. Lattanzi, and A. Panconesi. Almost tight bounds on rumour spreading by conductance. In *Proc. of 42nd ACM STOC*, pages 399–408, 2010.
8. F. Chierichetti, S. Lattanzi, and A. Panconesi. Rumour spreading and graph conductance. In *Proc. of 21th ACM-SIAM SODA*, pages 1657–1663, 2010.
9. A. Clementi, P. Crescenzi, C. Doerr, P. Fraigniaud, M. Isopi, F. Pasquale, A. Panconesi, and R. Silvestri Rumor Spreading in Random Evolving Graphs http://arxiv.org/abs/1302.3828
10. A. Clementi, C. Macci, A. Monti, F. Pasquale, and R. Silvestri. Flooding time of edge-Markovian evolving graphs. *SIAM J. Discrete Math.*, 24(4):1694–1712, 2010.
11. A. Clementi, A. Monti, F. Pasquale, and R. Silvestri. Broadcasting in dynamic radio networks. *J. Comput. Syst. Sci.*, 75(4):213–230, 2009.
12. A. Clementi, A. Monti, F. Pasquale, and R. Silvestri. Information spreading in stationary Markovian evolving graph. *IEEE Trans. Parallel Distrib. Syst.*, 22(9):1425–1432, 2011.
13. A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry. Epidemic algorithms for replicated database maintenance. In *Proc. of 6th ACM PODC*, pages 1–12, 1987.
14. B. Doerr, M. Fouz, and T. Friedrich. Social networks spread rumors in sublogarithmic time. In *Proc. of 43rd ACM STOC*, pages 21–30. ACM New York, 2011.

15. B. Doerr, A. Huber, and A. Levavi Strong robustness of randomized rumor spreading protocols *Discrete Applied Mathematics*, 161(6):778–793, 2013.
16. P. Erdős and A. Rényi. On Random Graphs. *Publ. Math.*, 6:290–297, 1959.
17. U. Feige, D. Peleg, P. Raghavan, and E. Upfal. Randomized broadcast in networks. *Random Structures and Algorithms*, 1(4):447–460, 1990.
18. A. Ferreira. On models and algorithms for dynamic communication networks: The case for evolving graphs. In *Proc. of 4th ALGOTEL*, pages 155–161, 2002.
19. N. Fountoulakis, A. Huber, and K. Panagiotou. Reliable broadcasting in random networks and the effect of density. In Proc. of *29th IEEE INFOCOM*, pp2552–2560, 2010.
20. N. Fountoulakis and K. Panagiotou. Rumor spreading on random regular graphs and expanders. In *Proc. of 13th APPROX/RANDOM*, *LNCS* 6302, pp560–573, 2010.
21. N. Fountoulakis, K. Panagiotou, and T. Sauerwald. Ultra-fast rumor spreading in social networks. In *Proc. of 23rd ACM-SIAM SODA*, pp1642–1660, 2012.
22. A. Frieze and G. Grimmett. The shortest-path problem for graphs with random arc-lengths. *Discrete Applied Mathematics*, 10(1):57–77, 1985.
23. G. Giakkoupis. Tight bounds for rumor spreading in graphs of a given conductance. In *Proc. of 28th STACS*, volume 9 of *LIPIcs*, pages 57–68. Schloss Dagstuhl, 2011.
24. G. Giakkoupis and T. Sauerwald. Rumor spreading and vertex expansion. In *Proc. of 23rd ACM-SIAM SODA*, pages 1623–1641. SIAM, 2012.
25. E.N. Gilbert. Random graphs. *Annals of Math. Statistics*, 30(4):1141–1144, 1959.
26. P. Grindrod and D.J. Higham. Evolving graphs: dynamical models, inverse problems and propagation. In *Proc. R. Soc. A, 466(2115)*, pages 753–770, 2010.
27. M. Harchol-Balter, T. Leighton, and D. Lewin. Resource discovery in distributed networks. In *Proc. of 18th PODC*, pages 229–237. ACM New York, 1999.
28. P. Jacquet, B. Mans, and G. Rodolakis. Information Propagation Speed in Mobile and Delay Tolerant Networks. *IEEE Trans. on Inf. Theory*, 56:5001–5015, 2010.
29. M. Jelasity, S. Voulgaris, R. Guerraoui, A.-M. Kermarrec, and M. van Steen. Gossip-based Peer Sampling. *ACM Trans. Comp. Syst.*, 25(3):Article 8, 2007.
30. R. Karp, C. Schindelhauer, S. Shenker, and B. Vocking. Randomized rumor spreading. In *Proc. of 41st IEEE FOCS*, pages 565–574. IEEE, 2000.
31. A.-M. Kermarrec and M. van Steen. Gossiping in distributed systems. *SIGOPS Oper. Syst. Rev.*, 41(5):2–7, 2007.
32. F. Kuhn, N. Linch, and R. Oshman. Distributed Computation in Dynamic Networks. In *Proc. of 42nd ACM STOC*, pages 513–522. ACM New York, 2010.
33. F. Kuhn and R. Oshman. Dynamic networks: models and algorithms. *ACM SIGACT News*, 42(1):82–96, 2011.
34. B. Pittel. On spreading a rumor. *SIAM Journal on Applied Mathematics*, 47(1):213–223, 1987.
35. T. Sauerwald and A. Stauffer. Rumor spreading and vertex expansion on regular graphs. In *Proc. of 22nd ACM-SIAM SODA*, pages 462–475. SIAM, 2011.
36. R. Van Renesse, Y. Minsky, and M. Hayden. A Gossip-Style Failure Detection Service. In *Proc. of Middleware*, pages 55–70, 1998.
37. M. Vojnovic and A. Proutier. Hop limited flooding over dynamic networks. In *Proc. of 30th IEEE INFOCOM*, pages 685–693. IEEE, 2011.
38. J. Whitbeck, V. Conan, and M.D. de Amorim. Performance of Opportunistic Epidemic Routing on Edge-Markovian Dynamic Graphs. *IEEE Transactions on Communications*, 59(5):1259–1263, 2011.