Randomized Rumor Spreading in Social Networks & Complete Graphs

DISSERTATION

zur Erlangung des Grades
des Doktors der Ingenieurswissenschaften (Dr.-Ing.)
der Naturwissenschaftlich-Technischen Fakultäten
der Universität des Saarlandes

Mahmoud Fouz

Juli 2012

Tag des Kolloqiums: 16.07.2012

Dekan der Naturwissenschaftlich-Technischen Fakultät I:
Prof. Dr. Mark Groves

Vorsitzender des Prüfungsausschusses:
Prof. Dr. Kurt Mehlhorn

Gutachter:
Prof. Dr. Benjamin Doerr
Prof. Dr. Markus Bläser
Prof. Dr. Joel Spencer

Promovierte akademische Mitarbeiterin:
Dr. Carola Winzen

To the source of the Syrian revolution: the children of Daraa

Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet. Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder ähnlicher Form in einem Verfahren zur Erlangung eines akademischen Grades vorgelegt.

Saarbrücken, 16.07.2012 *Mahmoud Fouz*

Acknowledgements

All good things come to an end. This thesis is the conclusion of my stay at Saarland University. I would like to thank everyone who supported me during this time.

- Mutter: Du bist die Beste. Ich habe Dir fast alles zu verdanken.
- Vater: Du hast viel für mich aufgegeben. Ohne Deine Unterstützung wäre ich nicht so weit gekommen.
- Hiba: Danke für den Ansporn, es Dir nachzumachen.
- Umar: Danke dafür, dass Du immer ein guter Freund warst.
- Prof. Benjamin Doerr: Ganz besonders möchte ich Dir als Mentor und Betreuer danken. Ich habe unsere wissenschaftlichen, wie auch sonstigen Diskussionen in einer freundschaftlichen Atmosphäre immer genossen. Zudem warst Du es, der entscheidend meine Forschungsinteressen geprägt hat.
- Prof. Markus Bläser: Vielen Dank für die Möglichkeit, die Du mir geboten hast, an Deinem Lehrstuhl unter besten Bedingungen zu promovieren. Danke Dir auch für Dein Vertrauen, mir freie Hand zu lassen bei der Wahl meiner Forschungsgebiete.
- Muhammad Shaheen: Thanks for being a great friend and all the fun that we had together.
- Khaled Elbassioni: Thanks a for being like an older brother at MPI. I truly enjoyed working with you a lot and will never forget our trip to the Netherlands.
- Bodo Manthey: Danke für die gemeinsamen Jahre im Raum 422. Ich werde diese Zeit immer in guter Erinnerung behalten.
- Tobias Friedrich: Danke für die gemeinsame Forschung. Ich habe Deine herzliche Art immer geschätzt.
- Raghavendra Rao: You have been a nice roommate. Thanks, man!

- Alexey Pospelov: Danke Dir für die immer interessanten, manchmal auch hitzigen Diskussionen.

Thank you and the many other people who I have met during my stay in Saarbrücken.

Contents

Preface

We live in the Information Age. The amount of information created, processed, and transferred day by day is enormous. The speed at which this is taking place is changing our world dramatically. In fact, the almost instantaneous availability of information around the world is arguably the strongest force of globalization; it is influencing our way of thinking, our way of working and our way of living. Not surprisingly understanding how information spreads quickly is therefore of great importance. Think of a company that is advertising for a new product, a party that is planning an election campaign, or a news agency that is seeking to broaden its reach.

So what are the processes that enable such fast spread of information in practice? And can we identify simple mechanics that facilitate a fast information spread?

To answer these questions, we distinguish two aspects of information spread. In the *model view* we analyze the existing processes that enable a fast spread of information. This is done by defining theoretical *models* that try to capture the main mechanics of information dissemination. In the *algorithmic view* we design new *algorithms* that guarantee a fast spread of information. In a sense both perspectives are related to each other. A better understanding of the existing processes can help us to design new algorithms. On the other hand, simple algorithms can serve as models for existing processes.

In this thesis, we contribute to both aspects from a rigorous mathematical point of view. In the first part, we study a very natural information dissemination process on *preferential attachment graphs*, which are a popular theoretical model for real-world networks. The theoretical analysis of this simple process allows us to identify characteristic properties of such networks that arguably help the fast spread of information. At the same time, the protocol that we propose lends itself to spread information efficiently in such networks.

In the second part, we propose a simple and asymptotically optimal solution to the classical telephone call problem. Here, the setting is that a person has a piece of information and wants to spread it among a group of people, say the town or a club, as fast as possible. Although this problem is well-known and has received a lot of attention in the research community, our algorithm surprisingly outperforms previous solutions in terms of running time, scalability and robustness. Our algorithm is also simple and easy to implement. In this sense, despite the maturity of this field, we strongly believe that there are still a lot of neat algorithmic ideas waiting to be discovered.

All the protocols that we study make use of *randomness*. This means that we allow the process at certain steps during the execution to choose among different options according to a well-defined probability distribution. The use of randomness has led to a large number of *probabilistic algorithms* that utilize random choices to perform difficult tasks efficiently. Often, these probabilistic algorithms beat deterministic algorithms not only in terms of running time, but also in terms of simplicity. These advantages are usually traded against a small uncertainty in terms of running time (*Las-Vegas algorithms*) or the correctness of the result (*Monte-Carlo algorithms*). In this thesis, this uncertainty becomes arbitrarily small the larger our input is; we say that the protocols run in the stated bounds *with high probability*. When we adopt the model view, randomness is used to represent the non-determinism in real-world processes.

Abstract

This thesis deals with two rumor spreading problems. In the first part, we study the rumor spreading problem in social networks modelled by preferential attachment graphs. We consider the push-pull strategy by Karp, Schindelhauer, Shenker, and Vöcking [49], where in each round, each vertex chooses a random neighbor and exchanges information with it. We prove the following. The push-pull strategy delivers a message to all nodes within $\Theta(\log n)$ rounds with high probability, where n is the number of nodes in the graph. The best known bound so far was $O(\log^2 n)$ by Chierichetti, Lattanzi, and Panconesi [19]. If we slightly modify the protocol so that contacts are chosen uniformly from all neighbors but the one contacted in the previous round, then this time reduces to $\Theta(\log n / \log \log n)$. This is asymptotically optimal since it matches the diameter of the graph. In an asynchronous version of the protocol, the running time is shown to be even $O(\sqrt{\log n})$. Parts of these results are published in [29] and accepted for publication in [30].

In the second part, we consider the rumor spreading problem on the complete graph. We propose a new push protocol that achieves an asymptotically optimal time of $(1+o(1)) \log_2 n$. It needs only $O(nf(n))$ calls, where $f(n) = \omega(1)$ can be arbitrary. The protocol is robust against random node failures. We also extend it to deal with *adversarial* node failures efficiently. These results were published in [23].

Zusammenfassung

Diese Promotionsarbeit beschäftigt sich mit zwei Problemstellungen im Zusammenhang mit dem Verbreiten von Informationen.

Im ersten Teil untersuchen wir die Verteilung von Informationen auf sozialen Netzwerken anhand des “Preferential Attachment” Modells. Hierzu betrachten wir das “Push-Pull” Protokoll von Karp, Schindelhauer, Shenker, and Vöcking [49]: In jeder Runde wählt ein Knoten einen zufälligen Nachbarknoten aus und tauscht sich mit ihm aus, d.h., wenn einer der beiden Knoten eine Information hat, erhält sie der andere. Wir zeigen folgende Resultate. Das Push-Pull Protokoll verbreitet mit hoher Wahrscheinlichkeit eine Nachricht an alle Knoten innerhalb von $\Theta(\log n)$ Runden, wobei n die Zahl der Knoten im Graph darstellt. Die beste bisher bekannte Laufzeitschranke war $O(\log^2 n)$ von Chierichetti, Lattanzi, and Panconesi [19]. Wenn wir das Protokoll leicht anpassen, so dass jeder Knoten bei der zufälligen Wahl eines Nachbarknoten den zuletzt kontaktierten ausschlieÃ§t, verbessert sich diese Schranke auf $\Theta(\log n / \log\log n)$. Dies ist asymptotisch optimal, da es dem Durchmesser des Graphen entspricht. In einer asynchronen Fassung des Protokolls reduziert sich die Laufzeit sogar auf $O(\sqrt{\log n})$. Diese Ergebnisse wurden teilweise in [29] veröffentlicht oder sind zur Veröffentlichung in [30] angenommen worden.

Im zweiten Teil betrachten wir die Verteilung von Informationen auf dem vollständigen Graphen. Wir führen ein neues “Push” Protokoll ein, das eine asymptotisch optimale Laufzeit von $(1 + o(1)) \log n$ erreicht. Dabei benötigt es nur $O(nf(n))$ Anrufe, wobei $f(n) = \omega(1)$ beliebig ist. Das Protokoll ist zudem robust gegenüber zufälligen Knotenausfällen. Ferner erweitern wir das Protokoll, so dass es auch bei gezielten Knotenausfällen effizient bleibt. Diese Ergebnisse sind in [23] veröffentlicht worden.

Chapter 1

Introduction to Rumor Spreading

Broadcasting a piece of information ("rumor") from one node ("source") to all nodes of a network is a classical problem in computer science. There are different reasons why this *rumor spreading* problem is interesting.

First, it serves as a simple model for many processes like the spread of rumors, computer viruses and diseases. It is therefore instructional to understand basic information dissemination processes before analyzing more complex models for epidemics.

Second, it serves as a basic algorithmic paradigm to *consensus problems* where different parties are seeking to compute a common value. There are many applications for such problems. One important example is that of databases that are replicated at different locations (see, e.g., [22]). To maintain consistency among the different copies, a protocol is required that forwards any update to all locations. Such synchronization processes are ubiquitous in data centers, but also in peer-to-peer networks. Another important application is the distributed computation problem where a network of processors must compute a common function. An example would be a network of sensors each of which measures the temperature at its location and the goal is to compute the average of these values (see, e.g., [13]).

Third, from a research perspective, it is a good field to try out new ideas and develop techniques for randomized algorithms. For example, ideas from the *quasirandom* protocol by Doerr, Friedrich, and Sauerwald [24] were later successfully applied for evolutionary algorithms by Doerr, Fouz, and Witt [27].

Thus, not surprisingly, rumor spreading has received a lot of attention in the research community and remains a very active topic of research. Hedetniemi, Hedetniemi, and Liestman [46] give an excellent survey of older results. More references can be found in the seminal papers by Feige, Peleg, Raghavan, and Upfal [37] and Karp, Schindelhauer, Shenker, and Vöcking [49].

For reports on the actual use of such protocols, see Demers, Greene, Hauser, Irish, Larson, Shenker, Sturgis, Swinehart, and Terry [22] and Kempe, Dobra, and Gehrke [50].

In this thesis, we assume that we are given a graph G and that at the beginning exactly one node has a rumor. The goal is to spread this rumor to all nodes in the graph. In all protocols, we only allow neighboring nodes to exchange information. We will call such broadcasting algorithms also rumor spreading *strategies* or *processes*.

The standard model assumes that each node calls at most one node per unit of time (*round*). In consequence, if only informed nodes place a call, the number of informed nodes can at most double in each round; thus, at least $\lceil \log_2 n \rceil$ rounds are needed to spread a rumor to n nodes. Using a broadcast tree that spans the network this bound can be achieved. Such deterministic protocols, however, are vulnerable to failures. In addition, the broadcast tree depends on the source; for each source, each node has to compute or store which neighbors to contact upon receiving the rumor from that source. In consequence, when the network grows, the broadcast tree has to be recomputed.

Feige et al. [37] identify three desirable properties of broadcasting algorithms: simplicity, scalability and robustness. Simple algorithms allow for an easy, less error-prone implementation. Usually, such algorithms are *local*, i.e., each node carries out a protocol without the need to know the whole network structure or how far the rumor has spread already. Scalable algorithms need no or only minor modifications when the network grows. Robustness ensures that the algorithm also works under certain failure models. No efficient deterministic algorithm is known to achieve these properties together. On the other hand, there are several randomized protocols that perform surprisingly well.

Probably the simplest of these protocols is called *randomized rumor spreading* (or *push strategy*), see, e.g., the works by Feige et al. [37], Frieze and Grimmett [43], Karp et al. [49]. It has been used to transmit information in computer networks [22, 50]. The protocol proceeds in rounds as follows: in each round, each node that already knows the rumor chooses a communication partner uniformly at random and sends her a copy of this rumor. Thus, each node runs the same randomized process independent of the source node.

For many network topologies, this strategy is a very efficient way to spread a rumor. Let n denote the number of vertices of a graph. Then the push model with high probability (i.e., with probability $1 - o(1)$) sends the rumor to all vertices in time $\Theta(\log n)$, if the graph is a complete graph [43, 61], a hypercube [37], an Erdős-Rényi random graph $G_{n,p}$ with $p \geq (1 + \varepsilon) \ln(n)/n$ [37, 44], a random regular graph [41], or an expander graph [41, 63]. Recently, Chierichetti, Lattanzi, and Panconesi [18] showed that rumor spreading is doable in logarithmic time for graphs of bounded conductance if the degrees of neighboring nodes have a constant ratio. For Cayley graphs [35] and random geometric graphs [14], the bounds of $O(\mathrm{diam}(G) + \log n)$ are known, where $\mathrm{diam}(G)$ denotes the diameter of the graph. In contrast to this, Chierichetti, Lattanzi, and Panconesi [19] showed that the push model with non-vanishing probability needs $\Omega(n^{\alpha})$ rounds in preferential attachment graphs for some $\alpha > 0$.

Opposite to the push strategy is the *pull strategy*: each vertex in each round contacts a random neighbor and learns the rumor if its contact knows the rumor. There is a symmetry between the two models. This was observed for a quasirandom version of the two models by Doerr, Friedrich, and Sauerwald [26], but similar arguments apply to the two random models discussed so far. Thus, the results also hold for the pull model.

Karp et al. [49] pointed out that for complete graphs, the pull strategy is inferior to the push strategy until roughly $n/2$ vertices are informed, and then the pull strategy becomes more effective. This motivates to combine both approaches. In this so-called *push-pull strategy* of Demers et al. [22] (see also [49]), in each round, each vertex contacts another vertex chosen uniformly at random among its neighbors. It *pushes* the rumor in case it has the rumor, and *pulls* the rumor in case the neighbor has the rumor. For complete graphs this protocol also needs $\Theta(\log n)$ rounds, though with better implicit constants [22, 34, 49]. Elsässer [34] also proved a bound of $\Theta(\log n)$ rounds for Erdős-Rényi random graphs $G_{n,p}$ with $p \geq \mathrm{polylog}(n)/n$. Chierichetti et al. [18] relate the broadcast time of the push-pull strategy to the conductance of graphs; graphs with conductance Φ have a broadcast time of $O\big(\log^2(\Phi^{-1})\,\Phi^{-1}\,\log n\big)$ with high probability. Giakkoupis [45] recently improved this bound to $O(\Phi^{-1}\log n)$ which is tight.

For preferential attachment graphs, the push-pull strategy is much better than push or pull alone. Chierichetti et al. [19] showed that with this strategy, $O(\log^2 n)$ rounds suffice with high probability. Apart from the advantages in terms of its running time, the push-pull protocol also captures the effect of gossipping in social networks better than a push or pull strategy alone.

All these results assume a *synchronous* model where all nodes take action simultaneously at discrete time steps. In many applications and certainly in real world social networks, this assumption is not very plausible. One can also argue (see, e.g., [13, 25]) that time-synchronization contradicts the idea of a self-organized broadcasting protocol. Boyd, Ghosh, Prabhakar, and Shah [13] therefore proposed an *asynchronous time model* with a *continuous* time line. Each node has its own clock that ticks at the times of a rate 1 Poisson process independent from the clocks of other nodes. It is well-known (see, e.g., [58]) that the time between two ticks (*interarrival times*) is exponentially distributed with the same parameter as the corresponding Poisson process. The protocol now specifies for each node what to do whenever its own clock ticks. Note that the idea of using a Poisson distribution to model events that occur continuously and independently from each other is not new. It is often used to model the arrival of customers in a queue (see, e.g., [58]), incoming telephone calls, etc.

The rumor spreading problem in the asynchronous time model has so far received less attention. The push-pull protocol in this model, however, turns out to be closely related to Richardson's model for the spread of a disease [62] and to first-passage percolation. In this sense, for the hypercube, Fill and Pemantle [38] and Bollobás and Thomason [11] showed that the asynchronous push-pull protocol spreads a rumor to all nodes in time $\Theta(\log n)$. Similarly, for the complete graph, Janson [48] showed a bound of $\Theta(\log n)$. Note that these bounds match the same asymptotics as in the synchronous case. We also suspect that the same bounds hold in case all but $o(n)$ nodes are to be informed.

Fountoulakis, Panagiotou, and Sauerwald [42] have recently studied the push-pull protocol in the asynchronous time model for random graphs with a given expected degree distribution that follows a power law with exponent in $(2, 3)$. These are quite different from preferential attachment graphs, e.g., their average diameter is known to be $\Theta(\log \log n)$ (see [20]), whereas for preferential attachment graphs the average diameter is also $\Theta(\log n/ \log \log n)$ (see [31]). For these random power law graphs, they show a constant runtime to inform $n - o(n)$ nodes.

1.1 Overview

This thesis has two parts. Although both parts study an information dissemination process, they are self-contained and can be read independently from each other.

In the first part, we study the performance of the push-pull strategy in preferential attachment graphs that serve as a model for social networks.

In Chapter 3, we prove the following. The push-pull strategy delivers a message to all nodes within $\Theta(\log n)$ rounds with high probability. The best known bound so far was $O(\log^2 n)$ by Chierichetti, Lattanzi, and Panconesi [17]. We prove the lower bound in Section 3.1 and the upper bound in Section 3.2. If we slightly modify the protocol so that contacts are chosen uniformly from all neighbors but the one contacted in the previous round, then this time reduces to $\Theta(\log n/ \log \log n)$, which is the diameter of the graph. This is the first time that a sublogarithmic broadcast time is proven for a natural setting. Also, this is the first time that avoiding double-contacts reduces the runtime to a smaller order of magnitude (see Section 3.2).

In Chapter 4, we study the push-pull protocol in a continuous time setting where each vertex takes action at times given by an independent Poisson process with rate 1. We show that this *asynchronous* push-pull protocol spreads a message in preferential attachment graphs in time $O(\sqrt{\log n})$ to all but a lower order fraction of the nodes with high probability.

In Chapter 6, we conduct an experimental investigation, where we confirm that memory indeed reduces the runtime of the synchronous push-pull protocol already for small network sizes. We observe that one memory cell per node suffices to reduce the runtime significantly. Besides extremely sparse graphs, preferential attachment graphs perform faster than all other graph classes examined. We complement our findings on theoretical network models by the corresponding experiments on crawls of popular online social networks. Here again, we observe extremely rapid information dissemination. We also consider the asynchronous version of the rumor spreading protocol. Here, we cannot confirm the theoretically predicted asymptotic advantage.

In the second part, we propose a new protocol for the fundamental problem of disseminating a piece of information to all members of a group of n players that are all connected to each other. It builds upon the classical randomized rumor spreading protocol and several extensions. The main achievements are the following.

Our protocol spreads a rumor from one node to all other nodes in the asymptotically optimal time of $(1 + o(1)) \log_2 n$. The whole process can be implemented in a way such that only $O(nf(n))$ calls are made, where $f(n) = \omega(1)$ can be arbitrary (see Chapter 9).

In spite of these quantities being close to the theoretical optima, the protocol remains relatively robust against failures; for *random* node failures, our algorithm again comes arbitrarily close to the theoretical optima (see Section 9.2).

The protocol can be extended to also deal with *adversarial* node failures (see Chapter 10). The price for that is only a constant factor increase in the runtime, where the constant factor depends on the fraction of failing nodes the protocol is supposed to cope with. It can easily be implemented such that only $O(n)$ calls to properly working nodes are made.

In contrast to the push-pull protocol, our algorithm only uses push operations, i.e., only informed nodes take active actions in the network. On the other hand, we discard address-obliviousness. To the best of our knowledge, this is the first *randomized* push algorithm that achieves an asymptotically optimal running time.

1.2 Notation and Basic Tools

The following are some mainly probabilistic tools that we use in this thesis. For an excellent introduction to randomized algorithms containing these results, we refer the reader to the books by Mitzenmacher and Upfal [58] and Motwani and Raghavan [59]. For a more detailed survey on large deviation bounds that are essential in the analysis of randomized algorithms we recommend the book by Dubhashi and Panconesi [32].

We start with a very rough, yet surprisingly useful estimate. The *union bound* is an essential tool in the analysis of randomized algorithms. Its strength lies in its simplicity; there are no special assumptions needed.

Lemma 1.2.1 (Union Bound). *For any finite or countably infinite sequence of events $E_1, E_2, \dots,$ we have*

$$\mathbb{P}[\bigcup_{i\geq 1} E_i] \leq \sum_{i\geq 1} \mathbb{P}[E_i].$$

A main difficulty in the analysis of randomized algorithms comes from dependencies between different random variables. A simple way to deal with such problems is to compute the expectation, which is linear *regardless of the dependencies between the random variables.*

Lemma 1.2.2 (Linearity of Expectation). *For any finite collection of discrete random variables $X_1, X_2, \dots, X_n$, with finite expectations,*

$$\mathbb{E}[\sum_{i=1}^{n} X_i] = \sum_{i=1}^{n} \mathbb{E}[X_i].$$

Once we have computed the expectation, we can easily convert it to a rough probabilistic bound by *Markov's inequality.* Note however that it only bounds the probability that a nonnegative random variable *exceeds* its expected value by much. It can not be used to bound the probability that it is significantly smaller than the expected value.

Lemma 1.2.3 (Markov's Inequality). *Let X be a random variable that assumes only nonnegative values. Then, for all $a > 0$,*

$$\mathbb{P}[X \geq a] \leq \frac{\mathbb{E}[X]}{a}.$$

For random variables that are *independent*, we can get much stronger bounds on their sum. It is usually highly concentrated around the expectation.

For independent Bernoulli random variables, i.e., variables that take only values 0 or 1, we can bound their sum by *Chernoff's bound.*

Lemma 1.2.4 (Chernoff's Inequality). *Let $X_1, X_2, \dots, X_n$ be independent Bernoulli random variables. Let $X = \sum_{i=1}^{n} X_i$ and $\mu := \mathbb{E}[X]$. Then for any $\delta > 0$, we have*

$$\mathbb{P}[X \geq (1+\delta)\mu] \leq \exp(-\min\{\delta, \delta^2\}\mu/3),$$

and

$$\mathbb{P}[X \leq (1-\delta)\mu] \leq \exp(-\delta^2\mu/2).$$

Chernoff's bound can also be used to bound the sum of *geometric* random variables, i.e., random variables that take on an integer value $i > 0$ with probability $p(1-p)^{i-1}$, where $p \in [0,1]$ is a parameter. Such variables have an expected value of $1/p$.

Lemma 1.2.5. *Let $X_1, X_2, \dots, X_n$ be independent geometric random variables with parameter $p \in [0,1]$. Let $X = \sum_{i=1}^n X_i$. Then for any $\delta > 0$, we have*

$$\mathbb{P}[X \geq (1+\delta)\tfrac{n}{p}] \leq \exp(-\tfrac{\delta^2}{2(1+\delta)}n),$$

Since this bound is not found in the given references, we give a quick proof here.

Proof. Let $k := (1+\delta)\frac{n}{p}$. We reduce the event $X \geq k$ to an equivalent event with Bernoulli random variables and then apply Chernoff's bound. Let $Y = \sum_{i=1}^k Y_i$ be the sum of k independent and identically distributed Bernoulli variables with $\mathbb{P}[Y_i = 1] = p$. But then,

$$\mathbb{P}[X \geq k] = \mathbb{P}[Y \leq n].$$

For the latter term, we can now apply Chernoff's bound and obtain:

$$\begin{aligned}\mathbb{P}[Y \leq n] &= \mathbb{P}[Y \leq (1 - \tfrac{\delta}{1+\delta})kp] \\ &\leq \exp(-(\tfrac{\delta}{1+\delta})^2\, \mathbb{E}[Y]/2) \\ &= \exp(-\tfrac{\delta^2}{2(1+\delta)}n).\end{aligned}$$

□

In case the random variables are bounded arbitrarily, we use *Hoeffding's bound.*

Lemma 1.2.6 (Hoeffding's Inequality)**.** *Let $X_1, X_2, \dots, X_n$ be independent bounded random variables such that $X_i \in [a_i, b_i]$ with probability 1. Let $X = \sum_{i=1}^n X_i$. Then for any $t > 0$, we have*

$$\mathbb{P}[X - \mathbb{E}[X] \geq t] \leq \exp\Big(-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\Big),$$

and

$$\mathbb{P}[\mathbb{E}[X] - X \geq t] \leq \exp\Big(-\frac{2t^2}{\sum_{i=1}^n (b_i - a_i)^2}\Big).$$

Sometimes we want to bound the probability that some *function* defined on a set of independent random variables deviates significantly from its expectation, when the value of the function is affected only slightly if a single argument is changed. Such a bound is provided by *Azuma's inequality.* We will use it in the following version (see, e.g., [32, Corollary 5.2]).

Lemma 1.2.7 (Azuma's Inequality)**.** *Let $X_1, \dots, X_n$ be independent random variables, with X_i taking values in a set Ω_i for each i. Suppose that the (measurable) function $f : \prod_{i=1}^n \Omega_i \to \mathbb{R}$ satisfies*

$$|f(x) - f(x')| \leq c_i,$$

whenever the vectors x and x' differ only in the ith coordinate. Let $Y := f(X_1, \dots, X_n)$. Then for any $t > 0$,

$$\mathbb{P}(|Y - \mathbb{E}(Y)| \geq t) \leq 2 \exp\Big(-\frac{2t^2}{\sum_{i=1}^n c_i^2}\Big).$$

Finally, we will also need the following inequality that holds for $x \in (0, 1)$:

$$1 - e^{-x} \geq \tfrac{x}{2}. \tag{1.2.1}$$

Throughout the thesis, we denote by $\log n$ the logarithm to base 2 and by $\ln n$ the natural logarithm.

Part I

Randomized Rumor Spreading in Social Networks

Chapter 2

Introduction to Rumor Spreading in Social Networks

Social networks like Facebook and Twitter are reshaping the way people take collective actions. They have played a crucial role in the recent uprisings of the 'Arab Spring' and the 'London riots'. It has been argued that the 'instantaneous nature' of these networks influenced the speed at which the events were unfolding [4].

It is quite remarkable that social networks spread news so fast. Both the structure of social networks and the process that distributes the news are not designed with this purpose in mind. On the contrary, they are not designed at all, but have evolved in a random and decentralized manner.

So is our view correct that social networks ease the spread of information ("rumors"), and if so, what particular properties of social networks are the reason for this? To answer these questions, we analyze a simple rumor spreading process on an abstract model of social networks, the so-called preferential attachment graphs introduced by Barabási and Albert [3]. We assume that the rumor is sufficiently interesting so that people learn it when talking to someone knowing it. This is substantially different to the probabilistic virus spreading model [5], where the probability of becoming infected is proportional to the number of neighbors being infected.

We obtain a mathematical proof that rumors in such networks spread much faster than in many other network topologies—even faster than in networks having a communication link between any two nodes (*complete graphs*). As an explanation, we observe that nodes of small degree build a short-cut between those having large degree (*hubs*), which due to their large number of possible communication partners less often talk to each other directly.

We also simulate this process on several graphs having the structure of existing large social networks. We see, for example, that a rumor started at a random node of the Twitter network in average reaches 45.6 million of the total of 51.2 million members within only eight rounds of communication.

We note that the communication process is different in each social network. The push-pull model we regard naturally captures best a personal communication between two individuals as by phone or exchanging text messages, emails or other directed communications. Many online social networks allow also other ways of communication like posts on user's personal pages, possibly resulting and his friends to be notified of the post when they next log in, and them forwarding the news given that it is sufficiently interesting. Such forms of communication can be modelled only by more complicated mechanisms than the push-pull protocol.

2.1 Social Networks

Social networks arise in a variety of contexts. They are formed by people, who are connected by knowing each other, *Facebook* members by agreeing on being friends (in Facebook), scientific authors by having a joint publication, or actors appearing in the same movie.

Despite this diversity, many networks share characteristic properties. Well known is the observation that any two individuals are connected by just "six degrees of separation", which was first formulated by Karinthy (see Barabási [2]) and became known to a broad audience through Milgram's "small world study" [56]. Similarly for the world wide web, Albert, Jeong, and Barabasi [1] predicted a *diameter* (maximum distance between two nodes in the graph) of only 19 in the network formed by web pages and links between them.

Several experimental studies [1, 15, 52] revealed another intrinsic property of social networks: the histogram of the node connectivity follows a power-law; the number of nodes with k neighbors is inversely proportional to a polynomial in k.

To explain this phenomenon, Barabási and Albert [3] suggested the preferential attachment (PA) model for real-world networks that show a power-law. The model is widely used, also because of its simplicity. The paper [3] is currently the fifth most cited article in "Science" according to ISI Web of Knowledge. In the preferential attachment model, the graphs are constructed in a random, 'rich-get-richer' fashion: a newly entering node connects to m existing ones chosen randomly, but gives preference to nodes that are already popular, that is, have many neighbors. Note that the parameter m controls the density of the graph, i.e., the ratio of the number of present edges to the number of all possible edges. For these graphs, the authors of [3] empirically discovered a power-law of k^{-3}, which was later proven mathematically by Bollobás, Riordan, Spencer, and Tusnády [12]. A number of similar models emerged at the same time, all leading to a power-law distribution. It is known, though, that the PA model does not share all properties observed in the real-world networks, e.g., it is less clustered.

Still, the preferential attachment model has been successfully used to deduce many interesting properties of social networks. Famous structural results prove a small diameter of such graphs [10], determine their degree distribution [12], show high expansion properties [55], and a high robustness against random damage, but a vulnerability to malicious attacks [8, 9, 21, 39]. Algorithmic works show that in such networks, viruses spread more easily than in many other network topologies [5], or that gossip-based decentralized algorithms can approximate averages better [13].

2.2 Our Results

We study the push-pull protocol in both the synchronous and asynchronous time model on PA graphs.

In the synchronous time model, we prove that the rumor is spread to all nodes in time $\Theta(\log n)$. If we assume a slightly more clever process, namely that contacts are chosen uniformly at random among all neighbors except the one that was chosen just in the round before, then $O(\log n/\log\log n)$ rounds suffice (see Theorem 2.6.1). This is asymptotically optimal as the diameter of a PA graph is $\Theta(\log n/\log\log n)$ [12]. We note that the same asymptotic runtime is achieved with the standard push-pull protocol without any memory when *almost* all nodes are to be informed, i.e., all but $o(n)$ nodes. On the other hand, excluding nodes contacted in a constant number of previous rounds has almost no effect on classic network topologies; by checking the proofs of the results cited above, we see that also in this case the $\Theta(\log n)$ bound remains valid for complete graphs, hypercubes and random graphs. The quasirandom protocol of Doerr et al. [24] is a way of excluding *all* previous contacts. It has been investigated only in the push model, where again many known $\Theta(\log n)$ run time bounds have been verified.

The idea of excluding previously contacted nodes is not new. Elsässer and Sauerwald [36] used the exclusion of the previous three contacts to design protocols that reduce the number of messages sent, an aspect important when using such protocols to disseminate information in networks, e.g., to maintain distributed databases [22]. However, excluding previous contacts so far did not yield a faster rumor spreading. In fact, Elsässer and Sauerwald [36] have shown that the $\Omega(\log n)$ lower bound for rumor spreading in Erdős-Rényi random graphs $G_{n,p}$, where $p > \operatorname{polylog}(n)/n$, remains true if arbitrary exclusion schemes are used.

In the asynchronous model, we prove that the push-pull protocol spreads a rumor in time $O(\sqrt{\log n})$ to n nodes in the PA model with high probability. The protocol thus beats the average distance of $\Theta(\log n/\log\log n)$. To inform all nodes, however, the protocol also needs $\Theta(\log n)$ time. This is mainly due to a few nodes that require $\Omega(\log n)$ time to contact or be contacted by a neighbor for the first time. In contrast to the synchronous model, even when previously contacted nodes are excluded, the protocol still needs $\Omega(\log n)$ to inform all nodes.

These results show that the asynchronous push-pull protocol behaves quite differently from the synchronous one, despite the fact that each node still contacts one neighbor per time unit on average. The discrepancy between informing all nodes and almost all nodes reflects an often observed 'long tail' behavior in real world networks. Such effects are less visible in the synchronous case.

We complement these theoretical results by the corresponding experiments. In the synchronous case, these empirical results show that memory truly reduces the runtime of the protocol already for small network sizes in the preferential attachment model. Such a reduction is not visible for other graph classes.

Also the asynchronous protocol shows a clear advantage over the synchronous protocol. However, contrary to the theoretical findings, no asymptotic advantage for preferential attachment graphs over other graph classes was observed. We expect that the theoretically proven asymptotic behavior can only be observed for very large graphs.

To support the common observation that news spreads very fast in social networks, we have also simulated the rumor spreading process on samples of the *Twitter* and *Orkut* social networks (taken from [6, 53]) as well as preferential attachment graphs of the same size. As most social networks have a similar structure, we have chosen these large networks, for which data was readily available, as instances of social networks. For comparison, we have also included into our investigation complete graphs and *random-attachment graphs* (also called m-out model, see, e.g., Bohman and Frieze [7]), in which each node chooses m neighbors uniformly at random from all nodes. These experiments show that news spreads much faster in the real-world networks and the preferential attachment graphs than in the complete and random-attachment graphs. For the Twitter experiment, a considerable difference between the preferential attachment model and the real-world graph is visible, indicating that the Twitter graph is not captured that well by the theoretical model.

2.3 Preferential Attachment Graphs

Preferential attachment graphs were first introduced by Barabási and Albert [3]. In this work, we follow the formal definition of Bollobás et al. [10, 12]. Let G be an undirected graph. We denote by $\deg_G(v)$ the degree of a vertex v in G.

Definition 2.3.1 (Preferential attachment graph)**.** *Let $m \geq 2$ be a fixed parameter. The random graph G_m^n is an undirected graph on the vertex set*

$V := \{1, \dots, n\}$ inductively defined as follows.

- *G_m^1 consists of a single vertex with m self-loops.*
- *For all $n > 1$, G_m^n is built from G_m^{n-1} by adding the new node n together with m edges $e_n^1 = \{n, v_1\}, \dots, e_n^m = \{n, v_m\}$ inserted one after the other in this order. Let $G_{m,i-1}^n$ denote the graph right before the edge e_n^i is added. Let $M_i = \sum_{v \in V} \deg_{G_{m,i-1}^n}(v)$ be the sum of the degrees of all the nodes in $G_{m,i-1}^n$. The endpoint v_i is selected randomly such that $v_i = u$ with probability $\deg_{G_{m,i-1}^n}(u)/(M_i + 1)$, except for n that is selected with probability $(\deg_{G_{m,i-1}^n}(n) + 1)/(M_i + 1)$.*

This definition implies that when e_n^i is inserted, the vertex v_i is chosen with probability proportional to its degree (except for $v_i = n$). Since many real-world social networks are conjectured to evolve using similar principles, the PA model can serve as a model for social networks. Note that we can generate G_m^n from G_1^{mn} by identifying the nodes $1, \dots, m$ to form node 1, $m + 1, \dots, 2m$ to form node 2 and so on.

An important property observed in many real-world networks is a characteristic degree distribution that follows a power law. For preferential attachment graphs it has been formally proven that the degree distribution follows a power law with exponent equal to 3.

Theorem 2.3.1 (Bollobás et al. [12]). *Let $m \geq 1$ be fixed. We denote by $\#_m^n(d)$ the number of nodes of indegree d in G_m^n. Let*

$$\alpha_m(d) := \frac{2m\,(m+1)}{(d+m)\,(d+m+1)\,(d+m+2)},$$

and let $\varepsilon > 0$ be fixed. Then, with probability $1 - o(1)$, we have

$$(1-\varepsilon)\,\alpha_m(d) \leq \frac{\#_m^n(d)}{n} \leq (1+\varepsilon)\,\alpha_m(d),$$

for every d in the range $0 \leq d \leq n^{1/5}$.

For $m = 1$ the graph is disconnected with high probability; so we focus on the case $m \geq 2$. Under this assumption, Bollobás and Riordan [10] showed that the diameter is only $\Theta(\ln n / \ln \ln n)$ with high probability.

With a slight abuse of notation we write $(u, v) \in E$ or $(v, u) \in E$ both to denote $\{u, v\} \in E$. The definition of G_m^n can lead to multiple edges and self-loops, though they typically make up only a vanishing fraction of the edges.

2.4 The Synchronous Protocol

The synchronous push-pull protocol assumes a discrete time line. One unit of time is called a time step or round. All nodes take action simultaneously in these rounds. We assume that in the beginning one node s has the rumor.

Definition 2.4.1 (Synchronous push-pull protocol). *Let $M \geq 0$ be a fixed parameter. Assume that every vertex can store M vertices. The protocol runs as follows:*

- *In each round $t \geq 1$, every vertex u chooses uniformly at random a neighbor v which it has not contacted in the last $\min\{\deg(u) - 1, M\}$ rounds. If u knows the rumor, it sends the rumor to v ("push"). If v knows the rumor, it sends the rumor to u ("pull").*

Note that for $M = 0$, this is the standard push-pull strategy.

2.5 The Asynchronous Protocol

The asynchronous protocol assumes a continuous time line; each node has a clock that ticks independently from the clocks of other nodes at the times of a rate 1 Poisson process. A node takes action whenever its clock ticks.

Definition 2.5.1 (Asynchronous push-pull strategy). *Whenever the clock of a vertex u ticks, it chooses uniformly at random a neighbor v. If u knows the rumor, it sends the rumor to v ("push"). If v knows the rumor, it sends the rumor to u ("pull").*

The following lemma follows directly from [58, Theorem 8.13]

Lemma 2.5.1. *Let $N(t)$ denote the number of ticks of a Poisson process with parameter λ in the time interval $[0, t]$. Then, for all $s > 0$, and any integer $k \geq 0$, we have,*

$$\mathbb{P}[N(t+s) - N(t) = k] = e^{-\lambda t} \frac{(\lambda t)^k}{k!}. \tag{2.5.1}$$

We call the time span between two ticks of a clock a *round*. The length of a round is *exponentially* distributed with mean 1 (see, e.g., [58]), i.e., if T denotes the time span until the clock of a node ticks, then

$$\mathbb{P}[T > x] = e^{-x}.$$

Since the exponential distribution is memoryless (i.e., $\mathbb{P}[T \geq s + t \mid T > t] = \mathbb{P}[T > s]$) the length of a round is independent over time.

The following lemma shows that also the time when a node contacts a *specific* neighbor is exponentially distributed.

Lemma 2.5.2. *Let u be a node of degree d that is connected to a node v. Let T denote the time span until u contacts v. Then, $\mathbb{P}[T > x] = e^{-x/d}$. Thus, T is also memoryless.*

The proof follows immediately from the following lemma by declaring the event that u contacts v as *type 1.*

Lemma 2.5.3. *[58, Theorem 8.13] Suppose that we have a Poisson process $N(t)$ with rate λ. Each event is independently labeled as being type 1 with probability p or type 2 with probability $1-p$. Then the type-1 events form a Poisson process $N_1(t)$ of rate λp, the type-2 events form a Poisson process $N_2(t)$ of rate $\lambda(1-p)$, and the two Poisson processes are independent.*

We say that an edge (u, v) *fires*, whenever the clock of node u ticks and u contacts v.

2.6 Statement of Results and Proof Structure

For the synchronous case, we prove the following result.

Theorem 2.6.1. *Let s be an arbitrary node in G_m^n. With probability $1-o(1)$, the synchronous push-pull protocol broadcasts a rumor from s to all nodes in*

- $\Theta(\ln n)$ *rounds, if* $M = 0$,
- $\Theta(\ln n/ \ln\ln n)$ *rounds, if* $M \geq 1$.

For the asynchronous case, we achieve a much better running time in informing *almost* all nodes.

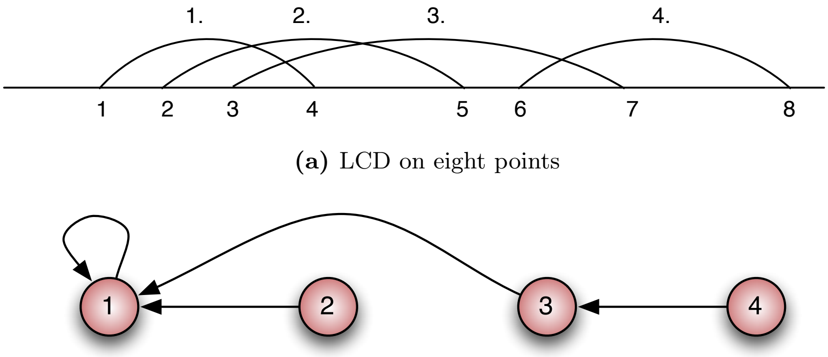
Theorem 2.6.2. *With probability $1-o(1)$, the asynchronous push-pull protocol broadcasts a rumor from any node of G_m^n to*

- *all nodes in time* $\Theta(\ln n)$,
- *all but $o(n)$ nodes in time* $O(\sqrt{\ln n})$.

It should be noted that the improved runtime of $O(\ln n/ \ln\ln n)$ in the synchronous case is also achieved without memory if we are only interested in informing almost nodes. On the other hand, the use of memory does not improve the runtime in the asynchronous protocol except by constant factors.

The proofs of the upper bounds in Theorem 2.6.1 and Theorem 2.6.2 consist of three main steps. First, we analyze the time needed until the rumor reaches a so-called *useful node.* Roughly speaking, a node is useful if its degree is at least polylogarithmic (see Section 2.8 for details).

Second, we show that once a useful node u has been informed, in the synchronous and asynchronous case, within $O(\ln n/ \ln\ln n)$ steps and $O(\sqrt{\ln n})$ time, respectively, the rumor is propagated to node 1. To this aim, we show that there is a short path from u to 1 such that every second node has degree exactly m and that is traversed in $O(\ln n/ \ln\ln n)$ steps or $O(\sqrt{\ln n})$ time,



(a) LCD on eight points

(b) Corresponding graph

Figure 2.1: Example of a linear chord diagram and corresponding graph

respectively. For the synchronous case, the nodes of degree m act as 'fast nodes', i.e., they forward the rumor from one informed neighbor to another one in a constant number of steps. Since the path has length $O(\ln n/\ln\ln n)$, the result follows. For the asynchronous case, we prove a much faster traversal by exploiting edges that fire fast. In particular, we use the fact that the minimum of k i.i.d. exponential random variables with mean 1 is also exponentially distributed with mean $1/k$. Thus, by moving along fast edges, the rumor reaches node 1 in time less than the distance to node 1.

Finally, we use a symmetry property of both protocols to show that in $O(\ln n/\ln\ln n)$ steps and $O(\sqrt{\ln n})$ time, respectively, the rumor is sent from node 1 to the other nodes.

2.7 Alternative Model

In the random process generating G_m^n, the random decisions made at each step depend heavily on the previous random decisions. To deal with these dependencies, Bollobás and Riordan [10] suggested an alternative way of generating G_m^n, that is more accessible. We first describe the model for $m = 1$. Since this case suffices to generate G_m^n for general m (by reducing G_m^n to G_1^{mn}), it is easy to generalize the model to arbitrary m.

Consider a partition of the set $\{1, 2, \ldots, 2n\}$ into n pairs. We can represent such a *pairing* by a *linear chord diagram* (LCD) (see Figure 2.1a), that consists of $2n$ distinct points on a horizontal line that are paired off by chords. We can transform such a diagram into a graph as follows (see Figure 2.1b). We insert a node for each chord. Starting from the left endpoint of each chord, we connect the corresponding node with the node corresponding to the chord of the first right endpoint reached. We claim that an LCD with $2n$ distinct points where the set of chords are chosen uniformly at random generates G_1^n. To see this note that a random LCD with n chords can be generated from a random LCD with $n-1$ chords by inserting a new chord whose right endpoint is to the right of all points and its left endpoint is inserted uniformly at random among all $2n-1$ possible places. This corresponds to adding a node to the graph and connect it to another node chosen with probabilities proportional to the degrees just like in the preferential attachment model.

Here, we use the following way to generate a random pairing of points in $[0,1]$. Let (x_i, y_i) for $i \in [n] := \{1, 2, \dots, n\}$ be n independently and uniformly chosen pairs from $[0,1] \times [0,1]$. With probability 1, all these numbers are distinct. By reordering within each pair, we assume that $x_i < y_i$ for every $i \in [n]$. It is easy to see that if we regard each pair (x_i, y_1) as a chord, we obtain an LCD with n chords that is distributed uniformly at random among all LCDs with n chords. Suppose that after relabeling, we have $y_1 < y_2 < \cdots < y_n$. We set $W_0 := 0$ and $W_i := y_i$ for $i \in [n]$. The graph G_1^n is now defined by having an edge (i,j) if and only if $W_{j-1} < x_i < W_j$. Note that this corresponds to the same transformation of an LCD to a graph described above. Define $w_j := W_j - W_{j-1}$.

Similarly, for G_m^n, we sample mn pairs $(x_{i,j}, y_{i,j})$ independently and uniformly from $[0,1] \times [0,1]$ with $x_{i,j} < y_{i,j}$ for $i \in [n]$ and $j \in [m]$. We relabel the variables such that

$$y_{1,1} < y_{1,2} < \cdots < y_{1,m} < y_{2,1} < \cdots < y_{2,m} < \cdots < y_{n,1} < \cdots < y_{n,m}.$$

We set $W_0 := 0$ and $W_i := y_{i,m}$ for $i \in [n]$. The graph is now defined by having an edge (i,j) for each $k \in [m]$ such that $W_{j-1} < x_{i,k} < W_j$. As before, define $w_j = W_j - W_{j-1}$. We write $\ell_{i,k}$ for the node j such that $W_{j-1} < x_{i,k} < W_j$.

Note that given $y_{1,1}, \dots, y_{n,m}$, the random variables $x_{1,1}, \dots, x_{n,m}$ are independent with x_{i_k} being chosen uniformly from $[0, y_{i,k}]$. For a better readability, we will always use the following bounds. For $i \geq j$, we have

$$\mathbb{P}[\ell_{i,k} = j] = \frac{w_j}{y_{i,k}},$$

and thus,

$$\frac{w_j}{W_i} \leq \mathbb{P}[\ell_{i,k} = j] \leq \frac{w_j}{W_{i-1}}. \tag{2.7.1}$$

The bounds (2.7.1) allow us to work with the values of the W_i's and ignore the values of the $y_{i,j}$'s.

We give a few properties of the alternative model, which hold with high probability and are useful in the analysis. Let $s = 2^a$ be the smallest power of 2 larger than $\ln^7 n$, and let 2^b be the largest power of 2 smaller than $2n/3$. Let $I_t = [2^t + 1, 2^{t+1}]$.

Lemma 2.7.1 (Bollobás and Riordan [10])**.** *Let $m \geq 2$ be fixed. Using the definitions above, each of the following five events holds with probability $1 - o(1)$:*

- $E_1 := \big\{|W_i - \sqrt{i/n}| \leq \frac{1}{10}\sqrt{i/n}$ *for all* i*, where* $s \leq i \leq n\big\}$
- $E_2 := \big\{I_t$ *contains at least* 2^{t-1} *nodes* i *with* $w_i \geq \frac{1}{10\sqrt{in}}$ *for all* t*, where* $a \leq t < b\big\}$
- $E_3 := \{w_1 \geq \frac{4}{\ln(n)\sqrt{n}}\}$
- $E_4 := \{w_i \geq \ln^2(n)/n$ *for* $i < n^{1/5}\}$
- $E_5 := \{w_i < \ln^2(n)/n$ *for* $i \geq n/2\}$*.*

Note that the event E_5 is slightly adjusted for our purposes. In the original paper, the authors show that for $i \geq n/\ln^5 n$, we have $w_i < n^{-4/5}$. It is easy to check that essentially the same proof holds for the above version. For completeness, we provide the slightly modified proof.

Proof of E_5. Suppose that E_1 holds, but E_5 does not hold. Let $\delta = \frac{\ln^2(n)}{n}$. Then for some x, $0.6 < x < 1 - \delta$, the interval $[x, x + \delta]$ contains no W_i, and hence contains at most $m - 1$ of the $y_{i,j}$. We partition this interval into m disjoint intervals of each of size $\delta' = \delta/m$.

Setting $\delta' = \delta/(m+1)$, each such interval contains m disjoint intervals of the form $[t\delta', (t+1)\delta')$ with t an integer and $0.6 < t\delta' < 1\delta'$, one of which must contain no $y_{i,j}$. For a given t, the number of $y_{i,j}$ in $[t\delta', (t+1)\delta')$ has a binomial distribution $\mathrm{Bi}(mn, p_t)$ with

$$p_t = (2t+1)\delta'^2 > 1.2\delta > \ln^2(n)/((m+1)n),$$

where the last inequality holds for sufficiently large n. The probability that no $y_{i,j}$ lies in this interval is thus

$$(1p_t)^{mn} \leq e^{-mnp_t} < e^{-m\ln^2 n/(m+1)} = n^{-O(\ln n)}.$$

Summing over the $O(n/\ln^2 n)$ values of t shows that the probability that E_1 holds, but E_5 does not is $o(1)$, completing the proof of the lemma. □

Instead of working directly with the alternative model where the W_i's are random variables, we use the following *typical social network* model where we assume the W_i's to be fixed numbers that satisfy the properties $E_1, \ldots, E_5$. Since by Lemma 2.7.1, these properties hold with high probability, all results proven for a typical social network model carry over to G^n_m with high probability. More precisely, Let $0 < W_1 < \cdots < W_n < 1$ be distinct real numbers and let $w_i = W_i - W_{i-1}$. Assume that $W_1, \ldots, W_n$ satisfy the properties $E_1, \ldots, E_5$. A typical social network $G_m(W_1, \ldots, W_n)$ is obtained by connecting each node i with the nodes $\ell_{i,1}, \dots, \ell_{i,m}$, where each $\ell_{i,k}$ is a node chosen randomly with $\frac{w_j}{W_i} \leq \mathbb{P}[\ell_{i,k} = j] \leq \frac{w_j}{W_{i-1}}$ for all $j \leq i$.

In the remainder of the paper we will always assume to have a typical social network $G_m(W_1, \dots, W_n)$. For simplicity, we will write $G := G_m(W_1, \dots, W_n)$ to denote a (random) typical social network.

2.8 Useful Nodes

We use the notion of a *useful* node that was introduced by Bollobás and Riordan [10]. A node i is useful if $w_i \geq \ln^2(n)/n$. Note that we are slightly relaxing the original definition in [10] where the authors also assumed that $i \leq n/\ln^5(n)$. For our purposes, by E_5, we have $i < n/2$ for all useful nodes. Furthermore by E_4, every $i < n^{1/5}$ is useful. We now prove several properties of non-useful nodes. Remember that $\deg_G(v)$ denotes the degree of node v in graph G.

Lemma 2.8.1. *With probability* $1 - n^{-\Omega(\ln n)}$, *the following event holds*

- $E_6 := \{\deg_G(v) \leq 5m \ln^2 n$ *for all non-useful* $v\}$.

Proof. Let i be a fixed non-useful node. So $w_i < \ln^2(n)/n$ and by E_4, $i \geq n^{1/5}$. Consider any node $j > i$. By E_1, we have $W_{j-1} \geq \frac{1}{2}\sqrt{(j-1)/n}$. Moreover, for any $k \in \{1, \dots, m\}$, we have by (2.7.1)

$$\mathbb{P}[\ell_{j,k} = i] \leq w_i/W_{j-1} \leq \frac{2 \ln^2 n}{n\sqrt{(j-1)/n}}.$$

Denote by $\deg_G^+(i)$ the number of edges $(j, i) \in E$ with $j > i$. Then $\deg_G(i) \leq 2m + \deg_G^+(i)$, where the first term is due to the at most m self-loops at i. We have

$$\begin{aligned}
\mathbb{E}[\deg_G^+(i)] &= \sum_{j>i} \sum_{k=1}^{m} \mathbb{P}[\ell_{j,k} = i] \\
&\leq 2m \ln^2(n) n^{-1/2} \sum_{j>i}^{n} (j-1)^{-1/2} \\
&\leq 2m \ln^2(n) n^{-1/2} \sum_{j \geq i}^{n-1} j^{-1/2} \\
&\leq 2m \ln^2(n) n^{-1/2} \int_i^n j^{-1/2} \, \mathrm{d}j \\
&\leq 4m \ln^2(n).
\end{aligned}$$

By Chernoff's bound, we have $\mathbb{P}[\deg_G^+(i) \geq 4.5m \ln^2 n] \leq e^{-\Omega(\ln^2(n))} = n^{-\Omega(\ln n)}$. By a union bound over all non-useful nodes, we conclude that with probability $1 - n^{-\Omega(\ln n)}$ all non-useful nodes have degree at most $2m + 4.5m \ln^2 n \le 5m \ln^2 n$. □

We call a cycle *non-useful* if it consists only of non-useful nodes.

Lemma 2.8.2. *With probability* $1 - o(1)$, *the following event holds*

- $E_7 := \{G$ *contains* $(\ln n)^{O(\ln^{3/4} n)}$ *non-useful cycles of length at most* $\ln^{3/4} n\}$.

Proof. Let $\ell \in [n]$. We first bound the number of non-useful cycles of length ℓ. For simplicity, we assume that ℓ is even. The case when ℓ is odd is similar. Let $i_1 < i_2 < \cdots < i_\ell$ be ℓ distinct non-useful nodes. We set $i_{\ell+1} := i_1$. The probability that $i_1, \ldots, i_\ell, i_{\ell+1} = i_1$ form a cycle in G in this order is

$$\begin{aligned}
\mathbb{P}\Big[\bigwedge_{j=1}^{\ell} (i_j, i_{j+1}) \in E\Big] &\overset{\text{by (2.7.1)}}{\le} \prod_{j=1}^{\ell} \Big(m \max\{ \frac{w_{i_{j+1}}}{W_{i_j - 1}}, \frac{w_{i_j}}{W_{i_{j+1}-1}} \} \Big) \\
&\le m^\ell \prod_{j=1}^{\ell} \Big(\frac{\ln^2 n}{n} \max\{ W_{i_j-1}^{-1}, W_{i_{j+1}-1}^{-1} \} \Big) \\
&\overset{\text{by } E_1}{\le} m^\ell \prod_{j=1}^{\ell} \Big(\frac{10 \ln^2 n}{9\sqrt{n}} \max\{ \tfrac{1}{\sqrt{i_j - 1}}, \tfrac{1}{\sqrt{i_{j+1}-1}} \} \Big) \\
&\le m^\ell \prod_{j=1}^{\ell/2} \Big(\Big(\frac{10 \ln^2 n}{9\sqrt{n}} \Big)^2 \frac{1}{i_j - 1} \Big) \\
&\le m^\ell \Big(\frac{10 \ln^2 n}{9\sqrt{n}} \Big)^\ell \prod_{j=1}^{\ell/2} \frac{1}{i_j - 1}.
\end{aligned}$$

Note that the same upper bound holds for every other cycle consisting of $i_1, i_2, \ldots, i_\ell$. Thus we can bound the expected number of cycles consisting of these nodes by $\ell!\, m^\ell \Big(\frac{10 \ln^2 n}{9\sqrt{n}} \Big)^\ell \prod_{j=1}^{\ell/2} \frac{1}{i_j - 1}$. In consequence, the expected number of non-useful cycles of length ℓ is bounded by

$$\begin{aligned}
&\sum_{i_1 < \cdots < i_\ell} \ell!\, m^\ell \Big(\frac{10 \ln^2 n}{9\sqrt{n}} \Big)^\ell \prod_{j=1}^{\ell/2} \frac{1}{i_j - 1} \\
&\le \ell!\, m^\ell \Big(\frac{10 \ln^2 n}{9\sqrt{n}} \Big)^\ell \sum_{i_{\ell/2+1} < \cdots < i_\ell} \sum_{i_1 < \cdots < i_{\ell/2}} \prod_{j=1}^{\ell/2} \frac{1}{i_j - 1} \\
&\le \ell!\, m^\ell \Big(\frac{10 \ln^2 n}{9\sqrt{n}} \Big)^\ell n^{\ell/2} \Big(\sum_{i=1}^{n} \tfrac{1}{i} \Big)^{\ell/2} \\
&\le \ell!\, m^\ell \, (\tfrac{10}{9})^\ell \, (\ln n)^{2\ell} \, (1 + \ln n)^{\ell/2} \\
&\le \ell!\, m^\ell \, (\ln n)^{3\ell},
\end{aligned}$$

where the last inequality holds for sufficiently large n.

By Markov's inequality, we conclude that with probability at most $\frac{1}{\ln n}$, there are more than $\ell!\, m^{\ell}(\ln n)^{3\ell+1}$ non-useful cycles of length ℓ. By a simple union bound, it follows that the number of non-useful cycles of length at most $\ln^{3/4} n$ is at most $(\ln^{3/4} n)^{1+\ln^{3/4} n}\, m^{\ln^{3/4} n}\, (\ln n)^{3\ln^{3/4}(n)+1} = (\ln n)^{O(\ln^{3/4} n)}$ with probability at least $1-(\ln n)^{-1/4} = 1-o(1)$. □

Lemma 2.8.3. *Assume that E_6 holds and let $K = \frac{\ln n}{(\ln\ln n)^2}$. With probability $1-n^{-1/5+o(1)}$, the following event holds*

- *$E_8 :=$ {for all non-useful v, there exists at most one cycle whose nodes are all connected to v via paths of length at most K that consist only of non-useful nodes}.*

In order to prove Lemma 2.8.3, we show a few auxiliary results. The following lemma bounds the probability that two fixed non-useful nodes (not necessarily distinct) are neighbors.

Lemma 2.8.4. *Let v, v' be two fixed non-useful nodes. Then the probability that $(v, v') \in E$ is at most $n^{-3/5+o(1)}$.*

Proof. W.l.o.g. assume that $v \geq v'$. Since v' is not useful, we have $w_{v'} < \ln^2(n)/n$. By E_4, we have $v \geq v' \geq n^{1/5}$. Using E_1 we obtain

$$W_{v-1} \geq W_{\lceil n^{1/5}\rceil -1} \geq \tfrac{1}{2} n^{-2/5}$$

and thus

$$\mathbb{P}[(v,v') \in E] \leq m w_{v'}/W_{v-1} \leq n^{-3/5+o(1)}. \tag{2.8.1}$$

□

Lemma 2.8.5 (Bollobás and Riordan [10])**.** *Let v be a fixed non-useful node. Then for all $k \in [m]$, the probability that $\ell_{v,k}$ is a useful node is at least $\ln^{-3} n$. This event is independent from all other random decisions $\ell_{v',k'}$ with $(v',k') \neq (v,k)$.*

Note that in the original lemma, the authors only state a bound on the probability that $\ell_{v,1}$ is a useful node. However, the same proof yields the above version. Also note that Lemma 2.8.4 and Lemma 2.8.5 remain valid if we condition on E_6.

For the next technical lemma, we need some notation. Let v be a non-useful node. Let $L^0 = \{v\}$ and for $k \geq 1$, we define

$$L^k := \{i \in [n] \mid i \text{ is not useful} \wedge i \notin L^1, \ldots, L^{k-1} \wedge \exists i' \in L^{k-1} \colon (i',i) \in E\}.$$

We define $L^{\leq k} := \bigcup_{i=0}^{k} L^i$. Let $n_k := |L^k|$. We say that level L^i *causes a collision* if there exist two nodes $j, j' \in L^i$ (not necessarily distinct) that are either neighbors or share a common neighbor in L^{i+1}, or if there exists a node $j \in L^i$ that is connected to a node in L_{i+1} by two links, i.e., j is incident to a multi-edge. Note that each collision caused by L^i corresponds to a cycle that is connected to v via a path consisting only of nodes in $L^{\leq i+1}$ and vice versa, every such cycle corresponds to one collision in some L^j where $j \leq i+1$. Hence, the number of collisions in $L^{\leq k}$ is exactly the number of cycles in $L^{\leq k+1}$.

Lemma 2.8.6. *Assume that E_6 holds and let v be a non-useful node. Let $k \leq \frac{\ln n}{(\ln \ln n)^2}$ and $c > 0$ be a constant. We have*

$$\mathbb{P}\left[L^k \text{ causes } c \text{ collisions} \mid L^1, \ldots, L^k\right] \leq n^{-3c/5+o(1)}.$$

Proof. In the following all probabilities are conditioned on $L_1, \ldots, L^k$. Since all L^i contain only non-useful nodes, by E_6, we have

$$|L^k| \leq (5m \ln^2 n)^k. \tag{2.8.2}$$

For any two fixed nodes $j, j' \in L^k$ (not necessarily distinct), we have

$$\begin{aligned}
&\mathbb{P}\left[j, j' \text{ cause a collision}\right] \\
&\quad \leq \mathbb{P}\left[(j, j') \in E\right] + \mathbb{P}[\exists j'' \notin L^{\leq k} \colon j'' \text{ not useful} \\
&\qquad\qquad\qquad\qquad\qquad\qquad \wedge (j, j'') \in E \wedge (j', j'') \in E] \\
&\quad \leq n^{-3/5+o(1)} + 5m \ln^2(n)\, n^{-3/5+o(1)} && \text{by (2.8.1)} \\
&\quad = n^{-3/5+o(1)}.
\end{aligned}$$

Similarly, for any fixed node $j \in L^k$, using (2.8.1), we have

$$\begin{aligned}
&\mathbb{P}\left[\exists j' \notin L^{\leq k} \colon j' \text{ not useful } \wedge j \text{ and } j' \text{ are connected by two edges}\right] \\
&\quad \leq 5m \ln^2(n)\, n^{-3/5+o(1)} = n^{-3/5+o(1)}.
\end{aligned}$$

Thus, we have

$$\mathbb{P}\left[L^k \text{ causes } c \text{ collisions}\right] \leq (|L^k|^2 + |L^k|)^c n^{-3c/5+o(1)} = n^{-3c/5+o(1)}. \quad \square$$

We are now ready to prove Lemma 2.8.3.

Proof of Lemma 2.8.3. Let v be a fixed non-useful node. By Lemma 2.8.6, the probability that there exists a single level that causes two collisions is at most $Kn^{-6/5+o(1)} = n^{-6/5+o(1)}$. Similarly, the probability that there exist two levels that cause a collision each is at most $K^2 n^{-6/5+o(1)} = n^{-6/5+o(1)}$. The result now follows from a simple union bound over all non-useful nodes. □

Chapter 3

The Synchronous Protocol

In this chapter, we analyze the synchronous protocol. To see why a single memory slot can lead to an asymptotic speed-up, it is instructional to prove the lower bound first.

3.1 Lower Bound

We prove the following lower bound.

Lemma 3.1.1. *Let s be an arbitrary node in G_m^n. With probability $1 - o(1)$, the synchronous push-pull protocol broadcasts a rumor from s to all nodes in*

- $\Omega(\ln n)$ *rounds, if* $M = 0$,
- $\Omega(\ln n / \ln \ln n)$ *rounds, if* $M \geq 1$.

For $M \geq 1$, the lower bound follows immediately from the fact that G_m^n has a diameter of $\Theta(\ln n / \ln \ln n)$ with high probability. For $M = 0$, the proof strategy is as follows. We first show that with high probability there are $\Omega(n)$ edges with incident nodes both of constant degree. Both nodes of such an edge remain uninformed with constant probability in each round since with constant probability the two nodes contact each other and are not contacted by any other of their neighbors. It is then easy to show that at least for one edge the incident nodes remain uninformed after $\Omega(\ln n)$ rounds with high probability.

Proof of Lemma 3.1.1. Let X_c denote the total degrees of all nodes of indegree at most c for some constant $c > 0$. By Theorem 2.3.1, we have with probability $1 - o(1)$,

$$\begin{aligned}
X_c &\geq (1-\varepsilon)\sum_{d=0}^{c}(d+m)n\alpha_m(d) \\
&= (1-\varepsilon)2m(m+1)n\sum_{d=m}^{c+m}\frac{1}{(d+1)(d+2)} \\
&= (1-\varepsilon)2m(m+1)n\sum_{d=m}^{c+m}\Big(\frac{1}{d+1}-\frac{1}{d+2}\Big) \\
&= (1-\varepsilon)2m(m+1)n\big(\tfrac{1}{m+1}-\tfrac{1}{c+m+2}\big) \\
&= (1-\varepsilon)2mn\big(1-\tfrac{m+1}{c+m+2}\big).
\end{aligned}$$

Note that each edge that connects two nodes each of degree at most c is counted twice in X_c, whereas each edge that connects one node of degree at most c with a node of degree larger than c is counted only once. Hence, $X_c - mn$ is a lower bound on the number of edges that connect two nodes of degree at most c. So with probability $1 - o(1)$, we have at least $(1 - \varepsilon)2mn\big(1 - \frac{m+1}{c+m+2} - \frac{1}{2(1-\varepsilon)}\big)$ such edges. Thus, for sufficiently large constant c, we have $\Omega(n)$ such edges and therefore also $\Omega(n)$ such pairs of nodes each of degree at most $c + m$ that are connected to each other. Moreover, since each of these nodes has constant degree, we can select $\Omega(n)$ pairwise disjoint pairs so that no node appears in more than one selected pair. Let P denote such a pairwise disjoint set of these edges.

Now consider an edge $(u, v) \in P$. Assume that both nodes are uninformed. Then, they remain uninformed after one round if in this round (i) both nodes contact each other and (ii) none of the other neighboring nodes contacts u or v. Note that the first event occurs with probability at least $\big(\frac{1}{c+m}\big)^2$, and the second event occurs with probability at least $(\frac{1}{m+1})^{2(m+c)}$ since in the worst case there could be m multi-edges between a neighboring node w and u or v and only one edge from w to a different node. Since these two events are independent, the probability that u and v remain uninformed after one round is at least

$$\big(\tfrac{1}{c+m}\big)^2(\tfrac{1}{m+1})^{2(m+c)} =: \delta.$$

There is a small technicality here. One of the neighbors w could be connected only to these nodes (e.g., via a multi-edge), so event (ii) would actually be impossible. However, in that case w can only be informed if u or v is informed.

Note that $\delta \in \Omega(1)$. The probability that $e \in P$ remains uninformed after $\alpha \ln n$ rounds for $\alpha = 1/(2\ln\delta^{-1})$ is therefore at least $n^{-1/2}$. Let I_e denote the event that in one of the $\alpha \ln n$ rounds, one of the events (i) and (ii) does not occur. So $\mathbb{P}[I_e] \leq 1 - n^{-1/2}$. Note that two edges in P

might have a common node incident to them. Therefore, we can not assume that the events $(I_e)_{e \in P}$ are independent. However, since the edges in P are pairwise disjoint, the events (i) for different edges in P are independent, and furthermore, conditioning on the event (ii) for an edge in P can only reduce the probability of event (ii) for another edge. Thus, the events $(I_e)_{e \in P}$ are *negatively correlated*, i.e., for any subset $P' \subseteq P$ and any $e \in P \setminus P'$, $\mathbb{P}[I_e \mid \bigwedge_{f \in P'} I_f] \le \mathbb{P}[I_e]$. Let $e_1, \ldots, e_{|P|}$ be an arbitrary ordering of the edges in P. Then, the probability that none of the $\Omega(n)$ pairs remains uninformed after $\alpha \ln n$ rounds is at most

$$\begin{aligned}
\mathbb{P}[\bigwedge_{e \in P} I_e] &= \mathbb{P}[I_{e_1}] \cdot \mathbb{P}[I_{e_2} \mid I_{e_1}] \cdot \mathbb{P}[I_{e_3} \mid I_{e_1} \wedge I_{e_2}] \cdots \mathbb{P}[I_{e_{|P|}} \mid I_{e_1} \wedge \cdots \wedge I_{e_{|P|-1}}] \\
&\le \mathbb{P}[I_{e_1}] \cdot \mathbb{P}[I_{e_2}] \cdots \mathbb{P}[I_{e_{|P|}}] \\
&= (1 - n^{-1/2})^{\Omega(n)} \\
&\le e^{-\Omega(n^{1/2})}.
\end{aligned}$$

□

Note that this proof fails when nodes do not contact the same neighbor twice in a row. For a similar argument to work in that case, one would need to show that there exists a polynomial number of triangles that consist of small degree nodes. In Lemma 2.8.2, we prove that this is not the case.

3.2 Upper Bound

We now prove an upper bound on the runtime of the push-pull protocol that matches our previous lower bound (up to constant factors).

Lemma 3.2.1. *Let s be an arbitrary node in G_m^n. With probability $1 - o(1)$, the synchronous push-pull protocol broadcasts a rumor from s to all nodes in*

- *$O(\ln n)$ rounds, if $M = 0$,*
- *$O(\ln n / \ln \ln n)$ rounds, if $M \ge 1$.*

3.2.1 Informing the First Useful Node

Let $G = G_m(W_1, \ldots, W_n)$ be a typical social network. Assume that also E_6, E_7, and E_8 hold. In this section, all probabilities are taken over the product space of the random graph G and the random decisions of the rumor spreading process.

We show that with high probability, the rumor reaches a useful node within $O(\ln n)$ steps for $M = 0$ and $O(\ln^{3/4}(n) \ln \ln n)$ time steps for $M \ge 1$.

We make use of the following simple observation that follows directly from the definition of preferential attachment graphs.

Fact 3.2.2. *With probability 1, both neighbors of a node of degree 2 have degree at least 3. Thus, every node either has at least three neighbors or is connected to a node with at least three neighbors.*

Lemma 3.2.3. *In the synchronous push-pull protocol with $M = 0$, the rumor initiated by any node reaches a useful node in $O(\ln n)$ rounds with probability $1 - o(n^{-2})$.*

Proof. We make some assumptions that simplify the analysis while only slowing down the process. In this *delayed* process, we first assume that all nodes perform only push operations. Second, we set some nodes to be *inactive*, i.e., they stop informing other nodes until they are activated again. Note that when a node is deactivated, it does not lose its memory on the node it contacted the last.

We consider *phases* of several rounds. Let D_k denote the set of active nodes in phase k and $n_k = |D_k|$. Let $K = C \ln\ln n$ for a sufficiently large constant C. By E_8, there is at most one cycle in $D_1 \cup \cdots \cup D_K$. For simplicity, we first assume that there is no cycle in $D_1 \cup \cdots \cup D_K$. It follows that all nodes in D_k have at most one informed neighbor in D_{k-1}.

We distinguish two stages. In the first stage, we do not fix the length of each phase: a phase will last for a constant number of rounds in expectation. Active nodes of degree 2 remain active until they contact one uninformed neighbor; those of higher degree remain active until they contact two uninformed neighbors. A phase lasts until all active nodes stop. Then, a new phase starts in which all nodes become active that have been informed in the previous phase. In the first phase, only the initially informed node u is active. The first stage ends at the earliest phase K' such that $n_{K'} \geq C \ln n$ nodes. By Fact 3.2.2, both neighbors of a node of degree 2 must have degree at least 3. Thus, we have $n_{k+2} \geq 2n_k$. It follows that $K' = O(\ln\ln n)$ and $N := \sum_{1 \leq k \leq K'} n_k \leq 4C \ln n$.

We now bound the length of the first stage. Let X_i denote the number of rounds needed until an active node i contacts one or two uninformed neighbors depending on whether it has degree 2 or more. We can then bound from above the length the first stage by $X = \sum_{1 \leq k < K'} \sum_{i \in D_k} X_i$. Note that for all i, X_i is stochastically dominated by a geometric variable Y_i with parameter $\frac{1}{2}$ in case i has degree 2 and the sum of two geometric random variables $Y_{i,1} + Y_{i,2}$ each with parameter $\frac{1}{3}$ in case i has degree larger than 2. It follows that X is stochastically dominated by $Y = \sum_{1 \leq i \leq 8C \ln n} Y_i$ where Y_i is a geometric random variable with parameter $p = \frac{1}{3}$. By Lemma 1.2.5, we have

$$\begin{aligned}\mathbb{P}[Y > 48C \ln n] &= \mathbb{P}[Y > (1+1)p^{-1} 8C \ln n] \\ &\leq \exp(-\tfrac{1}{2} C \ln n) \\ &\leq n^{-2C}.\end{aligned}$$

We conclude that the first stage lasts for at most $48C \ln n$ iterations with probability $1 - O(n^{-C})$.

In the second stage, each phase will last for exactly five rounds. The probability that an active node of degree 2, contacts an uninformed node within five rounds is $1 - 2^{-5} = \frac{31}{32}$ and similarly, for an active node of degree at least 3, the probability that it contacts two uninformed nodes is at least $1 - 3(\frac{2}{3})^5 \geq 0.6$. Assume that we are given $D_1, \dots, D_k$ and $n_k \geq C \ln n$. Let $i \in D_k$. If node i has degree at least 3, then let X_i be the indicator variable for the event that it contacts two uninformed nodes in D_{k+1} each of which in turn contacts one uninformed node in D_{k+2}. By the previous discussion, we have $\mathbb{P}[X_i = 1] \geq 0.6\big(\frac{31}{32}\big)^2 \geq 0.563$. Similarly, if node i has degree 2, let X_i be the indicator variable for the event that it contacts one node (of degree at least 3) in D_{k+1} which in turn contacts two nodes in D_{k+2}. Here, we have $\mathbb{P}[X_i = 1] \geq \frac{31}{32} 0.6 \geq 0.581$. Let $X = \sum_{i \in D_k} X_i$. Note that all X_i are independent from each other and $n_{k+2} \geq 2X$. Thus, we have $\mathbb{E}[n_{k+2} \mid n_k] \geq 1.16 n_k$. By Chernoff's bound, we further get

$$\mathbb{P}[X \leq 1.1 n_k] \leq \exp\big(-\Omega(n_k)\big) \leq n^{-3}, \tag{3.2.1}$$

where the last inequality follows from $n_k \geq C \ln n$ by choosing C sufficiently large, but constant. Thus, by a union bound over all $k \leq K$, with probability $1 - o(n^{-2})$, we have $n_K \geq \Omega(1.1^K) \geq \Omega((\ln n)^{10})$ where the last inequality follows from $K = C \ln \ln n$ by choosing C sufficiently large. Although so far we have ignored the possibility of encountering a cycle, it is easy to check that a single cycle does not affect this bound. Thus, by the same argument as for Lemma 2.8.5, once phase $K = C \ln \ln n$ is active, either a useful node was already informed or the probability that a useful node is informed in one round is $1o(n^{-2})$. □

We now come to the case $M = 1$.

Lemma 3.2.4. *Let u be any node. In the synchronous push-pull protocol with $M \geq 1$, the probability p_u that the rumor initiated by u does not reach any useful node in $O(\ln^{3/4}(n) \ln \ln n)$ rounds, satisfies $p_u = o(1)$. Moreover, the sum $\sum_{u \in V} p_u$ of all these failure probabilities is also $o(1)$.*

Proof of Lemma 3.2.4. Let u be a node that initiates the rumor. Assume that u is useful, as otherwise we are finished. Let C be a sufficiently large constant. By E_8, there is at most one cycle of non-useful nodes that are connected to u via paths of length at most K, where $K = \frac{\ln n}{(\ln \ln n)^2}$. Assume first that there is no such cycle. Then during the first K rounds, we inform a useful node, or due to the absence of non-useful cycles, we have the property that whenever a node becomes informed all its neighbors except one are still uninformed.

As before, we analyze a delayed process where only active nodes perform operations in each round.

We consider four stages. The first three stages last for $O(\sqrt{\ln n})$ rounds each. At the end of the third stage we will have informed $\Omega(\ln n)$ nodes with high probability. The fourth stage lasts for $O(\ln \ln n)$ rounds and informs a useful node with high probability. Until the last round of the third stage, we always assume that we have not encountered a useful node so far.

The first stage. In the first $2\sqrt{C \ln n}$ rounds, we run the following delayed process. In the first round, only u is active and it informs one new node. In every subsequent round, the last node that was most recently informed becomes the only active node. Since a node never contacts the same node twice in a row, every active node informs an uninformed node in at most two rounds. Thus after $2\sqrt{C \ln n}$ rounds, with probability 1, we have a path of at least $\sqrt{C \ln n}$ informed nodes (and at most $2\sqrt{C \ln n} + 1$ informed nodes).

The second stage. This stage again lasts for $2\sqrt{C \ln n}$ rounds. Let I be the path of informed nodes from the first stage. In this stage the active nodes are all nodes in I that have degree at least 3. By Fact 3.2.2, we have at least $|I|/2$ active nodes. In contrast to the first stage, most of these active nodes have two informed neighbors. Therefore, it is not true anymore that in every two rounds an active node contacts at least one uninformed node. However, each active node that has not contacted an uninformed node in this stage yet will do so in each round with probability at least $\frac{1}{2}$. Thus, the probability that an active node fails to contact an uninformed node in this stage is at most $2^{-2\sqrt{C \ln n}}$. Let J denote all nodes that become informed in this stage. We have

$$\begin{aligned} \mathbb{P}[|J| \leq |I|/4] &\leq (|I|/2)^{|I|/4} \, 2^{-2\sqrt{C \ln n}\,|I|/4} \\ &\leq (\sqrt{C \ln n})^{\sqrt{C \ln n}} \, 2^{-C \ln(n)/2} \\ &\leq n^{-0.34\,C}. \end{aligned}$$

Thus, with probability $1 - n^{-0.34C}$, we have at least $\sqrt{C \ln n}/4$ nodes that were informed during the second stage. For the next stage we assume that this is the case.

The third stage. This stage lasts for $8\sqrt{C \ln n}$ rounds. The active nodes are all nodes that were informed in the second stage. Similar to the first stage, in every subsequent round, every active node becomes inactive once it has contacted an uninformed node which in turn becomes active. Since it takes an active node at most two rounds until it has contacted an uninformed node, after $8\sqrt{C \ln n}$ rounds, with probability 1, we have at least $\sqrt{C \ln n}/4$ disjoint paths, each with at least $4\sqrt{C \ln n}$ informed nodes. Hence, in total we have $C \ln n$ informed nodes in this stage.

The fourth stage. Let L denote all nodes that were informed in the third stage and have a degree of at least 3. By Fact 3.2.2, we have $|L| \geq C \ln(n)/2$. We now activate all nodes in L for a single round. Let D_1 denote the set of all nodes that were informed by the nodes in L in one round. Note that all nodes in L have at most two informed neighbors, since otherwise there must be a cycle. Thus, similar to the second stage, the probability that a node in L contacts an uninformed node in one round is at least $\frac{1}{2}$ (it cannot contact the node it informed the last). Hence, $\mathbb{E}[|D_1|] \geq \frac{1}{2}|L| \geq C\ln(n)/4$. By Chernoff's bound, we have

$$\mathbb{P}\left[|D_1| \leq C\ln(n)/6\right] \leq n^{-\frac{C}{72}}. \tag{3.2.2}$$

We now consider *phases* of three rounds. In the first phase only nodes in D_1 are active. For $k \geq 2$, we define the set D_k of nodes active in phase k as follows. Given $D_1, \ldots, D_{k-1}$, let D_k denote all nodes that become informed by nodes in D_{k-1} (i.e., in phase $k-1$). Let $n_k = |D_k|$ and $M = C\ln\ln n$. Note that at the beginning of phase k, all nodes in D_k have exactly one informed neighbor. Thus, the probability that an active node of degree at least 3 contacts two uninformed nodes in three rounds is easily seen to be at least $\frac{2}{3}$. Assume that we are given $D_1, \ldots, D_k$ and $n_k \geq C\ln(n)/6$. Let $i \in D_k$. If node i has degree at least 3, then let X_i be the indicator variable for the event that it contacts two distinct nodes in D_{k+1} which in turn contact one node in D_{k+2}, respectively. Similarly, if node i has degree 2, let X_i be the indicator variable for the event that it contacts one node (of degree at least 3) in D_{k+1} which in turn contacts two nodes in D_{k+2}. In both cases, we have $\mathbb{P}[X_i = 1] \geq \frac{2}{3}$. Let $X^k = \sum_{i \in D_k} X_i$. Note that the X_i are mutually independent and $n_{k+2} \geq 2X^k$. Thus, we have $\mathbb{E}[n_{k+2} \mid n_k] \geq \frac{4}{3}n_k$. By Chernoff's bound, we further obtain

$$\mathbb{P}\left[n_{k+2} \leq \tfrac{5}{4}n_k\right] \leq \exp\left(-\tfrac{1}{384}n_k\right) \leq n^{-C/2304}, \tag{3.2.3}$$

where the last inequality follows from $n_k \geq C\ln(n)/6$. Thus, with probability at least $1 - Mn^{-C/2304} \geq 1 - n^{-C/2305}$, we have $n_M \geq \Omega\big(\big(\frac{5}{4}\big)^{M/2}\big) \geq \Omega(\ln^7 n)$, for sufficiently large (but constant) C.

For every $i \in D_M$, at most one of the nodes $\ell_{i,1}$ and $\ell_{i,2}$ are informed. Let ℓ_i denote any of the nodes $\ell_{i,1}$ and $\ell_{i,2}$ that is not informed. Given the sequence $D_1, \ldots, D_M$, the nodes ℓ_i, where $i \in D_M$, are mutually independent. Furthermore, conditioned on $\ell_i \notin D_1, \ldots, D_M$, the probability that ℓ_i is useful can only increase since $D_1, \ldots, D_M$ only contain non-useful nodes. Hence by Lemma 2.8.5, for any $i \in D_M$, we have $\mathbb{P}[\ell_i \text{ is useful}] \geq \ln^{-3} n$. Furthermore, for any $i \in D_M$, the probability that i contacts ℓ_i in one round is at least $1/\deg(i) \geq 1/(5m\ \ln^2 n)$ due to E_6. Since for each $i \in D_m$ both events are independent, the probability that no node in D_M informs a useful node in one round is at most $(1 - 1/(5m\ \ln^5 n))^{n_M} \leq \exp(-n_M/(5m\ \ln^5 n)) \leq n^{-\Omega(\ln n)}$.

The total failure probability of all four stages is at most $n^{-0.34C} + n^{-C/72} + n^{-C/2305} + n^{-\Omega(\ln n)} = o(n^{-2})$ for sufficiently large (but constant) C. Consequently, by a simple union bound over all nodes, we conclude that the total probability that a rumor started by any node u reaches a useful node in $O(\sqrt{\ln n})$ rounds is $1 - o(n^{-1})$.

So far we have assumed that u is not connected to a non-useful cycle via paths of length K. By E_8, there can be at most one such cycle. We now argue how the above argument can be easily adjusted for that case. First we consider the case when u is not itself part of the cycle. Clearly, if throughout the stages no cycle is detected, then the above argument still applies. Otherwise, if the cycle is encountered in the first stage, then after the first $2\sqrt{C \ln n}$ rounds, we again make u the only active node and run the same delayed process for another $2\sqrt{C \ln n}$ rounds. Thus the first stage now lasts for $4\sqrt{C \ln n}$ rounds. Since u does not contact the last contacted node again and there can be no other cycle, we obtain a new path of $\sqrt{C \ln n}$ informed nodes that is disjoint from the nodes informed in the first $2\sqrt{C \ln n}$ rounds. If the cycle is encountered in the second stage, at most two active nodes are affected. Thus, at the end of the second stage we will still have at least $\sqrt{C \ln n}/4 - 2$ newly informed nodes which is sufficient for the subsequent stages. Similarly, if the cycle is encountered in the third stage, at most two paths are affected and so at the end of the third stage, we will still have at least $C \ln n - 8\sqrt{C \ln n}$ informed nodes. By slightly adjusting the constant C, it is easy to see that this is sufficient for the subsequent stages. If the cycle is encountered in the fourth stage, we have $n_{k+2} \geq 2X^k - 2$ for all $k \geq 1$. Again, the computation can be easily adjusted for this case.

It remains to consider the case when u lies in a cycle of length at most K. Our goal is to bound the number of rounds needed until we reach a node that lies outside the cycle. Since by E_8 there can be only one cycle, this node itself cannot lie in cycle and we can apply the previous case again. W.l.o.g. assume that u has degree at least 3 (otherwise we take one of the neighbors). In each round, the probability that u contacts a node outside of the cycle is at least $\frac{1}{3}$. Thus the probability that u does so within $C \ln^{3/4}(n) \ln\ln n$ rounds is $1 - 3^{-C \ln^{3/4}(n) \ln\ln n}$. By E_7, there are $(\ln n)^{O(\ln^{3/4} n)} = 2^{O(\ln^{3/4}(n) \ln\ln n)}$ cycles of length $O(\sqrt{\ln n})$. Thus, for sufficiently large C, we obtain, by a simple union bound, that with probability $1 - o(1)$, all nodes that lie in such cycles reach a node that is outside of the cycle within $O(\ln^{3/4}(n) \ln\ln n)$ rounds. By the previous case, it then takes another $O(\sqrt{\ln n})$ rounds until a useful node is reached. □

3.2.2 Informing Node 1

What ultimately makes rumor spreading in preferential attachment graphs fast, are vertices of small (constant) degree. Each of them, with constant probability, has the beautiful property that, once a neighbor becomes informed, it pulls the rumor from such a neighbor and pushes it to all other neighbors in a total number of time steps equal to its degree. As we will see in this section, this property alone suffices to spread the rumor among all useful nodes.

More specifically, we show that between any two useful nodes there is a path of length $O(\ln n/\ln\ln n)$ such that every second node on the path has this property. Since these nodes (by definition with probability one) propagate the rumor in constant time, we see that the rumor is propagated along such a path in time $O(\ln n/\ln\ln n)$.

Consider a fixed graph G and a run of the rumor spreading process started in some node u. Let $v \neq u$ be a node of degree exactly m. Let t be the first time when some neighbor of v is informed and v' be the smallest neighbor of v that is informed at time t. We say that v is *fast* (in this run of the process) if the following is true. (i) In time step $t+1$, v contacts v' and thus pulls the rumor from v'. (ii) In time steps $t+2,\dots,t+m$, v contacts all other neighbors without repetition and thus informs all these neighbors.

The event that a node v is fast, is independent from the random decisions of all other nodes in the process. For this reason, the following lemma is the key to our analysis (to be continued with Corollary 3.2.9).

Lemma 3.2.5. *Let $G = G_m(W_1,\dots,W_n)$ be a typical social network. Let $p \in [0,1]$ be a constant. For each node $v \in \left[\frac{2}{3}n, n\right]$ of degree m decide independently with probability p that it is marked.*

Let $u \in [n]$ be a useful node. Then, with probability $1-o(n^{-1})$ in the product space of random graph and random marks, there exists a path of length $O(\ln n/\ln\ln n)$ between u and 1 such that every second node is marked.

We start by showing that with high probability, the random graph regarded contains a linear number of marked nodes. Of course, the main ingredient for this statement is the fact that there is a linear number of nodes $i \in \left[\frac{2}{3}n, n\right]$ that have a degree equal to m. If not explicitly stated, all probabilities in this section are taken over the product space of the typical social network $G_m(W_1,\dots,W_n)$ and the random marks.

Lemma 3.2.6. *Let $\varepsilon_m := \frac{1}{8}pe^{-3m}$. With probability $1-e^{-\Omega(n)}$, there are at least $\varepsilon_m n$ marked nodes.*

Proof. Since $\sum_{i=1}^n w_i = 1$, at least half of the $i \in \left[\frac{2}{3}n, n\right] := C$ have $w_i \leq 6/n$. Let $i \in C$ be such that $w_i \leq 6/n$. Note that i has degree equal to m if and only if no node $j > i$ is a neighbor of i. Even conditioning arbitrarily on the degrees of all nodes in $C \setminus \{i\}$, we have for all $k \in \{1,\dots,m\}$, $\mathbb{P}[\ell_{j,k} = i] \leq w_i/(W_j - \sum_{r=\frac{2}{3}n}^{j} w_r) \leq (6/n)/W_{\frac{2}{3}n} \leq (6/n)/(0.9\cdot\sqrt{2/3}) \leq 9/n$, using the lower bound on $W_{\frac{2}{3}n}$ from property E_1. Consequently, the degree of i equals m with probability at least $(1-9/n)^{(1/3)nm} \geq (1-o(1))\exp(-\frac{9nm}{3n}) = (1-o(1))e^{-3m}$.

Thus, the expected number of nodes in C having degree m is at least $(1-o(1))\frac{1}{6}e^{-3m}n$. Since we allowed arbitrary conditioning on other degrees in C, we may apply Chernoff bounds and see that with probability $1-e^{-\Omega(n)}$, at least $\frac{1}{7}e^{-3m}n$ of the nodes in C have degree equal to m.

Each of these nodes was marked independently with probability p. Hence the expected number of marked nodes is at least $\frac{1}{7}e^{-3m}pn$, and with probability $1-e^{-\Omega(n)}$, at least $\frac{1}{8}e^{-3m}pn$ of the nodes in C have degree equal to m and are marked. □

We now construct a path from a useful node u to node 1 that has each second node marked. We say a node i is *good* if

$$i \in [s+1, 2^b] \text{ and } w_i \geq \frac{1}{10\sqrt{in}}, \tag{3.2.4}$$

where, as before, $s = 2^a$ is the smallest power of 2 larger than $\ln^7 n$ and 2^b is the largest power of 2 smaller than $\frac{2}{3}n$. We consider sets Γ_k and Γ'_k defined recursively as follows. We set $\Gamma_0 = \{u\}$. Given Γ_k, Γ'_k is defined to be the set of all *marked* nodes $i \geq \frac{2}{3}n$ that have a neighbor in Γ_k and have not been included in any Γ'_ℓ with $\ell \leq k-1$. Similarly, Γ_k is defined as the set of all *good* nodes that have a neighbor in Γ'_{k-1} and have not been included in any Γ_ℓ with $\ell \leq k-1$. Note that for all $k \geq 0$, Γ_k only contains nodes $i < \frac{2}{3}n$, while Γ'_k only contains nodes $i \geq \frac{2}{3}n$. This is true for Γ_0 since u is useful and by E_5, all useful nodes are smaller than $n/2$. We define the *weight* of a set Γ_k by

$$f_k := \begin{cases} w_u & \text{if } k = 0 \\ \sum_{i\in\Gamma_k} \frac{1}{\sqrt{in}} & \text{if } k \geq 1. \end{cases} \tag{3.2.5}$$

Since for $k \geq 1$, Γ_k only contains good nodes, and by definition, $w_u = f_0$, we have for $k \geq 0$,

$$\sum_{i\in\Gamma_k} w_i \geq f_k/10. \tag{3.2.6}$$

Let $N_k = \bigcup_{0\leq i\leq k} \Gamma_i$ and $N'_k = \bigcup_{0\leq i\leq k} \Gamma'_i$. We denote by $C_0 \subseteq \left[\frac{2}{3}n, n\right]$ the set of marked nodes and for $k \geq 1$, by $C_k = C_0 \setminus N'_{k-1}$ the set of marked nodes excluding nodes in $\Gamma'_0, \Gamma'_1, \ldots, \Gamma'_{k-1}$. By Lemma 3.2.6, we have $C_0 \geq \varepsilon_m n$ with probability $1-e^{-\Omega(n)}$. We also need the following technical lemma.

Lemma 3.2.7 (Bollobás and Riordan [10]). *Let $\varepsilon > 0$, and $K := (1/2+\varepsilon)(\ln(n)/\ln\ln(n)) - 1$. Let $f_0, f_1, \ldots$ be a sequence of real numbers with $f_0 \geq \ln^2(n)/n$ and*

$$f_{k+1} \geq \min\{2\log_2(\varepsilon_m f_k n/\ln n) - 29, b-a\}\varepsilon_m f_k/3564 \tag{3.2.7}$$

for all $k \geq 0$. Then, for n sufficiently large, $\ell = \min\{k\colon f_k \geq \ln^3(n)/\sqrt{n}\}$ exists and is at most K.

Note that in the original paper the authors assume

$$f_{k+1} \ge \min\{2\log_2(f_k n/\ln n) - 32, b - a - 1\} f_k/1000$$

and obtain that $\ell = \min\{k\colon f_k \ge \ln^2(n)/\sqrt{n}\}$ is at most K. It is easy to check that essentially the same proof holds for the above version. For completeness, we provide the proof here.

Proof. Provided n is sufficiently large we have

$$\log_2(\varepsilon_m f_0 n/\ln n) \ge \log_2(\varepsilon_m \ln n) \ge 4015/\varepsilon_m.$$

Thus (3.2.7) implies that $f_1 2f_0$. It follows inductively that $f_{k+1} \ge 2f_k$ and $f_k \ge 2^k f_0$ hold for all $k \ge 0$. This shows that ℓ exists. As $ba \ge \ln n 8 \ln\ln n$, the minimum in (3.2.7) is different from the first term only if $f_k \ge (\ln n)^{-3}/\sqrt{n}$. If this first happens at $k = k_0$, say, then for $k \ge k_0$ we have

$$f_{k+1} \ge (ba)\varepsilon_m f_k/3564 = (\log_2 n)^{1-o(1)} f_k,$$

which implies that $\ell \le k_0 + 7$. Thus

$$f_{k+1} \ge \frac{\log_2(\varepsilon_m f_k n/\ln n)15}{1782} f_k, \tag{3.2.8}$$

for $0 \le k < \ell 6$. As $f_k \ge 2^k f_0$ and $\log_2(\varepsilon_m f_0 n/\ln n) \ge 16$ (if n is large enough), we have $\log_2(\varepsilon_m f_k n/\ln n) \ge k + 16$. Combined with (3.2.8) this implies that $f_{k+1} \ge f_k(k+1)/1782$ for $0 \le k < \ell 6$, and hence that

$$f_{\ell-6} \ge \frac{(\ell 6)!}{1782^{\ell 6}} f_0 \ge \left(\frac{\ell - 6}{1782e}\right)^{\ell-6} f_0,$$

using StirlingâĂŹs formula. As $f_{\ell 6} < (\ln n)^3/\sqrt{n} \le \sqrt{n} f_0$ this implies that $\ell 6 \le (1/2 + \varepsilon/2)\ln n/\ln\ln n < K6$, and the lemma follows. □

Remember that $I_t := [2^t + 1, 2^{t+1}]$ for $t \in [a, b)$.

Lemma 3.2.8. *Let $k \ge 0$ be such that $f_k \ge \ln^2(n)/n$ and $|C_k| \ge \varepsilon_m n/2$. Then given C_k and $\Gamma_0, \Gamma'_0, \Gamma_1, \Gamma'_1, \ldots, \Gamma_k$, with a probability of at least $1 - O(n^{-6/5})$, one of the following is satisfied:*

- *$|N_{k+1} \cap I_t| \ge 2^{t-2}$, for some $t \in [a, b)$, or*
- *$f_{k+1} \ge \min\{2\log_2(\varepsilon_m f_k n/\ln n) - 29, b - a\}\varepsilon_m f_k/3564$.*

Proof. All probabilities are conditioned on the assumptions in the lemma.

We first show that $|\Gamma'_k| = \Omega(nf_k)$ holds with high probability. Let $j \in C_k$. By definition, node j can not be connected to any node in N_{k-1} or C_0 via its first link (i.e., $\ell_{j,1} \notin N_{k-1} \cup C_0$). Thus, we have

$$\begin{aligned}\mathbb{P}[\ell_{j,1} \in \Gamma_k] &= \Big(\sum_{i\in\Gamma_k} w_i\Big)/\Big(W_j - \sum_{i\in N_{k-1}} w_i - \sum_{i\colon i\in C_0,\, i\le j} w_i\Big)\\ &\ge \Big(\sum_{i\in\Gamma_k} w_i\Big)/W_j \ge f_k/10,\end{aligned}$$

where the last inequality follows from (3.2.6). Hence,

$$\mathbb{E}[|\Gamma'_k|] \ge |C_k| f_k/10 \ge \varepsilon_m n f_k/20.$$

By Chernoff's bound, we obtain

$$\mathbb{P}[|\Gamma'_k| \le \varepsilon_m n f_k/21] \le \exp(-\Omega(\varepsilon_m n f_k)) \le n^{-\Omega(\ln n)}, \tag{3.2.9}$$

where the last inequality follows from $f_k \ge \ln^2(n)/n$. In the following, we assume

$$|\Gamma'_k| \ge \varepsilon_m n f_k/21. \tag{3.2.10}$$

We now show that either $|N_{k+1} \cap I_t| \ge 2^{t-2}$ for some $t \in [a,b)$, or with high probability, for sufficiently many $t \in [a,b)$, we have $|\Gamma_{k+1} \cap I_t| = \Omega\big(|\Gamma'_k|\sqrt{2^t/n}\big)$. Let $t \in [a,b)$. By E_2, I_t contains at least 2^{t-1} good nodes. Let S be initially the set of good nodes in $I_t \setminus N_k$. So $|S| \ge 2^{t-1} - |N_k \cap I_t|$. We consider the elements of Γ'_k one by one in any order. Let $\widetilde{\Gamma}$ be an initially empty set. Whenever we encounter some node $i \in \Gamma'_k$ that is connected to some node $j \in S$ via its second link (i.e., $\ell_{i,2} = j$), we remove j from S and include it into $\widetilde{\Gamma}$. Note that $\widetilde{\Gamma} \subseteq \Gamma_{k+1}$ throughout this process. Moreover, $\ell_{i,2}$ has not been revealed before i is considered ($\ell_{i,1}$ is independent from $\ell_{i,2}$ given all W_j values). Hence, as long as $|S| \ge 2^{t-2}$, we have for every node $i \in \Gamma'_k$,

$$\mathbb{P}[\ell_{i,2} \in S] \ge \frac{2^{t-2}}{10W_i\sqrt{2^{t+1}n}} \ge \frac{\sqrt{2^t}}{60\sqrt{n}}.$$

If $|S| < 2^{t-2}$ at some point, then $|N_{k+1} \cap I_t| \ge 2^{t-2}$ and we are finished. Otherwise

$$\mathbb{E}[|\Gamma_{k+1} \cap I_t|] \ge \mathbb{E}[|\widetilde{\Gamma}|] \ge |\Gamma'_k|\frac{\sqrt{2^t}}{60\sqrt{n}} =: \mu_t. \tag{3.2.11}$$

Let $Y = \sum_{1\le i\le|\Gamma'_k|} Y_i$ where $Y_1, \dots, Y_{|\Gamma'_k|}$ are mutually independent 0/1-random variables with $\mathbb{P}[Y_i = 1] = \frac{\sqrt{2^t}}{60\sqrt{n}}$, where $1 \le i \le |\Gamma'_k|$. Note that $|\Gamma_{k+1} \cap I_t|$ stochastically dominates Y, i.e., $\mathbb{P}[|\Gamma_{k+1} \cap I_t| > t] \ge \mathbb{P}[Y > t]$. Now, if $\mu_t \ge 10\ln n$, we have by Chernoff's bound $\mathbb{P}[Y \le \mu_t/2] \le n^{-5/4}$, and thus $\mathbb{P}[|\Gamma_{k+1} \cap I_t| \le \mu_t/2] \le \mathbb{P}[Y \le \mu_t/2] \le n^{-5/4}$. Taking a union bound over all indices $t \in [a, b)$ with $\mu_t \geq 10 \ln n$, we conclude that with probability at least $1 - \log_2(n) \cdot n^{-5/4} = 1 - O(n^{-6/5})$, we have for all these $t \in [a, b)$,

$$\sum_{i \in \Gamma_{k+1} \cap I_t} \frac{1}{\sqrt{in}} \geq \frac{\mu_t}{2} \cdot \frac{1}{\sqrt{2^{t+1} n}} = \frac{|\Gamma'_k|}{120 n \sqrt{2}} \geq \frac{\varepsilon_m f_k}{3564},$$

where the last inequality follows from (3.2.10). Let $\mathcal{T}$ be the set of indices $t \in [a, b)$ with $\mu_t \geq 10 \ln n$. Since $2^b \geq n/3$, we have

$$|\mathcal{T}| \geq \min\{2 \log_2(\varepsilon_m f_k n / \ln n) - 29, b - a\}.$$

Therefore $f_{k+1} \geq \min\{2 \log_2(\varepsilon_m f_k n / \ln n) - 29, b - a\} \varepsilon_m f_k / 3564$. The total failure probability is at most $n^{-\Omega(\ln n)} + O(n^{-6/5}) = O(n^{-6/5})$ which finishes the proof. □

We can now show our key lemma that there exists a 'fast' path between a useful node u and node 1 with high probability.

Proof of Lemma 3.2.5. The idea is to apply Lemma 3.2.8 consecutively for $k = 0, \ldots, K$, where $K = (1/2 + \varepsilon) \ln n / \ln \ln n$. The probability that the event considered in Lemma 3.2.8 holds for all $k = 0, \ldots, K$ is at least $1 - O(K\, n^{-6/5}) = 1 - O(n^{-7/6})$. In the following we assume that this is the case.

Note that $f_0 = w_u \geq \ln^2(n)/n$ since u is useful. Also, by Lemma 3.2.6, we have $|C_0| \geq \varepsilon_m n$ with probability $1 - e^{-\Omega(n)}$. Assume that this is the case. Hence, we can apply Lemma 3.2.8 for $k = 0$. Since $\min\{2 \log_2(\varepsilon_m f_k n / \ln n) - 29, b - a\} \varepsilon_m f_k / 3564 \geq f_k$ (for large enough n), the only way we fail to apply Lemma 3.2.8 for some k', where $0 < k' < K$, is when $|C_{k'}| < \varepsilon_m n/2$ or $|N_{k'} \cap I_t| \geq 2^{t-2}$ for some $t \in [a, b)$.

If $|C_{k'}| < \varepsilon_m n/2$, then there must be a k'', $0 \leq k'' \leq k'$, with $|\Gamma'_{k''}| \geq \frac{\varepsilon_m n/2}{K} \geq \frac{n}{\ln n}$ for n sufficiently large. We stop the sequence at $\Gamma'_{k''}$ as soon as we encounter such a k''. Given the sequence $\Gamma_0, \Gamma'_0, \Gamma_1, \ldots, \Gamma'_{k''}$, the second links of the nodes in $\Gamma'_{k''}$ are mutually independent random variables. So the probability that no node in $\Gamma'_{k''}$ connects to 1 via its second link is at most

$$(1 - w_1)^{|\Gamma'_{k''}|} \leq \left(1 - \tfrac{4}{\ln(n)\sqrt{n}}\right)^{n/\ln n} \leq \exp\left(-\tfrac{4\sqrt{n}}{\ln^2 n}\right) = n^{-\Omega(\sqrt{n}/\ln^3 n)},$$

where the first inequality follows from E_3. Thus, we can assume that $|C_k| \geq \varepsilon_m n/2$ for all $k = 0, \ldots, K - 1$.

Similarly, if $|N_{k'} \cap I_t| \geq 2^{t-2}$ for some $t \in [a, b)$, there must be a k'' where $0 < k'' \leq k'$, with $|\Gamma_{k''} \cap I_t| \geq 2^{t-2}/K$. We stop the construction of the sequence $\Gamma_0, \Gamma'_0, \Gamma_1, \ldots$ at $\Gamma'_{k''}$. By (3.2.9) we have with probability $1 - n^{-\Omega(\ln n)}$,

$$\begin{aligned} |\Gamma'_{k''}| &\geq \varepsilon_m n f_{k''}/21 \geq \varepsilon_m n |\Gamma_{k''} \cap I_t| / (21\sqrt{2^{t+1} n}) \\ &\geq \varepsilon_m n 2^{t-2} / (21 K \sqrt{2^{t+1} n}) \geq \varepsilon_m 2^{t/2} \sqrt{n} / \ln n \geq \varepsilon_m \ln^{5/2}(n) \sqrt{n}, \end{aligned}$$

where the last inequality follows from $2^t \geq \ln^7 n$. So given $\Gamma'_0, \Gamma_1, \ldots, \Gamma'_{k''}$, the probability that no node in $\Gamma'_{k''}$ connects to 1 by its second link is at most

$$(1-w_1)^{|\Gamma'_{k''}|} \leq \big(1 - \tfrac{4}{\ln(n)\sqrt{n}}\big)^{\varepsilon_m \ln^{5/2}(n)\sqrt{n}} \leq \exp\big(-4\varepsilon_m \ln^{3/2} n\big) \leq n^{-\Omega(\sqrt{\ln n})},$$

where the last inequality holds since ε_m is a constant.

So assume now that $|C_k| \geq \varepsilon_m n/2$ and $f_{k+1} \geq \min\{2\log_2(\varepsilon_m f_k n/\ln n) - 29, b-a\}\varepsilon_m f_k/3564 \geq f_k$, for all k, $0 \leq k < K$, where $K = \big(\frac{1}{2} + \varepsilon\big)\frac{\ln n}{\ln\ln n}$. Then, by Lemma 3.2.7, we have $f_\ell \geq \ln^3(n)/\sqrt{n}$, for some $\ell \leq K$. Again, by (3.2.9), we have $|\Gamma'_\ell| \geq \varepsilon_m n f_\ell/21 \geq \varepsilon_m \sqrt{n}\ln^3(n)/21$ with probability $1 - n^{-\Omega(\ln n)}$. Furthermore, given $\Gamma_0, \Gamma'_0, \Gamma_1, \ldots, \Gamma'_\ell$, the probability that no node in Γ'_ℓ connects to 1 by its second link is at most

$$(1-w_1)^{|\Gamma'_\ell|} \leq \big(1 - \tfrac{4}{\ln(n)\sqrt{n}}\big)^{\varepsilon_m \sqrt{n}\ln^3(n)/21} \leq \exp\big(-\tfrac{4\varepsilon_m}{21}\ln^2 n\big) \leq n^{-\Omega(\ln n)}.$$

The total failure probability is $O(n^{-7/6}) + e^{-\Omega(n)} + n^{-\Omega(\sqrt{n}/\ln^3 n)} + n^{-\Omega(\ln n)} + n^{-\Omega(\sqrt{\ln n})} + n^{-\Omega(\ln n)} = O(n^{-7/6})$. □

We can now use Lemma 3.2.5 to show that the rumor quickly proceeds from a useful node to node 1.

Corollary 3.2.9. *Let $G = G_m(W_1, \ldots, W_n)$ be a typical social network. Let $u \in [n]$ be a useful node. With probability $1 - o(n^{-1})$, a rumor present at u reaches node 1 in $O(\ln n/\ln\ln n)$ steps.*

Proof. Consider a run of the process started with a rumor in u. Let $v \in \big[\frac{2}{3}n, n\big] =: C$ be a node of degree m. Note that $u \neq v$, since u is useful and thus $u \leq n/2$ by E_5.

The probability p that v is fast is at least $(m-1)!/m^m$. This remains true if we condition arbitrarily on random decisions of other nodes during the run of the process. In consequence, the set of fast nodes is a random subset of the nodes of degree m in C with each such node being included independently with probability $p \geq (m-1)!/m^m$.

Applying Lemma 3.2.5 with fast nodes being marked, we see that with probability $1 - o(n^{-1})$, there is a path of length $O(\ln n/\ln\ln n)$ such that every second node is fast. Even ignoring all rumor transmissions by nodes that are not fast, the rumor is propagated along the path in $O(\ln n/\ln\ln n)$ time steps. □

3.2.3 Informing All Nodes

The following lemma allows us to invert the spread of the rumor: from node 1 to all other nodes. Note that the lemma has been shown for the simple push-pull protocol in [18].

Lemma 3.2.10. *Assume that if initially node u is informed, then node v is informed in k time steps with probability p. Then also the reverse statement is true, i.e., if initially node v is informed, then u is informed in k time steps with probability p.*

Proof of Lemma 3.2.10. We define a *snapshot* of the process to be a vector $s = (s_1, \ldots, s_n)$ where s_i denotes a neighbor of i. We call a (finite) vector of snapshots $S = (S_1, \ldots, S_\ell)$ a *series.* Note that by interpreting a snapshot as the ordered set of nodes that were contacted in one time step, a series corresponds to a (possibly infeasible) run of the process. Moreover, a series $S = (S_1, \ldots, S_\ell)$ corresponds to a feasible run of the process if and only if for any $i > 1$, $k \in [n]$ and $j \in \{1, \ldots, \min\{\deg(i) - 1, M\}\}$, we have $(S_i)_k \neq (S_{i-j})_k$.

Let $\mathcal{S}$ be the set of all series S of snapshots such that if the rumor is initiated by node u and the process follows S, then node v is informed for the first time after $|S|$ steps. Since v is informed with probability 1 after finitely many steps, $\mathcal{S}$ naturally defines a probability space if we associate with every series $S \in \mathcal{S}$ the probability that the process follows S (infeasible series have probability 0). Note that the probability that node v is informed in k steps is just $\sum_{\substack{S \in \mathcal{S} \\ |S|=k}} \mathbb{P}[S]$.

We give a mapping $\phi \colon \mathcal{S} \to \mathcal{S}$ such that for all $S \in \mathcal{S}$,

(i) $|S| = |\phi(S)|$,

(ii) $\mathbb{P}[S] = \mathbb{P}[\phi(S)]$,

(iii) if the rumor is initiated by node v and the process follows $\phi(S)$, then node u is informed after $|\phi(S)| = |S|$ steps.

It follows that if the rumor is initiated from node v, then the probability that node u is informed in k steps is (at least) $\sum_{\substack{S \in \mathcal{S} \\ |S|=k}} \mathbb{P}[S]$. Equality then follows by a symmetric argument.

The mapping is defined as follows. For $S = (S_1, \ldots, S_\ell)$, let $\phi(S) = (S_\ell, S_{\ell-1}, \ldots, S_1)$, i.e., ϕ simply inverts the series. It remains to check that all three properties are indeed satisfied. Property (i) and (iii) are immediate. For (ii), note that there exists $\rho_1, \rho \in [0, 1]$, such that for all series, $\mathbb{P}[S] = 0$ if S is infeasible, and $\mathbb{P}[S] = \rho_1 \rho^{|S|-1}$ if S is feasible. Since S is feasible if and only if $\phi(S)$ is feasible, (ii) follows. □

We are now ready to prove Lemma 3.2.1.

Proof of Lemma 3.2.1. By Lemmas 2.7.1, 2.8.1, 2.8.2, and 2.8.3, assumptions $E_1, \ldots, E_8$ hold with probability $1 - o(1)$. Hence we can assume that this is the case.

We first consider the case $M \geq 1$. By Lemma 3.2.4, with probability $1 - o(1)$, we have for all nodes v that a rumor initiated by v reaches a useful node u in $O(\ln^{3/4}(n) \ln \ln n)$ time steps. By Corollary 3.2.9, a rumor starting from a useful node u reaches node 1 in $O(\ln(n)/\ln \ln(n))$ time steps with probability $1 - o(n^{-1})$.

Corollary 3.2.9, Lemma 3.2.10, and a simple union bound show that with probability $1 - o(1)$ after another $O(\ln n/\ln \ln n)$ time steps, all useful nodes are informed. Similarly, Lemmas 3.2.4 and 3.2.10 together with a union bound prove that another $O(\ln^{3/4}(n) \ln \ln n)$ time steps suffice to have all nodes informed with probability $1 - o(1)$.

The case $M = 0$ is similar. The only difference is that instead of using Lemma 3.2.4, we use Lemma 3.2.3 which requires $O(\ln n)$ rounds.

The total failure probability is bounded by the sum of the probability that anyone of $E_1, \ldots, E_8$ does not hold and the probability that a node does not get informed within $O(\ln n/\ln \ln n)$ and $O(\ln n)$ rounds for $M \geq 1$ and $M = 0$, respectively, conditioned on $E_1, \ldots, E_8$. Since both probabilities are $o(1)$, the result follows. □

Chapter 4

The Asynchronous Protocol

We now consider the asynchronous protocol where each node takes action at times independently from the other nodes. In particular, we prove Theorem 2.6.2.

Theorem 2.6.2. *With probability $1 - o(1)$, the asynchronous push-pull protocol broadcasts a rumor from any node of G_m^n to*

- *all nodes in time $\Theta(\ln n)$,*
- *all but $o(n)$ nodes in time $O(\sqrt{\ln n})$.*

As in the synchronous case, throughout this section we consider a typical social network $G = G_m(W_1, \ldots, W_n)$ that satisfies $E_1, \ldots, E_5$ as well as E_6 and E_7. The proof is structured in a similar way as in the synchronous case. First, we show that in time $O((\ln \ln n)^2)$ a useful node is informed. Then, we prove that starting from a useful node, node 1 is informed in time $O(\sqrt{\ln n})$. Finally, using the symmetry lemma, we can extend these results to all nodes.

4.1 Informing the First Useful Node

In this section, all probabilities are taken over the product space of the random graph G and the random decisions of the rumor spreading process.

Lemma 4.1.1. *Let u be a fixed node. The rumor initiated by u reaches a useful node in time $O((\ln \ln n)^2)$ with probability $1 - o(1)$.*

Proof of Lemma 4.1.1. As in the proof of Lemma 3.2.3, we analyze a delayed protocol, i.e., we assume that all nodes perform only push operations and set some nodes to be inactive.

We consider *phases* of $33 \ln \ln n$ units of time. In the first phase, only u is active. In every subsequent phase, we assume that only the nodes that became informed in the previous phase are active. In other words, we assume that nodes that became informed in one phase *delay* their actions till the beginning of the next phase, and remain active only for that phase.

Let D_k be the set of all nodes that are active in phase k and let $n_k = |D_k|$. Let $K = 25 \ln\ln n$. Suppose that there is no useful vertex in any D_k for $k < K$. Then by E_7, we encounter at most one cycle consisting solely of nodes in $D_1 \cup \cdots \cup D_K$ (including self-loops or cycles due to multiple edges). To ease the presentation, we first regard the case that there is no such cycle and that $m \geq 3$ (i.e., every node has degree at least 3).

Note that a node of degree at least three contacts three distinct neighbors (and thus at least two uninformed neighbors) in at most $1+\frac{3}{2}+3 = \frac{11}{2}$ rounds in expectation. Since each round lasts for one unit of time in expectation, the expected time until such a node contacts three distinct neighbors is also $\frac{11}{2}$. By Markov's inequality, the probability that this does not happen within 33 units of time is at most $\frac{11}{2\cdot 33} = \frac{1}{6}$. Thus, for each node in D_k independently, the probability that it does not contact three distinct nodes in one phase is at most $(\frac{1}{6})^{\ln\ln n}$. Let X denote the set of such 'bad' nodes. Then, we have $\mathbb{E}[X] \leq (\frac{1}{6})^{\ln\ln n} n_k$. By Markov's inequality, we conclude that

$$\mathbb{P}[X \geq n_k/3] \leq 3E[X]/n_k \leq 3(\tfrac{1}{6})^{\ln\ln n} \leq 1/\ln n,$$

where the last inequality holds for sufficiently large n. Thus, with probability at least $1 - 1/\ln n$, we have $n_{k+1} \geq \frac{4}{3} n_k$. By a simple union bound, we conclude that with probability $1 - K/\ln n = 1 - o(1)$, we have $n_K \geq (\frac{4}{3})^K = \Omega(\ln^7 n)$. Assume that this is the case.

For each $i \in D_K$, only one of the nodes $\ell_{i,1}$ and $\ell_{i,2}$ can be informed (otherwise there would be a cycle). Let ℓ_i denote any of the nodes $\ell_{i,1}$ and $\ell_{i,2}$ that is not informed. Given the sequence $D_1, \ldots, D_K$, the nodes ℓ_i, with $i \in D_K$, are mutually independent. Conditioned on $\ell_i \notin D_1, \ldots, D_K$, the event that ℓ_i is useful can only become more likely since $D_1, \ldots, D_K$ only contain non-useful nodes. So by Lemma 2.8.5, for any $i \in D_K$, we have $\mathbb{P}[\ell_i \text{ is useful}] \geq \ln^{-3} n$. Also, for any $i \in D_K$, by Lemma 2.5.2, the probability that i contacts ℓ_i in one unit of time is at least $1 - \exp^{-1/\deg(i)} \geq 1 - \exp^{-1/(5m \ln^2 n)} \geq 1/(10m \ln^2 n)$, where the first inequality is due to E_6 and the second follows from (1.2.1). Since both events are independent, the probability that no node in D_K informs a useful node in one unit of time is at most $(1 - 1/(10m \ln^5 n))^{n_K} \leq \exp(-n_K/(10m \ln^5 n)) \leq n^{-\Omega(\ln n)}$. Thus, the total failure probability is $o(1) + n^{-\Omega(\ln n)} = o(1)$.

So far we have assumed that there is no cycle in $D_1, \ldots, D_K$ and that $m \geq 3$. It is easy to see that a single cycle reduces n_K only by a constant factor. Moreover, in case $m = 2$, note that both neighbors of a node of degree 2 must have degree at least 3. Using this fact, we can easily adjust the above argument. □

To reduce the error probability to $o(n^{-2})$, we need logarithmic time.

Lemma 4.1.2. *Let u be a fixed node. The rumor initiated by u reaches a useful node in time $O(\ln n)$ with probability $1 - o(n^{-2})$.*

Proof. The proof is almost identical to the proof of Lemma 3.2.3 in the synchronous case. As before we consider the same two stages. For the first stage, let X_i denote the time needed until an active node i contacts one or two uninformed nodes depending on whether it has degree 2 or more. Then, similar to the synchronous case, X_i is stochastically dominated by a geometric variable with parameter $1/3$ or a sum of two geometric variables each with paramter $1/4$. To see this, note that by Lemma 2.5.2, the probability that i contacts a (specific) uninformed neighbor within one unit of time is $1 - e^{-1/2} \geq \frac{1}{3}$ and $1 - e^{-1/3} \geq \frac{1}{4}$ if i has degree 2 or more, respectively. Since the distribution of the time to contact a specific neighbor is memoryless, we can bound it by a geometric distribution with these parameters. For the second stage, the probability that a node of degree 2 contacts a specific neighbor within five time units is at least $1 - e^{-5/2} \geq 0.91$. Similarly, the probability that a node of degree at least 3 with only one informed neighbor contacts two uninformed neighbors within five time units is $1 - e^{-5/3} \frac{(5/3)^2}{2} \geq 0.73$ respectively, which follows from Lemma 2.5.1 and Lemma 2.5.3 by defining the event that an uninformed node is contacted to be a type-1 event. Modifying the proof of Lemma 3.2.3 with these numbers, the result follows. □

4.2 Informing Node 1

Similar to the synchronous case, we use constant degree nodes to establish *fast links* between large degree nodes. More precisely, once a neighbor of a constant degree node is informed, the time until it has pulled the rumor from this neighbor and pushed it to one specific neighbor is (essentially) exponentially distributed. Thus, independent of their own degrees, two nodes that are connected via a third node of constant degree exchange information in time exponentially distributed.

Starting from one informed useful node, we study how fast the rumor spreads to the surrounding 'neighborhoods' of nodes. We consider 'levels' alternating between small degree nodes and *good* nodes i of relatively large weight w_i. The small degree nodes act as fast links between the levels of good nodes that ensure a large expansion. In particular, we make use of the fact that the good nodes in one level have a large neighborhood and since every small node in this neighborhood independently pulls the rumor in time exponentially distributed, we can argue that a considerable fraction of the neighborhood will be informed very fast. The larger this neighborhood is, the faster the rumor spreads to a sufficiently large fraction of it. These informed nodes then form the next level. In contrast, in the synchronous case, it would always take at least one time step for a neighbor to pull the rumor.

We consider informed neighborhoods at suitably chosen time steps on the continuous time line. The smaller these steps are chosen, the smaller the achieved expansion factor is at each step. On the other hand, smaller time steps allow us to progress faster through the different neighborhood levels. By carefully choosing each step size, we can balance out these opposing effects in order to achieve the following runtime.

Theorem 4.2.1. *Let $G = G_m(W_1, \dots, W_n)$ be a typical social network. Let $u \in [n]$ be a useful node. With probability $1 - o(n^{-1})$, using the asynchronous push-pull protocol, a rumor present at v reaches node 1 in time $O(\sqrt{\ln n})$.*

If not explicitly stated, all probabilities in this section are taken over the typical social network G.

For our argument using fast links, we will need many nodes of constant degree. We call nodes $i \in \left[\frac{2}{3}n, n\right]$ that have a degree equal to m *small*. In the proof of Lemma 3.2.6, we have already shown that there exists a linear number of small nodes with high probability.

Lemma 4.2.2. *Let $\varepsilon_m := \frac{1}{8}e^{-3m}$. With probability $1 - e^{-\Omega(n)}$, there are at least $\varepsilon_m n$ small nodes in $\left[\frac{2}{3}n, n\right]$.*

Crucial for a large expansion in each step are good nodes of large weight. As in the synchronous case, we say a node i is *good* if

$$i \in [s+1, 2^b] \text{ and } w_i \geq 1/(10\sqrt{in}),$$

where, as before, $s = 2^a$ is the smallest power of two larger than $\ln^{10} n$ and 2^b is the largest power of two smaller than $\frac{2}{3}n$. Let u be a useful node. Let $t_0 < t'_0 < t_1 < t'_1 < \dots$ denote discrete time steps to be specified later. We consider neighborhoods of u that are informed in the time intervals defined by any two consecutive time steps of these. In particular, we define sets Γ_k and Γ'_k recursively as follows. We set $\Gamma_0 = \{u\}$. Given the set Γ_k, Γ'_k consists of all *small* nodes $i \geq \frac{2}{3}n$ that contact a neighbor in Γ_k in time $[t_k, t'_k]$ and have not been included in any Γ'_ℓ with $\ell \leq k-1$. Similarly, Γ_k is defined as the set of all *good* nodes that are contacted by a neighbor in Γ'_{k-1} in time $[t'_{k-1}, t_k]$ and have not been included in any Γ_ℓ with $\ell \leq k-1$. Note that for all $k \geq 0$, Γ_k only contains nodes $i < \frac{2}{3}n$, while Γ'_k only contains nodes $i \geq \frac{2}{3}n$. This is true for Γ_0 since u is useful and by E_5, all useful nodes are smaller than $n/2$, and for $k > 0$, it holds by the definition of the sets. Thus, all these sets are pairwise disjoint. The following definitions are identical to the synchronous case. We define the *weight* of a set Γ_k by

$$f_k := \begin{cases} w_u & \text{if } k = 0 \\ \sum_{i \in \Gamma_k} \frac{1}{\sqrt{in}} & \text{if } k \geq 1. \end{cases}$$

Since for $k \geq 1$, Γ_k only contains good nodes, and by definition, $w_u = f_0$, we have for $k \geq 0$,

$$\sum_{i \in \Gamma_k} w_i \geq f_k/10. \tag{4.2.1}$$

Let $N_k = \bigcup_{0 \leq i \leq k} \Gamma_i$ and $N'_k = \bigcup_{0 \leq i \leq k} \Gamma'_i$. We denote by $C_0 \subseteq \left[\frac{2}{3}n, n\right]$ the set of small nodes and for $k \geq 1$ and by $C_k = C_0 \backslash N'_{k-1}$ the set of small nodes excluding nodes in $\Gamma'_0, \Gamma'_1, \ldots, \Gamma'_{k-1}$. By Lemma 4.2.2, we have $C_0 \geq \varepsilon_m n$ with probability $1 - e^{-\Omega(n)}$.

The next lemma shows that we achieve an exponential expansion in terms of f_k in each level as long as there is still a linear number of small nodes in C_k and similarly, as long as for each interval $I_t := [2^t + 1, 2^{t+1}]$, where $t \in [a, b)$, there are still 2^{t-2} good nodes not in N_k.

Lemma 4.2.3. *Let $c > 0$ be a sufficiently large constant, $k \geq 0$ be such that $\ln^4(n)/\sqrt{n} \geq f_k \geq \ln^2(n)/n$ and $|C_k| \geq \varepsilon_m n/2$. Let*

$$\Delta_k = m \ln(1 - c \log_2^{-1/2}(f_k n / \ln^2 n)).$$

Set $t'_k := t_k + \Delta_k$ and $t_{k+1} := t'_k + \Delta_k$. Given C_k and $\Gamma_0, \Gamma'_0, \Gamma_1, \Gamma'_1, \ldots, \Gamma_k$, with probability $1 - O(n^{-6/5})$, one of the following is satisfied:

- $|N_{k+1} \cap I_t| \geq 2^{t-2}$*, for some $t \in [a, b)$, or*
- $f_{k+1} \geq 2f_k$.

Proof. All probabilities are conditioned on the assumptions in the lemma. We first bound $|\Gamma'_k|$ from below. Let $C'_k \subseteq C_k$ denote the set of all $j \in C_k$ such that j contacts $\ell_{j,1}$ in time $[t_k, t'_k]$. For each $j \in C_k$ independently, by Lemma 2.5.2, the probability that $j \in C'_k$ is $1 - e^{-\Delta_k/m} = c \log_2^{-1/2}(f_k n/\ln^2 n)$. Let

$$d := c \log_2^{-1/2}(f_k n/\ln^2 n)/2$$

. Thus, we have $\mathbb{E}[|C'_k|] \geq 2d|C_k|$ and by Chernoff's bound, we have $\mathbb{P}[|C'_k| \leq d|C_k|] \leq \exp(-\Omega(d|C_k|)) = e^{-\Omega(n/\sqrt{\ln n})}$, where the last inequality follows from

$$d = \tfrac{c}{2} \log_2^{-1/2}(f_k n/\ln^2 n) \geq \tfrac{c}{2} \log_2^{-1/2}(\ln^2(n)\sqrt{n}) = \Omega(1/\sqrt{\ln n}), \tag{4.2.2}$$

where we have used $f_k \leq \ln^4(n)/\sqrt(n)$. So assume that $|C'_k| \geq d|C_k|$. Let $j \in C'_k$. The event that $\ell_{j,1} \in \Gamma_k$ can only become more likely if we condition on $j \in C'_k$ since this implies that either $\ell_{j,1} \notin N_{k-1}$ or for all $k' < k$, we have $j \notin C'_{k'}$. Thus, we have

$$\mathbb{P}[\ell_{j,1} \in \Gamma_k] \geq \Big(\sum_{i \in \Gamma_k} w_i\Big)/W_j \geq f_k/10,$$

where the last inequality follows from (4.2.1). Then, $\mathbb{E}[|\Gamma'_k|] \geq |C'_k| f_k/10 \geq d\,\varepsilon_m n f_k/20$. By Chernoff's bound, we obtain

$$\mathbb{P}[|\Gamma'_k| \leq d\varepsilon_m n f_k/21] \leq \exp(-\Omega(d\varepsilon_m n f_k)) \leq n^{-\Omega(\sqrt{\ln n})}, \tag{4.2.3}$$

where the last inequality follows from (4.2.2) and $f_k \geq \ln^2(n)/n$. In the following, we assume

$$|\Gamma'_k| \geq d\varepsilon_m n f_k/21. \tag{4.2.4}$$

Similarly, let $\widetilde{\Gamma}'_k \subseteq \Gamma'_k$ be the set of nodes $j \in \Gamma'_k$ such that j contacts $\ell_{j,2}$ in time $[t'_k, t_{k+1}]$. Again we have $\mathbb{E}[|\widetilde{\Gamma}'_k|] \geq 2d|\Gamma'_k|$ and by Chernoff's bound,

$$\mathbb{P}[|\widetilde{\Gamma}'_k| \leq d|\Gamma'_k|] \leq \exp(-\Omega(d|\Gamma'_k|)) = \exp(-\Omega(d^2 \ln^2(n))) = n^{-2}, \tag{4.2.5}$$

where the last inequality holds for sufficiently large, but constant $c > 0$ (note that d depends on c). So, assume that

$$|\widetilde{\Gamma}'_k| \geq d|\Gamma'_k| \geq d^2\,\varepsilon_m n f_k/21, \tag{4.2.6}$$

where the last inequality follows from (4.2.4).

We now show that either $|N_{k+1} \cap I_t| \geq 2^{t-2}$ for some $t \in [a,b)$, or with high probability, for sufficiently many $t \in [a,b)$, we have $|\Gamma_{k+1} \cap I_t| = \Omega\big(|\widetilde{\Gamma}'_k|\sqrt{2^t/n}\big)$. Let $t \in [a,b)$. By E_2, I_t contains at least 2^{t-1} good nodes. Let S be initially the set of good nodes in $I_t \setminus N_k$. So $|S| \geq 2^{t-1} - |N_k \cap I_t|$. Let $\widetilde{\Gamma}$ be initially an empty set. We consider the elements of $\widetilde{\Gamma}'_k$ one by one in any order. Whenever we encounter some node $i \in \widetilde{\Gamma}'_k$ that is connected to some node $j \in S$ via its second link (i.e., $\ell_{i,2} = j$), we remove j from S and include it into $\widetilde{\Gamma}$. Note that $\widetilde{\Gamma} \subseteq \Gamma_{k+1}$ throughout this process. Moreover, $\ell_{i,2}$ has not been revealed before i is considered ($\ell_{i,1}$ is independent from $\ell_{i,2}$ given all W_j values). Since S only contains good nodes from $[2^t+1, 2^{t+1}]$, we have for every $i \in S$, $w_i \geq 1/(10\sqrt{2^{t+1}n})$. Hence, as long as $|S| \geq 2^{t-2}$, we have for every node $i \in \widetilde{\Gamma}'_k$,

$$\mathbb{P}[\ell_{i,2} \in S] \geq \frac{2^{t-2}}{10W_i\sqrt{2^{t+1}n}} \geq \frac{\sqrt{2^t}}{60\sqrt{n}}.$$

If $|S| < 2^{t-2}$ at some point, then $|N_{k+1} \cap I_t| \geq 2^{t-2}$ and we are finished. Otherwise let $Y = \sum_{1 \leq i \leq |\widetilde{\Gamma}'_k|} Y_i$, where $Y_1, \ldots, Y_{|\widetilde{\Gamma}'_k|}$ are mutually independent 0/1-random variables with $\mathbb{P}[Y_i = 1] = \frac{\sqrt{2^t}}{60\sqrt{n}}$ for all $1 \leq i \leq |\widetilde{\Gamma}'_k|$. Note that $|\Gamma_{k+1} \cap I_t|$ stochastically dominates Y, i.e., $\mathbb{P}[|\Gamma_{k+1} \cap I_t| > t] \geq \mathbb{P}[Y > t]$. We have

$$\mathbb{E}[Y] = |\widetilde{\Gamma}'_k|\frac{\sqrt{2^t}}{60\sqrt{n}} =: \mu_t. \tag{4.2.7}$$

Now, if $\mu_t \geq 10 \ln n$, we have by Chernoff's bound $\mathbb{P}[Y \leq \mu_t/2] \leq n^{-5/4}$, and thus

$$\mathbb{P}[|\Gamma_{k+1} \cap I_t| \leq \mu_t/2] \leq \mathbb{P}[Y \leq \mu_t/2] \leq n^{-5/4}.$$

Let $\mathcal{T}$ be the set of indices $t \in [a, b)$ with $\mu_t \geq 10 \ln n$. Taking a union bound over all indices $t \in \mathcal{T}$, we conclude that with probability at least $1 - \log_2(n) \cdot n^{-5/4} = 1 - O(n^{-6/5})$, we have for all these $t \in [a, b)$,

$$\sum_{i \in \Gamma_{k+1} \cap I_t} \frac{1}{\sqrt{in}} \geq \frac{\mu_t}{2} \cdot \frac{1}{\sqrt{2^{t+1} n}} = \frac{|\widetilde{\Gamma'_k}|}{120 n \sqrt{2}} \geq \frac{d^2 \varepsilon_m f_k}{3564},$$

where the last inequality follows from (4.2.6). Since $2^b \geq n/3$, we have $|\mathcal{T}| \geq \min\{2 \log_2(d^2 \varepsilon_m f_k n / 21824 \ln n), b - a\}$. Therefore

$$\begin{aligned} f_{k+1} &\geq \min\{2 \log_2(d^2 \varepsilon_m f_k n / 21824 \ln n), b - a\} d^2 \varepsilon_m f_k / 3564 \\ &\geq \min\{2 \log_2(f_k n / \ln^2 n), b - a\} d^2 \varepsilon_m f_k / 3564 \\ &\geq 2 \left(\tfrac{c}{2}\right)^2 \varepsilon_m f_k / 3564 \geq 2 f_k, \end{aligned}$$

where the last two inequalities hold for sufficiently large, but constant c. The total failure probability is $e^{-n/\sqrt{\ln n}} + n^{-\Omega(\sqrt{\ln n})} + n^{-2} + O(n^{-6/5}) = O(n^{-6/5})$. □

We can now use Lemma 4.2.3 to show that the rumor quickly proceeds from a useful node to node 1.

Proof of Theorem 4.2.1. Consider a run of the process started with a rumor in u. We apply Lemma 4.2.3 consecutively for $k = 0$ until K, where $K = \ln(n)$. The probability that we can do so K times is at least $1 - O(K\, n^{-6/5}) = 1 - O(n^{-7/6})$. In the following we assume this is the case.

Note that for $k = 0$, $f_k = w_u \geq \ln^2(n)/n$ holds since u is useful. Also, by Lemma 4.2.2, we have $|C_0| \geq \varepsilon_m n/2$ with probability $1 - e^{-\Omega(n)}$. Assume that this is the case. Then we can apply Lemma 4.2.3 for $k = 0$. The only way we fail to apply Lemma 4.2.3 for some k', where $0 < k' < K$, is when $f_{k'} \geq \ln^4(n)/\sqrt{n}$ or $|C_{k'}| < \varepsilon_m n/2$.

If $|C_{k'}| < \varepsilon_m n/2$, then there must be a k'', $0 \leq k'' \leq k'$, with $|\Gamma'_{k''}| \geq \frac{\varepsilon_m n/2}{K} \geq \Omega(\frac{n}{\ln n})$. We stop the sequence at $\Gamma'_{k''}$ as soon as we encounter such a k''. Given the sequence $\Gamma_0, \Gamma'_0, \Gamma_1, \ldots, \Gamma'_{k''}$, the second links of the nodes in $\Gamma'_{k''}$ are mutually independent random variables. Moreover, for each $j \in \Gamma'_{k''}$, the probability that j contacts $\ell_{j,2}$ within m units of time is $1 - e^{-m/m} = 1 - e^{-1}$ (independent from $\ell_{j,2}$).

So the probability that no node in $\Gamma'_{k''}$ contacts node 1 within m units of time is at most

$$\begin{aligned} (1 - (1 - e^{-1}) w_1)^{|\Gamma'_{k''}|} &\leq \left(1 - \tfrac{4(1 - e^{-1})}{\ln(n)\sqrt{n}}\right)^{\Omega(n/\ln n)} \\ &\leq \exp\left(-\Omega(\sqrt{n}/\ln^2 n)\right) = n^{-\Omega(\sqrt{n}/\ln^3 n)}, \end{aligned}$$

where the first inequality follows from E_3. We can therefore assume that $|C_k| \geq \varepsilon_m n/2$ for all $k = 0, \ldots, K-1$.

Similarly, if $|N_{k'} \cap I_t| \geq 2^{t-2}$ for some $t \in [a, b)$, there must be a k'' where $0 < k'' \leq k'$, with $|\Gamma_{k''} \cap I_t| \geq 2^{t-2}/K$. We stop the construction of the sequence $\Gamma_0, \Gamma'_0, \Gamma_1, \ldots$ at $\Gamma'_{k''}$. By (4.2.3) we have with probability $1 - n^{-\Omega(\sqrt{\ln n})}$,

$$\begin{aligned}
|\Gamma'_{k''}| &\geq d\,\varepsilon_m n f_{k''}/21 \geq d\,\varepsilon_m n |\Gamma_{k''} \cap I_t|/(21\sqrt{2^{t+1}n}) \\
&\geq d\,\varepsilon_m n 2^{t-2}/(21K\sqrt{2^{t+1}n}) \geq \Omega(d\,2^{t/2}\sqrt{n}/\ln n) \\
&\geq \Omega(\ln^3(n)\sqrt{n}),
\end{aligned}$$

where the last inequality follows from $2^t \geq \ln^{10} n$ and (4.2.2). So, by the same argument as above, given $\Gamma'_0, \Gamma_1, \ldots, \Gamma'_{k''}$, the probability that no node in $\Gamma'_{k''}$ contacts node 1 within m time units is at most

$$\begin{aligned}
(1-(1-e^{-1})w_1)^{|\Gamma'_{k''}|} &\leq \big(1 - \tfrac{4(1-e^{-1})}{\ln(n)\sqrt{n}}\big)^{\Omega(\ln^3(n)\sqrt{n})} \\
&\leq \exp\big(-\Omega(\ln^2 n)\big) = n^{-\Omega(\ln n)}.
\end{aligned}$$

Similarly, if $f_{k'} \geq \ln^4(n)/\sqrt{n}$, then by (4.2.3), we have

$$|\Gamma'_{k'}| \geq d\,\varepsilon_m n f_{k'}/21 \geq \Omega(\sqrt{n}\ln^3(n)) \tag{4.2.8}$$

with probability $1 - n^{-\Omega(\ln n)}$. Given $\Gamma_0, \Gamma'_0, \Gamma_1, \ldots, \Gamma'_{k'}$, the probability that no node in $\Gamma'_{k'}$ contacts node 1 within m time units is at most

$$\begin{aligned}
(1-(1-e^{-1})w_1)^{|\Gamma'_{k'}|} &\leq \big(1 - \tfrac{4(1-e^{-1})}{\ln(n)\sqrt{n}}\big)^{\Omega(\sqrt{n}\ln^3(n))} \\
&\leq \exp(-\Omega(\ln^2 n)) \leq n^{-\Omega(\ln n)}.
\end{aligned}$$

So assume now that $|C_k| \geq \varepsilon_m n/2$ and $f_{k+1} \geq 2f_k$, for all k, $0 \leq k < K$, where $K = \ln(n)$. But then since $2^K f_0 \geq \ln^2 n$, we conclude that there must be a $0 \leq k'' < K$ such that $f_{k''} \geq \ln^4(n)/\sqrt{n}$ and we can apply the previous argument.

Thus the total time needed to inform node 1 is at most

$$\begin{aligned}
2\sum_{k=0}^{K}\Delta_k &= 2m\sum_{k=0}^{K}\ln(1 - c\,\log_2^{-1/2}(f_k n/\ln^2 n)) \\
&\leq 2mc\sum_{k=0}^{K}\log_2^{-1/2}(f_k n/\ln^2 n) && \text{by } 1-x \leq e^{-x} \\
&\leq 2mc\sum_{k=0}^{K}\log_2^{-1/2}(2^k f_0 n/\ln^2 n) \\
&\leq 2mc\sum_{k=0}^{K}\log_2^{-1/2}(2^k) && \text{by } f_0 \geq \ln^2(n)/n \\
&= O\Big(\sum_{k=1}^{K} k^{-1/2}\Big) \\
&= O\Big(\int_0^K x^{-1/2}\,\mathrm{d}x\Big) = O(\sqrt{\ln n}).
\end{aligned}$$

The total failure probability is $O(n^{-7/6}) + e^{-\Omega(n)} + n^{-\Omega(\sqrt{n}/\ln^3 n)} + n^{-\Omega(\ln n)} + n^{-\Omega(\sqrt{\ln n})} + n^{-\Omega(\ln n)} = O(n^{-7/6})$. □

4.3 Informing All Nodes

As in the synchronous case, the 'symmetry lemma' (Lemma 3.2.10) also holds for the asynchronous case. The proof is almost identical to the proof for the synchronous push-protocol with $M = 0$ [18]. The only difference is that instead of considering time steps, we consider a time span.

Lemma 4.3.1. *In the asynchronous push-pull protocol, assume that if the rumor starts in node u, it reaches node v in time t with probability p. This implies the reverse statement: if the rumor is initiated by v, then it reaches u in time t with probability p.*

We are now ready to prove the main result.

Proof of Theorem 2.6.2. By Lemmas 2.7.1, 2.8.1, and 2.8.2, assumptions $E_1, \ldots, E_7$ hold with probability $1 - o(1)$. Hence, we can assume that this is the case. Let v be any node that initiates the rumor. By Lemma 4.1.1, we have with probability $1 - o(1)$ that the rumor reaches a useful node u in $O((\ln\ln n)^2)$ units of time. By Theorem 4.2.1, after another $O(\sqrt{\ln n})$ units of time node 1 is informed with probability $1 - o(n^{-1})$.

Theorem 4.2.1, Lemma 4.3.1, and a simple union bound show that with probability $1-o(1)$ after another $O(\sqrt{\ln n})$ units of time, all useful nodes are informed. Similarly, Lemmas 4.1.1 and 4.3.1 prove that another $O((\ln\ln n)^2)$

units of time suffice to inform each non-useful node with probability $1-o(1)$. Thus, by Markov's inequality, we conclude that with probability $1-o(1)$, $(1-o(1))n$ nodes are informed after $O(\sqrt{\ln n})$ units of time.

By a simple union bound, we can bound the total failure probability by $o(1)$. By using Lemma 4.1.2 instead of Lemma 4.1.1 in the previous argument, it follows that *all* nodes are informed after $O(\ln n)$ units of time with probability $1-o(1)$.

For the lower bound, note that the probability that the clock of a node ticks for the first time after x units of time is e^{-x}. Thus, for a node u of degree m, with probability at least $e^{-(m+1)x}$, it takes at least x units of time until u contacts or is contacted by any of its neighbors. Since by Lemma 4.2.2, there are $\Omega(n)$ such small nodes with probability $1-o(1)$, we conclude that the probability that no small node remains uninformed after $\frac{\ln n}{2(m+1)}$ units of time is at most $(1-e^{-\ln(n)/2})^{\Omega(n)} = (1-\frac{1}{\sqrt{n}})^{\Omega(n)} \leq e^{-\Omega(\sqrt{n})}$. It follows that with probability $1-o(1)$, after $\Omega(\ln n)$ units of time, there are still uninformed nodes. □

Chapter 5

Excursus: The Alternative Model Revisited

This section is special. We have included it only because it puts the alternative model that was a crucial part in the analysis of the protocol into a more systematic perspective. The alternative model allowed us to deal with the dependencies in the definition of the preferential attachment graph in a very convenient way. Once we conditioned on the W_i values, all dependencies were resolved. It turns out that this seemingly ad-hoc solution can be seen as an application of a more general framework. We now introduce this framework and then apply it to preferential attachment graphs. The material covered in this section can be found in more detail in [33] and [47].

As a warm-up consider the following simple example. Let $p \in [0, 1]$ be a random variable. Let $X_1, X_2, \ldots, X_n$ be the indicator random variables, each for a new random experiment that succeeds with probability p and fails with probability $1 - p$. If p was fixed, then these are just *independent* Bernoulli trials with parameter p. We therefore call such a distribution a *mixture* of independent and identically distributed (i.i.d.) Bernoulli variables. In our case, the X_i's are not independent. However, their order does not matter; for any permutation σ of $\{1, \ldots, n\}$, the sequence

$$X_{i_1}, X_{i_2}, \ldots, X_{i_n}$$

has the same joint probability distribution. Such sequences are called *exchangeable.* Infinite sequences of random variables are said to be exchangeable if every finite collection of its variables is exchangeable.

Interestingly, not only is a mixture of i.i.d. Bernoulli random variables exchangeable, but also the converse is true; every exchangeable sequence of Bernoulli random variables is a mixture of i.i.d. Bernoulli variables. This is the statement of De-Finetti's Theorem.

Theorem 5.0.2. *Let $X_1, X_2, \ldots,$ be an infinite sequence of exchangeable Bernoulli random variables. Then there exists a random variable U, called* *the* mixing measure*, with* $\mathbb{P}[U \in [0,1]] = 1$ *such that, for all* $1 \leq k \leq n$*,*

$$\mathbb{P}[X_1 = \cdots = X_k = 1, X_{k+1} = \cdots = X_n = 0] = \mathbb{E}[U^k(1-U)^{n-k}].$$

Let $S_n := \sum_{i=1}^n X_i$*. Then, if*

$$\lim_{n\to\infty} n\,\mathbb{P}[S_n = \lceil un \rceil] = f(u), \tag{5.0.1}$$

then $f(u)$ *is the density of* U*.*

We now apply this framework to the preferential attachment graph model. We describe the preferential attachment graph with the following urn model. For simplicity, we consider the case $m = 1$, i.e., every newly added node connects to exactly one of the existing nodes using the preferential attachment rule. Thus, we start with a single node with one loop. We can model this case by an urn containing two balls numbered 1. In step $k > 1$, we draw a ball uniformly at random and return it to the urn with a copy of it and a new node numbered k. This procedure is equivalent to choosing a node proportional to its current degree in the PA model. Therefore, in each step, the number of balls numbered i is sampled exactly like the degree of node i in the preferential attachment graph after the same number of steps.

Suppose now that we stop this process at step k; so we have $2k$ balls in the urn. We color all balls $j < k$ blue, and the ball numbered k red. We then proceed according to the following rule. In each step, we choose one ball uniformly at random and return it with another ball of the same color. Note that this corresponds to the selection of a node under the PA rule if we condition on the event that the node selected is among the first k nodes selected and if we identify all nodes $j < k$. Thus, in each step, the probability of choosing a red ball is the same as the conditional probability of choosing node k when we assume that the node selected is among the first k nodes. This modified process is the well-known Polya urn scheme (see [54]) where we start with $r = 1$ red balls and $b = 2k - 1$ blue balls.

Let X_i be the indicator random variable for the event that the i-th draw in this process is a red ball. We argue that $X_1, X_2, \ldots$ is an exchangeable sequence. To see this, note that the probability that the first n_1 draws are red balls, and the following $n_2 = n - n_1$ draws are blue is

$$\frac{r}{r+b}\frac{r+1}{r+b+1}\cdots\frac{r+n_1-1}{r+b+n_1-1}\frac{b}{r+b+n_1}\frac{b+1}{r+b+n_1+1}\cdots\frac{b+n_2-1}{r+b+n-1}.$$

Note that we have the same probability for any order of the n_1 draws of red balls. It follows that the probability of any particular sequence of draws only depends on the number of red balls in this sequence, and not on the order of these draws. Thus, the sequence is exchangeable. By Theorem 5.0.2, the sequence is conditionally independent and identically distributed. To obtain the mixing measure, we compute the probability that $S_n := \sum_{i=1}^n X_i = n_1$. This probability is given by

$$\begin{aligned}&\binom{n}{n_1}\frac{(r+n_1-1)!}{(r-1)!}\frac{(b+n_2-1)!}{(b-1)!}\frac{(r+b-1)!}{(r+b+n-1)!}\\=&\frac{(r+b-1)!}{(r-1)!(b-1)!}\frac{(r+n_1-1)!}{n_1!}\frac{(b+n_2-1)!}{n_2!}\frac{n!}{(r+b+n-1)!}.\end{aligned}$$

The last three factors can be rewritten as

$$\frac{(r+n_1-1)(r+n_1-2)\cdots(n_1+1)(b+n_2-1)\cdots(n_2+1)}{(r+b+n-1)(r+b+n-2)\cdots(n+1)}.$$

So for fixed r and b and $n_1/n \to u$, the whole term converges to

$$\frac{(r+b-1)!}{(r-1)!(b-1)!}u^{r-1}(1-u)^{b-1}\frac{1}{n}.$$

By (5.0.1), we conclude that U has density

$$f(u) = \frac{(r+b-1)!}{(r-1)!(b-1)!}u^{r-1}(1-u)^{b-1}.$$

This is the so-called beta(r, b) distribution. Thus, by Theorem 5.0.2, the Polya urn scheme with $r = 1$ red balls and $b = 2k-1$ blue balls is equivalent to first sampling $\psi_k \sim$ beta$(1, 2k-1)$ and then, *in each round independently*, adding to the urn a red ball with probability ψ_k and a blue ball with probability $1-\psi_k$. Similarly, we can simulate the original urn model with numbered balls by first setting $\psi_1 = 1$ and sampling $\psi_k \sim$ beta$(1, 2k-1)$ for $k > 1$. Then, in the j-th step, we draw a ball numbered $k \leq j$ with probability

$$\psi_k \prod_{i=k+1}^{j}(1-\psi_i), \tag{5.0.2}$$

since with probability $\prod_{i=k+1}^{j}(1-\psi_i)$ we draw a ball numbered at most k, and with probability ψ_k we draw a ball numbered k conditioned on the event of drawing a ball numbered at most k. Alternatively, we can define for $1 \leq k \leq n$,

$$\phi_k = \psi_k \prod_{i=k+1}^{n}(1-\psi_i).$$

Then, by a simple induction on n, we see that $\sum_{k=1}^{n} \phi_k = 1$. Then, the term in (5.0.2) is equivalent to $\frac{\phi_k}{\sum_{i=1}^{j} \phi_i}$ for $k \leq j$, since

$$\begin{aligned}\frac{\phi_k}{\sum_{i=1}^{j} \phi_i} &= \frac{\psi_k \prod_{i=k+1}^{n}(1-\psi_i)}{\sum_{\ell=1}^{j} \psi_\ell \prod_{i=\ell+1}^{n}(1-\psi_i)} \\ &= \frac{\psi_k \prod_{i=k+1}^{j}(1-\psi_i)}{\sum_{\ell=1}^{j} \psi_\ell \prod_{i=\ell+1}^{j}(1-\psi_i)} \\ &= \psi_k \prod_{i=k+1}^{j} (1-\psi_i),\end{aligned}$$

where the last equality follows from noting that $\sum_{\ell=1}^{j} \psi_\ell \prod_{i=\ell+1}^{j}(1-\psi_i) = 1$ since it is the sum of the probabilities that we draw a ball numbered $k \leq j$ in the j-th step over all possible k.

In conclusion, it follows that the preferential attachment graph G_1^n can be obtained by first sampling the ψ_k values, and then for each node i *independently*, we add an edge to node $j \leq i$ where j is chosen with probability $\phi_j / \sum_{k=1}^{i} \phi_k$. Thus, the ϕ_j values correspond exactly to the w_j values in the alternative model that we used in the analysis.

Chapter 6

Experiments

To the hard-core theoretician this chapter might come as a surprise. After all, we have proved *rigorously* bounds on the running time of the protocols in social networks. However, the theoretical analysis falls short of telling us whether the proven *asymptotical* advantage of using memory also translates to a better performance already for *practical* graph sizes. Or whether, in practice, such a speed-up is special for preferential attachment graphs at all. The same question can also be asked for the asynchronous protocol.

We conducted an experimental investigation in order to (a) better understand the performance of randomized rumor spreading protocols in preferential attachment networks which might help in the design of efficient communication networks, and (b) to better understand the advantage of equipping nodes with a fixed amount of memory to avoid contacting a constant number of previous contacts; this is interesting from the viewpoint of algorithm design.

In summary, our main findings are the following. Generally, rumor spreading is very fast in preferential attachment networks, significantly faster than in random-attachment networks (same density) and hypercubes (much denser), and faster than in complete networks (unless the density is very small).

There is a clearly visible advantage of keeping track of the most recently contacted neighbor (using a memory of size one) in preferential attachment networks, particularly if the density is small. There is less to be gained from memory on random attachment networks and almost no gain in complete networks and hypercubes. Additional memory is of some benefit, but not very much.

Furthermore, our experiments show that the asynchronous model is faster on all graph classes, but a clearly greater advantage for preferential attachment graphs is not visible.

We conducted similar experiments on crawls of the Twitter and Orkut online social network. Interestingly, we observe an even faster information dissemination than in preferential attachment graphs of corresponding size and density. These experiments also confirm that tracking one neighbor (memory of size one) leads to a significant improvement, whereas using additional memory to track more neighbors does not produce significant gains.

In each experiment, we chose a new random source node per run. Also for the experiments on random graphs, we sampled a new graph per run.

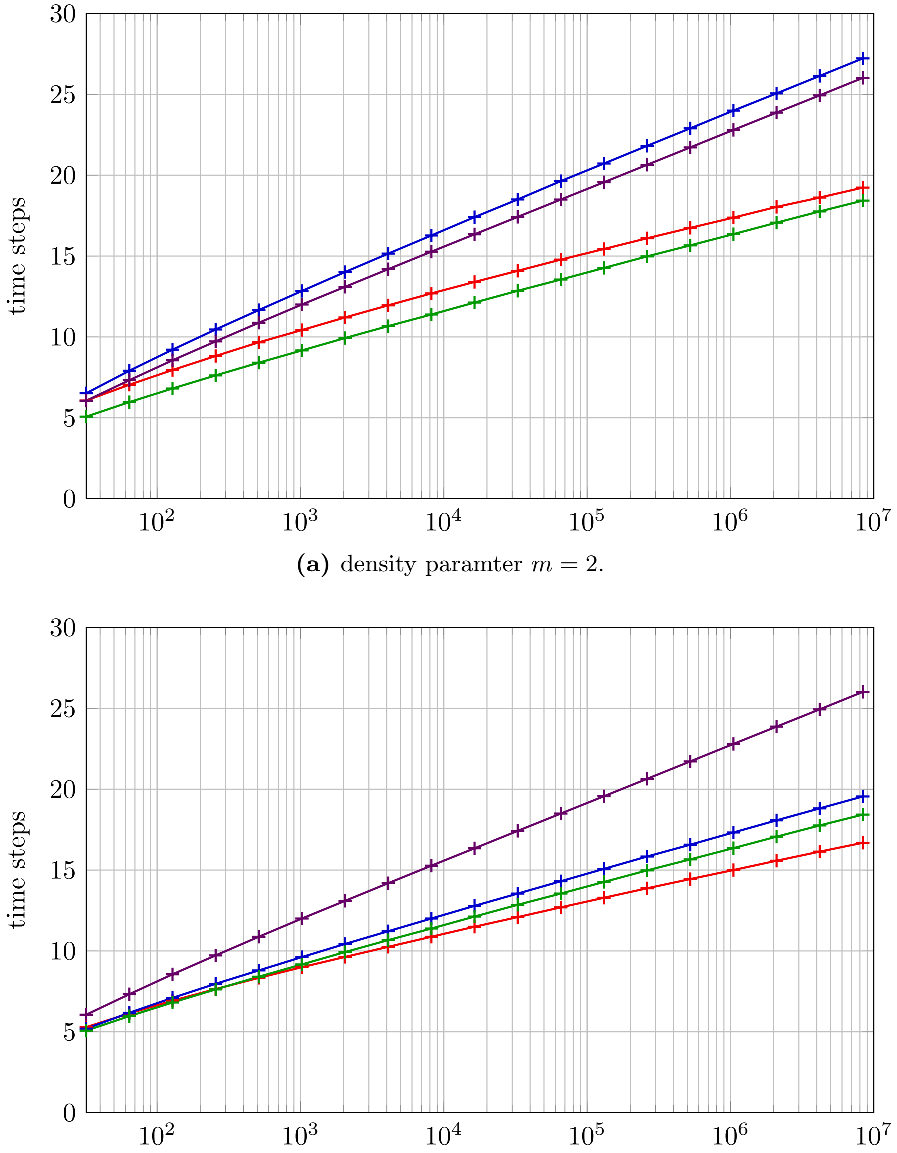
6.1 Fast Broadcasting in Preferential Attachment Graphs, Influence of Graph Density

Our theoretical result shows that rumor spreading in the random phone call model with memory at least one has an asymptotically faster runtime of $\Theta(\ln n/\ln\ln n)$ in preferential attachment graphs, in contrast to the $\Theta(\ln n)$ time observed (i) for the no-memory version in preferential attachment graphs and (ii) regardless of memory on most classic graphs like complete graphs, hypercubes, and random attachment graphs. Since only asymptotic results were proven, it is not clear if the proven differences are apparent for reasonable graph sizes. This is the focus of the current section. We have examined the average time needed to inform all vertices, starting from a random vertex, for different graphs.

We compare our results with similar experiments on the complete graph, hypercubes and *random-attachment networks*. In this network model, also known as the m-out model [7], each node chooses m other nodes as neighbors uniformly at random; finally, this neighbor relation is made symmetric and multiple edges are removed. Consequently, we obtain a random graph with average degree close to $2m$ and minimum degree at least m. These graphs form a good point of comparison with preferential attachment graphs with density parameter m, where nodes also choose m random neighbors, but according to the preferential attachment paradigm.

In Figure 6.1, we show the broadcast times observed for complete graphs, hypercubes, and preferential and random attachment graphs with density parameters $m = 2$ and $m = 10$, with memory one. We observe that rumor spreading is quite fast in preferential attachment graphs, faster than in hypercubes for both density parameters $m = 2$ and $m = 10$, and even faster than in complete graphs for $m = 10$. The relatively sparse preferential attachment graphs with $m = 2$, also become faster than complete graphs for sufficiently large graph size.

We observed structurally different behavior of the information spreading process on the different graphs. To be precise, let us consider graphs with $n = 10^6$ vertices and $m = 2$, averaged over 10,000 runs. Then on average 57% of the nodes of a random attachment graph are informed with a pull operation (and 43% via push). On the other hand, in preferential attachment graphs 73% of the nodes are informed by a pull operation. Moreover, on



(a) density paramter $m = 2$.

(b) density paramter $m = 10$.

Figure 6.1: Comparison of synchronous rumor spreading with memory one on preferential attachment graph (——), random-attachment graph (——), complete graph (——), and hypercube (——). The two charts show different density parameters of the preferential and random-attachment graph. The results for complete graphs and hypercubes are equivalent in both charts; they are given for comparison. The x-axis corresponds to the number of vertices $n = 2^5 \dots 2^{23}$. The y-axis corresponds to the runtime to inform all vertices, averaged over 10,000 runs.
For $m = 10$ the preferential attachment graph performs faster than all other graph classes. For the considered small ($n \leq 2^{23}$) and very sparse case ($m = 2$), the complete graph is even faster than the preferential attachment graph.

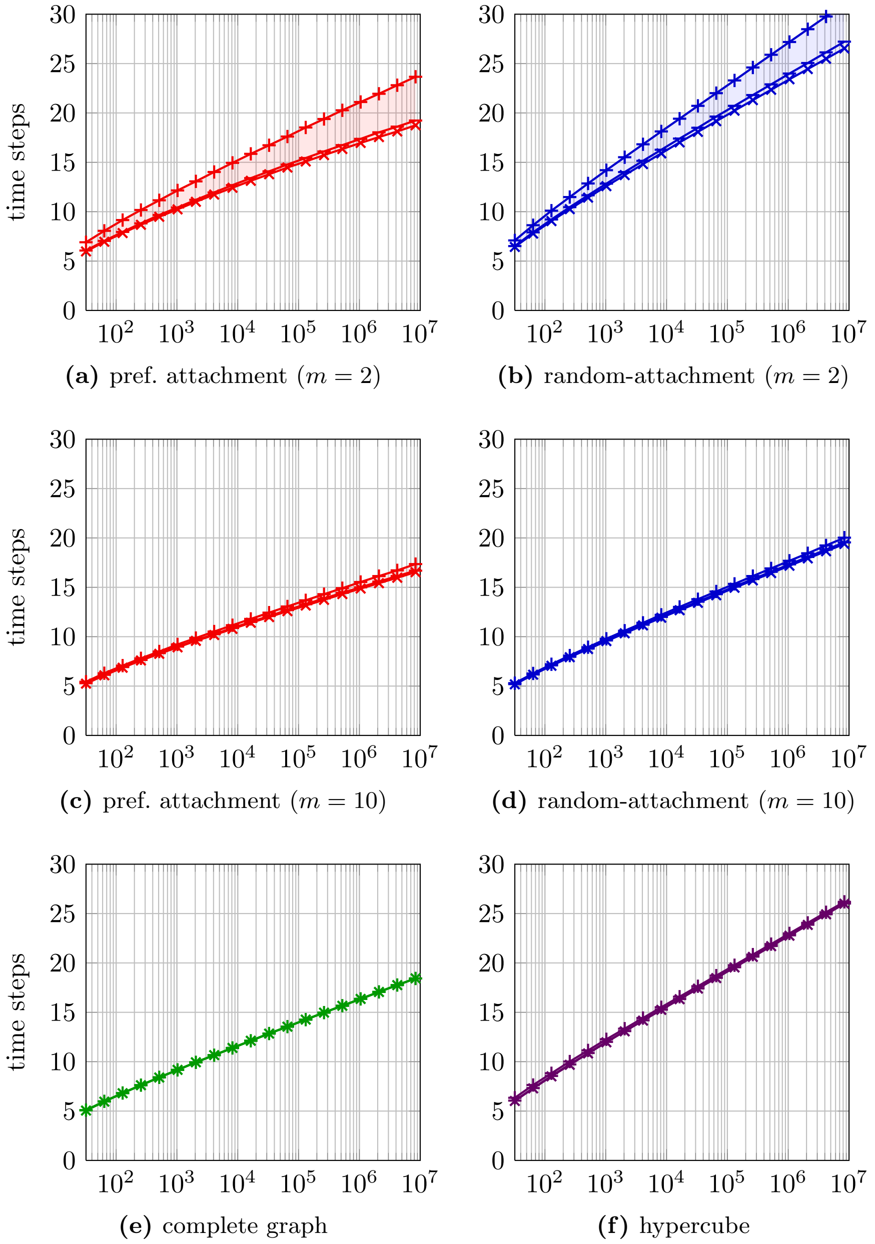
average such a pull information transfers the rumor from a large degree node (with degree 66 on average) to a node with small degree (with degree 3 on average). This complies with our proof that makes use of small degree nodes (fast nodes) who pull the rumor from large degree nodes (useful nodes).

The path along which a rumor is spread in a preferential attachment graph seems to differ from the typical paths in a random attachment graph. We measured the number of rounds needed to inform a node and compared it with the graph distance of the node to the source. In general, it is preferable to have a good correlation between the two measures [51]. The graph distance from the source gives a lower bound for the number of rounds needed to inform a node. We call the difference between the number of rounds needed and the graph distance the *delay*. If the delay is small, the information is spread on nearly shortest paths. On random attachment graphs we observed that vertices which are less than six steps away from the source have a delay of less than one on average. In preferential attachment graphs, nodes with distances between two and six from the source have on average a delay of four. This shows that in preferential attachment graphs the information is *not* spread via shortest paths, but via detours. Again this complies with our proof that constructs paths alternating between useful and fast nodes that are not shortest.

6.2 The Effect of Short-Term Memory

Perhaps the most surprising finding of our theoretical result is that keeping track of a certain small number of recently-contacted neighbors, and avoiding selecting any of these when randomly choosing the next communication partner, significantly reduces the time needed to inform all nodes of preferential attachment networks. Remember that for the classic random phone call model, this time is $\Theta(\ln n)$. If the communication partners are chosen uniformly at random from all neighbors except the one called in the previous round (short-term memory of size one), then this time decreases to $\Theta(\ln n/\ln\ln n)$. Using additional memory to track more than one recent contact, however, does not yield times better than $\Theta(\ln n/\ln\ln n)$.

In this section, we experimentally investigate this phenomenon. Figure 6.2 shows the average time needed to inform all nodes. We first discuss the results on preferential attachment graphs with $m = 2$ shown in Figure 6.2 (a). As expected, we observe a significant improvement between no exclusion and exclusion of one neighbor. In fact, for all graph sizes, a memory of size one yields a speed-up between between 14% and 21% faster compared to using no memory. Observing the curves for different graph sizes also suggests that we have a logarithmic broadcast time in the no-memory case and a sublogarithmic broadcast time for any memory size greater than one. We do observe additional but very small improvements if we increase



(a) pref. attachment ($m = 2$)

(b) random-attachment ($m = 2$)

(c) pref. attachment ($m = 10$)

(d) random-attachment ($m = 10$)

(e) complete graph

(f) hypercube

Figure 6.2: Comparison of synchronous rumor spreading without memory (marked with +), memory one (marked with −), and unbounded memory (marked with ×) on different graphs. The x-axis corresponds to the number of vertices $n = 2^5 \dots 2^{23}$. The y-axis corresponds to the runtime to inform all vertices, averaged over 10,000 runs.
The benefit of more than one memory is very limited for all graphs. The benefit of memory one compared to no memory is the largest for the sparse preferential and random-attachment graphs. The complete graph and hypercube benefit very little from additional memory.

	90% informed	**99% informed**	**100% informed**
memory=0	15.74±0.99	16.87±1.00	23.13±2.28
memory=1	15.51±0.98	16.60±1.00	20.97±1.59
memory=2	15.47±0.98	16.55±0.99	20.31±1.30
memory=3	15.45±0.98	16.54±0.99	20.18±1.22
memory=25	15.45±0.97	16.54±0.99	20.11±1.13

Table 6.1: Comparison of the average time needed to inform a certain fraction of the vertices on the Orkut network depending on the amount of memory. For each combination, the average and standard deviation of 100,000 runs is. With regard to the time needed to inform all vertices, we observe a large difference between excluding none and excluding the one most recently contacted. If only a 90% or 99% fraction should be informed, the gap is significantly smaller.

the memory to a size larger than the runtime, that is, when avoiding all previous contacts. For the considered graph sizes the improvement of unbounded memory compared to memory of size only one is around 2%. The advantage of memory for preferential attachment graphs gets smaller for larger m as shown in Figure 6.2 (c).

The results on random-attachment graphs are similar, just generally slower. Figure 6.2 (b) shows that the difference for $m = 2$ between no memory and memory of size one is between 10% and 13% while the additional improvement of unbounded memory is again around 2%. Theoretical consideration suggests that these gains can be at most by constant factors[1], and our experiments show that this can be at most a small constant.

In contrast, for other network topologies we see little advantage from using memory. For complete graphs, we observe in Figure 6.2 (e) barely any advantage even with unbounded memory. The difference between no memory and unbounded memory is less than 1% for complete graphs of all sizes. Because of the large vertex degrees, little benefit was expected; however, this is a notable difference from the results of using a pure push protocol without pull. Here, Doerr et al. [25] observed at least a small advantage for the quasirandom protocol, which, when used with random lists, is equivalent to random choices with previous contacts excluded. The results of Figure 6.2 (f) for hypercubes show a similarly small impact of memory. For graphs with more than a few thousand nodes the difference between no memory and unbounded memory is smaller than 2%.

The benefit of little memory can also be observed on real-world graphs. We examined the time needed to spread a rumor on a crawl of the Orkut

[1] It is known that these graphs have a diameter of $\Theta(\ln n)$ (see [60]), so this is a natural lower bound. On the other hand, with high probability, every pair of vertices is connected by a path such that the sum of the degrees of the vertices on the path is at most $O(\ln n)$. Consequently, with probability $1 - o(n^{-1})$, $O(\ln n)$ rounds suffice to transmit a rumor along such a path. This yields an upper bound of $O(\ln n)$ for the broadcast time on random attachment graphs.

network (for details on the network see Section 6.3). Table 6.1 shows a large difference between no memory and memory one for the time needed to inform all vertices. It is clearly visible that (a) more memory is of very little benefit and (b) this difference vanishes when considering the time needed to inform only a fraction of the vertices.

In summary, we also observe in experiments that a small memory helps a lot for preferential and random attachment graphs, but much less for classic network topologies like complete graphs and hypercubes.

6.3 Real-World Social Networks

Most previous statistics were based on mathematically defined graph models. To support our claim that news spreads very fast in social networks in general, we have also simulated the rumor spreading process on crawls of the *Twitter* and *Orkut* social networks.

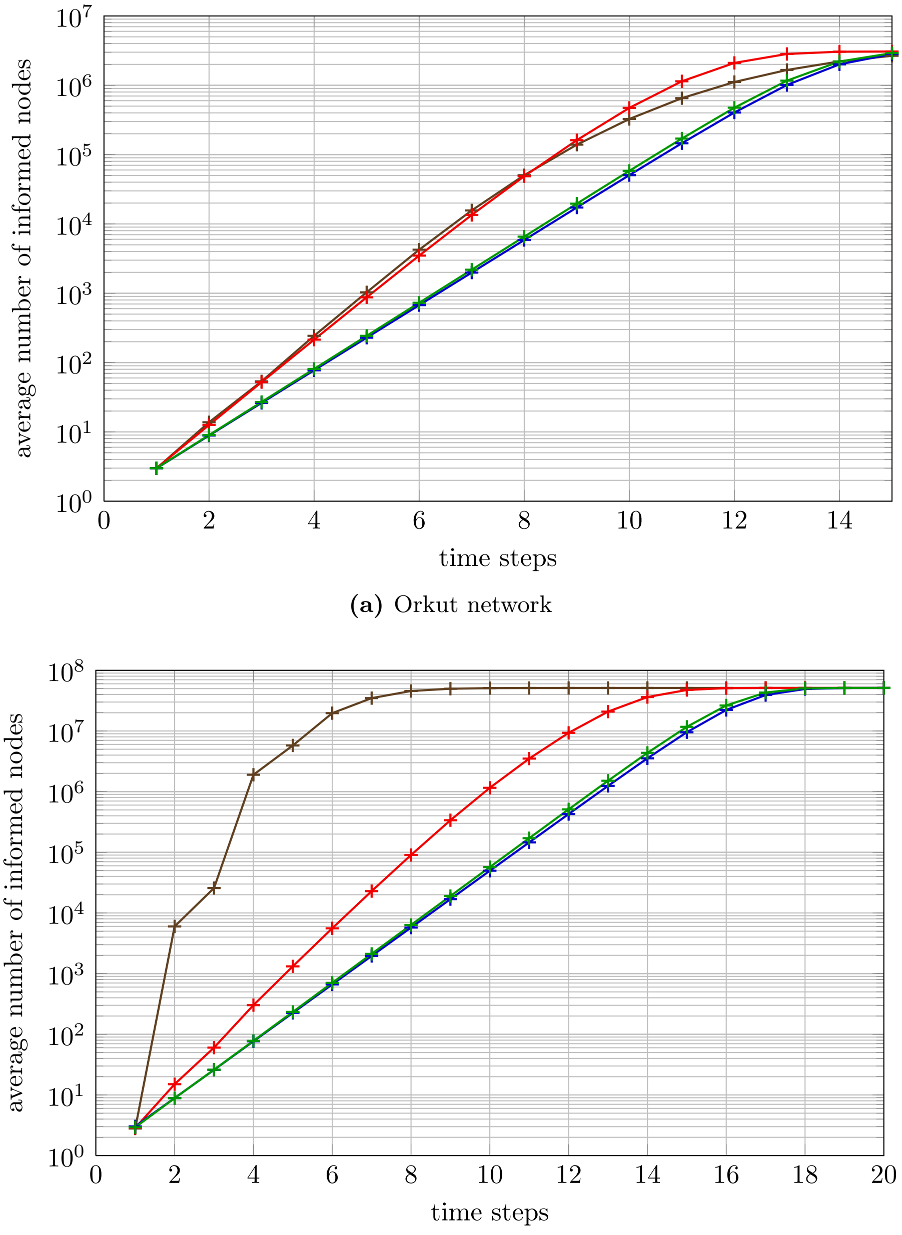
Twitter is a social networking site which allows users to send and read short messages (so-called "tweets") of up to 140 characters. It is currently one of the top ten most visited sites on the web[2]. We performed our experiments on a snapshot of the Twitter network that was crawled in September 2009 by Cha, Haddadi, Benevenuto, and Gummadi [16], available from [6]. It consists of 51,217,936 nodes and 1,963,263,821 directed edges. By making all edges undirected and considering the largest connected component, we obtained a connected graph with 51,161,011 nodes and 1,613,927,450 undirected edges.

Orkut is a social networking site operated by Google Inc. It is one of the top ten most visited websites in India and Brazil[2]. We used the data crawled in October and November 2006 by Mislove, Marcon, Gummadi, Druschel, and Bhattacharjee [57], which can be downloaded from [6]. The crawled graph contains 3,072,441 nodes and 117,185,083 edges. The edges are undirected, since Orkut requires consent from both users before a link between the two is created. At the time of the crawl, new users had to be invited by an existing user; therefore, the graph consists of a single component. The data covers roughly 11% of the total user population. The technical reason for this is that Orkut limits the rate at which a single IP address can download information. As a result, it took more than a month to crawl even this currently available part of the graph.

We chose these online social networks because of the available network data and because we feel that their structure might be similar to that of other real-world social networks. We are aware of the fact that interactions in Twitter and Orkut are more complex than in our simple randomized rumor spreading model.

We ran the protocol with memory one on these real-world graphs and, for comparison, in preferential attachment, random-attachment and complete

[2] See "Top 500 Sites on the web" at `www.alexa.com`.



(a) Orkut network

(b) Twitter network

Figure 6.3: Comparison of synchronous rumor spreading with memory one on two real networks (—) with preferential attachment graph (—), random-attachment graph (—), and complete graph (—) of same size and density (where applicable). The Orkut network in (a) has $n = 3,072,441$ vertices and density parameter $m = 38$, the Twitter network in (b) has $n = 51,217,936$ vertices and density parameter $m = 32$. The Orkut network behaves very similarly to the corresponding preferential attachment graph. The Twitter network is even faster than the corresponding preferential attachment graph. The complete and random-attachment graphs are significantly slower.

graphs with size and density as close as possible to the corresponding values of the real-world graph, that is, $m = 32$ for Twitter network and $m = 38$ for the Orkut network. The numbers shown in Figure 6.3 are averages over 500 runs[3] for the Twitter network and 100,000 runs for the Orkut network.

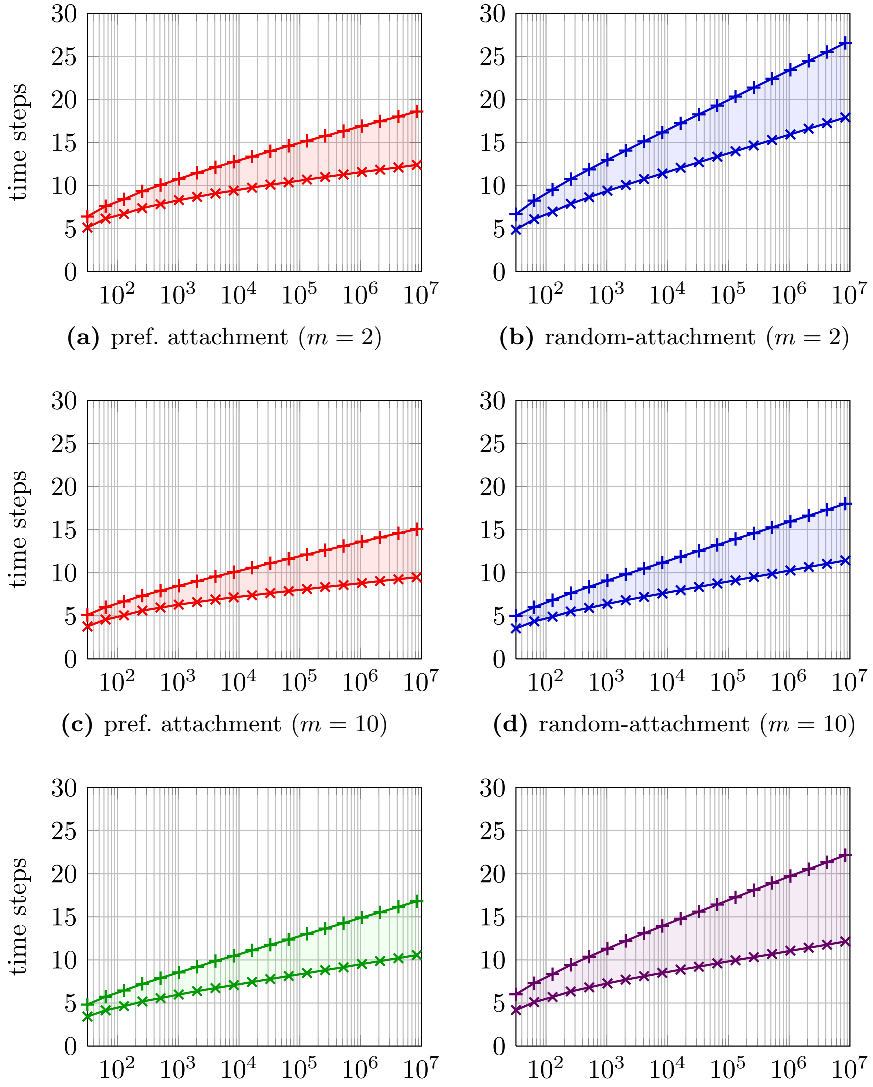
Figure 6.3 shows that news spreads much faster in the real-world networks and the preferential attachment graphs than in the complete and random-attachment graphs. Interestingly, rumor spreading in the Orkut network and the comparable preferential attachment graph proceeds very similarly, whereas the Twitter network leads to much faster rumor propagation.

6.4 Asynchronous Rumor Spreading

It is not surprising that asynchronous rumor spreading can be slow to inform *all* vertices. Note that it takes $\Theta(\ln n)$ time until every node has performed at least one action. For this reason, in Figure 6.4 we consider the time needed to inform 99% of the nodes. Note, however, that the time needed to inform 100% were also lower for the asynchronous model compared to the synchronous one. The charts clearly show a substantial speedup. Interestingly, for $n = 2^{20}$, the speedup for preferential and random-attachment graphs (47-48% for $m = 2$, 54-55% for $m = 10$) is smaller than for complete graphs (57%) and hypercubes (78%).

These empirical observations for moderately sized graphs do not fully comply with our theoretical findings. Remember that for the preferential attachment graph, we showed that the time to inform $n - o(n)$ vertices without memory decreases from $\Theta(\ln n)$ for the synchronous model without memory to $O(\sqrt{\ln n})$ for the corresponding asynchronous model. On the other hand, it has been argued that random-attachment graphs, complete graphs and hypercubes keep their $\Theta(\ln n)$ times, while our experiments show that the asynchronous model is faster on all graph classes. An asymptotic advantage for preferential attachment graphs is not apparent. We expect that the theoretically proven asymptotic behavior can be observed only for very large graphs. For the real-world social networks Orkut and Twitter, Figure 6.5 shows that, especially at the beginning, the asynchronous protocol performs much faster than its synchronous counterpart.

[3] The reason for the relatively small number of runs is that the Twitter network has more than one billion edges and we needed more than 50 GB of main memory to process it. A single simulation of the process required a runtime of several hours on a Hewlett Packard DL980 G7 server with eight eight-core Intel Xeon X7560 processors and 2048 GB of main memory.



(a) pref. attachment ($m = 2$)

(b) random-attachment ($m = 2$)

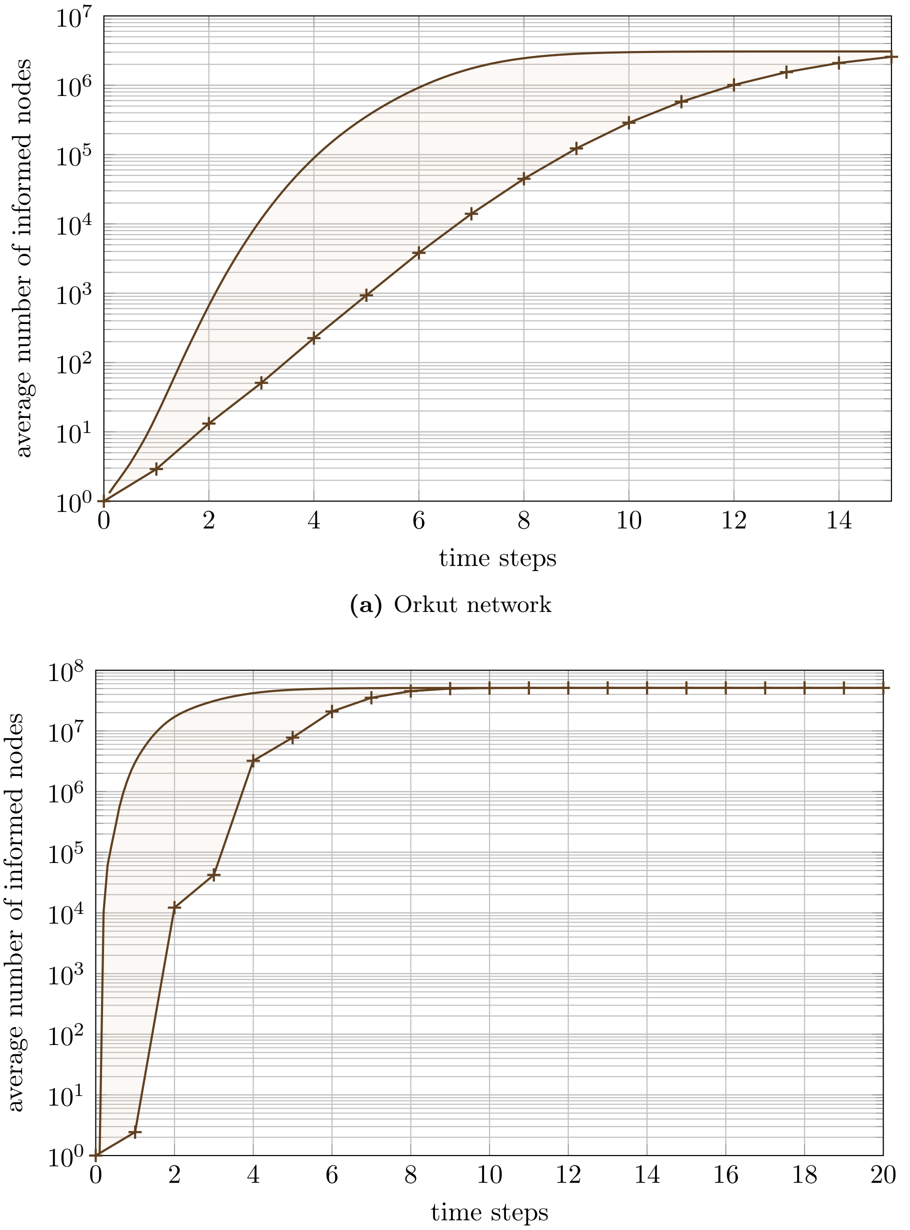
(c) pref. attachment ($m = 10$)

(d) random-attachment ($m = 10$)

(e) complete graph

(f) hypercube

Figure 6.4: Comparison of the average number of time steps needed to inform 99% of the vertices with synchronous (marked with +) and asynchronous (marked with ×) rumor spreading without memory on different graphs. The x-axis corresponds to the number of vertices $n = 2^5 \dots 2^{23}$. The y-axis corresponds to the runtime to inform 99% of the vertices, averaged over 10,000 runs.
The asynchronous protocol spreads information faster than the synchronous protocol on all graphs. The difference is of the same order of magnitude for all graphs.



(a) Orkut network

(b) Twitter network

Figure 6.5: Comparison of synchronous (—+—) and asynchronous (——) rumor spreading without memory on two real social networks. The x-axis corresponds to the time steps (in the synchronous setting) or the time (in the asynchronous setting). The y-axis corresponds to the number of informed vertices after this time, averaged over 1000 runs for the Orkut network and 50 runs for the Twitter network.
In both cases, the asynchronous counterparts spread the rumor significantly faster than the synchronous models.

Chapter 7

Conclusion

We simulated a natural rumor spreading process on different graphs representing real-world social networks and several classical network topologies. We also performed a mathematical analysis of this process in preferential attachment graphs. Both works demonstrate that rumor spreading is extremely fast in social networks.

A key observation in the mathematical proof and a good explanation for this phenomenon is that small-degree nodes quickly learn a rumor once one of their neighbors knows it, and then again quickly forward the news to all their neighbors. This in particular facilitates sending a rumor from one large-degree node to another.

What does this mean for our everyday life? It partially explains why social networks are observed to spread information extremely rapidly even though this process is not organized centrally and the network is not designed in some intelligent way. Crucial is the fruitful interaction between hubs, which have many connections, and average users with few friends. The hubs make the news available to a broad audience, whereas average users quickly convey the information from one neighbor to another.

Part II

Asymptotically Optimal Randomized Rumor Spreading in Complete Graphs

Chapter 8

Introduction

Suppose we want to inform all people in the town of a new rumor. We do not know the email addresses of all people, so we want to launch a telephone chain. How can we do that quickly and still be sure that at the end all people will have received a call? Formally, we are given a complete graph and want to spread a rumor from one source node to all nodes in the graph. We can use the simple push protocol: in each round, each informed node calls a random node and informs it of the rumor. It is well-known that this protocol succeeds to spread a rumor in $(1+o(1))(\log n+\ln n)$ rounds with high probability ([37, 43]). One problem with this protocol, however, is that a node can never be sure that the whole graph has been informed, so there is no safe termination rule. More seriously is the large number of $\Theta(n \log n)$ calls that are necessary. The later disadvantage was overcome in the seminal work by Karp et al. [49]. They present two variations of the randomized rumor spreading protocol which spread the rumor with $O(n \log \log n)$ messages only while still using $O(\log n)$ rounds. Their second protocol is also robust against node failures. A central ingredient in their approach are *pull* operations, which allow nodes not yet informed to call random nodes and ask for news. Pull operations, however, have the disadvantage that they create network traffic even if there is no news to be spread. Hence, the assumption underlying the model by Karp et al. [49] is that new rumors are constantly injected into the network.

Here, we present an alternative solution to the problem. It completely avoids the problematic pull operations. It achieves a broadcast time of $(1 + o(1)) \log n$ and it uses a total number $O(nf(n))$ calls, where $f = \omega(1)$ can be any function tending to infinity arbitrarily slow. This is arbitrarily close to the theoretically optimal values of $\lceil \log n \rceil$ rounds and $n - 1$ calls. Still the protocol is very simple; every node follows the same (randomized) process. Due to its randomized nature, we still have a good robustness. If a constant fraction of the nodes chosen uniformly at random crashes at arbitrary times, the time needed to inform all properly working nodes comes still arbitrarily close to the theoretical optimum. In case of adversarial node failures, we can adjust the protocol to make sure that it still performs well. Our protocol is also scalable. If the network size changes, no significant modifications are necessary. Finally, it has a simple termination rule that ensures in the error-free case that all nodes are informed at the end of the protocol with probability 1.

The only point in which we assume the protocol to be more powerful compared to previous works is that we discard the address-obliviousness. That is, we assume that each node has a unique label chosen arbitrarily from some ordered set (e.g., the integers). This seems to be reasonable in many settings.

8.1 The Protocol by Karp et al.

As described above, Karp et al. [49] showed how to modify the simple randomized rumor spreading protocol such that instead of $\Theta(n \log n)$ messages only $O(n \log\log n)$ are sufficient to spread a rumor. Roughly speaking, their protocol proceeds as follows. The rumor is equipped with a time stamp (or age counter) in such a way that all nodes that receive the rumor also know for how many rounds it has been in the network. In each round, each node chooses a random other node as a communication partner. The communication then proceeds in both directions, that is, any partner who knows the rumor forwards it to the other partner. In particular, it is shown that after only $\log_3 n + \Theta(\log\log n)$ rounds of this protocol, all nodes know the rumor with high probability. In addition, a rumor is transmitted in this time interval at most $O(n \log\log n)$ times.

Note that this way of counting ignores all communication effort which does not result in a rumor to be sent, i.e., all calls between two uninformed nodes that arise due to pull operations are not counted. The way this is usually justified is by assuming that there is sufficient traffic in the network due to regular insertions of new rumors. Still, we feel that this is slightly dissatisfying. Note that when using pull operations, there is no way to avoid such communication overhead—a node that did not receive a rumor recently has no way of finding out whether there are rumors around that justify starting pull operations or not. Even nodes that did receive a rumor recently cannot be sure that there is no new rumor that would justify starting pull operations again.

Karp et al. [49] further extend the protocol just sketched to improve its robustness. The one above greatly relies on a very precise estimate of the time when to stop sending on the rumor (here, $\log_3 n + \Theta(\log\log n)$). With transmission failures present, the initial phase of exponential growth might take longer. If this time span is not correctly guessed by the protocol, either nodes remain uninformed, or for too long a time a linear number of nodes keep sending out the rumor, leading to too many messages sent.

This problem is overcome by a clever median-counting trick. Here, very roughly speaking, nodes average their estimation on how well-known the rumor is with the estimations of their communication partners. This allows the following robustness result. An adversary may specify a set $\mathcal{F}$ of nodes together with arbitrary times at which each of them drops out of the game. Nevertheless, within $O(\log n)$ rounds and using $O(n \log\log n)$ messages, all but $O(|\mathcal{F}|)$ nodes are informed. Note that this does allow that up to $O(|\mathcal{F}|)$ properly working nodes remain uninformed.

Karp et al. [49] also prove lower bounds, which show that if in each round all communication is restricted to random matchings of communication partners (i) any address-oblivious algorithm has to make $\Omega(n \log\log n)$ calls and (ii) that any algorithm informing all but a $o(1)$ fraction of the vertices in logarithmic time has to make $\omega(n)$ calls.

8.2 Our Results

The first lower bound stated in the previous paragraph suggests that asking for an address-oblivious protocol may result in only a limited performance being achievable. In addition, one might also wonder if really many broadcasting problems ask for address-oblivious protocols, or if not rather in the majority of settings each participant naturally has a unique address, simply to organize the transport of a message to an addressee.

In this work, we shall drop the requirement of address-obliviousness. However, we shall keep the concept of contacting random neighbors without any preference, as this seems to be the key to obtaining good broadcasting times, robustness and scalability in all previous works.

Contrary to the model by Karp et al. [49], we do not perform pull operations; all transmissions are initiated by nodes that know the rumor. So the initiator of a transmission is always the informed node, which chooses its addressee uniformly at random, but not always independently.

We do allow, however, two-way communication, in that the addressee acknowledges his readiness to receive the rumor or his knowledge of the rumor. Such a mechanism makes sense anyway, because it allows to reduce the amount of data sent through the network (if the addressee cannot receive the rumor or already knows it, we do not need to send it). In practice, most communication protocols (e.g., the standard network protocol FTP) allow some kind of two-way communication to ensure an error-free transmission.

In this, as we think, natural setting, we propose a protocol that informs all nodes in only $(1 + o(1)) \log n$ rounds while using only $nf(n)$ calls, where $f = \omega(1)$ can be chosen arbitrarily. Note that every protocol that only uses push operations needs at least $\lceil \log n \rceil$ rounds and makes at least $n - 1$ calls.

More precisely, we have the following tradeoff between rounds and messages. For all $f : \mathbb{N} \to \mathbb{N}$, we give a protocol that needs only

$$\log(n) + f(n) + O(f(n)^{-1} \log n)$$

rounds with high probability and $O(nf(n))$ calls. In terms of running time, this is optimal for $f(n) = \Theta(\sqrt{\log n})$, leading to $\log n + O(\sqrt{\log n})$ rounds and $O(n\sqrt{\log n})$ calls.

The protocol in its basic version is very simple. For the presentation, let us assume that the nodes are numbered from 1 to n, even though what we really need is only that nodes are able (i) to compute the label of a node chosen uniformly at random and (ii) given a label of a node, to compute a uniquely defined successor along a cyclic order of the labels (label plus one, modulo n).

Let $f : \mathbb{N} \to \mathbb{N}$ be given (to formulate the tradeoff scenario). Then the basic protocol works as follows. Each newly informed node sends its first message to another node chosen uniformly at random. From then on, it does the following. If the previous message was sent to a node that was not informed yet, then the next message is sent to the successor of that node in the cyclic order. Otherwise, the next message is sent again to a node chosen uniformly at random. After having encountered $f(n)$ nodes that were already informed, the node stops and does not transmit the rumor anymore. This protocol can be interpreted as a variant of the quasirandom rumor spreading protocol investigated in [24, 26]. In contrast to the latter, all nodes have the same cyclic permutation and can re-start at a random position when they call a node that is already informed.

Despite its simplicity, this protocol is robust against random node failures. When a randomly chosen fraction of $1-p$ of all nodes fails at arbitrary times, where $p \in (0, 1]$, we still have a running time of

$$(1 + o(1)) \log_{1+p} n$$

with high probability. It is easy to see that $\lceil \log_{1+p} n \rceil$ is a lower bound on the expected number of rounds needed to distribute the rumor. Thus, we achieve an asymptotically optimal running time even under the presence of *random* node failures.

Similar to the basic protocol by Karp et al. [49], however, this protocol is not very robust against *adversarial* node failures. If an adversary chooses a large consecutive segment of the nodes to be out of order (say ℓ nodes), then there is a reasonable chance (of $\frac{\ell}{2n}$) that the first transmission ever sent hits the first half of this 'failed' segment, and no progress is made for the next $\ell/2$ rounds.

We develop a number of enhancements to cope with such problems. Together, they yield the following. For a given security parameter $p \in (0, 1]$, we have a protocol that is robust against adversarial node destruction of up to $(1-p)n$ nodes, i.e., an adversary may destruct arbitrary nodes (excluding the initially informed node) at the beginning of each round such that in total up to $(1-p)n$ nodes are destructed. Such failed nodes do not answer calls directed to them, nor do callers get a feedback if other nodes tried to call a failed node before (this is what causes most of the difficulties). In spite of this strong adversarial setting, we do inform all non-destructed nodes in

$$(1+\varepsilon)(\log_{1+p} n + \tfrac{1-p}{p} \ln n)$$

rounds for any constant $\varepsilon > 0$, while using $O(n)$ calls to properly-working nodes. The main difficulty in designing such a protocol lies in balancing out the following two effects: on the one hand, a node should not fall back to sending a random message after encountering failed nodes too early, as this would destroy the advantage of following the given order of nodes. On the other hand, in order to be able to cope with long segments of failed nodes, in particular in the early stages of the protocol, such random restarts are necessary.

The main technical difficulty in the analysis of the proposed protocols is that the transmission of messages at each node is not independent, and thus, many classical tools cannot be employed. The key to the solution here is to exploit the existing independence stemming from communications started with random partners.

In summary, we show that considerable improvements over the fully independent rumor spreading protocol are possible if we do not require the protocol to be address-oblivious. It is thus worth questioning whether this assumption is really needed in previous applications of the protocol. From the methodological side, our results again show that spicing up randomized algorithms with well-chosen dependencies can yield additional gains. The theoretical analysis might become more complicated, but not so much the algorithm itself.

To come back to our motivating example with the telephone chain at the beginning, our protocol suggests the following simple solution. We look up our number in the phone directory and call the subsequent number. After informing the called person, we choose a random number in the phone directory and call it. If every called person places two such calls, we inform all nodes after $(1+o(1))(\log n + \ln n)$ rounds with high probability. Note that this solution is slightly different than our protocol since it first makes a non-random call and then places a random call, whereas in our protocol we switch the order.

8.3 Preliminaries

For the upper bounds, we make some simplifying assumptions that can only slow down the running time of the protocol and help us to cope with the dependencies between different nodes. In particular, we will assume that certain vertices stop informing (*ignoring*), and that other vertices do not immediately start their own informing process after becoming informed (*delaying*). Delaying turns out to be useful as it gives us some influence on when a node uses its first random choice. Nodes that have been informed but have not yet begun informing new nodes play an important role in our analysis. We will call them *newly informed* vertices.

The following fact holds for all protocols that we consider.

Fact 8.3.1. *If a node is* delayed*, i.e., halted for a number of rounds, then the protocol can only become slower.*

To see why this holds, we can use the following argument for our protocols. Fix for each node the set of random addressees. Then, a simple induction over time shows that for any set of delays the following holds. No vertex in the delayed model is informed earlier than in the original model. Since this is true for any choice of the random addressees, the fact follows. This allows us to delay a node for any number of rounds in our analysis for the upper bounds.

Chapter 9

Quasirandom Protocol with Restarts

Let $G = (V, E)$ be the complete, undirected graph on n nodes. We assume that the nodes of the complete graph are ordered and denote by i the i-th node according to that order. Our goal is to spread a rumor known initially to one node to all nodes in V. We call the node initiating the rumor the *starting node*. A rumor can be transmitted along each edge of the graph in both directions. Every transmission is always initiated by a node that knows the rumor. We count every contact of a node to another node as a *call*. For simplicity, we assume that two nodes never call a node exactly simultaneously even if they both call the same node in the same round; thus, a node is only informed by a single node.

We introduce a simple algorithm that for a certain instantiation achieves, up to lower order terms, an optimal running time for a push protocol. The algorithm is related to the quasirandom protocol by Doerr et al. [24]. There, every node v is equipped with a cyclic permutation $\pi_v : V \to V$ of all nodes in V. Once a node v becomes informed, it chooses one position on its list uniformly at random and contacts the corresponding node in the next round. In each following round, v proceeds according to its permutation π_v, i.e, if v contacted u in the previous round, it now contacts $\pi_v(u)$. Note that different nodes can have different permutations.

Our protocol differs from this quasirandom protocol in two main aspects. First, the permutations of all nodes are identical. Second, we introduce the notion of a *restart*: if a node calls an already informed node, it chooses a *random* communication partner in the next round instead of choosing the next one according to the permutation. Each node performs R such restarts, where R is a parameter of the protocol that can be a function of n, and then terminates its rumor spreading. Thus, once a node has called $R+1$ informed nodes, it stops. This rule allows us to bound the total number of calls made. Note that the aspect of keeping the number of calls small was not discussed in [24].

A detailed description of the protocol is given in Algorithm 1, where we denote by $j + 1$ the successor of j according to the cyclic order.

There is a small technicality here. The starting node is treated as if it is not informed at the beginning, i.e., the first node that calls the starting node will send the rumor to it and only after this call, the starting node will be treated as an informed node.

Algorithm 1: Procedure started by newly informed node

```
let $R \in \mathbb{Z}^+$ be the number of random calls per node
for $i = 1$ to $R$ do
    select node $j$ uniformly at random;
    while $j$ not informed // iteration counts as call even if $j$
        informed
    do
        inform $j$;
        $j \leftarrow j + 1$;
```

9.1 Running Time And Number of Calls

We give an upper bound and an almost matching lower bound on the number of rounds and calls needed by the protocol to spread a rumor from an arbitrary starting node to all nodes of the complete graph with high probability.

Theorem 9.1.1. *Let $\varepsilon > 0$ be an arbitrarily small constant. With probability $1 - o(1)$, the protocol with R random calls per node informs all nodes in*

$$\begin{array}{ll} \log n + (1+\varepsilon)\ln(n)/R + R + h(n), & \text{if } R \leq \sqrt{\ln n} \\ \log n + (2+\varepsilon)\sqrt{\ln n}, & \text{if } R \geq \sqrt{\ln n} \end{array}$$

rounds and $n(R+1)$ calls, where $h(n)$ is a function of arbitrarily slow growth.

Note that by adjusting the stopping parameter R, we get a tradeoff between the number of rounds needed to inform all nodes and the number of calls.

Before analyzing the protocol for general R, we describe two special cases that achieve an (almost) optimal number of rounds and calls, respectively. For $R = \sqrt{\ln n}$, we achieve, up to a lower order term, an optimal running time while using only $O(n\sqrt{\ln n})$ calls. For $R = 1$, we get a very simple broadcasting protocol that, up to constant factors, is both optimal in terms of rounds needed as well as the number of calls.

Corollary 9.1.2. *Let $\varepsilon > 0$ be an arbitrarily small constant. With probability $1 - o(1)$, the protocol with*

- *$R = \sqrt{\ln n}$ informs all nodes in $\log(n) + (2 + \varepsilon)\sqrt{\ln n}$ rounds using $2(1 + 2\varepsilon)n\sqrt{\ln n}$ calls,*
- *$R = 1$ informs all nodes in $\log(n) + (1 + \varepsilon) \ln n$ rounds using $2n$ calls.*

Before we prove Theorem 9.1.1, we make the following observation for any $R \geq 1$.

Fact 9.1.3. *The protocol is always at least as fast as the quasirandom model implemented with identical lists.*

This observation follows from the fact that every node acts as in the quasirandom model until it encounters an informed node. In this case, since we assumed all lists to be the same, the node becomes useless in the quasirandom model as all successive nodes on its list will have also been informed once it tries to call them. In our protocol, however, the node might still call uninformed nodes.

Proof of Theorem 9.1.1. We distinguish three phases of the process.

The first phase lasts for $\log n + h(n)$ rounds where $h(n)$ is a function that is growing arbitrarily slowly. Using Fact 8.3.1, we assume that every node is delayed to the second phase once it contacts an informed node. Note that this delayed protocol remains at least as fast as the protocol with $R = 1$ and thus, by Fact 9.1.3, also at least as fast as the quasirandom model implemented with identical lists. Fountoulakis and Huber [40] showed that for any arbitrarily small constant $\varepsilon > 0$ the quasirandom model informs $(1 - \varepsilon)n$ nodes with probability $1 - o(1)$ in this phase. Thus, we get the same result for our delayed protocol.

The second phase lasts for R rounds. By our delaying assumption, every node that is informed in the first phase will remain active for at least $R - 1$ rounds before the second phase ends. The crucial observation is that, in each round, each informed round either makes a random call or it must have called an uninformed node in the previous round. The latter happens at most εn times in total. We conclude that at the end of the second phase the number of random calls made is at least $(1 - \varepsilon)nR - \varepsilon n \geq (1 - 2\varepsilon)nR$ (including the random calls made in the first phase). We use this to bound the largest interval of uninformed nodes by $(1 + 3\varepsilon) \ln(n)/R$.

Let I be an interval of length $(1 + 3\varepsilon) \ln(n)/R$. Then, the probability that no node in I becomes informed in the second phase by these random calls is at most

$$\begin{aligned}\left(1 - \tfrac{(1+3\varepsilon)\ln n}{nR}\right)^{(1-2\varepsilon)nR} &\leq \exp\left(-(1 - 2\varepsilon)(1 + 3\varepsilon) \ln n\right)\\ &= n^{-1-\varepsilon+6\varepsilon^2} = n^{-1-\varepsilon'},\end{aligned}$$

for some constant $\varepsilon' > 0$ (when ε is sufficiently small). By a union bound argument, we conclude that there is no completely uninformed interval of length $(1+3\varepsilon)\ln(n)/R$ after the second phase with probability at least $1 - n^{-\varepsilon'}$ for some constant $\varepsilon' > 0$.

In the last phase, all the remaining uninformed intervals are 'processed'. This takes at most the length of the largest uninformed interval, which is at most $(1+3\varepsilon)\ln(n)/R$. Note that here we exploit our assumption that the starting node is treated as if it is not informed in the beginning. Otherwise, there could be an uninformed interval on the cyclic list after the starting node that is not further processed, since the starting node 'blocks' any node from processing this interval.

Using a simple union bound, we bound the total failure probability by $o(1)$.

It remains to bound the number of calls. Note that each node calls at most R informed nodes in total. Hence, we use n calls to inform all nodes and, in addition, at most nR calls until all nodes stop informing.

□

We can also show that the upper bound is essentially sharp.

Theorem 9.1.4. *Let $\varepsilon > 0$. If the protocol with R random calls per node is run for less than*

$$\begin{aligned} \log(n) + (1-\varepsilon)\ln(n)/R + \tfrac{1}{2}R, &\qquad \text{if } R \le \sqrt{2(1-\varepsilon)\ln n}, \\ \log(n) + \sqrt{2(1-\varepsilon)\ln n}, &\qquad \text{if } R \ge \sqrt{2(1-\varepsilon)\ln n} \end{aligned}$$

rounds, then with probability $1 - \exp(-n^{\Theta(\varepsilon)})$ not all nodes are informed.

Proof. W.l.o.g. we assume that n is a power of two. Otherwise, we can use the largest power of two smaller than n instead.

We first consider the case $R \le \sqrt{2(1-\varepsilon)\ln n}$. Let $T = \log(n) + (1-\varepsilon)\ln(n)/R + \frac{1}{2}R$. We bound the probability from below that a specific node u becomes informed within T by $n^{-1+\varepsilon}$. A simple union bound then shows that if we run the protocol for less than T rounds, there will be an uninformed node with high probability.

We call the event that a node chooses another node to inform uniformly at random a *random call*. Hence, random calls occur as first calls of a node after the node encountered an already informed node. Clearly, u remains uninformed if for all $i \le T$ all random calls happening at time $T - i$ avoid u and the i vertices to the left of it. We say that u is *unaffected* by such a random call.

The probability of informing u within T rounds only increases if we assume that the random calls are made as early as possible. We will therefore assume that a node that is informed in round k, makes its R random calls in rounds $k+1, k+2, \ldots, k+R$. Furthermore, note that in the i-th round at most 2^{i-1} nodes start calling. The probability that the j-th call of a node informed in round $i \leq \log n$ does not affect u is $1 - \frac{T-i-j}{n}$. Note that $T - i - j \geq 0$ by the choice of R and i.

Using $1 - x \geq e^{-x-x^2}$ for $x \leq \frac{1}{2}$, and the fact that all random transmissions of the starting node are independent from each other, the probability that u is not informed in T rounds is at least

$$\begin{aligned}
\prod_{i=1}^{\log(n)} \prod_{j=0}^{R-1} \left(1 - \tfrac{T-i-j}{n}\right)^{2^{i-1}} &\geq \prod_{i=1}^{\log(n)} \prod_{j=0}^{R-1} \exp\left(-2^{i-1}\tfrac{T-i-j}{n} - 2^{i-1}\left(\tfrac{T-i-j}{n}\right)^2\right) \\
&\geq \prod_{i=1}^{\log(n)} \prod_{j=0}^{R-1} \exp\left(-2^{i-1}\tfrac{T-i-j}{n}\right) \exp\left(-\tfrac{T^2}{n}\right) \\
&\geq \underbrace{\exp\Big(-\sum_{i=1}^{\log(n)} \sum_{j=0}^{R-1} 2^{i-1}\tfrac{T-i-j}{n}\Big)}_{X_1} \underbrace{\exp\Big(-\tfrac{\log(n)RT^2}{n}\Big)}_{X_2},
\end{aligned}$$

Note that for $T \in O(\log n)$, we have $X_2 = 1 - o(1)$. For X_1, we first simplify the sum in the exponent.

$$\begin{aligned}
\sum_{i=1}^{\log n} \sum_{j=0}^{R-1} 2^{i-1}\tfrac{T-i-j}{n} &= \tfrac{1}{n} \sum_{i=1}^{\log n} 2^{i-1} \sum_{j=0}^{R-1} (T - i - j) \\
&= \tfrac{1}{n} \sum_{i=1}^{\log n} 2^{i-1}(R(T-i) - R(R-1)/2) \\
&= \tfrac{R}{n}\Big((n-1)(T-(R-1)/2) - \sum_{i=1}^{\log n} i2^{i-1}\Big) \\
&= \tfrac{R}{n}\Big((n-1)(T-(R-1)/2) - 1 - (\log n - 1)n\Big) \\
&\leq R\big(T - (R-1)/2 - \log n + 1\big) \\
&= R\big(\log n + (1-\varepsilon)\ln(n)/R \\
&\quad + \tfrac{1}{2}R - (R-1)/2 - \log n + 1\big) \\
&\leq (1-\varepsilon)\ln n + 3R/2 \\
&\leq (1-\varepsilon/2)\ln n,
\end{aligned}$$

where the last inequality holds by our assumption $R \leq \sqrt{2(1-\varepsilon)\ln n}$ and n large enough. Thus, the probability that u is not informed within T rounds is at least

$$(1 - o(1))\exp(-(1-\varepsilon/2)\ln(n)) = n^{-1+\Theta(\varepsilon)}.$$

Now let $k = \lfloor \frac{n}{T+1} \rfloor - 1$. Let $u_1, \ldots, u_k$ be nodes each having distance more than T from each other (in the cyclic order). We argue that, with sufficiently high probability, one such node will remain uninformed after the first T rounds. Let U_i denote the event that node u_i is informed. Note that since these nodes have a distance of T from each other, a random call that informs one such node u_i can not lead to the informing of any other node u_j during the first T rounds. Hence, these events are *negatively correlated*: if some nodes are informed, the probability that another one is also informed decreases, or formally, for any subset $S \subseteq \{1, \ldots, k\}$, and $j \notin S$, we have $\mathbb{P}(U_j \mid \bigwedge_{i \in S} U_i) \leq \mathbb{P}(U_j)$. We compute

$$\begin{aligned}
\mathbb{P}(\text{no node remains uninformed}) &\leq \mathbb{P}(U_1 \wedge \cdots \wedge U_k) \\
&\leq \prod_{1 \leq j \leq k} \mathbb{P}(U_j) \\
&\leq (1 - n^{-1+\Theta(\varepsilon)})^k \\
&\leq \exp(-n^{\Theta(\varepsilon)}).
\end{aligned}$$

So far we have considered the case $R \leq \sqrt{2(1-\varepsilon)\ln n}$. For larger R, inequality (9.2.1) no longer holds since the factors in the product can become larger than 1. Instead we assume that for larger R, every node keeps making random calls in every round till the end of the protocol. Clearly, this assumption only makes the protocol faster. Thus, we have

$$\mathbb{P}(u \text{ is not informed in } T \text{ rounds}) \geq \prod_{i=1}^{\log(n)} \prod_{j=0}^{T-i} \left(1 - \tfrac{T-i-j}{n}\right)^{2^{i-1}}.$$

By a similar argument as before, we can bound this probability by analyzing the following sum, where $T := \log(n) + \sqrt{2(1-\varepsilon)\ln n}$.

$$\begin{aligned}
\sum_{i=1}^{\log n}\sum_{j=0}^{T-i} 2^{i-1}\tfrac{T-i-j}{n} &= \tfrac{1}{2n}\sum_{i=1}^{\log n} 2^i \sum_{j=0}^{T-i}(T-i-j) \\
&= \tfrac{1}{2n}\sum_{i=1}^{\log n} 2^i \sum_{j=0}^{T-i} j \\
&= \tfrac{1}{2n}\sum_{i=1}^{\log n} 2^i (T-i)(T-i+1)/2 \\
&= \tfrac{1}{4n}\sum_{i=1}^{\log n} 2^i (T^2+T-(2T+1)i+i^2) \\
&= \tfrac{1}{4n}\Big((T^2+T)2(n-1)-(2T+1)\sum_{i=1}^{\log n} i2^i + \sum_{i=1}^{\log n} i^2 2^i\Big) \\
&= \tfrac{1}{4n}\big((T^2+T)2(n-1)-(2T+1)(2+(\log n-1)2n) \\
&\qquad -6+(-2\log n+3+\log^2 n)2n\big) \\
&\le \tfrac{1}{2}\big(T^2+T-(2T+1)(\log n-1)-2\log n+3+\log^2 n\big) \\
&= \tfrac{1}{2}\big(T^2+3T-2T\log n-3\log n+4+\log^2 n\big) \\
&= \tfrac{1}{2}\big(T^2-T(2\log n-3)-3\log n+4+\log^2 n\big) \\
&= \tfrac{1}{2}\big((T-\tfrac{2\log n-3}{2})^2-(\tfrac{2\log n-3}{2})^2-3\log n+4+\log^2 n\big) \\
&= \tfrac{1}{2}\big((T-\tfrac{2\log n-3}{2})^2 \\
&\qquad -\tfrac{4\log^2 n-12\log n+9}{4}-3\log n+4+\log^2 n\big) \\
&= \tfrac{1}{2}\big((T-\log n+\tfrac{3}{2})^2+\tfrac{7}{4}\big) \\
&= \tfrac{1}{2}\big((\sqrt{2(1-\varepsilon)\ln n}+\tfrac{3}{2})^2+\tfrac{7}{4}\big) \\
&\le (1-\varepsilon/2)\ln n.
\end{aligned}$$

Using the same argument then as in the first case, the result follows. □

9.2 Robustness Against Random Node Failures

Despite its simplicity, our protocol offers reasonable robustness. We consider the following natural node failure model: each node apart from the starting node independently sampled with some constant probability $p \in (0,1]$ works properly. Nodes that do not work properly are called *failed* nodes. These nodes may stop answering calls or sending out messages at arbitrary times (specified by an adversary). If a node contacts one that has stopped working, it does not get a feedback and continues with the successor of the failed node in the next round (hence a failed node does not pretend to be informed). Not surprisingly, we cannot hope to achieve robustness in such a situation without any sacrifices. For example, when we are confronted with a linear number of randomly distributed failed nodes, it is unreasonable to assume that it is possible to inform all properly working nodes in $(1+o(1))\log n$ rounds. It is easy to see that, with high probability, any push algorithm needs at least $(1-o(1))\log_{1+p} n$ rounds to inform all properly working nodes, as the following lemma shows.

Lemma 9.2.1. *Let $p > 0$ be a constant. Any push protocol needs at least $(1-o(1))\log_{1+p} n$ rounds with high probability to inform all properly working nodes.*

Proof. We consider the following equivalent failure model. Whenever a node is contacted for the first time, we consider it to be a properly working node with probability p, otherwise it is failed node. By Chernoff's bound, we have at least $(1-o(1))pn$ properly working nodes with high probability. Assume that this is the case.

Let n_i denote the number of nodes that were informed in round i, and $N_i = \sum_{j\leq i} n_j$ the number of nodes informed within the first i rounds. At best, in one round, all informed nodes contact nodes that have not been called before. Since each of these nodes is properly working independently with probability p, we have

$$\mathbb{E}[n_i \mid N_{i-1}] \leq pN_{i-1}.$$

Let $\varepsilon := 1/\log n$ and let i be the first round such that $N_i \geq 2\varepsilon^{-2}p^{-1}\ln n$. Then, by Chernoff's bound, we have for every $j > i$,

$$\mathbb{P}[n_j \geq (1+\varepsilon)pN_{j-1} \mid N_{j-1}] \leq \exp(-\varepsilon^2 pN_{j-1}) \leq n^{-2}. \qquad (9.2.1)$$

So, by a simple union bound, with probability $1-(\log n)^2 n^{-2} \geq 1-n^{-3/2}$, we can assume for each round $j \in [i, i+(\log n)^2]$, that $n_j \leq (1+\varepsilon)pN_{j-1}$. Thus, we need at least $\log_{1+(1+\varepsilon)p}((1-\varepsilon)pn) = (1-o(1))\log_{1+p} n$ rounds to inform all properly working nodes. □

It turns out that in this case the quasirandom protocol with restarts can be instantiated to match this lower bound for push algorithms up to lower order terms.

Lemma 9.2.2. *Let $p > 0$ be a constant. If nodes fail independently with probability $1-p \in (0,1]$, then all properly working nodes are informed in*

$$\begin{aligned} (1+o(1))(\log_{1+p} n + \tfrac{1}{Rp^2}\ln n) + R, \qquad & \text{if } R < \tfrac{1}{p}\sqrt{\ln n}, \\ (1+o(1))(\log_{1+p} n + \tfrac{2}{p}\sqrt{\ln n}), \qquad & \text{if } R \geq \tfrac{1}{p}\sqrt{\ln n} \end{aligned}$$

rounds, with probability $1-o(1)$. In expectation, $(R+1)n/p$ calls are placed.

Proof. The proof resembles the one for Theorem 9.1.1, but the analysis of each phase is more complicated. In the following we assume that all node failures occur at the beginning of the protocol. Note that this assumption can only slow down the protocol since every call to such a node goes wasted. Let $\varepsilon > 0$ be an arbitrarily small constant. Let $\mathcal{W}$ denote the set of properly working nodes. In the following, we assume that

$$|\mathcal{W}| \geq (1-\varepsilon)pn, \tag{9.2.2}$$

which, by Chernoff's bound, holds with probability $1 - e^{-\Omega(n)}$.

The first phase lasts for $(1+\varepsilon)\log_{1+p} n + 1$ rounds. As in the proof for Theorem 9.1.1, we delay every node that contacts an already informed node to the next phase. Let $\mathcal{W}^1 \subseteq \mathcal{W}$ denote the subset of the properly working nodes informed in the first $(1+\varepsilon)\log_{1+p} n$ rounds. The additional round at the end of the first phase only makes sure that every node in $\mathcal{W}^1$ has placed exactly one random call before the second phase begins.

We use the fact that our protocol is at least as fast as the quasirandom protocol *with random node failures* implemented with identical permutations. In Lemma 9.2.3, we prove that the latter protocol informs a fraction of $1-\varepsilon$ of all properly working nodes in $(1+\varepsilon)\log_{1+p} n$ rounds with probabililty $1 - o(1)$. Hence, we have, with probabililty $1 - o(1)$,

$$|\mathcal{W}^1| \geq (1-\varepsilon)|\mathcal{W}|. \tag{9.2.3}$$

The second phase lasts for $R - 1$ rounds. We bound from below the total number of random calls that have been made by the informed nodes at the end of the second phase. By our delaying assumption, every node that is informed in the first phase will remain active for at least $R-1$ rounds before the second phase ends. Number the nodes in $\mathcal{W}^1$ from 1 to $|\mathcal{W}^1|$. Let W_i denote the set of nodes contacted by the i-th node in this ordering. We define the function f such that

$$f\bigl(W_1, \ldots, W_{|\mathcal{W}^1|}\bigr) := \sum_{i \in \mathcal{W}^1} |W_i \cap \mathcal{W}|.$$

For convenience, we write $Y := f\bigl(W_1, \ldots, W_{|\mathcal{W}^1|}\bigr)$. Note that Y is the number of properly working nodes contacted in this phase. By linearity of expectation we have $\mathbb{E}[Y] \geq (R-1)p|\mathcal{W}^1|$. Furthermore, we see that for any i and any possible set of values $w_1, \ldots, w_{|\mathcal{W}^1|}$ and w_i',

$$f\bigl(w_1, \ldots, w_i, \ldots, w_{|\mathcal{W}^1|}\bigr) - f\bigl(w_1, \ldots, w_i', \ldots, w_{|\mathcal{W}^1|}\bigr) \leq R.$$

Thus, we can apply Azuma's inequality (Lemma 1.2.7), and get

$$\begin{aligned}\mathbb{P}(Y \leq (1-\varepsilon')(R-1)p|\mathcal{W}^1|) &\leq \mathbb{P}(Y \leq \mathbb{E}[Y] - \varepsilon'(R-1)p|\mathcal{W}^1|)\\ &\leq 2\exp\Big(-\frac{(\varepsilon'(R-1)p|\mathcal{W}^1|)^2}{|\mathcal{W}^1|R^2}\Big)\\ &\leq 2\exp(-\Omega(|\mathcal{W}^1|))\\ &\leq \exp(-\Omega(n)),\end{aligned}$$

where the last inequality follows from (9.2.2) and (9.2.3).

At most $\varepsilon|\mathcal{W}|$ of all calls made by nodes in $\mathcal{W}^1$ go to an uninformed node in the second phase. So by the same argument as in the proof of Theorem 9.1.1 we can bound the number of random calls from below by $|\mathcal{W}^1| + Y - \varepsilon|\mathcal{W}|$. It follows that with probability $1 - o(1)$ all the nodes informed in the first phase will have contributed at least

$$\begin{aligned}&|\mathcal{W}^1| + (1-\varepsilon)(R-1)p|\mathcal{W}^1| - \varepsilon|\mathcal{W}|\\ &\geq (1-2\varepsilon)(1-\varepsilon)pn + (1-\varepsilon)^2(R-1)p^2n\\ &\geq (1-3\varepsilon)Rp^2n\end{aligned}$$

random calls at the end of the second phase.

We now show that the largest interval of uninformed nodes is at most $(1+4\varepsilon)\ln(n)/(Rp^2)$. Let I be an interval of length $(1+4\varepsilon)\ln(n)/(Rp^2)$. Then the probability that no node in I becomes informed in the first or second phase by the random calls is at most

$$\begin{aligned}\Big(1 - \frac{(1+4\varepsilon)\ln n}{Rp^2n}\Big)^{(1-3\varepsilon)Rp^2n} &\leq \exp\left(-(1-3\varepsilon)(1+4\varepsilon)\ln n\right)\\ &= n^{-1-\varepsilon+12\varepsilon^2}\\ &= n^{-1-\varepsilon'},\end{aligned}$$

for some constant $\varepsilon' > 0$ (when ε is sufficiently small). Hence, by a simple union bound, there is no completely uninformed interval of length $(1+4\varepsilon)\ln(n)/(Rp^2)$ after the second phase with probability at least $1 - n^{-\varepsilon'}$ for some constant $\varepsilon' > 0$.

In the last phase, all the remaining free intervals are filled up. This takes at most the length of the largest uninformed interval which is $(1+4\varepsilon)\ln(n)/(Rp^2)$ by the previous argument.

So in total, we need

$$\begin{aligned}&(1+\varepsilon)(\log_{1+p} n) + 1 + R - 1 + (1+4\varepsilon)\ln(n)/(Rp^2)\\ &\qquad\leq (1+4\varepsilon)(\log_{1+p} n + \tfrac{1}{Rp^2}\ln n) + R\end{aligned}$$

rounds. Using a simple union bound, we can bound the total failure probability of all phases by $o(1)$.

It remains to bound the number of calls. At most n calls go to properly working nodes. Apart from that, each node contacts at most R informed nodes. Furthermore, in expectation, a node contacts a properly working node after p^{-1} rounds. Hence, we use at most n messages to inform all properly working nodes and additionally, in expectation, nR/p calls until all nodes stop informing. □

In the proof of Lemma 9.2.2, we used the fact that the quasirandom protocol with random node failures informs a fraction of $1-\varepsilon$ of all properly working nodes in $(1+\varepsilon)\log_{1+p} n$ rounds in probability $1-o(1)$. We now prove this fact.

Lemma 9.2.3. *The quasirandom protocol with random node failures and failure probability* $1-p$ *for* $p \in (0,1]$ *informs a fraction of* $1-\varepsilon$ *of all properly working nodes in* $(1+\varepsilon)\log_{1+p}(n)$ *rounds for an arbitrarily small constant* $\varepsilon > 0$ *in probability* $1-o(1)$.

Proof. We rely on the proof for the runtime of the quasirandom protocol with random *transmission* failures by Doerr, Huber, and Levavi [28]. In that model, the authors assume that each call fails independently with probability p. We now couple our node failure model with the transmission failure model. For each node i, we let i make the same random decisions in both models. If i tries to contact a node j in the transmission failure model and no other node has tried to call j before, then if this contact attempt is successful, we set in the node failure model j to be properly working, otherwise it is a failed node. Since each contact is successful independently with probability p, this coupling is correct. Thus, if in the transmission failure model we consider only the calls of nodes that were informed in the first contact attempt (i.e., no node tried to contact them before but failed), the runtime of both protocols is the same. It is straightforward to check that the proof for the transmission failure model only takes into account these contacts. Therefore, we can carry over all results to our case. Lemma 9.2.3 then follows from [28, Lemma 15]. □

Chapter 10

Robustness Against an Adversary

The main disadvantage of the previous protocol is its lack of robustness in the face of general, *non-random* transmission failures. Assume for example that a large segment of nodes that come consecutively in the permutation is not working. Then there is a reasonable chance that the first transmission sent by the starting node ends up in this faulty segment and hence the protocol needs time linear in the length of that segment to get out of it again. In particular, if we assume to have a linear number of such failed nodes, then with constant probability the protocol will take a linear number of rounds until all nodes are informed.

We now describe a modification of the algorithm that overcomes this problem, still achieves a logarithmic running time and performs a linear number of calls to properly working nodes. We assume that an adversary specifies a set of failed nodes. In contrast to the previously studied model, we distinguish between a *successful* call, i.e., a call to a properly working node, and an *unsuccessful* call, i.e., a call to a failed node.

This setting turns out to be more difficult to analyse. Even the more complex *median-counter algorithm* by Karp et al. [49], which relies on pull operations by uninformed nodes, only achieves a running time of $\Theta(\ln n)$ while using $\Theta(n \ln \ln n)$ calls. Moreover, their algorithm only guarantees to inform all but $O(|F|)$ nodes, where F is the set of failed nodes.

On the other hand, our algorithm informs *all* properly working nodes in $O(\ln n)$ rounds, relies only on push operations, needs only $O(n)$ successful calls and overall $O(n \ln n)$ calls. Moreover, in case we only have a small (constant) fraction of failed nodes, the runtime is close to the theoretical lower bound.

The reason why the total number of calls can be significantly larger than the number of successful calls lies in the end phase of the protocol. Just as in the basic protocol, the end phase is characterized by having informed almost all properly working nodes. An adversary could have distributed the remaining properly working nodes in a large bunch of such failed nodes. Since an informed node that is processing the list in such a segment of failed nodes cannot decide whether another node already passed by these nodes, we could end up spending a linear number of useless calls per node. To bound the number of calls, we introduce a security parameter p that the user can set for the fraction of properly working nodes. As long as p is a valid lower bound, we guarantee that the rumor is spread to all properly working nodes in at most $(1+\varepsilon)\big(\log_{1+p} n + \frac{1-p}{p}\ln n\big)$ rounds and that at most $\frac{1+\varepsilon}{p} n \ln(n)\big(2+\log_p(\varepsilon/12)\big)$ calls are made for any constant $\varepsilon > 0$. For example, if we assume that 99% of all nodes are properly working, then the protocol runs in time $1.011 \log_{1+p} n$ which is again very close to the lower bound of $\log_{1+p} n$.

Algorithm 2: Procedure started by newly informed node

```
$r \leftarrow 0, \mathit{lfc} \leftarrow 0, \mathit{gfc} \leftarrow 0,\ R \leftarrow \log_p(\varepsilon/12),\ L \leftarrow (1+\varepsilon)\frac{1-p}{p}\ln(n),$
$G \leftarrow (1+\varepsilon)\frac{1-p}{p}\ln(n)/p;$
repeat
    select node $j$ uniformly at random;
until $j$ properly working node;
while $r < R$ and $\mathit{gfc} < G$ do
    while $j$ not informed and $\mathit{lfc} < L$ do
        if $j$ failed then
            $\mathit{lfc} \leftarrow \mathit{lfc} + 1;$
            $\mathit{gfc} \leftarrow \mathit{gfc} + 1;$
        else
            inform $j$;
            $j \leftarrow j + 1;$
            $\mathit{lfc} \leftarrow 0;$
    $r \leftarrow r + 1;$ select node $j$ uniformly at random;
```

The protocol is very similar to the previous protocol. When a node receives the rumor, it starts the procedure described in Algorithm 2. As before, the starting node is treated as if it is uninformed until it is being called for the first time. The main difference to the basic protocol is that a newly informed node first performs random calls until it has found a properly working node. It then proceeds just as before. Intuitively, this modification prevents the bad scenario described above where the first call goes to a large segment of failed nodes and gets stuck there for a long time. For the stopping rule, we introduce three counters for each node. All these counters track the following quantities.

- The *restart counter* r of a node counts how many *informed* nodes it has called in total.
- The *global failure counter gfc* of a node counts how many *failed* nodes it has called in total.
- The *local failure counter lfc* of a node counts the number of failed nodes it has encountered *in a row* since its last random call.

The stopping rule is as follows. A node stops informing immediately if either the restart counter reaches $R = \log_p(\varepsilon/12)$ or the local failure counter reaches $L = (1+\varepsilon)\frac{1-p}{p}\ln(n)$. Furthermore, if the global failure counter reaches $G = (1+\varepsilon)\frac{1-p}{p}\ln(n)$, then the node stops informing the next time it calls an informed node. Whereas the restart counter fulfills the same role as in the basic protocol, the global failure counter makes sure that not too many calls are wasted on failed nodes. The local failure counter ensures that all properly working nodes are informed, even those that might be hidden in a segment of failed nodes.

Theorem 10.0.4. *Let $\varepsilon \in (0,1)$ and $p \le (n-|F|)/n < 1$, where F is the set of failed nodes in a graph. Then, the quasirandom protocol with restarts and a random start-up phase informs, with probability $1 - O(n^{-p\varepsilon/32})$, all properly working nodes in at most*

$$(1+\varepsilon)\bigl(\log_{1+p} n + \tfrac{1-p}{p}\ln n\bigr)$$

rounds using $O(n)$ successful calls and $2(1+\varepsilon)\frac{1-p}{p}n\ln n + O(n)$ calls in total.

In the following, we first prove the upper bound on the number of calls and then the upper bound on the running time.

10.1 Number of Calls

Lemma 10.1.1. *The number of calls is at most $2(1+\varepsilon)\frac{1-p}{p}n\ln n + O(n)$ with probability $e^{-\Omega(n)}$, out of which at most $n(R+1)$ are successful.*

Proof. We first bound the total number of calls made by nodes initially before they call the first properly working node. Let M_i^{init} denote the number of these calls made by node i and M^{init} the total number of these calls. Note that M_i^{init} is geometrically distributed with mean p^{-1}. By linearity of expectation, we get $\mathbb{E}(M^{\text{init}}) = \frac{n}{p}$. By Lemma 1.2.5, we have

$$\mathbb{P}(M^{\text{init}} > (1+\varepsilon)\tfrac{n}{p}) < e^{-\Omega(n)}.$$

Apart from these initial calls, each node calls at most R uninformed working nodes and at most $G+L$ failed nodes. So we spend n calls to inform the properly working nodes and at most $n(R+G+L)$ unsuccessful calls on either failed nodes or already informed properly working nodes.

In total, at most $n(G+L+R+1+\frac{1+\varepsilon}{p}) = n\big(2(1+\varepsilon)\frac{1-p}{p}\ln n + \log_p(\varepsilon/12) + 1 + \frac{1+\varepsilon}{p}\big) = 2(1+\varepsilon)\frac{1-p}{p} n \ln n + O(n)$ calls are made with probability $e^{-\Omega(n)}$. Of these calls, at most $n(R+1)$ are successful.

□

10.2 Running Time

10.2.1 Outline of Proof

The proof for the number of rounds follows the same structure as the proof in case of random node failures. As before, we also rely on the proof for the runtime of the quasirandom protocol with random transmission failures by Doerr et al. [28]. In particular, we use their proof to show that after $(1+\varepsilon)\log_{1+p} n$ rounds all but a constant fraction of the properly working nodes are informed. Note that in expectation a node will make the same number of failed contact attempts in both the adversarial model and the random transmission failure model. Since in the proofs of most results in the latter model, only this expectation is needed, we can transfer the results to the adversarial failure model. Whenever this is possible, we will skip the proof.

To obtain bounds that are precise up to the leading constant, we have to be careful that we do not lose too much through delaying and ignoring some nodes. For this reason, we distinguish two different types of phases.

Lazy phases were also used in the time analysis of [24]. Only nodes that are considered active at the beginning of the phase are considered active for the remainder of the phase. Nodes that are called during the phase, although they are still considered to be informed, remain inactive, and are therefore unable to spread the rumor themselves for the rest of the phase.

Since lazy phases neglect the rumor spreading potential of a significant portion of the nodes, we also need *busy phases*. Here, all nodes informed during the busy phase become active immediately. By choosing the lengths of the busy phases suitably, we balance out the difficulties with the inherent dependencies and the losses due to ignoring informed vertices at the end of each phase.

We now give the outline for the proof. Let I_t denote the set of informed vertices in round t. We call a node *newly informed* if it is informed but has not yet started the dissemination process (either because it was just informed or because it is delayed). We denote the set of newly informed vertices at round t by N_t. Let p denote the security parameter and assume that it is a lower bound on the fraction of properly working nodes.

Ideally, we would like to apply a similar proof as for Theorem 9.1.1.

Unfortunately, the phases used there turn out to be more difficult to accomplish with the presence of failed nodes. This is particularly true for the first phase. We refine the analysis by splitting the process into more phases. In all phases except for the *shooting phase*, we stop an active node after it has called at most two informed node. In the shooting phase, we make sure that a node stops after contacting R informed nodes. Thus, the limit of the restart counter is never violated.

1. In the *start-up phases* I and II, we inform in total $\Theta((\varepsilon \ln n)^2)$ nodes in $\frac{1}{3}\varepsilon \ln n$ rounds with high probability. Here, we exploit the fact that nodes first perform random calls until they find a properly working node to prevent that the process dies out early.

   Lemma 10.2.1. *Let $\varepsilon \in (0,1)$. After $t_1 := \frac{1}{3}\varepsilon \ln n$ rounds, we have $(\frac{1}{4}\varepsilon \ln n)^2 \geq |N_{t_1}| \geq (\frac{1}{12}p\varepsilon \ln n)^2$ with probability $1 - O(n^{-p\varepsilon/32})$ and $|I_{t_1}| \leq 1 + \frac{1}{4}\varepsilon \ln n + |N_{t_1}|$ informed nodes.*

2. In the *busy phases*, we still have only very few informed nodes. Hence, the probability of a *collision*, in which a node calls a node that is already informed, is still so low that only a few messages are wasted on already informed nodes. On the other hand, in expectation only a fraction p of all calls goes to properly working nodes (the rest is wasted in unsuccessful calls to failed nodes). We therefore witness an exponential growth of the number of informed nodes by a factor of almost $1 + p$ in each round. After $(1 + \varepsilon) \log_{1+p} n$ rounds, this phase ends having informed a small constant fraction of the properly working nodes.

   Lemma 10.2.2. *Let $\varepsilon \in (0,1)$. Let*

   $$k := \tfrac{1+\varepsilon}{\varepsilon}\left(2 - \log_{1+p} p\right) \quad \textit{and} \quad \zeta \leq \tfrac{1}{k}(2e)^{-\frac{2^{k-1}}{p^3(1+p)^{k-3}}-k-1}.$$

   Let $\zeta' := 2^{-k}\zeta$ and $\varepsilon' > 0$. Let t_1 be such that in our model at time t_1 we have $|N_{t_1}| \geq (p\varepsilon' \ln n)^2$ and $|I_{t_1}| \leq \min\left\{\zeta' n, \frac{2^k-1}{p(1+p)^{k-2}-1}|N_{t_1}|\right\}$. Let ℓ denote the smallest integer such that if we perform ℓ busy phases with k rounds we have $|I_{t_1+\ell k}| \geq \zeta' n$. Then

   $$\ell \leq \frac{(1+\varepsilon)\log_{1+p} n}{k} \quad \textit{and} \quad |I_{t_1+\ell k}| \leq \frac{2^k - 1}{p(1+p)^{k-2} - 1}|N_{t_1+\ell k}|$$

   hold with probability $1 - n^{-c}$ for any $c > 0$.

   Proof. The lemma is identical to [28, Theorem 14]. In these phases, we immediately stop the nodes that call an informed node. Thus, our protocol behaves just like the quasirandom protocol with identical lists analyzed in [28]. Although the failure models are different, we can still apply their proof since it only relies on the expected number of failed contacts for each node (which is the same in both models). □

3. In the *harvesting* phase, we inform almost all nodes in a constant number of rounds. In particular, we inform all properly working nodes except for a small constant fraction.

   Lemma 10.2.3. *Let* $\varepsilon \in (0,1)$ *and* $k := \frac{1+\varepsilon}{\varepsilon}\left(2 - \log_{1+p} p\right)$*. Let* t_2 *be such that* $|I_{t_2}| \le \frac{2^k-1}{p(1+p)^{k-2}-1}|N_{t_2}|$*, and* $\zeta, \zeta' \in (0,1)$ *such that* $\zeta' n \le |I_{t_2}| \le \zeta n$ *holds. Let* $S := 1 - \frac{2^k \ln(\zeta)}{(1-\zeta')\zeta'^2(p(1+p)^{k-2}-1)}$*.*

   After one lazy phase of S *rounds starting at time* t_2*, at least* $(p - 3\zeta)\, n$ *nodes will be newly informed with probability* $1 - e^{-\Omega(n)}$*.*

4. In the *shooting* phase, we now exploit the large number of informed nodes to ensure that the largest uninformed interval is small. We prove that at the end of this phase, the largest uninformed interval has length at most $(1 + \frac{2}{3}\varepsilon)\frac{1-p}{p}\ln n$.

   Lemma 10.2.4. *Let* $\varepsilon \in (0,1)$ *and* $\eta < \varepsilon p/48$*. Let* t_3 *be such that in our model at round* t_3 *we have* $|N_{t_3}| \ge (p - \eta)\, n$*. Then, after* $\log_{1-p}(\varepsilon/13) + \log_p(\varepsilon/12)$ *rounds, the largest uninformed interval of nodes is of length at most* $(1 + \frac{2}{3}\varepsilon)\frac{1-p}{p}\ln n$ *with probability* $1 - n^{-\varepsilon/3}$*.*

5. In the *fill-up* phase, we have to spend at most as many rounds as the largest interval of uninformed nodes until all properly working nodes are informed. It follows that we need at most $(1 + \frac{2}{3}\varepsilon)\frac{1-p}{p}\ln n$ additionally rounds.

In the analysis of the phases we have to take into account that nodes that are active in some phase could stop before the end of the phase because of our stopping rule. Note that all phases last for at most $(1+\varepsilon)\frac{1-p}{p}\ln n$ rounds (each of the ℓ busy phases has constant length). Hence, in those phases where only the newly informed nodes of the previous phase are active, we can guarantee that no active node will stop because of the limits set by the death counters before the end of the phase. This applies to all but the last phase. In this fill-up phase also nodes that were active in the previous phase remain active. However, since the last two phases combined do not last for more than $(1+\varepsilon)\frac{1-p}{p}\ln n$ rounds, we can again be sure that no active node will stop prematurely.

We now give the proof for Theorem 10.0.4 using Lemmas 10.2.1-10.2.4.

Proof of Theorem 10.0.4. Let $k := \frac{1+\varepsilon}{\varepsilon}\left(2 - \log_{1+p} p\right)$. Furthermore, let

$$\zeta := \min\left\{ \tfrac{1}{k}(2e)^{-\frac{2^{k-1}}{p^3(1+p)^{k-3}} - k - 1}, \tfrac{p\varepsilon}{3\cdot 48} \right\} \qquad \text{and} \qquad \zeta' := 2^{-k}\zeta.$$

We start the rumor spreading protocol from Algorithm 2 with security parameter p and with one initially informed node. After $t_1 := \frac{1}{3}\varepsilon \ln n$ rounds, we have by Lemma 10.2.1, with probability $1 - O(n^{-p\varepsilon/32})$,

$$(\tfrac{1}{4}\varepsilon \ln n)^2 \geq |N_{t_1}| \geq (\tfrac{1}{12}p\varepsilon \ln n)^2$$

and

$$|I_{t_1}| \leq 1 + \tfrac{1}{4}\varepsilon \ln n + |N_{t_1}| = (1 + o(1))|N_{t_1}| \leq \zeta' n, \tag{10.2.1}$$

where the last inequality holds for sufficiently large n. Furthermore, we have

$$\frac{2^k - 1}{p(1+p)^{k-2} - 1}|N_{t_1}| \geq \frac{2^k - 1}{2^{k-2} - 1}|N_{t_1}| \geq 4|N_{t_1}| \geq (1 + o(1))|N_{t_1}| \geq |I_{t_1}|. \tag{10.2.2}$$

So we can apply Lemma 10.2.2 with $\varepsilon' := \frac{\varepsilon}{12}$. This gives us an $\ell \leq \frac{(1+\varepsilon)\log_{1+p} n}{k}$ such that if we set $t_2 := t_1 + \ell k$, then for any $c > 0$, we have with probability $1 - n^{-c}$,

$$\zeta' n \leq |I_{t_2}| \leq \zeta n \qquad \text{and} \qquad |I_{t_2}| \leq \frac{2^k - 1}{p(1+p)^{k-2} - 1}|N_{t_2}|.$$

Thus, with probability $1 - n^{-c}$, the preconditions of Lemma 10.2.3 are fulfilled. Therefore, if we set $S := 1 - \frac{2^k \ln(\zeta)}{(1-\zeta')\zeta'^2}$ and $t_3 := t_2 + S$, we get $|N_{t_3}| \geq (p - 3\zeta)\, n$ with probability $1 - n^{-c}$. We can consequently apply Lemma 10.2.4 with $\eta := 3\zeta$. We conclude that after $\log_{1-p}(\varepsilon/13) + \log_p(\varepsilon/12)$ more rounds the largest uninformed interval of nodes is of length at most $(1 + \frac{2}{3}\varepsilon)\frac{1-p}{p} \ln n$ with probability $1 - n^{-\frac{1}{8}\varepsilon}$. Hence, we need at most $(1 + \frac{2}{3}\varepsilon)\frac{1-p}{p} \ln n$ more rounds until all nodes are informed.

Overall, we perform at most

$$\begin{aligned}
&\tfrac{1}{3}\varepsilon \ln(n) + (1+\varepsilon)\log_{1+p} n + S + \log_{1-p}(\varepsilon/13) \\
&\quad + \log_p(\varepsilon/12) + (1 + \tfrac{2}{3}\varepsilon)\tfrac{1-p}{p} \ln n \\
&\leq (1+\varepsilon)\left(\tfrac{1-p}{p} \ln n + \log_{1+p} n\right)
\end{aligned}$$

rounds in our delayed quasirandom rumor spreading protocol with message success probability p.

The overall failure probability is at most

$$O(n^{-p\varepsilon/32}) + n^{-c} + e^{-\Omega(n)} + n^{-\varepsilon/3} \leq O(n^{-p\varepsilon/32}).$$

□

In the next sections, we give the missing proofs for all phases.

10.2.2 Start-up Phases I and II - Proof of Lemma 10.2.1

The first phase is exceptional in the sense that it is neither a lazy nor a busy phase. In particular, we assume that only one node is active in each round, namely the most recently informed node. The second phase is a lazy phase.

Lemma 10.2.5. *Let $\varepsilon \in (0,1)$. After $\frac{1}{4}\varepsilon \ln n$ rounds, at least $\frac{1}{8}p\varepsilon \ln n$ nodes are informed with probability $1 - n^{-p\varepsilon/32}$. These nodes are distributed uniformly at random. Furthermore, in total, at most $1 + \frac{1}{4}\varepsilon \ln n$ nodes will be informed by the end of this phase.*

Proof. Each informed node that has not called a properly working node yet has a probability of at least p to call such a properly working node in each round. Once this happens, we delay the node till the next phase and let the newly informed node continue the process. In this sense, the node gives the baton to the next informed node. Since we start with one informed node, there will always be just one node trying to inform a properly working node. It follows that we can bound the number of informed nodes at the end of the phase by $1 + \frac{1}{4}\varepsilon \ln n$. If, during the process, a node calls an already informed node, we consider the whole phase a failure. So clearly, if no failure occurs, no active node will stop informing before the end of the phase. Note that the probability of such a failure is at most $(\frac{1}{4}\varepsilon \ln n)^2/n$. Assume now that no such failure occurs. Then we have,

$$\mathbb{E}[|I_{t_1}|] \geq \tfrac{1}{4}p\varepsilon \ln n. \tag{10.2.3}$$

By Chernoff's bound, we get

$$\begin{aligned}\mathbb{P}(|I_{t_1}| < \tfrac{1}{8}p\varepsilon \ln n) &< \exp(-\mathbb{E}[|I_{t_1}|]/8)\\ &\leq n^{-p\varepsilon/32}.\end{aligned}$$

Hence, by a simple union, we conclude that with probability at least $1 - (\frac{1}{4}\varepsilon \ln n)^2/n - n^{-p\varepsilon/32} = 1 - O(n^{-p\varepsilon/32})$ we will have informed at least $\frac{1}{8}p\varepsilon \ln n$ nodes.

Note that apart from the initial node all nodes were informed as a result of a random call. Hence, the nodes are distributed uniformly at random among the properly working nodes. □

Lemma 10.2.6. *Let $\varepsilon > 0$. Assume that at time t, we have $|I_t| \leq 1 + \frac{1}{4}\varepsilon \ln n$, out of which at least $\frac{1}{4}p\varepsilon \ln n$ nodes are distributed uniformly at random among all properly working nodes. Then, after one lazy phase of length $\frac{1}{12}\varepsilon \ln n$, at least $(\frac{1}{12}\varepsilon p \ln n)^2$ nodes are newly informed with probability $1 - O(n^{-1+\varepsilon})$.*

Proof. The active nodes throughout this lazy phase are the nodes I_t. We order the nodes in I_t arbitrarily. For $i \in \{1, \ldots, |I_t|\}$, let $X_i \in \{0, \ldots, \frac{1}{12}\varepsilon \ln n\}$

be the random variable denoting the number of properly working nodes in the interval of length $\frac{1}{12}\varepsilon \ln n$ following the i-th node. Since the nodes in I_t are distributed uniformly at random among all properly working nodes, we have

$$\mathbb{E}[X_i] \geq \tfrac{1}{12} p\varepsilon \ln n.$$

Let $X = \sum_{i=1}^{|I_t|} X_i$. Note that, if we assume that no collisions occur, no active node will stop informing before the end of the phase and no node will restart its informing process because of a collision. In this case, X is the number of informed nodes at the end of the phase. By linearity of expectation, we have

$$\mathbb{E}(X) = \sum_{i=1}^{|I_t|} \mathbb{E}(X_i) \geq (\tfrac{1}{4} p\varepsilon \ln n)(\tfrac{1}{12} p\varepsilon \ln n).$$

Since the X_i's are independently distributed and bounded by $\frac{1}{12}\varepsilon \ln n$, we can apply Hoeffding's inequality.

$$\begin{aligned}
\mathbb{P}[X < (\tfrac{1}{12} p\varepsilon \ln n)^2] &\leq \mathbb{P}[X < (1 - \tfrac{2}{3})\,\mathbb{E}(X)] \\
&\leq \exp\Big(\frac{-2(\frac{2}{3}\,\mathbb{E}(X))^2}{(\frac{1}{12}\varepsilon \ln n)^2}\Big) \\
&\leq \exp\big(-(p\varepsilon \ln n)^2/18\big) \\
&\leq n^{-\Omega(\ln n)}.
\end{aligned}$$

We bound the probability of a collision as follows. By our assumption, there are at least pn properly working nodes. For every node $i \in I_t$, the probability that it calls an informed node is at most $2(|I_t| - 1)\frac{1}{12}\varepsilon \ln(n)/(pn) \leq (\varepsilon \ln n)^2/(pn)$. So, by a simple union bound, the probability of a collision is at most $|I_t|(\varepsilon \ln n)^2/(pn) \leq (\varepsilon \ln n)^3/(pn) \leq n^{-1+\varepsilon}$. In total, the probability that at least $(\frac{1}{12} p\varepsilon \ln n)^2$ nodes are newly informed is at least $1 - n^{-\Omega(\ln n)} - n^{-1+\varepsilon} \leq 1 - O(n^{-1+\varepsilon})$.

Since we are considering a lazy phase, we can bound the total number of informed nodes at the end of the phase by $|I_t|(1 + \frac{1}{12}\varepsilon \ln n) \leq (\frac{1}{4}\varepsilon \ln n)^2$. □

Lemma 10.2.1 now follows immediately from combining Lemma 10.2.5 and Lemma 10.2.6. Note that since the second phase is a lazy phase, we have $|I_{t_1}| \leq 1 + \frac{1}{4}\varepsilon \ln n + |N_{t_1}|$, where t_1 is the time step at the end of the start-up phase.

10.2.3 Harvesting Phase - Proof of Lemma 10.2.3

Now that we have a small constant fraction of newly informed nodes, a lazy phase of a constant number of rounds suffices to yield a large fraction of newly informed nodes. We want to bound the probability that nodes in N_{t_2} fail to inform an uninformed node from above. Remember that newly informed nodes keep making random calls until they contact a properly working node. Thus, the probability of calling an uninformed node depends on the position of these failed nodes. To simplify the analysis, we will therefore only work with newly informed nodes that call an informed node in the first round. These nodes then proceed just as in the basic quasirandom protocol with restarts. In particular, they will call all nodes with the same probability.

We perform one lazy phase of S rounds starting at time t_2. Let $N'_{t_2} \subseteq N_{t_2}$ denote those nodes that call an already informed node in the first round of the phase. We have

$$\mathbb{E}(|N'_{t_2}|) \geq \zeta'|N_{t_2}| \geq \zeta' \frac{p(1+p)^{k-2}-1}{2^k-1}|I_{t_2}| \geq \zeta'^2 \frac{p(1+p)^{k-2}-1}{2^k-1}n.$$

Since each node in N_{t_2} makes a random call in the first round independently of other nodes, we can apply Chernoff's inequality. We have

$$\mathbb{P}\big(|N'_{t_2}| < (1-\zeta')\zeta'^2 \frac{p(1+p)^{k-2}-1}{2^k-1}n\big) = e^{-\Omega(n)}.$$

In the following we assume that $|N'_{t_2}| \geq (1-\zeta')\zeta'^2 \frac{p(1+p)^{k-2}-1}{2^k-1}n$. The nodes in N'_{t_2} now proceed according to the basic quasirandom model with restarts, i.e., they choose a random starting point (regardless of whether it is a failed or properly working node) and try to inform new nodes from that point on.

Let $v \in V \backslash I_{t_2}$. If there exists an informed node in $[v-S, v)$, then v will be informed in this phase with probability 1. Otherwise, even if we assume that the nodes in N'_{t_2} stop informing immediately after calling an informed node, the probability that a node in N'_{t_2} does not call v is $1 - \frac{S-1}{n}$. Thus, we have

$$\begin{aligned}&\mathbb{P}(\text{no node in } N'_{t_2} \text{ calls } v \text{ in this phase}) \leq \Big(1 - \frac{S-1}{n}\Big)^{|N'_{t_2}|} \\ &\leq \exp\Big(-\frac{(S-1)|N'_{t_2}|}{n}\Big) \leq \exp\Big(-\frac{(S-1)(1-\zeta')\zeta'^2(p(1+p)^{k-2}-1)}{2^k}\Big) = \zeta,\end{aligned}$$

where the last inequality follows from $S = 1 - \frac{2^k \ln(\zeta)}{(1-\zeta')\zeta'^2(p(1+p)^{k-2}-1)}$. We now compute the expected number of newly informed nodes after S rounds. Let $t_3 := t_2 + S$. We have

$$\begin{aligned}\mathbb{E}\left(|N_{t_3}|\right) &= |V \backslash (F \cup I_{t_2})| \cdot \mathbb{P}(v \in V \backslash I_{t_2} \text{ is informed in this phase}) \\ &\geq \left(n - |F| - |I_{t_2}|\right)(1-\zeta) \\ &\geq \left(pn - \zeta n\right)(1-\zeta) \\ &\geq (p - 2\zeta)n.\end{aligned}$$

Using Azuma's Inequality (see Lemma 1.2.7) we show that the probability that $|N_{t_3}|$ deviates significantly from its expected value is exponentially small. Number the nodes of N'_{t_2} from 1 to $|N'_{t_2}|$. Then for all

$i \in \{1, \ldots, |N'_{t_2}|\}$, define the random variable X_i as the set of vertices that node i calls in this phase. We define the function f such that

$$f(X_1, \ldots, X_{|N'_{t_2}|}) := \left| \bigcup_{i=1}^{|N'_{t_2}|} X_i \backslash I_{t_2} \right| = |N_{t_3}|.$$

Since each node can inform at most S nodes in this phase, we have for any set of values $x_1, \ldots, x_{|N'_{t_2}|}$ and x'_i,

$$f(x_1, \ldots, x_i, \ldots, x_{|N'_{t_2}|}) - f(x_1, \ldots, x'_i, \ldots, x_{|N'_{t_2}|}) \leq S.$$

Thus, we can apply Azuma's inequality to bound the probability that we inform less than $(p - 3\zeta)n$ vertices in this phase. We have

$$\begin{aligned}
\mathbb{P}\left(|N_{t_3}| < (p - 3\zeta)\, n\right) &= \mathbb{P}\left(|N_{t_3}| < (p - 2\zeta)n - \zeta n\right) \\
&\leq \mathbb{P}\left(|N_{t_3} - \mathbb{E}(|N_{t_3}|)| \geq \zeta n\right) \\
&\leq 2 \exp\left(-\frac{2\zeta^2 n^2}{\sum_{i=1}^{|N'_{t_2}|} S^2}\right) \\
&\leq 2 \exp\left(-\frac{2\zeta^2 n^2}{\zeta n S^2}\right) \\
&= e^{-\Omega(n)}.
\end{aligned}$$

10.2.4 Shooting Phase - Proof of Lemma 10.2.4

We analyse this phase as a lazy phase. Similar to the preceding phase, we will first perform a constant number of rounds to get a large fraction of the newly informed nodes to call an already informed node. If more than ηn nodes call an uninformed properly working node during these rounds at the beginning of the phase, then all properly working nodes must have been informed and we are finished. Otherwise there must be at least $(p-2\eta)n$ nodes in N_{t_3} that did not call an uninformed properly working node in these rounds. These nodes have a probability of p to call a properly working node in every round. By a similar argument as in the preceding phase, we can show that a large fraction of these nodes will call an informed node and then proceed as in the quasirandom model with restarts. Let $N'_{t_3} \subseteq N_{t_3}$ denote those nodes that call an informed node in the first $\log_{1-p}(\varepsilon/13)$ rounds of the phase but do not call an uninformed node. For each node $i \in N_{t_3}$ independently, conditioned on the event that i does not call an uninformed node in these rounds, the probability that i does not call an informed node is $(1-p)^{\log_{1-p}(\varepsilon/13)} = \varepsilon/13$.

By Chernoff's bound, we get

$$\mathbb{P}(|N'_{t_3}| < (1 - \tfrac{\varepsilon}{12})(p - 2\eta)n) = e^{-\Omega(n)}. \qquad (10.2.4)$$

So assume that $|N'_{t_3}| > (1 - \frac{\varepsilon}{12})(p - 2\eta)n$.

We now bound from below the number of random calls made by nodes after they have called an informed node in this phase. Since a random call is always made after a node contacts an informed node, we get a bound by considering the number of calls to informed nodes. For that, we first bound the number of calls to properly working nodes. Since there can be at most ηn calls to uninformed nodes, we then get a bound for the random calls. For the lower bound, we do not count any calls made by a node in N'_{t_3} after it has called a failed node. Thus, we consider the following random experiment. A node in N'_{t_3} keeps making random calls until it calls a failed node or until it has made R calls. Let A_i be the random variable denoting this number of random calls. Note that A_i is a geometrically distributed random variable with 'success probability' $1 - p$ and an upper bound of R, i.e.,

$$\mathbb{P}(A_i = k) = \begin{cases} p^{k-1}(1-p) & \text{if } k \leq R-1 \\ p^{k-1} & \text{if } k = R \\ 0 & \text{else.} \end{cases} \tag{10.2.5}$$

Let $A = \sum_{i \in N'_{t_3}} A_i$. Note that A is a lower bound for the number of random calls made in this phase since a node makes at most R random calls according to our stopping rule. We have

$$\begin{aligned}
\mathbb{E}[A_i] &= Rp^{R-1} + \sum_{1 \leq k \leq R-1} kp^{k-1}(1-p) \\
&= Rp^{R-1} + (1-p)\big(\sum_{1 \leq k \leq R-1} p^k\big)' \\
&= Rp^{R-1} + (1-p)\big(\frac{1-p^R}{1-p} - 1\big)' \\
&= Rp^{R-1} + (1-p)\big(\frac{1-p^R}{1-p} - 1\big)' \\
&= Rp^{R-1} + \frac{(R-1)p^R - Rp^{R-1} + 1}{1-p} \\
&= \frac{Rp^{R-1} - Rp^R + (R-1)p^R - Rp^{R-1} + 1}{1-p} \\
&= \frac{1-p^R}{1-p} \\
&\geq (1 - \varepsilon/12)(1-p)^{-1},
\end{aligned}$$

where the last equality follows from our choice $R = \log_p(\varepsilon/12)$. Since the A_i's are independent random variables and bounded by R, we can apply

Hoeffding's bound on the sum A. In particular, we get

$$\begin{aligned}
&\mathbb{P}(A < (1-\varepsilon/12)|N'_{t_3}|(1-p)^{-1}(1-\varepsilon/12))\\
&< \exp\Big(-\frac{2\big(\varepsilon/12|N'_{t_3}|(1-p)^{-1}(1-\varepsilon/12)\big)^2}{|N'_{t_3}|R^2}\Big)\\
&< \exp\Big(-\frac{2\varepsilon^2|N'_{t_3}|(1-\varepsilon/12)^2}{144(1-p)^2R^2}\Big)\\
&< e^{-\Omega(n)},
\end{aligned}$$

where the last inequality follows from our assumption that $|N'_{t_3}| > (1-\varepsilon/12)(p-2\eta)n$. Since there are at most ηn nodes in N'_{t_3} that call an uninformed node, we conclude that we have at least

$$\begin{aligned}
&(1-\varepsilon/12)|N'_{t_3}|(1-p)^{-1}(1-\varepsilon/12)-\eta n\\
&\geq (1-\varepsilon/4)(p-4\eta)(1-p)^{-1}n\\
&\geq (1-\varepsilon/4)(p-\varepsilon p/12)(1-p)^{-1}n \qquad \text{since } \eta \leq \varepsilon p/48,
\end{aligned}$$

random calls in this phase. Let I be an interval of length $(1+\frac{2}{3}\varepsilon)\frac{1-p}{p}\ln(n)$. Then the probability that no node in I becomes informed in this phase by these random calls is at most

$$\begin{aligned}
&\Big(1-\frac{(1+\frac{2}{3}\varepsilon)(1-p)\ln n}{pn}\Big)^{(1-\varepsilon/4)(p-\varepsilon p/12)n(1-p)^{-1}}\\
&< \exp\big(-(1+\tfrac{2}{3}\varepsilon)(1-\varepsilon/4)(1-\varepsilon/12)\ln n\big)\\
&< n^{-1-\varepsilon/3}.
\end{aligned}$$

Hence, by a union bound argument, it follows that there is no completely uninformed interval of length $(1+\frac{2}{3}\varepsilon)\frac{1-p}{p}\ln n$ after this phase with probabililty at least $1-n^{-\varepsilon/3}$.

10.2.5 The Fill-up Phase

Since by the previous section the largest free interval has length at most $(1+\frac{2}{3}\varepsilon)\frac{1-p}{p}\ln n$, we also need at most that many rounds until all nodes are informed.

Bibliography

[1] R. Albert, H. Jeong, and A.-L. Barabasi. Diameter of the world-wide web. *Nature*, 401:130–131, 1999.

[2] A.-L. Barabási. *Linked: How Everything Is Connected to Everything Else and What It Means*. Plume, 2003.

[3] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[4] P. Beaumont. The truth about Twitter, Facebook and the uprisings in the Arab world, Feb 2011. `www.guardian.co.uk/world/2011/feb/25/twitter-facebook-uprisings-arab-libya`.

[5] N. Berger, C. Borgs, J. T. Chayes, and A. Saberi. On the spread of viruses on the internet. In *Proceedings of the* 16*th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 301–310, 2005.

[6] B. Bhattacharjee, P. Druschel, K. Gummadi, et al. Online social networks research at the Max Planck Institute for Software Systems. `socialnetworks.mpi-sws.org`.

[7] T. Bohman and A. M. Frieze. Hamilton cycles in 3-out. *Random Structures & Algorithms*, 35:393–417, 2009.

[8] B. Bollobás and O. Riordan. Robustness and vulnerability of scale-free random graphs. *Internet Mathematics*, 1:1–35, 2003.

[9] B. Bollobás and O. Riordan. Coupling scale-free and classical random graphs. *Internet Mathematics*, 1:215–225, 2003.

[10] B. Bollobás and O. Riordan. The diameter of a scale-free random graph. *Combinatorica*, 24:5–34, 2004.

[11] B. Bollobás and A. Thomason. *On Richardson's Model on the Hypercube*. Cambridge University Press, 1997.

[12] B. Bollobás, O. Riordan, J. Spencer, and G. Tusnády. The degree sequence of a scale-free random graph process. *Random Structures & Algorithms*, 18:279–290, 2001.

[13] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52:2508–2530, 2006.

[14] M. Bradonjic, R. Elsässer, T. Friedrich, T. Sauerwald, and A. Stauffer. Efficient broadcast on random geometric graphs. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1412–1421, 2010.

[15] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web. *Computer Networks*, 33:309–320, 2000.

[16] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi. Measuring user influence in twitter: The million follower fallacy. In *Proceedings of the 4th Annual International AAAI Conference on Weblogs and Social (ICWSM)*, 2010.

[17] F. Chierichetti, S. Lattanzi, and A. Panconesi. Rumor spreading in social networks. In *Proceedings of the 36th Annual International Colloquium on Automata, Languages and Programming (ICALP)*, pages 375–386, 2009.

[18] F. Chierichetti, S. Lattanzi, and A. Panconesi. Almost tight bounds for rumour spreading with conductance. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 399–408, 2010.

[19] F. Chierichetti, S. Lattanzi, and A. Panconesi. Rumor spreading in social networks. *Theoretical Computer Science*, 412:2602–2610, 2011.

[20] F. R. K. Chung and L. Lu. The average distance in a random graph with given expected degrees. *Internet Mathematics*, 1:91–113, 2003.

[21] C. Cooper, A. M. Frieze, and J. Vera. Random deletion in a scale-free random graph process. *Internet Mathematics*, 1:463–483, 2004.

[22] A. J. Demers, D. H. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. E. Sturgis, D. C. Swinehart, and D. B. Terry. Epidemic algorithms for replicated database maintenance. *Operating Systems Review*, 22: 8–32, 1988.

[23] B. Doerr and M. Fouz. Asymptotically optimal randomized rumor spreading. In *Proceedings of the 36th Annual International Colloquium on Automata, Languages and Programming (ICALP)*, pages 502–513, 2011.

[24] B. Doerr, T. Friedrich, and T. Sauerwald. Quasirandom rumor spreading. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 773–781, 2008.

[25] B. Doerr, T. Friedrich, M. Künnemann, and T. Sauerwald. Quasirandom rumor spreading: An experimental analysis. In *Proceedings of the 10th Annual Workshop on Algorithm Engineering and Experiments (ALENEX)*, pages 145–153, 2009.

[26] B. Doerr, T. Friedrich, and T. Sauerwald. Quasirandom rumor spreading: Expanders, push vs. pull, and robustness. In *Proceedings of the 36th Annual International Colloquium on Automata, Languages and Programming (ICALP)*, pages 366–377, 2009.

[27] B. Doerr, M. Fouz, and C. Witt. Quasirandom evolutionary algorithms. In M. Pelikan and J. Branke, editors, *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation (GECCO)*, pages 1457–1464. ACM, 2010.

[28] B. Doerr, A. Huber, and A. Levavi. Strong robustness of randomized rumor spreading protocols. *CoRR*, abs/1001.3056, 2010.

[29] B. Doerr, M. Fouz, and T. Friedrich. Social networks spread rumors in sublogarithmic time. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 21–30, 2011.

[30] B. Doerr, M. Fouz, and T. Friedrich. Why rumors spread fast in social networks. *Communications of the ACM*, to appear.

[31] S. Dommers, R. van der Hofstad, and G. Hooghiemstray. Diameters in preferential attachment models. *Journal of Statistical Physics*, 139: 72–107, 2010.

[32] D. Dubhashi and A. Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms.* Cambridge University Press, New York, NY, USA, 2009.

[33] R. Durrett. *Random Graph Dynamics.* Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, New York, NY, USA, 2006.

[34] R. Elsässer. On the communication complexity of randomized broadcasting in random-like graphs. In *Proceedings of the 18th Annual ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*, pages 148–157, 2006.

[35] R. Elsässer and T. Sauerwald. Broadcasting vs. mixing and information dissemination on cayley graphs. In *Proceedings of the 24th Annual*

International Symposium on Theoretical Aspects of Computer Science (STACS), pages 163–174, 2007.

[36] R. Elsässer and T. Sauerwald. The power of memory in randomized broadcasting. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 218–227, 2008.

[37] U. Feige, D. Peleg, P. Raghavan, and E. Upfal. Randomized broadcast in networks. *Random Structures & Algorithms*, 1:447–460, 1990.

[38] J. A. Fill and R. Pemantle. Percolation, first-passage percolation, and covering times for Richardson's model on the n-cube. *The Annals of Applied Probability*, 3:593–629, 1993.

[39] A. D. Flaxman, A. M. Frieze, and J. Vera. Adversarial deletion in a scale-free random graph process. *Combinatorics, Probability & Computation*, 16:261–270, 2007.

[40] N. Fountoulakis and A. Huber. Quasirandom rumour spreading on the complete graph is as fast as randomized rumour spreading. *SIAM Journal on Discrete Mathematics*, 23:1964–1991, 2009.

[41] N. Fountoulakis and K. Panagiotou. Rumor spreading on random regular graphs and expanders. In *Proceedings of the 14th Annual International Workshop on Randomization and Computation (RANDOM)*, volume 6302 of *LNCS*, pages 560–573, 2010.

[42] N. Fountoulakis, K. Panagiotou, and T. Sauerwald. Ultra-fast rumor spreading in social networks. In *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2012 (to appear).

[43] A. M. Frieze and G. R. Grimmett. The shortest-path problem for graphs with random arc-lengths. *Discrete Applied Mathematics*, 10:57–77, 1985.

[44] A. M. Frieze and M. Molloy. Broadcasting in random graphs. *Discrete Applied Mathematics*, 54:77–79, 1994.

[45] G. Giakkoupis. Tight bounds for rumor spreading in graphs of a given conductance. In *Proceedings of the 28th Annual International Symposium on Theoretical Aspects of Computer Science (STACS 2011)*, pages 57–68, 2011.

[46] S. M. Hedetniemi, S. T. Hedetniemi, and A. L. Liestman. A survey of gossiping and broadcasting in communication networks. *Networks*, 18: 319–349, 1988.

[47] R. V. D. Hofstad. Random graphs and complex networks. In preparation, 2011.

[48] S. Janson. One, two and three times $\log n/n$ for paths in a complete graph with random weights. *Combinatorics, Probability & Computation*, 8:347–361, 1999.

[49] R. Karp, C. Schindelhauer, S. Shenker, and B. Vöcking. Randomized rumor spreading. In *Proceedings of the 41st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 565–574, 2000.

[50] D. Kempe, A. Dobra, and J. Gehrke. Gossip-based computation of aggregate information. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 482–491, 2003.

[51] D. Kempe, J. M. Kleinberg, and A. J. Demers. Spatial gossip and resource location protocols. *Journal of the ACM*, 51(6):943–967, 2004.

[52] S. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Extracting large-scale knowledge bases from the web. In *Proceedings of the 25th Annual International Conference on Very Large Data Bases (VLDB)*, 1999.

[53] J. Leskovec. Stanford large network dataset collection. `http://snap.stanford.edu/data/`.

[54] H. Mahmoud. *Polya Urn Models.* Chapman & Hall/CRC, 2008.

[55] M. Mihail, C. Papadimitriou, and A. Saberi. On certain connectivity properties of the internet topology. *Journal of Computer and System Sciences*, 72:239–251, 2006.

[56] S. Milgram. The small-world program. *Psychology Today*, 2:60–67, 1967.

[57] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th Annual ACM SIGCOMM Conference on Internet Measurement (IMC)*, pages 29–42, 2007.

[58] M. Mitzenmacher and E. Upfal. *Probability and computing — randomized algorithms and probabilistic analysis.* Cambridge University Press, 2005.

[59] R. Motwani and P. Raghavan. *Randomized algorithms.* Cambridge University Press, New York, NY, USA, 1995.

[60] T. K. Philips, D. F. Towsley, and J. K. Wolf. On the diameter of a class of random graphs. *IEEE Transactions on Information Theory*, 36(2): 285–288, 1990.

[61] B. Pittel. On spreading a rumor. *SIAM Journal on Applied Mathematics*, 47:213–223, 1987.

[62] D. Richardson. Random growth in a tessellation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 74:515–528, 1973.

[63] T. Sauerwald. On mixing and edge expansion properties in randomized broadcasting. *Algorithmica*, 56:51–88, 2010.