Low Randomness Rumor Spreading via Hashing

George Giakkoupis[*]
University of Calgary, Canada
ggiakkou@@ucalgary.ca

Thomas Sauerwald
Max Planck Institute, Germany
sauerwal@mpi-inf.mpg.de

He Sun
Max Planck Institute, Germany
hsun@mpi-inf.mpg.de

Philipp Woelfel[†]
University of Calgary, Canada
woelfel@@ucalgary.ca

January 2, 2012

Abstract

We consider the classical rumor spreading problem, where a piece of information must be disseminated from a single node to all n nodes of a given network. We devise two simple push-based protocols, in which nodes choose the neighbor they send the information to in each round using pairwise independent hash functions, or a pseudo-random generator, respectively. For several well-studied topologies our algorithms use exponentially fewer random bits than previous protocols. For example, in complete graphs, expanders, and random graphs only a polylogarithmic number of random bits are needed in total to spread the rumor in $\mathcal{O}(\log n)$ rounds with high probability. Previous explicit algorithms, e.g., [6, 10, 15, 17], require $\Omega(n)$ random bits to achieve the same round complexity. For complete graphs, the amount of randomness used by our hashing-based algorithm is within an $\mathcal{O}(\log n)$-factor of the theoretical minimum determined by Giakkoupis and Woelfel [15].

1 Introduction

Broadcasting a piece of information to all nodes in a network is one of the fundamental problems in the theory of network algorithms. A basic variant of the problem is the *rumor spreading problem*: One node in a graph with n nodes initially obtains a piece of information, called the *rumor*. In subsequent synchronous rounds, nodes communicate with randomly chosen neighbors in order to spread the rumor. Protocols for rumor spreading are of fundamental interest and have several applications, such as in the maintenance of distributed replicated database systems [4, 10], failure detection [24], resource discovery [16], and data aggregation [1]. As such, the problem has been well-studied in the literature.

Several design goals have been considered when devising rumor spreading protocols: Most importantly, the algorithm should be *efficient*, in the sense that the rumor reaches every node in a small number of rounds. In addition, rumor spreading protocols should be *local*, i.e., nodes should

[*] Supported by the Pacific Institute for the Mathematical Sciences (PIMS), and the Natural Sciences and Engineering Research Council of Canada (NSERC).

[†] Supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

not need to have any information about the global connectivity of the network. Another important property is *robustness*, that is, the protocol can tolerate the failure of some links [10, 17].

Under standard model assumptions, *deterministic* rumor spreading protocols are often inefficient. For example, on complete graphs, it is not possible to spread the rumor in $o(n)$ rounds if nodes in the adjacency lists can be ordered arbitrarily (and nodes do not know the orderings). Hence, essentially all rumor spreading protocols of note are *randomized*.

In the standard models, a node u can open a communication channel to one of its neighbors, v. If u knows the rumor, u can then *push* it to v, and if u does not know the rumor, it can try to *pull* it from v. In *push-protocols*, where the rumor is disseminated solely by push communication, it is easy to share randomness among all nodes: The first node that obtains the rumor can generate a random string and then nodes pass the same random string along with the rumor. It is known that for push-protocols it is necessary to share randomness in order to achieve both, time and randomness efficiency [15]. On the other hand, in order to benefit from pull-communications, nodes that have not received any messages would have to generate their own "private" random strings to determine their random communication partners. Thus, it seems that pull-communications cannot help to spread the rumor to many nodes, unless many nodes generate random strings, and thus in total a large number of random bits is generated. Therefore, in this paper we restrict our attention to push-protocols.

Precisely, a push-protocol proceeds in synchronous rounds as follows: Initially, in round 0, an arbitrary node receives the rumor. In every succeeding round, every *informed* node (i.e., every node that received the rumor in a previous round) chooses a random neighbor (according to some probability distribution), which it then informs about the rumor.

In the *fully random* protocol, in each round every informed node chooses its neighbor uniformly at random. This simple classical protocol, which has been extensively studied, is local, robust, and efficient. For instance, for a variety of graphs, such as complete graphs [13, 23], hypercubes [10], random graphs [12], expanders [14, 21], or regular graphs with constant conductance [14], only $\mathcal{O}(\log n)$ rounds are needed to spread the rumor to all nodes with high probability (w.h.p.).[1] In a graph of degree d, every informed node needs to choose $\log d$ random bits in every round, and thus typically a total of $\Theta(t \cdot n \cdot \log d)$ random bits are needed, if the protocol runs for t rounds.

The so-called *quasi-random* protocol was proposed by Doerr, Friedrich, and Sauerwald with the aim of "imitating properties of the classical push model with a much smaller degree of randomness" [6]. The idea is that each node, once it becomes informed, only chooses one starting point in its adjacency list uniformly at random. From then on, it contacts its neighbors in the order they appear in the adjacency list, beginning with that starting point (and in a round-robin fashion). The protocol has very similar properties as the fully random algorithm, and in particular it has been proven to be as efficient on complete graphs, random graphs, strong expanders and hypercubes (see also Table 1). Only $\mathcal{O}(n \log d)$ random bits are needed in total for the quasi-random protocol on a graph of degree d.

Doerr and Fouz [5] showed that in the complete graph one cannot further reduce the amount of randomness of the quasi-random protocol by limiting each node's choice of its starting point in its list without sacrificing the efficiency of the protocol. Giakkoupis and Woelfel [15] proved more general upper and lower bounds for the amount of randomness required to spread a rumor on the complete graph: They provided a relatively simple protocol that needs only $\mathcal{O}(n \log \log n)$ random bits in total to spread the rumor to all nodes of the complete graph. Moreover, they showed that any protocol that uses only $\log n - \log \log n - \omega(1)$ random bits needs $\omega(\log n)$ rounds to inform

[1] We say an event occurs with high probability, if there exists a constant $\varepsilon > 0$ such that the probability of the event is $1 - \mathcal{O}(n^{-\varepsilon})$.

Graph family	Rumor spreading time		Random bits	Reference
Graphs with $\Delta/\delta = \mathcal{O}(1)$	$\mathcal{R}(G) = \mathcal{O}((1/\phi)\log n)$,	w.h.p.	$\Omega(n \log n \log \Delta)$	[14, 21]
	$\mathcal{H}(G) = \mathcal{O}((1/\phi)\log n)$,	w.h.p.	$\mathcal{O}((1/\phi)\log^2 n)$	Thm. 3.3
Expanders	$\mathcal{R}(G) = \mathcal{O}(\log n)$,	w.h.p.	$\Theta(n \log n)$	[14, 21]
	$\mathcal{H}(G) = \mathcal{O}(\log n)$,	w.h.p.	$\mathcal{O}(\log^2 n)$	Thm. 3.3
Strong Expanders	$\mathcal{R}(G) = \log n + \ln n + o(\log n)$,	w.p. $1 - o(1)$	$\Theta(n \log n \log \Delta)$	Cor. 4.2
	$\mathcal{Q}(G) = \mathcal{O}(\log n)$,	w.h.p.	$\Theta(n \log \Delta)$	[6, 7]
	$\mathcal{P}(G) = \log n + \ln n + o(\log n)$,	w.p. $1 - o(1)$	$\mathcal{O}(\log^3 n)$	Thm. 4.1
Complete Graphs	$\mathcal{R}(G) = \log n + \ln n + o(\log n)$,	w.p. $1 - o(1)$	$\Theta(n \log^2 n)$	[23]
	$\mathcal{Q}(G) = \log n + \ln n + o(\log n)$,	w.p. $1 - o(1)$	$\Theta(n \log n)$	[11]
	$\mathcal{P}(G) = \log n + \ln n + o(\log n)$,	w.p. $1 - o(1)$	$\mathcal{O}(\log^3 n)$	Thm. 4.1
$G(n, p)$ with $p = \omega(\log n/n)$	$\mathcal{R}(G) = \log n + \ln n + o(\log n)$,	w.p. $1 - o(1)$	$\Theta(n \log n \log(pn))$	Cor. 4.2 & [12]
	$\mathcal{Q}(G) = \mathcal{O}(\log n)$,	w.h.p.	$\Theta(n \log(pn))$	[6]
	$\mathcal{P}(G) = \log n + \ln n + o(\log n)$,	w.p. $1 - o(1)$	$\mathcal{O}(\log^3 n)$	Cor. 4.4

Table 1: Comparison of the rumor spreading time and the required number of random bits for various topologies. By $\mathcal{R}(G)$ and $\mathcal{Q}(G)$ we denote the rumor spreading time of the fully random and the quasi-random push-protocols, respectively, on graph G. By $\mathcal{H}(G)$ and $\mathcal{P}(G)$ we denote the rumor spreading time of our hashing-based and PRG-based protocols. By δ and Δ we denote the minimum and maximum degrees of G, and ϕ denotes the conductance of G (see Section 3.2 for the definition of conductance). All the time bounds for $\mathcal{H}(G)$ listed above also hold for $\mathcal{P}(G)$.

all nodes. While the probabilistic method can be employed to show that there *exists* a protocol which needs only $\mathcal{O}(\log n)$ random bits to inform all nodes in the complete graph [15], no explicit construction of a protocol that uses less than $\mathcal{O}(n)$ random bits was known prior to this work.

1.1 Our Results

We present the first explicit rumor spreading protocols that use a sub-linear number of random bits in total to efficiently spread the rumor in a wide class of networks. We describe two protocols: one that uses hash functions, and one that uses pseudo-random generators (short: PRGs). If the protocols run for a number of t rounds, then they need $\mathcal{O}(t \cdot \log n)$ and $\mathcal{O}(t \cdot \log^2 n)$ random bits, respectively. We prove that for many standard graph topologies a logarithmic or polylogarithmic number of rounds suffice to broadcast the rumor w.h.p., so only a polylogarithmic number of random bits are consumed. In particular, using only a polylogarithmic number of random bits, our protocols are asymptotically as efficient as the best known protocol (i.e., the fully random one) on expanders and "strong" expanders (for the definition of strong expanders see the beginning of Section 4). For strong expanders, such as the complete graph and random graphs $G(n, p)$ with $p = \omega(\log n/n)$, our time bound matches the lower bound for regular graphs shown in [9] for the fully random protocol. We also prove a general upper bound of $\mathcal{O}((1/\phi)\log n)$ rounds, where ϕ is the conductance of the underlying graph. This bound is tight in the sense that there are graphs for which the diameter is at least $\Omega((1/\phi)\log n)$ [2]. The same upper bound was shown for the fully random push-protocol in [2, 14, 21]. For a more complete overview of our results and a comparison with previous results, see Table 1.

In our hashing-based protocol, nodes use *pairwise independent hash functions* (one for each round) to determine the neighbors to send the rumor to. The intuition is the following: In some round every informed node v establishes a communication link to a random neighbor $X(v)$. For the efficiency of the protocol it is important that many of the random variables $X(v)$ are distinct, i.e., that the number of "message collisions" is small. In order to bound the number of collisions, we can use second-moment methods and thus rely on pairwise independence of the random variables $X(v)$, as opposed to complete independence. This is in spirit similar to the first application of pairwise independence to reduce the amount of randomness, namely Luby's derandomization of his parallel Minimum Independent Set algorithm [20]. In our PRG-based protocol, nodes employ Nisan's *pseudo-independent block generator* [22] with a different seed in every round. We assume that nodes have no initial IDs, so we combine our protocols with a mechanism to distribute IDs to all nodes. (Such a mechanism was already presented in [15], but in our new protocols the size of the IDs is much smaller.)

The analyses of both protocols deviates sometimes significantly from previous analyses of rumor spreading protocols that use full randomness. Since we are limited to pairwise independence or pseudo-independence, we cannot employ strong tail bounds such as Chernoff-type bounds.

Our protocols are local, i.e., no information about the graph topology is needed. Further, the fact that their analysis works for such a wide range of graphs indicates that the protocols are robust. While the protocols are not quite as simple as the fully random and the quasi-random protocols, the computation a node must perform in each round involves only a constant number of arithmetic operations in the hashing-based protocol, and $\mathcal{O}(\log n)$ operations in the PRG-based protocol. The randomness requirement of the latter protocol is also by a $(\log n)$-factor higher. The hashing-based protocol is asymptotically as efficient as the PRG-based protocol on all graphs that we consider. However, only for the PRG-based protocol the constant factor in our upper bound on the rumor spreading time for strong expander graphs matches the lower bound for the fully random protocol [9].

We assume the *standard adversary model*, which was also used in [6, 15]: In each round, every informed node u chooses an index $j \in \{1, \ldots, \deg(u)\}$, and sends a message to the j-th node in its adjacency list. No edge connection information is available to u other than its adjacency list; and the order of u's neighbors in this list is determined by an oblivious adversary (before the algorithm is executed).[2] For this model, it is known that any protocol for the complete graph that uses at most $b < \log n$ random bits ($b = 0$ for a deterministic algorithm) needs at least roughly $b + n/2^b$ rounds to inform all nodes [15].

1.2 Preliminaries

Throughout the paper, $G = (V, E)$ is a connected, undirected graph on n nodes. For each node $u \in V$, we let $N(u)$ denote the set of neighbors of u, and $\deg(u) := |N(u)|$ is the degree of u. By δ and Δ we denote the minimum and maximum degrees of G, respectively; if G is a regular graph, we denote its degree by d. For any node sets $S, T \subseteq V$, we define the edge set $E(S,T) := \{\{u,v\} \in E \mid u \in S \text{ and } v \in T\}$. The volume of S is $\mathrm{vol}(S) := \sum_{u \in S} \deg(u)$.

By I_t, for $t \geq 1$, we denote the set of informed nodes at the end of round t, and $U_t := V \setminus I_t$ is the set of uninformed nodes at that time. By I_0 and U_0 we denote the corresponding sets initially, before the algorithm is executed. We assume that $I_0 = \{s\}$ for some arbitrary initial node s. By $\log x$ we denote the binary logarithm of x.

For each $1 \leq i \leq m$, let X_i be a discrete uniform random variable with (finite) range R_i. We say that the sequence of random variables $X_1, \ldots, X_m$ is *pseudo-independent with parameter* ε, if for

[2] In fact, our results hold for a slightly stronger adversary: the adversary is allowed to choose a different ordering of the neighbors of a node for each round—but these orderings must be fixed before the algorithm starts.

all $A_i \subseteq R_i$, $1 \le i \le m$,

$$\left| \mathbf{Pr}\left[\, X_1 \in A_1 \wedge \cdots \wedge X_k \in A_m \,\right] - \frac{|A_1 \times \cdots \times A_m|}{|R_1 \times \cdots \times R_m|} \right| \le \varepsilon. \tag{1.1}$$

The sequence is *ε-approximate independent*, if equation (1.1) is true as long as all sets A_i, $1 \le i \le m$, have cardinality 1. Finally, the sequence is *ε-approximate k-wise independent*, if any k random variables in that sequence are ε-approximate independent.

We recall the following Chernoff bound which can be easily derived from the Chernoff bound for binomial random variables (cf. [8, Problem 3.6]).

Lemma 1.1. *Fix any $0 < p < 1$ and let $X_1, \ldots, X_n$ be independent identical geometric random variables with $\mathbf{Pr}\left[\, X_i = k \,\right] = (1-p)^{k-1} \cdot p$, for every $k \ge 1$. Let $X = \sum_{i=1}^{n} X_i$ and $\mu = \mathbf{E}\left[\, X \,\right]$. Then it holds for all $\beta > 0$ that $\mathbf{Pr}\left[\, X \ge (1+\beta)\mu \,\right] \le \mathrm{e}^{-n\beta^2/(2(1+\beta))}$.*

Due to space limitations, several proofs are omitted from this extended abstract.

2 Description of the Protocols

Both our protocols are of the same structure. We consider only T-round protocols, in which nodes send messages only for the first T rounds. In addition to T, each protocol is parameterized by n, the number of nodes in the graph, and c, a constant to adjust the error term. We assume that all nodes know n. (As long as the first node to receive the rumor knows some upper bound n' on n, the protocols work fine, but then the required amount of randomness is a function in n'.) Moreover, all nodes have access to a common random string s, the current round number, and the parameters c and T. (We make this assumption for presentational reasons. In practice, the first node can determine s, c, and T, and then send them together with the rumor. The current round number can also be sent together with the rumor, and updated in each round. From Table 1 it is immediate that the increase in message size incurred by this additional information is dominated by the length of s, i.e., the randomness requirements of the protocol.) Whenever a node sends the rumor to one of its neighbors in round t it also sends a message containing a unique string x that we call ID. A node is *uninformed* as long as it has not received a message. Once a node receives the first message, it becomes *informed* and from then on uses the ID of the received message as its own ID. Once a node becomes informed, it ignores all further incoming messages. If in some round an uninformed node receives multiple messages, then it only considers an arbitrary one and discards all others.

We assume that in a T-round protocol, the first node to receive the rumor (in round 0) also receives the ID 0. Now, suppose a node v receives its first message with ID x_v in round t'. (Recall that v discards all other messages it receives.) Then, in round $t > t'$ node v uses the values of t, s, as well as the ID x_v to determine two functions $f_t(s, x_v)$ (called the *link function*) and $g_t(x_v)$ (called the *ID function*). The function value of f_t modulo $\deg(v)$ determines to which neighbor v sends the rumor in round t, and $g_t(x_v)$ is the ID of the message sent by v to its neighbor in round t.

Both our protocols use the same ID function g_t, which is defined by $g_t(x) := x + 2^{t-1}$.

Claim 2.1. *All messages of a T-round protocol with ID function $g_t(x) = x + 2^{t-1}$ have distinct IDs in $\left[2^T\right]$.*

Let us briefly describe the intuition why this ID function works. Consider any path $P = (u_0, u_1, \ldots, u_t)$ of length t that follows the "spread" of the rumor, i.e., if $u_{\ell-1} \ne u_\ell$, then $u_{\ell-1}$

sends the rumor to u_ℓ in round ℓ. Then, we associate with P a bitstring $S(P) = (s_1, \dots, s_t)$ where $s_\ell = 1$ if $u_\ell \neq u_{\ell-1}$ and $s_\ell = 0$ otherwise. Note that $S(P) \neq S(P')$ for any two distinct paths P and P' that the rumor has followed in the first t rounds. Since additionally $S(P)$ is precisely the ID of the message that u_{t-1} sends to u_t in round t if $u_{t-1} \neq u_t$, it follows that all messages sent in round t have distinct IDs of length t whose last bit is 1.

For every node $v \in I_{t-1}$, let $X_t(v)$ be the random variable which assumes value $j \in \{1, \dots, \deg(v)\}$ if in round t node v sends a message to its j-th neighbor (i.e., the j-th node in its adjacency list). If $v \notin I_{t-1}$, then define $X_t(v) = 0$. A protocol is *ε-approximate k-wise independent in round t*, if given any values of the random variables $X_{t'}(v)$, $t' < t$, $v \in V$, the sequence of random variables $X_t(v)$, $v \in I_{t-1}$, is ε-approximate k-wise independent. The protocol is *ε-approximate k-wise independent*, if it is so in every round. The protocol is *pseudo-independent with parameter ε in round t*, if given any values for the random variables $X_{t'}(v)$, $t' < t$, $v \in V$, the sequence of random variables $X_t(v)$, $v \in I_{t-1}$, is pseudo-independent with parameter ε.

In the following we present two T-round protocols: Our first protocol is based on pairwise independent hash functions. It is approximate pairwise independent and uses $\mathcal{O}\big(T \cdot (\log T + \log n)\big)$ random bits. Our second protocol is based on Nisan's PRG. It uses $\mathcal{O}(T \cdot \log^2 n)$ random bits, and is pseudo-independent.

2.1 The Hashing-Based Protocol

We present a simple protocol based on pairwise independent hash functions that achieves approximate pairwise independence. A family $\mathcal{H}$ of hash functions $h : [M] \to [N]$ is called *ε-approximate k-wise independent*, if for a randomly chosen function $h \in \mathcal{H}$ the sequence of hash values $h(x)$, $x \in [M]$, is ε-approximate k-wise independent. (In the case $\varepsilon = 0$, $\mathcal{H}$ is called k-wise independent.)

Claim 2.2. *For all integer functions $R = R(n), M = M(n), N = N(n)$, there is an $\mathcal{O}(1/R)$-approximate pairwise independent family $\mathcal{H}_{M,N,R}$ of hash functions $h : [M] \to [N]$, such that each hash function in $\mathcal{H}_{M,N,R}$ can be described with $\mathcal{O}(\log R + \log\log M)$ bits.*

The construction of the hash class $\mathcal{H}_{M,N,R}$ is standard: Every function in $\mathcal{H}_{M,N,R}$ has the same form $h_{a,b,p}(x) = (ax + b) \bmod p \bmod N$, where p is a prime in $\{M', 2M'\}$, $M' = \lceil R \cdot \log M \rceil$, and $a, b \in [p]$. The random linear functions over $[p]$ yield pairwise independence over $[p]$ for all pairs of keys that are not in the same congruence class modulo p. Since p is a random prime for a randomly chosen hash function, the probability that two keys are congruent modulo p is small, and we obtain approximate pairwise independence.

Our protocol uses a sequence of randomly chosen hash functions. More precisely, the random string s used by the protocol is a sequence of T hash functions, i.e., $s = (h_1, \dots, h_T)$, where $h_i \in \mathcal{H}_{2^T, n^c, n^{3c}}$ is chosen uniformly (and independently) at random. The protocol uses the link function $f_t\big(s, x_v\big) = h_t(x_v)$. That is, any node $v \in I_{t-1}$ with ID x_v sends the rumor in round t to the $(\ell_t + 1)$-th node in its neighbor list, where $\ell_t = \big(h_t(x_v)\big) \bmod \deg(v)$. (Recall that it also sends a message with the ID $g_t(x_v)$ along with the rumor.)

Lemma 2.3. *The hashing-based T-round protocol is $\mathcal{O}(1/n^c)$-approximate pairwise independent and uses $\mathcal{O}(T \cdot (\log T + \log n))$ random bits.*

Proof. By Claim 2.2, a hash function $h_i \in \mathcal{H}_{2^T, n^c, n^{3c}}$ can be described with $\mathcal{O}(\log T + \log n)$ random bits. Hence, the protocol uses T times that many random bits.

Now fix some round number $t \leq T$ and some hash functions $h_1, \dots, h_{t-1} \in \mathcal{H}_{2^T, n^c, n^{3c}}$. Then the execution of the protocol during the first $t-1$ rounds is uniquely determined by the choice of $h_1, \dots, h_{t-1}$. Now, let $u, v \in I_{t-1}$ be distinct nodes and fix arbitrary $y_u \in [\deg(u)]$ and $y_v \in [\deg(v)]$.

Suppose that u and v obtained IDs x_u and x_v, respectively, when they received their first messages. By Claim 2.1, $x_u \neq x_v$. Then, even though the execution of the protocol during the first $t-1$ rounds is fixed (and the values of x_u and x_v may depend on $h_1, \ldots, h_{t-1}$), $h_t(x_u)$ and $h_t(x_v)$ are $\mathcal{O}(1/n^{3c})$-approximate pairwise independent random variables with range $[n^c]$. Thus, for every pair $(z_u, z_v) \in [n^c]^2$ the probability that $h_t(x_u) = z_u$ and $h_t(x_v) = z_v$ is at most $1/n^{2c} + \mathcal{O}(1/n^{3c}) = (1 + \mathcal{O}(1/n^c))/n^{2c}$. Hence,

$$\begin{aligned}
&\mathbf{Pr}\left[\, h_t(x_u) \bmod \deg(u) = y_u \,\wedge\, h_t(x_v) \bmod \deg(v) = y_v \,\right] \\
&\leq \left(1 + \mathcal{O}\left(\frac{1}{n^c}\right)\right) \cdot \frac{1}{n^{2c}} \cdot \lceil n^c/\deg(u)\rceil \cdot \lceil n^c/\deg(v)\rceil \\
&\leq \left(1 + \mathcal{O}\left(\frac{1}{n^c}\right)\right) \cdot \left(\frac{1}{\deg(u)} + \frac{1}{n^c}\right) \cdot \left(\frac{1}{\deg(v)} + \frac{1}{n^c}\right) = \frac{1}{\deg(u)\cdot\deg(v)} + \mathcal{O}\left(\frac{1}{n^c}\right).
\end{aligned}$$

A similar calculation gives a lower bound of $\frac{1}{\deg(u)\deg(v)} - \mathcal{O}(\frac{1}{n^c})$ on the above probability. □

In order to select their random neighbors, nodes have to randomly choose hash functions from the class $\mathcal{H}_{2^T,n^c,n^{3c}}$. Evaluating a hash function involves only a few integer arithmetic operations with integers of value at most 2^T. However, in order to select a random hash function, one also has to select a random prime of logarithmic length (in n). This can be avoided, though, by using hash functions in $\mathcal{H}_{2^T,n^c,R}$, where $R = \max\{n^{3c}, 2^T\}$, as in this case it can be shown that the hash functions do not need to use *random* primes. Doing this increases the total number of random bits used by the protocol to $\mathcal{O}\big(T \cdot (T + \log n)\big)$. For most of the graph topologies analyzed in this paper, $T = \mathcal{O}(\log n)$, so in this case sampling random primes can be avoided without affecting the randomness requirement.

2.2 The PRG-Based Protocol

Let m, ℓ, and k be positive integers and let $\varepsilon > 0$. Let $B : \{0,1\}^m \to \big(\{0,1\}^\ell\big)^k$ be some mapping. For $0 \leq i < k$ and $s \in \{0,1\}^m$, we define $B_i(s)$ to be the projection of $B(s)$ to the $(i+1)$-th component. That is, if $B(s) = y_0 y_1 \ldots y_{k-1}$, where each $y_j \in \{0,1\}^\ell$ is a block of length ℓ, then $B_i(s) = y_i$. The mapping B is a *pseudo-independent block generator with parameter ε*, if for a randomly chosen *seed* $w \in \{0,1\}^m$ the random variables $B_0(w), \ldots, B_{k-1}(w)$ are pseudo-independent with parameter ε.

Theorem 2.4 ([22]). *There is a constant $\alpha > 0$ such that for any integers ℓ and $k \leq \ell$ there is a pseudo-independent block generator $B^{(\ell,k)} : \{0,1\}^{\alpha\cdot\ell\cdot k} \to \big(\{0,1\}^\ell\big)^{2^k}$ with parameter $2^{-\ell}$.*

The block generator is based on pairwise independent hash functions. In particular the random seed is an ℓ-bit string x and a sequence of k hash functions $h_1, \ldots, h_k$ from a family of pairwise independent hash functions with universe and range of size roughly 2^ℓ. In order to determine a random value $B_i(x)$ it suffices to evaluate the composition of up to k hash functions at point x. (Hence, it is not required for nodes to generate the entire pseudo-random string.)

Let α be the constant from Theorem 2.4 and $\ell = \lceil c \cdot \log n\rceil$. Our T-round protocol uses a random string $s = \big((p_1, w_1), \ldots, (p_T, w_T)\big)$, where each $w_i \in \{0,1\}^{\alpha\cdot\ell^2}$ and p_i is a random prime in $\{2^{\ell-1}, \ldots, 2^\ell\}$. As previously, nodes send messages in order to distribute IDs using the ID function $g_t(x_v)$. If a node $v \in I_{t-1}$ has ID x_v, it uses the link function $f_t(s, x_v) = B^{(\ell,\ell)}_{x_v \bmod p_t}(w_t)$ to determine to which neighbor to send the rumor to in round t. That is, if prior to round t node v receives its first message with ID x_v, then in round t it determines $b = x_v \bmod p_t$ and looks up the $(b+1)$-th block of the random string determined by the block-generator $B^{(\ell,\ell)}$ with seed w_t. (This random string consists of 2^ℓ blocks of length ℓ.)

Lemma 2.5. *The PRG-based T-round protocol is pseudo-independent with parameter $\mathcal{O}(T/n^{c-2})$ and uses $\mathcal{O}(T \cdot \log^2 n)$ random bits.*

As in the hashing based protocol, one can avoid generating a random prime for each round by choosing $\ell = \max\{T, \lceil c \cdot \log n \rceil\}$ and defining $f_t(s, x_v) = B^{(\ell,\ell)}_{x_v}(w_t)$ (thus, nodes do not consider their IDs modulo a prime). This way, the protocol needs $\mathcal{O}\big(T(T^2 + \log^2 n)\big)$ random bits, which is no different than before as long as $T = \mathcal{O}(\log n)$.

3 Analysis of the Hashing-Based Protocol

In Section 3.1 we study the progress achieved in a single round of an approximate pairwise independent protocol. Then in Section 3.2, we use this result to derive a general upper bound for the hashing-based protocol in terms of graph conductance.

3.1 Analysis of a Single Round

The next lemma provides lower bounds on the number of nodes informed in a single round of an ε-approximate pairwise independent protocol, for a sufficiently small ε. Specifically, it counts the nodes informed by rumor transmissions along the edges in a given subset F of the set $E(I_t, U_t)$ of edges between informed and uninformed nodes. Note that the expected number of such transmissions in the fully random protocol is $\sum_{\{u,v\} \in F\colon u \in I_t, v \in U_t} 1/\deg(u)$, which is at least $|F|/\Delta$ and at most $|F|/\delta$. For an ε-approximate pairwise independent protocol, the expected number of such transmissions is different by at most $\varepsilon\Delta \cdot |F|$. Let X_u, for each $u \in I_t$, denote the neighbor $v \in N(u)$ that u chooses in round $t+1$. By the law of total probability, for any node $u' \in I_t \setminus \{u\}$,

$$\begin{aligned}\mathbf{Pr}\left[\, X_u = v \,\right] &= \sum_{v' \in N(u')} \mathbf{Pr}\left[\, X_u = v \wedge X_{u'} = v' \,\right] \le |N(u')| \cdot \left(\frac{1}{\deg(u) \cdot \deg(u')} + \varepsilon \right) \\ &\le 1/\deg(u) + \varepsilon\Delta,\end{aligned}$$

and similarly, $\mathbf{Pr}\left[\, X_u = v \,\right] \ge 1/\deg(u) - \varepsilon\Delta$.

Lemma 3.1. *Consider an ε-approximate pairwise independent protocol with $\varepsilon = o(1/n^3)$. Fix a round $0 \le t < T$ and the set I_t of informed nodes before round $t+1$ begins. Fix also an arbitrary set of edges $F \subseteq E(I_t, U_t)$. Let J be the set of nodes that become informed in round $t+1$ if we consider only transmissions of the rumor along the edges in F.*

(a) $\mathbf{Pr}\left[\, |J| \ge 1 \,\right] \ge (1 - o(1)) \dfrac{|F|/\Delta}{2|F|/\delta + 6}$.

(b) If $|F| \ge 16\Delta$, then $\mathbf{Pr}\left[\, |J| \ge \dfrac{1}{19} \cdot \dfrac{\delta^2}{\Delta^2} \cdot \dfrac{|F|}{\Delta} \,\right] \ge \dfrac{1}{2} - o(1)$.

(c) For any $v \in U_t$, let $\gamma_v := |\{u \in V \colon \{u,v\} \in F\}|$ be the number of edges in F that are incident to v. If $\sum_{v \in V} \gamma_v^2 = o(\Delta \cdot |F|)$, and $|F| = \omega(\Delta)$, and $\Delta/\delta = 1 + o(1)$, then

$$\mathbf{Pr}\left[\, |J| \ge (1 - o(1)) \cdot \frac{|F|}{\Delta} \,\right] \ge 1 - o(1).$$

Remark 3.2. *We will use Lemma 3.1 in the analysis of the PRG-based protocol as well; in fact, (c) is only used there. Recall from Lemma 2.5 that the PRG-based protocol is pseudo-independent with parameter $\varepsilon = \mathcal{O}(T/n^{c-2})$, and thus it is also ε-approximate pairwise independent.*

We now give an outline of the proof of Lemma 3.1. For any uninformed node $v \in U_t$, let Z_v denote the number of rumor transmissions to v through edges in F in round $t+1$. To prove (a) we bound the probability that $\sum_v Z_v = 0$, which is the same as the probability that $|J| = 0$, using Chebyshev's inequality. Note that we cannot use stronger concentration tools, such as Chernoff bounds, since we only have (approximate) pairwise independence among the choices of different nodes in a round. In general, $\sum_v Z_v \geq |J|$, because a node may receive the rumor more than once in a round. Thus we cannot prove (b) and (c) just by showing a lower bound on $\sum_v Z_v$. In addition to lower-bounding $\sum_v Z_v$, we also upper-bound $\sum_v Z_v^2$. For the latter bound we use Markov's inequality (we cannot apply Chebyshev's inequality, as 4-wise independence among the node's choices is needed for that). Then we apply Cauchy-Schwartz's inequality to lower-bound $|J| = \sum_v \mathbf{1}_{Z_v>0}$ by $\left(\sum_v Z_v\right)^2/\sum_v Z_v^2$.

3.2 An Upper Bound in Terms of Conductance

Let G be an arbitrary graph and let $S \subseteq V$ be any set of size $0 < |S| < n$. The conductance of S is defined as $\phi(S) = \frac{|E(S,V\setminus S)|}{\min\{\mathrm{vol}(S),\mathrm{vol}(V\setminus S)\}}$. The conductance of G is defined as the minimum conductance over all sets S,

$$\phi(G) = \min_{S\subseteq V, 0<|S|<n} \Phi(S) = \min_{S\subseteq V, 0<|S|<n} \frac{|E(S, V\setminus S)|}{\min\{\mathrm{vol}(S), \mathrm{vol}(V\setminus S)\}}.$$

Note that the conductance of a d-regular graph G is $\phi(G) = \min_{S\subseteq V, 0<|S|<n} \frac{|E(S,V\setminus S)|}{d\cdot\min\{|S|, n-|S|\}}$.

Theorem 3.3. *For any graph with conductance ϕ and $\Delta/\delta = \mathcal{O}(1)$, the hashing-based protocol informs all nodes in $\mathcal{O}((1/\phi)\log n)$ rounds w.h.p. using $\mathcal{O}((1/\phi)\log^2 n)$ random bits.*

Proof. We choose the parameters of the hashing-based protocol to be $T = \Theta((1/\phi)\log n)$ and $c > 3$. From Lemma 2.3 then it follows that the protocol is $o(1/n^3)$-approximate pairwise independent, and the total number of random bits used is $\mathcal{O}(T \cdot (\log T + \log n)) = \mathcal{O}((1/\phi)\log^2 n)$, since $1/\phi = \mathcal{O}(n^2)$ in any connected graph.

The proof is divided into four phases according to the number of informed nodes $|I_t|$.

Phase 1: $1 \leq |I_t| \leq 16(\Delta/\delta)\cdot(1/\phi)$. This phase is divided into several subphases. For every $1 \leq i \leq \log(16(\Delta/\delta)\cdot(1/\phi))$, subphase i begins when the number of informed nodes is at least 2^{i-1} and ends when this number is at least 2^i. Assume that we are at the beginning of the i-th subphase. Fix an arbitrary round t of the i-th subphase and the set of informed nodes I_t; thus, $2^{i-1} \leq |I_t| < 2^i$. We consider the number of nodes that become informed in round $t+1$. Applying Lemma 3.1(a) with $F = E(I_t, U_t)$ gives

$$\mathbf{Pr}\left[\,|I_{t+1} \setminus I_t| \geq 1\,\right] \geq (1 - o(1))\frac{|E(I_t, U_t)|/\Delta}{2|E(I_t, U_t)|/\delta + 6}. \tag{3.1}$$

Suppose first that $|E(I_t, U_t)|/\delta \geq 6$. Then, the above probability is at least

$$(1 - o(1))\cdot(\delta/\Delta)\cdot(1/3) \geq (\delta/\Delta)^2\cdot(1/49)\cdot|I_t|\cdot\phi =: p,$$

where the inequality follows from the upper bound on $|I_t|$. On the other hand, if $|E(I_t, U_t)|/\delta \leq 6$, then the probability in equation (3.1) is at least

$$(1 - o(1)) \cdot |E(I_t, U_t)|/(18\Delta) \geq (1 - o(1)) \cdot \phi\, \delta\, |I_t|/(18\Delta) \geq p.$$

Therefore, the expected time to increase $|I_t|$ from 2^{i-1} to 2^i is at most

$$2^{i-1} \cdot \frac{1}{p} \leq 2^{i-1} \cdot \frac{1}{(1/49)\,(\delta/\Delta)^2 \cdot \phi\, 2^{i-1}} = 49 \cdot (\Delta/\delta)^2 \cdot (1/\phi) =: \tau.$$

By Markov's inequality,

$$\mathbf{Pr}\left[\, |I_{t+2\tau}| \leq 2^i \;\mid\; |I_t| \geq 2^{i-1} \right] \leq 1/2.$$

Hence the time to complete Phase 1 can be upper bounded by $\tau = \mathcal{O}((1/\phi))$ multiplied with the sum of $\log(16(\Delta/\delta) \cdot (1/\phi)) = \mathcal{O}(\log n)$ independent geometric random variables each with parameter $1/2$. Applying a Chernoff bound for the sum of independent geometric random variables (Lemma 1.1) yields that the number of rounds required for Phase 1 is at most $\mathcal{O}((1/\phi) \cdot \log n)$ w.h.p.

Phase 2: $16(\Delta/\delta) \cdot (1/\phi) \leq |I_t| \leq n/2$. Fix a round t and the set of informed nodes I_t. We apply Lemma 3.1(b), with $F = E(I_t, U_t)$. Note that the precondition $|F| \geq 16\Delta$ is satisfied, as

$$|F| = |E(I_t, U_t)| \geq \phi \cdot \delta \cdot |I_t| \geq \phi \cdot \delta \cdot 16(\Delta/\delta) \cdot (1/\phi) = 16\Delta.$$

Hence we conclude from Lemma 3.1(b),

$$\mathbf{Pr}\left[|I_{t+1} \setminus I_t| \geq \frac{1}{19} \cdot \frac{\delta^3}{\Delta^3} \cdot \phi \cdot |I_t| \right] \geq \frac{1}{2} - o(1),$$

and thus, with probability $1/2 - o(1)$, $|I_{t+1}| \geq \left(1 + \frac{1}{19} \cdot \frac{\delta^3}{\Delta^3} \cdot \phi\right) \cdot |I_t|$. So, the number of rounds until we have $|I_t| \leq n/2$ can be upper bounded by the sum of $\log_{1+\frac{1}{19}\cdot\frac{\delta^3}{\Delta^3}\cdot\phi}(n/2) = \mathcal{O}((1/\phi)\log n)$ independent geometric random variables with parameters $1/2 - o(1)$. Using again the Chernoff bound in Lemma 1.1 we obtain that Phase 2 is completed within at most $\mathcal{O}((1/\phi)\log n)$ rounds w.h.p.

Phase 3: $n/2 \leq |I_t| \leq n - 16(\Delta/\delta) \cdot (1/\phi)$. The analysis is the same as in Phase 2 with the roles of I_t and U_t switched.

Phase 4: $n - 16(\Delta/\delta) \cdot (1/\phi) \leq |I_t| \leq n$. Again, the analysis is the same as in Phase 1 with the roles of I_t and U_t switched.

Since each of the four phases requires only $\mathcal{O}((1/\phi) \cdot \log n)$ rounds w.h.p., the result follows by applying the union bound. □

4 Analysis of the PRG-Based Protocol

We now consider graph families with strong expansion properties. We prove that by increasing the number of random bits slightly, from $\mathcal{O}(\log^2 n)$ to $\mathcal{O}(\log^3 n)$, we can obtain precise time bounds that are comparable to the ones for the fully random protocol.

We describe a condition that implies such tight bounds, in terms of the following version of conductance (see, e.g., [18]),

$$\tilde{\phi}(G) := \min_{S \subseteq V, 0<|S|<|V|} \frac{|E(S, V \setminus S)| \cdot \mathrm{vol}(V)}{\mathrm{vol}(S) \cdot \mathrm{vol}(V \setminus S)}.$$

This definition is slightly different than the one given in Section 3.2, but it is easy to verify that $\phi(G) \leq \tilde{\phi}(G) \leq 2 \cdot \phi(G)$.

The following theorem concerns so-called *strong expanders*, which are almost-regular graphs for which the conductance $\tilde{\phi}(G)$ tends to one.

Theorem 4.1. *For any graph with $\Delta/\delta = 1 + o(1)$ and $\tilde{\phi} \geq 1 - o(1)$, the PRG-based protocol informs all nodes in $\log n + \ln n + o(\log n)$ rounds with probability $1 - o(1)$, using $\mathcal{O}(\log^3 n)$ random bits in total.*

The proof of Theorem 4.1 is similar to that of Theorem 3.3. We consider different phases according to the size of I_t and apply Lemma 3.1(c) to lower bound the increase of the number of informed nodes.

It was shown in [9, Theorem 1 & Lemma 2] that on any d-regular graph with $d = \omega(1)$, the fully random protocol requires at least $\log n + \ln n - o(\log n)$ rounds to spread the rumor to all n nodes. We observe the following simple corollary of Theorem 4.1.

Corollary 4.2. *Under the same assumptions as in Theorem 4.1, all nodes are informed by the fully random protocol within $\log n + \ln n + o(\log n)$ rounds with probability $1 - o(1)$.*

Theorem 4.1 can be used to obtain tight bounds for several interesting graph families. For that, we consider the algebraic expansion of graphs. For any graph $G = (V, E)$, let M be the normalized adjacency matrix of G, i.e., $M_{i,j} = 1/\sqrt{\deg(i)\deg(j)}$ if $\{i, j\} \in E$ and $M_{i,j} = 0$ otherwise. Moreover, let $\lambda_2 = \lambda_2(G)$ be the second largest eigenvalue of M. Since M is real and symmetric, λ_2 is a real number. Also, since $\tilde{\phi}(G) \geq 1 - \lambda_2$ [18, Theorem 5.3], we can apply Theorem 4.1 to obtain the following result.

Corollary 4.3. *For any graph with $\lambda_2 = o(1)$ and $\Delta/\delta = 1 + o(1)$, the PRG-based protocol informs every node in $\log n + \ln n + o(\log n)$ rounds with probability $1 - o(1)$, using $\mathcal{O}(\log^3 n)$ random bits in total.*

Notice that this corollary can be applied to regular graphs. In particular, d-regular Ramanujan graphs [19] satisfy the preconditions of the corollary. Moreover, we can use Corollary 4.3 to obtain a time bound for certain families of random graphs.

Corollary 4.4. *In the $G(n, p)$ random graph with $p = \omega(\log n/n)$, the PRG-based protocol informs every node in $\log n + \ln n + o(\log n)$ rounds with probability $1 - o(1)$, using $\mathcal{O}(\log^3 n)$ random bits in total.*

Proof. Since $p = \omega(\log n/n)$, we have $\lambda_2 = o(1)$ by [3, Theorem 1.2], and $\Delta/\delta = 1 + o(1)$ by a Chernoff bound. Applying Corollary 4.3 then yields the claim. □

References

[1] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory and IEEE/ACM Transactions on Networking*, 52(6):2508–2530, 2006.

[2] F. Chierichetti, S. Lattanzi, and A. Panconesi. Almost tight bounds on rumour spreading with conductance. In *42nd ACM Symposium on Theory of Computing (STOC'10)*, pages 399–408, 2010.

[3] A. Coja-Oghlan. On the Laplacian eigenvalues of $G_{n,p}$. *Combinatorics, Probability & Computing*, 16(6):923–946, 2007.

[4] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry. Epidemic algorithms for replicated database maintenance. In *6th ACM Symposium on Principles of Distributed Computing (PODC'87)*, pages 1–12, 1987.

[5] B. Doerr and M. Fouz. A time-randomness tradeoff for quasi-random rumour spreading. *Electronic Notes in Discrete Mathematics*, 34:335–339, 2009.

[6] B. Doerr, T. Friedrich, and T. Sauerwald. Quasirandom rumor spreading. In *19th ACM-SIAM Symposium on Discrete Algorithms (SODA'08)*, pages 773–781, 2008.

[7] B. Doerr, T. Friedrich, and T. Sauerwald. Quasirandom rumor spreading: Expanders, push vs. pull and robustness. In *36th International Colloquium on Automata, Languages, and Programming (ICALP'09)*, pages 366–377, 2009.

[8] D. Dubhashi and A. Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms*. Cambridge University Press, 2009.

[9] R. Elsässer and T. Sauerwald. On the runtime and robustness of randomized broadcasting. *Theoretical Computer Science*, 410(36):3414–3427, 2009.

[10] U. Feige, D. Peleg, P. Raghavan, and E. Upfal. Randomized broadcast in networks. *Random Structures and Algorithms*, 1(4):447–460, 1990.

[11] N. Fountoulakis and A. Huber. Quasirandom rumor spreading on the complete graph is as fast as randomized rumor spreading. *SIAM Journal on Discrete Mathematics*, 23(4):1964–1991, 2009.

[12] N. Fountoulakis, A. Huber, and K. Panagiotou. Reliable broadcasting in random networks and the effect of density. In *29th IEEE Conference on Computer Communications (INFOCOM'10)*, pages 2552–2560, 2010.

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

[14] G. Giakkoupis. Tight bounds for rumor spreading in graphs of a given conductance. In *28th International Symposium on Theoretical Aspects of Computer Science (STACS'11)*, pages 57–68, 2011.

[15] G. Giakkoupis and P. Woelfel. On the randomness requirements of rumor spreading. In *22nd ACM-SIAM Symposium on Discrete Algorithms (SODA'11)*, pages 449–461, 2011.

[16] M. Harchol-Balter, F. T. Leighton, and D. Lewin. Resource discovery in distributed networks. In *18th ACM Symposium on Principles of Distributed Computing (PODC'99)*, pages 229–237, 1999.

[17] R. Karp, C. Schindelhauer, S. Shenker, and B. Vöcking. Randomized rumor spreading. In *41st IEEE Symposium on Foundations of Computer Science (FOCS'00)*, pages 565–574, 2000.

[18] L. Lovász. Random walks on graphs: A survey. *Combinatorics, Paul Erdös is Eighty*, 2:1–46, 1993.

[19] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277,

1988.

[20] M. Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM Journal on Computing*, 15(4):1036–1053, 1986.

[21] D. Mosk-Aoyama and D. Shah. Fast distributed algorithms for computing separable functions. *IEEE Transactions on Information Theory*, 54(7):2997–3007, 2008.

[22] N. Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.

[23] B. Pittel. On spreading a rumor. *SIAM Journal on Applied Mathematics*, 47(1):213–223, 1987.

[24] R. van Renesse, Y. Minsky, and M. Hayden. A gossip-style failure detection service. In *15th IFIP International Conference on Distributed Systems Platforms (Middleware)*, pages 55–70, 1998.