Randomized Protocols for Information Dissemination

by
Thomas Sauerwald

University of Paderborn, 2008

Randomized Protocols for Information Dissemination

Randomized Protocols for Information Dissemination

by
Thomas Sauerwald

Schriftliche Arbeit zur Erlangung des Grades
Doktor der Naturwissenschaften
an der Fakultät für Elektrotechnik, Informatik und Mathematik
der Universität Paderborn

August 26, 2008

University of Paderborn, 2008

ACKNOWLEDGEMENTS

First of all, I would like to thank my main supervisor Robert Elsässer for the countless stimulating discussions during the last three years. Moreover, I wish to express my thanks to Burkhard Monien and Christian Scheideler for reviewing this thesis. Also a thank you goes to Michael Dellnitz for being my Ph.D. co-advisor.

Moreover, I am thankful to Petra Berenbrink and Henning Meyerhenke for the collaboration in Paderborn. I would also like to thank Peter Bürgisser and Friedhelm Meyer auf der Heide for several helpful discussions.

Some research was done while I was visiting the University of Cyprus, the University of Liverpool and the Max-Planck Institute for Computer Science in Saarbrücken. Therefore, a thank you goes to my collaborators there, Marios Mavronicolas, Leszek Gascienec, Benjamin Doerr and Tobias Friedrich.

I am grateful to the Graduate School of the Paderborn Institute for Scientific Computing for their financial support during the last three years which enabled me the participation at several international conferences and workshops.

I am also indebted to David Cook, Dominic Dumrauf, Henning Meyerhenke, Florian Schoppmann, Ulf-Peter Schroeder, Tobias Tscheuschner and Sven-Ake Wegner for the reading of preliminary parts of this thesis.

Last but not least, I would like to thank my family for their great support.

CONTENTS

1. INTRODUCTION

1.1 Motivation and Framework

Nowadays we are surrounded by many large networks of different kinds. These networks include physical networks such as the Internet, parallel computers and embedded systems. Also of increasing importance are wireless networks formed by mobile phones or robots. First of all, these networks are often extremely large (e.g., the number of users of the Internet is currently estimated to be 1.5 billion [CIA08]). Another important aspect is that these networks dynamically evolve over time. These two reasons explain why it is virtually impossible to control such large networks centrally and for many important tasks *local* protocols have to be developed.

One fundamental and overarching problem in the context of large networks is the one of efficient dissemination of information [HKP+04]. Various tasks such as broadcasting, gossiping, sorting, routing, leader election, load balancing, etc. can be regarded as special cases of information dissemination. For many of those tasks, several *deterministic* and *randomized* algorithms have been proposed and analyzed [DGH+87, Lei92, MR95, HKP+04, MU05, BGPS06]. One drawback of many deterministic algorithms is that they often require a central control for coordination. Additionally, it is difficult for them to handle changes in the topology like removing (or adding) nodes (or edges).

Randomized algorithms (or protocols) enable the nodes to spread entities (e.g., packets to be routed) rather uniformly to other nodes by making only *local* decisions. This ensures that these algorithms are fairly robust under dynamic changes of the topology. Moreover, while an adversary is able to foil a *deterministic* algorithm by constructing some special input, it may be often quite difficult to devise an input which defeats a *randomized* algorithm. For some applications including Monte-Carlo simulations and cryptographic algorithms [MU05], some randomized algorithms are also more efficient than the best known deterministic ones. Furthermore, especially in a distributed environment, the randomized approaches are much simpler and easier to implement than its deterministic counterparts.

A price one has to pay is that the result of a randomized protocol can occasionally be of poor quality (or even incorrect). However, as the probability of failure is in many cases rather small, the afore-mentioned advantages may well be worthwhile.

In this thesis we present results on four randomized protocols in the field of information dissemination. All of our findings demonstrate the great use of randomization. Some of them also show that apparently minor changes in the protocol may sometimes lead to vast improvements.

1.2 Outline of our Results

A widely used paradigm in the design of randomized protocols is that each decision is made *independently* of *all* previous decisions in order to achieve maximum fault-tolerance against crashes, resets or changes in the topology. One such protocol is the classical push algorithm for broadcasting a rumor initially known by some node in a network. This is done by letting each node which knows the rumor forward it to some random neighbor in each step. In Chapter 3 we analyze the runtime of this algorithm, i.e., the number of iterations before all nodes know the rumor, on general networks and identify the network on which the performance is (approximately) worst. In Chapter 4 we show that the push algorithm performs asymptotically optimal on several Cayley graphs.

The random walk [Lov93] is another process where all decisions are made independently. To be more specific, a random walk moves in each iteration from its current location to some random neighbor. Random walks are at the heart of many algorithms for graph exploration, load balancing or sampling [Sin93, MR95]. For the design and analysis of these algorithms, the following two parameters of random walks are of great importance. The *mixing time* is the minimum number of steps before the distribution of the random walk has approached the equilibrium distribution up to some deviation. Many algorithms rely on a small mixing time to sample certain objects efficiently [Sin93, MR95, MU05]. The *cover time* is the expected number of steps to visit every node in a network and is therefore relevant for graph exploration [MR95].

We study the mixing time, the cover time and their relationship to the push algorithm in Chapter 5. One of our results is that the runtime of the push algorithm is upper bounded by the mixing time on any regular graph. While intuitions about the relationship between randomized rumor spreading and the cover time were already mentioned [FPRU90, CRR$^+$97], no formal results were known so far. We fill this gap by presenting a comprehensive collection of upper and lower bounds relating these processes. As our main result, we prove that on every regular network of large degree, the cover time and runtime of the push algorithm differ always by a factor of approximately the number of nodes n. This contrasts with examples demonstrating a much weaker correspondence on sparse networks.

All random protocols mentioned so far are composed of sequential and independent random decisions. However, it may sometimes be advantageous to allow dependencies between the nodes' decision. A classic example is the so-called *power of two choices* in the balls-and-bins model [MU05], which has been intensively studied for *centralized* load balancing [MU05]. In the original model, we sequentially place n balls (representing jobs) into n bins (representing servers) by putting each ball into a randomly chosen bin. It was found out that by allowing in each step a ball to choose among two random bins and taking the bin with the smallest load, the maximum load drops from $\Theta(\log n / \log \log n)$ to $\Theta(\log \log n)$ [ABKU99, MU05]. In this thesis we demonstrate a very similar effect in a *distributed* environment.

We consider so-called *randomized smoothing networks*, which consist of balancers and wires. Such a network receives indivisible jobs (called tokens) at w different input wires and routes them asynchronously to servers residing at the w output wires. Wires may

be connected by certain switches called *balancers* which have two input and two output wires. Every balancer is initially directed randomly to one of its two output wires and tokens arriving at a balancer are forwarded alternately to the output wires. This local rule guarantees that each balancer distributes the arriving tokens as uniformly as possible. In particular, this improves on a balancer that would forward each token to a randomly chosen output wire. The *smoothness* of a network is the maximum discrepancy between the number of tokens arriving at two different output wires.

In Chapter 7 we study randomized smoothing networks and resolve all three problems posed by Herlihy and Tirthapura in a recent work [HT06a]. First, we prove that the smoothness of a block network (which is isomorphic to the cube-connected-cycles) is $\log\log w + \Theta(1)$ with high probability. (The previously best known upper bound was of order $\mathcal{O}(\sqrt{\log w})$ [HT06a] and no lower bound was known so far.) In our main result we consider the cascade of two block networks. We prove that the smoothness is 17 with high probability. Note that this represents a vast improvement on a network where every balancer would send each token to one of its output wires chosen uniformly at random. The smoothness of such a network would be only $\Omega(\log w/\log\log w)$ (assuming that w tokens arrive), even for the sequential cascade of two block networks. We also provide a negative result: no randomized smoothing network of reasonable size can achieve 1-smoothness. This implies a separation between randomized and deterministic smoothing networks, since there are small 1-smoothing networks with a *global* (deterministic) initialization of all balancers [KP92, Klu94].

Then we consider a scenario where random initialization is combined with *adversarial* choices. We revisit the broadcast problem and propose a new model called *quasirandom rumor spreading* in Chapter 6. In our new model we assume that each node has a cyclic list of its neighbors, which may be completely specified by some adversary. Once a node becomes informed, it starts at a random position of the list, but from then on informs its neighbors in the order of the list. Surprisingly, irrespective of the orders of the lists, a bound of $\mathcal{O}(\log n)$ on the runtime can be shown for hypercubes and random graphs with n nodes. This bound includes sparsely connected random graphs, where the classical push algorithm needs $\Theta(\log^2 n)$ steps. Hence our new model achieves the same or even better performance than the classical model despite the presence of the adversary (and consequently, the reduced amount of randomness). However, each node must be equipped with some memory to keep track of the current list position. With this respect, the quasirandom model is closely related to the modified push&pull algorithm of [ES08a]. Roughly speaking, it was shown that if every vertex is able to remember the nodes chosen in the most recent three steps, then the number of generated transmissions during the broadcast procedure can be reduced from $\mathcal{O}(n(\log\log n + \log n/\log(pn))$ to $\mathcal{O}(n\log\log n)$ on random graphs with edge probability p.

To summarize, most our results provide evidence for the power of randomization for efficient information dissemination in large networks. Beyond this, the findings described in the previous two paragraphs demonstrate that apparently minor changes in the protocols may result in surprisingly vast improvements: from $\Theta(\log^2 n)$ to $\Theta(\log n)$ in case of the quasirandom push algorithm and from $\Theta(\log w/\log\log w)$ to 17 in case of the cascade of

two block networks.

1.3 Publications

The results presented here are published in parts as joint work in the Proceedings of the *24th International Symposium on Theoretical Aspects of Computer Science (STACS'07)* [ES07], the Proceedings of the *18th International Symposium on Algorithms and Computation (ISAAC'07)* [Sau07], the Proceedings of the *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'08)* [DFS08], the Proceedings of the *27th Annual ACM-SIGOPT Principles of Distributed Computing (PODC'08)* [MS08] (to appear) and the Journal of *Theoretical Computer Science, 2008* [ES08a] (to appear). The paper [Sau07] received the *Best-Student-Paper-Award* and was invited to a special issue of *Algorithmica*.

Moreover, some results have been presented at the *Workshop on Algorithms'08 in Kiel*, at the *Dagstuhl Seminar: Probabilistic Methods in the Design and Analysis of Algorithms*, and as invited talks at the *Max Planck Institute for Computer Science in Saarbrücken* and the *Algorithms & Complexity Group at the University of Liverpool*.

Some results obtained while I was working to my PhD are not included in this thesis. These have been published (or are accepted for publication) in the following proceedings: *17th International Symposium on Algorithms and Computation (ISAAC'06)* [MS06], *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'08)* [ES08a], in *22nd IEEE International Parallel and Distributed Processing Symposium (IPDPS'08)* [MMS08] and *15th International Colloquium on Structural Information and Communication Complexity (SIROCCO'08)* [SS08]. The paper [MMS08] received the *Best-Algorithms-Paper-Award* and was invited to a special issue of *Journal of Parallel and Distributed Computing*.

1.4 Organization

We begin with a description of some probabilistic tools and graph-theoretical notation in Chapter 2. In Chapter 3 we introduce the push algorithm and investigate its runtime on general graphs. In Chapter 4 we analyze the performance of the push algorithm on Cayley graphs. Chapter 5 relates the runtime of the push algorithm to the mixing time and cover time of random walks. We propose and analyze a quasirandom rumor spreading algorithm in Chapter 6. Finally, Chapter 7 comprises our results about randomized load balancing.

2. PROBABILISTIC PRELIMINARIES

2.1 Combinatorial Inequalities

We give a list of several combinatorial inequalities which are needed throughout this thesis.

Lemma 2.1 (cf. [MR95]).

1. *For every* $x \in \mathbb{R}$, $e^x \geqslant x + 1$.
2. *For every integer* $n \geqslant 2$ *we have* $\frac{1}{4} \leqslant (1 - \frac{1}{n})^n \leqslant e^{-1}$ *and* $e \geqslant (1 + \frac{1}{n})^n$.
3. *For every integer* $n \geqslant 1$, $\ln n \leqslant \sum_{k=1}^{n} \frac{1}{k} \leqslant \ln n + 1$.
4. *For sufficiently large* n, $n! \geqslant (n/3)^n$.
5. *For every integers* $n \geqslant 1$, $1 \leqslant k \leqslant n$ *we have* $\binom{n}{k} \leqslant \sum_{i=0}^{k} \binom{n}{i} \leqslant (\frac{en}{k})^k$.
6. *For every sequence of positive numbers* $x_1, x_2, \ldots, x_n$, $\prod_{k=1}^{n}(1 - x_k) \geqslant 1 - \sum_{k=1}^{n} x_k$.

2.2 Probability Distributions

We briefly review the most essential tools required for the probabilistic analysis in this thesis. For a more comprehensive introduction to probability theory, we refer the reader to [GS01].

For a random variable X, we denote by $\mathbf{E}[X]$ and $\mathbf{Var}[X]$ the *expectation* and *variance* of X, respectively. $\mathbf{Pr}[\mathcal{E}]$ denotes the *probability* of some *event* $\mathcal{E}$ of a *probability space* Ω.

Lemma 2.2. *For any positive random variable* X, $\mathbf{E}[X] = \sum_{k=0}^{\infty} \mathbf{Pr}[X > k]$.

We will frequently use the binomial distribution $\mathsf{Bin}(n, p), n \in \mathbb{N}, 0 < p < 1$, the Bernoulli distribution $\mathsf{Ber}(p), 0 < p < 1$, the geometric distribution $\mathsf{Geo}(p), p > 0$ and the exponential distribution $\mathsf{Exp}(\lambda)$ with parameter $\lambda > 0$. If a random variable X has one of these distributions, say exponential distribution, we write $X \sim \mathsf{Exp}(\lambda)$. In this case we call X also an exponential variable with parameter λ. (Recall that in this case, $\mathbf{Pr}[X \geqslant x] = e^{-\lambda x}$ for any $x \in \mathbb{R}, x \geqslant 0$). We collect three properties of the exponential distribution $\mathsf{Exp}(\lambda)$.

Lemma 2.3 ([MU05]). *Let* $X_1, X_2, \ldots, X_n$ *be independent exponential variables with parameters* $\lambda_1, \lambda_2, \ldots, \lambda_n$, *respectively. Then,* $\min\{X_1, X_2, \ldots, X_n\}$ *is an exponential variable with parameter* $\sum_{i=1}^{n} \lambda_i$.

Observation 2.4. *Let* X *be an exponential variable with parameter* $\lambda > 0$ *and* $\gamma \in \mathbb{R}$, $\gamma > 0$. *Then,* $Y := \gamma \cdot X$ *is an exponential variable with parameter* λ/γ.

Lemma 2.5. *Let $X_1, X_2, \ldots$ be an infinite sequence of independent exponential variables with parameter $\lambda > 0$. Let $Y_1, Y_2, \ldots$ be an infinite sequence of independent Bernoulli variables with parameter $p > 0$ (and independent of $X_1, X_2, \ldots$). Then, $\min_{t\in\mathbb{N}}\{\sum_{k=1}^{t} X_k : Y_t = 1\}$ is an exponential variable with parameter λ/p.*

Proof. Note that $N_t := \max\{k \in \mathbb{N} : \sum_{i=1}^{k} X_i \leqslant t\}$ is a Poisson process [MU05] with rate λ. Consider now $N'_t := \sum_{i=1}^{N_t} Y_i$. By [MU05, Theorem 8.12], N'_t is a Poisson process with rate λ/p, and by [MU05, Theorem 8.10], $\min_{t\in\mathbb{N}}\{\sum_{i=1}^{t} X_i : Y_t = 1\}$ is an exponential variable with parameter λ/p. □

2.3 Deviation Bounds

2.3.1 Elementary Bounds

Lemma 2.6 (Union Bound). *For any finite or countably infinite sequence $\mathcal{E}_1, \mathcal{E}_2, \ldots$,*

$$\mathbf{Pr}\left[\bigcup_{i\geqslant 1}\mathcal{E}_i\right] \leqslant \sum_{i\geqslant 1}\mathbf{Pr}\left[\mathcal{E}_i\right].$$

Lemma 2.7 (Markov's Inequality). *Let X be a positive random variable. Then,*

$$\mathbf{Pr}\left[X \geqslant a\right] \leqslant \frac{\mathbf{E}\left[X\right]}{a} \quad \text{for any } a > 0.$$

Theorem 2.8 (Central Limit Theorem). *Let $X_1, X_2, \ldots$ be a sequence of independent identically distributed random variables with finite expectation μ and finite non-zero variance σ^2, and let $S_n := \sum_{i=1}^{n} X_i$. Then for any $a \in \mathbb{R}$,*

$$\lim_{n\to\infty}\mathbf{Pr}\left[\frac{S_n - n\mu}{\sqrt{n\sigma^2}} \leqslant a\right] = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{a} e^{-\frac{x^2}{2}}dx.$$

2.3.2 Chernoff Bounds

Definition 2.9. *Let X be a random variable. Then the* moment-generating function *of X is defined as $\mathbf{E}\left[e^{tX}\right]$ for any $t \in \mathbb{R}$ such that the expectation is finite.*

For the case $X \sim \mathsf{Exp}(\lambda)$ it is known that $\mathbf{E}\left[e^{tX}\right] = \frac{\lambda}{\lambda - t}$ for any $t < \lambda$ [Ros03, p. 66].

Theorem 2.10 ([MU05]). *If X and Y are independent random variables, then we have $\mathbf{E}\left[e^{t(X+Y)}\right] = \mathbf{E}\left[e^{tX}\right] \cdot \mathbf{E}\left[e^{tY}\right]$.*

To obtain a Chernoff bound for a random variable X, one applies Markov's inequality to e^{tX} and obtains for any $t > 0$ and $a > 0$,

$$\mathbf{Pr}\left[X \geqslant a\right] = \mathbf{Pr}\left[e^{tX} \geqslant e^{ta}\right] \leqslant \frac{\mathbf{E}\left[e^{tX}\right]}{e^{ta}}. \tag{2.1}$$

Similarly for any $t < 0$ and $a > 0$,

$$\mathbf{Pr}\,[\,X \leqslant a\,] \leqslant \frac{\mathbf{E}\left[\,e^{tX}\,\right]}{e^{ta}}.$$

For sums of independent Bernoulli trials (Bernoulli variables), we state the following Chernoff bound that will be used frequently throughout this thesis.

Theorem 2.11 (Chernoff Bound for Bernoulli Trials, [MU05]). *Let $X_1, X_2, \ldots, X_n$ be independent Bernoulli trials such that $\mathbf{Pr}\,[\,X_k = 1\,] = p_k$, $\mathbf{Pr}\,[\,X_k = 0\,] = 1 - p_k$. Let $X := \sum_{i=1}^{n} X_k$ and $\mu = \mathbf{E}\,[\,X\,] = \sum_{k=1}^{n} p_k$. Then,*

$$\mathbf{Pr}\,[\,X \leqslant (1-\delta)\mu\,] \leqslant e^{-\mu\delta^2/2} \quad \text{for any } 0 < \delta < 1,$$
$$\mathbf{Pr}\,[\,X \geqslant (1+\delta)\mu\,] \leqslant e^{-\mu\delta^2/3} \quad \text{for any } 0 < \delta.$$

As it is the case for many Chernoff bounds, the theorem above gives *exponentially* decreasing bounds on the tail distribution of X.

Theorem 2.12 (Hoeffding Bound, [McD98]). *Let $X_1, X_2, \ldots, X_n$ be independent random variables with $X_k \in [a_k, b_k]$ for each $1 \leqslant k \leqslant n$. Let $X := \sum_{k=1}^{n} X_k$ and $\mu = \mathbf{E}\,[\,X\,]$. Then for any $\lambda \geqslant 0$,*

$$\mathbf{Pr}\,[\,|X - \mu| \geqslant \lambda\,] \leqslant 2 \cdot e^{-2\lambda^2/\sum_{k=1}^{n}(b_i - a_i)^2}.$$

We also require Chernoff bounds for sums of *unbounded* random variables.

Theorem 2.13 (Chernoff Bound for Geometric Variables). *Let $Y_1, Y_2, \ldots, Y_n$ be independent geometric variables with the same parameter $0 < p < 1$. Let $Y := \sum_{k=1}^{n} Y_k$. Then for any $\alpha > 0$,*

$$\mathbf{Pr}\left[\,Y \geqslant 2\frac{n}{p} + 2\frac{\ln \alpha}{p}\,\right] \leqslant \frac{1}{\alpha}.$$

Proof. The random variable Y has the so-called negative binomial distribution [GS01, p. 61], i. e., one has to wait for n successes in an infinite sequence of independent Bernoulli variables with success probability p. Define $x := 2\frac{n}{p} + 2\frac{\ln \alpha}{p}$. Let $X := \sum_{i=1}^{x} X_i$ be a sum of x independent Bernoulli variables with success probability p. Then, $\mu := \mathbf{E}\,[\,X\,] = x/p$. Choosing $\delta := 1 - \frac{n}{xp}$ we observe that $0 < \delta < 1$, whence it follows from Theorem 2.11 that

$$\begin{aligned}\mathbf{Pr}\left[\,X \leqslant \left(1 - \left(1 - \frac{n}{xp}\right)\right) \cdot xp\,\right] &= \mathbf{Pr}\,[\,X \leqslant n\,] \leqslant e^{-(xp(-\frac{n}{xp}+1)^2)/2} \\ &= e^{-xp(\frac{n^2}{x^2p^2} - 2\frac{n}{xp} + 1)/2} = e^{-(\frac{n^2}{xp} - 2n + xp)/2} \leqslant e^{(2n - xp)/2} \\ &= e^{(2n - 2n - 2\ln\alpha)/2} = \alpha^{-1}.\end{aligned}$$

But $\mathbf{Pr}[\,Y \geqslant 2\frac{n}{p} + 2\frac{\ln\alpha}{p}\,] \leqslant \mathbf{Pr}\,[\,X \leqslant n\,]$, and the claim follows. □

Theorem 2.14 (Chernoff Bound for Exponential Variables). *Let $Y_1, Y_2, \ldots, Y_n$ be independent exponential variables with parameters $\lambda_1, \lambda_2, \ldots, \lambda_n > 0$. Let $Y := \sum_{k=1}^{n} Y_k$, $\mu := \mathbf{E}[\,Y\,] = \sum_{k=1}^{n} 1/\lambda_k$ and $\lambda_{\min} := \min_{k=1}^{n} \lambda_k$. Then for any $\gamma > 0$,*

$$\mathbf{Pr}[\,Y \geqslant \gamma\,] \leqslant \frac{2^{\lambda_{\min}\mu}}{e^{\frac{\lambda_{\min}}{2}\gamma}}.$$

Proof. Recall that the moment-generating function of $Y_k \sim \mathsf{Exp}(\lambda_k)$ is given by $\mathbf{E}[\,e^{tY_k}\,] = \frac{\lambda_k}{\lambda_k - t}$, where $t < \lambda_i$. Using the general version of the Chernoff-Bound (2.1) and Theorem 2.10 we obtain for any $\gamma > 0$

$$\begin{aligned}\mathbf{Pr}[\,Y \geqslant \gamma\,] &\leqslant \frac{\mathbf{E}\left[\,e^{\sum_{k=1}^{n} Y_k}\,\right]}{e^{t\gamma}} = \frac{\prod_{k=1}^{n} \mathbf{E}\left[\,e^{tY_k}\,\right]}{e^{t\gamma}} = \frac{\prod_{k=1}^{n} \frac{\lambda_k}{\lambda_k - t}}{e^{t\gamma}} \\ &= \frac{\prod_{k=1}^{n} (1 - \frac{t}{\lambda_k})^{-1}}{e^{t\gamma}} = \frac{\prod_{k=1}^{n} (1 - \frac{t}{\lambda_k})^{-\frac{\lambda_k}{t}\frac{t}{\lambda_k}}}{e^{t\gamma}},\end{aligned}$$

and substituting $t = \frac{\lambda_{\min}}{2}$ leads for any $\gamma > 0$ to

$$\mathbf{Pr}[\,Y \geqslant \gamma\,] \leqslant \frac{\prod_{k=1}^{n} 4^{\frac{\lambda_{\min}}{2\lambda_k}}}{e^{\frac{\lambda_{\min}}{2}\gamma}} = \frac{4^{\lambda_{\min}\frac{1}{2}\mu}}{e^{\frac{\lambda_{\min}}{2}\gamma}} = \frac{2^{\lambda_{\min}\mu}}{e^{\frac{\lambda_{\min}}{2}\gamma}}.$$

□

Theorem 2.15 (Method of Bounded Independent Differences [McD98]). *Let $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ be a vector of independent random variables with X_k taking values in a set A_k for each $1 \leqslant k \leqslant n$. Suppose that f is a real-valued function defined on $\prod_{k=1}^{n} A_k$ such that $|f(\mathbf{x}) - f(\mathbf{x}')| \leqslant c_k$, whenever $\mathbf{x}$ and $\mathbf{x}'$ differ only in the k-th coordinate. Let $\mu := \mathbf{E}[\,f(\mathbf{X})\,]$. Then for any $\lambda \geqslant 0$,*

$$\mathbf{Pr}[\,|f(\mathbf{X}) - \mu| \geqslant \lambda\,] \leqslant 2 \cdot e^{-2\lambda^2 / \sum_{k=1}^{n} c_k^2}.$$

2.3.3 Martingale Bounds

Definition 2.16. *A sequence of random variables $Z_0, Z_1, \ldots$ is a* martingale with respect to the sequence $X_0, X_1, \ldots$ *if for all $n \geqslant 0$ the following conditions hold:*

1. *Z_n is a function of $X_0, X_1, \ldots, X_n$,*
2. *$\mathbf{E}[\,|Z_n|\,] < \infty$,*
3. *$\mathbf{E}[\,Z_{n+1} \mid X_0, X_1, \ldots, X_n\,] = Z_n$.*

A sequence of random variables $Z_0, Z_1, \ldots$ is called martingale when it is a martingale with respect to itself.

Lemma 2.17 (Doob Martingale [GS01]). *Let $X_0, X_1, \ldots$ be a sequence of random variables. Let Y be a random variable with $\mathbf{E}[\,|Y|\,] < \infty$. Then*

$$Z_i := \mathbf{E}[\,Y \mid X_0, X_1, \ldots, X_i\,]$$

is a martingale with respect to $X_0, X_1, \ldots$.

Roughly speaking, $Z_0, Z_1, \ldots$ represents a sequence of refined estimates of Y, gradually using more information on Y gained by $X_0, X_1, \ldots$.

Theorem 2.18 (Azuma-Hoeffding Inequality [McD98]). *Let $X_0, X_1, \ldots$ be a martingale such that for each $k \geqslant 1$, $|X_k - X_{k-1}| \leqslant c_k$. Then, for any $n \geqslant 0, \lambda > 0$,*

$$\mathbf{Pr}[\,|X_n - X_0| \geqslant \lambda\,] \leqslant 2 \cdot e^{-\lambda^2/(2\sum_{k=1}^{n} c_k^2)}$$

We note that a certain extension of this inequality is given in Theorem 4.14. We close this section with some results on the famous coupon collector's problem [MR95, MU05].

Theorem 2.19 (Coupon Collector's Problem). *Consider the process where in each iteration a coupon is drawn independently and uniformly at random from $\{1, \ldots, n\}$. Let X be the number of iterations until all coupons have been drawn.*

1. *[MU05, p. 275] For any (not necessarily constant) $\varepsilon > 0$, $\mathbf{Pr}\left[\,X \geqslant n \cdot \ln\left(\frac{n}{\varepsilon}\right)\,\right] \leqslant \varepsilon$.*
2. *[MR95, p. 63] For any constant $c > 0$,*

$$\lim_{n\to\infty} \mathbf{Pr}[\,n \cdot (\ln n - c) \leqslant X \leqslant n \cdot (\ln n + c)\,] = e^{-e^{-c}} - e^{-e^{c}}.$$

3. *[MR95, p. 57] $n \ln n \leqslant \mathbf{E}[\,X\,] \leqslant n \ln n + n$.*

The double inequality of the second line is a typical example of a high concentration (or sharp threshold) result. Similar results will be derived at the end of Section 4.4 and in Section 7.3.

2.4 Couplings

Definition 2.20. *Let X_1 and X_2 be two real-valued random variables. We say that X_1 is* stochastically smaller *than X_2, $X_1 \preceq X_2$, if $\mathbf{Pr}[\,X_1 \geqslant r\,] \leqslant \mathbf{Pr}[\,X_2 \geqslant r\,]$ for every $r \in \mathbb{R}$.*

We shall write $X_1 \stackrel{D}{=} X_2$ if X_1 and X_2 have the same distribution.

Definition 2.21 ([GS01]). *Let X_1 and X_2 be two random variables on Ω_1 and Ω_2, respectively. A* coupling *of X_1 and X_2 is a random variable $\widehat{X} = (\widehat{X}_1, \widehat{X}_2)$ on Ω such that $\widehat{X}_1 \stackrel{D}{=} X_1$ and $\widehat{X}_2 \stackrel{D}{=} X_2$.*

Theorem 2.22 ([GS01]). *Let X_1 and X_2 be two random variables taking values in $\mathbb{R}$. Then the following two statements are equivalent.*

1. $X_1 \preceq X_2$,
2. *There is a coupling $\widehat{X} = (\widehat{X}_1, \widehat{X}_2)$ with $\widehat{X}_1 \stackrel{D}{=} X_1$, $\widehat{X}_2 \stackrel{D}{=} X_2$, and $\mathbf{Pr}[\,\widehat{X}_1 \leqslant \widehat{X}_2\,] = 1$.*

Using this Theorem, the following lemma is immediate (see also [Sch00]).

Lemma 2.23. *Let $\{X_k : k \in \mathbb{N}\}$ and $\{Y_k : k \in \mathbb{N}\}$ be any sets of independent random variables. Then $\sum_{k=1}^{n} X_k \preceq \sum_{k=1}^{n} Y_k$ and $\min_{k=1}^{n} X_k \preceq \min_{k=1}^{n} Y_k$.*

Lemma 2.24. *Let X_1 and X_2 be two random variables taking values in a finite set S. Let $S' \subseteq S$ be any subset such that for every $s' \in S'$, $\mathbf{Pr}\,[\,X_1 = s'\,] \geqslant \mathbf{Pr}\,[\,X_2 = s'\,]$. Then there exists a coupling $\widehat{X} = (\widehat{X}_1, \widehat{X}_2)$ of X_1 and X_2 such that*

$$\mathbf{Pr}\left[\,\widehat{X}_2 \in S' \;\Rightarrow \widehat{X}_1 = \widehat{X}_2\,\right] = 1.$$

Proof. Assume for notational convenience that $S = \{1, \ldots, n\}$ and $S' = \{1, \ldots, n'\}$, $n \geqslant n'$. Let U be uniformly distributed on the interval $[0, 1]$. The coupling $\widehat{X} = (\widehat{X}_1, \widehat{X}_2)$ is a function from U to $S \times S$ defined as follows. If $U \in [\sum_{k=1}^{i-1} \mathbf{Pr}\,[\,X_1 = k\,], \sum_{k=1}^{i} \mathbf{Pr}\,[\,X_1 = k\,])$, for some $1 \leqslant i \leqslant n$, then $\widehat{X}_1 = i$. Clearly, $\widehat{X}_1$ and X_1 have the same distribution.

Similarly, if $U \in [\sum_{k=1}^{i-1} \mathbf{Pr}\,[\,X_1 = k\,], \sum_{k=1}^{i-1} \mathbf{Pr}\,[\,X_1 = k\,] + \mathbf{Pr}\,[\,X_2 = i\,])$ for some $1 \leqslant i \leqslant n'$, we set $\widehat{X}_2 = i$, and otherwise $\widehat{X}_2$ is distributed according to $X_2 \mid \mathbf{Pr}\,[\,X_2 \notin S'\,]$. By construction, also $\widehat{X}_2$ and X_2 have the same distribution. Since $\mathbf{Pr}\,[\,X_2 = i\,] \leqslant \mathbf{Pr}\,[\,X_1 = i\,]$ for $1 \leqslant i \leqslant n'$, $\widehat{X}_2 \in S'$ implies $\widehat{X}_1 = \widehat{X}_2$ with probability 1 and the claim follows. □

Lemma 2.25. *Let $X \sim \mathsf{Geo}(p)$ and $Y \sim \mathsf{Exp}(p) + 1$ for some $0 < p < 1$. Then $X \preceq Y$.*

Proof. Clearly, for any $r \in \mathbb{R}, 0 \leqslant r \leqslant 1$, $\mathbf{Pr}\,[\,X \geqslant r\,] \leqslant \mathbf{Pr}\,[\,Y \geqslant r\,]$, since the latter probability is 1. Now, for any $r \geqslant 1$,

$$\mathbf{Pr}\,[\,X \geqslant r\,] = \mathbf{Pr}\,[\,X \geqslant \lceil r \rceil\,] = (1-p)^{\lceil r \rceil - 1} \leqslant e^{-p(\lceil r \rceil - 1)} \leqslant e^{-p(r-1)} = \mathbf{Pr}\,[\,Y \geqslant k\,].$$

□

Corollary 2.26 (Chernoff Bound for Non-Identical Geometric Variables). *Let $Y_1, Y_2, \ldots, Y_n$ be independent geometric variables with parameters $p_1, p_2, \ldots, p_n > 0$. Let $Y := \sum_{k=1}^{n} Y_k$, $\mu := \mathbf{E}\,[\,Y\,] = \sum_{k=1}^{n} 1/p_k$ and $p_{\min} := \min_{k=1}^{n} p_k$. Then for any $\gamma > 0$,*

$$\mathbf{Pr}\,[\,Y \geqslant \gamma + n\,] \leqslant \frac{2^{p_{\min}\mu}}{e^{\frac{p_{\min}}{2}\gamma}}.$$

Moreover, if $p_{\min} \cdot \mu \geqslant C \cdot \log_2 n$ holds, we have

$$\mathbf{Pr}\left[\,Y \geqslant \left(1 + \frac{1}{C}\right) \cdot 2 \ln 2 \cdot \mu + n\,\right] \leqslant n^{-1}.$$

Proof. Let $Z_1, Z_2, \ldots, Z_n$ be independent with $Z_i \sim \mathsf{Exp}(p_i)$. By Lemma 2.25, $Y_i \preceq Z_i + 1$ for each $1 \leqslant i \leqslant n$ and it follows from Lemma 2.23 that $\sum_{i=1}^{n} Y_i \preceq \sum_{i=1}^{n} Z_i + n$, that is, for any $k \in \mathbb{R}$, $\mathbf{Pr}[Y \geqslant k] \leqslant \mathbf{Pr}[Z + n \geqslant k]$, which is equivalent to $\mathbf{Pr}[Y \geqslant k + n] \leqslant \mathbf{Pr}[Z \geqslant k]$. The first claim follows by applying the Chernoff bound from Theorem 2.14 to Z. To see the second claim, we estimate

$$\mathbf{Pr}\left[Z \geqslant \left(1 + \frac{1}{C}\right) \cdot 2\ln 2 \cdot \mu\right] \leqslant \frac{2^{p_{\min}\mu}}{e^{\frac{p_{\min}}{2}\cdot(2\ln 2\cdot\mu)+\frac{p_{\min}}{2}\cdot(2\cdot\ln 2\cdot\frac{1}{C}\cdot\mu)}} \leqslant \frac{2^{p_{\min}\mu}}{2^{p_{\min}\cdot\mu}\,\cdot\, 2^{\frac{1}{C}\cdot p_{\min}\cdot\mu}} \leqslant n^{-1}.$$

□

2.5 Graph-Theoretical Notation and Preliminaries

2.5.1 Graph-Theoretical Notation

We use the following common graph-theoretical notation. Unless otherwise stated, we will consider undirected, unweighted, simple and connected graphs $G = (V, E)$. We denote the number of vertices (occasionally we also refer to vertices as nodes) by n and the number of edges by $|E|$. As most of our results are asymptotic bounds, we often consider families of graphs $G_d = (V_d, E_d)$ depending on $d \in \mathbb{N}$, where $n = n_d = |V_d|$ tends to infinity as $d \to \infty$. For example, the graph K_d is a *complete graph* with $n_d = n$ vertices and we may also simply write K_n. Q_d denotes the d-dimensional *hypercube* where $|V_d| = 2^d$. The graph $\mathsf{K}_{p,q}$ is the *complete bipartite graph* with two partitions of size p and q, respectively.

By $N(v)$ we denote the set of *neighbors* of v, i. e., the vertices which have a common edge with v. Similarly, for some $V' \subseteq V$, $N(V')$ denotes the set of all neighbors of vertices from V'. The *degree* of some vertex $v \in V$ denoted by $\deg(v)$ is the number of neighbors of v. The *maximum degree* and *minimum degree* of G are denoted by $\delta(G)$ and $\Delta(G)$, respectively. A graph is *regular* (or $\Delta(G)$-regular), if $\delta(G) = \Delta(G)$. For some $v \in V$ and subset $V' \subseteq V$ we define $\deg_{V'}(v) := |N(v) \cap V'|$.

A *path* of length ℓ is a sequence of vertices $(u_1, u_2, \ldots, u_\ell)$ with $\{u_i, u_{i+1}\} \in E(G)$ for $1 \leqslant i \leqslant \ell - 1$. A *cycle* of length ℓ is a path of length ℓ where additionally $u_1 = u_\ell$ holds. The *girth* of a graph G is the length of the smallest cycle in G. We define $\mathrm{dist}(u, v)$ to be the *distance* between two vertices u, v which is the minimum length of a shortest path between u and v. For some $\emptyset \neq V' \subseteq V$, $\mathrm{dist}(u, V') := \min_{v' \in V'}\{\mathrm{dist}(u, v')\}$ is the distance of some vertex u to the set V'.

A graph H is a *subgraph* of G, if there is a bijection $\varphi : V(H) \to V(G)$ such that $\{\varphi(u), \varphi(v)\} \in E(G)$ for every $\{u, v\} \in E(H)$.

Definition 2.27. *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs. The* Cartesian product *of G_1 and G_2, $G_1 \times G_2$, is defined by*

$$V(G) := V_1 \times V_2,$$
$$E(G) := \Big\{\{(u_1, u_2), (v_1, v_2)\} \mid (u_1 = v_1 \wedge \{u_2, v_2\} \in E_2) \vee (\{u_1, v_1\} \in E_1 \wedge u_2 = v_2))\Big\}.$$

Since $\times$ is an associative operation, the Cartesian product of more than two graphs can be reduced to Definition 2.27.

We recall some useful graph-theoretical lemmas from Feige et al. [FPRU90].

Lemma 2.28 ([FPRU90]). *Let $(u_1, u_2, \ldots, u_\ell)$ be a shortest path from u_1 to u_ℓ in G. Then, $\sum_{i=1}^{\ell} \deg(u_i) \leqslant 3n$ and hence $\operatorname{diam}(G) \leqslant (3n)/\delta$.*

Definition 2.29 ([FPRU90]). *Let $G = (V, E)$ be a graph. A set $X \subseteq V$ is an α-cover of G, if for all vertices $v \in V$ a vertex $x \in X$ exists such that $\operatorname{dist}(v, x) \leqslant \alpha$.*

Lemma 2.30 (cmp. [FPRU90]). *Let $X \subseteq V$ be a subset of a regular graph G with $\Delta \geqslant 2$. Then there is a subset $Y \subseteq X$ such that $|Y| \geqslant \frac{|X|}{2 \cdot \Delta^\alpha}$ and for all $y, y' \in Y$: $\operatorname{dist}(y, y') \geqslant \alpha+1$.*

Proof. Consider the following iterative procedure. We start with the given set X and an empty set Y. In each iteration we choose an arbitrary vertex $x \in X$. Then we put x into Y and remove x together with all vertices having distance less than α to x from X. It follows that in each iteration, Y increases by 1 while X decreases by $|\{v \in X \mid 0 \leqslant \operatorname{dist}(v, x) \leqslant \alpha\}| \leqslant \sum_{i=0}^{\alpha} \Delta^i \leqslant 2 \cdot \Delta^\alpha$, since $\Delta \geqslant 2$. Therefore $|X|/(2 \cdot \Delta^\alpha)$ iterations are necessary to make the set X empty and the claim follows. □

Definition 2.31. *Let G be a graph. A* graph automorphism *$\varphi : V \to V$ is a permutation of the vertices such that $\{u, v\} \in E \Leftrightarrow \{\varphi(u), \varphi(v)\} \in E$. A graph $G = (V, E)$ is called* vertex-transitive*, if for every pair $u, v \in V$, there is an automorphism φ with $\varphi(u) = v$.*

2.5.2 Expansion of Graphs

Definition 2.32. *Let $G = (V, E)$ be a graph with n vertices. Then, the* edge-expansion *for an integer $1 \leqslant m < n$, is defined by*

$$\Phi(m) := \min_{X \subseteq V, |X| = m} \left\{ \frac{|E(X, X^c)|}{\min\{\operatorname{vol}(X), \operatorname{vol}(X^c)\}} \right\},$$

where $\operatorname{vol}(X) := \sum_{v \in X} \deg(v)$ and $X^c = V \backslash X$. We define $\Phi := \min_{m=1,2,\ldots,n-1} \Phi(m)$ as the (global) edge expansion which is always between 0 and 1. The vertex-expansion *for some $1 \leqslant m < n$ is*

$$\partial(m) := \min_{X \subseteq V, |X| = m} \left\{ \frac{|N(X) \backslash X|}{|X|} \right\}.$$

Definition 2.33. *A family of graphs $G_d = (V_d, E_d)$ is called an* α-edge-expander*, if for every d and $1 \leqslant m \leqslant \lfloor \frac{|V_d|}{2} \rfloor$, $\Phi_d(m) \geqslant \alpha$. A family of graphs $G_d = (V_d, E_d)$ is called a* β-vertex-expander*, if for every d and $1 \leqslant m \leqslant \lfloor \frac{|V_d|}{2} \rfloor$, $\partial_d(m) \geqslant \beta$. A family of graphs $G_d = (V_d, E_d)$ is an* edge-expander *(*vertex-expander*), if it is an γ-edge-expander (γ-vertex-expander) for some constant $\gamma > 0$.*

By definition, every regular β-edge-expander is also a β-vertex-expander. Conversely, every β-vertex-expander is a (β/Δ)-edge-expander. The graph (family) $\mathsf{K}_{n/2} \times \mathsf{K}_2$ provides an example of a vertex-expander which is not an edge-expander.

3. RANDOMIZED RUMOR SPREADING: GENERAL RESULTS

3.1 Introduction

For a survey about rumor spreading and related topics we refer the reader to [HKP+04].

3.1.1 Motivation

Rumor spreading (also known as broadcast) is a central task in the field of distributed computing. As an example, consider the maintenance of replicated databases in name servers in a large corporate network [DGH+87]. Updates are injected at nodes and these updates must be disseminated to all other nodes in the network. In order to let all copies of the database converge to the same content, efficient rumor spreading algorithms have to be developed. In the rumor spreading problem, one node of a network initially knows of a rumor which has to be spread to all other nodes of the graph. A common assumption is that every node may send the rumor in each step to at most one other node.

A simple randomized rumor spreading algorithm performing this task is the so-called push algorithm, also known as randomized broadcast [FPRU90]. At each step, every informed node, i. e., a node knowing the rumor, chooses one of its neighbors uniformly at random and sends the rumor to it. The question is how many steps are required until every node becomes informed. Following [FPRU90], the major advantages of this randomized algorithm are as follows.

- Simplicity: The algorithm is simple and local, a node does not need to remember to whom the rumor has been sent.
- Scalability: The algorithm is independent of the size and the structure of the network.
- Robustness: Using this *randomized* algorithm, we expect to achieve fault-tolerance against node or edge-failures and against dynamic changes in the network.

In contrast to that, *deterministic* protocols are often rather complex. Moreover, they usually need substantially more time or can only tolerate a small number of faults [KSSV00].

From a theoretical point of view, any upper bound on the runtime of the (randomized) push algorithm implies the existence of a deterministic broadcast protocol with the same runtime. However, such an upper bound on the push algorithm additionally demonstrates that, in a certain sense, a large fraction of all possible protocols achieves this runtime.

3.1.2 Related Work

The study of randomized rumor spreading was initiated by Frieze and Grimmett [FG85] who proved that it takes about $\log_2 n + \ln n \pm o(\log n)$ steps to spread a rumor in complete graphs formed by n vertices with high probability. This result was further improved by Pittel [Pit87] who showed that the runtime is $\log_2 n + \ln n \pm o(1)$. Feige et al. [FPRU90] were the first considering the push algorithm on other topologies. They proved two general upper bounds of $\mathcal{O}(n \log n)$ and $\mathcal{O}(\Delta \cdot (\mathrm{diam}(G) + \log n))$. For random graphs and hypercubes, an asymptotically tight upper bound of $\mathcal{O}(\log n)$ was derived. The result for hypercubes was extended to star graphs in [ES05].

In parallel, a lot of effort was made to find the fastest deterministic broadcast algorithm on certain topologies. While for the cube-connected-cycles and the shuffle-exchange-network the minimal broadcast time has been determined exactly, for the butterfly and De-Bruijn-network there is still a multiplicative gap between the best known lower and upper bound (see [HKP$^+$04, p. 78] for an overview). It should be noted that the problem of approximating the fastest deterministic broadcast protocol up to a factor of $3 - \varepsilon$ is already NP-hard in general [EK06]. The best approximation algorithms have a polylogarithmic ratio [EK06].

Related to broadcast is the so-called *gossiping*-problem: every node has its *own* rumor which should be disseminated to all other nodes in the network. It is clear that any broadcast protocol with runtime t can be used for the construction of a gossiping protocol with runtime $2 \cdot t$ [HKP$^+$04]. Since we are mainly interested in *asymptotic* bounds, we focus on the broadcast problem here (for more details about gossiping cf. [HKP$^+$04]).

Besides the runtime also the number of transmissions during the execution of the rumor spreading algorithm is an important aspect. It was observed that in complete graphs, the push algorithm needs at least $\Omega(n \log n)$ transmissions to inform all vertices with high probability. Consider instead the so-called pull algorithm where uninformed vertices call a neighbor uniformly at random, and if this called neighbor is informed the vertex itself becomes informed. Here, if a constant fraction of the vertices are informed, then within $\mathcal{O}(\log \log n)$ additional rounds all vertices of a complete graph become informed. This implies that in such graphs at most $\mathcal{O}(n \log \log n)$ transmissions are needed provided that the distribution of the rumor is stopped at the right time. Based on this idea, Karp et al. [KSSV00] combined the push and pull algorithm and devised a termination mechanism in order to bound the number of total transmissions by $\mathcal{O}(n \log \log n)$ in complete graphs.

Recently their results were extended to random graphs (A random graph is constructed by placing an edge between each pair of vertices independently with probability $0 < p < 1$). Elsässer [Els06] proved that the algorithm of Karp et al. generates $\Omega(n \cdot (\log \log n + \log n / \log(pn)))$ transmissions on random graphs. In a very recent work [ES08a], we considered a slightly different algorithm where each vertex may remember (and avoid) the vertices chosen in the most recent three steps. Surprisingly, this minor change in the ability of the vertices leads to a significant decrease to $\mathcal{O}(n \log \log n)$ transmissions.

A continuous-type model for the spread of a rumor is the so-called *Richardson's Growth Model* also known as *first-passage-percolation*. In this model every edge is assigned a weight

according to an exponential distribution with mean 1. The weights can be viewed as different times required for the propagation of the rumor. The parameter of interest is the diameter of this weighted graph. Networks on which this model was considered are again complete graphs [FG85]. Additionally, a considerable amount of research has been concerned with the diameter for hypercubes (e.g., [FP93, BK97]).

Another rumor spreading problem that has received a lot of attention, especially in the context of wireless networks, is the so-called *radio-broadcast-problem*. The crucial difference to our model is that transmissions made by a node can be, in principal, received by *all* neighbors. However, if two transmissions are sent to the same node simultaneously, then a collision occurs and the node is not able to receive anything. In fact, coping with these collisions is one of the major challenges in the design of efficient protocols for radio-broadcast.

Rumor spreading is also closely related to epidemic diseases such as the spread of viruses in networks (cf. [Het00] for a recent survey on epidemics). When question arises as to the speed with which the disease infects the whole network, the problem reduces to rumor spreading. However, in most of these epidemic models, the spreaders recover after a while and are only able to spread the disease within a certain time frame. Hence the most important question is how many infections arise and whether an epidemic outbreak occurs. Another difference to rumor spreading is that in many of these models, the underlying networks are complete graphs [KM27, FPRU90, Het00]. However, there are certain results on epidemics in internet-like random graphs [New02, BBCS05] relating the rate of infection to the probability of an epidemic outbreak.

Another important mathematical model associated to the spread of epidemic diseases is the so-called *percolation*. Percolation theory deals with connectivity properties of certain random structures. Consider for example *bond percolation* where each edge of some given network fails independently with some probability p. For an infinite two-dimensional torus, a celebrated result by Kesten [Kes80] says that the critical probability for the origin being in an infinitely large connected component is $1/2$. More precisely, for $p > \frac{1}{2}$ the origin is in an infinitely large connected component with some non-zero probability, while if $p < \frac{1}{2}$ the origin is in a finitely large component with probability 1. Some results on *site percolation* [BBC$^+$06, ABS04] on finite networks also deal with the expansion, diameter and size of connected components. Motivated by the fact that in some networks faults hardly occur independently, Kranakis et al. [KPP07] investigated recently a variant where the faults are dependent to some degree.

3.1.3 Our Results

We first establish some preliminary results relating the runtime of the classical push algorithm to the runtime of certain modifications of it. In particular, we prove that the parallel (classical) push algorithm has asymptotically the same runtime as a natural sequential version. We note that these results are useful for at least two different reasons. First, they demonstrate that all our bounds on the push algorithm are fairly robust under small alterations of the broadcast scheme. Secondly, we may derive results for the paral-

lel push algorithm by considering other broadcast algorithms such as the sequential push algorithm, which seems to be more appropriate for a analysis like the one in Section 4.4.

Concerning the parallel push algorithm, we present the first tight upper bound of $(1+o(1))\cdot n\ln n$ holding for every graph. Our bound significantly improves on the previous bound of $12\cdot n\log n$ by Feige et al. [FPRU90] and is easily shown to be tight up to a factor of $(1+o(1))$ by considering the graph $\mathsf{K}_{1,n-1}$. Hence our result provides an approximative characterization of the worst-case graph. Finally, we present some simple but useful upper bounds on the push algorithm which depend on some edge-expansion based measure.

3.2 Notations, Definitions and Preliminaries

As already mentioned before, we mainly consider the following *push algorithm* [DGH$^+$87] (also known as randomized rumor spreading [Pit87] or randomized broadcast [FPRU90]): place at time $t=0$ a piece of information (or rumor) r on one vertex s of a graph G. In every succeeding time-step $t=1,2,\ldots$, each *informed* vertex forwards a copy of r to a communication partner over an incident edge selected independently and u. a. r. (uniformly at random). Throughout this thesis, we denote by I_t the set of informed vertices at time-step t. With this notation at hand, an additional description of the push algorithm can be found in Figure 3.1. We denote this (parallel) push algorithm by PAR (We remark that a *sequential* push algorithm will be defined later). Note, that the so-called *pull algorithm* $\mathsf{PULL}-\mathsf{PAR}$ is defined similarly with the only difference that the roles of u and v are interchanged. Here, in each time-step every uninformed vertex calls some neighbor u. a. r. and becomes informed if this neighbor has been already informed.

The question is how many time-steps are required by PAR to disseminate the rumor r to the vertices of G. To formalize this question, fix some subset $V'\subseteq V$ and define $\mathsf{PAR}(s,V') := \min\{t\in\mathbb{N} : V'\subseteq I_t \mid I_0=\{s\}\}$, i. e., the first step after every vertex in V' becomes informed when s is informed at step 0. For any $0<p<1$, let $\mathsf{PAR}_p(s,V') := \min\{t\in\mathbb{N} : \mathbf{Pr}\,[\,I_t\subseteq V \mid I_0=\{s\}\,]\geqslant 1-p\}$. Most of our results will refer to the following definition.

Definition 3.1. *Given some $0<p<1$, the runtime of the (parallel) push algorithm is defined by*

$$\mathsf{PAR}_p(G) := \max_{s\in V(G)} \min\{t\in\mathbb{N} \mid \mathbf{Pr}\,[\,I_t=V \mid I_0=\{s\}\,]\geqslant 1-p\},$$

i. e., the minimum t such that all vertices are informed at time-step t with probability at least $1-p$.

Occasionally we also consider the expected runtime of the push algorithm $\mathbf{E}\,[\,\mathsf{PAR}(G)\,]$ which is defined as $\max_{s\in V(G)}\mathbf{E}\,[\,\mathsf{PAR}(s,V)\,]$. Note that the push algorithm requires clearly at least $\max\{\log_2 n,\operatorname{diam}(G)\}$ time-steps to inform all $n=|V(G)|$ vertices on any graph [FPRU90]. Therefore we may consider a runtime of $\mathcal{O}(\log n+\operatorname{diam}(G))$ on some graph as *asymptotically optimal*. In our framework, the runtime of the *fastest* deterministic

broadcast protocol can be defined formally by

$$\mathsf{DET}(G) := \max_{s \in V(G)} \min\{t \in \mathbb{N} \mid \mathbf{Pr}\,[\, I_t = V \mid I_0 = \{s\}\,] > 0\}.$$

Clearly, $\mathsf{DET}(G) \leqslant \mathsf{PAR}_p(G)$ for any $p > 0$.

Recall that for any $u \in V(G)$, $\mathsf{PAR}(s,u)$ is the first time step at which u is informed. The following observation will be useful later.

Observation 3.2. *In any execution of* PAR*, there is for each vertex $u \in V(G)$ at least one path $\mathcal{P}_{\min}(s,u)$ from s to u in G,*

$$s = u_0 \overset{D_1}{\to} u_1 \overset{D_2}{\to} \dots \overset{D_{l-1}}{\to} u_l = u$$

with the property that for every $0 \leqslant i \leqslant l-1$, u_i sends the rumor to u_{i+1} at time $\mathsf{PAR}(s,u_i)+D_i = \mathsf{PAR}(s,u_{i+1})$ and $D(\mathcal{P}_{\min}(s,u)) := \sum_{i=1}^{l-1} D_k$ is minimized.

Though we are mainly interested in the parallel push algorithm, it turns out to be useful to consider the following *sequential push algorithm* SEQ whose definition is given in Figure 3.1. In order to make both algorithms comparable, the time axis of SEQ consists now of subtime-steps $\mathbb{T} := \{i + \frac{j}{n} \mid i \in \mathbb{N}, j \in \{0,\dots,n-1\}\}\backslash\{0\} \subseteq \mathbb{Q}$, i. e., one time-step consists of n consecutive subtime-steps. The definitions for the runtime of SEQ correspond to the ones for PAR, we only have to replace $\mathbb{N}$ by $\mathbb{T}$.

Parallel Push Algorithm (PAR)

```
1: I_0 ← {s}
2: t ← 0
3: while I_t ≠ V do
4:     t ← t + 1
5:     I_t ← I_{t−1}
6:     for all vertices u ∈ V do
7:         if u ∈ I_{t−1} then
8:             choose v ∈ N(u) u. a. r.
9:             I_t ← I_t ∪ {v}
10:        end if
11:    end for
12: end while
```

Sequential Push Algorithm (SEQ)

```
1: I_0 ← {s}
2: t ← 0
3: while I_t ≠ V do
4:     t ← t + 1/n
5:     choose u ∈ V u. a. r.
6:     if u ∈ I_{t−1/n} then
7:         choose v ∈ N(u) u. a. r.
8:         I_t ← I_{t−1/n} ∪ {v}
9:     else I_t ← I_{t−1/n}
10:    end if
11: end while
```

Fig. 3.1: *Definition of the parallel (original) and sequential push algorithm.*

Sometimes, we are also interested in the time required to inform not all, but a large number of vertices. Given some integer $1 \leqslant \tilde{n} \leqslant n$, let $\mathsf{SEQ}(G,\tilde{n})$ be the time required to inform at least $\tilde{n}$ vertices. Correspondingly, we define $\mathsf{SEQ}_p(G,\tilde{n}) := \max_{s\in V} \min\{t \in \mathbb{N} \mid \mathbf{Pr}\,[\,|I_t| \geqslant \tilde{n} \mid I_0 = \{s\}\,] \geqslant 1-p\}$.

At some point we also require the sequential broadcast algorithm $\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}$ (cf. [KSSV00] for a similar parallel model) which combines the sequential push- and pull algorithms as follows. In each subtime-step $t \in \mathbb{T}$ a vertex $u \in V(G)$ is chosen uniformly at random which then chooses a neighbor $v \in N(u)$ uniformly at random. If any of the two vertices is already informed, then both vertices become informed.

We summarize some basic relations between the expected runtime and the runtime guaranteed with some probability p. The proofs are easy and can also be found in [Sau05, FPRU90].

Lemma 3.3. *Fix $\bigstar \in \{\mathsf{PAR}, \mathsf{SEQ}, \mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}\}$. Let $G=(V,E)$ be graph and let $s \in V$ and $V' \subseteq V$. Then,*

$$\begin{aligned} \mathbf{E}\left[\bigstar(s,V')\right] &\leqslant \frac{1}{1-p} \cdot \bigstar_p(s,V'), \\ \bigstar_p(s,V') &\leqslant 2 \cdot \log\left(\frac{1}{p}\right) \cdot \mathbf{E}\left[\bigstar(s,V')\right]. \end{aligned}$$

3.3 Representations of the Push Algorithm

For the use of couplings, the following representations of executions (instances) of the push algorithm are helpful. As before, we fix some vertex s which initially knows a rumor.

Let $(N_{t,v})_{t\in\mathbb{N},v\in V}$ be any fixed (infinite) matrix with $N_{t,v} \in N(v)$ for every $t \in \mathbb{N}, v \in V$. Such a matrix describes an (infinite) execution of the parallel push algorithm where v sends the rumor to vertex $N_{t,v}$ at step $t+\mathsf{PAR}(s,v)$, unless PAR has terminated. Then the uniform probability space of all possible $(N_{t,v})_{t\in\mathbb{N},v\in V}$, denoted by Ω_1, is the probability space of all executions of PAR.

To describe the sequential push algorithm SEQ, we keep Ω_1 as before and let Ω_2 be the probability space of all possible $(S_t)_{t\in\mathbb{N}}$ with $S_t \in V$ for every $t \in \mathbb{N}$. The (infinite) sequence $(S_t)_{t\in\mathbb{N}}$ specifies the choices among V in line 5 of the sequential push algorithm in Figure 3.1 (unless SEQ has terminated). Hence, SEQ chooses in step $t \in \mathbb{T}$ the vertex S_t and if S_t is informed, it sends the rumor to N_{t',S_t}, where $t' := |\{\widehat{t} \in \mathbb{T} \mid \mathsf{SEQ}(s,u) < \widehat{t} \leqslant t, S_{\widehat{t}} = v\}| + 1$.

3.4 Equivalence between Sequential and Parallel Push Algorithm

We prove two results demonstrating that the sequential and the parallel push algorithm have asymptotically the same performance on all graphs.

For the proof of one inequality, the following result is useful. It refers to a modification of the parallel push algorithm, denoted by $\overline{\mathsf{PAR}}$. The difference is that in each step $t \in \mathbb{N}$ every vertex v *fails* with some probability $0 < f < 1$, independent of all previous steps. In case an informed vertex fails at step t, it does not forward the rumor to any neighbor at that step. If it does not fail, then v sends the rumor to some random neighbor at step t as in the unmodified push algorithm. The following theorem relates the runtime of $\overline{\mathsf{PAR}}$ to PAR.

Theorem 3.4 ([ES08b]). *For any graph $G = (V, E)$ and $0 < f < 1$,*

$$\overline{\mathsf{PAR}}_{n^{-1}}(G) = \mathcal{O}\left(\frac{1}{1-f} \cdot \mathsf{PAR}_{n^{-1}}(G)\right).$$

We prove the following.

Theorem 3.5. *For any graph $G = (V, E)$,*

$$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\mathsf{SEQ}_{n^{-1}}(G)).$$

Moreover, for any vertex $u \in V, u \neq s$ we have

$$\mathbf{E}\left[\,\mathsf{PAR}(s,u)\,\right] = \Theta(\mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right]).$$

Proof. We first prove that $\mathsf{PAR}_{n^{-1}}(G) = \Theta(\mathsf{SEQ}_{n^{-1}}(G))$ and begin with the less difficult part which shows that the sequential algorithm is asymptotically at least as fast as the parallel model with high probability. Consider the following modification $\overline{\mathsf{SEQ}}$ of the sequential algorithm SEQ. An *informed* vertex u may only send the rumor to some neighbor beginning at subtime-step $\lceil \mathsf{SEQ}(s,u) + \frac{1}{n} \rceil \in \mathbb{N}$; recall that $\mathsf{SEQ}(s,u) \in \mathbb{T}$ is the first subtime-step at which u is informed. Furthermore, every informed vertex is allowed to send the rumor to at most one randomly chosen neighbor within every time-interval $[t, t+1)$, where $t \in \mathbb{N}$. Clearly, these modifications slow down the propagation of the rumor and thus $\mathsf{SEQ}_{n^{-1}}(G) \leqslant \overline{\mathsf{SEQ}}_{n^{-1}}(G)$.

Note that some vertex u which is informed at step $t \in \mathbb{N}$ sends the rumor to some neighbor within the time-interval $[t, t+1)$ with probability

$$1 - \left(1 - \frac{1}{n}\right)^n \geqslant 1 - e^{-1},$$

independent of all other time-intervals $[t', t'+1)$, where $t' \neq t$. In other words, every informed vertex u fails in some time-interval $[t, t+1), t \geqslant \lceil \mathsf{SEQ}(s,u) + \frac{1}{n} \rceil$ with probability at most e^{-1} and makes no transmissions, whereas with probability at least $1 - e^{-1}$, u sends the rumor to *exactly* one neighbor selected uniformly at random. Hence, by Theorem 3.4, we obtain for $f = e^{-1}$ that

$$\overline{\mathsf{SEQ}}_{n^{-1}}(G) \leqslant \overline{\mathsf{PAR}}_{n^{-1}}(G) + 1 = \mathcal{O}(\mathsf{PAR}_{n^{-1}}(G))$$

and thus $\mathsf{SEQ}_{n^{-1}}(G) \leqslant \overline{\mathsf{SEQ}}_{n^{-1}}(G) = \mathcal{O}(\mathsf{PAR}_{n^{-1}}(G))$, which finishes the first part of the proof for the first claim.

For the second part of the first claim we use the representations introduced in Section 3.3. We will consider a coupling between PAR and SEQ, where $(N_{t,v})_{t \in \mathbb{N}, v \in V}$ is fixed, but $(S_t)_{t \in \mathbb{N}}$ is chosen uniformly at random.

For the next, consider the following correspondence between bitstrings and certain paths in an instance of SEQ. Let $\mathcal{S} = (s_1 s_2 \cdots s_l) \in \{0,1\}^l$ denote some bitstring of

length l, where $s_l = 1$. Let $|\mathcal{S}|_1$ be the number of ones in $\mathcal{S}$. Denote by s the initially informed vertex. Informally, the corresponding path starting at vertex s is constructed iteratively by scanning the bitstring $\mathcal{S}$ from its leftmost to its rightmost position. We wait for some transmission of the current endpoint of the path and skip further time-steps at which other vertices are selected by SEQ. If our current endpoint has been selected (in line 5 of Figure 3.1) and the current bit is zero we go to the next bit and wait for another transmission of the current endpoint. Otherwise, if it is one, we go to the next bit and we move to the neighbor to which the current endpoint has sent the rumor and extend the path by this neighbor. Before giving the formal definition, we remark that an example of this construction is given in Figure 3.2.

The startpoint of this path is the initially informed vertex s and we set $\mathcal{P} := (s)$. Let $j_1 = \min\{i \geqslant 1 \mid s_i = 1\}$ be the position of the leftmost one in $\mathcal{S}$. Then the path is extended by $\mathcal{P} := (s = u_0, u_1)$, where u_1 is the j'_1-th vertex to which the rumor is sent by s, i. e., $u_1 = N_{j'_1}(u_1)$, where $j'_1 = j_1$. Let $t_1 \in \mathbb{T}$ be the subtime-step at which u_1 sends the rumor for the j'_1-th time.

More generally, assume that we have constructed a path $\mathcal{P} = (s = u_0, u_1, \ldots, u_k), 1 \leqslant k < |S|_1$ in that way. Let j_{k+1} be the position of the $(k+1)$-th leftmost one in $\mathcal{S}$. The path is extended by $\mathcal{P} := (s = u_0, u_1, \ldots, u_k, u_{k+1})$, where $u_{k+1} = N_{j'_{k+1}}(u_k)$ for

$$j'_{k+1} := j_{k+1} - j_k + |\{\widehat{t} \in \mathbb{T}, \mathsf{SEQ}(s, u_k) < \widehat{t} \leqslant t_k \mid S_{\widehat{t}} = u_k\}| \in \mathbb{N}\backslash\{0\}.$$

(The extra term added to $j_{k+1} - j_k$ is to take the subtime-steps into account at which u_k had already been informed before $\mathcal{P}$ reached u_k at step t_k.) Secondly, we define t_{k+1} as the subtime-step at which u_k sends for the j'_{k+1}-th time. This iterative procedure constructs a (finite) path $\mathcal{P}$ starting from $s = u_0$ and ending at $u_{|\mathcal{S}|_1}$ for an arbitrary given (finite) bitstring $\mathcal{S}$ with $s_l = 1$. Hence, the number of vertices on $\mathcal{P}$ equals $|\mathcal{S}|_1 + 1$.

For some path $\mathcal{P} = \mathcal{P}(\mathcal{S})$ as constructed above, we define the *weight* of the path $w(\mathcal{P})$ as the length of the bitstring $\mathcal{S}$. One can view the weight of such a path $\mathcal{P}$ as the number of (local) transmissions we have to wait until the path has been completely traversed. Obviously, the number of such paths having a weight of x is at most 2^x since there are 2^x different bitstrings of length x. For some path $\mathcal{P} := (s = u_0, \ldots, u_{|\mathcal{S}|_1})$ corresponding to $\mathcal{S}$, denote by $t(\mathcal{P}) := t_{|\mathcal{S}|_1} \in \mathbb{T}$ the *time* of the path $\mathcal{P}$, i. e., the time the path requires to reach the vertex $u_{|\mathcal{S}|_1}$ in SEQ. We note that $t_{|\mathcal{S}|_1}$ can be larger than $\mathsf{SEQ}(s, u_{|\mathcal{S}|_1})$. The next lemma relates the weight to the time of some path $\mathcal{P}$ and basically says that every path with a low time must also have a low weight.

Lemma 3.6. *The probability that every path $\mathcal{P} = \mathcal{P}(\mathcal{S})$ with $w(\mathcal{P}) = |\mathcal{S}| \geqslant x$ satisfies $w(\mathcal{P}) \leqslant 200 \cdot t(\mathcal{P})$ is at least $1 - 2^{-x+1}$.*

Proof. Let $\mathcal{S} = (s_1 s_2 \cdots s_l)$ be an arbitrary bitstring with length $l \geqslant x$ and $s_l = 1$. First, we have to wait for the j_1-th transmission of s. After that, the path $\mathcal{P}$ reaches another vertex $u_1 \neq s$ and we have to wait for the next $(j_2 - j_1)$-th transmission of u_1, and so on. Note that the time period (in subtime-steps) to reach u_1 from s is a sum of j_1 independent geometric variables with mean n. Summing up, the time $t(\mathcal{P})$ for traversing the whole path

t	$N_{t,s}$	$N_{t,b}$	$N_{t,c}$	$N_{t,d}$
1	b	s	s	c
2	c	s	b	s
3	b	c	s	s
4	b	d	b	b
5	c	c	d	s
6	d	s	d	c
7	d	s	s	b
⋮	⋮	⋮	⋮	⋮

t	S_t	S_t informs:	$\mathcal{S}$	k	j_k	j'_k	t_k	$\mathcal{P}(\mathcal{S})$
1/4	b	(not informed)						(s)
2/4	s	b	0					
3/4	s	c	0					
1	b	s						
5/4	c	s						
6/4	s	b	1	1	3	3	6/4	(s,b)
7/4	c	b						
2	s	b						
9/4	b	s	1	2	4	2	9/4	(s,b,s)
10/4	d	(not informed)						
11/4	s	c	0					
3	c	s						
13/4	b	c						
14/4	d	c						
15/4	s	d	1	3	6	6	15/4	(s,b,s,d)

Fig. 3.2: *An example of the construction of a path $\mathcal{P}(\mathcal{S})$ for the bitstring $\mathcal{S} = (001101)$.*

$\mathcal{P}$ is a sum of $j_1 + \sum_{k=2}^{|\mathcal{S}|_1}(j_{k+1} - j_k) = j_{|\mathcal{S}|_1} = |\mathcal{S}| = w(\mathcal{P})$ independent geometric variables with mean n. The probability that some vertex is *not* selected within a time-interval of length $\frac{1}{100}$ (consisting of $\frac{1}{100}n$ independent subtime-steps) is at least

$$\left(1 - \frac{1}{n}\right)^{\frac{1}{100}n} \geqslant 4^{-\frac{1}{100}}.$$

Therefore, the probability that more than half of the $w(\mathcal{P})$ waiting times are below $\frac{1}{100}$ (in time-steps) is bounded by

$$\binom{w(\mathcal{P})}{\frac{w(\mathcal{P})}{2}}\left(1 - 4^{-\frac{1}{100}}\right)^{\frac{w(\mathcal{P})}{2}} \leqslant 2^{w(\mathcal{P})}8^{-w(\mathcal{P})} = 4^{-w(\mathcal{P})},$$

and in this case $t(\mathcal{P}) \geqslant \frac{1}{2} \cdot w(\mathcal{P}) \cdot \frac{1}{100} = \frac{1}{200}w(\mathcal{P})$. Now fix some $k \geqslant x$. Then the latter holds for every path with weight $w(\mathcal{P}) = k$ with probability at least $1 - \frac{2^k}{4^k} \geqslant 1 - 2^{-k}$, having used the union bound over all possible paths of weight k. By another application of the union bound, the probability that $w(\mathcal{P}) \leqslant 200 \cdot t(\mathcal{P})$ holds for every path $\mathcal{P}$ with weight *at least* x is not less than $1 - \sum_{k=x}^{\infty} 2^{-k} \geqslant 1 - 2^{-x+1}$ and the lemma follows. □

Observation 3.7. *Let s be the initially informed vertex in the* SEQ *model. Then for any vertex $u \neq s$, there exists a* minimal path $\mathcal{P}_{\min}(s,u)$

$$s = u_0 \xrightarrow{D_1} u_1 \xrightarrow{D_2} \ldots \xrightarrow{D_l} u_l = u,$$

with the property that for every $0 \leqslant i < l-1$, u_i *sends the rumor to* u_{i+1} *at time-step* $\mathsf{SEQ}(s,u_i)+D_i$, *and at this time-step,* u_{i+1} *becomes informed, i. e.,* $\mathsf{SEQ}(s,u_i)+D_i = \mathsf{SEQ}(s,u_{i+1})$. *The corresponding bitstring of* $\mathcal{P}_{\min}(s,u)$ *is* $\mathcal{S} = 0^{D_1-1}10^{D_2-1}1\dots0^{D_l-1}1$ *with length* $\sum_{i=1}^{l} D_i$.

We are now able to finish the proof of the first result of the theorem. Choose some arbitrary vertex $u \in V, u \neq s$ and consider the minimal path $\mathcal{P}_{\min}(s,u)$ from s to u in the sequential model. By definition, $t(\mathcal{P}_{\min}(s,u))$ is the minimum time-step (in the sequential model) at which u becomes informed. Notice that $w(\mathcal{P}_{\min}(s,u)) = \sum_{i=1}^{l} D_i$ is an upper bound for the number of time-steps required in the corresponding instance of PAR to inform u. Seeking a contradiction assume that

$$w(\mathcal{P}_{\min}(s,u)) \geqslant \max\{16\log n, 201 \cdot t(\mathcal{P}_{\min}(s,u))\}$$

with probability at least n^{-2}.

Given that $w(\mathcal{P}_{\min}(s,u)) \geqslant 16\log n$, we may apply Lemma 3.6 to conclude that with probability at least $1-n^{-3}$, $t(\mathcal{P}_{\min}(s,u)) \geqslant \frac{1}{200} \cdot w(\mathcal{P}_{\min}(s,u))$ which contradicts the assumption that $w(\mathcal{P}_{\min}(s,u)) \geqslant 201 \cdot t(\mathcal{P}_{\min}(s,u))$ with probability at least n^{-2}. Thus u will be also informed by the *same* path $\mathcal{P}_{\min}(s,u)$ in the corresponding instance of PAR after $\max\{16\log n, 201 \cdot t(\mathcal{P}_{\min}(s,u))\}$ time-steps with probability at least $1-n^{-2}$. Finally, by the union bound over all vertices $u \neq s$, every vertex u becomes informed after $\max\{16\log n, 201 \cdot t(\mathcal{P}_{\min}(s,u))\}$ steps with probability at least $1-n^{-1}$. As

$$\max_{u\in V, u\neq s} t(\mathcal{P}_{\min}(s,u)) \leqslant 3 \cdot \mathsf{SEQ}_{n^{-1}}(G)$$

with probability $1-n^{-3}$,

$$\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\mathsf{SEQ}_{n^{-1}}(G) + \log n).$$

Since it is easy to verify that $\mathbf{E}[\,\mathsf{SEQ}(G)\,] = \Omega(\log n)$ (cf. [Sau07, Proposition 1]), we conclude that $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\mathsf{SEQ}_{n^{-1}}(G)$. To summarize, we have shown that

$$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\mathsf{SEQ}_{n^{-1}}(G)),$$

which is exactly the first claim of Theorem 3.5.

We continue to prove the second claim and first show $\mathbf{E}[\,\mathsf{SEQ}(s,u)\,] = \mathcal{O}(\mathbf{E}[\,\mathsf{PAR}(s,u)\,])$, which is again the less difficult inequality. Recall that all executions of PAR can be described by the probability space Ω_1 consisting of all $(N_{t,v})_{t\in\mathbb{N},v\in V}$. Similarly, all executions of SEQ can be described by $\Omega_1 \times \Omega_2$, where Ω_2 consists of all $(S_t)_{t\in\mathbb{N}}$. We consider a coupling of PAR and SEQ with the same $\omega_1 \in \Omega_1$, but ω_2 is uniformly at random from Ω_2. Let $\mathcal{P}_{\min}(s,u)$ be a minimal path (cf. Observation 3.2) from s to u in PAR according to some fixed $\omega_1 \in \Omega_1$. Notice that the expected time (in subtime-steps) required to traverse $\mathcal{P}_{\min}$ (w. r. t. Ω_2) is a sum of $\mathsf{PAR}(s,u)(\omega_1)$ independent geometric variables with parameter n.

Let $\mathcal{A}_k$ be the event that $\mathsf{PAR}(s,u)(\omega_1) = k$. By using conditional expectations we obtain

$$\begin{aligned}\mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right] &= \sum_{k=1}^{\infty} \mathbf{E}\left[\,\mathsf{SEQ}(s,u) \mid \mathcal{A}_k\,\right] \cdot \mathbf{Pr}\left[\,\mathcal{A}_k\,\right] \\ &\leqslant \sum_{k=1}^{\infty} \frac{1}{n} \cdot (k \cdot n) \cdot \mathbf{Pr}\left[\,\mathcal{A}_k\,\right] = \mathbf{E}\left[\,\mathsf{PAR}(s,u)\,\right].\end{aligned}$$

Now we proceed to prove the remaining inequality $\mathbf{E}\left[\,\mathsf{PAR}(s,u)\,\right] = \mathcal{O}(\mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right])$. For any $x \geqslant 2$ let $\mathcal{B}_x$ be the event that every path $\mathcal{P} = \mathcal{P}(\mathcal{S})$ with $w(\mathcal{P}) = |\mathcal{S}| \geqslant x+1$ satisfies $w(\mathcal{P}) \leqslant 200 \cdot t(\mathcal{P})$. By Lemma 3.6, $\mathbf{Pr}\left[\,\mathcal{B}_x\,\right] \geqslant 1 - 2^{-x}$. By Markov's inequality, we have

$$\mathbf{Pr}\left[\,\mathsf{SEQ}(s,u) \geqslant 2 \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right]\,\right] \leqslant 2^{-1}.$$

Hence for all integers $x \geqslant 1$ we get

$$\mathbf{Pr}\left[\,\mathsf{SEQ}(s,u) \geqslant 2 \cdot x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right]\,\right] \leqslant 2^{-x}.$$

Moreover, we have for all integers $x \geqslant 1$ by using conditional probabilities

$$\begin{aligned} & \mathbf{Pr}\left[\,\mathsf{PAR}(s,u) \geqslant 400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right]\,\right] \\ \leqslant\ & \mathbf{Pr}\left[\,\mathcal{B}_x\,\right] \cdot \mathbf{Pr}\left[\,\mathsf{PAR}(s,u) \geqslant 400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right] \mid \mathcal{B}_x\,\right] + \mathbf{Pr}\left[\,\overline{\mathcal{B}_x}\,\right] \cdot 1 \\ \leqslant\ & \mathbf{Pr}\left[\,\mathsf{PAR}(s,u) \geqslant 400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right] \mid \mathcal{B}_x\,\right] + 2^{-x}. \end{aligned} \tag{3.1}$$

The event $\mathsf{PAR}(s,u) \geqslant 400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right]$ implies that the minimal path from s to u, $\mathcal{P}_{\min}(s,u)$ in the instance of SEQ has a weight of at least $400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right] \geqslant x+1$, where the last inequality holds since with probability at least $1/4$, the vertex s does not send the rumor to any neighbor within the time interval $[1/n, 1]$. Recall that the time to traverse $\mathcal{P}_{\min}(s,u)$ is exactly the weight of this path, $w(\mathcal{P}_{\min}(s,u))$. Now if $\mathcal{B}_x$ holds we know that every path $\mathcal{P}$ with $w(\mathcal{P}) \geqslant x+1$ satisfies $w(\mathcal{P}) \leqslant 200 \cdot t(\mathcal{P})$. Hence if $\mathcal{B}_x$ holds,

$$\mathsf{PAR}(s,u) \leqslant w(\mathcal{P}_{\min}(s,u)) \leqslant 200 \cdot t(\mathcal{P}_{\min}(s,u)) = 200 \cdot \mathsf{SEQ}(s,u),$$

and therefore

$$\begin{aligned}\mathbf{Pr}\left[\,\mathsf{PAR}(s,u) \geqslant 400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right] \mid \mathcal{B}_x\,\right] &= \frac{\mathbf{Pr}\left[\,\mathsf{PAR}(s,u) \geqslant 400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right] \wedge \mathcal{B}_x\,\right]}{\mathbf{Pr}\left[\,\mathcal{B}_x\,\right]} \\ &\leqslant \frac{\mathbf{Pr}\left[\,\mathsf{SEQ}(s,u) \geqslant 2x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right]\,\right]}{1-2^{-x}} \\ &\leqslant \frac{2^{-x}}{1-2^{-x}} \leqslant 2^{-x+1}. \end{aligned} \tag{3.2}$$

and substituting 3.2 into 3.1 gives

$$\mathbf{Pr}\left[\,\mathsf{PAR}(s,u) \geqslant 400x \cdot \mathbf{E}\left[\,\mathsf{SEQ}(s,u)\,\right]\,\right] \ \leqslant\ 2^{-x+1} + 2^{-x} \leqslant 2^{-x+2}. \tag{3.3}$$

Once this bound has been established, the rest of the proof is straightforward. To simplify notation, let $\alpha := 400 \cdot \mathbf{E}[\mathsf{SEQ}(s,u)]$. Using 3.3 and Lemma 2.2 we obtain

$$\begin{aligned}\mathbf{E}[\mathsf{PAR}(s,u)] &\leqslant \sum_{k=0}^{\infty} \mathbf{Pr}[\mathsf{PAR}(s,u) \geqslant k] \\ &= \sum_{i=0}^{\infty}\sum_{j=0}^{\alpha-1} \mathbf{Pr}[\mathsf{PAR}(s,u) \geqslant i\cdot\alpha + j] \\ &\leqslant \sum_{i=0}^{\infty}\sum_{j=0}^{\alpha-1} \mathbf{Pr}[\mathsf{PAR}(s,u) \geqslant i\cdot\alpha] \\ &\leqslant \sum_{i=0}^{\infty} \alpha \cdot 2^{-i+2} \leqslant 8\cdot\alpha\end{aligned}$$

and therefore $\mathbf{E}[\mathsf{PAR}(s,u)] = \mathcal{O}(\mathbf{E}[\mathsf{SEQ}(s,u)])$. This completes the proof of the second claim and the proof of this Theorem. □

We note that on complete graphs, the sequential push algorithm with expected runtime $2\ln n \pm o(1)$ (Proposition 3.13) is much faster than the parallel push algorithm whose runtime is $\ln n + \log_2 n \pm o(1)$ with probability $1 - o(1)$ by [Pit87]. Hence, it is not possible to establish an equivalence of both runtimes up to some $1 + o(1)$-factor.

Employing similar methods as in the proofs before, the following lemma can be shown.

Lemma 3.8 ([Sau07]). *If G is a regular graph, then*

$$\mathsf{SEQ}_{n^{-1}}(G) = \Theta\Big(\mathsf{PUSH} - \mathsf{PULL} - \mathsf{SEQ}_{n^{-1}}(G)\Big).$$

3.5 A Tight Upper Bound for General Graphs

Feige et al. [FPRU90] showed that $\mathsf{PAR}_{n^{-1}}(G) \leqslant 12n\ln n$ for any graph G. This result follows from the fact that the sum of the degrees on any shortest path is at most $3n$ (cf. Lemma 2.28) and hence the expected time to reach the endpoint of this path is $3n$. Using Markov's inequality and the independence between subsequent time periods of length $6n$, the desired bound follows. The stronger bound which we shall prove holds with a weaker probability, but reduces and tightens the constant from 12 to $1 + o(1)$. Recall that $\tilde{\mathcal{O}}$ suppresses all polylogarithmic factors in n, e.g., $n^2 \log^4 n = \tilde{\mathcal{O}}(n^2)$.

Theorem 3.9. *For any graph $G = (V, E)$ it holds that*

$$\mathsf{PAR}_{\tilde{\mathcal{O}}(e^{-\ln^{1/2} n})}(G) = (1 + o(1)) \cdot n \ln n.$$

Proof. Let us briefly describe the basic idea of the proof, leaving out some details. As in [FPRU90], we consider some shortest path $\mathcal{P}$ between the initially informed vertex s

and some vertex u. To improve on the bound of [FPRU90], we make use of the fact that if two vertices v, w share more than $n^{2/3}$ neighbors, the rumor reaches w in $n^{5/6}$ steps, which improves on the trivial bound of $\deg(v)$ if $\deg(v) >> n^{5/6}$. Using this fact we will construct a path $\mathcal{P}'$ such that the sum of the degrees of critical vertices, i. e., large-degree vertices which have not $n^{2/3}$ common neighbors with any following vertex on $\mathcal{P}'$, is $(1 + o(1))n$. This path construction along with some more careful probabilistic analysis constitutes the main ingredients of our improvement. The formal proof follows.

Let $u_0 = s$ be initially informed and u_l be an arbitrary, but fixed vertex different from u_0. Let $\mathcal{P} := (u_0, u_1, \ldots, u_l)$ be a shortest path from u_0 to u_l. Let us define

$$A := \left\{ u_i \in \mathcal{P}, i \neq l \mid \deg(u_i) > \frac{n}{\ln \ln n} \right\},$$
$$B := \left\{ u_i \in \mathcal{P}, i \neq l \mid \deg(u_i) \leqslant \frac{n}{\ln \ln n} \right\}.$$

We begin with a very obvious claim.

Claim. $|A| \leqslant 3 \ln \ln n$.

Proof. By Lemma 2.28, $\sum_{i=0}^{l} \deg(u_i) \leqslant 3n$. As every vertex in A has a degree of at least $\frac{n}{\ln \ln n}$, the claim follows. □

We will deal with both sets of vertices separately and start with the set B, i. e., the set of vertices with a rather small degree. For some vertex $u_i \in B$, let $X_{u_i} := \min_{t \in \mathbb{N}}\{N_t(u_i) = u_{i+1}\} - \mathsf{PAR}(s, u_i)$, i. e., X_{u_i} is the number of rounds required for u_i to transmit the rumor to u_{i+1}. It is evident that X_{u_i} is a geometric variable with parameter $1/(\deg(u_i))$. To upper bound the random variable $X_B := \sum_{u_i \in B} X_{u_i}$ we apply the Chernoff bound of Corollary 2.26. According to the notation of Corollary 2.26, we know that $\mu := \mathbf{E}[X_B] \leqslant 3n$, $p_{\min} = \frac{1}{\max_{v \in B} \deg(v)} \geqslant \frac{\ln \ln n}{n}$ and choose $\gamma = \frac{8}{\ln \ln n} \cdot n \ln n$ to get

$$\begin{aligned}
\mathbf{Pr}\left[X_B \geqslant \frac{8}{\ln \ln n} \cdot n \cdot \ln n + n \right] &\leqslant \frac{2^{p_{\min}\mu}}{e^{\frac{p_{\min}}{2} \cdot (\frac{8}{\ln \ln n} \cdot n \cdot \ln n)}} \\
&\leqslant \frac{2^{p_{\min} \cdot 3n}}{e^{p_{\min} \cdot \frac{2}{\ln \ln n} \cdot n \ln n}} \cdot \frac{1}{e^{\frac{\ln \ln n}{n} \cdot (\frac{2}{\ln \ln n} \cdot n \cdot \ln n)}} \\
&\leqslant e^{-2 \ln n} = n^{-2},
\end{aligned}$$

whenever n is large enough. In the second and more complex part of this proof we consider the vertices of the set A.

Claim. If u_i and u_{i+1} (or u_{i+2}) have more than $n^{2/3}$ common neighbors, then the expected time for the rumor to reach u_{i+1} (or u_{i+2}) from u_i is less than $2 \cdot n^{5/6}$.

Proof. Now fix any vertex $u_i \in \mathcal{P}$. The expected time to inform $n^{1/2}$ vertices of $N(u_i, u_{i+1})$ (or $N(u_i, u_{i+2})$) is at most $\sum_{k=1}^{n^{1/2}} \frac{\deg(u_i)}{n^{2/3}-k} \leqslant \frac{3}{2} n^{5/6}$. Having informed $\sqrt{n}$ of these common neighbors, u_{i+1} (or u_{i+2}) becomes informed in one of the succeeding rounds with probability

at least

$$1-\left(1-\frac{1}{n}\right)^{\sqrt{n}} \geqslant 1-\left(\frac{1}{e}\right)^{1/\sqrt{n}} \geqslant 1-\frac{1}{1+\frac{1}{\sqrt{n}}}=\frac{1}{\sqrt{n}+1},$$

where $e^x \geqslant x+1$ was used in the last inequality. Hence after further expected $\sqrt{n}+1$ rounds, u_{i+1} (or u_{i+2}) becomes informed. □

Unfortunately, the situation where a vertex u_i is connected to many neighbors of u_{i+1} (or u_{i+2}) has also a drawback. When u_{i+1} (or u_{i+2}) are supposed to propagate the rumor further, this vertex may be distracted by the large set of common neighbors with u_i. We therefore require some more detailed analysis.

Let $N(u_i, u_{i+1})$ denote the set of common neighbors of u_i and u_{i+1}. In order to simplify notation we write $u \sim v$ for two vertices u and v if $|N(u,v)| \geqslant n^{2/3}$. In this case, it is helpful to imagine the set $N(u,v)$ as some supervertex connected to u and v by multiple edges. We denote by $S(u,v)$ this supervertex.

In order to benefit from the detours via supervertices, we now describe a transformation of the original path $\mathcal{P}$ (which is an arbitrary, but fixed shortest path) to another path $\mathcal{P}'$ from $u_0 = s$ to u_l. As $\mathcal{P}$, $\mathcal{P}'$ starts with the vertex u_0. Assume that we have constructed the path $\mathcal{P}' = (v_0 = u_0, v_1, \ldots, v_j)$ until some vertex $v_j = u_i, i < l-2$ lying on $\mathcal{P}$. We distinguish now between three cases on how to extend $\mathcal{P}'$ further.

1. $u_i \sim u_{i+2}$. Then we extend $\mathcal{P}'$ by the supervertex $S(u_i, u_{i+2})$ and u_{i+2}, i. e. $\mathcal{P}' = (v_0 = u_0, v_1, \ldots, v_j, v_{j+1} = S(u_i, u_{i+2}), v_{j+2} = u_{i+2})$.

2. $u_i \not\sim u_{i+2} \wedge u_i \sim u_{i+1}$. Here we extend $\mathcal{P}'$ by the supervertex $S(u_i, u_{i+1})$ and u_{i+1},

3. $u_i \not\sim u_{i+2} \wedge u_i \not\sim u_{i+1}$. In this case, we extend $\mathcal{P}'$ just by u_{i+1} as in $\mathcal{P}$.

Note that in the case $v_j = u_{l-1}$, we may only choose between the second and third case.

Let us first consider the subset of vertices $A'' \subseteq A$ which are followed in $\mathcal{P}'$ by a supervertex. Let $X_{A''}$ be the sum over all times it requires for the rumor to proceed from $v_i \in A''$ via some supervertex $S(v_i, v_{i+1})$ to v_{i+1}. As $|A| \leqslant 3 \ln\ln n$, we have $\mathbf{E}[X_{A''}] \leqslant 6n^{5/6} \cdot \ln\ln n$. Thus, the probability that $X_{A''} \geqslant 12n^{5/6} \ln\ln n$ steps is at most $\frac{1}{2}$. By repeating we can decrease the failure probability to

$$\mathbf{Pr}[X_{A''} \geqslant 12n \ln\ln n] \leqslant 2^{-n^{1/6}} < n^{-2}.$$

It remains to consider vertices on $\mathcal{P}'$ in A whose successors on $\mathcal{P}'$ are not supervertices. Let $A' \subseteq A$ be this subset of vertices and let $|N'(u_i)|$ denote the neighbors of u_i which are only adjacent to u_i on $\mathcal{P}$. We will now prove the key claim of this theorem.

Claim. We have

$$\sum_{i:u_i \in A'} \deg(u_i) \leqslant n + \mathcal{O}(n^{2/3} \log\log n).$$

Proof. Recall that any $u_i \in \mathcal{P}$ can only have common neighbors with $u_{i-2}, u_{i-1}, u_{i+1}, u_{i+2}$ since $\mathcal{P}$ is a shortest path. By definition of A',

$$\begin{aligned}
&\sum_{i:u_i\in A'} \deg(u_i)\\
\leqslant\ &\sum_{i:u_i\in A'} |N(u_i,u_{i-2})\cup N(u_i,u_{i-1})\cup N'(u_i)\cup N(u_i,u_{i+1})\cup N(u_i,u_{i+2})| + 2\\
\leqslant\ &\sum_{i:u_i\in A'} |N(u_i,u_{i-2})\cup N(u_i,u_{i-1})| + |N'(u_i)| + n^{2/3} + n^{2/3} + 2\\
\leqslant\ &\sum_{i:u_i\in A'} (|N(u_i,u_{i-2})\cup N(u_i,u_{i-1})|) + \sum_{i:u_i\in A'} |N'(u_i)| + \mathcal{O}(n^{2/3}\cdot|A|)\\
\leqslant\ &\sum_{\substack{i:u_i\in A'\\ u_{i-2}\not\sim u_i, u_{i-1}\not\sim u_i}} (|N(u_i,u_{i-2})| + |N(u_i,u_{i-1})|) + \sum_{\substack{i:u_i\in A'\\ u_{i-2}\not\sim u_i, u_{i-1}\sim u_i}} (|N(u_i,u_{i-2})| + |N(u_i,u_{i-1})|) +\\
&\sum_{\substack{i:u_i\in A'\\ u_{i-2}\sim u_i, u_{i-1}\not\sim u_i}} (|N(u_i,u_{i-2})\cup N(u_i,u_{i-1})|) + \sum_{\substack{i:u_i\in A'\\ u_{i-2}\sim u_i, u_{i-1}\sim u_i}} (|N(u_i,u_{i-2})\cup N(u_i,u_{i-1})|) +\\
&\sum_{i:u_i\in A'} |N'(u_i)| + \mathcal{O}(n^{2/3}\cdot\log\log n).
\end{aligned}$$

Consider some $u_i \in A'$ with $u_{i-1} \not\sim u_i$. Then

$$n^{2/3} \geqslant |N(u_i,u_{i-1})| \geqslant |N(u_i,u_{i-2})\cap N(u_{i-1},u_{i-2})|$$

and so

$$\begin{aligned}
&|N(u_i,u_{i-2})\cup N(u_i,u_{i-1})|\\
\leqslant\ &|N(u_i,u_{i-2})| + |N(u_i,u_{i-1})|\\
\leqslant\ &|N(u_i,u_{i-2})\backslash N(u_{i-1},u_{i-2})| + |N(u_i,u_{i-2})\cap N(u_{i-1},u_{i-2})| + |N(u_i,u_{i-1})|\\
\leqslant\ &|N(u_i,u_{i-2})\backslash N(u_{i-1},u_{i-2})| + 2\cdot n^{2/3}.
\end{aligned}$$

Therefore we can bound the sum of the degrees in A' as follows,

$$
\begin{aligned}
&\sum_{i:u_i\in A'} \deg(u_i)\\
\leqslant\ &\sum_{\substack{i:u_i\in A'\\ u_{i-2}\not\sim u_i, u_{i-1}\not\sim u_i}} (|N(u_i,u_{i-2})| + |N(u_i,u_{i-1})|) + \sum_{\substack{i:u_i\in A'\\ u_{i-2}\not\sim u_i, u_{i-1}\sim u_i}} (|N(u_i,u_{i-2})| + |N(u_i,u_{i-1})|) +\\
&\sum_{\substack{i:u_i\in A'\\ u_{i-2}\sim u_i, u_{i-1}\not\sim u_i}} (|N(u_i,u_{i-2})\backslash N(u_{i-1},u_{i-2})|) + \sum_{\substack{i:u_i\in A'\\ u_{i-2}\sim u_i, u_{i-1}\sim u_i}} (|N(u_i,u_{i-2})\cup N(u_i,u_{i-1})|) +\\
&\sum_{i:u_i\in A'} |N'(u_i)| \ +\ \mathcal{O}(n^{2/3}\cdot\log\log n)\\
=\ &\underbrace{\sum_{\substack{i:u_i\in A'\\ u_{i-2}\not\sim u_i, u_{i-1}\sim u_i}} |N(u_i,u_{i-1})|}_{(2)} \ +\ \underbrace{\sum_{\substack{i:u_i\in A'\\ u_{i-2}\sim u_i, u_{i-1}\not\sim u_i}} |N(u_i,u_{i-2})\backslash N(u_{i-1},u_{i-2})|}_{(3)} +\\
&\underbrace{\sum_{\substack{i:u_i\in A'\\ u_{i-2}\sim u_i, u_{i-1}\sim u_i}} |N(u_i,u_{i-2})\cup N(u_i,u_{i-1})|}_{(4)} \ +\ \underbrace{\sum_{i:u_i\in A'} |N'(u_i)|}_{(1)} \ +\ \mathcal{O}(n^{2/3}\cdot\log\log n).
\end{aligned}
$$

We claim that every vertex not lying on $\mathcal{P}'$ is counted at most once in one of the sums (1) – (4). This will be proved by a case analysis.

1. Let x be some vertex which occurs in sum (1) for some i, i. e., $x \in N'(u_i)$. By definition of N', x is only adjacent to u_i on $\mathcal{P}$. Consequently, x is only adjacent to one vertex of A' and is counted once.

2. Suppose that x is a vertex which occurs in sum (2) for some i, i. e., $x \in N(u_i, u_{i-1})$. Thus, $|N(u_i, u_{i-2})| < n^{2/3}$ but $|N(u_i, u_{i-1})| \geqslant n^{2/3}$ and consequently $u_{i-1} \notin A'$. Hence, the only remaining possibility for x to be counted in one of the four sums is as a common neighbor of u_i and u_{i+1}. Consequently, x could only be counted in (3) with index $i+1$, but this is not possible as $x \notin N(u_{i+1}, u_{i-1})\backslash N(u_i, u_{i-1})$ due to our assumption $x \in N(u_i, u_{i-1})$.

3. Assume that x occurs in sum (3) for some i, i. e., $x \in N(u_i, u_{i-2})\backslash N(u_{i-1}, u_{i-2})$. Clearly, x is counted only once in this case.

4. Finally, let x be counted in sum (4) for some i, i. e., $x \in N(u_i, u_{i-2}) \cup N(u_i, u_{i-1})$. If x is a common neighbor of u_{i-2}, u_{i-1} and u_i, x is only counted once, as $u_{i-1} \sim u_i$ implies $u_{i-1} \notin A'$. Otherwise, x could be a common neighbor of u_{i-1}, u_i and u_{i+1} and x could only be counted additionally in sum (3) with summation index $i+1$. However, as $x \notin N(u_{i+1}, u_{i-1})\backslash N(u_i, u_{i-1})$, x is only counted once.

By the case analysis above we obtain as desired,

$$\sum_{i:u_i\in A'} \deg(u_i) \leqslant n + \mathcal{O}(n^{2/3}\log\log n).$$

□

Let $X_{A'}$ be the number of steps required for the rumor to reach from each vertex of A' the corresponding successor on $\mathcal{P}'$. By the previous argumentation we can express $X_{A'}$ as follows:

$$X_{A'} := \sum_{i:u_i\in A'} \mathsf{Geo}\Big(\frac{1}{\deg(u_i)}\Big),$$

where $\mathbf{E}[\,X_{A'}\,] = \sum_{i:u_i\in A'} \deg(u_i) = n + \tilde{\mathcal{O}}(n^{2/3})$. We note the following lemma.

Lemma 3.10. *Let $Y := \sum_{i=1}^{k} \mathsf{Geo}(\frac{1}{x_i})$ with $x_i \geqslant 2$ for all i. Then for any $y \in \mathbb{N}, y \geqslant k$,*

$$\mathbf{Pr}[\,Y = y\,] \leqslant (y+k-1)^{k-1} \cdot e^{-\frac{y-k}{\sum_{i=1}^{k} x_i}} \cdot \prod_{i=1}^{k}\Big(\frac{1}{x_i}\Big).$$

Proof. We have

$$\begin{aligned}
\mathbf{Pr}[\,Y = y\,] &= \sum_{\substack{1\leqslant\alpha_i\leqslant y-k+1\\ \sum_{i=1}^{k}\alpha_i = y}} \prod_{i=1}^{k}\left(\Big(1-\frac{1}{x_i}\Big)^{\alpha_i-1}\frac{1}{x_i}\right)\\
&\leqslant \left(\prod_{i=1}^{k}\Big(\frac{1}{x_i}\Big)\right)\cdot \sum_{\substack{1\leqslant\alpha_i\leqslant y-k+1\\ \sum_{i=1}^{k}\alpha_i = y}} \prod_{i=1}^{k}\Big(\frac{1}{e}\Big)^{\frac{\alpha_i-1}{x_i}}\\
&= \left(\prod_{i=1}^{k}\Big(\frac{1}{x_i}\Big)\right)\cdot \sum_{\substack{1\leqslant\alpha_i\leqslant y-k+1\\ \sum_{i=1}^{k}\alpha_i = y}} \Big(\frac{1}{e}\Big)^{\sum_{i=1}^{k}\frac{\alpha_i-1}{x_i}}\\
&\leqslant \left(\prod_{i=1}^{k}\Big(\frac{1}{x_i}\Big)\right)\cdot \sum_{\substack{1\leqslant\alpha_i\leqslant y-k+1\\ \sum_{i=1}^{k}\alpha_i = y}} \Big(\frac{1}{e}\Big)^{\frac{\sum_{i=1}^{k}(\alpha_i-1)}{\sum_{i=1}^{k}x_i}}\\
&\leqslant \left(\prod_{i=1}^{k}\Big(\frac{1}{x_i}\Big)\right)\cdot \sum_{\substack{0\leqslant\alpha_i\leqslant y\\ \sum_{i=1}^{k}\alpha_i = y}} \Big(\frac{1}{e}\Big)^{\frac{y-k}{\sum_{i=1}^{k}x_i}} = \left(\prod_{i=1}^{k}\Big(\frac{1}{x_i}\Big)\right)\cdot \binom{y+k-1}{k-1}\Big(\frac{1}{e}\Big)^{\frac{y-k}{\sum_{i=1}^{k}x_i}},
\end{aligned}$$

as claimed. □

Reconsidering $X_{A'}$ and recalling that $|A'| \leqslant 3 \ln \ln n$, Lemma 3.10 yields for any $y \in \mathbb{N}, n \ln n + n(\ln n)^{2/3} \leqslant y \leqslant 12n \log n$ that

$$\begin{aligned}
&\mathbf{Pr}\left[X_{A'} = y \right] \\
&\leqslant (y + |A'| - 1)^{|A'|-1} \cdot e^{-\frac{y-|A'|}{n+\tilde{\mathcal{O}}(n^{2/3})}} \left(\frac{n}{\ln \ln n}\right)^{-|A'|} \\
&\leqslant \left(\mathcal{O}(1) n \ln n\right)^{|A'|-1} \cdot \exp\left(-\frac{n \cdot (\ln n + \ln^{2/3} n)}{n + \tilde{\mathcal{O}}(n^{2/3})} + \frac{\mathcal{O}(\log \log n)}{n + \tilde{\mathcal{O}}(n^{2/3})} \right) \cdot (\ln \ln n)^{\mathcal{O}(\log \log n)} \cdot n^{-|A'|} \\
&\leqslant \tilde{\mathcal{O}}(1) \cdot n^{|A'|-1} \cdot (\ln n)^{\mathcal{O}(\log \log n)} \cdot \exp\left(-(\ln n + \ln^{2/3} n) + \frac{\tilde{\mathcal{O}}(n^{-1/3}) \cdot (\ln n + \ln^{2/3} n)}{1 + \tilde{\mathcal{O}}(n^{-1/3})} \right) \\
&\quad \cdot (\ln \ln n)^{\mathcal{O}(\log \log n)} \cdot n^{-|A'|} \\
&= \tilde{\mathcal{O}}(1) \cdot n^{|A'|-1} \cdot n^{-|A'|} \cdot e^{-\ln n} \cdot e^{-\ln^{2/3} n} \cdot (\ln n)^{\mathcal{O}(\log \log n)} \\
&= \tilde{\mathcal{O}}(1) \cdot n^{-2} \cdot e^{-\ln^{2/3} n} \cdot e^{\mathcal{O}(\log \log n)^2} \\
&= \tilde{\mathcal{O}}\left(n^{-2} \cdot e^{-\ln^{1/2} n} \right).
\end{aligned}$$

By the union bound we have

$$\mathbf{Pr}\left[n \ln n + n(\ln n)^{2/3} \leqslant X_{A'} \leqslant 12n \log n \right] = \tilde{\mathcal{O}}(n^{-1} e^{-\ln^{1/2} n}).$$

As $\mathbf{E}\left[X_{A'} \right] = n + \tilde{\mathcal{O}}(n^{2/3})$ and $X_{A'}$ is a sum of geometric variables, we have (cf. [FPRU90])

$$\mathbf{Pr}\left[X_{A'} \geqslant 12n \log n \right] \leqslant n^{-2}.$$

Therefore we conclude that

$$\begin{aligned}
& \mathbf{Pr}\left[X_{A'} \geqslant n \ln n + n(\ln n)^{2/3} \right] \\
= \; & \mathbf{Pr}\left[n \ln n + n(\ln n)^{2/3} \leqslant X_{A'} \leqslant 12n \log n \right] + \mathbf{Pr}\left[X_{A'} > 12n \log n \right] \\
= \; & \tilde{\mathcal{O}}(n^{-1} e^{-\ln^{1/2} n})
\end{aligned}$$

and together with the previous results of this proof we obtain for $X := X_{A'} + X_{A''} + X_B$

$$\mathbf{Pr}\left[X \geqslant n \ln n + n(\ln n)^{2/3} + 12n \ln \ln n + \frac{8}{\ln \ln n} \cdot n \ln n + n \right] = \tilde{\mathcal{O}}(n^{-1} e^{-\ln^{1/2} n}).$$

Since the endpoint u_l of $\mathcal{P}$ was a fixed but arbitrary vertex, the claim of the theorem follows by the union bound. □

It is clear that the runtime of the graph $\mathsf{K}_{1,n-1}$ reduces to the famous *coupon collector's problem* (Theorem 2.19). Hence, $(1 - o(1)) n \ln n$ steps are necessary to disseminate the rumor to all vertices with probability $1 - o(1)$ which matches the bound of the theorem.

3.6 Use of an Edge-Expansion-Based Measure

The following kind of measure of edge-expansion properties turns out to be more appropriate for bounding the rumor spreading time than the classical edge-expansion from Definition 2.32.

Definition 3.11. *For any graph G and any integer $1 \leqslant m \leqslant n-1$ define*

$$\Lambda(m) := \min_{X \subseteq V(G), |X|=m} \left\{ \sum_{v \in X} \frac{\deg_{X^c}(v)}{\deg(v)} \right\}.$$

If G is Δ-regular, then the first formula can be simplified to

$$\Lambda(m) = \min_{X \subseteq V(G), |X|=m} \left\{ \frac{|E(X, X^c)|}{\Delta(G)} \right\}.$$

Recall that $\mathsf{SEQ}(s, \tilde{n})$ is the time to inform $\tilde{n}$ vertices.

Proposition 3.12. *Let $G = (V, E)$ be a graph and let s be initially informed. We have for any $\tilde{n} \leqslant n$ that*

$$\mathbf{E}\left[\, \mathsf{SEQ}(s, \tilde{n}) \,\right] \leqslant \sum_{m=1}^{\tilde{n}-1} \frac{1}{\Lambda(m)}.$$

If for each $1 \leqslant m' \leqslant \tilde{n}-1$, $\Lambda(m') \cdot \left(\sum_{m=1}^{\tilde{n}-1} (\Lambda(m))^{-1} \right) = \Omega(\log n)$, then

$$\mathsf{SEQ}_{n^{-1}}(s, \tilde{n}) = \mathcal{O}\left(\sum_{m=1}^{\tilde{n}-1} \frac{1}{\Lambda(m)} \right).$$

Proof. Let X_i be the waiting time in subtime-steps until $|I_t|$ increases from i to $i+1$. Note that this time can be computed by the probability of first choosing a vertex $u \in I_t$ and then choosing a vertex $v \in N(u) \cap I_t^c$. Therefore, the probability that I_t increases by 1 in one subtime-step equals

$$\begin{aligned} \sum_{u \in I_t} \mathbf{Pr}\left[\, u \text{ chosen} \,\right] \cdot \mathbf{Pr}\left[\, v \in N(u) \cap I_t^c \text{ chosen} \,\right] &= \sum_{u \in I_t} \frac{1}{n} \cdot \frac{\deg_{I_t^c}(u)}{\deg(u)} \\ &\geqslant \frac{1}{n} \cdot \Lambda(i). \end{aligned}$$

Hence $\mathbf{E}\left[\, X_i \,\right] \leqslant n/\Lambda(i)$ and simply summing up over all i and translating this into time-steps yields the first claim. The second statement follows directly from Corollary 2.26. □

The next result is obtained by Proposition 3.12 and some straightforward calculations (cf. [Sau07]); recall that $\mathbf{E}\left[\, \mathsf{SEQ}(G) \,\right] = \max_{s \in G} \mathbf{E}\left[\, \mathsf{SEQ}(s, V) \,\right]$.

Proposition 3.13. *In the sequential model we have:*

1. *for* $G = \mathsf{K}_{n/2} \times \mathsf{K}_2$, *where* 2 *divides* n, $\mathbf{E}[\,\mathsf{SEQ}(G)\,] \leqslant 4 \ln n + 1$,
2. *for* $G = \mathsf{K}_n$, $(2 - \frac{2}{n}) \ln(n-1) \leqslant \mathbf{E}[\,\mathsf{SEQ}(G)\,] \leqslant (2 - \frac{2}{n}) \ln n$.

One obvious question is whether upper bounds on the edge expansion can be used for lower bounding the broadcast time. We can only state the following trivial bound.

Observation 3.14. *For any graph* $G = (V, E)$, $\mathbf{E}[\,\mathsf{SEQ}(G)\,] \geqslant \frac{1}{\min_{1 \leqslant m \leqslant n-1} \Lambda(m)}$.

3.7 Conclusion

In this chapter we derived several general results on randomized rumor spreading. Our first results were concerned with the runtime of the push algorithm and some modifications of it. These auxiliary results will be used in Chapters 4 and 5. As our main result, we proved a tight upper bound demonstrating that the graph $\mathsf{K}_{1,n-1}$ is (approximately) the worst-case graph for the push algorithm. Finally, we established upper bounds on the push algorithm by means of a certain measure based on the edge expansion of differently large sets. One open problem is whether one can also derive non-trivial lower bounds here.

4. RANDOMIZED RUMOR SPREADING ON CAYLEY GRAPHS

4.1 Introduction

As an introduction to randomized rumor spreading was already given in Section 3, we confine ourselves to Cayley graphs in this section. For more details about Cayley graphs we refer to [Alo95, Bab95].

4.1.1 Motivation

Cayley graphs were introduced as mathematical objects in 1878 by Cayley. The group-theoretical construction of Cayley graphs for the design of interconnection networks was initiated by Akers and Krishnamurthy [AK89]. The objective is to construct networks with a small degree, small diameter, high connectivity and simple algorithms for routing, broadcast etc. [AK89] presented several networks such as star graphs and pancake graphs, and analyzed them with group-theoretical methods. Nowadays, the role of Cayley graphs as interconnection networks seems to be less important, as real-world networks such as the internet are far from being symmetric.

However, Cayley graphs have also received a lot of attention in mathematics, in particular in graph-theory and combinatorics where Cayley graphs are used for explicit constructions of certain graphs. For example, the first explicit expander constructed by Margulis [Mar73] was a Cayley graph. Later Cayley graphs were also used to construct so-called Ramanujan graphs [LPS88], which are, roughly speaking, graphs which are extremal w. r. t. a certain algebraic condition. Finally, some known constructions of expanders with large girth [Mar82] are Cayley graphs, too. Nevertheless, a lot of combinatorial problems about Cayley graphs remain open. One such intriguing question is whether every undirected Cayley graph based on the group of all permutations has a polynomial diameter (cf. [BH05] for more details and some results towards an affirmative answer). Another interesting problem posed by Lovász is whether every Cayley graph has a Hamilton path (cf. [PR04]).

From an algorithmic perspective, Cayley graphs arise frequently when dealing with sorting (e.g., Theorem 4.11) or card-shuffling (cf. next subsection about related work). The reason is that these algorithms can be viewed as moving along edges of Cayley graphs.

4.1.2 Related Work

Feige et al. [FPRU90] proved that the push algorithm takes only $\mathcal{O}(\log n)$ steps on the hypercube with n vertices. They also proved an upper bound of $\mathcal{O}(\log n + \operatorname{diam}(G))$ for any bounded-degree graph. In particular, this implies that the push algorithm has an asymptotically optimal runtime on any of the popular bounded-degree interconnection networks like butterfly, cube-connected-cycles, shuffle-exchange- and De-Bruijn-network [Lei92].

GowriSankaran [Gow94] showed that there is a deterministic broadcast algorithm requiring only $\log_2 n + o(\log n)$ steps in Cayley graphs satisfying the so-called *recursively decomposable* property. He proved that pancake and star graphs satisfy this property. Additionally, the optimal broadcast protocol for the star graph is trivially inherited to transposition graphs, as star graphs are subgraphs of transposition graphs. Other deterministic broadcast algorithms dealing with star graphs are given in [AK89, SWC96]. Also a lot of effort has been spent on determining the best possible (deterministic) broadcast algorithms in De-Bruijn and butterfly networks (cf. [HKP$^+$04, p. 78] for an overview).

There is a long history of the analysis of random processes (in particular Markov chains) on Cayley graphs. A huge body of literature is concerned with *card shuffling procedures*. The main question is the *mixing time*, i. e., how many times must a deck of d cards be shuffled until it is close to random. As an example, consider the shuffling where in each step two random cards are chosen and switched. Diaconis and Shahshahani [DS81] proved that after approximately $(1/2)d \ln d$ steps, the cards are well mixed. For the so-called random-to-top-shuffle, Flatto et al. [FOW85] established that $d \ln d$ rounds suffice. We remark that both card-shuffling schemes are nothing else than random walks (cf. Section 5.2) on the transposition graph and star graph, respectively (cf. Definition 4.3). Another natural, but slower card-shuffling scheme arises when in each step a random adjacent pair of cards is switched. Wilson [Wil04] proved that the mixing time of this scheme is of order $d^3 \log d$.

Recently, a randomized version of the bubble sort algorithm was examined by Benjamini et al. [BBHM05]. It was basically shown that the performance of the bubble sort algorithm is not worsened by comparing in each step a *random* pair of adjacent elements. Moreover, the result is also robust in the sense that it suffices to arrange the pair in the correct order with some probability larger than $1/2$.

One of the most well-known *concentration result* refers to the coupon collector's problem (cf. Theorem 2.19). It is known that the probability for having collected all coupons shows a sharp concentration around the point $n \ln n$. Before this point the probability stays close to zero, it increases doubly exponentially fast near $n \ln n$, while it goes exponentially fast to one past the point $n \ln n$. Similar concentration results are also known for several properties of random graphs like connectivity, emergence of the giant-component, triangle-freeness etc. [AS00].

Diaconis [Dia96] investigated the concentration phenomenon of mixing times of Markov chains. His conclusion was that often a sharp concentration (which he called cutoff) can be explained by a high multiplicity of the second largest eigenvalue of the transition matrix, which is often the case for symmetric Markov chains. This also provides an explanation of a cutoff for the afore-mentioned random-to-random and random-to-top shuffles. More

recently, Peres [DSC06] observed that for many (families of) Markov chains a cutoff exists if and only if the product of the spectral gap (one minus the second largest eigenvalue) and the mixing time (first time the distance to uniformity is less than 1/2) tends to infinity.

4.1.3 Our Results

We analyze the runtime of the push algorithm from Chapter 3 on several important Cayley graphs. In Section 4.3 we prove that for every Cayley graph satisfying four properties, the runtime of the push algorithm is $\mathcal{O}(\log n)$. This implies an upper bound of $\mathcal{O}(\log n)$ for the star graph, pancake graph and transposition graph.

Since the diameter of the bubble sort graph with $d!$ vertices is $\Theta(d^2)$, a separate analysis is required and done in Section 4.4. It follows from the afore mentioned result on a randomized version of the bubble sort algorithm [BBHM05], that *one* fixed vertex of the bubble sort graph becomes informed in expected time $\mathcal{O}(d^2)$. To prove that all vertices become informed in $\mathcal{O}(d^2)$ time with high probability, we develop a martingale-based approach. The whole method depends crucially on the time β required for the rumor to be spread from one vertex to some fixed neighbor (possibly using some path via other vertices). Using the fact that on bubble sort graphs many short node-disjoint paths between adjacent vertices exist, we derive a non-trivial upper bound on β which allows us to conclude that all vertices become informed in $\mathcal{O}(d^2)$ steps. As a by-product of our techniques, we obtain concentration results (for the sequential push algorithm) on hypercubes, star graphs and pancake graphs.

Together with the results of Feige et al. [FPRU90] for the hypercube and bounded degree graphs, we may conclude that on any of the popular interconnection networks the push algorithm takes $\Theta(\log n + \operatorname{diam}(G))$ steps (cf. Figure 4.1). Note that the graph $\mathsf{K}_{\sqrt{n}} \times C_{\sqrt{n}}$ provides an example of a Cayley graph with runtime $\omega(\log n + \operatorname{diam}(G))$ [ES05].

Graph class	Broadcast time	Reference
Hypercube	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log n)$	[FPRU90]
Bounded degree graphs	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log n + \operatorname{diam}(G))$	[FPRU90]
Star Graph	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log n)$	[ES05]
Class including star graph, pancake graph and transposition graph	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log n)$	Theorem 4.6
Bubble sort graph	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\operatorname{diam}(G)) = \Theta(\frac{(\log n)^2}{(\log \log n)^2})$	Theorem 4.20
Hamming graphs (cf. Defintion 5.26)	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log n)$	Theorem 5.27

Fig. 4.1: *Overview on the runtime of the push algorithm on different Cayley graphs.*

4.2 Notations, Definitions and Preliminaries

Definition 4.1. *An (undirected)* Cayley graph $CG = (H, F)$ *is a graph with vertex set* $V(CG)$ *and edge set* $E(CG)$ *defined by a group* $H = (H, \circ)$ *and a subset* $F \subseteq H$ *with* $F^{-1} \subseteq F$ *as follows:*

$$V(CG) \;:=\; H, \quad E(CG) \;:=\; \big\{\{h, h \circ f\} \mid h \in H, f \in F\big\}.$$

We recall some basic facts about Cayley graphs.

Lemma 4.2 ([DSV03]). *Let* $CG = (H, F)$ *be an undirected Cayley graph.*

1. $CG = (H, F)$ *is a simple,* $|F|$*-regular, vertex-transitive graph,*
2. $CG = (H, F)$ *has no loop iff* $\mathsf{id} \notin F$ *with* id *being the neutral element of* H*,*
3. $CG = (H, F)$ *is connected if and only if* F *generates* G*.*

We recall some basic notation of group theory. A *permutation* π is a bijection from $[d] = \{1, 2, \ldots, d\}$ to $\{1, 2, \ldots, d\}$. It is known that the *symmetric group* $\mathfrak{S}_d$ of all $d!$ permutations of $\{1, 2, \ldots, d\}$ forms a group w. r. t. the concatenation ($\circ$) of permutations (to simplify notation, we will frequently omit $\circ$ between the concatenation of two permutations). We use the following common vector representation for permutations, i. e., a permutation $\pi \in \mathfrak{S}_d$ is represented by $(\pi(1), \pi(2), \ldots, \pi(d))$. The *identity permutation* id is the neutral element of the group $\mathfrak{S}_d$ and satisfies $\mathsf{id}(k) = k$ for every $k \in [d]$. A *transposition* $(i\ j), i, j \in [d]$ is the permutation π which maps i to j and vice versa, i. e., $\pi(i) = j$ and $\pi(j) = i$; if $i = j$, then this is just the identity permutation. An *inversion* of a permutation $\pi \in \mathfrak{S}_d$ is a pair $i, j \in [d]$, $i < j$, such that $\pi(i) > \pi(j)$.

We define four important examples of Cayley graphs mentioned in [AK89].

Definition 4.3. *The following undirected Cayley graphs* $CG_d = (H_d, F_d)$ *have all vertex set* $H_d = \mathfrak{S}_d$*. The respective generating sets* $F_d \subseteq \mathfrak{S}_d$ *are defined as follows.*

1. *The* bubble sort graph B_d *is defined by*

$$F(\mathsf{B}_d) \;:=\; \{(i\ \ i+1) \mid 1 \leqslant i \leqslant d-1\}\,.$$

2. *The* pancake graph P_d *is defined by*

$$F(\mathsf{P}_d) \;:=\; \left\{(1\ i)(2\ i-1)\cdots\left(\left\lfloor\frac{i+1}{2}\right\rfloor\ \ \left\lceil\frac{i+1}{2}\right\rceil\right) \;\middle|\; 2 \leqslant i \leqslant d\right\},$$

   e.g., for $d = 6$*,* $F(\mathsf{P}_d) := \{(1\ 2), (1\ 3), (1\ 4)(2\ 3), (1\ 5)(2\ 4), (1\ 6)(2\ 5)(3\ 4)\}$.

3. *The* star graph S_d *is defined by*

$$F(\mathsf{S}_d) \;:=\; \{(1\ \ i) \mid 2 \leqslant i \leqslant d\}.$$

4. The transposition graph T_d *is defined by*

$$F(\mathsf{T}_d) \;:=\; \{(i\ \ j) \mid i,j \in [d], i \neq j\}.$$

For any sequence of distinct numbers $k_1, \ldots, k_i \in [1,d]$, we define $\mathfrak{S}_d(k_1,\ldots,k_i) := \{\pi \in \mathfrak{S}_d \mid \pi(d-i+j) = k_j, j \in [i]\}$.

4.3 A General Class of Cayley Graphs

To compare our main result, we restate the result of [Gow94].

Definition 4.4 ([Gow94]). *A Cayley graph $CG = (\mathfrak{S}_d, F_d)$ fulfills the following property of being* recursively decomposable*, if:*

1. *CG is of degree $d-1$,*
2. *there exists an ordering $f_2, f_3, \ldots, f_d$ of the $d-1$ generators such that*
   (a) *for all i with $2 \leqslant i \leqslant d-1$, $f_i(k) = k$ for any k, $i < k \leqslant d$,*
   (b) *for all i with $2 \leqslant i \leqslant d$, $f_i(1) = i$ and $f_i(i) = 1$.*

Theorem 4.5 ([Gow94]). *For any recursively decomposable Cayley graph $CG = (\mathfrak{S}_d, F)$,*

$$\mathsf{DET}(CG) = \log_2 n + o(\log n),$$

i.e., there is a deterministic broadcast protocol on CG with runtime $\log_2 n + o(\log n)$.

Now we state a similar result for *randomized* broadcast.

Theorem 4.6 ([ES07]). *Assume that a family of Cayley graphs $CG_d = (\mathfrak{S}_d, F_d)$ fulfills the following properties:*

1. *For each $d \in \mathbb{N}$ it holds that $c_1 d^c \leqslant \Delta_d \leqslant c_2 d^c$, where c_1, c_2 and c are constants > 0,*
2. *$F_d \subseteq F_{d+1}$ for each $d \in \mathbb{N}$,*
3. *for every $\tau \in \mathfrak{S}_d$ and $k \in [d]$ it holds that $\mathrm{dist}(\tau, \mathfrak{S}_d(k)) \leqslant c'$ for some constant $c' > 0$,*
4. *$|E(X, X^c)| = \Omega(d^c \cdot |X|)$ for every $X \subseteq V, |X| = \mathcal{O}(d^{c \cdot c'})$.*

Then

$$\mathsf{PAR}_{n^{-1}}(CG_d) = \mathcal{O}(\log n).$$

We will now show that the conditions of Theorem 4.6 are satisfied by *families of* recursively decomposable Cayley graphs (including star graph and pancake graph). We also prove that Theorem 4.6 applies to transposition graphs. To establish the required lower bound on the edge expansion, the following basic result is helpful (Recall that the girth of the graph is the length of a shortest cycle).

Proposition 4.7 ([Sau05]). *Let $G = (V, E)$ be a graph with n vertices, minimum degree δ and girth $g \geqslant 2r$, $r \geqslant 2$. Then, for every subset $X \subseteq V$,*

$$|E(X, X^c)| \geqslant \left(\delta - 2 \cdot \lfloor |X|^{\frac{1}{r-1}} \rfloor\right) \cdot |X|.$$

Lemma 4.8. *Every recursively decomposable Cayley graph has girth at least 6.*

Proof. As every recursively decomposable Cayley graph is bipartite (cf. [Gow94]), it suffices to show that there are no cycles of length 4. Seeking a contradiction, assume $f_{a_2} \circ f_{a_1} = f_{b_2} \circ f_{b_1},\ a_2 \neq a_1,\ b_2 \neq b_1, a_2 \neq b_2,\ a_1 \neq b_1,\ a_1, a_2 > 1, b_2, b_1 > 1$. Observe that $\max\{a_2, a_1\} = \max\{b_2, b_1\}$ and thus w. l. o. g. let $a_1 = b_2 = \max\{b_2, b_1\}$. Hence $f_{a_2} f_{a_1}(a_1) = f_{a_2}(1) = a_2$, but $f_{b_2} f_{b_1}(a_1) \overset{b_1 \leq a_1}{=} f_{b_2}(a_1) = 1$, and the claim follows. □

Proposition 4.9. *Any family of recursively decomposable Cayley graphs $CG_d = (H_d, F_d)$ with $F_d \subseteq F_{d+1}$ (including the star graph S_d and pancake graph P_d), and the transposition graph T_d satisfy all four conditions of Theorem 4.6 for $d \geqslant 3$, and hence for all these graphs $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\log n)$.*

Proof.

1. Consider a family of recursively decomposable Cayley graphs CG_d with $F_d \subseteq F_{d+1}$. As its degree is $d - 1$, the first condition of Theorem 4.6 holds with $c = 1$. By assumption, the second condition is satisfied. For the third condition, consider an arbitrary vertex (permutation) $\tau \in \mathfrak{S}_d$ and assume first that $\tau(j) = k$ for $j > 1$. Then, $\tau f_j f_d(d) = \tau f_j(1) = \tau(j) = k$. Otherwise if $\tau(1) = k$, then $\tau f_d(d) = \tau(1) = k$ and the third condition is satisfied for $c' = 2$. To prove the fourth condition, recall that by Lemma 4.8, $\text{girth}(CG_d) = 6$. Hence for any $X \subseteq V$, $|X| \leqslant \frac{1}{16}(d-1)^2$, Proposition 4.7 gives

$$\begin{aligned} |E(X, X^c)| &\geqslant \left(\delta - 2 \cdot \lfloor |X|^{\frac{1}{2}} \rfloor\right) \cdot |X| \\ &\geqslant \left(d - 1 - 2 \cdot \left\lfloor \frac{1}{4}(d-1) \right\rfloor\right) \cdot |X| \geqslant \frac{d-1}{2} \cdot |X|. \end{aligned}$$

2. Now consider the graph T_d. Here, the degree is $\binom{d}{2}$ and the first condition is satisfied with $c = 2$. The second condition $F(\mathsf{T}_d) \subseteq F(\mathsf{T}_{d+1})$ for any $d \geqslant 1$ is immediate. As $F(\mathsf{T}_d)$ contains every transposition, Condition (3) holds with $c' = 1$. Since T_d is bipartite, we have $\text{girth}(\mathsf{T}_d) \geqslant 4$ and conclude by Proposition 4.7 that for any $X \subseteq V, |X| \leqslant \frac{1}{64}(d(d-1))^2$,

$$\begin{aligned} |E(X, X^c)| &\geqslant \left(\delta - 2 \cdot \lfloor |X|^{\frac{1}{2}} \rfloor\right) \cdot |X| \\ &\geqslant \left(\frac{d(d-1)}{2} - 2 \cdot \left\lfloor \frac{1}{8}(d(d-1))^2 \right\rfloor\right) \cdot |X| \geqslant \frac{d(d-1)}{4} \cdot |X|. \end{aligned}$$

□

4.4 Bubble-Sort Graphs

As Theorem 4.6 is not applicable to the bubble sort graph B_d, we have to do a separate analysis.

Lemma 4.10 ([Knu98]). *Consider the graph* B_d *and fix a permutation* $\sigma \in V(\mathsf{B}_d) = \mathfrak{S}_d$. *Then,* $\operatorname{dist}(\sigma, \mathsf{id})$ *equals the number of inversions in* σ *and therefore* $\operatorname{diam}(\mathsf{B}_d) = \binom{d}{2}$.

Consider the following sorting algorithm in Figure 4.2 which is a randomized version of the Bubble Sort algorithm. If $p < 1$, then there are two sources of randomness: the chosen adjacent pair $(i \;\; i+1)$ and the resulting swap depending on r. Let $\mathsf{SORT}(\sigma)$ be the number of iterations until $\sigma \in \mathfrak{S}_d$ is sorted and define $\mathsf{SORT}_q(\sigma) := \min\{t \in \mathbb{N} \mid \mathbf{Pr}[\mathsf{SORT}(\sigma) \leqslant t] \geqslant 1 - q\}$.

Randomized Bubble-Sort

Input: $\sigma \in \mathfrak{S}_d, \frac{1}{2} < p \leqslant 1$
Output: $\sigma = \mathsf{id}$

while $\sigma \neq \mathsf{id}$ **do**
  choose an integer $i \in \{1, \ldots, d-1\}$ uniformly at random
  let $r = 1$ with probability p and $r = 0$ otherwise
  if $r = 1$ **then**
    if $\sigma(i) > \sigma(i+1)$ **then** swap $\sigma(i)$ and $\sigma(i+1)$
  end if
  if $r = 0$ **then**
    if $\sigma(i) < \sigma(i+1)$ **then** swap $\sigma(i)$ and $\sigma(i+1)$
  end if
end while

Fig. 4.2: *The randomized bubble sort algorithm of [BBHM05]*

Theorem 4.11 ([BBHM05]). *For any constant* $\frac{1}{2} < p < 1$ *and permutation* σ *we have* $\mathsf{SORT}_{e^{-1}}(\sigma) = \mathcal{O}(d^2)$ *and hence* $\mathbf{E}[\mathsf{SORT}(\sigma)] = \mathcal{O}(d^2)$.

Lemma 4.12. *For* B_d, $\mathbf{E}[\mathsf{SEQ}(\tau, \rho)] = \mathcal{O}(d^2)$ *for every permutations* $\tau, \rho \in \mathfrak{S}_d$.

Proof. We will first prove that the time after the rumor has reached id from some $\sigma \in \mathfrak{S}_d$ is stochastically smaller than the time to sort σ in the randomized bubble sort algorithm for $p = 1$. Consider an instance of the randomized rumor spreading algorithm. Recall that such an instance (execution) of the parallel push algorithm is completely described by $(N_{t,v})_{t \in \mathbb{N}, v \in V}$ where each $N_{t,v}$ is chosen uniformly at random among $N(v)$ (cf. Section 3.3). Define a sequence of vertices $v_1, v_2, \ldots, v_{\operatorname{dist}(\sigma,\mathsf{id})}$ inductively as follows:

- $v_1 := \sigma$ and $t_1 := 0$,

- for any $1 < i \leqslant \operatorname{dist}(\sigma, \mathsf{id})$,

$$\begin{aligned} t_i &:= \min\{t \in \mathbb{N} \mid t > t_{i-1}, \operatorname{dist}(N_{t-\mathsf{PAR}(\sigma,v_{i-1}),v_{i-1}}, \mathsf{id}) < \operatorname{dist}(v_{i-1}, \mathsf{id})\} \\ v_i &:= N_{t_i-\mathsf{PAR}(\sigma,v_{i-1}),v_{i-1}}. \end{aligned}$$

So, roughly speaking, we extend the path with endpoint v_{i-1} by the first vertex v_i satisfying the two properties that v_i is closer to id and v_{i-1} sends the rumor to v_i.

It follows that $v_{\operatorname{dist}(\sigma,\mathsf{id})} = \mathsf{id}$ and by Lemma 4.10, $\operatorname{dist}(\sigma, \mathsf{id})$ equals the number of inversion in v_i. Since each $N_{t,v_{i-1}}$ is chosen independently and uniformly at random among $v_i \circ (j\ j+1), 1 \leqslant j \leqslant d-1$, the random variable $t_{\operatorname{dist}(\sigma,\mathsf{id})}$ has the same distribution as $\mathsf{SORT}(\sigma, \mathsf{id})$. Therefore, $\mathsf{PAR}(\sigma, \mathsf{id}) \preceq t_{\operatorname{dist}(\sigma,\mathsf{id})} \stackrel{D}{=} \mathsf{SORT}(\sigma, \mathsf{id})$. Using Theorem 3.5 and Theorem 4.11,

$$\mathbf{E}[\,\mathsf{SEQ}(\sigma, \mathsf{id})\,] \leqslant \mathbf{E}[\,\mathsf{PAR}(\sigma, \mathsf{id})\,] \leqslant \mathbf{E}[\,\mathsf{SORT}(\sigma, \mathsf{id})\,] = \mathcal{O}(d^2)$$

for an arbitrary $\sigma \in \mathfrak{S}_d$. To complete the proof, choose $\sigma = \tau\rho^{-1}$ and apply the automorphism $\pi \mapsto \pi \circ \rho$ to find that $\mathbf{E}[\,\mathsf{SEQ}(\tau\rho^{-1}, \mathsf{id})\,] = \mathbf{E}[\,\mathsf{SEQ}(\tau, \rho)\,]$. □

We will now introduce some further notation for the runtime analysis on B_d. For the use of tail bounds for martingales, we think that it is more convenient to work with random processes whose time-steps are integers. Consequently, we will consider a scaled variant of the sequential push algorithm denoted by SCALED defined as follows. We choose at every integral time-step first some vertex $u \in V$ randomly and then some random neighbor $v \in N(u)$. In case of u being already informed, the vertex v becomes informed. Note that SCALED does exactly the same as the sequential push algorithm SEQ when multiplying each subtime-step $t \in \mathbb{T}$ by n.

Let $s \in V$ be the initially informed vertex and consider any nonempty subset $V' \subseteq V$. According to the notation introduced in Chapter 3, define $\mathsf{SCALED}(s, V') := \min\{t \in \mathbb{N} : V' \subseteq I_t \mid I_0 = \{s\}\}$ and for any $0 < q < 1$, $\mathsf{SCALED}_q(s, V') := \min\{t \in \mathbb{N} : \mathbf{Pr}[\,V' \subseteq I_t \mid I_0 = \{s\}\,] \geqslant 1 - q\}$.

Define $Z_0 := \mathbf{E}[\,\mathsf{SCALED}(s, V')\,]$ and more generally, for any $t \in \mathbb{N}\backslash\{0\}$

$$Z_t := \mathbf{E}[\,\mathsf{SCALED}(s, V') \mid I_0, I_1, \ldots, I_t\,].$$

Notice that $\mathbf{E}[\,\mathsf{SCALED}(s, V') \mid I_0, I_1, \ldots, I_t\,] = \mathbf{E}[\,\mathsf{SCALED}(s, V') \mid I_t\,]$, as I_{t+1} only depends on I_t. The fact that Z_t is a martingale follows from the so-called Doob martingale construction (cf. Lemma 2.17). Z_t estimates the expected time to inform v conditioned on the outcomes of the first t time-steps. We list some basic properties of Z_t.

Observation 4.13. *For any graph G, the martingale Z_t has the following properties.*

1. *If $Z_t \leqslant t$, then $V' \subseteq I_t$ and consequently, $Z_t = Z_{t+1} = \cdots$.*

2. *For any* $A \subseteq B \subseteq V$ *we have*

$$\mathbf{E}\left[\,\mathsf{SCALED}(s,v) \mid I_t = A\,\right] \geqslant \mathbf{E}\left[\,\mathsf{SCALED}(s,v) \mid I_t = B\,\right].$$

3. *For any subset* $A \subseteq V$ *and any* $t \in \mathbb{N}$,

$$\mathbf{E}\left[\,\mathsf{SCALED}(s,v) \mid I_{t-1} = A\,\right] + 1 = \mathbf{E}\left[\,\mathsf{SCALED}(s,v) \mid I_t = A\,\right].$$

One building block of our approach will be the following concentration inequality.

Theorem 4.14 ([McD98, CL07]). *Let* $Z_0, Z_1, \ldots$ *be a martingale w. r. t. the sequence* $I_0, I_1, \ldots$ *such that for all* $k \geqslant 1$

1. $|Z_k - Z_{k-1}| \leqslant M$,
2. $\mathbf{Var}\left[\,Z_k \mid I_0, \ldots, I_{k-1}\,\right] \leqslant \sigma_k^2$.

Then for any $i \geqslant 0$ *and any* $\lambda > 0$

$$\mathbf{Pr}\left[\,|Z_i - Z_0| \geqslant \lambda\,\right] \leqslant 2 \cdot \exp\left(-\frac{\lambda^2}{2 \cdot (\sum_{k=1}^{i} \sigma_k^2 + M\lambda/3)}\right).$$

Thus to obtain good tail estimates, we have to derive upper bounds on the differences $Z_k - Z_{k-1}$ and on the conditioned variance of Z_k. To do so, we make the following definition.

Definition 4.15. *For any graph* G *define* $\beta(G) := \max_{\{u,v\} \in E(G)} \mathbf{E}\left[\,\mathsf{SCALED}(u,v)\,\right]$.

So, $\beta(G)$ provides an upper bound on the expected time required for the rumor to reach some fixed $v \in N(u)$ from some vertex u, possibly via some path of length larger than 1. A trivial bound is $\beta(G) \leqslant \Delta(G)$ on which following the lemma improves for certain graphs.

Lemma 4.16. *Let* G *be a* Δ*-regular graph with* $\Delta = \omega(1)$. *If for every two adjacent vertices* $u, v \in N(u)$ *there are* $\Omega(\Delta)$ *node-disjoint paths of length at most* $1 < l = \mathcal{O}(1)$, *then*

$$\beta(G) = \mathcal{O}\left(\Delta^{\frac{l-1}{l}} \cdot n\right).$$

Proof. In this proof we consider the parallel push algorithm PAR. Consider the set of node-disjoint paths $\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_{c\Delta}$, $0 < c \leqslant 1$, of length l between u and v. Divide the vertices into levels $\mathcal{L}_0, \mathcal{L}_1, \ldots, \mathcal{L}_l$, where $\mathcal{L}_i := \{z \in \mathcal{P}_j, 1 \leqslant j \leqslant c\Delta \mid \mathrm{dist}(u,z) = i, \mathrm{dist}(z,v) = l - i\}$. By assumption, $|\mathcal{L}_i| = c\Delta$ if $1 \leqslant i \leqslant l-1$, and $|\mathcal{L}_i| = 1$ otherwise. We first consider the waiting time X_0 until $\frac{c}{2}\Delta^{\frac{l-1}{l}}$ vertices of $\mathcal{L}_1$ become informed. Clearly,

$$\mathbf{E}\left[\,X_0\,\right] = \sum_{k=1}^{\frac{c}{2}\Delta^{\frac{l-1}{l}}} \frac{\Delta}{\Delta - k} \leqslant \Delta \cdot \sum_{k=1}^{\frac{c}{2}\Delta^{\frac{l-1}{l}}} \frac{1}{\frac{1}{2}\Delta} \leqslant \Delta^{\frac{l-1}{l}}.$$

As a result of Markov's inequality, $\mathbf{Pr}[X_0 \geqslant 4 \cdot \mathbf{E}[X_0]] \leqslant \frac{1}{4}$.

Assuming that there are $\frac{c}{2}\Delta^{\frac{l-i}{l}}$ informed vertices in $\mathcal{L}_i, i < l-1$ we consider phase i with waiting time X_i until $\frac{c}{2}\Delta^{\frac{l-i-1}{l}}$ vertices become informed in $\mathcal{L}_{i+1}$. The probability that some vertex $v \in \mathcal{L}_{i+1}$ becomes informed within the next $4\Delta^{\frac{l-1}{l}}$ steps, given that its neighbor $u \in \mathcal{L}_i$ has been informed in phase i, is

$$\begin{aligned} 1-\left(1-\frac{1}{\Delta}\right)^{4\Delta^{\frac{l-1}{l}}} &\geqslant 1-\exp(-4\Delta^{-\frac{1}{l}}) \\ &\geqslant 1-\frac{1}{4\Delta^{-\frac{1}{l}}+1} \geqslant \frac{4\Delta^{-\frac{1}{l}}}{1+4\Delta^{-\frac{1}{l}}} \geqslant 2\Delta^{-\frac{1}{l}}, \end{aligned}$$

where the second inequality is due to $e^{-x} \leqslant \frac{1}{x+1}$. Hence the expected number of informed neighbors in $\mathcal{L}_{i+1}$ after $4\Delta^{\frac{l-1}{l}}$ steps is at least

$$\frac{c}{2}\Delta^{\frac{l-i}{l}} \cdot 2\Delta^{-\frac{1}{l}} = c\Delta^{\frac{l-i-1}{l}}.$$

Using the Chernoff bound for Bernoulli variables (Theorem 2.11), $\frac{c}{2}\Delta^{\frac{l-i-1}{l}}$ vertices become informed after $4\Delta^{\frac{l-1}{l}}$ steps with probability $1-o(1)$, as $\Delta = \omega(1)$. Finally, suppose that $\mathcal{L}_{l-1}$ contains $\frac{c}{2} \cdot \Delta^{\frac{1}{l}}$ informed vertices. As each of these vertices informs $v \in \mathcal{L}_l$ with probability $\frac{1}{\Delta}$ in one step, v becomes informed after $\frac{2}{c}\Delta^{\frac{l-1}{l}}$ steps with probability

$$1-\left(1-\frac{1}{\Delta}\right)^{\frac{c}{2}\Delta^{\frac{1}{l}}\cdot\frac{2}{c}\Delta^{\frac{l-1}{l}}} \geqslant 1-\left(1-\frac{1}{\Delta}\right)^{\Delta} \geqslant 1-\frac{1}{e},$$

if $\Delta \geqslant 2$. Summarizing, we have shown that with probability $1-\frac{1}{4}-\frac{1}{e}-(l-2)\cdot o(1) \geqslant 1-\frac{2}{3}$,

$$\begin{aligned} \sum_{k=0}^{l-1} X_k &\leqslant X_0 + \sum_{k=1}^{l-2} X_k + X_{k-2} \\ &\leqslant 4\Delta^{\frac{l-1}{l}} + (l-1)\cdot 4\Delta^{\frac{l-1}{l}} + \frac{2}{c}\Delta^{\frac{l-1}{l}} = \mathcal{O}\left(\Delta^{\frac{l-1}{l}}\right). \end{aligned}$$

Hence $\mathsf{PAR}_{2/3}(u,v) = \mathcal{O}(\Delta^{\frac{l-1}{l}})$ and thus $\mathbf{E}[\mathsf{PAR}(u,v)] = \mathcal{O}(\Delta^{\frac{l-1}{l}})$. By Theorem 3.5,

$$\mathbf{E}[\mathsf{SCALED}(u,v)] = n \cdot \mathbf{E}[\mathsf{SEQ}(u,v)] = \mathcal{O}\Big(n \cdot \mathbf{E}[\mathsf{PAR}(u,v)]\Big) = \mathcal{O}\left(n \cdot \Delta^{\frac{l-1}{l}}\right),$$

and the claim follows. □

We now apply this Lemma to three Cayley graphs.

Corollary 4.17. *For the graphs* $\mathsf{B}_d, \mathsf{T}_d$ *and* Q_d,

$$\beta(G) = \mathcal{O}\left(\Delta^{\frac{2}{3}} \cdot n\right).$$

Proof. We use Lemma 4.16 and prove the existence of $\Omega(\Delta(G))$ node-disjoint paths of length 3 between any pair of adjacent vertices u, v in each of the three graphs.

1. We start with B_d. W. l. o. g. consider the two vertices $\pi = \mathsf{id}$ and $\sigma = \mathsf{id}(i\ i+1)$, where $1 \leqslant i \leqslant d-1$. Clearly, $(j\ j+1)(i\ i+1)(j\ j+1) = (i\ i+1)$ for every $j \in \{1, \ldots, i-1\} \cup \{i+2, \ldots, d-1\}$. Hence,

$$\bigcup_{j \in \{1,\ldots,i-1\} \cup \{i+2,\ldots,d-1\}} \Big(\mathsf{id}, (j\ j+1), (j\ j+1)(i\ i+1), (i\ i+1)\Big)$$

   is a set of $d-3$ node-disjoint paths of length 3 from π to σ.

2. The reasoning for T_d is very similar. Again, consider $\pi = \mathsf{id}$ and $\sigma = \mathsf{id}(r\ s)$, where $r, s \in [d], r < s$. Clearly, $(x\ y)(r\ s)(x\ y) = (r\ s)$, for any $x, y \in [d], x, y \notin \{r, s\}$ and the claim follows.

3. Consider any $u \in \{0,1\}^d$ and $v = u(i)$, i. e., v is obtained by flipping bit i in v. It is easily seen that $\cup_{j=1, j\neq i}^{d}(u, u(j), u(j)(i), u(j)(i)(j) = v)$ is a set of $d-1$ node-disjoint paths from u to v.

□

Lemma 4.18 ([McD98]). *Let* X *be a random variable with* $\mathbf{E}[X] - a \leqslant X \leqslant \mathbf{E}[X] + b$ *for* $a, b \geqslant 0$. *Then,* $\mathbf{Var}[X] \leqslant a \cdot b$.

Let us now reconsider the martingale Z_k.

Lemma 4.19. *For any graph* $G = (V, E)$ *we have for all* $k \in \mathbb{N}\backslash\{0\}$

$$-\beta(G) \leqslant Z_k - Z_{k-1} \leqslant 1,$$
$$\mathbf{Var}[Z_k \mid I_{k-1}] \leqslant \beta(G).$$

Proof. We begin by proving the first inequality. Recall that $Z_k - Z_{k-1}$ is a random variable depending on I_k, while I_{k-1} is fixed. Let $I_{k-1} = I$ for some arbitrary $I \subseteq V$. We distinguish now between the cases $I_k = I$ or $I_k = I \cup \{v\}$ for some $v \in N(I) \cap I^c$.

1. Let us first consider the case $I_k = I$. By Observation 4.13 (3),

$$\mathbf{E}[\mathsf{SCALED}(s, V') \mid I_k = I] = \mathbf{E}[\mathsf{SCALED}(s, V') \mid I_{k-1} = I] + 1$$

   and hence $\mathbf{E}[\mathsf{SCALED}(s, V') \mid I_k = I] - \mathbf{E}[\mathsf{SCALED}(s, V') \mid I_{k-1} = I] = 1$.

2. Assume that $I_k = I \cup \{v\}$ for some $v \in N(u) \cap I^c, u \in I$. Observation 4.13 yields

$$\begin{aligned} & \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_{k-1} = I\,] \\ \leqslant\ & \mathbf{E}\,[\,\mathsf{SCALED}(s, v) \mid I_{k-1} = I\,] - (k-1) + \\ & \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_{k-1} = I \cup \{v\}\,] \\ \leqslant\ & \mathbf{E}\,[\,\mathsf{SCALED}(s, v) \mid I_0 = I\,] + \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_{k-1} = I \cup \{v\}\,] - 1 \\ \leqslant\ & \beta(G) + \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_k = I \cup \{v\}\,] \end{aligned}$$

and thus

$$\mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_k = I \cup \{v\}\,] - \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_{k-1} = I\,] \geqslant -\beta(G).$$

Another application of Observation 4.13 results in

$$\begin{aligned} & \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_k = I \cup \{v\}\,] - \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_{k-1} = I\,] \\ \leqslant\ & \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_k = I\,] - \mathbf{E}\,[\,\mathsf{SCALED}(s, V') \mid I_{k-1} = I\,] \\ =\ & 1. \end{aligned}$$

Combining the two cases, we have shown the first claim $-\beta(G) \leqslant Z_k - Z_{k-1} \leqslant 1$.

We will now prove the second claim. By the first claim we know that Z_k is a random variable taking only values in the interval $[Z_{k-1} - \beta(G), Z_{k-1} + 1]$. Moreover, by the martingale property we have $\mathbf{E}\,[\,Z_k \mid I_{k-1}\,] = Z_{k-1}$. It follows from Lemma 4.18 that $\mathbf{Var}\,[\,Z_k\,] \leqslant \beta(G)$ which finishes the proof of the second claim. □

We now apply our machinery to show the main result of this section.

Theorem 4.20. *For the bubble sort graph* B_d*,*

$$\mathsf{SCALED}_{n^{-1}}(\mathsf{B}_d) = \mathcal{O}(d^2 \cdot n),$$

and therefore $\mathsf{PAR}_{n^{-1}}(\mathsf{B}_d) = \mathsf{SEQ}_{n^{-1}}(\mathsf{B}_d) = \mathcal{O}(d^2)$.

Proof. Assume that $\tau \in V(\mathsf{B}_d)$ is informed at the beginning. We aim at proving that an arbitrary fixed $\sigma \in V(\mathsf{B}_d)$ becomes informed after $\mathcal{O}(d^2 \cdot n)$ steps with probability $1 - n^{-2}$, by which the claim follows. Substituting $s = \tau$ and $V' = \{\sigma\}$ into the martingale Z_k we obtain

$$\begin{aligned} \mathbf{Pr}\,[\,\sigma \notin I_{2C \cdot d^2 n}\,] &\leqslant \mathbf{Pr}\,\big[\,Z_{2C \cdot d^2 n} > 2C \cdot d^2 n\,\big] && \text{(by Observation 4.13)} \\ &= \mathbf{Pr}\,\big[\,Z_{2C \cdot d^2 n} - Z_0 > 2C \cdot d^2 n - Z_0\,\big] \\ &\leqslant \mathbf{Pr}\,\big[\,Z_{2C \cdot d^2 n} - Z_0 > C \cdot d^2 n\,\big] && \text{(by Lemma 4.12),} \end{aligned}$$

provided that C is a sufficiently large constant. Combining Theorem 4.14 with $\beta(\mathsf{B}_d) = \mathcal{O}(d^{2/3}n)$ (by Lemma 4.16) and $|Z_k - Z_{k-1}| \leqslant \max\{1, \beta(G)\}$, $\mathbf{Var}[\, Z_k - Z_{k-1}\,] = \mathcal{O}(d^{2/3}n)$ (by Lemma 4.19) yields

$$\begin{aligned}\mathbf{Pr}[\, \sigma \notin I_{2C\cdot d^2 n}\,] &\leqslant 2 \cdot \exp\left(-\frac{(C\cdot d^2 n)^2}{2\cdot\left(\sum_{k=1}^{2C\cdot d^2 n} d^{2/3}n + \frac{d^{2/3}n\cdot C\cdot d^2 n}{3}\right)}\right)\\ &\leqslant 2\cdot\exp\left(-\frac{C^2 d^4 n^2}{\frac{14}{3}\cdot(C\cdot d^{8/3}n^2)}\right)\\ &= 2\cdot\exp\left(-\frac{3}{14}\cdot C\cdot d^{4/3}\right) \leqslant d^{-2d} \leqslant (d!)^{-2},\end{aligned}$$

and the claim follows by taking the union bound over all $\sigma \in \mathfrak{S}_d$. □

With almost the same arguments, we obtain concentration results on $\mathsf{SCALED}(G) = \mathsf{SCALED}(s, V(G))$ by choosing $V' = V$ for the martingale Z_k. For simplicity, we only state them for $\mathsf{SCALED}(G)$, however, it is clear that by an appropriate scaling, equivalent bounds also hold for $\mathsf{SEQ}(G)$.

Theorem 4.21. *For* SCALED *the following concentration results hold.*

1. *For the graph* B_d*, we have for any* α *with* $d^{4/3}n < \alpha = o(Z_0) = o(d^2 n)$,

$$\mathbf{Pr}[\, |\mathsf{SCALED}(\mathsf{B}_d) - Z_0| \geqslant \alpha\,] \leqslant \exp\left(-\Omega\left(\frac{\alpha^2}{d^{8/3}n^2}\right)\right).$$

2. *For any graph* $CG_d \in \{\mathsf{S}_d, \mathsf{P}_d\}$ *we have for any* $d^{5/6}n < \alpha = o(Z_0) = o(d\log d\cdot n)$,

$$\mathbf{Pr}[\, |\mathsf{SCALED}(CG_d) - Z_0| \geqslant \alpha\,] \leqslant \exp\left(-\Omega\left(\frac{\alpha^2}{d^{5/3}n^2}\right)\right).$$

3. *For the graph* Q_d*, we have for any* $d^{5/6}n < \alpha = o(Z_0) = o(dn)$,

$$\mathbf{Pr}[\, |\mathsf{SCALED}(\mathsf{Q}_d) - Z_0| \geqslant \alpha\,] \leqslant \exp\left(-\Omega\left(\frac{\alpha^2}{d^{5/3}n^2}\right)\right).$$

Proof. We only give the proof for the first statement, as the other two are shown in the same way. Recall that $\beta(G) = \mathcal{O}(d^{2/3}n)$ by Lemma 4.16. Substituting $s = \tau$ and $V' = V$

into Z_i, we obtain by Observation 4.13 and Theorem 4.14,

$$\begin{aligned}
\mathbf{Pr}\left[\,\mathsf{SCALED}(\tau, V) \geqslant Z_0 + \alpha\,\right] &\leqslant \mathbf{Pr}\left[\,Z_{Z_0+\alpha} \geqslant Z_0 + \alpha\,\right] \\
&= \mathbf{Pr}\left[\,Z_{Z_0+\alpha} - Z_0 \geqslant \alpha\,\right] \\
&\leqslant 2 \cdot \exp\left(-\frac{\alpha^2}{2 \cdot \left(\sum_{k=1}^{Z_0+\alpha} d^{2/3}n + \frac{d^{2/3}n\cdot\alpha}{3}\right)}\right) \\
&\leqslant 2 \cdot \exp\left(-\frac{\alpha^2}{\mathcal{O}(d^2 \cdot n + \alpha) \cdot d^{2/3}n + \frac{2d^{2/3}n\cdot\alpha}{3}}\right) \\
&= \exp\left(-\Omega\left(\frac{\alpha^2}{d^{8/3}n^2}\right)\right).
\end{aligned}$$

□

Finally, we remark that by connecting two complete graphs by one edge we obtain a graph G where $\mathsf{SCALED}(G)$ is *not* highly concentrated around its mean.

4.5 Conclusion

In this chapter we derived tight bounds on the runtime of a randomized rumor spreading algorithm on several important Cayley graphs. First, we proved that the push algorithm takes only $\mathcal{O}(\log n)$ steps on every graph lying in a certain subclass of Cayley graphs. From this we concluded that the runtime is asymptotically optimal on star graphs, pancake graphs and transposition graphs.

To obtain also a tight bound for bubble sort graphs, we combined recent results of [BBHM05] on the runtime of a randomized version of the bubble sort algorithm with a new martingale-based approach. Roughly, we showed that if the time required to spread the rumor to any fixed neighbor is small, then the runtime is highly concentrated around its expected value. As a by-product, we derived concentration results for star graphs, pancake graphs and hypercubes.

Our results leave open some of the following questions. To begin with, can the result of Theorem 4.6 be extended to the *mixing time* (cf. Chapter 5), i. e., the time until a random walk has approached the stationary distribution up to some constant deviation? For star graphs and transposition graphs, the mixing time is known to be $\Theta(d \log d)$ (cf. Subsection 4.1.2). For the pancake graph, Fill [Fil91] mentioned an upper bound of $\mathcal{O}(d^4 \log d)$. He conjectured that the mixing time is $\Theta(d \log d)$, but to the best of our knowledge, no progress has been made towards this conjecture.

The techniques of Section 4.4 do not work for the transposition graph, and it might be the case that the time required to inform a fixed neighbor (β) is not significantly smaller than the time to inform all vertices in the graph. Nevertheless, we believe that the runtime is also concentrated on this graph class and believe that it could be proven by some other

techniques such as a decomposition into disjoint subgraphs. However, Rabinovich [Rab07] even conjectured that such type of concentration holds for *all* Cayley graphs.

Another limitation of our concentration results is the restriction to the sequential push algorithm. It would be interesting to see if one could prove similar results for the parallel push algorithm.

5. RANDOMIZED RUMOR SPREADING VS. RANDOM WALKS

5.1 Introduction

For a survey on random walks we refer the reader to [Lov93], which also deals with the cover time and its connection to electrical networks. Another recommendable survey is [Gur00] with an emphasis on techniques to bound the mixing time. Also the textbooks [MR95, MU05] contain a lot of material about random walks and its application in computer science.

5.1.1 Motivation

A *random walk* on a graph $G = (V, E)$ with n vertices is the following process. Starting from some vertex $s \in V(G)$, we select a neighbor of s uniformly at random and move to this neighbor. After that, we select a neighbor of this new vertex randomly, move to it and so on. By repeating this procedure we obtain an infinite (random) sequence of vertices called *random walk*. While here we shall only deal with discrete-time random walks on discrete structures (graphs), there are also continuous versions of random walks arising in mathematics and physics. The archetypal example is the *Brownian Motion* [GS01] which is used as a mathematical model for random movements of molecular particles, the evolution of stock prices etc.

The classical theory of random walks was also concerned with infinitely large networks. A classic result of Pólya states that a random walk in a d-dimensional grid returns to the starting point in expected finite time, if and only if $d \leqslant 2$. Spurred by recent algorithmic developments, the interest has shifted from infinite to finite networks and, correspondingly, from qualitative to quantitative questions like the following ones. How many steps are required until a random walk has visited all vertices? How many steps are required to get a "good" sample of all vertices? For this purpose we define the *cover time* of a random walk on a graph G as the expected number of steps to visit all vertices on G. Closely related to the cover time is the commute time between two vertices u and v which is the expected time for a random walk starting from u to return to u after at least one visit to v. The *mixing time* is the first time-step t after the distance between the distribution at step t and the equilibrium distribution is smaller than some given threshold.

While parameters like the cover and mixing time are interesting in their own right, there is a variety of results relating them to graph-theoretical parameters. Since the random walk

is a repeated matrix-vector multiplication, it does not come as a great surprise that the spectrum, i. e., the eigenvalues of the graph, characterize the mixing time fairly accurately. More precisely, the reciprocal of the difference between 1 and the second largest eigenvalue of the transition matrix, called *spectral gap*, captures this rate of convergence. Along with the spectral gap comes a geometric measure called *conductance*, which can be viewed as the edge expansion of the underlying graph. The conductance provides similar but slightly weaker bounds on the mixing time than the spectral gap.

Another fruitful connection of random walks is the one to electrical networks. Viewing the graph as an electrical network, the *effective resistance* between u and v is the voltage difference when one ampere is injected into u and removed from v. It turns out that the commute time between two vertices u and v is precisely the effective resistance between them times twice the number of edges. Of great use for bounding the effective resistance (and thus the commute time) are the following rules known as *Rayleigh's Short-cut Principle*: the effective resistance is never increased by adding an edge or by gluing two vertices together [CRR$^+$97] (possibly leading to multiple edges).

An early application of the cover time in computer science was the so-called *undirected $s-t$-connectivity* problem USTCON defined as follows. Given some graph G and two vertices $s, t \in V(G)$, the task is to decide whether s and t are connected or not. It is clear that a depth-first search solves this problem in $\mathcal{O}(|E|)$ steps, however, it requires $\Omega(n)$ space. Another solution is to take a random walk starting from s and see whether t is visited during the first $2n^3$ steps. This gives the correct answer with some constant probability and uses only $2n^3$ steps and $\mathcal{O}(\log n)$ space for keeping track of the random walk's position. In a recent result by Reingold [Rei05] it was shown that one can also decide USTCON *deterministically* by using $\mathcal{O}(\log n)$ space and polynomial time.

An associated parameter to the cover time is the so-called *blanket time* (also known as multiple cover time) [WZ96]. The blanket time is the first time-step t after which the number of visits to any vertex differs from the expected number of visits within t steps only by a factor of 2. It is obvious that the blanket time is bounded below by the cover time and it is conjectured that they are the same up to some constant factor [WZ96].

For many #P-problems such as counting independent sets in a graph, counting the number of Hamilton cycles in a graph or computing the permanent of a matrix, efficient approximation algorithms based on random walks were developed [Sin93, MR95, MU05]. Often it is relatively easy to construct a random walk with the desired stationary distribution, e.g., a distribution which assigns every independent set the same probability. The crux however is to bound the mixing time. This explains why such a variety of techniques to upper bound the mixing time have been developed (cf. [Lov93, Gur00]).

Moreover, random walks can be viewed as a simple diffusive load balancing scheme when interpreting the (normalized) load distribution as a probability distribution. Also the analysis of randomized algorithms can frequently be reduced to questions about random walks on certain graphs (see [MR95, p. 128] for a simple example).

5.1.2 Related Work

As most results in this chapter deal with the cover time, we correspondingly put emphasis on the related work on the cover time. Still, there is a vast body of literature devoted to the cover time and we can only point to some results directly related to our study. At the end of this subsection, we briefly explain the basic connection between mixing and cover time. For more details on the state of the art in the field of mixing time, the reader is referred to [Lov93, MR95, Gur00, MU05].

Cover Time. The study of the cover time was initiated 1979 by Aleliunas et al. [AKL+79]. Amongst other results, it was shown that the cover time is upper bounded by the weight of a minimum spanning tree of G, where the edges are weighted according to the commute times between the corresponding vertices. In particular, this implies a (polynomial) upper bound of $\mathcal{O}(n^3)$ for any graph. This approach was refined by Feige [Fei97b] to get improved upper bounds on the cover time, e. g., an upper bound of $(2 - o(1)) \cdot n^2$ for regular graphs was derived.

In [Ald83] Aldous showed that for certain Cayley graphs, the cover time is $(1 + o(1))n \ln n$. Broder and Karlin [BK89] proved several bounds which rely on the spectral gap of the transition matrix. Their bounds imply that the cover time on any regular edge-expander is $\Theta(n \log n)$. The seminal work of Chandra et al. [CRR+97] established a close connection between the electrical resistance of a graph and its cover time. Furthermore, several methods for bounding the resistance were introduced and applied to obtain tight bounds for various graph classes. One of their results relates the cover time to the vertex-expansion of the graph and can be considered as an improvement of the bound from [BK89] based on the spectral gap.

Also a lot of effort has been made to prove Aldous' conjecture that the cover time is at least $(1 + o(1))n \ln n$ on every graph. Affirmative answers for special graph classes had been given [BK89, Zuc92], until the conjecture was finally resolved by Feige [Fei95a]. In the same year, Feige also proved an almost exact upper bound of $(4/27)n^3 + o(n^3)$ for general graphs [Fei95b]. This bound is matched by the so-called *lollipop-graph*, a clique of size $2n/3$ with an $n/3$-path attached.

Winkler and Zuckerman [WZ96] introduced the *blanket time*. They conjectured that the blanket time is upper bounded by the cover time up to some constant factor, and indeed this was shown for several important cases. In [KKLV00] Kahn et al. proved that the blanket time is at most the cover time times an $\mathcal{O}((\ln \ln n)^2)$-factor on *any* graph.

More recently, research on the cover time seems to have focused on special graph classes, e.g., random regular graphs [CF05], random geometric graphs [AE06] and planar graphs [JS00].

Motivated by time-space tradeoffs for the USTCON problem, one has also considered multiple random walks. A usual assumption is that these random walks move independently and synchronously. One possibility is to start each random walk from a randomly chosen vertex. [BKRU94] proved a general result which implies that in this case n random walks cover any graph in $\mathcal{O}(\log^2 n)$ steps. Recently, Alon et al. [AAK+08] considered multiple random walks starting all from the *same* vertex s. They investigated the tradeoff

between adding more random walks on s and the resulted decline of the cover time on different graph classes. Also very recently, the cover time of dynamically evolving graphs was considered [AKL08]. It was found out that a lazy random walk could guarantee a polynomial cover time, while the standard random walk may take an exponential time on certain classes of evolving graphs.

Mixing Time. The best bounds on the mixing time in terms of the spectral gap are due to Sinclair [Sin92]. Also Sinclair [Sin93] introduced the so-called *conductance* and related this measure to the spectral gap. By means of this relationship, he devised the first polynomial approximation scheme for computing the permanent of a matrix [Sin93].

The basic idea behind a reduction of the cover time to the mixing time is as follows [Ald83, Dia88]. Assume for simplicity that G is a regular graph and divide the random walk into subsequent periods of length mixing time. Then the first vertices of each period are nearly independent and uniformly chosen among V. Therefore, the problem reduces to the Coupon Collector's Problem (cf. Theorem 2.19) and we obtain a bound of $n \ln n$ times the mixing time. The proofs of [Ald83, CF05] rely on this simple idea, however, by a much more involved analysis they get *exact* bounds such as $(1 + o(1))n \ln n$ on certain graphs.

5.1.3 Our Results

We begin by showing that the broadcast time on every graph is upper bounded by the mixing time of a random walk and a factor of Δ/δ. The rest and the main body of this chapter is devoted to the first comprehensive comparison of the cover time with the runtime of randomized broadcast. All results at a glance can be found in Figure 5.1. Most of these results provide strong evidence for the intuition already formulated in the seminal work of Chandra et al. [CRR+97], saying that "the cover time of the graph is an appropriate metric for the performance of certain kinds of randomized broadcast algorithms". Note that at a first look these processes seem not to be closely related. In fact it was pointed out by Feige et al. [FPRU90] that randomized broadcast is a parallel process where propagation occurs at *every* informed vertex, while the random walk moves only from *one* vertex to another.

We start by proving that the cover time of any graph $G = (V, E)$ is at most $\mathcal{O}(\frac{|E|}{\delta} \log n)$ times the broadcast time. Along with known bounds relating the two parameters to spectral and geometric properties of G, we obtain as a by-product a multitude of several inequalities which might be of independent interest. As one example, it was asked in [CRR+97] whether one can derive an upper bound on the cover time whose dependence on the vertex-expansion is less than *quadratic*. We obtain the first upper bound on the cover time which depends *linearly* on the edge-expansion at the cost of an additional logarithmic factor. Note that the bound of [BK89] based on the spectral gap implies only a bound with a quadratic dependence on the edge-expansion.

As the afore-mentioned upper bound on the cover time of $\mathcal{O}(\frac{|E|}{\delta} \log n)$ times the broadcast time is always $\Omega(n \log^2 n)$, we focus on more restricted graph classes to get down to the optimal value $\mathcal{O}(n \log n)$. We state two upper bounds on the cover time which are shown to be tight for some important graph classes. More importantly, we show that both upper bounds improve on previous bounds based on the spectral gap [BK89] or even the

Bound	Comment	Reference
$\mathsf{C}(u,v) \leqslant 4\frac{\lvert E\rvert}{\delta} \cdot \mathbf{E}\,[\,\mathsf{PAR}(u,v)\,]$	tight for paths/cycles and complete k-ary trees with $k = o(1)$	Thm. 5.19
$\mathsf{COV} = \mathcal{O}(\frac{\lvert E\rvert \log n}{\delta} \max_{u,v} \mathbf{E}\,[\,\mathsf{PAR}(u,v)\,])$	tight for k-ary trees, $k = o(1)$	Thm. 5.19
$\mathsf{COV} = \mathcal{O}(\frac{\lvert E\rvert}{\delta} \cdot \frac{\Delta}{\delta} \cdot \log^3 n \cdot \Phi^{-1})$	first upper bound with *linear* dependence on expansion/conductance	Cor. 5.21
$\mathsf{COV} = \mathcal{O}(n \cdot (\mathsf{PAR}_{n^{-1}} + \mathsf{MIX}_{e^{-1}}))$ (Δ-regular graphs with $\Delta \geqslant 5 \ln n$)	improves on bounds of [BK89] and [CRR+97] for Hamming graphs	Cor. 5.25
$\max_{u,v} \mathsf{C}(u,v) = \mathcal{O}(n \cdot \sqrt{\mathsf{MIX}_{e^{-1}}} \cdot \log n)$ (Cayley graphs)	tight for paths and cycles (up to $\log n$)	Thm. 5.37
$\max_{u,v} \mathsf{C}(u,v) = \mathcal{O}\big(n + \frac{n \cdot \mathsf{MIX}_{e^{-1}}^{2/3} \cdot \log n}{\Delta^{1/3}}\big)$ (Cayley graphs)	tight for dense Hamming graphs	Thm. 5.37
$\mathsf{C}(u,v) \geqslant 2 \cdot \mathrm{dist}(u,v)^2$	extends lower bound of [Zuc92] from trees to general graphs	Cor. 5.41
$\mathsf{COV} = \Omega(\frac{\sqrt{n \log n}}{\Delta} \cdot \mathsf{PAR}_{n^{-1}})$	matched by two-dimensional torus graph up to $\sqrt{\log n}$-factor	Prop. 5.42
$\mathsf{COV} = \Omega(\frac{\sqrt{n}}{\sqrt{\Delta} \log^2 n} \cdot \mathsf{PAR}_{n^{-1}})$ ($n^{\Omega(1)}$-regular graphs)	establishes polynomial gap for Δ-regular graphs whenever $\Delta = o(n)$	Thm. 5.43
$\mathsf{COV} = \Omega(\frac{\Delta^2}{n} \cdot \frac{1}{\log n} \cdot \mathbf{E}\,[\,\mathsf{PAR}\,])$ (regular graphs)	shows together with Thm. 5.19 that on dense regular graphs cover time and broadcast time differ by $\approx n$	Cor. 5.50
$\mathsf{COV} = \Omega(\mathrm{diam} \cdot \Delta \cdot \log n)$ (regular graphs, $\Delta \geqslant n^{1/2}$)	tight for $n^{1/2} \leqslant \Delta \leqslant n$ (cf. next result)	Prop. 5.63
$\mathsf{COV} = \mathcal{O}((n + \frac{n^2}{\Delta^2}) \cdot \log n)$ (Harary graphs)	establishes tightness of Prop. 5.63 for $\Delta \geqslant \sqrt{n}$, as $\mathrm{diam} = \Theta(\frac{n}{\Delta})$	Prop. 5.55
$\mathsf{COV} = \Omega(\mathrm{diam} \cdot \sqrt{n \log n})$ (regular graphs, $\Delta \leqslant n^{1/2}$)	$\log^2 n$-tight for $4 \leqslant \Delta \leqslant n^{1/2}$ (cf. next result)	Prop. 5.63
$\mathsf{COV} = \mathcal{O}(\mathrm{diam} \cdot \sqrt{n} \cdot \log^2 n)$ (Graph of Def. 5.57)	establishes $\log^{3/2} n$-tightness of Prop. 5.63 for $\Delta \leqslant \sqrt{n}$	Thm. 5.61

Fig. 5.1: *Summary of all derived bounds on the cover time* $\mathsf{COV} = \mathsf{COV}(G)$. $\mathsf{C}(u,v)$ *denotes the commute time between* u *and* v; PAR *refers to runtime of the parallel push algorithm (cf. Section 3.2).*

bound based on the vertex-expansion [CRR+97] for these graphs.

Conversely, we address the problem of lower bounding the cover time by the broadcast time. As the cover time and broadcast time may coincide on non-regular graphs, we aim at proving lower bounds only for regular graphs. Nevertheless, many of our results give also reasonable bounds for graphs where Δ/δ is not too large.

By showing that the commute time between two vertices is bounded below by the square of their distance, we conclude that for bounded-degree graphs the cover time is at least $\Omega(\sqrt{n \log n})$ times the broadcast time. This bound is easily seen to be tight up to a factor of $(\log n)^{3/2}$ by considering a two-dimensional torus graph.

On the other end of the scale, we consider dense graphs, i. e., regular graphs with a degree of $\Omega(n)$ and establish a much larger gap. For every such graph we prove that the cover time is at least n times the broadcast time, neglecting logarithmic factors. We complement this bound by the construction of Δ-regular graphs, $\Delta \geqslant \sqrt{n}$, such that the cover time is at most $\mathcal{O}(\Delta \cdot \log n)$ times the broadcast time. Combining this with the upper bound on the cover time of the second paragraph of this subsection, we have shown that the cover time is captured by the runtime of the push algorithm up to logarithmic factors on all Δ-regular graphs if and only if $\Delta = \Omega(n)$.

For graphs with some polynomial, but sublinear degree, our lower bounds are weaker. More precisely, there is a polynomial gap between the bounds and the examples we were able to find. Nevertheless, we can to conclude that the cover time is always a factor of $n^{1/5}$ larger than the broadcast time on any regular graph. We remark that a figure illustrating most of our results for regular graphs can be found in Figure 5.2.

We believe that our findings are also of interest due to the variety of applied proof techniques. These include couplings, ideas from group-theory and the use of certain flows from the theory of electrical networks. To the best of our knowledge, couplings have not been used for bounding the cover time before. Furthermore, Feige et al. [FPRU90] mentioned certain difficulties in applying methods from electrical network theory for their study of the push algorithm.

5.1.4 Road Map

We give the basic notation and some preliminary results in Section 5.2. In Section 5.3 we relate the runtime of the push algorithm to the mixing time (and associated parameters) of random walks. In Section 5.4 we derive upper bounds on the cover time by means of the runtime of the push algorithm. Section 5.5 comprises our lower bounds on the cover time. Finally, we conclude by summarizing our results and pointing at some directions for further research in Section 5.6.

5.2 Notations, Definitions and Preliminaries

Random Walk. A *random walk* on a given undirected, unweighted, simple and connected graph $G = (V, E)$ starts at some specified vertex $s \in V$ and moves in each step along some

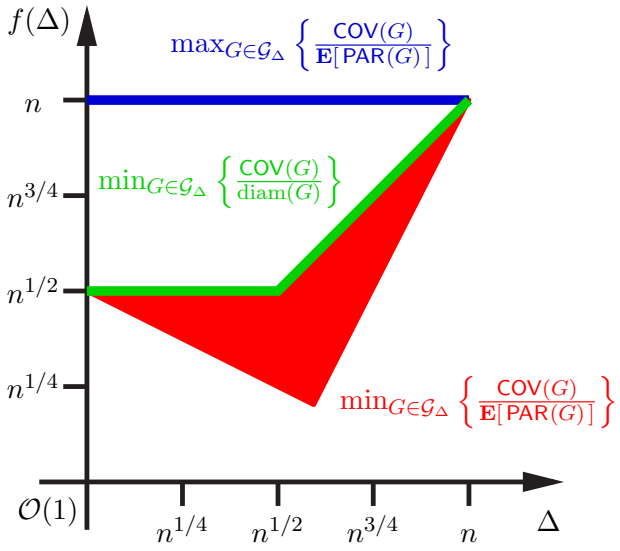

Fig. 5.2: *All bounds relating the cover time to the broadcast time (or diameter) on Δ-regular graphs at a glance. $\mathcal{G}_\Delta$ denotes the class of Δ-regular graphs. For instance, the red polygon indicates the gap between our lower bounds on the cover time and the examples which we were able to find for different values of Δ. For simplicity, polylogarithmic factors have been ignored.*

adjacent edge chosen uniformly at random. This can be described by a *transition matrix* $\mathbf{P}$, where $p_{ij} = 1/\deg(i)$ if $\{i,j\} \in E(G)$, and $p_{ij} = 0$ otherwise. Then, the random walk is an infinite sequence of vertices $X_0, X_1, \ldots$, where $X_0 := s$ is the starting point of this random walk, and X_t denotes the vertex visited by the random walk at step t. Note that X_t is in fact a random variable with a distribution $\mathbf{p}_s(t)$ on $V(G)$. Denoting by $\mathbf{p}_s(0)$ the unit-vector (regarded as column vector) with 1 at the component corresponding to s and 0 otherwise, we obtain the iteration

$$\mathbf{p}_s(t+1) = \mathbf{p}_s(t) \cdot \mathbf{P}$$

for every step $t \in \mathbb{N}$. It is well-known that on non-bipartite graphs, $\mathbf{p}_s(t)$ converges for $t \to \infty$ towards the *stationary distribution* vector π satisfying $\pi(v) = \deg(v)/(2|E|)$. For simplicity, we confine ourselves to non-bipartite graphs in the following. This causes no loss of generality as for bipartite graphs, one can consider the modified transition matrix $\frac{1}{2}\mathbf{I} + \frac{1}{2}\mathbf{P}$ (with $\mathbf{I}$ being the identity matrix) instead of $\mathbf{P}$. This change of the transition matrix slows down the mixing time and the cover time only by some constant factor [Lov93]. In addition, the broadcast time increases by at most a constant factor (cf. Theorem 3.4). Therefore, all coming inequalities would remain the same (modulo the multiplication of some constant factor).

Mixing Time, Spectral Gap and Conductance. We give some basic definitions and auxiliary results.

Definition 5.1. *The* mixing time *of a random walk on G is defined as*

$$\mathsf{MIX}_\varepsilon(G) := \max_{s\in V} \min\{t \in \mathbb{N} : \ \|\mathbf{p}_s(t) - \pi\|_1 \leqslant \varepsilon, X_0 = s\}.$$

The so-called separation time *of a random walk is defined as*

$$\mathsf{SEP}_{e^{-1}}(G) := \min\{t \in \mathbb{N} : \ \mathbf{Pr}\,[\, X_t = u \mid X_0 = s\,] \geqslant (1 - e^{-1}) \cdot \pi(u) \ \ \forall u, s \in V\}.$$

Lemma 5.2 ([AF02]). *For any graph $G = (V, E)$, $\mathsf{SEP}_{e^{-1}}(G) = \Theta(\mathsf{MIX}_{e^{-1}}(G))$.*

Definition 5.3. *The* conductance *of a random walk with transition matrix $\mathbf{P}$ is defined as*

$$\Phi = \min_{X\subseteq V, \pi(X)>0} \frac{\sum_{x\in X, y\in X^c} \pi(x)\mathbf{P}_{x,y}}{\min\{\pi(X), \pi(X^c)\}},$$

which is, for our choice of $\mathbf{P}$, exactly the same as the edge expansion of G (cf. Definition 2.32).

Since G is connected and non-bipartite, and $\mathbf{P}$ is a stochastic matrix, it is known that the eigenvalues of $\mathbf{P}$ are $\lambda_1 = 1 > \lambda_2 \geqslant \cdots \geqslant \lambda_n > -1$.

Theorem 5.4 ([Sin93]). *The second largest eigenvalue λ_2 of $\mathbf{P}$ satisfies*

$$1 - 2\Phi \leqslant \lambda_2 \leqslant 1 - \frac{\Phi^2}{2}.$$

Theorem 5.5 ([Sin92]). *Consider a random walk on G with transition matrix $\mathbf{P}$. Let $\lambda_{\max} := \max\{\lambda_2, |\lambda_n|\}$. Then*

$$\frac{\lambda_{\max}}{2\cdot(1-\lambda_{\max})} \cdot \ln(2\varepsilon)^{-1} \leqslant \mathsf{MIX}_\varepsilon(G) \leqslant \max_{x\in V}\left\{\frac{1}{1-\lambda_{\max}} \cdot \left(\ln(\pi(x))^{-1} + \ln \varepsilon^{-1}\right)\right\}.$$

Commute Time, Resistance and Cover Time. For two vertices $u, v \in V(G)$ let $\mathsf{H}(u, v) := \mathbf{E}\,[\min\{t \in \mathbb{N}\backslash\{0\} : X_t = v, X_0 = u\}\,]$ be the *hitting time* (also known as *first passage time*) from u to v, i. e., the expected number of steps to reach v from u. It is known that for any $u \in V$, $\mathsf{H}(u, u) = (1/\pi(u)) = 2|E|/\deg(u)$. The *commute time* $\mathsf{C}(u, v)$ is defined as the expected number of steps to reach v when starting from u and then returning back to u, thus $\mathsf{C}(u, v) := \mathsf{H}(u, v) + \mathsf{H}(v, u)$ (notice that even on regular graphs, $\mathsf{H}(u, v) = \mathsf{H}(v, u)$ may not be true [Lov93]).

We only sketch the definition of the *effective resistance* $\mathsf{R}(u, v)$ between two vertices u and v; the justification is given in the theorem below. Consider the graph G as an electrical network where each edge represents a unit resistance. Assume that one ampere were injected into vertex u and removed from vertex v in the network. Then $\mathsf{R}(u, v)$ is the voltage difference between u and v. For more details on electrical networks the reader is referred to, e.g., [Lov93, MR95, CRR+97]

Theorem 5.6 ([CRR+97]). *For any pair of vertices $u, v \in V$, $\mathsf{C}(u,v) = 2|E| \cdot \mathsf{R}(u,v)$.*

We will mainly be concerned with the so-called *cover time*, which is the expected number of steps a random walk takes to visit all vertices of G. Denote by $\mathsf{COV}(s)$ this time for a random walk which starts from s. Note that if we can prove that a random walk visits all vertices within x steps with some probability $p > 0$, then the bound $\mathsf{COV}(s) \leqslant x/p$ follows by the expectation of the geometric distribution [GS01]. Usually, we are interested in $\mathsf{COV}(G) := \max_{s \in V} \mathsf{COV}(s)$. The following well-known result relates the maximum commute time to the cover time.

Theorem 5.7 ([MR95, CRR+97]). *For any graph $G = (V, E)$,*

$$\frac{1}{2} \cdot \max_{u,v \in V} \mathsf{C}(u,v) \leqslant \mathsf{COV}(G) \leqslant e^3 \cdot \max_{u,v \in V} \mathsf{C}(u,v) \ln n + n$$

Consequently, also $\max_{u,v \in V} \mathsf{R}(u,v)$ captures the cover time up to a factor of $\mathcal{O}(\log n)$. We will also frequently use the following lower bound of Feige [Fei95a].

Theorem 5.8 ([Fei95a]). *For any graph $G = (V, E)$, $\mathsf{COV}(G) \geqslant (1 + o(1)) \cdot n \ln n$.*

5.3 Conductance, Spectral Gap and Mixing Time of Random Walks

We use the inequality of Proposition 3.12 to get upper bounds on the broadcast time by means of one of the three measures mentioned in the title of this section.

Corollary 5.9. *Let $G = (V, E)$ be any graph. Then*

$$\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}\left(\frac{\Delta}{\delta} \cdot \Phi^{-1} \cdot \log n\right)$$

and

$$\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}\left(\frac{\Delta}{\delta} \cdot \frac{1}{1 - \lambda_2} \cdot \log n\right),$$

where λ_2 is the second largest eigenvalue of $\mathbf{P}$.

Proof. In this proof we consider the sequential push algorithm SEQ. Clearly for any $1 \leqslant m \leqslant n - 1$,

$$\begin{aligned}
\Lambda(m) &= \min_{X \subseteq V, |X| = m} \left\{ \sum_{v \in X} \frac{\deg_{X^c}(v)}{\deg(v)} \right\} \geqslant \min_{X \subseteq V, |X| = m} \left\{ \frac{|E(X, X^c)|}{\Delta} \right\} \\
&= \min\{m, n-m\} \cdot \frac{\delta}{\Delta} \cdot \min_{X \subseteq V, |X| = m} \left\{ \frac{|E(X, X^c)|}{\min\{m \cdot \delta, (n-m) \cdot \delta\}} \right\} \\
&\geqslant \min\{m, n-m\} \cdot \frac{\delta}{\Delta} \cdot \min_{X \subseteq V, |X| = m} \left\{ \frac{|E(X, X^c)|}{\min\{\mathrm{vol}(X), \mathrm{vol}(X^c)\}} \right\} \\
&\geqslant \min\{m, n-m\} \cdot \frac{\delta}{\Delta} \cdot \Phi.
\end{aligned}$$

By Proposition 3.12 we have

$$\mathbf{E}[\,\mathsf{SEQ}(G)\,] \leqslant \sum_{m=1}^{n-1} \frac{1}{\Lambda(m)} \leqslant \sum_{m=1}^{n-1} \frac{1}{\min\{m, n-m\} \cdot \frac{\delta}{\Delta} \cdot \Phi} \leqslant \frac{\Delta}{\delta} \cdot \Phi^{-1} \cdot 2 \ln n.$$

To bound $\mathsf{SEQ}_{n^{-1}}(G)$, consider $X := \sum_{m=1}^{n-1} X_m$, where X_m is the number of steps required to increase the number of informed vertices from m to $m+1$. By Lemma 2.23, we may assume that each X_m is a geometric variable with mean $(\min\{m, n-m\}\frac{\delta}{\Delta}\Phi)^{-1}$. Hence $\mathbf{E}[\,X_m\,] \leqslant \frac{\Delta}{\delta}\Phi^{-1}$ and thus Corollary 2.26 gives $\mathsf{SEQ}_{n^{-1}}(G) = \mathcal{O}\left(\frac{\Delta}{\delta} \cdot \Phi^{-1} \cdot \log n\right)$. The second claim follows directly from $1-2\Phi \leqslant \lambda_2 \Leftrightarrow \Phi^{-1} \leqslant 2/(1-\lambda_2)$ due to Theorem 5.4. □

Corollary 5.10. *For any graph $G = (V, E)$,*

$$\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}\left(\frac{\Delta}{\delta} \cdot (\mathsf{MIX}_{n^{-1}}(G) + \log n)\right).$$

Proof. Recall that $\lambda_{\max} = \max\{\lambda_2, |\lambda_n|\}$. Since we assume that G is not bipartite, $\lambda_{\max} < 1$ [Lov93]. Hence we get by Corollary 5.9 and Theorem 5.5 that

$$\begin{aligned}
\mathsf{PAR}_{n^{-1}}(G) &= \mathcal{O}\left(\frac{\Delta}{\delta} \cdot \frac{1}{1-\lambda_2} \cdot \log n\right) = \mathcal{O}\left(\frac{\Delta}{\delta} \cdot \frac{1}{1-\lambda_{\max}} \cdot \log n\right) \\
&= \mathcal{O}\left(\frac{\Delta}{\delta} \cdot \left(\frac{\lambda_{\max}}{1-\lambda_{\max}} \cdot \log n + \log n\right)\right) \\
&= \mathcal{O}\left(\frac{\Delta}{\delta} \cdot (\mathsf{MIX}_{n^{-1}}(G) + \log n)\right).
\end{aligned}$$

□

To see that the factor $\frac{\Delta}{\delta}$ cannot be omitted in general, consider the following graph G. Take the graph $\mathsf{K}_{1,n-1}$ and add some perfect matching to the vertices with degree 1 (this is just to make the graph non-bipartite). For this graph one can verify that $\mathsf{MIX}_{n^{-1}}(G) = \mathcal{O}(\log n)$ but $\mathsf{PAR}_{n^{-1}}(G) = \Omega(n \log n)$.

An example where the mixing time is much larger than the broadcast time is given after Theorem 5.20 in the next section.

5.4 Upper Bounds on the Cover Time

We will frequently compare our upper bounds with the following two known bounds.

Theorem 5.11 (Spectral Gap Bound, [BK89]). *Let $G = (V, E)$ be any graph. Assume that the vertices $1, 2, \ldots, n$ are ordered such that $\pi(1) \leqslant \pi(2) \leqslant \ldots \leqslant \pi(n)$. Then*

$$\mathsf{COV}(G) \leqslant \frac{1}{1-\lambda_2} \cdot \left[(2+\varepsilon)\ln n - n \ln \pi(1) + \sum_{1 \leqslant i \leqslant n} \left(\sum_{1 \leqslant j \leqslant i} \pi(j)\right)^{-1}\right] \cdot (1 + o(1)),$$

where $\varepsilon > 0$ is an arbitrary constant. Moreover, if G is a regular graph, we have

$$\mathsf{COV}(G) \leqslant \frac{(4+\varepsilon) \cdot n \ln n}{1-\lambda_2} \cdot (1 + o(1)),$$

where $\varepsilon > 0$ is an arbitrary constant.

Theorem 5.12 (Vertex-Expansion Bound, [CRR+97]). *Let G be a graph with vertex-expansion α. Then*

$$\mathsf{COV}(G) = \mathcal{O}\left(\frac{|E|}{\alpha^2 \cdot \delta} \cdot \log n\right).$$

Essentially, Theorem 5.11 requires a regular graph to be an edge-expander to become $\mathcal{O}(n \log n)$ while for Theorem 5.12 a vertex-expander suffices. Nevertheless, the dependence on the vertex-expansion in Theorem 5.12 is quadratic and the corresponding dependence of $1 - \lambda_2$ on the edge-expansion may not necessarily be quadratic (cmp. Theorem 5.4).

5.4.1 An Upper Bound for General Graphs

In order to establish an upper bound on the cover time for general graphs, we first prove a general inequality between *first-passage-percolation* times and broadcast times and apply then a result of Lyons et al. [LPP99] relating first-passage-percolation to the cover time.

Definition 5.13 (Undirected first-passage-percolation (cf. [LPP99, FP93])). *The undirected first-passage-percolation model* UFPP *is defined as follows. Each (undirected) edge $e \in E(G)$ is assigned some weight $w(e)$ which is an independent exponential variable with parameter 1 (and mean 1). Specify some vertex s. Then the* first-passage-percolation time *from s to v is defined by*

$$\mathsf{UFPP}(s, v) := \inf_{\mathcal{P}=(s,\ldots,v)} \sum_{e \in \mathcal{P}} w(e),$$

where the inf *is over all possible paths from s to v in G. Note that $\mathsf{UFPP}(s, s) = 0$.*

Theorem 5.14. *For any graph $G = (V, E)$ and two vertices s, v we have*

$$\mathbf{E}\left[\,\mathsf{UFPP}(s, v)\,\right] \leqslant \frac{1}{\delta} \cdot \mathbf{E}\left[\,\mathsf{PAR}(s, v)\,\right].$$

Proof. The proof will be divided into several (in-)equalities between different models. First we introduce a directed version of UFPP, denoted by DFPP. In this model each undirected edge $\{u, v\}$ is replaced by two directed edges (u, v) and (v, u), and each directed edge e is assigned a weight $w(e)$, which is an exponential variable with mean 1. Denote by $\mathsf{DFPP}(s, v)$ the corresponding first-passage-percolation time of this directed version. We prove:

Lemma 5.15. *For every pair of vertices s, v,*

$$\mathsf{UFPP}(s,v) \preceq 2 \cdot \mathsf{DFPP}(s,v).$$

Proof. Consider the modification $\mathsf{UFPP}_2(s,v)$ of the undirected first-passage-percolation model $\mathsf{UFPP}(s,v)$ where all edges are independent exponential random variables with parameter 2. By Observation 2.4,

$$\mathsf{UFPP}(s,v) \stackrel{D}{=} 2 \cdot \mathsf{UFPP}_2(s,v).$$

Consider now the directed model DFPP and a pair of edges (u,v) and (v,u). By definition, $w(u,v)$ and $w(v,u)$ are independent exponential variables with parameter 1. Hence by Lemma 2.3, $w(\{u,v\}) := \min\{w(u,v), w(v,u)\}$ is an exponential variable with parameter 2. Clearly, the random variables $\cup_{\{u,v\}\in E} w(\{u,v\})$ are independent. Therefore

$$\begin{aligned}\mathsf{DFPP}(s,v) &\stackrel{D}{=} \inf_{\mathcal{P}=(s,\ldots,v)} \sum_{e=(u,v)\in P} w((u,v)) \\ &\succeq \inf_{\mathcal{P}=(s,\ldots,v)} \sum_{e=(u,v)\in P} w(\{u,v\}) \stackrel{D}{=} \mathsf{UFPP}_2(s,v) \stackrel{D}{=} \frac{1}{2} \cdot \mathsf{UFPP}(s,v).\end{aligned}$$

□

Next consider another broadcast model denoted by $\overline{\mathsf{SEQ}}$. At the beginning, a vertex s knows a rumor which should be spread to all other vertices. Once a vertex v receives the rumor at step t, it sends the rumor at each step $t + X_{1,v}$, $t + X_{1,v} + X_{2,v}, \ldots$ to some randomly chosen neighbor, where the $X_{i,v}, i \geqslant 1$, are independent exponential variables with parameter $\deg(v)$. Let $\overline{\mathsf{SEQ}}(s,v)$ be the first time when v is informed.

Lemma 5.16. *For every pair of two vertices s, v, the random variables $\overline{\mathsf{SEQ}}(s,v)$ and $\mathsf{DFPP}(s,v)$ have the same distribution.*

Proof. Fix some arbitrary vertex $u \in G$ and let $\overline{\mathsf{SEQ}}(s,u)$ be the time until u becomes informed. We shall be interested in the time until u transmits the rumor to some neighbor $u' \in N(u)$. Recall that the sequence $X_{1,u}, X_{2,u}, \ldots$ are independent exponential variables with parameter $\deg(u)$. Consider

$$X(u,u') := \min_{t\in\mathbb{N}} \left\{ \sum_{i=1}^{t} X_{i,u} \;\middle|\; N_{t,u} = u' \right\}.$$

At each step $\overline{\mathsf{SEQ}}(s,u) + X_{1,u}, \overline{\mathsf{SEQ}}(s,u) + X_{1,u} + X_{2,u}, \ldots$, u transmits the rumor to some uniformly chosen neighbor. Therefore Lemma 2.5 implies that $X(u,u')$ is an exponential random variable with mean $\deg(u)/\deg(u) = 1$. It follows that for every directed edge $(u,u') \in E(G)$ the time until $\overline{\mathsf{SEQ}}$ forwards the rumor along this edge is $\overline{\mathsf{SEQ}}(s,u)$ plus an exponential variable with mean $\deg(u)$. For every $u, u' \in V(G)$, all these increments to

$\overline{\mathsf{SEQ}}(s,u)$ are independent random variables. Hence, $\overline{\mathsf{SEQ}}(s,v)$ and $\mathsf{DFPP}(s,v)$ have the same distribution. □

Finally, our aim is to relate $\overline{\mathsf{SEQ}}$ and PAR. In the proof we will use the concept of minimal path from Observation 3.2, similarly to the proof of Theorem 3.5.

Lemma 5.17. *For any pair of vertices $s, v \in V$ we have*

$$\mathbf{E}\left[\,\overline{\mathsf{SEQ}}(s,v)\,\right] \leqslant \frac{\mathbf{E}\left[\,\mathsf{PAR}(s,v)\,\right]}{\delta}.$$

Proof. We first recall the following description of an instance of PAR from Section 3.3. A fixed instance of PAR is described by $(N_{t,u})_{t\in\mathbb{N},u\in V}$, where $N_{t,u} \in N(u)$ is the neighbor of u chosen in step $\mathsf{PAR}(s,u)+t$ and $\mathsf{PAR}(s,u)$ is the time-step at which u becomes informed. Recall that Ω_1 is the set of all instances, i. e., the set of all $(N_{t,u})_{t\in\mathbb{N},u\in V}$.

In order to describe an instance of $\overline{\mathsf{SEQ}}$, let Ω_1 be as before and let Ω_2 be the probability space which specifies the time-steps at which a vertex sends the rumor to some neighbor. Thus Ω_2 can be described by $(X_{t,u})_{t\in\mathbb{N},u\in V}$, where the $X_{t,u}$ are independent exponential variables, each of which has parameter $\deg(u)$ (mean $1/\deg(u)$). Then given $\omega_1 \in \Omega_1$ and $\omega_2 \in \Omega_2$, a vertex $u \in V$ sends the rumor at time-step $\mathsf{SEQ}(s,u) + \sum_{k=1}^{t} X_{k,u}$ to neighbor $N_{t,u}$ for any $t \geqslant 1$.

We consider a coupling between $\overline{\mathsf{SEQ}}$ and PAR where the same ω_1 occurs, but $\omega_2 \in \Omega_2$ is chosen uniformly at random, i. e., each $X_{k,u}$ is an independent exponential variable with parameter $\deg(u)$. Let $\mathcal{P}_{\min}(s,v)$ be a minimal path from s to v in some instance $\omega_1 \in \Omega_1$ of PAR. Note that in this case the expected time (w. r. t. Ω_2) to traverse $\mathcal{P}_{\min}(s,v)$ can be bounded by the sum of $D(\mathcal{P}_{\min}(s,v))$ independent exponential variables with parameter at least δ (mean at most $1/\delta$). Hence the expected time in $\overline{\mathsf{SEQ}}$ for traversing $\mathcal{P}_{\min}(s,v)$ is at most $D(\mathcal{P}_{\min}(s,v))/\delta$. For some integer k, let $\mathcal{A}_k$ be the event that $k = D(\mathcal{P}_{\min}(s,v))$ for some minimal path from s to v in the instance $\omega_1 \in \Omega_1$. By using conditional expectations we obtain

$$\mathbf{E}\left[\,\overline{\mathsf{SEQ}}(s,v)\,\right] = \sum_{k=1}^{\infty} \mathbf{E}\left[\,\overline{\mathsf{SEQ}}(s,v) \mid \mathcal{A}_k\,\right] \cdot \mathbf{Pr}\left[\,\mathcal{A}_k\,\right] \leqslant \sum_{k=1}^{\infty} \frac{k}{\delta} \cdot \mathbf{Pr}\left[\,\mathcal{A}_k\,\right] = \frac{\mathbf{E}\left[\,\mathsf{PAR}(s,v)\,\right]}{\delta},$$

and the lemma follows. □

We are now ready to finish the proof of Theorem 5.14. For every pair of vertices $s, v \in V$,

$$\mathbf{E}\left[\,\mathsf{UFPP}(s,v)\,\right] \leqslant 2 \cdot \mathbf{E}\left[\,\mathsf{DFPP}(s,v)\,\right] = 2 \cdot \mathbf{E}\left[\,\overline{\mathsf{SEQ}}(s,v)\,\right] \leqslant 2 \cdot \frac{\mathbf{E}\left[\,\mathsf{PAR}(s,v)\,\right]}{\delta}.$$

□

Theorem 5.18 ([LPP99]). *Let $u, v \in V(G)$ with $u \neq v$. Then, $\mathsf{R}(u,v) \leqslant \mathbf{E}\left[\,\mathsf{UFPP}(u,v)\,\right]$.*

Combining the prior two theorems we arrive at the main result of this section.

Theorem 5.19. *For any graph $G = (V, E)$ we have for every pair of vertices $u \neq v$,*

$$\mathsf{C}(u,v) \leqslant 4 \cdot \frac{|E|}{\delta} \cdot \mathbf{E}\left[\,\mathsf{PAR}(u,v)\,\right],$$

and hence

$$\mathsf{COV}(G) = \mathcal{O}\left(\frac{|E|}{\delta} \cdot \log n \cdot \max_{u,v \in V} \mathbf{E}\left[\,\mathsf{PAR}(u,v)\,\right]\right).$$

Proof. Using the identity $\mathsf{C}(u,v) = 2|E| \cdot \mathsf{R}(u,v)$ (Theorem 5.6) we have for any $u \neq v$

$$\begin{aligned}\mathsf{C}(u,v) = 2|E| \cdot \mathsf{R}(u,v) &\leqslant 2|E| \cdot \mathbf{E}\left[\,\mathsf{UFPP}(u,v)\,\right] && \text{(by Theorem 5.18)}\\ &\leqslant 4|E| \cdot \frac{\mathbf{E}\left[\,\mathsf{PAR}(u,v)\,\right]}{\delta} && \text{(by Theorem 5.14),}\end{aligned}$$

and the first claim follows. If $u = v$ it is known that $\mathsf{C}(u,u) = 4|E|/(\deg(u)) \leqslant 4|E|/\delta$ and since $\max_{u,v} \mathbf{E}\left[\,\mathsf{PAR}(u,v)\,\right] \geqslant 1$ (unless $|V| = 1$ for which the claim is trivial), the second claim follows immediately from Theorem 5.7. □

Note that the first inequality is matched by paths, cycles ($\max_{u,v} \mathsf{C}(u,v) = \Theta(n^2)$, cf. [Lov93]) and complete k-ary trees, where $k = \mathcal{O}(1)$ ($\max_{u,v} \mathsf{C}(u,v) = \Theta(n \log n)$ [Zuc92, Cor. 9]). The first inequality may be even tight for some highly non-regular graphs like lollipop graphs (a complete graph with $2n/3$ vertices attached by a path of length $n/3$) which have a cubic maximum commute time [Lov93, Fei95b], but a linear broadcast time.

Complete k-ary trees with $k = \mathcal{O}(1)$ provide an example where also the second inequality is tight up to a constant factor: the cover time is known to be $\Theta(n \log^2 n)$ [Zuc92, Cor. 9] and furthermore, $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\operatorname{diam}(G) + \log n)$ for any bounded-degree graph [FPRU90].

Furthermore, Theorem 5.19 is matched up to logarithmic factors by graphs of an *arbitrary* density, because every regular graph with $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\log n)$ matches Theorem 5.19 up to a logarithmic factor. This in sharp contrast to the results we derive in Section 5.5.

In the remainder of this subsection, we give some graph-theoretical inequalities which can be derived from Theorem 5.19 and the following results. It was shown by Zuckerman [Zuc92] that $\mathsf{COV}(G) \geqslant \sum_{k=2}^{n} (1-\lambda_k)^{-1}$. Combining this lower bound with Theorem 5.19, Proposition 3.12 and the inequality

$$\mathbf{E}\left[\,\mathsf{PAR}(G)\,\right] = \mathcal{O}(\mathsf{PAR}_{n^{-1}}(G)) = \mathcal{O}(\mathsf{SEQ}_{n^{-1}}(G)) = \mathcal{O}(\mathbf{E}\left[\,\mathsf{PAR}(G)\,\right] \cdot \log n),$$

we obtain the following chain of inequalities.

Theorem 5.20. *For any graph G we have the following chain of inequalities (suppressing constant factors),*

$$\sum_{k=2}^{n} \frac{1}{1-\lambda_k} \leqslant \mathsf{COV}(G) \leqslant \frac{|E|}{\delta} \cdot \log n \cdot \mathbf{E}\left[\,\mathsf{PAR}(G)\,\right] \leqslant \frac{|E|}{\delta} \cdot \log^2 n \cdot \sum_{k=1}^{n-1} \frac{1}{\Lambda(k)}. \tag{5.1}$$

In particular, we have

$$\sum_{k=2}^{n} \frac{1}{1-\lambda_k} \leqslant \frac{|E|}{\delta} \cdot \log^2 n \cdot \sum_{k=1}^{n-1} \frac{1}{\Lambda(k)}. \tag{5.2}$$

We think that these inequalities are interesting from different points of view. First, 5.1 gives a combinatorial upper bound on the cover time based on some edge expansion-based measure. Conversely, we get a spectral lower bound on the broadcast time. Finally, in 5.2 we obtain a graph-theoretical inequality by relating all nontrivial eigenvalues of $\mathbf{P}$ to the edge expansion-based measure Λ. This inequality bears some resemblance to the inequality

$$\frac{2}{1-\lambda_2} \leqslant \frac{1}{\Phi^2} \tag{5.3}$$

implied by Theorem 5.4. However, the following example shows that all expressions involved in 5.1 can be substantially smaller than the expressions involved in 5.3.

Consider the Cayley graph $G = \mathsf{K}_{n/2} \times \mathsf{K}_2$. Clearly, G is far from being an edge-expander and Theorem 5.11 overestimates the cover time of G by a factor of almost n. By [Zuc92, Theorem 13], $\mathsf{COV}(G) = \Theta(\log n \cdot \sum_{k=2}^{n} \frac{1}{1-\lambda_k})$ for every Cayley graph. As a consequence, the left inequality of 5.1 is tight up to a logarithmic factor. Using Proposition 3.13, the right side of 5.1 is $\mathcal{O}(n \cdot \log^3 n)$. Putting everything together, the rightmost and leftmost side of 5.1 differ only by a factor of $\mathcal{O}(\log^3 n)$ for this graph.

This graph $G = \mathsf{K}_{n/2} \times \mathsf{K}_2$ provides also an example, where the mixing time is polynomial, but broadcast time and cover time are close to their optimal values $\mathcal{O}(\log n)$ and $\mathcal{O}(n \log n)$, respectively.

Before concluding this section, we obtain as a simple corollary the first upper bound on the cover time that depends *linearly* on the edge-expansion (however, at the cost of an additional factor of $\log n$ in comparison with Theorem 5.12).

Corollary 5.21. *For any graph $G = (V, E)$,*

$$\mathsf{COV}(G) = \mathcal{O}\left(\frac{|E|}{\delta} \cdot \frac{\Delta}{\delta} \cdot \log^3 n \cdot \Phi^{-1}(G)\right).$$

Proof. Follows immediately by substituting the first bound of Corollary 5.9 into the second one of Theorem 5.19. □

5.4.2 Upper Bounds depending on the Mixing Time

In this subsection we also derive upper bounds on the cover time. In contrast to the previous subsection, these bounds depend additionally on the mixing time.

Bounds for General Graphs.

Definition 5.22 ([BKRU94]). *For some integer k, consider k independent parallel random walks on G, each starting from a uniform random vertex. Let $\mathsf{COV}^k(G)$ be the expected*

time until all vertices have been visited by at least one of the k random walks.

Theorem 5.23 ([BKRU94]). *Let G be a regular graph and let $6\log n \leqslant k \leqslant n$. Then*

$$\mathsf{COV}^k(G) = \mathcal{O}\left(\frac{n^2}{k^2}\cdot \log^2 n\right).$$

Notice that for $k = o(\log n)$ the statement of the theorem would be useless, as it is known that $\mathsf{COV}(G) = \mathcal{O}(n^2)$ for every regular graph [Fei97b]. We prove the following.

Theorem 5.24. *For any graph $G = (V, E)$,*

$$\mathsf{COV}(G) = \mathcal{O}\big(n \cdot (\mathsf{COV}^n(G) + \mathsf{MIX}_{e^{-1}}(G))\big).$$

Proof. To simplify notation, define $\mathcal{M} := \mathsf{MIX}_{e^{-1}}(G)$. As a first step, we introduce another model of $4n$ parallel random walks covering a graph. In this model the starting positions of the $4n$ random walks denoted by $Y^1, Y^2, \ldots, Y^{4n}$ are determined as follows. Each random walk starts from some vertex in $v \in G$ with probability $(1 - e^{-1})\pi(v)$. With probability e^{-1}, the random walk *fails*, i. e., the random walk is removed from the network and is not able to visit any vertex. The cover time of this model is denoted by $\overline{\mathsf{COV}}^{4n}(G)$.

The key idea is to define a proper coupling between one (long) random walk X and the $4n$ (short) random walks $Y^1, Y^2, \ldots, Y^{4n}$. This coupling will have the property that if the $4n$ short random walks cover $V(G)$, then also the (coupled) random walk X covers $V(G)$.

Consider the random walk X starting from some arbitrary vertex s until step $n\cdot\alpha(\mathcal{M} + \mathsf{COV}^n(G))$, where α is a large enough constant. Let $X_0, X_1, \ldots$ be the sequence of vertices visited by the random walk at step $0, 1, \ldots$. Note that the distribution of $X_{t+\alpha\mathcal{M}}$ depends on X_t, but Lemma 5.2 implies that for any $v, v' \in V(G)$ and any time-step $t \geqslant 0$

$$\mathbf{Pr}\left[\, X_{t+\alpha\mathcal{M}} = v \mid X_t = v' \,\right] \geqslant (1 - e^{-1}) \cdot \pi(v).$$

From the above it follows that,

$$\mathbf{Pr}\left[\, Y_0^1 = v \,\right] = (1 - e^{-1}) \cdot \pi(v) \leqslant \mathbf{Pr}\left[\, X_{\alpha\mathcal{M}} = v \mid X_0 = v' \,\right],$$

for any vertices $v, v' \in V$. Hence, by Lemma 2.24 there is a coupling between $X_{\alpha\mathcal{M}}$ and Y_0^1 such that $Y_0^1 = v$ implies $X_{\alpha\mathcal{M}} = v$. Now if $Y_1^1 = v$ for some $v \in V$ occurs, then we extend the coupling by setting $Y_t^1 := X_{t+\alpha\mathcal{M}}$ for every step $0 \leqslant t \leqslant \alpha\mathsf{COV}^n(G) - 1$. Otherwise, Y_1^1 fails and the first random walk is removed from G without visiting any vertex.

More generally, assume that we have coupled the first $i \cdot (\alpha\mathcal{M} + \alpha\mathsf{COV}^n(G)) - 1$ steps of X with $Y^1, Y^2, \ldots, Y^i$. As before, we have

$$\mathbf{Pr}\left[\, Y_0^{i+1} = v' \,\right] = (1 - e^{-1})\pi(v) \;\leqslant\; \mathbf{Pr}\left[\, X_{i\cdot\alpha(\mathcal{M}+\mathsf{COV}^n(G))+\alpha\mathcal{M}} = v \mid X_{i\cdot\alpha(\mathcal{M}+\mathsf{COV}^n(G))} = v' \,\right],$$

and Lemma 2.24 implies the existence of a coupling between $X_{i\cdot\alpha(\mathcal{M}+\mathsf{COV}^n(G))+\alpha\mathcal{M}}$ and Y_1^i such that

$$Y_1^i = v \;\Rightarrow\; X_{i\cdot\alpha(\mathcal{M}+\mathsf{COV}^n(G))+\alpha\mathcal{M}} = v.$$

In case the i-th random walk Y^i does not fail, we extend the coupling as before by $Y^i_t := X_{i\cdot\alpha(\mathcal{M}+\mathsf{COV}^n(G))+\alpha\mathcal{M}+t}$ for $0 \leqslant t \leqslant \alpha\mathsf{COV}^n(G) - 1$. To summarize, we have defined a coupling between the random walk X and the short random walks $Y^1, Y^2, \dots, Y^{4n}$ with the property that

$$\bigcup_{\substack{1\leqslant i\leqslant 4n:\\ Y^i \text{ does not fail}}} \bigcup_{t=0}^{\alpha(\mathcal{M}+\mathsf{COV}^n(G))-1} Y^i_t \subseteq \bigcup_{t=0}^{4n\alpha(\mathcal{M}+\mathsf{COV}^n(G))} X_t,$$

that is, every vertex visited by one of the Y^i is also visited by X. In particular, if the $4n$ random walks $Y^1, Y^2, \dots, Y^{4n}$ cover the whole graph, then also X does. It follows from the Chernoff bound for the sum of Bernoulli variables (Theorem 2.11) that at least $2n$ of the random walks $Y^1, Y^2, \dots, Y^{4n}$ does not fail with probability $1 - n^{-1}$. For the remainder of the proof, we have to introduce some further notation. With a slight abuse of notation, let $\mathsf{COV}_s(G)(\omega)$ (and correspondingly, $\overline{\mathsf{COV}}^{4n}(G)(\omega)$ and $\mathsf{COV}^{2n}(G)(\omega)$) be the *random variable* representing the first time-step when all vertices have been visited. With this notation at hand,

$$\begin{aligned}
& \mathbf{Pr}\left[\, \mathsf{COV}_s(G)(\omega) \geqslant \alpha n \cdot (\mathsf{COV}^n(G) + \mathcal{M}) \,\right] \\
\leqslant\ & \mathbf{Pr}\left[\, \overline{\mathsf{COV}}^{4n}(G)(\omega) \geqslant \alpha \cdot \mathsf{COV}^n(G) \,\right] && \text{(by the coupling)} \\
\leqslant\ & \mathbf{Pr}\left[\, \mathsf{COV}^{2n}(G)(\omega) \geqslant \alpha \cdot \mathsf{COV}^n(G) \,\right] + n^{-1} && \text{(by the Chernoff bound)} \\
\leqslant\ & \frac{1}{\alpha} + n^{-1} && \text{(by Markov's inequality),}
\end{aligned}$$

and the claim follows. □

Our goal is now to relate $\mathsf{COV}^n(G)$ to $\mathsf{PAR}_{n^{-1}}(G)$.

Corollary 5.25. *Let $G = (V, E)$ be any Δ-regular graph with $\Delta \geqslant 5 \ln n$. Then*

$$\mathsf{COV}(G) = \mathcal{O}\big(n \cdot (\mathsf{PAR}_{n^{-1}}(G) + \mathsf{MIX}_{e^{-1}}(G))\big).$$

Proof. For the proof, we require the so-called *agent-based-broadcast-model* from [ELS07] defined as follows. There are n agents performing independent and synchronous random walks on G. If an agent visits an informed vertex, the agent becomes informed. Similarly, if a vertex is visited by an informed agent, the vertex becomes informed. At the beginning step 0, there is only one informed vertex, and the starting points of the n agents are chosen independently and uniformly at random from V. Let $\mathsf{ABR}(G)$ be the random variable being the first step at which all n vertices are informed. It is evident that $\mathsf{COV}^n(G) \leqslant \mathbf{E}\left[\, \mathsf{ABR}(G) \,\right]$. Moreover, in [ES08b] it was shown that for every graph G with $\Delta \geqslant 5 \ln n$,

$\mathsf{ABR}_{n^{-1}}(G) = \mathcal{O}(\mathsf{PAR}_{n^{-1}}(G))$. Combining these findings with Theorem 5.24 yields

$$\begin{aligned}\mathsf{COV}(G) &= \mathcal{O}\big(n \cdot (\mathsf{COV}^n(G) + \mathsf{MIX}_{e^{-1}}(G))\big)\\ &= \mathcal{O}\big(n \cdot (\mathbf{E}\,[\,\mathsf{ABR}(G)\,] + \mathsf{MIX}_{e^{-1}}(G))\big)\\ &= \mathcal{O}\big(n \cdot (\mathsf{ABR}_{n^{-1}}(G) + \mathsf{MIX}_{e^{-1}}(G))\big)\\ &= \mathcal{O}\big(n \cdot (\mathsf{PAR}_{n^{-1}}(G) + \mathsf{MIX}_{e^{-1}}(G))\big),\end{aligned}$$

as desired. □

Note that there are certain graphs (e.g., star graphs and transposition graphs [Dia88, ES07]) for which $1-\lambda_2$ is not a constant, but $\mathsf{MIX}_{e^{-1}}(G) = \mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\log n)$. Hence, Corollary 5.25 gives the optimal bound of $\mathcal{O}(n \log n)$ for these graphs, while Theorem 5.11 yields an asymptotically larger upper bound.

Hamming graphs and Vertex-Expanders.

We will show that for certain Hamming graphs the bound of Theorem 5.19 outperforms not only the spectral gap-based bound (Theorem 5.11), but also the bound in Theorem 5.12 based on the vertex-expansion.

Definition 5.26. *Let* $\mathsf{Ham}_{c,d} = \prod_{i=1}^{d} \mathsf{K}_c$, *be the* (c,d)-Hamming graph *with* $n = c^d$ *vertices. That is,* $\mathsf{Ham}_{c,d}$ *is the* d*-wise Cartesian product of a complete graph with* c *vertices.*

It is easily seen that $\mathsf{Ham}_{c,d}$ is a regular graph with degree $(c-1)\cdot d$. Vertices of $\mathsf{Ham}_{c,d}$ are naturally represented as $\{1,2,\ldots,c\}^d = [c]^d$. Note that $\mathsf{Ham}_{2,d}$ gives the d-dimensional hypercube Q_d. Moreover, $\mathsf{Ham}_{c,d}$ is bipartite if and only if $c = 2$. We shall use the following bound on the broadcast time of Hamming graphs.

Theorem 5.27 ([Sau07]). *For every* $c \geqslant 2, d \geqslant 1$, $\mathsf{PAR}_{n^{-1}}(\mathsf{Ham}_{c,d}) = \mathcal{O}(\log n)$.

We will now extend the coupling method for bounding the mixing time on hypercubes (cf. [MU05, p. 276]) to general Hamming graphs.

Definition 5.28 ([MU05]). *A* coupling of a random walk *on* G *with transition matrix* $\mathbf{P}$ *is a sequence* $Z_t = (X_t, Y_t) \subseteq V \times V, t \in \mathbb{N} \cup \{0\}$, *such that for all* $x, x', y, y' \in V$,

$$\begin{aligned}\mathbf{Pr}\,[\,X_{t+1} = x' \mid Z_t = (x,y)\,] &= \mathbf{P}_{xx'},\\ \mathbf{Pr}\,[\,Y_{t+1} = y' \mid Z_t = (x,y)\,] &= \mathbf{P}_{yy'}.\end{aligned}$$

Hence, one of the sequences X_t and Y_t viewed separately behaves like the original random walk on G. The idea behind a coupling is to define Z_t such that X_t and Y_t will reach the same vertex rapidly and make identical moves from then on. The next lemma explains why we seek such couplings.

Lemma 5.29 (Coupling Lemma, [MU05]). *Let $Z_t = (X_t, Y_t)$ be a coupling for a random walk on G. Suppose that there is a step t' such that for every $x, y \in V(G)$*

$$\mathbf{Pr}\left[\, X_{t'} \neq Y_{t'} \mid X_0 = x, Y_0 = y \,\right] \leqslant \varepsilon.$$

Then

$$\mathsf{MIX}_{\varepsilon}(G) \leqslant t'.$$

The following result follows by a straightforward adaption of the well-known coupling for bounding the mixing time of hypercubes [MU05, p. 276].

Proposition 5.30. *For every $\mathsf{Ham}_{c,d}$ with $c \geqslant 3$,*

$$\mathsf{MIX}_{\varepsilon}(\mathsf{Ham}_{c,d}) \leqslant \frac{c-1}{c-2} \cdot d \cdot \ln\left(\frac{d}{\varepsilon}\right).$$

Proof. Consider two random walks X_t and Y_t. Recall that X_t and Y_t are infinite sequences of vectors $[c]^d$. For some $1 \leqslant d' \leqslant d$, $X_t(d')$ denotes the d'-th coordinate of the vector X_t. We define the coupling $Z_t = (X_t, Y_t)$ as follows.

In each step, choose some $d' \in [d]$ uniformly at random and $c' \in \{1, \ldots, c\} \backslash X_t(d')$ uniformly at random and let X_{t+1} be X_t with the d'-th coordinate replaced by c'. If $c' \neq Y_t(d')$, then let Y_{t+1} be Y_t with the d'-th coordinate replaced by c. Otherwise, we define Y_{t+1} as Y_t with the d'-th coordinate replaced by $X_t(d')$. We see that Y_t and X_t do exactly a random walk on $\mathsf{Ham}_{c,d}$ induced by $\mathbf{P}$. By construction, $X_t(d') = Y_t(d')$ implies $X_{t+1}(d') = Y_{t+1}(d')$. Furthermore, we reach $X_t(d') = Y_t(d')$ for some $d' \in [d]$, if there is some previous step $t' < t$, where $d' \in [d]$ and $c' \in [c]$ were chosen with $Y_{t'}(d') \neq c'$. In each step $t' \in \mathbb{N}$, we choose this fixed d' with probability $1/d$ and choose a $c' \in [c], c' \neq Y_{t'}(d)$ with probability $\frac{c-2}{c-1}$. Hence, the probability that for $t' := \frac{c-1}{c-2} d \log(\frac{d}{\varepsilon})$ steps still $X_{t'}(d) \neq Y_{t'}(d)$ holds is at most

$$\left(1 - \frac{1}{d} \cdot \frac{c-2}{c-1}\right)^{\frac{c-1}{c-2} d \ln(\frac{d}{\varepsilon})} \leqslant \frac{\varepsilon}{d}.$$

By the union bound we conclude that $X_{t'} \neq Y_{t'}$ with probability at most ε, and the claim follows by Lemma 5.29.

□

Combining our previous two findings we arrive at the following result.

Corollary 5.31. *Consider $\mathsf{Ham}_{c,d}$ with $n = c^d$ vertices and $d = o(\log n / \log\log n)$. Then*

$$\mathsf{COV}(\mathsf{Ham}_{c,d}) = \mathcal{O}(n \log n).$$

Proof. By an application of Theorem 5.27, $\mathsf{PAR}_{n^{-1}}(\mathsf{Ham}_{c,d}) = \mathcal{O}(\log n)$ for every $c \geqslant 3$. For $d = o(\log n / \log\log n)$ (and correspondingly, $c = \omega(\log n)$), Proposition 5.30 leads to $\mathsf{MIX}_{e^{-1}}(\mathsf{Ham}_{c,d}) = \mathcal{O}(\log n)$. Since $\deg(\mathsf{Ham}_{c,d}) = \omega(\log n)$ for this choice of d, Theorem

5.24 results in

$$\mathsf{COV}(\mathsf{Ham}_{c,d}) = \mathcal{O}\big(n \cdot (\mathsf{PAR}_{n^{-1}}(\mathsf{Ham}_{c,d}) + \mathsf{MIX}_{e^{-1}}(\mathsf{Ham}_{c,d}))\big) = \mathcal{O}(n \cdot \log n).$$

□

To show that the bounds of Theorems 5.11 and 5.12 do not give a bound of $\mathcal{O}(n \log n)$ on $\mathsf{Ham}_{c,d}$ when $d \in [\omega(1), o(\log n / \log\log n)]$, we prove that $\mathsf{Ham}_{c,d}$ is not a vertex-expander unless d is a constant.

Proposition 5.32. *For $d = \omega(1)$, $\mathsf{Ham}_{c,d}$ is not a vertex-expander.*

Proof. To show that $\mathsf{Ham}_{c,d}$ is not a vertex-expander, we prove the existence of a subset $X, |X| = \frac{n}{2}$ with the property that $|N(X)\backslash X| = o(n)$. First note that for any two adjacent vertices $u = (u_1, \ldots, u_d)$ and $v = (v_1, \ldots, v_d)$ in $\mathsf{Ham}_{c,d}$, $|\sum_{i=1}^{d} u_i - \sum_{i=1}^{d} v_i| \leqslant c-1$. Define

$$X := \Big\{(u_1, \ldots, u_d) \;\Big|\; 1 \leqslant u_i \leqslant c, \sum_{i=1}^{d} u_i \leqslant \frac{c+1}{2} d\Big\}.$$

In order to bound $|N(X)\backslash X|$, we apply the well-known probabilistic method (cf. [AS00]): we shall prove that a vertex $u = (u_1, \ldots, u_d)$ chosen uniformly at random among V satisfies

$$\Big|\sum_{i=1}^{d} u_i - \frac{c+1}{2} \cdot d\Big| = \omega(c)$$

with probability $1 - o(1)$, whence it follows that either $u \in X$ or $u \in X^c\backslash N(X)$, by which the claim follows. Note that the random variable $\sum_{i=1}^{d} u_i$ is a sum of d independent $\mathsf{Uni}[c]$-distributed random variables, i.e., each such random variables takes some value of $\{1, \ldots, c\}$ with the same probability. It is known that $\mathbf{E}[\,\mathsf{Uni}[c]\,] = \frac{c+1}{2}$ and $\mathbf{Var}[\mathsf{Uni}[c]] = \frac{c^2-1}{12}$ (cf. [GS01]). Now define an auxiliary random variable

$$Z_d := \frac{\sum_{i=1}^{d} u_i - \frac{c+1}{2} \cdot d}{\sqrt{\frac{c^2-1}{12}} \cdot \sqrt{d}}.$$

By the Central Limit Theorem (cf. Theorem 2.8),

$$\lim_{d\to\infty} \mathbf{Pr}[\,Z_d \leqslant z\,] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-\frac{x^2}{2}} dx. \tag{5.4}$$

By above normalization of $\sum_{i=1}^{d} u_i$ we have

$$\mathbf{Pr}\left[\Big|\sum_{i=1}^{d} u_i - \frac{c+1}{2} d\Big| \geqslant d^{\frac{1}{4}} c\right] \;\Leftrightarrow\; \mathbf{Pr}\left[|Z_d| \geqslant \frac{d^{\frac{1}{4}} c}{\sqrt{\frac{c^2-1}{12}} \cdot \sqrt{d}}\right].$$

Using the limit of 5.4,

$$\mathbf{Pr}\left[\,|Z_d| \geqslant \mathcal{O}(d^{-1/4})\,\right] \geqslant \left(1 - \frac{1}{\sqrt{2\pi}} \int_{-\mathcal{O}(d^{-1/4})}^{\mathcal{O}(d^{-1/4})} e^{-\frac{x^2}{2}}\, dx\right) - o(1) \geqslant 1 - o(1).$$

□

We note that with the same method, one could also show that star graphs and transposition graphs are not vertex-expanders.

Now for $d \in [\omega(1), o(\log n/\log\log n)]$, Proposition 5.32 demonstrates that $\mathsf{Ham}_{c,d}$ is not a vertex-expander (and hence not an edge-expander as well). Therefore in contrast to Corollary 5.24, the bounds of Theorem 5.12 and Theorem 5.11 are *not* tight up to a constant factor.

Cayley graphs.

Using a similar coupling idea as in the proof of Theorem 5.23, we establish two upper bounds for the cover time of Cayley graphs.

Theorem 5.33 ([Fei97a]). *Let $G = (V, E)$ be any graph. Then the expected time until $\tilde{n} \leqslant n$ distinct vertices have been visited by a random walk is bounded by*

$$\mathcal{O}\left(\tilde{n} + \tilde{n}^2 \cdot \min\left\{\Delta, \frac{\min\{\tilde{n}, \Delta\} \cdot \log n}{\delta}\right\}\right).$$

Hence for a Δ-regular graph G, the following upper bound holds:

$$\mathcal{O}\left(\min\left\{\tilde{n} + \frac{\tilde{n}^3}{\Delta}, \tilde{n}^2\right\} \cdot \log n\right).$$

Definition 5.34 (cf. [Tit00]). *Given a set S and a group $(H, \circ)$, a* group action *of H on S is a function $\Diamond$ from $H \times S$ to S satisfying*

$$\begin{aligned} (h \circ h')\Diamond s &= h\Diamond(h'\Diamond s) && \forall s \in S\ \forall h, h' \in H \\ \mathsf{id} \circ s &= s && \forall s \in S. \end{aligned}$$

A group action acts transitively *on a finite set S if for all $s, s' \in S$ there is some $h \in H$ with $h\Diamond s = s'$. Finally, we define $F(s, s') := \{h \in H \mid h\Diamond s = s'\}$.*

Lemma 5.35. *If $\Diamond$ is a transitive group action of H on S, then $|F(s, s')| = |H|/|S|$.*

Proof. We claim that for any $s, s', s'' \in S$, $|F(s, s')| = |F(s, s'')|$ holds. Let $x \in F(s, s')$. Since H acts transitively, there exists $y \in H$ such that $y\Diamond s' = s''$. Hence $(y \circ x)\Diamond s = y\Diamond(x\Diamond s) = y\Diamond s' = s''$. The translation $\varphi_y : H \to H, x \mapsto y \circ x$ for fixed y is a bijection and thus $|F(s, s')| = |F(s, s'')|$. Due to $\cup_{s' \in S} F(s, s') = H$ and $F(s, s') \cap F(s, s'') = \emptyset$ for every $s, s', s'' \in S$ we obtain $|F(s, s')| = |H|/|S|$, as needed. □

+++LOOK FOR A REFERENCE+++

Lemma 5.36. *Let H be a group which acts transitively on a finite set S and let D be some random variable on H such that for every $h \in H$, $\mathbf{Pr}[\, D = h \,] \geqslant (1 - e^{-1})|H|^{-1}$. Then for any two subsets $A, B \subseteq S$,*

$$\mathbf{E}[\, |A \cap D\Diamond B| \,] \geqslant (1 - e^{-1}) \cdot \frac{|A| \cdot |B|}{|S|}.$$

Proof. Let $A, B \subseteq S$ be two arbitrary subsets and for $a \in A, b \in B$ let $X_{a,b} = 1$ if $D\Diamond b = a$ and $X_{a,b} = 0$ otherwise. Observe that $X_{a,b}$ is a random variable, as it depends on the random variable D. By linearity of expectations and Lemma 5.35,

$$\begin{aligned}\mathbf{E}[\, |A \cap D\Diamond B| \,] &= \mathbf{E}\left[\sum_{a\in A, b\in B} X_{a,b}\right] = \sum_{a\in A, b\in B} \mathbf{Pr}[X_{a,b} = 1] = \sum_{a\in A, b\in B} \mathbf{Pr}[D\Diamond b = a] \\ &\geqslant \sum_{a\in A, b\in B} \frac{|F(b,a)|}{|H|}\left(1 - e^{-1}\right) = \left(1 - e^{-1}\right) \sum_{a\in A, b\in B} \frac{\frac{|H|}{|S|}}{|H|} = \left(1 - e^{-1}\right) \frac{|A| \cdot |B|}{|S|},\end{aligned}$$

as claimed. □

Recall that a Cayley graph $CG = (H, F)$ is defined by a group $(H, \circ)$ being the vertices and a generating set F corresponding to the edges (cf. Definition 4.1).

Theorem 5.37. *For any Cayley graph $CG = (H, F)$,*

$$\max_{u,v\in V} \mathsf{C}(u, v) = \mathcal{O}\left(\min\left\{n \cdot \sqrt{\mathsf{MIX}_{e^{-1}}(G)} \cdot \log n, n + \frac{n \cdot \left(\mathsf{MIX}_{e^{-1}}(G)\right)^{2/3} \cdot \log n}{\Delta^{1/3}}\right\}\right).$$

Proof. We begin with the first upper bound. To simplify notation, let $\mathcal{M} := \mathsf{MIX}_{e^{-1}}(G)$. Note that if $\sqrt{\mathcal{M}} \geqslant n$, then the first bound of the theorem holds trivially, as for regular graphs an upper bound of $\mathcal{O}(n^2)$ is known [Fei97b]. Henceforth we will assume $\sqrt{\mathcal{M}} \leqslant n$.

Consider a random walk and let $X_0, X_1, \ldots$ be the sequence of visited vertices at step $0, 1, \ldots$. Similar to the proof of the previous theorem, we divide the random walk into $2\frac{n}{\sqrt{\mathcal{M}}}$ consecutive phases, each of which lasts $2l$ steps, where $l := C \cdot \mathcal{M} \cdot \log n$ for some sufficiently large constant C. Each phase starts with l steps in order to "mix" the random walk and ends with another phase of length l during which we take care of the visited vertices.

Observe that the transitions of a random walk on a Cayley graph are made by choosing a generator of F uniformly at random. More precisely, if X_t is the location at step t, the next location is specified by $X_{t+1} = X_t \circ f_t$, where f_t is the uniformly chosen generator at step t.

Consider now some phase $i, 1 \leqslant i \leqslant 2\frac{n}{\sqrt{\mathcal{M}}}$ which begins at step $(i - 1) \cdot 2l$ and ends at

step $i \cdot 2l - 1$. By Lemma 5.2,

$$\mathbf{Pr}\left[X_{(i-1)\cdot 2l+l} = x \mid X_{(i-1)\cdot 2l} = x' \right] \geqslant (1 - e^{-1}) \cdot \frac{1}{|H|} \tag{5.5}$$

for every $x, x' \in H$. Consider now the sequence of l vertices

$$X_{(i-1)\cdot 2l+l}, X_{(i-1)\cdot 2l+l+1}, \ldots, X_{i\cdot 2l-1}.$$

Note that these vertices are known, once we know $X_{(i-1)\cdot 2l+l}$ and the chosen $l-1$ generators

$$f_{(i-1)\cdot 2l+l}, f_{(i-1)\cdot 2l+l+1}, \ldots, f_{i\cdot 2l-2},$$

because for every $j \geqslant 1$

$$X_{(i-1)\cdot 2l+l+j} = X_{(i-1)\cdot 2l+l+j-1} \circ f_{(i-1)\cdot 2l+l+j-1} = X_{(i-1)\cdot 2l+l} \circ \left(\bigcirc_{k=1}^{j} f_{(i-1)\cdot 2l+l+k-1}\right). \tag{5.6}$$

Let us define

$$B := \bigcup_{t=(i-1)\cdot 2l+l}^{i\cdot 2l-1} X_t = \bigcup_{j=0}^{l-1} \Big(X_{(i-1)\cdot 2l+l} \circ \left(\bigcirc_{k=1}^{j} f_{(i-1)\cdot 2l+l+k-1}\right)\Big). \tag{5.7}$$

We shall be interested in the number of distinct vertices visited between step $(i-1)\cdot 2l + l$ and $i \cdot 2l - 1$, that is $|B| = \left|\bigcup_{t=(i-1)\cdot 2l+l}^{i\cdot 2l-1} X_t\right|$. Multiplying each X_t by $X_{(i-1)\cdot 2l+l}^{-1}$ from the left side does not change $|B|$, as this multiplication is a bijection on H. Using this fact along with 5.6 yields

$$|B| = \left| \bigcup_{j=0}^{l-1} \left(\bigcirc_{k=1}^{j} f_{(i-1)\cdot 2l+l+k-1}\right) \right|,$$

demonstrating that the number of distinct visited vertices does not depend on $X_{(i-1)\cdot 2l+l}$, but only on the chosen generators in the $l-1$ subsequent steps after step $(i-1)2l + l$.

By Theorem 5.33, the expected number of steps after $\sqrt{\mathcal{M}}$ distinct vertices are visited, is bounded by $\mathcal{O}(\mathcal{M} \cdot \log n)$. It follows from Markov's inequality that after $2 \cdot \mathcal{O}(\mathcal{M} \cdot \log n)$ steps, $\sqrt{\mathcal{M}}$ distinct vertices are visited with probability at least $1/2$. Hence for sufficiently large C,

$$\mathbf{Pr}\left[|B| \geqslant \sqrt{\mathcal{M}} \right] \geqslant \frac{1}{2}. \tag{5.8}$$

To apply Lemma 5.36, note that the group operation $\circ$ from H can also be regarded as a group action on H. Moreover, $\circ$ is a transitive group action, as for any given $h_1 \in H$ and $h_2 \in H$, $h_1 \circ (h_1^{-1} \circ h_2) = h_2$. Let u be an arbitrary but fixed vertex and set $A := \{u\}$ and $D := X_{(i-1)\cdot 2l+l}$. Using Equations 5.5, 5.7 and Lemma 5.36 we obtain for any fixed $\widehat{B} \subseteq H$,

$$\mathbf{E}\left[|\{u\} \cap D\Diamond\widehat{B}| \right] = \mathbf{Pr}\left[u \in D\Diamond\widehat{B} \right] \geqslant \frac{|\widehat{B}|}{n} \cdot (1 - e^{-1}).$$

Taking conditional probabilities and using 5.8 we conclude that

$$\begin{aligned}\mathbf{Pr}\left[\, u \in D\Diamond B \,\right] &= \sum_{\widehat{B}\subseteq V} \mathbf{Pr}\left[\, B = \widehat{B} \,\right] \cdot \mathbf{Pr}\left[\, u \in D\Diamond B \mid B = \widehat{B} \,\right] \\ &\geqslant \sum_{\substack{\widehat{B}\subseteq V:\\ |\widehat{B}|\geqslant\sqrt{\mathcal{M}}}} \mathbf{Pr}\left[\, B = \widehat{B} \,\right] \cdot \frac{|\widehat{B}|}{n} \cdot (1-e^{-1}) \\ &\geqslant \frac{\sqrt{\mathcal{M}}}{n} \cdot (1-e^{-1}) \cdot \sum_{\substack{\widehat{B}\subseteq V:\\ |\widehat{B}|\geqslant\sqrt{\mathcal{M}}}} \mathbf{Pr}\left[\, B = \widehat{B} \,\right] \geqslant \frac{\sqrt{\mathcal{M}}}{n} \cdot (1-e^{-1}) \cdot \frac{1}{2}.\end{aligned}$$

Therefore, the probability that u is not visited during the $2\frac{n}{\sqrt{\mathcal{M}}}$ phases is at most

$$\left(1 - \frac{\sqrt{\mathcal{M}}}{n} \cdot (1-e^{-1}) \cdot \frac{1}{2}\right)^{2\frac{n}{\sqrt{\mathcal{M}}}} < 1,$$

having used the assumption $\sqrt{\mathcal{M}} \leqslant n$. Consequently, $\max_{u,v\in V} \mathsf{C}(u,v) = \mathcal{O}(\frac{n}{\sqrt{\mathcal{M}}} \cdot \mathcal{M} \cdot \log n)$ and the first bound follows.

The second upper bound is shown with similar arguments. Also here, we first remark that whenever $\mathcal{M}\cdot\Delta \geqslant n^3$, the second bound holds trivially, as $\frac{n\cdot\mathcal{M}^{2/3}}{\Delta^{1/3}} \geqslant \frac{n^3}{\Delta} \geqslant n^2$. Therefore, we assume $\mathcal{M} \cdot \Delta \leqslant n^3$ in what follows. Theorem 5.33 implies that the expected time to visit $\tilde{n}$ distinct vertices is bounded by $\tilde{n} + \frac{\tilde{n}^3}{\Delta}\log n$. Here, we set $\tilde{n} := (\mathcal{M}\cdot\Delta)^{1/3} \leqslant n$ and consider $2n/\tilde{n}$ phases each of which lasts $\mathcal{O}(\mathcal{M}) + \tilde{n} + \frac{\tilde{n}^3}{\Delta}\log n = \mathcal{O}(\mathcal{M}\log n + \tilde{n})$ steps. In each such a phase $\tilde{n}$ distinct vertices are visited with probability $1/2$ and since there are $n/\tilde{n}$ independent phases in total, after

$$\frac{n}{\tilde{n}} \cdot \mathcal{O}(\mathcal{M}\log n + \tilde{n}) = \mathcal{O}\left(\frac{n\cdot\mathcal{M}\log n}{\tilde{n}} + n\right) = \mathcal{O}\left(\frac{n\cdot\mathcal{M}^{2/3}\log n}{\Delta^{1/3}} + n\right)$$

time-steps, an arbitrary but fixed vertex has been visited with constant probability. □

The first bound on the maximum commute time is tight, as for the n-cycle $\mathsf{MIX}_{e^{-1}}(G) = \Theta(n^2) = \mathsf{COV}(G)$. The second bound implies that for Cayley graphs with $\mathsf{MIX}_{e^{-1}}(G) = \mathcal{O}(\sqrt{\Delta})$, the maximum commute time is of order n. For instance, for $\mathsf{Ham}_{c,d}$ with $c = \Omega(\log^2 n)$ we can derive an upper bound of $\mathsf{COV}(G) = \mathcal{O}(n\log n)$ since $\mathsf{MIX}_{e^{-1}}(\mathsf{Ham}_{c,d}) = \mathcal{O}(\log n)$ by Proposition 5.30.

In comparison to the bound of Aldous [Ald83], our two bounds are less precise since the leading constant is $\mathcal{O}(1)$ in comparison with $1+o(1)$. However, our bound is somewhat more general, as the bound of Aldous depends also on the expected number of returns to the starting point during $\mathsf{MIX}_{e^{-1}}(G)$ many steps. Furthermore, his bound requires $\mathsf{MIX}_{e^{-1}}(G)$ not to be polynomial in n.

5.5 Lower Bounds on the Cover Time

In this section we present lower bounds on the cover time depending on the runtime of the push algorithm. Since in the push algorithm propagation occurs at *all* informed vertices, while in the random walk propagation occurs only from the current location, the following observation is immediate.

Observation 5.38. *For any graph G,* $\mathbf{E}[\,\mathsf{COV}(G)\,] \geqslant \mathbf{E}[\,\mathsf{PAR}(G)\,]$.

The graph $\mathsf{K}_{1,n-1}$ provides an example where this inequality is almost tight. Indeed, a straightforward computation based on the famous coupon collector's problem (cf. Theorem 2.19) shows that $\mathbf{E}[\,\mathsf{COV}(\mathsf{K}_{1,n-1})\,] = (2 \pm o(1)) \cdot \mathbf{E}[\,\mathsf{PAR}(\mathsf{K}_{1,n-1})\,]$. This example demonstrates that further assumptions on the graph class are necessary in order to establish some considerable (e.g., polynomial) gap between the cover time and broadcast time. We first present bounds which are designed for graphs with a small maximum degree.

5.5.1 Sparse Graphs

In this subsection we derive lower bounds on the cover time that are strongest for sparse graphs, e.g., graphs with a bounded degree.

Definition 5.39. *Given a graph $G = (V, E)$, a set $\Pi \subseteq E(G)$ is called a* cutset separating *$u \in V$* from *$v \in V$ if every path from u to v includes an edge of Π.*

The next result relies on a technique which was already introduced in 1959 by Nash-Williams [NW59]. It confirms the intuition that if there are many disjoint small cutsets separating two vertices u, v, then the random walk requires a long time to commute between u and v.

Theorem 5.40 ([LPW06]). *If $\{\Pi_i\}_{i=1}^n$ are disjoint cutsets separating u from v, then*

$$\mathsf{R}(u, v) \geqslant \sum_{i=1}^{n} |\Pi_i|^{-1}.$$

With this theorem at hand, it is easy to prove the following fundamental result.

Corollary 5.41. *Let u, v be two vertices of G. Then* $\mathsf{C}(u, v) \geqslant 2\,(\mathrm{dist}(u, v))^2$.

Proof. For the proof of the corollary we first observe the following claim, which is a simple consequence of the inequality between the harmonic and arithmetic mean.

Claim. Let $x_1, \ldots, x_n$ be positive integers. Then $\sum_{k=1}^n x_k^{-1} \geqslant n^2 (\sum_{k=1}^n x_k)^{-1}$.

For each integer $0 \leqslant i \leqslant \mathrm{dist}(u, v) - 1$, let $\Pi_i := \{\{r, s\} \in E \mid \mathrm{dist}(u, r) = i \wedge \mathrm{dist}(u, r) = i + 1\}$. Clearly, all Π_i's are disjoint and every path from u to v must include at least one

edge of each Π_i. Hence, by Theorem 5.40 and the claim above

$$\mathsf{R}(u,v) \geqslant \sum_{i=0}^{\operatorname{dist}(u,v)-1} \frac{1}{|\Pi_i|} \geqslant \frac{(\operatorname{dist}(u,v))^2}{|E|}.$$

As $\mathsf{C}(u,v) = 2|E| \cdot \mathsf{R}(u,v)$, the corollary follows. □

A related result can be found in [Zuc92, Corollary 6] showing that for two arbitrary vertices u, v on a tree, $\mathsf{H}(u,v) \geqslant (\operatorname{dist}(u,v))^2$. However, the following simple example shows that it is not possible to replace $\mathsf{C}(u,v)$ by $\mathsf{H}(u,v)$ in the bound of Corollary 5.41 (at the cost of some constant factor). Take a complete binary tree with n leaves and connect all of them to a new vertex v. Let r be the root of the binary tree. By simple probabilistic arguments, one can verify that $\mathsf{H}(r,v) = \Theta(\operatorname{dist}(u,v)) = o(\operatorname{dist}(u,v)^2)$.

Proposition 5.42. *Let $G = (V,E)$ be a graph. Then* $\mathsf{COV}(G) = \Omega\left(\frac{\sqrt{n \log n}}{\Delta} \cdot \mathsf{PAR}_{n^{-1}}(G)\right)$.

Proof. This proof is done by a simple case analysis. Suppose first that $\operatorname{diam}(G) \leqslant \sqrt{n \log n}$. By a result of [FPRU90],

$$\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\Delta \cdot (\operatorname{diam}(G) + \log n)) = \mathcal{O}(\Delta \cdot \sqrt{n \log n}).$$

Since $\mathsf{COV}(G) \geqslant (1 + o(1)) \cdot n \ln n$ (Theorem 5.8) we obtain

$$\begin{aligned}\mathsf{COV}(G) &\geqslant (1 + o(1)) \cdot n \ln n \\ &= \Omega\left(\sqrt{n \log n} \cdot \sqrt{n \log n}\right) = \Omega\left(\sqrt{n \log n} \cdot \frac{\mathsf{PAR}_{n^{-1}}(G)}{\Delta}\right).\end{aligned}$$

Now assume $\operatorname{diam}(G) \geqslant \sqrt{n \log n}$. Then $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\Delta \cdot \operatorname{diam}(G))$ and by Corollary 5.41 we arrive at

$$\mathsf{COV}(G) \geqslant \operatorname{diam}(G)^2 \geqslant \Omega\left(\sqrt{n \log n} \cdot \frac{\mathsf{PAR}_{n^{-1}}(G)}{\Delta}\right),$$

which completes the proof. □

As demonstrated by the two-dimensional torus graph where $\mathsf{PAR}_{n^{-1}}(G) = \Theta(\sqrt{n})$ and $\mathsf{COV}(G) = \Theta(n \log^2 n)$ [Zuc92], the bound of Proposition 5.42 is tight up to a factor of $\log^{3/2} n$ for bounded degree graphs.

The next result improves on Proposition 5.42 by a factor of almost $\sqrt{\Delta}$ for regular graphs. Moreover, its proof *directly* relates the cover time to the broadcast time of the given graph.

Theorem 5.43. *Let G be a regular graph with $\Delta = n^\alpha$ where α is at least some constant > 0. Then*

$$\mathsf{COV}(G) = \Omega\left(\frac{\sqrt{n}}{\sqrt{\Delta} \cdot \log^2 n} \cdot \mathsf{PAR}_{n^{-1}}(G)\right).$$

Proof. The basic proof idea is as follows. We first define a modification of the push algorithm which performs not faster than the original push algorithm. We will show that this modification performs slowly if and only if many disjoint cutsets between two proper vertices exist.

The precise definition of the modified push algorithm and the construction of the disjoint cutsets is given in Figure 5.3.

Modified Push Algorithm

Input: Graph $G = (V, E)$
Output: Disjoint Cutsets $\{\Pi_w\}_{w=1}^{|\mathcal{W}|}$

$t \leftarrow 0$, $w \leftarrow 0$, $\mathcal{W} \leftarrow \emptyset$
$I_t \leftarrow \{s\}$
while $I_t \neq V$ **do**
  if $|E(I_t, I_t^c)| \geqslant \Delta^{1+\varepsilon}$ **then** mode $\leftarrow P$ **else** mode $\leftarrow W$
  if mode $= P$ **then** // perform like original push algorithm
    for all vertices $u \in I_t$ **do**
      choose $v \in N(u)$ u. a. r.
      $I_{t+1} \leftarrow I_{t+1} \cup \{v\}$
    end for
    $t \leftarrow t + 1$
    $I_{t+1} \leftarrow I_t$
  end if
  if mode $= W$ **then** // wait until $N(I_t)$ has been informed
    $I \leftarrow I_t$
    $w \leftarrow w + 1$
    $\mathcal{W} \leftarrow \mathcal{W} \cup \{t\}$
    $\Pi_w \leftarrow E(I, I^c)$ // another disjoint cutset
    while $I_t \neq I \cap N(I)$ **do**
      for all vertices $u \in I$ **do**
        choose $v \in N(u)$ u. a. r.
        $I_{t+1} \leftarrow I_{t+1} \cup \{v\}$
      end for
      $t \leftarrow t + 1$
      $I_{t+1} \leftarrow I_t$
    end while
    mode $\leftarrow P$
  end if
end while

Fig. 5.3: *Definition of the modified push algorithm* $\overline{\mathsf{PAR}}$.

Let $\overline{\mathsf{PAR}}(G)$ denote the random variable representing the runtime of this modified push algorithm. It is evident that this modified version is not faster and so $\mathsf{PAR}_{n^{-1}}(G) \leqslant \overline{\mathsf{PAR}}_{n^{-1}}(G)$. Next we prove an upper bound on $\overline{\mathsf{PAR}}_{n^{-1}}(G)$.

Claim. For any graph $G = (V, E)$, $\overline{\mathsf{PAR}}_{n^{-1}}(G) \leqslant |\mathcal{W}| \cdot \frac{5}{\alpha} \cdot \Delta \ln \Delta + 5n\Delta^{-\varepsilon}$.

Proof. We first distinguish between the two modes P and W.

1. Consider now some step t with $|E(I_t, I_t^c)| \geqslant \Delta^{1+\varepsilon}$ and mode P. By using a Chernoff bound [MU05] along with the fact that $\Delta = n^\alpha, \alpha > 0$, the set of informed vertices increases by at least $\frac{1}{5}\Delta^\varepsilon$ with probability $1 - n^{-3}$, whenever $\varepsilon > 0$ (cf. [ELS07]). Let $\mathcal{P}$ be the set of time-steps at which $\overline{\mathsf{PAR}}$ is in mode P after the execution of line 4. Using the union bound we conclude that $|\mathcal{P}| \leqslant \frac{n}{5\Delta^{1+\varepsilon}}$ with probability $1 - n^{-2}$.

2. On the other hand consider some time-step t when $|E(I_t, I_t^c)| \leqslant \Delta^{1+\varepsilon}$ occurs in line 4 and the mode changes to W. We are interested in the number of time-steps until the mode changes to P, which is the same as the required number of steps to inform every vertex in $N(I_t)$. The probability that on one edge of $E(I_t, I_t^c)$ the rumor is not sent within $(5/\alpha) \cdot \Delta \ln(\Delta)$ steps equals $\left(1 - \frac{1}{\Delta}\right)^{(5/\alpha)\Delta \ln \Delta} \leqslant \left(1 - \frac{1}{\Delta}\right)^{(5/\alpha)\Delta\alpha \ln n} \leqslant n^{-5}$. By applying the union bound over all edges, we conclude that within $(5/\alpha)\Delta \ln \Delta$ steps the rumor is sent along all edges of $E(I_t, V \backslash I_t)$ with probability $1 - n^{-3}$. Clearly, $|E(I_t, I_t^c)| \leqslant \Delta^{1+\varepsilon}$ in line 4 occurs at most $|\mathcal{W}| \leqslant n$ times. Hence, by the union bound we conclude that after each execution of line 14, $\overline{\mathsf{PAR}}$ spends at most $5/\alpha \cdot \Delta \ln \Delta$ steps in the while-loop between lines 18–25 with probability $1 - n^{-2}$.

Now the case study above implies that

$$\overline{\mathsf{PAR}}_{n^{-1}}(G) \leqslant |\mathcal{W}| \cdot \frac{5}{\alpha} \cdot \Delta \ln \Delta + |\mathcal{P}| \leqslant |\mathcal{W}| \cdot \frac{5}{\alpha} \cdot \Delta \ln \Delta + \frac{n}{5\Delta^{1+\varepsilon}},$$

which finishes the proof of the claim. □

We now derive a lower bound on $\mathsf{COV}(G)$ by means of $|\mathcal{W}|$.

Claim. For any graph G and any execution of $\overline{\mathsf{PAR}}$, $\mathsf{COV}(G) \geqslant \frac{1}{2}n|\mathcal{W}|\Delta^{-\varepsilon}$.

Proof. Denote by l one vertex which becomes informed during the last step t of the modified push algorithm. We first show that $\{\Pi_w\}_{w=1}^{|\mathcal{W}|}$ are disjoint cutsets separating s from l and then apply the inequality of Nash-Williams. Write $\mathcal{W} = \{t_1, t_2, \ldots, t_{|\mathcal{W}|}\}$ in the order the Π_w are assigned in line 17. We begin by proving that $\{\Pi_w\}_{w=1}^{|\mathcal{W}|}$ are disjoint.

To see this consider Π_w and $\Pi_{w'}$ where $1 \leqslant w < w' \leqslant |\mathcal{W}|$. Let t_w and $t_{w'}$ be the respective time-steps when Π_w and Π'_w are assigned in line 18. By definition of $\overline{\mathsf{PAR}}$, $\Pi_w = E(I_{t_w}, I_{t_w}^c)$ and $\Pi_{w'} = E(I_{t_{w'}}, I_{t_{w'}}^c)$. For the same reason, $I_{t_w}^c \cap N(I_{t_w}) \subseteq I_{t_{w+1}} \subseteq I_{t_{w'}}$ and therefore it is not possible that $\Pi_{w'}$ contains an edge of Π_w.

Next we prove that each $\Pi_w, 1 \leqslant w \leqslant |\mathcal{W}|$, is a cutset separating s from l. Fix some w and consider Π_w assigned in time-step t_w. Recall that the set Π_w contains all edges between I_{t_w} and $I_{t_w}^c$. The set I_{t_w} is a subset of vertices containing s, while $I_{t_w}^c$ is a subset of vertices containing l. Hence, every path from s to l must include at least one edge between I_{t_w} and $I_{t_w}^c$, and consequently, one edge of Π_w.

Since $\{\Pi_w\}_{w=1}^{|\mathcal{W}|}$ are disjoint cutsets, we obtain by an application of Theorem 5.40

$$\mathsf{R}(s,l) \geqslant \sum_{w=1}^{|\mathcal{W}|} |\Pi_w|^{-1} \geqslant \sum_{w=1}^{|\mathcal{W}|} \Delta^{-1-\varepsilon} = |\mathcal{W}| \cdot \Delta^{-1-\varepsilon}$$

and therefore $\mathsf{C}(s,l) = 2|E| \cdot \mathsf{R}(s,l) \geqslant 2 \cdot \frac{1}{2} n\Delta|\mathcal{W}|\Delta^{-1-\varepsilon} = n|W|\Delta^{-\varepsilon}$, implying $\mathsf{COV}(G) \geqslant \frac{1}{2} n|\mathcal{W}|\Delta^{-\varepsilon}$. □

We are now in a position to put our bounds together. Combining the results of the two claims above with the general lower bound $\mathsf{COV}(G) \geqslant (1+o(1))n\ln n$ (Theorem 5.8) we arrive at

$$\frac{\mathsf{COV}(G)}{\mathsf{PAR}_{n^{-1}}(G)} \geqslant \frac{\max\{(1+o(1))n\ln n, \frac{1}{2}n|\mathcal{W}|\Delta^{-\varepsilon}\}}{|\mathcal{W}| \cdot \frac{5}{\alpha}\Delta\ln\Delta + 5n\Delta^{-\varepsilon}}. \tag{5.9}$$

Finally, we choose $\varepsilon = \frac{1-\alpha}{2\alpha} > 0$ and apply a case analysis on 5.9. First, let $|\mathcal{W}| \cdot \frac{5}{\alpha}\Delta\ln(\Delta) \geqslant 5n\Delta^{-\varepsilon}$. Then 5.9 is lower bounded by

$$\frac{\frac{1}{2}n|\mathcal{W}|\Delta^{-\varepsilon}}{2|\mathcal{W}|\frac{5}{\alpha}\Delta\ln(\Delta)} = \frac{1}{20}\cdot\frac{n}{\ln n}\Delta^{-1-\varepsilon} = \frac{n}{20\ln n}n^{(\alpha\cdot(-1-\frac{1-\alpha}{2\alpha}))} = \frac{1}{20\ln n}\cdot\frac{n^{\frac{1}{2}-\frac{\alpha}{2}}}{\log n}.$$

Now let $|\mathcal{W}| \cdot \frac{5}{\alpha}\Delta\ln(\Delta) \leqslant 5n\Delta^{-\varepsilon}$. In this case 5.9 is bounded below by

$$\frac{(1+o(1))n\ln n}{10n\Delta^{-\varepsilon}} = \frac{1}{10}(1+o(1))\ln n\cdot\Delta^{\varepsilon} = \frac{1}{10}(1+o(1))\ln n\cdot n^{(\alpha\cdot\frac{1-\alpha}{2\alpha})} = \frac{1}{10}(1+o(1))\ln n\cdot n^{\frac{1}{2}-\frac{\alpha}{2}},$$

and the claim follows. □

5.5.2 Dense Graphs

In this section we focus on graphs with a degree of order n. First, we briefly mention results for the case $\delta(G) \geqslant \lfloor \frac{n}{2} \rfloor$. As shown by [CRR+97], $\mathsf{COV}(G) = \mathcal{O}(n\log n)$, whenever $\delta(G) \geqslant \lfloor \frac{n}{2} \rfloor$. The authors also observed that there is an abrupt change for $\delta(G) = \lfloor \frac{n}{2} \rfloor - 1$: for the graph G consisting of two complete graphs with $n/2$ vertices connected by an edge, $\mathsf{COV}(G) = \Theta(n^2)$.

We note that it is possible to proof the analogue result $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\log n)$ for every graph with $\delta \geqslant \lfloor \frac{n}{2} \rfloor$. Hence, for $\delta(G) \geqslant \lfloor \frac{n}{2} \rfloor$, the gap between cover time and broadcast time is (trivially) of order n. Instead of proving this bound, we will establish a gap of approximately n for any regular graph of degree $\Omega(n)$. Before doing so, we have to introduce some further notation and definitions.

Consider a random walk $X_0 = s, X_1, \ldots$ on G starting from s. Note that an instance of the random walk can be described (as the parallel push algorithm) by $(N_{t,v})_{t\in\mathbb{N}, v\in V}$. Here, $N_{t,v}$ represents the neighbor of v to which the random walk moves after the t-th visit of v. Denote the number of visits to v until time t as $W_t(s,v) := |\{0 \leqslant t' \leqslant t : X_{t'} = v, X_0 = s\}|$.

Definition 5.44 ([KKLV00]). *Consider a graph $G = (V, E)$ and a random walk starting from $s \in V$. Let*

$$\mathsf{BLA}_s := \mathbf{E}\left[\min\left\{t \in \mathbb{N} : \frac{W_t(s,v)\deg(v')}{W_t(s,v')\deg(v)} < 2 \ \forall v, v' \in V\right\}\right],$$

where $\frac{x}{0} := \infty$ for every $x \geqslant 0$. The blanket time *of G is $\mathsf{BLA}(G) = \max_{s\in V} \mathsf{BLA}_s$.*

Consider some step t such that the condition of the blanket time is satisfied, i. e., $\frac{W_t(s,v)\deg(v')}{W_t(s,v')\deg(v)} < 2$ for all $v, v' \in V$. Clearly, there is at least one $v' \in V(G)$ such that $W_t(s, v') \leqslant t \cdot \pi(v') = t \cdot \deg(v')/(2|E|)$. Hence for every $v \neq v'$,

$$2 > \frac{W_t(s,v)\deg(v')}{W_t(s,v')\deg(v)} \geqslant \frac{W_t(s,v)\cdot\deg(v')}{t\cdot\frac{\deg(v')}{2|E|}\cdot\deg(v)} = \frac{W_t(s,v)\cdot 2|E|}{t\cdot\deg(v)},$$

and so $W_t(s, v) < 2 \cdot \frac{\deg(v)}{2|E|} \cdot t$. Using symmetrical arguments we arrive at

$$\frac{1}{2}\cdot\frac{\deg(v)}{2|E|}\cdot t \leqslant W_t(s,v) \leqslant 2\cdot\frac{\deg(v)}{2|E|}\cdot t$$

for every vertex $v \in V$. Hence, after expected $\mathsf{BLA}(G)$ steps every vertex is visited as many times as predicted by the stationary distribution, up to a factor of 2. We will use the following upper bound on $\mathsf{BLA}(G)$ by Kahn et al.

Theorem 5.45 ([KKLV00]). *For any graph $G = (V, E)$,*

$$\mathsf{BLA}(G) = \mathcal{O}\left(\mathsf{COV}(G)\cdot(\ln\ln n)^2\right).$$

We observe the following simple graph-theoretical lemma (Recall that a 2-cover of G is a subset $X \subseteq V$ such that for all $v \in V$ there is an $x \in X$ with $\operatorname{dist}(x, v) \leqslant 2$).

Lemma 5.46. *Let $G = (V, E)$ be any graph with minimum degree δ. Then there is a 2-cover X of G with $|X| \leqslant \lceil \frac{n}{\delta} \rceil$.*

Proof. We give the algorithm for the construction of the 2-cover in Figure 5.4. It is obvious that this algorithm terminates and produces indeed a 2-cover of $V(G)$. We now derive a specific upper bound on the number of iterations, which equals the size of the returned 2-cover.

For some $i \in \mathbb{N}$, let $Y_i := \{v \in V(G) \mid \forall j \leqslant i : \operatorname{dist}(u, u_j) > 1\}$. Clearly, $Y_0 = V(G)$. Let u be a vertex such that for every $j \leqslant i$, $\operatorname{dist}(u, u_j) > 2$. Hence, for some $v \in N(u)$, $\operatorname{dist}(v, u_j) > 1$ for every $j \leqslant i$. Therefore, $|Y_i| \leqslant |Y_{i-1}| - |N(u)| \leqslant |Y_{i-1}| - \delta$ and there can be at most $\lceil \frac{n}{\delta} \rceil$ iterations of the while loop. □

It is interesting to compare the result of the prior lemma to the known results about the size of 1-covers, i. e., dominating sets.

GREEDY 2-COVER

Input: Graph $G = (V, E)$
Output: A 2-Cover $\cup_i u_i$

1: $i \leftarrow 0$
2: **while** $\exists u \in V : \forall j \leqslant i : \mathrm{dist}(u, u_j) > 2$ **do**
3: Choose such an $u \in V$
4: $u_i \leftarrow u$
5: $i \leftarrow i + 1$
6: **end while**

Fig. 5.4: *A simple greedy algorithm for the construction of a 2-cover.*

Theorem 5.47 ([AS00, p.4 and p.178]). *Let G be a graph with $\delta > 1$. Then G has a 1-cover of size at most $n \cdot \frac{1+\ln(\delta+1)}{\delta+1}$ and there are graphs with $\delta \geqslant n/2$ such that every 1-cover is of size $\Omega(\log n)$.*

Consequently, there is an unexpected discrepancy between 1- and 2-covers in dense graphs: for every dense graph, there are 2-covers of constant size, while there are dense graphs for which *every* 1-cover is of logarithmic order.

Together with Theorem 5.19, the next theorem essentially shows that the gap between cover time and broadcast time is approximately linear in n for dense graphs.

Theorem 5.48. *Let $G = (V, E)$ be a Δ-regular graph. Then*

$$\mathbf{E}\left[\,\mathsf{PAR}(G)\,\right] = \mathcal{O}\left(\frac{1}{\Delta} \cdot \mathsf{BLA}(G) + \frac{n^2}{\Delta^2} \cdot \log^2 n\right).$$

Proof. Let us briefly describe the main idea of the proof. We first show that for every vertex v there is a fixed, i.e., independent of the execution of the push algorithm, set of vertices $Y = Y(v) \subseteq V$ of size at least $\Delta/4$ such that v informs an arbitrary fixed vertex in Y within $\mathcal{O}((n/\Delta) \cdot \log^2 n)$ steps with high probability. We then establish that if vertex u informs v in $\mathcal{O}((n/\Delta) \cdot \log^2 n)$ steps with high probability, then also v informs u in $\mathcal{O}((n/\Delta) \cdot \log^2 n)$ steps with high probability. Using this fact and Lemma 5.46 we find that there is a partitioning of V into a constant number of partitions with the following property: once a vertex in some partition becomes informed, the whole partition becomes informed within $\mathcal{O}((n/\Delta) \cdot \log^2 n)$ steps. Finally, we use a coupling between the random walk and the push algorithm to show that if the random walk covers the whole graph quickly, then the rumor will also be quickly propagated from one partition to the other partitions.

We note that we will use the sequential push algorithm SEQ for this proof. We first consider the edge-expansion of $X \subseteq V$ where $1 \leqslant |X| \leqslant \Delta$. Clearly, $|E(X, X^c)| = |E(X)| -$

$2\cdot|E(X,X)| \geqslant \Delta\cdot|X| - 2\cdot\frac{|X|\cdot(|X|-1)}{2} \geqslant |X|\cdot(\Delta - |X| + 1)$. Consequently for $1 \leqslant m \leqslant \Delta - 1$,

$$\begin{aligned}\Lambda(m) &= \min_{X\subseteq V(G),|X|=m}\left\{\frac{|E(X,X^c)|}{\Delta(G)}\right\}\\ &= \frac{1}{\Delta}\min_{X\subseteq V(G),|X|=m}|E(X,X^c)| \geqslant \frac{1}{\Delta}\cdot m\cdot(\Delta - m + 1).\end{aligned}$$

By Proposition 3.12,

$$\begin{aligned}\mathbf{E}\left[\,\mathsf{SEQ}(G,\Delta)\,\right] &\leqslant \sum_{m=1}^{\Delta}\frac{1}{\Lambda(m)}\\ &\leqslant \Delta\cdot\sum_{m=1}^{\Delta}\frac{1}{m\cdot(\Delta-m+1)}\\ &= \Delta\cdot\left(\sum_{m=1}^{\Delta}\frac{1}{(\Delta+1)\cdot m} + \sum_{m=1}^{\Delta}\frac{1}{(\Delta+1)\cdot(\Delta+1-m)}\right)\\ &\leqslant 2\cdot\sum_{m=1}^{\Delta}\frac{1}{m} \leqslant 3\ln n.\end{aligned}$$

Since for each $1 \leqslant m' \leqslant \Delta$, $\Lambda(m')\cdot\left(\sum_{m=1}^{\Delta-1}(\Lambda(m))^{-1}\right) \geqslant 3\ln n$, by Proposition 3.12

$$\mathsf{SEQ}_{n^{-1}}(G,\Delta) = \mathcal{O}\left(\sum_{m=1}^{\Delta-1}\frac{1}{\Phi(m)}\right) = \mathcal{O}(\log n).$$

To summarize the first part of the proof, we have shown that $\mathbf{Pr}\left[\,|I_{C\log n}| \geqslant \Delta\,\right] \geqslant 1 - n^{-1}$, where C is a sufficiently large constant. By definition of expectation, $\mathbf{E}\left[\,|I_{C\log n}\backslash\{u\}|\,\right] \geqslant \Delta/2$. Define $p(u,v) := \mathbf{Pr}\left[\,v \in I_{C\log n} \mid I_0 = \{u\}\,\right]$, $Y := \{v \in V, v \neq u \mid p(u,v) \geqslant \Delta/(4n)\}$. It follows that

$$\begin{aligned}\frac{\Delta}{2} \leqslant \mathbf{E}\left[\,|I_{C\log n}\backslash\{u\}|\,\right] &= \sum_{v\in V(G),v\neq u}p(u,v)\\ &= \sum_{v\in Y}p(u,v) + \sum_{v\notin Y,v\neq u}p(u,v).\end{aligned}$$

To lower bound $|Y|$, assume that $p(u,v) = 1$ for all $v \in Y$ and $p(u,v) = \Delta/(4n)$ for $v \notin Y \cup \{u\}$. Then,

$$\sum_{v\in Y}1 + \sum_{v\notin Y,v\neq u}\frac{\Delta}{4n} = |Y| + (n - |Y| - 1)\cdot\frac{\Delta}{4n} \leqslant |Y| + \frac{\Delta}{4},$$

and hence $|Y| \geqslant \frac{\Delta}{4}$. Consider an arbitrary but fixed vertex $v \in Y$. By definition of Y,

$$\mathbf{Pr}\left[\, v \in I_{C \log n} \mid I_0 = \{u\} \,\right] \geqslant \frac{\Delta}{4n},$$

and therefore

$$\mathbf{Pr}\left[\, v \in I_{16C\frac{n}{\Delta}\log^2 n} \;\middle|\; I_0 = \{u\} \,\right] \;\geqslant\; 1 - \left(1 - \frac{\Delta}{4n}\right)^{\frac{4n}{\Delta} 4 \log n} \geqslant 1 - n^{-4}. \tag{5.10}$$

Lemma 5.49. *Let $u, v \in V(G)$ be two vertices of a regular graph G. Then $\mathsf{PAR}_{n^{-1}}(v,u) = \mathcal{O}(\mathsf{PAR}_{n^{-1}}(u,v))$, whenever C is large enough.*

Proof. Consider the algorithm $\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}$. Note that on regular graphs, this algorithm can be described as follows. In each step $t \in \mathbb{T}$ an edge $\{a(t), b(t)\}$ of E is chosen uniformly at random after which $a(t)$ and $b(t)$ become informed if one of the vertices have already been informed at step $t-1$. Let $(a(t), b(t))_{t=1}^{\infty}$ be the sequence of vertices which occur in an instance of $\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}$, i. e., in step t the edge $\{a(t), b(t)\}$ is chosen for a push and pull transmission. By our assumption, if the algorithm $\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}$ chooses some $(a(t), b(t))_{t=1}^{16C(n/\Delta)\log^2 n} \cup (a(t), b(t))_{t=16C(n/\Delta)\log^2 n+1}^{\infty}$, then the vertex v becomes informed within $16C(n/\Delta)\log^2 n$ steps with probability $1 - n^{-1}$, provided that u is the initially informed vertex. Suppose that $\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}$ chooses the sequence $(a(t), b(t))_{t=16C(n/\Delta)\log^2 n}^{1} \cup (a(t), b(t))_{t=16Cn/\Delta\log^2 n+1}^{\infty}$, where the order of the choices in the first $16C(n/\Delta)\log^2 n$ time-steps has been reversed, with the same probability. Then u becomes informed provided that v is initially informed. Hence,

$$\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}_{n^{-1}}(u,v) = \mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}_{n^{-1}}(v,u). \tag{5.11}$$

To relate this result to SEQ we write

$$\begin{aligned}
\mathsf{SEQ}_{n^{-1}}(v,u) &= \mathcal{O}\left(\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}_{n^{-1}}(v,u)\right) && \text{(by Lemma 3.8)}\\
&= \mathcal{O}\left(\mathsf{PUSH}-\mathsf{PULL}-\mathsf{SEQ}_{n^{-1}}(u,v)\right) && \text{(by 5.11)}\\
&= \mathcal{O}\left(\mathsf{SEQ}_{n^{-1}}(u,v)\right),
\end{aligned}$$

which completes the proof of the lemma. □

Consider the auxiliary graph $\widehat{G} = (\widehat{V}, \widehat{E})$ defined as follows: $\widehat{V} := V$ and $\{u,v\} \in \widehat{E}$ iff

$$\max\{\mathsf{PAR}_{n^{-1}}(u,v), \mathsf{PAR}_{n^{-1}}(v,u)\} \leqslant 16C \cdot \frac{n}{\Delta} \cdot \log^2 n.$$

Since $|Y| \geqslant \Delta/4$, Lemma 5.46 implies the existence of a 2-cover $u_1, u_2, \ldots, u_k$, $k \leqslant \lceil n/\Delta \rceil$ of $\widehat{G}$. Hence the sets $U_i := \{v \in \widehat{V} \mid \operatorname{dist}_{\widehat{G}}(v, u_i) \leqslant 2\}, 1 \leqslant i \leqslant k$ form a (possibly non-disjoint) partitioning of $\widehat{V}$. Take a disjoint partitioning $V_1, V_2, \ldots, V_k$ such that for every $1 \leqslant i \leqslant k$, $V_i \subseteq U_i$. Consider two arbitrary vertices u and v in the same partition V_i. If a

vertex $u \in V_i$ becomes informed at step t, then $V_i \subseteq I_{t+\mathcal{O}(\frac{n}{\Delta}\log^2 n)}$ with probability $1 - n^{-3}$ by 5.10.

Consider now the directed graph $G' := (V', E')$ with $V' := \{V_1, V_2, \ldots, V_k\}$ and

$$E' := \left\{ (V_i, V_j) \mid \exists u \in V_i : N_{t,u} \in V_j, 1 \leqslant t \leqslant 4 \cdot \frac{\mathsf{BLA}_s(G)}{n} \right\}.$$

Claim. Let $s \in V_i$. With probability $1/2$, there is a path from V_i to every V_j in G'.

Proof. Recall that a random walk *and* an instance of the push algorithm is described by an infinite matrix $(N_{t,v})_{t\in\mathbb{N},v\in V}$. Let $\mathcal{A}$ be the event that $\mathsf{BLA}_s(G)(\omega) \geqslant 2 \cdot \mathsf{BLA}(G)$, where $\mathsf{BLA}_s(G)(\omega)$ is the corresponding random variable to the expected value $\mathsf{BLA}_s(G)$. By Markov's inequality, $\mathbf{Pr}[\mathcal{A}] \leqslant \frac{1}{2}$. As an intermediate step, consider the auxiliary graph $\tilde{G} := (\tilde{V}, \tilde{E})$ with $\tilde{V} := V$ and

$$\tilde{E} := \left\{ (u, v) \in V \times V \mid N_{t,u} = v, 1 \leqslant t \leqslant 4 \cdot \frac{\mathsf{BLA}_s(G)}{n} \right\}.$$

From now on, we condition on the event $\mathcal{A}$. Then the random walk of length $2 \cdot \mathsf{BLA}_s(G)$ starting from s visits every vertex of V and no vertex more often than $4 \cdot \frac{\mathsf{BLA}_s(G)}{n}$ times. As a consequence, every vertex can be reached from s in the graph $\tilde{G}$. Since G' is obtained from $\tilde{G}$ by contracting vertices, this property must be preserved and every partition $V_j \in V'$ can be reached from V_i. As event $\mathcal{A}$ occurs with probability $1/2$, the proof of the claim follows. □

Reconsider now the partition $V_1, V_2, \ldots, V_k$, $k \leqslant \lceil n/\Delta \rceil$ of V. Let $\mathrm{part}(v)$ be the function which assigns a vertex v the index of its partition. Let $\mathcal{B}$ be the event that

$$\forall v \in V : V_{\mathrm{part}(v)} \subseteq I_{\mathsf{PAR}(s,v)+\mathcal{O}\left(\frac{n}{\Delta}\log^2 n\right)}$$

occurs. By Lemma 5.49 and the union bound over all n vertices it follows that $\mathbf{Pr}[\mathcal{B}] \geqslant 1 - n^{-2}$. Finally, let $\mathcal{C}$ be the event that every vertex $u \in V(G)$ sends the rumor to $(4/n) \cdot \mathsf{BLA}_s(G)$ vertices within the time-interval

$$\left[\mathsf{SEQ}(s,u), \mathsf{SEQ}(s,u) + \frac{32}{n} \cdot \mathsf{BLA}_s(G) \right).$$

Using a Chernoff bound for Bernoulli variables and the union bound over all vertices $u \in V(G)$, we conclude that $\mathbf{Pr}[\mathcal{C}] \geqslant 1 - n^{-2}$. We claim that if the events $\mathcal{A}, \mathcal{B}$ and $\mathcal{C}$ occur, the vertex s spreads the rumor to all other vertices in the graph within at most

$$\frac{2n}{\Delta} \cdot \left(\mathcal{O}\left(\frac{n}{\Delta}\log^2 n\right) + 32 \cdot \frac{\mathsf{BLA}_s(G)}{n} \right)$$

steps.

Let Ω_1 be the probability space of all possible $(N_{t,v})_{t\in\mathbb{N},v\in V}$. Recall that a random walk can be represented by some $\omega_1 \in \Omega_1$ as described at the beginning of this subsection. Moreover, let Ω_2 be the probability space of all $(S_t)_{t\in\mathbb{N}}$. Then $\Omega_1 \times \Omega_2$ is the set of all instances of the sequential push algorithm. We use a coupling between the random walk and the sequential push algorithm by fixing some $\omega_1 \in \Omega_1$ and regarding ω_2 as a random variable chosen uniformly from Ω_2. We note the following fundamental property of this coupling. If the random walk moves after the k-th visit of v to vertex u, then u sends the rumor to v at step

$$\min\Big\{t' \in \mathbb{T} : |\{t \in \mathbb{T} \mid \mathsf{SEQ}(s,u) \leqslant t \leqslant t', S_t = u\}| \geqslant k\Big\}. \tag{5.12}$$

Consider now an arbitrary vertex u. Recall that the event $\mathcal{A}$ implies the existence of a path $\mathcal{P} = (V_{\mathrm{part}(s)}, V_1, V_2, \ldots, V_{\mathrm{part}(u)})$ in G'. As G' contains at most $2n/\Delta$ vertices, there is such a path $\mathcal{P}$ of length $|\mathcal{P}| \leqslant 2n/\Delta$. Starting from vertex s at step 0, event $\mathcal{B}$ guarantees that the whole partition $V_{\mathrm{part}(s)}$ becomes informed at step $\mathcal{O}((n/\Delta)\cdot\log^2 n)$. As $V_{\mathrm{part}(s)}$ and V_1 are connected in G', there is a vertex $u \in V_{\mathrm{part}(s)}$ with $N_{t,u} \in V_1$ for some $t \leqslant (4/n)\cdot\mathsf{BLA}_s(G)$. Since $\mathcal{C}$ occurs, the vertex u sends the rumor to some vertex in V_1 after $(32/n)\cdot\mathsf{BLA}_s(G)$ steps. With the same arguments as before, partition V_1 becomes completely informed after further $\mathcal{O}((n/\Delta)\cdot\log^2 n)$ time-steps and so on. After $|\mathcal{P}|$ steps we reach the partition $V_{\mathrm{part}(u)}$ and spending another $\mathcal{O}((n/\Delta)\cdot\log^2 n)$ time-steps, the vertex u becomes informed. To summarize, we have shown that if the events $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ simultaneously occur, the fixed vertex u becomes informed after

$$|\mathcal{P}|\cdot\left(\mathcal{O}\Big(\frac{\mathsf{BLA}_s(G)}{n}\Big)+\mathcal{O}\big(\frac{n}{\Delta}\cdot\log^2 n\big)\right)=\mathcal{O}\left(\frac{1}{\Delta}\cdot\mathsf{BLA}_s(G)+\frac{n^2}{\Delta^2}\cdot\log^2 n\right)$$

steps. To finish the proof, we apply the union bound to get

$$\mathbf{Pr}\,[\,\mathcal{A}\cap\mathcal{B}\cap\mathcal{C}\,]\geqslant 1-\frac{1}{2}-n^{-2}-n^{-2}.$$

So, with probability larger than $1/3$, all vertices of G become informed after at most $\mathcal{O}\left(\frac{1}{\Delta}\cdot\mathsf{BLA}_s(G)+\frac{n^2}{\Delta^2}\cdot\log^2 n\right)$ steps. Using the expectation of the geometric distribution, the theorem follows. □

Let us simplify the bound of Theorem 5.48.

Corollary 5.50. *For any Δ-regular graph G,*

$$\mathsf{COV}(G)=\Omega\left(\frac{\Delta^2}{n}\cdot\frac{1}{\log n}\cdot\mathbf{E}\,[\,\mathsf{PAR}(G)\,]\right).$$

Proof. Rearranging Theorem 5.48 yields

$$\mathsf{BLA}(G)=\Omega\left(\Delta\cdot\mathbf{E}\,[\,\mathsf{PAR}(G)\,]-\frac{n^2}{\Delta}\cdot\log^2 n\right).$$

Substituting Theorem 5.45 into the inequality above gives

$$\mathsf{COV}(G) = \Omega\left(\frac{1}{(\log\log n)^2} \cdot \left(\Delta \cdot \mathbf{E}\left[\,\mathsf{PAR}(G)\,\right] - \frac{n^2}{\Delta} \cdot \log^2 n\right)\right).$$

If $\mathbf{E}\left[\,\mathsf{PAR}(G)\,\right] \geqslant 2 \cdot \frac{n^2}{\Delta^2} \cdot \log^2 n$,

$$\mathsf{COV}(G) = \Omega\left(\frac{1}{(\log\log n)^2} \cdot \Delta \cdot \mathbf{E}\left[\,\mathsf{PAR}(G)\,\right]\right) = \Omega\left(\frac{1}{(\log\log n)^2} \cdot \frac{\Delta^2}{n} \cdot \mathsf{PAR}_{n^{-1}}(G)\right).$$

Otherwise, if $\mathbf{E}\left[\,\mathsf{PAR}(G)\,\right] < 2 \cdot \frac{n^2}{\Delta^2} \cdot \log^2 n$, then $n > \frac{\Delta^2}{2n\log n} \cdot \mathbf{E}\left[\,\mathsf{PAR}(G)\,\right]$ and thus

$$\mathsf{COV}(G) = \Omega(n \cdot \log n) = \Omega\left(\frac{\Delta^2}{n} \cdot \frac{1}{\log n} \cdot \mathbf{E}\left[\,\mathsf{PAR}(G)\,\right]\right),$$

and the claim follows. □

5.5.3 Constructions

We complement the bounds of the previous subsection by some graph constructions. To upper bound the cover time of them, we use the well-known connection between (electrical) flows and commute times.

In the following, we shall assume that the vertices of G are numbered $1, 2, \ldots, n$.

Definition 5.51 (One Source-One Sink Unit-Flow Problem, [CRR+97]). *Given a graph $G = (V, E)$ and two vertices $s, t \in V$, a function $f : V \times V \to \mathbb{R}$ is a* unit-flow from s to t *if*

1. *f is antisymmetric, i. e., $f(u, v) = -f(v, u)$,*
2. *$f(u, v) = 0$ if $\{u, v\} \notin E$,*
3. *$\sum_{i=1}^{n} f(s, i) = 1$ and $\sum_{i=1}^{n} f(t, i) = -1$.*

The cost *(or power) of a flow f is defined by $P(f) := \sum_{e \in E(G)} f(e)^2$.*

Theorem 5.52 (Energy-Minimization-Principle, [CRR+97]). *For any unit-flow f from s to t, $\mathsf{R}(s, t) \leqslant P(f)$. Moreover, there is a flow $f_{\min}$ satisfying $\mathsf{R}(s, t) = P(f_{\min})$.*

In the language of electrical networks, this theorem basically says that the electric current flow (corresponding to $\mathsf{R}(s, t)$) is distributed such that the total energy consumption in the network is minimized.

Definition 5.53. *Let $\mathsf{Har}(n, k)$ be the graph $G = (V, E)$ defined as*

$$V(\mathsf{Har}(n, k)) := \{0, 1, \ldots, n-1\},$$
$$E(\mathsf{Har}(n, k)) := \big\{\{i, (i + j) \bmod n\} \mid 0 \leqslant i \leqslant n-1, 1 \leqslant j \leqslant k\big\}.$$

It follows immediately that $\mathsf{Har}(n,k)$ is a $2k$-regular, vertex-transitive graph. Notice that $\mathsf{Har}(n,k)$ is the same as the $(n,2k)$-Harary graph [Har62]. With a slight abuse of notation, we will call $\mathsf{Har}(n,k)$ Harary graph, though it does not match the definition of [Har62]. $\mathsf{Har}(n,1)$ gives the n-cycle and $\mathsf{Har}(n,k)), k \geqslant \frac{n}{2}$ yields the complete graph K_n.

Observation 5.54. *For two vertices $s < t \in V(\mathsf{Har}(n,k))$, $\operatorname{dist}(s,t) \geqslant \lfloor \frac{\min\{t-s, n-(t-s)\}}{k} \rfloor$. In particular, the diameter of $\mathsf{Har}(n,k)$ is at least $\frac{1}{2}\lfloor \frac{n}{k} \rfloor$.*

Proposition 5.55. *Consider a pair of vertices $s < t$ of $V(\mathsf{Har}(n,k))$. Then*

$$\mathsf{C}(s,t) = \mathcal{O}\left(n + \frac{n \cdot \min\{t-s, n-(t-s)\}}{k^2}\right).$$

In particular, $\max_{s,t} \mathsf{C}(s,t) = \mathcal{O}\left(n + \frac{n^2}{k^2}\right)$.

Proof. Let s and t be two vertices. Since $\mathsf{Har}(n,k)$ is vertex-transitive, we may set $s = 0$. Additionally, we assume that $t < n - t$. Consider first the case where $t \leqslant k$. This implies that s and t are both connected to every vertex l with $0 \leqslant l \leqslant k, l \notin \{s,t\}$. For such an l, we define

$$f(s,l) := \frac{1}{k-1} \text{ and } f(l,t) := \frac{1}{k-1}$$

and for all other pairs $p, q \in \{0,1,\ldots,n-1\}$, we set $f(p,q) = 0$. Since f has to be anti-symmetric, we have implicitly defined $f(l,s)$ and $f(t,l)$. Our claim is that f is a unit-flow from vertex $s = 0$ to t. Notice that $\sum_{0 \leqslant l \leqslant n-1} f(s,l) = (k-1) \cdot \frac{1}{k-1} = 1$. With the same arguments, $\sum_{0 \leqslant l \leqslant n-1} f(l,t) = 1$ and $\sum_{0 \leqslant l \leqslant n-1} f(l,l') = 0$ for any $l' \in V(\mathsf{Har}(n,k)) \setminus \{s,t\}$. This shows that f is a unit-flow from s to t. The cost of the flow f is

$$P(f) = \sum_{\{i,j\} \in E} f(i,j)^2 = \sum_{0 < l < k, l \neq t} \left(f(s,l)^2 + f(l,t)^2\right) \leqslant (k-1) \cdot \frac{2}{(k-1)^2} = \frac{2}{k-1}.$$

We proceed by considering the more general case $t > k$. Let $\alpha := \lceil \frac{k}{2} \rceil$ and $\gamma := \lfloor \frac{t-1}{\alpha} \rfloor$. We divide the vertices $0 < l < t$ into γ levels $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_\gamma$ defined by

$$\mathcal{L}_i := \left\{ l : \; 1 + (i-1) \cdot \left\lceil \frac{k}{2} \right\rceil \leqslant l < 1 + i \cdot \left\lceil \frac{k}{2} \right\rceil \right\}.$$

Hence, $|\mathcal{L}_i| = \alpha = \lceil \frac{k}{2} \rceil$. Note that by the definition of $\mathsf{Har}(n,k)$, every vertex in $\mathcal{L}_1$ is connected to s and every vertex in $\mathcal{L}_\gamma$ is connected to t. Moreover, for each $1 \leqslant i < \gamma$, every vertex in $\mathcal{L}_i$ is connected to all vertices in $\mathcal{L}_{i+1}$, since $(i+1) \cdot \lceil \frac{k}{2} \rceil - (1 + (i-1) \cdot \lceil \frac{k}{2} \rceil) = 2 \cdot \lceil \frac{k}{2} \rceil - 1 \leqslant k$. Let us define the flow f. For two vertices $p \in \mathcal{L}_i, 1 \leqslant i < \gamma$ and $q \in \mathcal{L}_{i+1}$ we define $f(p,q) := \frac{1}{\alpha^2}$. For the source s and some vertex $l \in \mathcal{L}_1$ we define $f(s,l) := \frac{1}{\alpha}$, and similarly for t and some $l \in \mathcal{L}_\gamma$ we set $f(l,t) := \frac{1}{\alpha}$. As before it follows that f is a unit-flow

from s to t. Let us compute the cost of f,

$$\begin{aligned}
P(f) &= \sum_{\{i,j\}\in E} f(i,j)^2 \\
&= \sum_{l\in\mathcal{L}_1} f(s,l)^2 + \sum_{i=1}^{\gamma-1} \sum_{p\in\mathcal{L}_i, q\in\mathcal{L}_{i+1}} f(p,q)^2 + \sum_{l\in\mathcal{L}_\gamma} f(l,t)^2 \\
&= \sum_{l\in\mathcal{L}_1} \left(\frac{1}{\alpha}\right)^2 + \sum_{i=1}^{\gamma-1} \alpha^2 \cdot \left(\frac{1}{\alpha^2}\right)^2 + \sum_{l\in\mathcal{L}_\gamma} \left(\frac{1}{\alpha}\right)^2 \\
&= \alpha \cdot \left(\frac{1}{\alpha}\right)^2 + \sum_{i=1}^{\gamma-1} \left(\frac{1}{\alpha}\right)^2 + \alpha \cdot \left(\frac{1}{\alpha}\right)^2 \\
&\leqslant \frac{2}{\alpha} + \left(\frac{2t}{k} - 2\right) \cdot \frac{4}{k^2} \leqslant \frac{2}{k} + \frac{8t}{k^3}.
\end{aligned}$$

To summarize, in both cases for t we have shown that $P(f) = \mathcal{O}\left(\frac{1}{k} + \frac{t}{k^3}\right)$. Combining the former equality with Theorem 5.52 yields

$$\mathsf{C}(s,t) = 2 \cdot |E| \cdot \mathsf{R}(s,t) \leqslant 2 \cdot nk \cdot P(f) = 2 \cdot nk \cdot \mathcal{O}\left(\frac{1}{k} + \frac{t}{k^3}\right) = \mathcal{O}\left(n + \frac{n \cdot t}{k^2}\right).$$

So far, we have been working under the assumption $t \leqslant n - t$. However, the general case follows by observing that $i \mapsto (-i \bmod n), i \in \{0, \ldots, n-1\}$ is an automorphism of $\mathsf{Har}(n,k)$. □

Corollary 5.56. *For every $n^{1/2} \leqslant \Delta \leqslant n$, there are Δ-regular graphs such that*

$$\mathsf{COV}(G) = \mathcal{O}(\Delta \cdot \log n \cdot \mathsf{PAR}_{n^{-1}}(G)).$$

Proof. Consider the graph $\mathsf{Har}(n, \Delta/2)$. From Proposition 5.55 it follows that

$$\mathsf{COV}(\mathsf{Har}(n,\Delta/2)) = \mathcal{O}\left(\max_{s,t\in V} \mathsf{C}(s,t) \cdot \log n\right) = \mathcal{O}\left(\left(n + \frac{n^2}{\Delta^2}\right) \cdot \log n\right) = \mathcal{O}(n \log n),$$

since $\Delta \geqslant n^{1/2}$, and by Observation 5.54, $\mathsf{PAR}_{n^{-1}}(\mathsf{Har}(n,\Delta/2)) \geqslant \operatorname{diam}(\mathsf{Har}(n,\Delta/2)) = \Omega(n/\Delta)$. □

We now move to the case where $\Delta(G) \leqslant n^{1/2}$ and construct the following graph.

Definition 5.57. *For integers n and k such that $\sqrt{n}$ is an integer and $2k$ divides $\sqrt{n}$, we define a graph $\mathsf{G}(n,k)$ as follows. The vertex set is given by*

$$V := \left\{(x,y,z) \mid 0 \leqslant x,y \leqslant k-1, 0 \leqslant z \leqslant \frac{n}{k^2} - 1\right\},$$

and the edge set is defined by

$$E := \Bigg\{ \Big\{ \{(x,y,z),(x,y,z')\} \;\Big|\; 0 \leqslant x,y \leqslant k-1, \{z,z'\} \in E\Big(\mathsf{Har}\big(\frac{n}{k^2},\frac{\sqrt{n}}{k}\big)\Big) \Big\} \tag{5.13}$$

$$\bigcup \Big\{ \{(x,y,z),(x,y-1,\frac{n}{2k^2}-z)\} \;\Big|\; 0 \leqslant x,y \leqslant k-1, 0 \leqslant z < \frac{\sqrt{n}}{k} \Big\} \tag{5.14}$$

$$\bigcup \Big\{ \{(x,y,z),(x-1,y,\frac{n}{2k^2}-z)\} \;\Big|\; 0 \leqslant x,y \leqslant k-1, \frac{3}{4}\frac{n}{k^2} \leqslant z < \frac{3}{4}\frac{n}{k^2}+\frac{\sqrt{n}}{k} \Big\} \Bigg\}, \tag{5.15}$$

where the first two components are meant to be modulo k and the third one is meant to be modulo $n/(k^2)$.

So, E consists of one set of edges 5.13 called Harary edges (as they are induced by $\mathsf{Har}(n/k^2,\sqrt{n}/k)$ and two sets of edges 5.14, 5.15 called torus edges. Moreover, by fixing the first two coordinates we obtain a Harary subgraph $\mathsf{Har}(n/k^2,\sqrt{n}/k)$ of $\mathsf{G}(n,k)$. A sketch of the graph $\mathsf{G}(n,k)$ can be found in Figure 5.5 at page 90.

We remark that the graph $\mathsf{G}(n,k)$ is similar to, but *not* the same as the Cartesian product of a two-dimensional torus $\mathsf{T}_{\mathsf{k,k}}$ and a $\mathsf{Har}(n/k^2,\sqrt{n}/k)$-graph. The major differences are revealed in the following observation.

Observation 5.58. *The graph $\mathsf{G}(n,k)$ as defined above has the following properties,*

1. *$\mathsf{G}(n,k)$ is an n-vertex graph with minimum degree $2\frac{\sqrt{n}}{k}$ and maximum degree $2\frac{\sqrt{n}}{k}+1$,*
2. *the diameter of $\mathsf{G}(n,k)$ is $\Omega(\sqrt{n})$.*

Before we upper bound the cover time of $\mathsf{G}(n,k)$, we give a natural extension of 5.52 to flows not necessarily being unit-flows with one sink and one source.

Definition 5.59 (Generalized Flow Problem). *Given a graph $G=(V,E)$ and a vector $\mathbf{b} \in \mathbb{R}^n$ with $\sum_{i=1}^n b_i = 0$, a function $f = f_{\mathbf{b}} : V \times V \to \mathbb{R}$ is a* flow *if*

1. *f is antisymmetric, i. e., $f(u,v) = -f(v,u)$,*
2. *$f(u,v) = 0$ if $\{u,v\} \notin E$,*
3. *$\sum_{j=1}^n f(i,j) = b_i$ for every $i \in \{1,\dots,n\}$.*

The cost of such a flow f is defined as in Definition 5.51 and the *total flow amount* is $\frac{1}{2}\|b\|_1$. In correspondence to Definition 5.51, a flow problem with a vector $\mathbf{b} = \mathbf{b}_{s,t} \in \{-1,0,1\}^n$ with $b_s = 1, b_t = -1$ and zero otherwise, is called a (s,t)-unit-flow problem.

Note that vertices with $b_i > 0$ can be viewed as sources which send some flow amount to the network, while vertices with $b_i < 0$ represent sinks which consume some flow amount.

The following lemma asserts that if we take the (s,t)-unit-flow problem with the highest cost, then these cost are an upper bound for any (general) flow with a total flow amount of 1. Moreover, the cost of a flow scale quadratically with the total flow amount.

Lemma 5.60. *Assume that $C > 0$ is some real value such that for any $(i,j), i \neq j$ unit-flow problem, there are flows $f_{i,j}$ satisfying $P(f_{i,j}) \leqslant C$. Then, for any vector $\mathbf{b} \in \mathbb{R}^n$ there is a flow $f_{\mathbf{b}}$ satisfying $P(f_{\mathbf{b}}) \leqslant C \cdot \frac{1}{4} \cdot \|\mathbf{b}\|_1^2$.*

Proof. By [DFM99], the minimal solution (w. r. t. $P(f)$) of the generalized flow problem is induced by one (of the infinitely many) vectors $\mathbf{z} = \mathbf{z}_b$ satisfying $\mathbf{L} \cdot \mathbf{z} = \mathbf{b}$ as follows. Given such a $\mathbf{z} \in \mathbb{R}^n$, we obtain the flow between two vertices u, v by setting $f((u,v)) = z_u - z_v$. Note that $\mathbf{z}$ can be viewed as a potential since the flow f can be computed as the differences from $\mathbf{z}$. Moreover, the cost of a flow f can be computed by means of

$$\sum_{e \in E(G)} f(e)^2 = \|\mathbf{A}^T \mathbf{z}\|_2^2,$$

where $\mathbf{A} \in \{-1, 0, 1\}^{n \times |E|}$ is the node-edge-incidence matrix [DFM99] of G.

Claim. Define $a_{ij} := 2|b_i b_j| / \|\mathbf{b}\|_1$ if $b_i > 0 \wedge b_j < 0$, and otherwise $a_{ij} := 0$. Then $\sum_{1 \leqslant i,j \leqslant n} a_{ij} \mathbf{b}_{ij} = \mathbf{b}$ and $\sum_{1 \leqslant i,j \leqslant n} a_{ij} = \|\mathbf{b}\|_1 / 2$.

Proof. Consider an arbitrary $1 \leqslant l \leqslant n$. For some vector $\mathbf{v}$, let $[\mathbf{v}]_l$ denote the l-th coordinate of $\mathbf{v}$. Then

$$\begin{aligned}
\Big[\sum_{1 \leqslant i,j \leqslant n} a_{ij} \mathbf{b}_{ij} \Big]_l &= \sum_{1 \leqslant i,j \leqslant n} \Big[a_{ij} \mathbf{b}_{ij} \Big]_l = \sum_{1 \leqslant i \leqslant n} a_{il} \Big[\mathbf{b}_{il} \Big]_l + \sum_{1 \leqslant j \leqslant n} a_{lj} \Big[\mathbf{b}_{lj} \Big]_l \\
&= \sum_{1 \leqslant i \leqslant n} a_{il} \cdot (-1) + \sum_{1 \leqslant j \leqslant n} a_{lj} \cdot (+1) \\
&= \sum_{1 \leqslant i \leqslant n:\ b_i > 0 \wedge b_l < 0} \frac{-2 \cdot |b_i \cdot b_l|}{\|\mathbf{b}\|_1} + \sum_{1 \leqslant j \leqslant n:\ b_l > 0 \wedge b_j < 0} \frac{2 \cdot |b_l \cdot b_j|}{\|\mathbf{b}\|_1},
\end{aligned}$$

and as exactly one of the two sums vanishes, $[\sum_{1 \leqslant i,j \leqslant n} a_{ij} \mathbf{b}_{ij}]_l = b_l$, as desired.

We continue to prove the second equation:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} = \sum_{i:b_i > 0} \sum_{j:b_j < 0} \frac{2 \cdot |b_i \cdot b_j|}{\|\mathbf{b}\|_1} = \sum_{i:b_i > 0} \frac{|b_i|}{\|\mathbf{b}\|_1} \sum_{j:b_j < 0} 2 \cdot |b_j| = \sum_{i:b_i > 0}^{n} |b_i| = \frac{\|\mathbf{b}\|_1}{2},$$

which establishes the second formula. □

Having disposed of this technical step, we return to the proof of the lemma. For fixed i, j, let $\mathbf{z}_{i,j} : V \to \mathbb{R}_+$ be the corresponding potential of the minimal unit-flow from i to j on G. Then $\mathbf{z}_{\mathbf{b}} := \sum_{i=1}^{n} a_{i,j} \mathbf{z}_{i,j}$ satisfies

$$\mathbf{L} \cdot \mathbf{z}_{\mathbf{b}} = \mathbf{L} \left(\sum_{1 \leqslant i,j \leqslant n} a_{i,j} \mathbf{z}_{i,j} \right) = \sum_{1 \leqslant i,j \leqslant n} a_{i,j} \cdot \mathbf{L} \mathbf{z}_{i,j} = \sum_{1 \leqslant i,j \leqslant n} a_{i,j} \mathbf{b}_{i,j} = \mathbf{b}.$$

So, $\mathbf{z_b}$ is a solution for the generalized flow problem. The cost for the flow f induced by $\mathbf{z_b}$ can be bounded from above by

$$\begin{aligned}\sum_{e\in E} f(e)^2 &= \|\mathbf{A}^T\mathbf{z_b}\|_2^2 = \left\|\mathbf{A}^T\left(\sum_{1\leqslant i,j\leqslant n} a_{i,j}\mathbf{z}_{i,j}\right)\right\|_2^2 = \left\|\sum_{1\leqslant i,j\leqslant n} a_{i,j}\cdot\mathbf{A}^T\mathbf{z}_{i,j}\right\|_2^2 \\ &\leqslant \left(\sum_{1\leqslant i,j\leqslant n} a_{i,j}\cdot\left\|\mathbf{A}^T\mathbf{z}_{i,j}\right\|_2\right)^2 \leqslant \left(\sum_{1\leqslant i,j\leqslant n} a_{i,j}\cdot\sqrt{C}\right)^2 \\ &= C\cdot\left(\sum_{1\leqslant i,j\leqslant n} a_{i,j}\right)^2 = C\cdot\left(\frac{\|\mathbf{b}\|_1}{2}\right)^2,\end{aligned}$$

where we have used the second formula of the claim. Taking the flow $f_{\mathbf{b}}$ induced by $\mathbf{z_b}$ gives the assertion of the lemma. □

Theorem 5.61. *For the graph $\mathsf{G}(n,k)$ as defined above, $\max_{s,t}\mathsf{C}(s,t) = \mathcal{O}(n\log k)$.*

Proof. We will now prove for every pair of vertices s,t that $\mathsf{C}(s,t) = \mathcal{O}(n\log n)$ by constructing a proper unit-flow from s to t. Basically, the flow of $\mathsf{G}(n,k)$ will imitate the minimal flow on a two-dimensional torus as illustrated in Figure 5.5.

Consider two vertices $s = (x_s, y_s, z_s)$ (source) and $t = (x_t, y_t, z_t)$ (sink) in $\mathsf{G}(n,k)$ and assume w. l. o. g. that $z_s \neq z_t$. Our aim is to construct a unit-flow $f = f_{s,t}$ from s to t with as few costs as possible. Throughout this proof, we denote by π the projection onto the first two coordinates, i. e., $\pi(x,y,z) = (x,y)$. Let f_{T} be a minimum unit-flow from $\pi(s)$ to $\pi(t)$ in a two-dimensional torus graph $\mathsf{T} = \mathsf{T}_{k,k}$ with k^2 vertices. By [CRR+97, Theorem 6.1], this optimal flow satisfies

$$P(f_{\mathsf{T}}) = \mathcal{O}(\log k). \tag{5.16}$$

For two vertices (x_1,y_1) and (x_2,y_2) in $V(\mathsf{T}_{k,k})$ write $(x_1,y_1)\sim(x_2,y_2)$ if they are connected. Define $[(x_1,y_1)]_\sim := \big\{(x_2,y_2)\in V(\mathsf{T}_{k,k}) : (x_1,y_1)\sim(x_2,y_2)\big\}$. For two vertices $(x_1,y_1,z_1),(x_2,y_2,z_2)\in V(\mathsf{G}(n,k))$, write $(x_1,y_1,z_1)\sim(x_2,y_2,z_2)$ if they are connected by a torus edge in $\mathsf{G}(n,k)$. Moreover, define

$$[(x_1,y_1,z_1)]_\sim := \big\{(x_2,y_2,z_2)\in V(\mathsf{G}(n,k)) : (x_1,y_1,z_1)\sim(x_2,y_2,z_2)\big\}.$$

Note that $[(x_1,y_1,z_1)]_\sim$ has cardinality one or zero. For any pair $(x_1,y_1,z_1)\sim(x_2,y_2,z_2)$ in $\mathsf{G}(n,k)$ we define

$$f((x_1,y_1,z_1),(x_2,y_2,z_2)) := \frac{k}{\sqrt{n}}\cdot f_{\mathsf{T}}((x_1,y_1),(x_2,y_2)) \tag{5.17}$$

It remains to extend the construction of f to the Harary edges (5.13) of $\mathsf{G}(n,k)$. We will first describe the extension to $\mathsf{Har}(n/k^2,\sqrt{n}/k)$-subgraphs that neither contain s nor t.

So, let (x,y) be different from $\pi(s)$ and $\pi(t)$, and consider the subgraph $(x,y)\times$

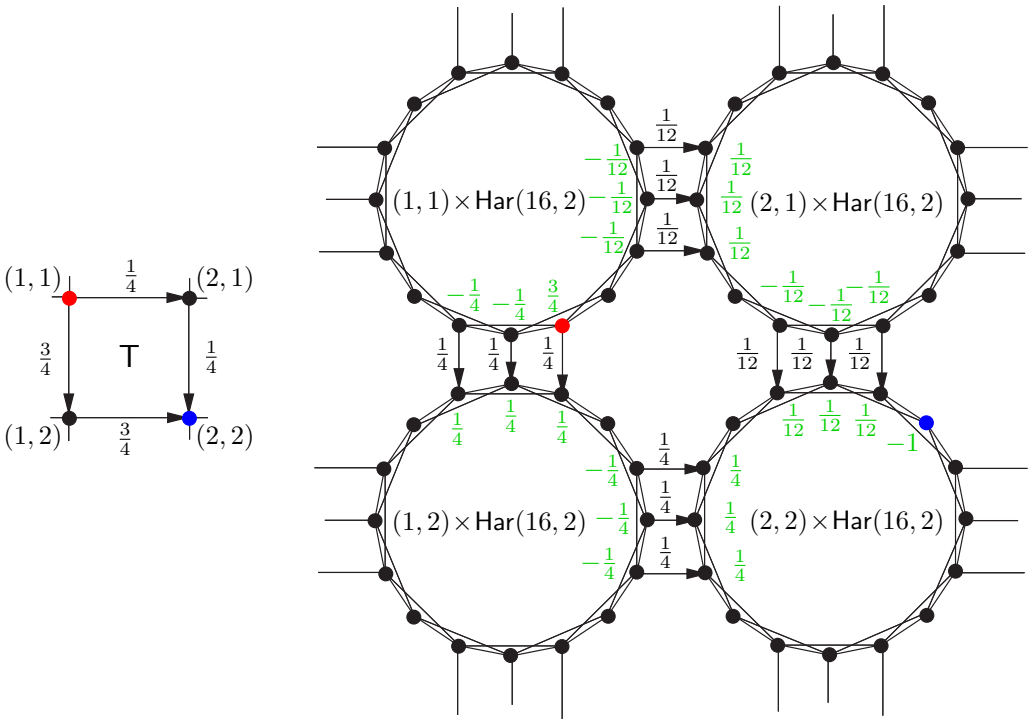

Fig. 5.5: *A flow on a two-dimensional torus graph* T *on the left side and its extension to a flow on* $\mathsf{G}(n,k)$ *on the right side. The green numbers represent the definition of the corresponding* $\mathbf{b}(x,y)$*. The red and blue vertices represent the source and the sink of the unit-flow-problem, respectively. For the sake of readability, the corresponding flow is only described for torus edges on* $\mathsf{G}(n,k)$*.*

$\mathsf{Har}(n/k^2, \sqrt{n}/k)$. Define

$$b(x,y)_z := - \sum_{(x',y',z') \in [(x,y,z)]_\sim} f((x,y,z),(x',y',z')), \tag{5.18}$$

and recall that the sum ranges over at most one summand, as every vertex is incident to at most one torus edge. For this reason,

$$\begin{aligned}
\sum_{0 \leqslant z < \frac{n}{k^2}} b(x,y)_z &= \sum_{0 \leqslant z < \frac{n}{k^2}} \left(- \sum_{(x',y',z') \in [(x,y,z)]_\sim} f((x,y,z),(x',y',z')) \right) \\
&= - \sum_{0 \leqslant z < \frac{n}{k^2}} \sum_{(x',y',z') \in [(x,y,z)]_\sim} \frac{k}{\sqrt{n}} \cdot f_{\mathsf{T}}((x',y'),(x,y)) \\
&= - \frac{k}{\sqrt{n}} \cdot \frac{\sqrt{n}}{k} \cdot \sum_{(x',y') \in [(x,y)]_\sim} f_{\mathsf{T}}((x',y'),(x,y)),
\end{aligned}$$

and this results in $\sum_{0 \leqslant z < \frac{n}{k^2}} b(x,y)_z = 0$ due to $\sum_{(x',y') \in [(x,y)]_\sim} f_{\mathsf{T}}((x',y'),(x,y)) = 0$, and gives $\sum_{0 \leqslant z < \frac{n}{k^2}} |b(x,y)_z| = \sum_{(x',y') \in [(x,y)]_\sim} |f_{\mathsf{T}}((x,y),(x',y'))|$. For such an $(x,y) \notin \{\pi(s), \pi(t)\}$, the vector $\mathbf{b}(x,y)$ induces a general flow problem with flow amount $\frac{1}{2} \cdot \sum_{(x',y') \in [(x,y)]_\sim} |f_{\mathsf{T}}((x,y),(x',y'))|$. By Proposition 5.55, every unit-flow-problem on the graph $\mathsf{Har}(n,k')$ has a solution f' with $P(f') \leqslant \frac{2}{k'} + \frac{8n}{k'^3}$. Substituting, we see that every unit-flow-problem on the graph $\mathsf{Har}(n/k^2, \sqrt{n}/k)$ has a solution f'' with

$$P(f'') \leqslant \frac{2k}{\sqrt{n}} + \frac{8n/k^2}{n^{3/2}/k^3} = 10\frac{k}{\sqrt{n}}.$$

Applying Lemma 5.60 on $\mathsf{Har}(n/k^2, \sqrt{n}/k)$, we conclude that there is a solution $f_{\mathbf{b}(x,y)}$ to the general flow problem given by the vector $\mathbf{b}(x,y)$ such that

$$P(f_{\mathbf{b}(x,y)}) \leqslant 10\frac{k}{\sqrt{n}} \cdot \|\mathbf{b}(x,y)\|_1^2.$$

It remains to specify f on the Harary subgraphs containing s or t. Begin with the subgraph $(x_s, y_s) \times \mathsf{Har}(n/k^2, \sqrt{n}/k)$. We define a vector $\mathbf{b}(x_s, y_s)$ by

$$b(x_s,y_s)_z := \begin{cases} -\sum_{(x',y',z') \in [x_s,y_s,z]_\sim} f((x_s,y_s,z),(x',y',z')) & \text{if } z \neq z_s, \\ 1 - \sum_{(x',y',z') \in [x_s,y_s,z_s]_\sim} f((x_s,y_s,z_s),(x',y',z')) & \text{otherwise.} \end{cases}$$

It follows that $\sum_{0 \leqslant z < (n/k^2)} b(x_s,y_s)_z = 0$ since $\sum_{(x',y') \in [(x_s,y_s)]_\sim} f_{\mathsf{T}}((x_s,y_s),(x',y')) = 1$. Applying Lemma 5.60 on $\mathsf{Har}(n/k^2, \sqrt{n}/k)$, we conclude that there is a solution $f_{\mathbf{b}(x_s,y_s)}$ to the general flow problem given by the vector $\mathbf{b}(x_s,y_s)$ such that

$$P(f_{\mathbf{b}(x_s,y_s)}) \leqslant 10\frac{k}{\sqrt{n}} \cdot \|\mathbf{b}(x_s,y_s)\|_1^2.$$

For the subgraph $(x_t, y_t) \times \mathsf{Har}(n/k^2, \sqrt{n}/k)$ we give the analogue definition to $b(x_s,y_s)_z$.

$$b(x_t,y_t)_z := \begin{cases} -\sum_{(x',y',z') \in [x_t,y_t,z]_\sim} f((x_t,y_t,z),(x',y',z')) & \text{if } z \neq z_t, \\ -1 - \sum_{(x',y',z') \in [x_s,y_s,z_s]_\sim} f((x_t,y_t,z_t),(x',y',z')) & \text{otherwise.} \end{cases}$$

Also here we have $\sum_{0 \leqslant z < (n/k^2)} b(x_t,y_t) = 0$. Applying Lemma 5.60 on $\mathsf{Har}(n/k^2, \sqrt{n}/k)$ we conclude that there is a solution $f_{\mathbf{b}(x_t,y_t)}$ to the general flow problem given by the vector $\mathbf{b}(x_t,y_t)$ such that

$$P(f_{\mathbf{b}(x_t,y_t)}) \leqslant 10\frac{k}{\sqrt{n}} \cdot \|\mathbf{b}(x_t,y_t)\|_1^2.$$

Combining these solutions for each (x,y) with the definition of f on the torus edges, we obtain a unit-flow f from $s = (x_s, y_s, z_s)$ to $t = (x_t, y_t, z_t)$. It remains to compute the cost

of the flow $P(f)$ which is done by the following decomposition,

$$P(f) = \sum_{e \in E(\mathsf{G}(n,k))} f(e)^2 \;=\; \underbrace{\sum_{\substack{e \in E(\mathsf{G}(n,k)) \\ e \text{ torus edge}}} f(e)^2}_{=A} + \underbrace{\sum_{\substack{e \in E(\mathsf{G}(n,k)) \\ e \text{ Harary edge}}} f(e)^2}_{=B}.$$

We start with A,

$$\begin{aligned} A &= \sum_{\substack{e \in E(\mathsf{G}(n,k)) \\ e \text{ torus edge}}} f(e)^2 = \sum_{\substack{e \in E(\mathsf{G}(n,k)) \\ e \text{ torus edge}}} \left(\frac{k}{\sqrt{n}} \cdot f_{\mathsf{T}}(\pi(e)) \right)^2 && \text{(by 5.17)} \\ &= \frac{k^2}{n} \cdot \sum_{\substack{e \in E(\mathsf{G}(n,k)) \\ e \text{ torus edge}}} f_{\mathsf{T}}(\pi(e))^2 \\ &= \frac{k^2}{n} \frac{n^{1/2}}{k} \cdot \sum_{e \in \mathsf{T}} f_{\mathsf{T}}(e)^2 \\ &= \frac{k}{\sqrt{n}} \cdot P(f_{\mathsf{T}}) = \frac{k}{\sqrt{n}} \cdot \mathcal{O}(\log k) && \text{(by 5.16)} \end{aligned}$$

Before we look at expression B, we note that by the convexity of $x \mapsto x^2$, $(x_1 + x_2 + x_3 + x_4)^2 \leqslant 4x_1^2 + 4x_2^2 + 4x_3^2 + 4x_4^2$ follows for every $x_1, x_2, x_3, x_4 \geqslant 0$. For this reason

$$\begin{aligned} B &= \sum_{\substack{e \in E(\mathsf{G}(n,k)) \\ e \text{ Harary edge}}} f(e)^2 = \sum_{0 \leqslant x,y \leqslant k} P(f_{\mathbf{b}(x,y)}) \\ &\leqslant \sum_{0 \leqslant x,y \leqslant k} 10 \frac{k}{\sqrt{n}} \cdot \|\mathbf{b}(x,y)\|_1^2 \\ &= 10 \frac{k}{\sqrt{n}} \cdot \sum_{0 \leqslant x,y \leqslant k} \left(\sum_{(x',y') \in [(x,y)]_\sim} |f_{\mathsf{T}}((x,y),(x',y'))| \right)^2 \\ &\leqslant 10 \frac{k}{\sqrt{n}} \cdot \sum_{0 \leqslant x,y \leqslant k} \sum_{(x',y') \in [(x,y)]_\sim} 4 \cdot f_{\mathsf{T}}((x,y),(x',y'))^2 \\ &\leqslant 160 \frac{k}{\sqrt{n}} \cdot \sum_{e \in \mathsf{T}} f_{\mathsf{T}}(e)^2 \\ &= 160 \frac{k}{\sqrt{n}} \cdot \mathcal{O}(\log k) && \text{(by 5.16).} \end{aligned}$$

To conclude the proof, we have shown that for every pair $s, t \in V(\mathsf{G}(n,k))$ there is a flow

$f = f_{s,t}$ on $\mathsf{G}(n,k)$ such that

$$P(f_{s,t}) \leqslant A + B \leqslant \frac{k}{\sqrt{n}} \cdot \mathcal{O}(\log k) + 160\frac{k}{\sqrt{n}} \cdot \mathcal{O}(\log k) = \mathcal{O}\left(\frac{k}{\sqrt{n}} \log k\right).$$

Finally, we apply Theorem 5.52 to obtain

$$\max_{s,t\in V} \mathsf{C}(s,t) = \max_{s,t\in V} \{2 \cdot |E| \cdot \mathsf{R}(s,t)\} \leqslant \max_{s,t\in V} \left\{\mathcal{O}\left(n \cdot \frac{\sqrt{n}}{k}\right) \cdot P(f(s,t))\right\} = \mathcal{O}(n \log k),$$

and the theorem follows. □

Returning to the gap between cover and broadcast time, we can now show the following result.

Corollary 5.62. *For any two integers n, k such that $\sqrt{n}$ is an integer and $2k$ divides $\sqrt{n}$,*

$$\mathsf{COV}(\mathsf{G}(n,k)) = \mathcal{O}\big(\mathsf{PAR}_{n^{-1}}(\mathsf{G}(n,k)) \cdot \sqrt{n} \cdot \log^2 n\big).$$

Proof. By the previous theorem, $\mathsf{COV}(\mathsf{G}(n,k)) = \mathcal{O}(n \log n \cdot \log k)$ and $\mathsf{PAR}_{n^{-1}}(\mathsf{G}(n,k)) \geqslant \operatorname{diam}(\mathsf{G}(n,k)) = \Omega(\sqrt{n})$. □

5.5.4 Minimum Gap between Cover Time and Diameter

We believe that the examples constructed in the preceding section are rather tight when $\deg(G) \in [\omega(1), o(n)]$. Conversely, we think that the bounds are probably rather loose in this case. The reason for this belief is that the examples are tight up to logarithmic factors if we consider the minimum possible gap between the cover time and the diameter (cf. Figure 5.2).

Proposition 5.63. *Let $G = (V, E)$ be a Δ-regular graph. Then*

$$\mathsf{COV}(G) \geqslant \operatorname{diam}(G) \cdot \begin{cases} \sqrt{n \log n} & \text{if } \Delta \leqslant n^{1/2}, \\ \Delta \cdot \log n & \text{if } \Delta \geqslant n^{1/2}. \end{cases}$$

Proof. We first consider the case where $\Delta \leqslant n^{1/2}$. Recall that $\mathsf{COV}(G) = \Omega(n \log n + \operatorname{diam}^2(G))$. If now $\operatorname{diam}(G) \leqslant \sqrt{n \log n}$, then we obtain

$$\mathsf{COV}(G) = \Omega(\sqrt{n \log n} \cdot \sqrt{n \log n}) = \Omega(\operatorname{diam}(G) \cdot \sqrt{n \log n}).$$

Otherwise we have $\operatorname{diam}(G) \geqslant \sqrt{n \log n}$ which results in

$$\mathsf{COV}(G) = \Omega(\operatorname{diam}^2(G)) \geqslant \Omega(\operatorname{diam}(G) \cdot \sqrt{n \log n}).$$

To prove a better bound if $\Delta \geqslant n^{1/2}$, recall that $\operatorname{diam}(G) \leqslant \frac{3n}{\Delta}$ by Lemma 2.28. Hence,

$$\mathsf{COV}(G) = \Omega(n \log n) = \Omega(\Delta \cdot \operatorname{diam}(G) \cdot \log n).$$

□

5.6 Conclusion

Inspired by the intuition of Chandra et al. [CRR+97] about the relationship between cover time of random walks and the runtime of randomized broadcast, we proved a multitude of new bounds relating the cover time not only to the push algorithm but also to other parameters like conductance/expansion and the mixing time. As our main result, we provided a tight characterization of graphs for which the cover time is an appropriate metric for the runtime of the push algorithm. More precisely, we proved that the cover time is captured by the runtime of the push algorithm up to logarithmic factors if G a regular graph of degree $\Omega(n)$. On the negative side, for the class of all regular graphs with some fixed degree $\mathcal{O}(n^{1-\varepsilon}), \varepsilon > 0$, we showed that the push algorithm does not determine the cover time more accurately than up to a factor of n^{ε} (neglecting logarithmic factors). Nevertheless, our results show that the relationship between both processes is rather close and substantially closer than the relationship between one of the processes to the mixing time. In particular, our findings provide evidence for the following (informal) hierarchy:

$$\text{low mixing time} \Rightarrow \text{low broadcast time} \Rightarrow \text{low cover time},$$

which refines the already known

$$\text{low mixing time} \Rightarrow \text{low cover time}$$

relation (cf. [Ald83, Dia88] or Theorem 5.11 taken from [BK89]).

Though the majority of our bounds on the cover time is tight up to some constant or at least to logarithmic factors, many of our results ask for improvement or extension. We single out the following questions.

- The second inequality of Theorem 5.19 is only known to be tight up to constant factors when G is a complete k-ary tree with $k = \mathcal{O}(1)$. We were not able to find any regular graph with a larger degree matching this inequality. So, can we improve Theorem 5.19 for these graphs?

- While the upper bound of Corollary 5.9 depends linearly on the conductance, there is an extra $\log^3 n$-factor in the bound. We conjecture that the bound remains valid without the $\log^3 n$-factor.

- Similarly, Theorem 5.37 provides for every Cayley graph an upper bound on the maximum commute time based on the square root of the mixing time. Can we

prove a similar statement for regular graphs, i. e., an upper bound on the maximum commute time for regular graphs with a sublinear dependence on the mixing time?

- For regular graphs with $\Delta \in [n^{\varepsilon}, n^{1-\varepsilon}]$ for some constant $\varepsilon > 0$, the lower bounds on the cover time are only tight up to polynomial factors. It would be nice to improve or even to tighten these bounds. In particular, is $n^{1/2} \cdot \mathsf{PAR}_{n^{-1}}(G)$ a general lower bound for the cover time on regular graphs (up to logarithmic factors)?

- We showed in the proof of Theorem 5.48 that every regular graph with degree $\Omega(n)$ can be partitioned into a constant number of subgraphs such that the rumor is disseminated within one subgraph after $\mathcal{O}(\log^2 n)$ steps. As a more general question, can dense graphs be partitioned into a constant number of disjoint subgraphs $G_1, G_2, \ldots, G_c$, such that every G_i, $1 \leqslant i \leqslant c$, has minimum-degree $\Omega(n)$ and edge-expansion α for some reasonably large α? Note that if this holds for constant α, then the $\log^2 n$-factor of Theorem 5.48 could be reduced to $\log n$, which implies that any such dense graph with an optimal blanket time of $\mathcal{O}(n \log n)$ would also have an optimal broadcast time of $\mathcal{O}(\log n)$.

6. QUASIRANDOM RUMOR SPREADING

6.1 Introduction

As an introduction of randomized rumor spreading and broadcast was already given in Section 3, we only mention some special aspects of *quasirandomness* in the following.

6.1.1 Motivation

We propose a quasirandom analogue of the randomized push algorithm. The basic setup is as in the randomized model, that is, in each time-step each informed node tries to inform one of its neighbors. However, the choices of these neighbors will not be independently at random. Instead, we assume that each node has a list of its neighbors and informs the neighbors in the order of the list.

It is easily seen that in this model without any randomness a bad choice of the lists can lead to a bad behavior of the protocol. Consider, e. g., the complete graph on n vertices labelled 1 to n and assume that each node informs its neighbors in increasing order. Then it takes $n-1$ time-steps to spread a rumor from node 1 to node n.

To avoid such behavior, we allow a little randomness. When a node receives the rumor for the first time, it chooses a random position on its list. In the sequel, it informs its neighbors starting with this position and then continuing in the order of the list. When the end of the list is reached, it continues at the beginning of the list.

We call this model *quasirandom push model*, as it aims at imitating properties of the classical push model with a much smaller degree of randomness. While the classical push algorithm requires at least $\Omega(n \log n \log \Delta)$ random bits on Δ-regular graphs (and even more if the runtime is larger), our quasirandom model needs only $\mathcal{O}(n \log \Delta)$ random bits.

In our analysis, we adopt a worst-case scenario, that is, we prove bounds for the broadcast times independent of the particular lists. Hence in a practical application, the lists may be chosen to suit internal technical representations of the network.

6.1.2 Related Work

Quasirandomness means that we try to imitate a particular property of a random process deterministically, or with a reduced amount of randomness. This concept occurs in several areas of mathematics and computer science. A prominent example are low-discrepancy point sets and Quasi-Monte Carlo methods (cf. [Nie92]), which proved to be superior over random sample points in numerical integration.

An example closer related to our model is a quasirandom analogue of random walks introduced by [PDDK96] and later popularized by Jim Propp. Here the vertices are equipped with a rotor pointing to a neighbor and a cyclic permutation of the neighbors. A walk arises from leaving the current vertex in the rotor direction and then updating the rotor to the next neighbor according to the order given by the permutation. Some beautiful results exist on this model, e. g., [CS06] showed that if an arbitrary large population of particles does such a quasirandom walk on an infinite grid $\mathbb{Z}^d$, then (under some mild conditions) the number of particles on a vertex at some time deviates from the expected number had the population done a random walk instead, by only a constant c_d. This constant is independent of the number of particles and their initial position. For the case $d = 1$, that is, the graph being the infinite path, the constant c_1 is approximately 2.29 [CDST07]. For the two-dimensional grid the constant is $c_2 \approx 7.87$ [DF06]. It was also shown that for the graph being an infinite k-ary tree ($k \geq 3$), the deviation between both models can be unbounded [CDFS08].

The quasirandomness in our broadcast model lies in the property that a vertex in the long run contacts each of its neighbors approximately equally often, similar to what would have happened in the random push model. In a sense, and this is typical for quasirandomness, we do better in that the deviations are at most one, whereas in the random push model a vertex v after k contacts would have contacted each neighbor only $k/\deg(v) \pm \Theta(\sqrt{k/\deg(v)})$ times.

Since our model aims at saving random bits, it is related to the so-called *probability amplification* problem. Suppose that we are given a randomized algorithm whose outputs are correct with some probability. The question is how to achieve a small error probability by the repeated use of this algorithm while using a minimum possible number of random bits. It was found out that taking (dependent) samples of a certain random walk on constant-degree expanders comes very close to this minimum [CW89, IZ89, MR95].

6.1.3 Our Results

As in the previous work done on the random push model, we analyze how long it takes to spread a rumor from one node of a network to all other nodes. Surprisingly, the greatly reduced degree of randomness does not make broadcast less efficient on the following topologies. For complete graphs, hypercubes and random graphs $\mathcal{G}(n,p)$, $p \geqslant (1+\varepsilon)(\log n)/n$, we obtain a bound of $\mathcal{O}(\log n)$ steps. These bounds hold for all starting vertices and all orders of the lists.

Our $\mathcal{O}(\log n)$ bound also includes sparsely connected random graphs $G \in \mathcal{G}(n,p)$ if $p = (\log(n) + f(n))/n$ with $\lim_{n\to\infty} f(n) \to \infty$. This contrasts with the $\Omega(\log^2 n)$ bound shown by Feige et al. [FPRU90] for the case that $c_n = 1 + \mathcal{O}(\log\log n/\log n)$ and shows a further superiority of our model (in addition to the reduced need of random bits).

For hypercubes and random graphs we prove upper bounds which hold with probability $1 - n^{-C}$ where $C > 0$ is some arbitrary constant. The reason is that due to the lack of independence, an upper bound of x holding with probability, say, $1 - n^{-1}$ does *not* imply an upper bound of $2 \cdot x$ with probability $1 - n^{-2}$.

Graph class	Broadcast times	Reference
General graphs	$\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\Delta(\operatorname{diam}(G) + \log n))$	[FPRU90]
	$\mathsf{QR}_0(G) \leqslant \Delta \cdot \operatorname{diam}(G)$	Thm. 6.2
	$\mathsf{PAR}_{n^{-1}}(G) \leqslant 12n \log n$	[FPRU90]
	$\mathsf{PAR}_{o(1)}(G) \leqslant (1 + o(1))n \ln n$	Thm. 3.9
	$\mathsf{QR}_0(G) \leqslant 2n - 3$	Thm. 6.2
Complete k-ary trees	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(k \log n)$	Cor. 6.3
	$\mathsf{QR}_0(G) = \Theta(k \log n / \log k)$	Cor. 6.3
Hypercubes	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log n)$	[FPRU90]
	$\mathsf{QR}_{n^{-C}}(G) = \Theta(\log n)$	Thm. 6.5
almost all $\mathcal{G}(n,p)$, $p = \frac{\log n + f(n)}{n}$, $f(n) \to \infty$ and $f(n) = \mathcal{O}(\log \log n)$	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log^2 n)$	[FPRU90]
	$\mathsf{QR}_{n^{-C}}(G) = \Theta(\log n)$	Thm. 6.6
almost all $\mathcal{G}(n,p)$, $p \geqslant \frac{(1+\varepsilon)\log n}{n}$, where $\varepsilon > 0$ is a constant	$\mathsf{PAR}_{n^{-1}}(G) = \Theta(\log n)$	[FPRU90]
	$\mathsf{QR}_{n^{-C}}(G) = \Theta(\log n)$	Thm. 6.6

Fig. 6.1: *Broadcasting times of different graphs G in the random (PAR) and quasirandom (QR) push model.*

We also prove tight upper bounds of $\Delta \cdot \operatorname{diam}(G)$ and $2n - 3$ for general graphs, which are again better than the corresponding bounds of [FPRU90] for the random model. Since the bound of $2n - 3$ holds with probability 1, we also obtain upper bounds of $\mathcal{O}(\log n)$ on the *expected* runtime on hypercubes and random graphs as a simple corollary. All bounds at a glance are summarized in Figure 6.1.

6.2 Notations, Definitions and Preliminaries

As in the classical push model we aim at spreading a rumor in an undirected graph $G = (V, E)$. We denote by s the vertex who knows the rumor at the beginning time-step. In our *quasirandom push model* QR, each vertex $v \in V$ is associated with a cyclic permutation $\pi_v \colon N(v) \to N(v)$ of its neighbors (usually simply viewed as list of neighbors). While above we said that a vertex when it first obtains the rumor has chosen a position on the list uniformly at random as starting point for its broadcast campaign, in the analysis the following equivalent model will be advantageous. We assume that initially each vertex has a position on the list chosen uniformly at random, and that it updates this position each time-step even if it is not informed ("ever rolling lists"). More precisely, at the start of the protocol each vertex v chooses an *initially contacted* neighbor i_v uniformly at random from $N(v)$. In each time-step $t = 1, 2, \ldots$, the vertex v sends the rumor to vertex $\pi_v^{t-1}(i_v)$,

if it is informed, and does nothing otherwise. In the first case, $\pi_v^{t-1}(i_v)$ becomes informed (if it is was not already). A supplementary description of the quasirandom push model is given in Figure 6.2.

Quasirandom Push Algorithm

Input: Graph $G = (V, E)$, Cyclic Permutations $\pi_v : N(v) \to N(v)$ for each $v \in V(G)$

1: $t \leftarrow 0$
2: $I_0 \leftarrow \{s\}$
3: **for** all vertices $v \in V$ **do**
4: choose $i_v \in N(v)$ uniformly at random
5: **end for**
6: **while** $I_t \neq V$ **do**
7: $t \leftarrow t + 1$
8: $I_t \leftarrow I_{t-1}$
9: **for** all vertices $v \in V$ **do**
10: **if** $v \in I_{t-1}$ **then**
11: $I_t \leftarrow I_t \cup \pi_v^{t-1}(i_v)$
12: **end if**
13: **end for**
14: **end while**

Fig. 6.2: *The definition of the quasirandom push algorithm*

The focus of our investigation is how long it takes until some rumor known only by a single vertex is broadcasted to all other vertices. We adopt a worst-case view in that we aim at bounds that are independent of the starting vertex s and of all the lists.

Given a graph $G = (V, E)$, the number of iterations (or time-steps) of a broadcast procedure until the rumor reaches all the vertices of G is a random variable that depends on the topology of G. Let $\mathsf{QR}(G) := \min\{t \in \mathbb{N} : \ I_t = V, I_0 = \{s\}\}$ be the runtime of the quasirandom model for some initially informed vertex s. Corresponding to the notation of $\mathsf{PAR}_p(G)$, let $\mathsf{QR}_p(G) := \max_{s \in V} \min\{t \in \mathbb{N} : \ \mathbf{Pr}\,[\, I_t = V \mid I_0 = \{s\}\,] \geqslant 1 - p\}$. Our aim is to bound $\mathsf{QR}(G)$ and to compare it with $\mathsf{PAR}(G)$ for different graph classes.

In the analysis of the quasirandom push model, it will occasionally be convenient to assume that a vertex after receiving the rumor does not transfer it on for a certain number of time-steps (*delayed model*). It is clear that this will only result in other vertices receiving the rumor later. Consequently, the random variable describing the broadcast time of this model is stochastically larger than $\mathsf{QR}(G)$. Of course, this also holds if several vertices delay the propagation of the rumor.

We also require the following definition which allows us to analyze the propagation of the rumor in the reverse order.

Definition 6.1 ([ES05]). *A vertex $u_1 \in V$* contacts *another vertex $u_l \in V$ within the time-interval $[a, b]$, $a < b$, if there is a path $(u_1, u_2, \ldots, u_l)$ in G and $t_1 < t_2 < \cdots < t_{l-1} \in [t_1, t_2]$*

such that for all $j \in [l-1]$, $\pi_{u_j}^{t_j-1}(i_{u_j}) = u_{j+1}$. *We denote by* $C_{a,b}(v)$ *the set of all vertices which contact a vertex* v *within the time-interval* $[a, b]$.

By definition, if u_1 contacts u_l within the time-interval $[a, b]$ and u_1 is informed at step a, then u_l is also informed at step b.

6.3 General Results

In this section, we give two bounds for the broadcast time in general graphs. The corresponding bounds for the random model are $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\Delta \cdot (\operatorname{diam}(G) + \log n))$ [FPRU90] and $\mathsf{PAR}_{o(1)}(G) \leqslant (1 + o(1))n \ln n$ (Theorem 3.9).

Theorem 6.2 ([DFS08]). *For any graph* $G = (V, E)$,

1. $\mathsf{QR}_0(G) \leqslant \Delta \cdot \operatorname{diam}(G)$,
2. $\mathsf{QR}_0(G) \leqslant 2n - 3$.

The first bound is asymptotically tight for every constant-degree graph, as $\Omega(\operatorname{diam}(G) + \log n)$ is an obvious lower bound for every graph. Moreover, a path of length $n-1$ matches the second bound of Theorem 6.2 exactly.

Compared to the general bound $\mathsf{PAR}_{n^{-1}}(G) = \mathcal{O}(\Delta(\operatorname{diam}(G) + \log n))$ [FPRU90], the quasirandom model may save a small factor on graphs with $\operatorname{diam}(G) = o(\log n)$.

Corollary 6.3. *For complete* k*-ary trees,* $\mathbf{E}[\,\mathsf{PAR}(G)\,] = \Theta(k \log n)$ *and* $\mathsf{QR}_0(G) = \Theta(k \frac{\log n}{\log k})$.

Proof. The expected time after the root of a complete k-ary tree has informed all its successors is $(1 \pm o(1))k \cdot \ln k$ by Theorem 2.19. Since every leaf has distance $\log_k n$ to the root, the first bound follows. The second is an immediate consequence of Theorem 6.2. □

6.4 Hypercubes and Random Graphs

Theorem 6.2 gives an upper bound of $\mathsf{QR}_0(\mathsf{Q}_d) = \mathcal{O}(\log^2 n)$. The following result shows that this is the best possible upper bound if we insist on a failure probability of 0. Hence, we have to rely on a small amount of randomness to achieve a broadcast time of $\mathcal{O}(\log n)$ in our model.

Proposition 6.4. *For the hypercube* Q_d *with* $n = 2^d$ *vertices,* $\mathsf{QR}_0(\mathsf{Q}_d) = \Theta(\log^2 n)$.

Proof. We prove that there are lists and initially contacted neighbors for each vertex such that $\Omega(\log^2 n)$ steps are required to inform all vertices independent of the initially informed vertex. For this proof, we drop the ever rolling list assumption from Section 6.2. More precisely, any informed vertex sends the rumor to $\pi_v^{t-\mathsf{QR}(s,v)-1}(i_v)$ in step $t > \mathsf{QR}(s, v)$. For any vertex $u \in \{0, 1\}^d$ and $i \in [1, d]$ let $u(i)$ be the vertex obtained by flipping the i-th bit of u. Then, for every vertex u we set the neighbor list $(u(1), u(2), \ldots, u(d))$ with $i_u = u(1)$. Assume that initially the vertex $s = (s_1, \ldots, s_d)$ owns the rumor. Due to

the construction, an arbitrary vertex v requires k steps to send the rumor to neighbor $v(k)$ and by simple induction we require $\sum_{k=1}^{d} k = \Omega(d^2) = \Omega(\log^2 n)$ steps to inform $\overline{s} = (1 - s_1, \ldots, 1 - s_d)$. □

For the random push model it is known that $\mathsf{PAR}_{n^{-1}}(\mathsf{Q}_d) = \Theta(\log n)$ [FPRU90]. The following theorem extends this result to the quasirandom model.

Theorem 6.5. *For the hypercube Q_d and every constant $C > 0$, $\mathsf{QR}_{n^{-C}}(\mathsf{Q}_d) = \Theta(\log n)$.*

Proof. By symmetry we may assume that $s = 0^d$ knows the rumor at the beginning. Our proof consists of three phases. In the first phase we show that after $\mathcal{O}(d)$ steps a large set of informed vertices I' exists. In the third phase we show that a large set of uninformed vertices $C(w)$ must to exist in order to keep a fixed vertex w at some step $\mathcal{O}(d)$ uninformed. In particular, every vertex of $u \in I'$ will be close to some proper vertex $v = v(u) \in C(w)$. Finally, in the second phase we show that one of the informed vertices in I' informs one close vertex of $C(w)$ with high probability implying that w must also be informed after $\mathcal{O}(d)$ steps. A graphical illustration of our proof can be found in Figure 6.3.

Forward Approximation: (from step 0 till step $4d$) We first show that after $4d$ steps we have $|I_{4d}| \geqslant 4^{d/7}$ with high probability. For some $0 \leqslant i \leqslant d$ define $\mathcal{L}_i := \{x \in \{0,1\}^i : \|x\|_1 = i\}$. Note that after $2d$ steps, $\mathcal{L}_0 \cup \mathcal{L}_1$ has been informed completely.

Fix some time-step t where $|I_t \cap \mathcal{L}_i| \neq \emptyset$. We may assume that $|I_t \cap \mathcal{L}_j| = \emptyset$ for any $j > i$ and that all initially contacted neighbors of $I_t \cap \mathcal{L}_i$ are still to be chosen u. a. r. Notice that the set of edges between $I_t \cap \mathcal{L}_i$ and $\mathcal{L}_{i+1}$ satisfies $|E(I_t \cap \mathcal{L}_i, \mathcal{L}_{i+1})| = \sum_{v \in \mathcal{L}_{i+1}} \deg_{I_t \cap \mathcal{L}_i}(v) = |I_t \cap \mathcal{L}_i| \cdot (d - i)$. Our goal is to lower bound $|I_{t+10} \cap \mathcal{L}_{i+1}|$ by means of $|I_t \cap \mathcal{L}_i|$. The probability that a vertex $v \in \mathcal{L}_{i+1}$ is still uninformed after 10 steps is

$$\mathbf{Pr}\left[\, v \notin I_{t+10} \,\right] \leqslant \prod_{u \in N(v) \cap I_t \cap \mathcal{L}_i} \left(1 - \frac{10}{d}\right) = \left(1 - \frac{10}{d}\right)^{\deg_{I_t \cap \mathcal{L}_i}(v)}.$$

By linearity of expectations we obtain

$$\mathbf{E}\left[\, |I_{t+10} \cap \mathcal{L}_{i+1}| \,\right] \geqslant \sum_{v \in \mathcal{L}_{i+1}} 1 - \left(1 - \frac{10}{d}\right)^{\deg_{I_t \cap \mathcal{L}_i}(v)} \geqslant \sum_{v \in \mathcal{L}_{i+1}} 1 - e^{-\frac{10 \deg_{I_t \cap \mathcal{L}_i}(v)}{d}}.$$

Let us assume in the following that $i \leqslant \frac{d}{7}$. Then due to $\deg_{I_t \cap \mathcal{L}_i}(v) \leqslant i + 1$ and $1 + \frac{x}{2} \geqslant \exp(x)$ for $-1.5 < x < 0$ we get

$$\mathbf{E}\left[\, |I_{t+10} \cap \mathcal{L}_{i+1}| \,\right] \geqslant \sum_{v \in \mathcal{L}_{i+1}} \frac{10 \deg_{I_t \cap \mathcal{L}_i}(v)}{2d} = |I_t \cap \mathcal{L}_i| \cdot \frac{(d-i)10}{2d} \geqslant 4.25 \cdot |I_t \cap \mathcal{L}_i|.$$

Let $f \colon N(u_1) \times N(u_2) \times \ldots \times N(u_{|I_t \cap \mathcal{L}_i|}) \to \mathbb{N}$ describe the random variable of $|I_{t+10} \cap \mathcal{L}_{i+1}|$ depending on the choices of the initially contacted neighbors of the vertices $I_t \cap L_i$. Since some fixed vertex can obviously inform at most 10 vertices within 10 steps, we apply the

method of independent bounded differences to f (cf. Theorem 2.15) with $\lambda = \frac{1}{4}|I_t \cap \mathcal{L}_i|$ and use the fact that $|I_t \cap \mathcal{L}_i| \geqslant \frac{d(d-1)}{2}$ to obtain

$$\mathbf{Pr}\left[\, f \leqslant 4|I_t \cap \mathcal{L}_i|\,\right] \leqslant 2 \cdot \exp\left(-\frac{\frac{1}{2}(|I_t \cap \mathcal{L}_i|)^2}{200|I_t \cap \mathcal{L}_i|}\right) \leqslant \exp(-\Omega(d^2)) \leqslant 2^{-(C+2)\cdot d},$$

for every constant $C > 0$ whenever d is large enough.

Iterating over all levels $0 \leqslant i \leqslant d/7$ we require at most $2d+(d/7-2)\cdot 10 \leqslant 4d$ time-steps to get with probability at least $1 - d2^{-(C+2)d} \geqslant 1 - 2^{-(C+1)d}$ that

$$|I_{4d} \cap \mathcal{L}_{d/7}| \geqslant \frac{d(d-1)}{2} 4^{d/7-2} \geqslant 4^{d/7}.$$

By Lemma 2.30, there is a set $I'_{4d} \subseteq I_{4d}$ such that the vertices in I'_{4d} have distance at least $d/64$ to each other and I'_{4d} is of size at least

$$\frac{4^{d/7}}{\sum_{k=0}^{d/64}\binom{d}{k}} \geqslant \frac{4^{d/7}}{(64e)^{d/64}} \geqslant \left(\frac{11}{10}\right)^d,$$

where we have used the inequality $\sum_{i=0}^{k}\binom{n}{i} \leqslant (\frac{en}{k})^k$.

Backward Approximation: (from step $7d$ till step $(7+512(C+3))d$) We will now analyze the propagation of the rumor in the reverse order. Intuitively speaking, we will show that to keep a vertex w uninformed at some time-step $t' = \Theta(d)$, also a lot of other vertices have to be uninformed at time-step $t' - \mathcal{O}(d)$. More precisely, these vertices contact the vertex w within the time-interval $[t' - \mathcal{O}(d), t']$.

Again due to the symmetry of H, we only consider the vertex $w = 1^n$. Recall that $C_{t,(7+512(C+3))d}(w)$ is defined as the set of vertices which contact the vertex w within the time-interval $[t, (7 + 512(C + 3))d]$ for any $t \leqslant (7 + 512(C + 3))d$. We will now show that $C_{7d,(7+512(C+3))d}(w)$ contains some vertex v such that $\mathrm{dist}(0^d, v) \leqslant d/256$. To this end define X as the maximal time-step $\tilde{t} \leqslant (7 + 512(C + 3))d$ such that there is a vertex $v \in \mathcal{L}_{d/256} \cap C_{\tilde{t},(7+512(C+3))d}(w)$.

Let us assume for some time-step t that $C_{t,(7+512(C+3))d}(w) \cap \mathcal{L}_i \neq \emptyset$ where $d/256 \leqslant i \leqslant d$. We shall lower bound the maximal time-step $t' \leqslant t$ such that $C_{t',(7+512(C+3))d}(w) \cap \mathcal{L}_{i-1} \neq \emptyset$. Let $v' \in C_{t,(7+512(C+3))d}(w) \cap \mathcal{L}_i$. By definition of the hypercube, there are exactly i vertices in $\mathcal{L}_{i-1}$ that share an edge with v'. For every $u \in N(v') \cap \mathcal{L}_{i-1}$, let $X_{t,i}(u) := t - \max_{t'<t}\{u \text{ contacts } v' \text{ in step } t'\}$ and $X_{t,i} := \max_{u\in\mathcal{L}_{i-1}} X_{t,i}(u)$. By the ever rolling lists assumption from Section 6.2, $X_{t,i}(u)$ is a uniformly distributed integer between 1 and d, since the position of v' in the neighbor list of u is uniformly at random from $N(v')$. Let Y be an exponential variable with parameter d (and expected value $1/d$). Fix some $r \in \{1, 2, \ldots, d\}$. As

$$\mathbf{Pr}\left[\, X_{t,i}(u) \geqslant r\,\right] = \frac{d-r+1}{d} = 1 - \frac{r-1}{d} \leqslant e^{-\frac{r-1}{d}} = \mathbf{Pr}\left[\, Y \geqslant r-1\,\right],$$

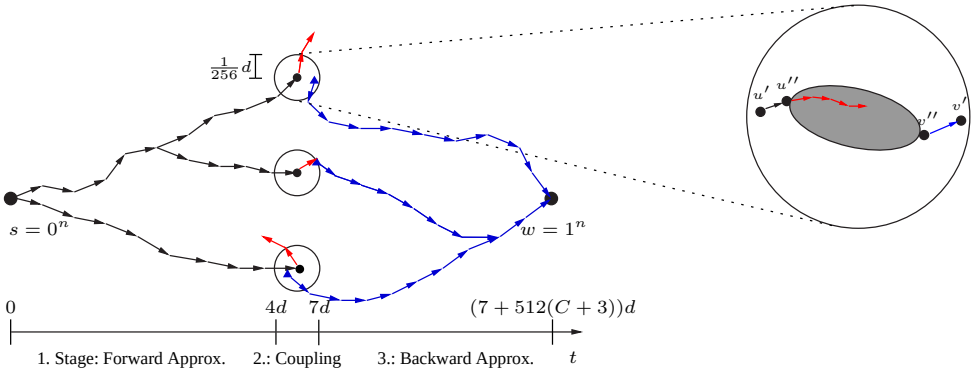

Fig. 6.3: *The left side contains a sketch of the whole proof of Theorem 6.5. The black circles represent I'_{4d}, and the blue triangles represent $\Phi(I'_{4d})$. The right side illustrates the analysis of the coupling phase. We find two vertices u'' and v'' such that every shortest path is included in a subcube of vertices whose initially contacted neighbors are unknown (indicated by the grey color).*

$X_{t,i}(u)$ is stochastically smaller than the random variable $Y+1$. Therefore, we may replace each $X_{t,i}(u)$ by an exponential variable with parameter $1/d$ plus 1. Recall that the minimum of i independent exponential variables with parameter $1/d$ is itself an exponential variable with parameter $i/d \geqslant 1/256$ (cf. Lemma 2.3). By Lemma 2.23, we may bound each $X_{t,i}$ by an exponential random variable with parameter $1/256$. Applying Theorem 2.14 we conclude that for Z being the sum of $255/256$ exponential variables with parameter $1/256$,

$$\mathbf{Pr}\left[\, X \geqslant \gamma + \frac{255}{256}d \,\right] \leqslant \mathbf{Pr}\left[\, Z \geqslant \gamma \,\right] \leqslant \frac{2^{\lambda_{\min}\mu}}{e^{\lambda_{min}\gamma/2}}.$$

Choosing $\gamma = 512(C+3)d - \frac{255}{256}d$ we conclude that

$$\mathbf{Pr}\left[\, X \geqslant 512(C+3)d \,\right] \leqslant \frac{2^{\frac{1}{256}\cdot 256d}}{e^{\frac{1}{256}\cdot(512(C+3)d - \frac{255}{256}d)/2}} = \frac{2^d}{e^{(C+3)d} \cdot e^{-\frac{255}{1024}d}} \leqslant 2^{-(C+2)\cdot d}.$$

Hence, a vertex with distance $d/256$ from 0^d is in $C_{7d,(7+512(C+3))d}(w)$ w. p. $1 - 2^{-(C+2)d}$. With the same arguments, we conclude that for an arbitrary vertex u there is a vertex $v(u)$ satisfying $\mathrm{dist}(u, v(u)) \leqslant d/256$ and $v(u) \in C_{7d,(7+512(C+3))d}(w)$ w. p. $1 - 2^{-(C+2)d}$. It follows by the union bound that for *all* vertices $u \in V(G)$ there is a vertex $v(u) \in C_{7d,(7+512(C+3))d}(w)$ with $\mathrm{dist}(u, v(u)) \leqslant d/256$ w. p. $1 - 2^{-(C+1)d}$. Recall again that due to the symmetry of H the vertex w could have been replaced by any other vertex of the graph.

Coupling: (from step $4d$ till step $7d$) We begin with the following claim.

Claim. For every $u \in I'_{4d}$ there are two vertices $u' \in I'_{4d}$ and $v' \in C_{7d,(7+512(C+3))d}(w)$ with the following properties:

1. $\mathrm{dist}(u, u') \leqslant d/256$,

2. $\operatorname{dist}(u', v') \leqslant d/256$,
3. Either $u' = v'$ or for every vertex m lying on a shortest path between u' and v', $m \notin I_{4d} \cup C_{7d,(7+512(C+3))d}(w)$.

Proof. Recall that all vertices of I'_{4d} have a pairwise distance of at least $d/64$. We have just seen that for all vertices $u \in V$ there is at least one vertex $u(v) \in C_{7d,(7+512(C+3))d}(w)$ such that $\operatorname{dist}(u, v(u)) \leqslant d/256$. Therefore, there is a bijection $\Phi\colon I'_{4d} \to C_{7d,(7+512(C+3))d}(w)$ such that $\operatorname{dist}(u, \Phi(u)) \leqslant d/256$ for all $u \in I'_{4d}$. Observe that all vertices which lie on some shortest path between u and $\Phi(u)$ form a subcube $\mathsf{Q}_{d'} = \mathsf{Q}_{d'}(u)$ of dimension $d' := \operatorname{dist}(u, \Phi(u))$. Now choose two vertices $u' \in I_{4d} \cap \mathsf{Q}_{d'}$ and $v' \in C_{7d,(7+512(C+3))d}(w) \cap \mathsf{Q}_{d'}$ for which $\operatorname{dist}(u', v')$ is minimal. If $u' \neq v'$, then all vertices m lying on a shortest path between u' and v' are neither in I_{4d} nor in $C_{7d,(7+512(C+3))d}(w)$. □

It remains to show that at least one vertex $u' \in I'_{4d}$ will contact $v' = v'(u)$ within the time interval $[4d+1, 7d]$ since this implies that w will be informed after $(7+512(C+3))d$ time-steps.

We now derive the probability that a specific $u' = u'(u) \in I'_{4d}$ informs its respective $v' = v'(u)$. Let $u'' = u''(u)$ be a neighbor of u' which is closer to v' than u'. It follows from $u' \in I_{4d}$ that $u'' \in I_{5d}$ and assume that $u'' \notin C_{7d,(7+512(C+3))d}(w)$, for if not, w becomes informed at step $(7+512(C+3))d$. Moreover, let $v'' = v''(u)$ be a neighbor of v' which is closer to u'' than v'. Again, it is clear that $v''(u) \in C_{7d-d,(7+512(C+3))d}(w)$. As before, we may assume that $v'' \notin I_{5d}$. Let m be some vertex which is on a shortest path between u'' and v''. By above claim, the initially contacted neighbor of m is uniformly at random. Hence the first neighbor to which u'' sends the rumor is u. a. r. and decreases the distance to v'' with probability $\operatorname{dist}(u'', v'')/d$. Iterating this process and using the fact that $n! \geqslant (n/3)^n$ for sufficiently large integer n gives a probability of at least

$$\prod_{k=1}^{d/256} \frac{k}{d} = \frac{(d/256)!}{d^{d/256}} \geqslant \frac{d^{d/256}}{(768d)^{d/256}} \geqslant 768^{-d/256} \geqslant \left(\frac{11}{12}\right)^d$$

for reaching v'' before time-step $7d$.

By construction and the fact that we have only considered shortest paths between $u''(u)$ and $v''(u)$, the corresponding events w. r. t. each $u''(u)$ and its respective $v''(u)$ are independent from each other. Hence, the probability that no path succeeds (conditioned on the success of the forward and backward approximation phase) is at most

$$\left(1 - \left(\frac{11}{12}\right)^d\right)^{(11/10)^d} \leqslant e^{-(\frac{121}{120})^d} \leqslant 2^{-(C+2)d},$$

for any constant $C > 0$. If we condition on the success of the first and third phase, which occurs with probability at least $1 - 2 \cdot 2^{-(C+1)d}$, then all vertices become informed with probability $1 - 2^{-(C+1)d}$, having used the union bound over all n vertices. Hence all vertices

become informed with (unconditional) probability

$$1 - 2 \cdot 2^{-(C+1)d} - 2^{-(C+1)d} \geqslant 1 - 2^{-Cd}.$$

Since $\Omega(\log n)$ is obviously a lower bound, as the number of informed vertices can at most double in each step, the claim of the theorem follows. □

Let $\mathcal{G}(n,p)$ be the probability space of random graphs with n vertices such that each edge is present independently with probability p. (Note that the case $p = 1$ gives the complete graph K_n with n vertices.) We consider the case that $p = (\log(n) + f(n))/n$, where $\lim_{n\to\infty} f(n) \to \infty$. By [ER59], such graphs are connected w. p. $1 - o(1)$.

Theorem 6.6 ([DFS08]). *Let $C > 1$ be some constant. Let $p = (\log(n) + f(n))/n$, where $\lim_{n\to\infty} f(n) \to \infty$. Then a random graph $G \in \mathcal{G}(n,p)$ with probability $1 - o(1)$ has the property that*

$$\mathsf{QR}_{n^{-C}}(G) = \Theta(\log n).$$

6.5 Conclusion

We proposed and investigated a quasirandom analogue of the classical push algorithm for broadcasting a rumor from one vertex to all vertices in a network.

We showed that for the network topologies of complete graphs, hypercubes and random graphs $\mathcal{G}(n,p)$, where p only needs to be slightly larger than the connectivity threshold, after $\Theta(\log n)$ iterations all vertices are informed with probability $1 - n^{-C}$. Hence the quasirandom model achieves asymptotically the same bounds as the random one, or even better ones (for random graphs with p very close to $\log(n)/n$). Our new model demonstrates that by equipping the vertices with some memory, the broadcast time can be reduced, even in the presence of some adversary.

From the methodological point of view, our results are also interesting. Our proofs show, in particular, that the difficulties usually invoked by highly dependent random experiments can be overcome. From the general perspective of using randomized methods in computer science, our results indicate that choosing the right amount of randomness might be a topic for fruitful further research.

One specific open problem is the runtime on edge-expanders. The proofs for hypercubes and random graphs use implicitly the fact that the vertex expansion of small subsets is of order $\Omega(\deg(G))$, a property which may not be shared by all edge-expanders.

Following the idea of probability amplification, it could be of interest to find the minimum number of random bits for efficient broadcast in the presence of some adversary, who knows the broadcast protocol but not the random bits in advance.

7. RANDOMIZED LOAD BALANCING

7.1 Introduction

For a survey on load balancing we refer the reader to [XL97].

7.1.1 Motivation

Load balancing is one of the key problems for computation-intensive applications. For this purpose, an application must be divided in several subtasks and these subtasks must be executed on different computing nodes of a parallel computer. The load balancing problem aims to subdivide the computational load as evenly as possible. In the past, this problem was mainly considered in the context of parallel computers where a central control instance exists. However in many networks we are facing nowadays, there cannot be a central authority responsible for the load balancing task. One way to deal with this limitation is the use of local load balancing schemes like diffusion [DFM99, Els02], dimension exchange schemes [Cyb89, Arn03] or the use of so-called smoothing networks [AVY94, HT06a].

A *smoothing network* is a distributed data structure that accepts tokens, which represent requests for services, at input wires and routes them asynchronously to output wires; the service is provided by the server residing on the output wire the token arrives at. The network consists of switching elements called balancers and wires. A balancer is an asynchronous switch with two input wires and two output wires, labelled top and bottom. All arriving tokens are alternately forwarded to the top and bottom output wire. The *smoothness* of such a network is the maximum discrepancy among the number of tokens arriving at different output wires. The smoothness depends, first, on the pattern of tokens arriving at the input wires and, secondly, on the balancers initializations.

We consider a random initialization [AVY94] where each balancer is initially top or bottom uniformly at random. Such a random initialization eliminates the need for global coordination and thereby offers fault-tolerance against crashes, resets or replacement of balancers. More importantly, we will show that random initialization achieves the same smoothness (up to some small additive constant) attained by the best known deterministic initializations. Such low-smoothing networks are in particular attractive for load balancing applications where low-contention is required like producers-consumers scenarios [HS08] and distributed numerical computations [BT97].

7.1.2 Related Work

A typical classification of local iterative load balancing schemes is the distinction between diffusion-based and dimension-exchange schemes. In *diffusion-based load balancing* every node is able to balance its load simultaneously with all its neighbors. The other class called *dimension-exchange schemes* is somewhat more restrictive since nodes may only communicate with *one* neighbor in each iteration. For these schemes one has to specify the balancing partners which can be done by means of some random matching [BGPS06, GM96] or some predetermined round-robin-like order [Cyb89, Arn03]. Another important classification must be made between *discrete* or *continuous* load balancing methods. In the discrete case, tokens can not be split and nodes can only send and receive integer amounts of load, while in the continuous case, tokens can be arbitrarily split. Finally, the results must be distinguished according to the used norm to measure the distance to the perfectly balanced distribution. Two popular choices are the norms ℓ_2 and ℓ_∞; the latter is also often called *discrepancy* or *smoothness*. In the following, we mention some related work about *discrete* load balancing on networks with n nodes.

Aiello et al. [AAMR93] proposed a diffusion-based load balancing algorithm which is $\mathcal{O}(\Delta n/\partial(G))$-smoothing in $\mathcal{O}(K \cdot \log(nK)/\partial(G))$ time, where K is the initial imbalance, $\Delta(G)$ the maximum degree and $\partial(G)$ is the vertex-expansion of G. That means that for any input vector, the discrepancy of the output vector is at most $\mathcal{O}(\Delta n/\partial(G))$ after this time. Ghosh et al. [GLM+99] considered a similar local load balancing scheme, where in each step only one token may be sent along an edge. They proved that their algorithm is $\mathcal{O}((\Delta^2 \log n)/\Phi)$-smoothing within $\mathcal{O}(K/\Phi)$ steps, where Φ is the edge-expansion of G. Rabani et al. [RSW98] introduced the so-called *local divergence* to measure the difference between continuous and discrete load balancing schemes. Their results hold both for a diffusive model and for a balancing circuit model (which corresponds to the dimension-exchange-paradigm). Amongst other things, they proved that every network is $\mathcal{O}(\Delta \log n/(1-\mu_2))$ smoothing within $\mathcal{O}(\log(Kn)/(1-\mu_2))$ rounds, where μ_2 is the second largest eigenvalue of a certain transition matrix of G.

All load balancing schemes mentioned so far were deterministic. Elsässer et al. [EMS06] proposed a fully distributed approach based on random walks. With λ_2 denoting the second largest eigenvalue of the transition matrix of G, they could prove that after $\mathcal{O}((\log^2 n + \log K)/(1-\lambda_2))$ steps, their algorithm reduces the discrepancy to $\mathcal{O}(1)$.

Also balls-and-bins models have been extensively studied as simple randomized, yet centralized protocols for assigning jobs (=balls) to servers (=bins). In the most basic model, one throws n balls sequentially into n bins uniformly at random. While in this case the maximum load is known to be of order $\log n/\log\log n$ [MU05], the following adaption known as the "power of two choices" [ABKU99, MU05] reduces the load to order $\log\log n$: each ball chooses *two* possible destination bins randomly and is placed in the least full bin among them. We remark that if each balancer of a block network (which is a simple and natural smoothing network, cf. Subsection 7.2.2) would forward each token to a randomly chosen output wire, the distribution of tokens at the output wires of the network would be the same as if throwing w balls (tokens) into w bins (output wires of the

network) uniformly at random, assuming that the number of tokens entering the network is w. Interestingly, if we employ each balancer with a local "power of two choices", i. e., the balancer forwards each token to its two output wires alternately, the smoothness decreases from $\Theta(\log w/\log\log w)$ to $\log\log w + \Theta(1)$.

This fair distribution of tokens at a balancer with random initialization bears some resemblance to the *randomized rounding* technique. Randomized rounding was pioneered by Raghavan and Thompson [RT87] for finding solution of certain integer programs. The basic idea is to solve the linear relaxation first and then to round all variables randomly. For more details and applications we refer the reader to [MR95].

In the work most related to our investigations, Herlihy and Tirthapura [HT06a] focused on a smoothing network called *block network* [DPRS89], which is a very simple network of depth $\log w$ that has been used in more advanced constructions such as the *periodic (counting) network* [DPRS89]. An upper bound of $2.36\sqrt{\log w}$ (with high probability) was shown in [HT06a]; this upper bound is trivially inherited to the *bitonic network* [AHS94, Bat68] and the periodic network [AHS94, DPRS89] since they both contain the block network. The upper bound from Herlihy and Tirthapura improved vastly over the smoothness of $\log w$ known before for simple constructions (such as the *bitonic merger* [KM96] and the *butterfly* [Klu94, KP92]) with global initialization and for the block network itself with local (*arbitrary* and not randomized) initialization [HT06b]. Klugerman [Klu94], and Klugerman and Plaxton [KP92] had earlier presented an elaborate construction of a network with smoothness 1; however, this network is considered to be impractical (e.g., [Knu98, AVY94, HT06a]) since it contains the AKS sorting network [AKS83] having huge constants. [AVY94] presented a randomized 2-smoothing network of depth $\log w + o(\log w)$ (with high probability). Their network contains randomized but also deterministic balancers, and consequently, requires a global initialization.

A *sorting network* is basically the same as a smoothing network, however the balancers are replaced by comparators. A sequence of w numbers to be sorted arrives synchronously and each comparator forwards the higher number to the top and the lower number to the bottom output wire, respectively. Such a network is a sorting network if the output sequence arriving at the output of the network is sorted correctly for every possible input sequence. [Bat68] presented a $\mathcal{O}(\log^2 w)$-depth sorting network called bitonic network. Later Ajtai et al. [AKS83] presented an $\mathcal{O}(\log w)$-depth sorting network known as AKS-network. However, as pointed out by Knuth [Knu98], "Batcher's method is much better, unless w exceeds the total memory capacity of all computers on earth!".

7.1.3 Our Results

Herlihy and Tirthapura formulated three interesting *open problems* about randomized smoothing networks in [HT06a, Section 5]:

1. Our bound for the smoothness of the block network does not make use of structure that may be present in the input sequence. Can we obtain better bounds if the input is already fairly smooth?

2. Can we get better bounds on the output smoothness of the randomized periodic or bitonic networks?

3. How tight is the $\mathcal{O}(\sqrt{\log w})$ upper bound for the block network? Can we get a matching lower bound?

In this thesis, we provide answers to all these problems of Herlihy and Tirthapura [HT06a].

We first consider the block network Block_w (as a randomized smoothing network). We prove that Block_w is $(\log\log w+3)$-smoothing with probability at least $1-4w^{-3}$ (Theorem 7.9). Our proof for the upper bound uses the elementary techniques developed by Herlihy and Tirthapura [HT06a] for their corresponding proof of the $\mathcal{O}(\sqrt{\log w})$ upper bound. However, our improvement is achieved through a partition into two group of layers and a separate analysis for each group. This result provides half an answer to *Open Problem 3* of Herlihy and Tirthapura [HT06a].

We proceed by establishing a matching lower bound (up to a small additive constant) on the smoothness of the block network. More precisely, we prove that the block network is a $(\log\log w-2)$-smoothing network with probability at most $2\exp(-\frac{4\sqrt{w}}{\log w})$ (Theorem 7.12.) The proof uses again a partition into two group of layers. We determine a *fixed point* input for the first group; we then prove that (with high probability) this input is not smoothed better than $\log\log w-1$ when traversing the second group. This result completes the answer to *Open Problem 3* of Herlihy and Tirthapura [HT06a]. Furthermore, we remark that our two bounds for the block network are another example of an extremely sharp threshold result (cf. Section 4.1.2): for certain input sequences the smoothness of the output of one block network is always between $\log\log w-2$ and $\log\log w+3$ with high probability.

We continue to consider the cascade of two block networks. As our main result, we prove that this network is 17-smoothing with probability at least $1-\frac{8\log\log w-94}{w}$ (Theorem 7.16). The proof uses a partition of the second block network into no more than $\frac{1}{2}\log\log w-6$ groups of layers; the number of layers per group increases as we proceed. We prove that each group is sufficient to drop the smoothness by 2. Hence, at the end, the application of Theorem 7.9 to the first cascaded block network implies a constant smoothness.

We remark that our result on the smoothness of the cascade of two block networks provides an answer to *Open Problem 1* of Herlihy and Tirthapura [HT06a]: When the input to a (randomized) block network has the properties of the output of a block network (in particular, it is $(\log\log w+3)$-smooth), then its output is 17-smooth. Also, note that the cascade of two block networks is a subnetwork of the periodic network [AHS94, DPRS89] (which consists of $\log w$ such blocks); hence, the latter is also 17-smoothing. This settles *Open Problem 2* of Herlihy and Tirthapura [HT06a]. Finally, we note that this result identifies the *first* (randomized) smoothing network that simultaneously *(i)* achieves constant smoothness, *(ii)* does not use the AKS network [AKS83] and *(iii)* does not require global initialization.

We conclude with an improbability result: Every randomized smoothing network of width w and depth d is 1-smoothing with probability at most $\frac{\mathsf{d}}{w-1}$ (Theorem 7.24). This

Network	Depth	Type	GI	Smoothness	Probability	Reference
KP network	$\Theta(\log w)$	D	✓	1	$= 1$	[Klu94, KP92]
r-butterfly	$\approx \log w$	D/R	✓	2	$\geqslant 1 - \frac{1}{\text{superpoly}(w)}$	[AVY94]
Bit. merger	$\log w$	D	✓	$\log w$	$= 1$	[KM96]
Butterfly	$\log w$	D	✓	$\log w$	$= 1$	[KP92]
Block	$\log w$	D	X	$\log w$	$= 1$	[HT06b, Thm. 3, 4]
Block	$\log w$	R	X	$2.36\sqrt{\log w}$	$\geqslant 1 - 4w^{-1}$	[HT06a, Thm. 10]
Block	$\log w$	R	X	$\log\log w + 3$	$\geqslant 1 - 4w^{-3}$	Thm. 7.9
Block	$\log w$	R	X	$\log\log w - 2$	$\leqslant 2\exp(-\frac{4\sqrt{w}}{\log w})$	Thm. 7.12
Two Blocks	$2\log w$	R	X	17	$\geqslant 1 - \frac{8\log\log w - 94}{w}$	Thm. 7.16
any	d	R	X	1	$\leqslant \frac{d}{w-1}$	Thm. 7.24

Fig. 7.1: *Summary of known bounds on the smoothness of smoothing networks.* D *and* R *stand for deterministic and randomized balancers, respectively;* D/R *stands for a combination of deterministic and randomized balancers.* GI *stands for global initialization; the corresponding column indicates whether* GI *is required or not.* KP *network stands for the network of Klugerman and Plaxton [Klu94, KP92]*

is bad news: it implies that the output of any of the common randomized smoothing networks of depth $\mathcal{O}(\log^2 w)$ (such as the periodic network [DPRS89] or the bitonic network [Bat68, AHS94]) is 1-smooth with an extremely small probability. Furthermore, only randomized smoothing networks of depth linear in w may guarantee 1-smoothness with a non-vanishing probability. Since there is a deterministic 1-smoothing network (relying, however, on the AKS network to achieve depth $\Theta(\log w)$) [Klu94, KP92], this result provides the *first* separation between deterministic and randomized (1-)smoothing networks. This separation demonstrates an unexpected limitation on the power of randomization in smoothing networks: there is some constant c between 1 and 16 such that there are randomized $(c+1)$-smoothing networks (of polylogarithmic depth) with high probability, but *not* such randomized c-smoothing networks.

Finally, we observe that our results for one (or the cascade of two) block(s) can be used for the analysis of a randomized dimension-exchange load balancing scheme on hypercubes. In particular, we get an upper bound of 17 on the resulting discrepancy after $2\log n$ rounds. This is a large improvement on the aforementioned bound by Rabani et al. [RSW98], however, their results hold for an *arbitrary* rounding procedure.

7.2 Notations, Definitions and Preliminaries

All logarithms are to the base 2 unless otherwise indicated. For an integer i, the binary representation of i is a binary word $i_1 i_2 \dots i_l$ with $l \geqslant \log i$ such that $\sum_{k=1}^{l} 2^{l-k} i_k = i$. For

an integer $i \geqslant 1$, denote $[i] = \{0, \ldots, i-1\}$. For an integer $i \geqslant 0$, the *odd-characteristic* function of i, denoted as $\mathsf{Odd}(i)$, is given by $\mathsf{Odd}(i) = 1$ if i is odd, and 0 otherwise.

We denote by $\mathbf{x}$ a vector $(x_0, \ldots, x_{w-1})$ of w integers. For a vector $\mathbf{x}$, denote $\sum \mathbf{x} = \sum_{i \in [w]} x_i$. We say that $\mathbf{x}$ is γ-*smooth* if for every pair $i, j \in [w]$, $|x_i - x_j| \leqslant \gamma$.

7.2.1 (Randomized) Smoothing Networks

General. A *smoothing network* [AHS94] is a special case of a *balancing network* [AHS94], which is a collection of interconnected *balancers*. A balancer is an asynchronous switch with two *input wires* and two *output wires*, called *top* and *bottom*. An *initialization* takes place at some preprocessing phase. The initialization chooses an *orientation* for each balancer: one of the two output wires, either the top or the bottom. A stream of tokens enters a balancer via its two input wires and is directed to the two output wires as follows. Each time a new token arrives on an input wire, it is directed to the output wire currently labelled top and at the same time, the orientation of the balancer changes. This ensures that the total number of tokens is (almost) evenly divided among the two output wires.

A *balancing network* is an acyclic network of balancers where output wires of balancers are connected to input wires of (other) balancers. The network's *input wires* $0, 1, \ldots, w-1$ may not be connected from any output wires; the network's *output wires* may not be connected to any input wire. When the numbers of input and output wires of the network are the same, this number w is called the *width* of the network and the network is denoted by B_w. The network's acyclicity ensures that each balancer can be assigned a unique *layer*, which is the length of the *longest* path from an input wire to that layer. The *depth* of a network is the maximum layer and is denoted by $\mathsf{d}(\mathsf{B}_w)$.

The network $\mathsf{Prefix}_\ell(\mathsf{B}_w)$ consists of the layers $1, \ldots, \ell$ of B_w; the network $\mathsf{Suffix}_\ell(\mathsf{B}_w)$ consists of the layers $\mathsf{d}(\mathsf{B}_w) - \ell + 1, \ldots, \mathsf{d}(\mathsf{B}_w)$. Finally, for an integer $k \geqslant 1$, B_w^κ denotes the sequential cascade of κ copies of B_w.

If a balancer b is located in layer ℓ, we shall write $\mathsf{b} \in \ell$. Say that a balancer $\mathsf{b} \in \ell$ of a balancing network B_w *depends* on balancer $\mathsf{b}' \in \ell', \ell' < \ell$ if there is a path from b' to b in B_w. Then, each output wire of a balancer b depends on balancer b' as well (and also trivially on b). The *dependency set* of a balancer b in layer ℓ is the set of all balancers $\mathsf{b}' \in \ell', \ell' < \ell$ such that b depends on b'. Consider two output wires j_1 and j_2 of layer ℓ in a balancing network B_w. Say that j_1 and j_2 are *independent for layer* $\ell', \ell' < \ell$ if there is no balancer $\mathsf{b}' \in \ell'$ such that both j_1 and j_2 depend on b'.

Deterministic and randomized balancers. This distinction refers to the way balancers are initialized. A *deterministic balancer* is one that is initialized in some deterministic way. A *deterministic balancing network* consists of deterministic balancers (and wires). A pair of a deterministic balancing network B_w and a (fixed) orientation for each of its balancers induces a set of (asynchronous) *executions* in the natural way (cf. [AHS94, Section 2]). Consider an *input vector* $\mathbf{x} = (x_0, x_1, \ldots, x_{w-1})$, where x_i is the number of tokens fed into input wire i of the network B_w. A quiescent state of the network B_w on the input vector $\mathbf{x}$ is reached in some execution when all input tokens $\sum \mathbf{x}$ have exited the network. It is simple to observe that *all* executions of network B_w (on the input vector $\mathbf{x}$) reach a

quiescent state with a common output vector $\mathbf{y} = (y_0, y_1, \ldots, y_{w-1})$. So, identify each such quiescent state with the vector $\mathbf{y}$ and denote this common vector as $\mathsf{B}_w(\mathbf{x})$. A vector $\mathbf{x}$ is a *fixed point* for the network B_w if $\mathsf{B}_w(\mathbf{x}) = \mathbf{x}$ (cf. [HT06b]).

Definition 7.1. *For some integer $\gamma \geqslant 1$, B_w is a* γ-smoothing network *if for each input vector $\mathbf{x}$, $\mathsf{B}_w(\mathbf{x})$ (corresponding to some local orientation of it) is γ-smooth.*

A *randomized balancer* [AVY94, HT06a] is initialized to each of top and bottom with probability $\frac{1}{2}$ and independently of all other balancers. A *randomized balancing network* consists of randomized balancers. Clearly, a fixed input vector $\mathbf{x}$ to a randomized balancing network induces a probability distribution on the set of possible output vectors for this input $\mathbf{x}$.

Definition 7.2. *For some integer $\gamma \geqslant 1$, B_w is a* γ-smoothing network with probability δ, *where $0 \leqslant \delta \leqslant 1$, if for all input vectors $\mathbf{x}$, $\mathbf{Pr}[\,\mathsf{B}_w(\mathbf{x})$ is γ-smooth$\,] \geqslant \delta$.*

Hence, B_w is a γ-smoothing network with probability δ if for all input vectors the probability that all output wires j and k, $j, k \in [w], j \neq k$ satisfy $|y_j - y_k| \leqslant \gamma$ is at least δ.

For a balancer b, denote as x_1 and x_2 the number of tokens arriving on the top and bottom input wires of b, respectively. (We shall sometimes use $x_1(\mathsf{b}), x_2(\mathsf{b}), y_1(\mathsf{b}), y_2(\mathsf{b})$ for x_1, x_2, y_1 and y_2, respectively, when reference to b is necessary.) Denote as y_1 and y_2 the number of tokens leaving through the top and bottom output wires of b, respectively. If b is oriented top,

$$y_1 = \left\lceil \frac{x_1 + x_2}{2} \right\rceil \quad \text{and} \quad y_2 = \left\lfloor \frac{x_1 + x_2}{2} \right\rfloor$$

and if b is oriented bottom,

$$y_1 = \left\lfloor \frac{x_1 + x_2}{2} \right\rfloor \quad \text{and} \quad y_2 = \left\lceil \frac{x_1 + x_2}{2} \right\rceil .$$

Assume now that b is oriented uniformly at random. Define a random variable r_b taking values $\frac{1}{2}$ and $-\frac{1}{2}$ with equal probability (cf. [HT06a]). (Clearly, $\mathbf{E}[\,r_\mathsf{b}\,] = 0$.) Define $x_\mathsf{b} = \mathsf{Odd}(x_1 + x_2) \cdot r_\mathsf{b}$ (cf. [HT06a]). Then,

$$\begin{aligned} y_1 &= \frac{x_1 + x_2}{2} + x_\mathsf{b} = \frac{x_1 + x_2}{2} + \mathsf{Odd}(x_1 + x_2) \cdot r_\mathsf{b}, \\ y_2 &= \frac{x_1 + x_2}{2} - x_\mathsf{b} = \frac{x_1 + x_2}{2} - \mathsf{Odd}(x_1 + x_2) \cdot r_\mathsf{b}. \end{aligned}$$

7.2.2 The Block Network (and its Relatives)

The Block network Block_w was introduced as a *comparator network* in [DPRS89]; it was later investigated as balancing networks in [AHS94] (Roughly speaking, the isomorphic balancing network to a comparator network replaces comparators with balancers.).

Construction and Structure. We use the following definition for Block_w [MMT99]. For any w a power of 2, the network Block_w has $\log w$ layers. For each layer ℓ, $1 \leqslant \ell \leqslant \log w$ and for each wire $u \in \{0,1\}^{\log w}$, there is a balancer b between wire $u = u_1 u_2 \dots u_{\log w}$ and wire $u_1 \dots u_\ell \overline{u_\ell} \dots \overline{u_{\log w}}$, i. e., the last $\log w - \ell + 1$ bits of u are flipped. The top output wire of b is the one among u and $u(\ell)$ such that $u_\ell = 0$. See Figure 7.3 for an illustration.

We will use the tree structure from [HT06a, Section 2] for the Block_w network:

- The *root* is the set of all $\frac{w}{2}$ balancers at layer 1 of Block_w; label this node $v_{1,1}$. The *leafs* of the tree are the balancers in layer $\log w$.
- For each ℓ, $2 \leqslant \ell \leqslant \log w$, layer ℓ is decomposed into $2^{\ell-1}$ nodes, denoted as $v_{\ell,1}, \dots, v_{\ell,2^{\ell-1}}$, each consisting of $\frac{w}{2^\ell}$ balancers. These nodes are defined inductively (given the nodes for layer $\ell - 1$) as follows: For each integer k, where $1 \leqslant k \leqslant 2^{\ell-2}$, the node $v_{\ell,2k-1}$ consists of all balancers (in layer ℓ) that the top output wires of balancers in node $v_{\ell-1,k}$ point to. Similarly, the node $v_{\ell,2k}$ consists of all balancers (in layer ℓ) which the bottom output wires of balancer in node $v_{\ell-1,k}$ point to.

The tokens that exit from output wire y_1 must follow the path $v_{1,1}, v_{2,1}, \dots, v_{\log w,1}$ and further exit on the top output wire of balancer $v_{\log w,1}$. See Figure 7.2 for an illustration of the tree structure.

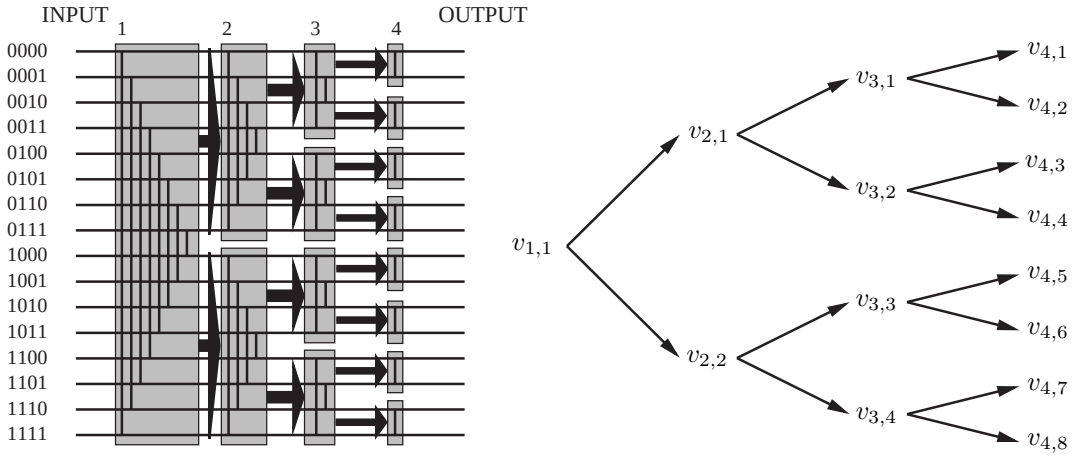

Fig. 7.2: *An example of the tree structure of [HT06a] within a* Block_{16}.

We observe a preliminary property of the network Block_w which will be used later.

Observation 7.3. *In* Block_w, *there is at most one path from a balancer* b *in layer* ℓ *to a balancer* b' *in layer* $\ell' > \ell$.

Relatives. The block network is very similar to (but different than) the well-known *merger* network of Batcher [Bat68]. In more detail, under the *standard* orientation (cf. [HT06a]), there is no permutation between the input wires of the two networks that yields one from the other while respecting the orientation of each balancer. However, if the balancers'

orientations are ignored, such permutations exist and the networks are called *isomorphic* (cf. [DPRS89, Section 2].) The $\mathsf{Periodic}_w$ network is the cascade of $\log w$ Block_w networks.

The *cube-connected-cycles* network: For any w a power of 2, the network CCC_w has $\log w$ layers. In layer ℓ, $1 \leqslant \ell \leqslant \log w$, for each wire $u \in \{0,1\}^{\log w}$, there is a balancer b between wire u and wire $u(\ell)$, where $u(\ell) = u_1 \dots u_{\ell-1}\overline{u_\ell}u_{\ell+1} \dots u_{\log w}$; the top output wire of b is the one among u and $u(\ell)$ such that $u_\ell = 0$. See Figure 7.3 for an illustration.

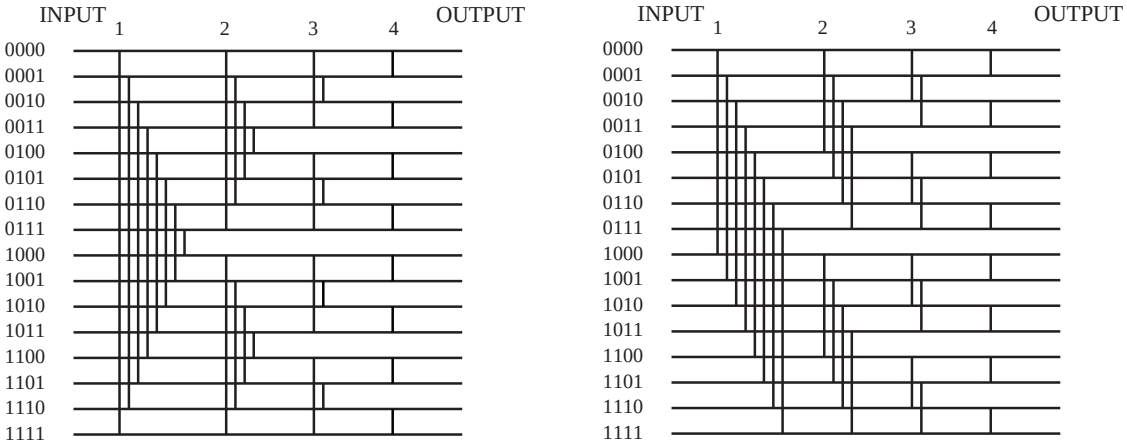

Fig. 7.3: *The* Block_{16} *and the* CCC_{16} *network.*

We observe the following simple property of CCC_w.

Lemma 7.4. *Consider two integers* l_1 *and* l_2 *such that* $l_1 + l_2 < \log w$. *Let* $\ell_1 \in \{0,1\}^{l_1}$ *and* $\ell_2 \in \{0,1\}^{l_2}$ *be arbitrary but fixed. Then, the restriction of* CCC_w *to the layers* $l_1 + 1, \dots, \log w - l_2$ *and wires* $\{\ell_1 u \ell_2 \mid u \in \{0,1\}^{\log w - l_1 - l_2}\}$ *is a network* $\mathsf{CCC}_{2^{\log w - l_1 - l_2}}$.

Proof. Consider an arbitrary wire $\ell_1 u \ell_2$ in layer ℓ, where $l_1 + 1 \leqslant \ell \leqslant \log w - l_2$. Map this wire to u in layer $\ell - l_1$ of $\mathsf{CCC}_{2^{\log w - l_1 - l_2}}$. Clearly, this defines a bijection between the restriction of wires $\ell_1 u \ell_2, u \in \{0,1\}^{\log w - l_1 - l_2}$, of a CCC_w to the wires of a $\mathsf{CCC}_{2^{\log w - l_1 - l_2}}$. Consider now a balancer b of CCC_w in layer ℓ, $l_1 + 1 \leqslant \ell \leqslant \log w - l_2$ connecting wires $\ell_1 u \ell_2$ and $\ell_1 u \ell_2(\ell)$. By definition, both wires are mapped to u and $u(\ell - l_1)$ in layer $\ell - l_1$ of $\mathsf{CCC}_{2^{\log w - l_1 - l_2}}$, for which there is a balancer connecting them by definition of the $\mathsf{CCC}_{2^{\log w - l_1 - l_2}}$. The claim follows. □

It is simple to see that the block network is a *bidelta* network [KS86] (A bidelta network is a one that is a *delta* network in both directions (from left to right and vice versa); roughly speaking, a delta network is one in which there is a unique path from each input wire to every output wire, and the path descriptors associated with paths leading to the same output wire are identical.) The cube-connected-cycles is another example of a bidelta network. It is known that any two bidelta networks of the same width (and degree 2, say) are isomorphic [KS86]. Hence, the block network is isomorphic to the cube-connected-cycles network (assuming the balancer orientations are ignored). This offers some convenience to our analysis since we are allowed to treat the two networks interchangeably when considering random orientations.

Block Network with Random Orientation. We now outline some tools for the analysis of the randomized block network developed in [HT06a, Section 2.1]. The numbers $y_0, y_1, \ldots, y_{w-1}$ of tokens on output wires $0, 1, \ldots, w-1$ are random variables (following some distribution). Since each balancer is initialized uniformly at random, the symmetry of the block network implies that all variables $y_j, 0 \leqslant j \leqslant w-1$ follow the same distribution (cf. [HT06a]).

Recall now the tree structure associated with the network Block_w. Since all random variables $y_j, 0 \leqslant j \leqslant w-1$, follow the same distribution, we focus on the number of tokens y_1 exiting on the top output wire of node $v_{\log w,1}$. To calculate y_0 we need to count the number of tokens following the path $v_{1,1}, v_{2,1}, \ldots, v_{\log w,1}$ and exiting on the top output wire of $v_{\log w,1}$. We restate the following lemmas.

Lemma 7.5 ([HT06a]). *For the* Block_w, $y_0 = \frac{\sum \mathbf{x}}{w} + \sum_{\ell=1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b} \in v_{\ell,1}} x_b$.

Lemma 7.6 ([HT06a]). *Consider a set of balancers* $\mathcal{B}$ *in* Block_w *and a constant* c_b *for each balancer* $\mathsf{b} \in \mathcal{B}$. *Then for any* $\delta > 0$,

$$\mathbf{Pr}\left[\sum_{\mathsf{b}\in\mathcal{B}} c_\mathsf{b} x_\mathsf{b} > \delta\right] \leqslant 2 \cdot \mathbf{Pr}\left[\sum_{\mathsf{b}\in\mathcal{B}} c_\mathsf{b} r_\mathsf{b} > \delta\right].$$

Note that $\sum_{\mathsf{b}\in\mathcal{B}} c_\mathsf{b} x_\mathsf{b}$ is a sum of *dependent* random variables, while $\sum_{\mathsf{b}\in\mathcal{B}} c_\mathsf{b} r_\mathsf{b}$ is a sum of *independent* random variables. Furthermore, note that for each set of balancers $\mathcal{B}$, linearity of expectations implies $\mathbf{E}\left[\sum_{\mathsf{b}\in\mathcal{B}} c_b r_b\right] = 0$. The following lemma can be easily derived by adapting the proof from [HT06a].

Lemma 7.7. *For the network* Block_w, $\mathbf{Pr}\left[\left|y_0 - \frac{\sum \mathbf{x}}{w}\right| \geqslant \delta\right] \leqslant 4\exp(-\delta^2)$.

Proof. First, we recall the following technical claim.

Lemma 7.8 ([HT06a]). *For the* Block_w, $\sum_{\ell=1}^{\log w} \sum_{\mathsf{b}\in v_{\ell,1}} \left(\frac{1}{2^{\log w-\ell+1}} - \left(-\frac{1}{2^{\log w-\ell+1}}\right)\right)^2 = 2 - \frac{2}{w}$.

By Lemma 7.6, $\mathbf{Pr}\left[\left|\sum_{\ell=1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} x_\mathsf{b}\right| \geqslant \delta\right] \leqslant 2 \cdot \mathbf{Pr}\left[\left|\sum_{\ell=1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} r_\mathsf{b}\right| \geqslant \delta\right]$. Note that for each pair of layer $\ell, 1 \leqslant \ell \leqslant \log w$ and a balancer $\mathsf{b} \in v_{\ell,1}$, the random variable r_b has range $\{-2^{-\log w+\ell-1}, 2^{-\log w+\ell-1}\}$. Hence, by Hoeffding bound (Theorem 2.12) and Lemma 7.8, we obtain $\mathbf{Pr}\left[\left|\sum_{\ell=1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} x_\mathsf{b}\right| \geqslant \delta\right] \leqslant 2 \cdot 2 \cdot \exp\left(-\frac{2\delta^2}{2}\right)$. □

7.3 One Block Network

We present both upper and lower bounds on the smoothness of the network Block_w.

Theorem 7.9. Block_w *is a* $(\log\log w + 3)$*-smoothing network with probability at least* $1 - 4w^{-3}$.

Proof. By Lemma 7.5,

$$\begin{aligned} y_0 &= \frac{\sum \mathbf{x}}{w} + \sum_{\ell=1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} x_{\mathsf{b}} \\ &= \frac{\sum \mathbf{x}}{w} + \underbrace{\sum_{\ell=1}^{\log w-\lceil \log\log w\rceil} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} x_{\mathsf{b}}}_{X_1} + \underbrace{\sum_{\ell=\log w-\lceil \log\log w\rceil+1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} x_{\mathsf{b}}}_{X_2}, \end{aligned}$$

where for each layer ℓ, $|v_{\ell,1}| = 2^{\log w-\ell}$. Before we investigate X_1, we observe the following lemma.

Lemma 7.10. *For* Block_w, $\sum_{\ell=1}^{\log w-\lceil \log\log w\rceil} \sum_{\mathsf{b}\in v_{\ell,1}} \left(\frac{1}{2^{\log w-\ell+1}} - (-\frac{1}{2^{\log w-\ell+1}})\right)^2 \leqslant \frac{2}{\log w}$.

Proof. Clearly,

$$\begin{aligned} & \sum_{\ell=1}^{\log w-\lceil \log\log w\rceil} \sum_{\mathsf{b}\in v_{\ell,1}} \left(\frac{1}{2^{\log w-\ell+1}} - (-\frac{1}{2^{\log w-\ell+1}})\right)^2 \\ = & \sum_{\ell=1}^{\log w-\lceil \log\log w\rceil} 2^{\log w-\ell} \left(\frac{1}{2^{\log w-\ell}}\right)^2 \\ = & \sum_{\ell=1}^{\log w-\lceil \log\log w\rceil} \frac{1}{2^{\log w-\ell}} = \frac{1}{w} \sum_{\ell=1}^{\log w-\lceil \log\log w\rceil} 2^{\ell} \\ \leqslant & \frac{1}{w}\cdot(2^{\log w-\log\log w+1}-1) \leqslant \frac{2}{\log w}, \end{aligned}$$

as needed. □

Lemma 7.11. $\mathbf{Pr}\left[\,|X_1| \geqslant 2\,\right] \leqslant 4w^{-4}$

Proof. Let $R_1 = \sum_{\ell=1}^{\log w-\lceil \log\log w\rceil} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} r_{\mathsf{b}}$. Lemma 7.6 gives $\mathbf{Pr}\left[\,|X_1| \geqslant 2\,\right] \leqslant 2 \cdot \mathbf{Pr}\left[\,|R_1| \geqslant 2\,\right]$. Note that for each pair of a layer ℓ, $1 \leqslant \ell \leqslant \log w - \lceil \log\log w\rceil$, and a balancer $\mathsf{b} \in v_{\ell,1}$, the random variable $\frac{1}{2^{\log w-\ell}} \cdot r_{\mathsf{b}}$ has range $\{-2^{-\log w+\ell-1}, +2^{-\log w+\ell-1}\}$. Using Lemma 7.10, we conclude by Hoeffding bound (Theorem 2.12) that

$$\mathbf{Pr}\left[\,|X_1| \geqslant 2\,\right] \leqslant 2\cdot 2\cdot \exp\left(-\frac{2\cdot 2^2}{\frac{2}{\log w}}\right) \leqslant 4w^{-4}.$$

□

To bound $|X_2|$ we use the triangle inequality to get

$$\begin{aligned}
|X_2| &\leqslant \sum_{\ell=\log w-\lceil\log\log w\rceil+1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} |x_{\mathsf{b}}| \\
&\leqslant \sum_{\ell=\log w-\lceil\log\log w\rceil+1}^{\log w} \frac{1}{2^{\log w-\ell}} \sum_{\mathsf{b}\in v_{\ell,1}} \frac{1}{2} \\
&= \sum_{\ell=\log w-\lceil\log\log w\rceil+1}^{\log w} \frac{1}{2^{\log w-\ell}} \cdot 2^{\log w-\ell} \cdot \frac{1}{2} = \frac{1}{2} \cdot \lceil\log\log w\rceil.
\end{aligned}$$

Using the fact that each $y_k, 0 \leqslant k \leqslant w-1$, follows the same distribution as y_0 and using the union bound over all y_k, we obtain

$$\mathbf{Pr}\left[\bigvee_{k\in[w]} \left(\left|y_k - \frac{\sum \mathbf{x}}{w}\right| \geqslant \frac{1}{2}\lceil\log\log w\rceil + 2\right)\right] \leqslant 4w^{-3}. \tag{7.1}$$

The event $\bigwedge_{k\in[w]} \left(\left|y_k - \frac{\sum \mathbf{x}}{w}\right| < \frac{1}{2}\lceil\log\log w\rceil + 2\right)$ implies for each pair of indices $k, l \in [w]$,

$$\begin{aligned}
|y_k - y_l| &\leqslant \left|y_k - \frac{\sum \mathbf{x}}{w}\right| + \left| - y_l + \frac{\sum \mathbf{x}}{w}\right| \leqslant \frac{1}{2}\lceil\log\log w\rceil + 1 + \frac{1}{2}\lceil\log\log w\rceil + 1 \\
&= \lceil\log\log w\rceil + 2 \leqslant \log\log w + 3.
\end{aligned}$$

By 7.1 and the union bound,

$$\begin{aligned}
\mathbf{Pr}\left[\,\mathbf{y} \text{ is } (\log\log w + 3)\text{-smooth}\,\right] &\geqslant \mathbf{Pr}\left[\bigwedge_{k\in[w]} \left(\left|y_k - \frac{\sum \mathbf{x}}{w}\right| \leqslant \frac{1}{2}\lceil\log\log w\rceil + 2\right)\right] \\
&\geqslant 1 - 4w^{-3}.
\end{aligned}$$

□

We continue with the lower bound. To avoid the extensive use of floors and ceilings, we confine ourselves to the case of $\log w$ being a power of 2.

Theorem 7.12. *For any w such that $\log w$ is a power of 2, Block_w is a $(\log\log w - 2)$-smoothing network with probability at most $2 \cdot \exp(-\frac{4\sqrt{w}}{\log w})$.*

Proof. Since the networks Block_w and CCC_w are isomorphic, we shall deal with the second.

We construct an input $\mathbf{x}$ such that the probability that $\mathbf{y} = \mathsf{CCC}_w(\mathbf{x})$ is $(\log\log w - 2)$-smooth is at most $2 \cdot \exp(-\frac{4\sqrt{w}}{\log w})$. Construct $\mathbf{x}$ as follows. For each input wire $i = i_1 i_2 \ldots i_{\log w}$, set $x_i := \sum_{k=\log w-\log\log w+2}^{\log w} i_k$. So, x_i is the number of 1's in the $\log\log w - 1$

least significant bits of $i_1 i_2 \dots i_{\log w}$. We note that an illustration for this input sequence for a CCC_{16} is given in Figure 7.4.

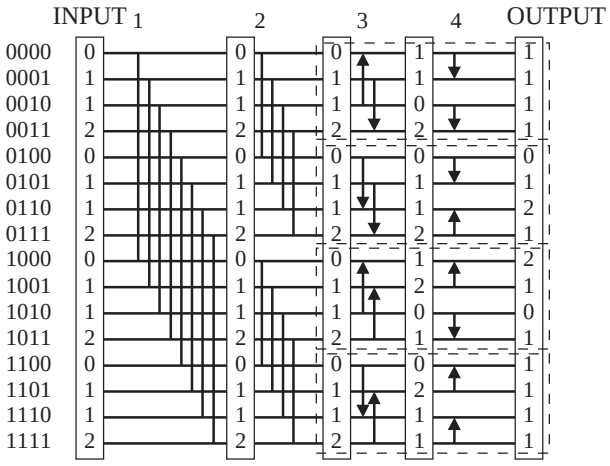

Fig. 7.4: *An example of an orientation of a CCC_{16} for the lower bound. The number of tokens in an input wire is the sum of 1's in the least two significant bits. The vertical boxes represent the number of tokens entering a wire before layers $1, 2, 3, 4$ and at the output, respectively. The orientation of the first two layers is not drawn, as the output vector of the second layer is a fixed point.*

Using the recursive structure of the CCC_w, we prove three elementary lemmas.

Lemma 7.13. $\mathbf{x}$ *is a fixed point of* $\mathsf{Prefix}_{\log w - \log\log w + 1}(\mathsf{CCC}_w)$.

Proof. We prove that for each layer $\ell \leqslant \log w - \log\log w + 1$, $\mathbf{x}$ is a fixed point of the subnetwork of CCC_w consisting of the ℓ leftmost layer of CCC_w. The proof is by induction on ℓ. For the basis case, where $\ell = 1$, consider a balancer b in layer ℓ connecting wires u and $u(1)$, where $u \in \{0,1\}^{\log w}$. By construction of $\mathbf{x}$, the inputs to balancer b are

$$x_1(\mathsf{b}) = \sum_{k=\log w - \log\log w + 2}^{\log w} u_k \quad \text{and} \quad x_2(\mathsf{b}) = \sum_{k=\log w - \log\log w + 2}^{\log w} u(1)_k.$$

By construction of the network, $u_k = u(1)_k$ for all $k \geqslant \log w - \log\log w + 2$ (since u and $u(1)$ differ only in bit 1.) Hence, $x_1(\mathsf{b}) = x_2(\mathsf{b})$. By definition of balancers, it follows that $y_1(\mathsf{b}) = y_2(\mathsf{b})$. Consequently, $\mathbf{x}$ is a fixed point of the network consisting of the subnetwork of CCC_w consisting of its leftmost layer.

Assume inductively that the claim holds for layer $\ell - 1$, where $1 < \ell < \log w - \log\log w + 2$. Consider a balancer b in layer ℓ connecting wires u and $u(\ell)$, where $u \in \{0,1\}^{\log w}$. By induction hypothesis, the inputs to b are $x_1(\mathsf{b}) = \sum_{k=\log w - \log\log w + 2}^{\log w} u_k$ and $x_2(\mathsf{b}) = \sum_{k=\log w - \log\log w + 2}^{\log w} u(\ell)_k$. By construction of the network, $u_k = u(\ell)_k$ for all $k \geqslant \log w - \log\log w + 2$ (since u and $u(\ell)$ differ only in bit ℓ.) Hence, $x_1(\mathsf{b}) = x_2(\mathsf{b})$. By definition of

balancers, it follows that $y_1(\mathsf{b}) = y_2(\mathsf{b})$. Hence, $\mathbf{x}$ is a fixed point of the network consisting of the subnetwork of CCC_w consisting of its leftmost ℓ layers. The claim follows. □

We now focus on the subnetwork $\mathsf{Suffix}_{\log\log w-1}(\mathsf{CCC}_w)$. By the construction of the CCC_w network (cmp. Lemma 7.4), this subnetwork is the parallel cascade of $\frac{2w}{\log w}$ $\mathsf{CCC}_{\frac{\log w}{2}}$ networks. Take any such network $\mathsf{CCC}_{\frac{\log w}{2}}$. Observe that the input wires of this network are the wires $u0^{\log\log w-1}, \ldots, u1^{\log\log w-1}$ for some $u \in \{0,1\}^{\log w-\log\log w+1}$. Lemma 7.13 implies that the input to wire $i = uv$, where $v \in \{0,1\}^{\log\log w-1}$ is $\sum_{k=\log w-\log\log w+2}^{\log w} i_k$. Consider the wire $u1^{\log\log w-1}$. Now we state a second lemma for this proof.

Lemma 7.14. $\mathbf{Pr}\left[\, y_{u1^{\log\log w-1}} = 0 \,\right] \geqslant 2^{-(\frac{\log w}{2}-1)}$ *and* $\mathbf{Pr}\left[\, y_{u1^{\log\log w-1}} = \log\log w - 1 \,\right] \geqslant 2^{-(\frac{\log w}{2}-1)}$.

Proof. Note that the output wire $u1^{\log\log w-1}$ of CCC_w depends on $1 + \sum_{k=1}^{\log\log w-2} 2^k = \frac{\log w}{2} - 1$ balancers in layers $\log w - \log\log w + 1, \ldots, \log w$. Notice also that there are $2^{\frac{\log w}{2}-1}$ orientations for these balancers, each occurring with the same probability. Hence, it suffices to prove that each of 0 and $\log\log w - 1$ is a possible value for the number of tokens emitting at output wire $u1^{\log\log w-1}$.

For simplicity, set $w' = \log w$. The proof is by induction on w'. For the basis case, where $w' = 4$, the claim is verified directly (see also Figure 7.3 or 7.4). Assume inductively that the output wire $u1^{\log w'}$ in the network $\mathsf{CCC}_{w'}$ can take the values 0 and $\log w'$. For the induction step, consider the network $\mathsf{CCC}_{2w'}$. Consider the output wire $0u1^{\log w'}$. By construction of the cube-connected-cycles network, $\mathsf{CCC}_{2w'}$ consists of a ladder network followed by two parallel $\mathsf{CCC}_{w'}$ networks. Consider the top of these $\mathsf{CCC}_{w'}$ networks.

- Assume that all balancers in layer 1 of the network $\mathsf{CCC}_{2w'}$ are initialized bottom. Then, the input to each of the input wires of $\mathsf{CCC}_{w'}$ equals the number of 1's in the corresponding input wire $0i'$, where $i' \in \{0,1\}^{\log(2w')-1} = \{0,1\}^{\log w'}$. Clearly, this number equals the number of 1's in the string i'. Induction hypothesis implies that the output wire $u1^{\log w'}$ can have value 0.

- Assume now that all balancers in layer 1 of the network $\mathsf{CCC}_{2w'}$ are initialized top. Then, the input to each of the input wires of CCC'_w equals the number of 1's in the corresponding input wire $i = 1i'$ where $i' \in \{0,1\}^{\log w'}$. Clearly, this number equals 1 plus the number of 1's in the string i'. Induction hypothesis implies that the output wire $u1^{\log w'}$ may have value $1 + \log w' = \log 2w'$.

□

Consider two different subnetworks $\mathsf{CCC}_{\frac{\log w}{2}}$ with input wires $u0^{\log\log w-1}, \ldots u1^{\log\log w-1}$ and $u'0^{\log\log w-1}, \ldots, u'1^{\log\log w-1}$, respectively. Correspondingly, consider the two output wires $u1^{\log\log w-1}$ and $u'1^{\log\log w-1}$, respectively.

Lemma 7.15. $\{y_{u1^{\log\log w-1}} \mid u \in \{0,1\}^{\log w-\log\log w+1}\}$ *is a set of independent random variables.*

Proof. By construction of the CCC_w, each random variable $y_{u1^{\log\log w-1}}$ is determined by *(i)* the input to the subnetwork $\mathsf{CCC}_{\frac{\log w}{2}}$ with input wires $u0^{\log\log w-1},\ldots,u1^{\log\log w-1}$, and *(ii)* the (randomly) chosen orientation of the same subnetwork $\mathsf{CCC}_{\frac{\log w}{2}}$. Lemma 7.13 implies that the input to the subnetwork in *(i)* is uniquely determined. Since all subnetworks $\mathsf{CCC}_{\frac{\log w}{2}}$ are disjoint, there is no balancer b in layer $\ell \geqslant \log w - \log\log w + 1$ in CCC_w such that more than one output wires $u1^{\log\log w-1}$ and $u'1^{\log\log w-1}$, with $u,u' \in \{0,1\}^{\log w-\log\log w+1}$, depend on. The claim follows. □

Using Lemma 7.15 we get

$$\begin{aligned}\mathbf{Pr}\left[\bigwedge_{u\in\{0,1\}^{\log w-\log\log w+1}} y_{u1^{\log\log w-1}} \neq 0\right] &= \prod_{u\in\{0,1\}^{\log w-\log\log w+1}} (1-\mathbf{Pr}\left[\, y_{u1^{\log\log w-1}} = 0\,\right])\\ &\leqslant \left(1-2^{-(\frac{\log w}{2}-1)}\right)^{2^{\log w-\log\log w+1}}\\ &\leqslant \exp(-2^{\log w-\log\log w+1-\frac{\log w}{2}+1}) = \exp\left(-\frac{4\sqrt{w}}{\log w}\right),\end{aligned}$$

and $\mathbf{Pr}\left[\bigwedge_{u\in\{0,1\}^{\log w-\log\log w+1}}(y_{u1^{\log\log w-1}} \neq \log\log w-1)\right] \leqslant \exp(-\frac{4\sqrt{w}}{\log w})$. By the union bound,

$$\begin{aligned}\mathbf{Pr}\left[\,\mathbf{y} \text{ is } (\log\log w-2)\text{-smooth}\,\right] &\leqslant \mathbf{Pr}\left[\bigwedge_{u\in\{0,1\}^{\log w-\log\log w+1}} (y_{u1^{\log\log w-1}} \neq 0)\right]\\ &\quad + \mathbf{Pr}\left[\bigwedge_{u\in\{0,1\}^{\log w-\log\log w+1}} (y_{u1^{\log\log w-1}} \neq \log\log w-1)\right]\\ &\leqslant 2\cdot\exp\left(-\frac{4\sqrt{w}}{\log w}\right).\end{aligned}$$

□

7.4 The Cascade of Two Block Networks

We consider the cascade of two networks Block_w. We prepare the reader that the analysis of the smoothness properties of the second cascaded block will require some smoothness properties of the first, which go beyond the one stated in Theorem 7.9.

We aim at proving that the cascade of two block networks is 17-smoothing with high probability. Note that this is an improvement on the upper bound of Theorem 7.9 only for extremely large values of w. Hence this upper bound of 17 is only of theoretical interest. Therefore, we confine ourselves to special values of w, namely those for which $(\log\log w)/2$

is an integer. It should be clear that a similar statement could be shown for any w, however, the proof would be rather cumbersome due to the extensive use of floors and ceilings.

Theorem 7.16. *Fix some $w \geqslant 2^{2^{12}}$ for which $(\log\log w)/2$ is an integer. Then, Block_w^2 is a 17-smoothing network with probability at least $1 - \frac{8\log\log w - 94}{w}$.*

Proof. Here is an informal outline of the proof. Recall that by Theorem 7.9, the first Block_w is $(\log\log w + 3)$-smoothing with high probability. The proof will use no more than $\frac{\log\log w}{2} - 6$ phases; in phase ρ, $1 \leqslant \rho \leqslant \frac{\log\log w}{2} - 6$, we consider a distinct group of $\frac{4\log w}{(\log\log w - 1 - \rho)^2} + \log\log w$ consecutive layers in the second Block_w. For each phase $\rho > 1$, we prove that the smoothness of the cascade of the first Block_w and the layers considered in the phase ρ drops by two over the corresponding smoothness for phase $\rho - 1$. At the end, this will establish that the smoothness will become constant. We remark that a sketch of the proof may be found in Figure 7.5. The formal proof follows.

Since the networks Block_w and CCC_w are isomorphic, we shall deal with the cascade of two CCC_w. Consider the second cascaded CCC_w. We prove the following lemma.

Lemma 7.17. *Fix a layer ℓ with $\log w + 1 \leqslant \ell \leqslant 2\log w$ in the network CCC_w^2. Consider the input vector $\mathbf{x} = \mathbf{x}(\ell)$ to layer ℓ. Then the following statement holds with probability $1 - 4w^{-3}$. For every ζ, $0 \leqslant \zeta \leqslant \log w - \log\log w$, and for all pairs $\ell_1 \in \{0,1\}^{\ell - 1 - \log w}$ and $\ell_2 \in \{0,1\}^{2\log w - \ell + 1 - \log\log w - \zeta}$,*

$$\left| \frac{\sum_{u \in \{0,1\}^{\log\log w + \zeta}} x_{\ell_1 u \ell_2}}{\log w \cdot 2^{\zeta}} - \frac{\sum \mathbf{x}}{w} \right| \leqslant 2.$$

Proof. For some $i \in \{0,1\}^{\log w}$, let $x_i(\ell)$ be the input to wire i in layer $\log w + 1 \leqslant \ell \leqslant 2\log w$ in CCC_w^2. Fix $\ell_1 \in \{0,1\}^{\ell - 1 - \log w}$ and $\ell_2 \in \{0,1\}^{2\log w - \ell + 1 - \log\log w - \zeta}$. Let $u \in \{0,1\}^{\log\log w + \gamma}$ be arbitrary, but fixed. By definition of the balancers,

$$\begin{aligned}
x_{\ell_1 u \ell_2}(\ell) &= \frac{x_{\ell_1(\ell - 1 - \log w) u \ell_2}(\ell - 1) + x_{\ell_1 u \ell_2}(\ell - 1)}{2} + x_{\mathsf{b}_{\ell-1}}(\ell_1 u \ell_2) \\
&= \frac{1}{2} \cdot \left(\frac{x_{\ell_1(\ell - 2 - \log w)(\ell - 1 - \log w) u \ell_2}(\ell - 2)}{2} + \frac{x_{\ell_1(\ell - 2 - \log w) u \ell_2}(\ell - 2)}{2} + x_{\mathsf{b}_{\ell-2}}(\ell_1 u \ell_2) \right. \\
&\qquad \left. + \frac{x_{\ell_1(\ell - 1 - \log w) u \ell_2}(\ell - 2)}{2} + \frac{x_{\ell_1 u \ell_2}(\ell - 2)}{2} + x_{\mathsf{b}'_{\ell-2}}(\ell_1 u \ell_2) \right) + x_{\mathsf{b}_{\ell-1}}(\ell_1 u \ell_2),
\end{aligned}$$

where $x_{\mathsf{b}_{\ell'}}(\ell_1 u \ell_2)$ (and $x_{\mathsf{b}'_{\ell'}}(\ell_1 u \ell_2)$) denote balancers in some layer $\ell' < \ell$ on which wire $\ell_1 u \ell_2$ in layer ℓ depends. Expanding this formula up to layer $\ell - \log w + \log\log w + \zeta$

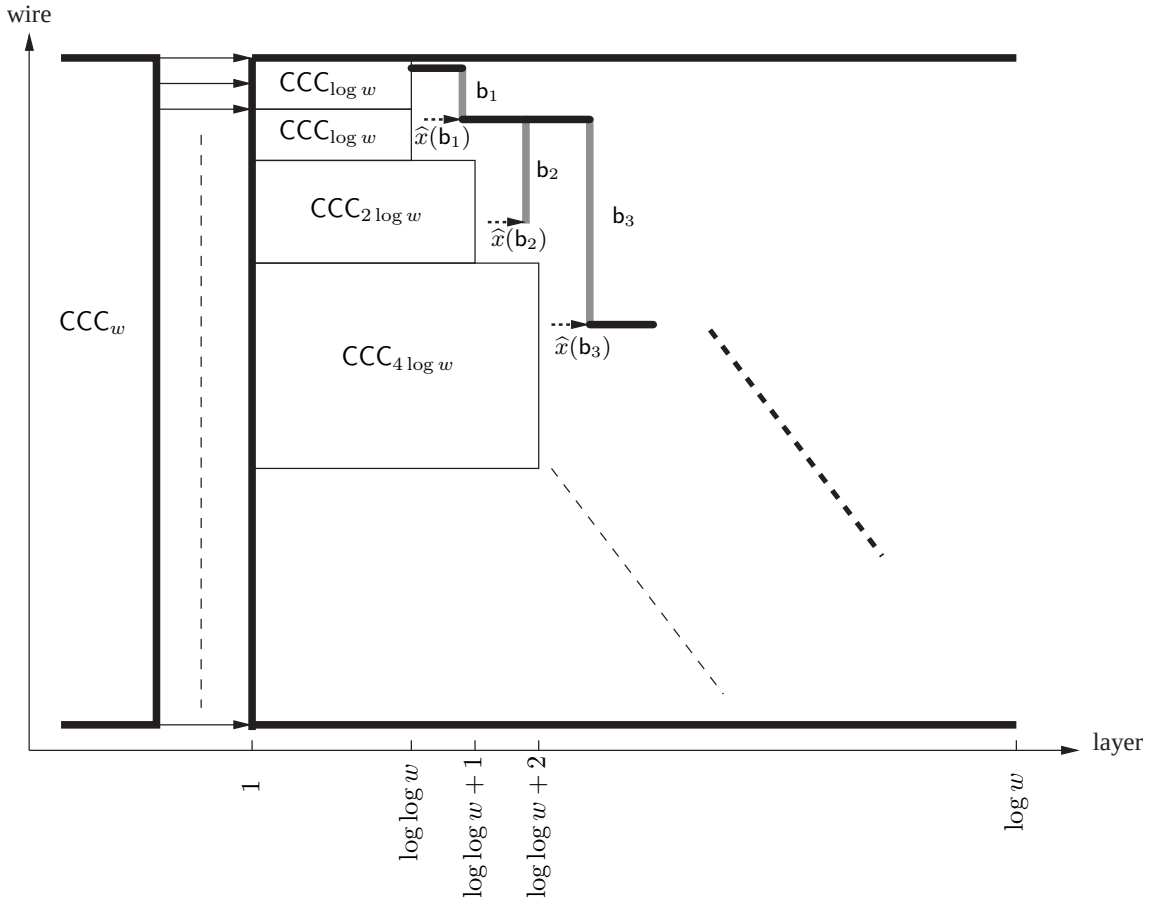

Fig. 7.5: *A sketch of the proof for the cascade of two* CCC_w *in group* $\mathcal{L}_1$*. The inputs* $\widehat{x}(\mathsf{b}_1), \widehat{x}(\mathsf{b}_2), \widehat{x}(\mathsf{b}_3), \ldots$ *on a path* π *from the top input wire in layer* $\log\log w$ *to some output wire in layer* l_ρ *are all independent (for fixed input at layer* 1*) and are output wires of a* $\mathsf{CCC}_{\log w}, \mathsf{CCC}_{2\log w}, \mathsf{CCC}_{4\log w}, \ldots$*, each of which gets the "correct" number of tokens up to some additive constant. Therefore, at least one of the balancers* $\mathsf{b}_1, \mathsf{b}_2, \mathsf{b}_3, \ldots$ *will drop the smoothness along* π*.*

(cf. [HT06a]) yields

$$x_{\ell_1 u \ell_2}(\ell) = \frac{1}{2^{\log w - \log\log w - \zeta}} \cdot \sum_{\substack{\widehat{\ell}_1 \in \{0,1\}^{\ell-1-\log w} \\ \widehat{\ell}_2 \in \{0,1\}^{2\log w - \ell + 1 - \log\log w - \zeta}}} x_{\widehat{\ell}_1 u \widehat{\ell}_2}(\ell - \log w + \log\log w + \zeta)$$
$$+ \sum_{k=\ell-1-\log w+\log\log w+\zeta+1}^{\ell-1} 2^{k-(\ell-1)} \sum_{\substack{\mathsf{b}_k \in k\text{: } \ell_1 u \ell_2 \\ \text{depends on } \mathsf{b}_k}} x_{\mathsf{b}_k}.$$

Summing over all $u \in \{0,1\}^{\log\log w+\zeta}$ and dividing by $\log w \cdot 2^\zeta$ gives

$$\begin{aligned}
&\frac{\sum_{u\in\{0,1\}^{\log\log w+\zeta}} x_{\ell_1 u \ell_2}}{\log w \cdot 2^\zeta} \\
&= \frac{\sum \mathbf{x}}{w} + \underbrace{\frac{1}{\log w \cdot 2^\zeta} \sum_{u\in\{0,1\}^{\log\log w+\zeta}} \sum_{k=\ell-\log w+\log\log w+\zeta}^{\ell-1} 2^{k-(\ell-1)} \sum_{\substack{\mathsf{b}_k\in k:\ \ell_1 u\ell_2 \\ \text{depends on } \mathsf{b}_k}} x_{\mathsf{b}_k}}_{X}.
\end{aligned}$$

Corresponding to X, we define

$$R = \frac{1}{\log w \cdot 2^\zeta} \sum_{u\in\{0,1\}^{\log\log w+\zeta}} \sum_{k=\ell-\log w+\log\log w+\zeta}^{\ell-1} 2^{k-(\ell-1)} \sum_{\substack{\mathsf{b}_k\in k:\ \ell_1 u\ell_2 \\ \text{depends on } \mathsf{b}_k}} r_{\mathsf{b}_k}.$$

By Lemma 7.6, $\mathbf{Pr}[\,|X| \geqslant 2\,] \leqslant 2 \cdot \mathbf{Pr}[\,|R| \geqslant 2\,]$. Note that for each pair of a $u \in \{0,1\}^{\log\log w+\zeta}$ and a layer k, where $\ell - 1 - \log w + \log\log w + \zeta + 1 \leqslant k \leqslant \ell - 1$, such that $\ell_1 u \ell_2$ depends on some balancer b_k of layer k, the random variable $\frac{1}{\log w \cdot 2^\zeta} \cdot 2^{k-(\ell-1)} \cdot r_{\mathsf{b}_k}$ has range $\{-\frac{1}{\log w} 2^{k-(\ell-1)-\zeta} \cdot \frac{1}{2}, +\frac{1}{\log w} 2^{k-(\ell-1)-\zeta} \cdot \frac{1}{2}\} = \{-\frac{1}{\log w} 2^{k-\ell-\zeta}, +\frac{1}{\log w} 2^{k-\ell-\zeta}\}$.

Lemma 7.18. *With the notation of Lemma 7.17,*

$$\sum_{u\in\{0,1\}^{\log\log w+\zeta}} \sum_{k=\ell-1-\log w+\log\log w+\zeta}^{\ell-1} \sum_{\substack{\mathsf{b}_k\in k:\ \ell_1 u\ell_2 \\ \text{depends on } \mathsf{b}_k}} \left(\frac{1}{\log w} 2^{k-\ell-\zeta} - (-\frac{1}{\log w} 2^{k-\ell-\zeta})\right)^2 \leqslant \frac{2}{\log w}.$$

Proof.

$$\begin{aligned}
&\sum_{u\in\{0,1\}^{\log\log w+\zeta}} \sum_{k=\ell-1-\log w+\log\log w+\zeta+1}^{\ell-1} \sum_{\substack{\mathsf{b}_k\in k:\ \ell_1 u\ell_2 \\ \text{depends on } \mathsf{b}_k}} \left(\frac{1}{\log w} 2^{k-\ell-\zeta} - (-\frac{1}{\log w} 2^{k-\ell-\zeta})\right)^2 \\
&= \sum_{u\in\{0,1\}^{\log\log w+\zeta}} \sum_{k=\ell-\log w+\log\log w+\zeta}^{\ell-1} \sum_{\substack{\mathsf{b}_k\in k:\ \ell_1 u\ell_2 \\ \text{depends on } \mathsf{b}_k}} \frac{1}{\log^2 w} \left(2^{k-\ell-\zeta+1}\right)^2 \\
&= \frac{1}{\log^2 w} \sum_{u\in\{0,1\}^{\log\log w+\zeta}} \sum_{k=\ell-\log w+\log\log w+\zeta}^{\ell-1} 2^{(\ell-1)-k} \cdot 2^{2k-2\ell-2\zeta+2} \\
&= \frac{2}{\log^2 w \cdot 2^{2\zeta}} \sum_{u\in\{0,1\}^{\log\log w+\zeta}} \sum_{k=1}^{\log w-\log\log w-\zeta} 2^{-k} \\
&\leqslant \frac{2}{\log^2 w \cdot 2^{2\zeta}} \cdot 2^{\log\log w+\zeta} \cdot 1 \leqslant \frac{2}{\log w},
\end{aligned}$$

as needed. □

By Hoeffding bound and Lemma 7.18, $\mathbf{Pr}\left[\,|X| \geqslant 2\,\right] \leqslant 2\cdot 2\cdot\exp(-\frac{2\cdot 2^2}{\frac{2}{\log w}}) \leqslant 4w^{-4}$. Hence,

$$\begin{aligned}
&\mathbf{Pr}\left[\bigvee_{\zeta=0}^{\log w-\log\log w}\ \bigvee_{\substack{\ell_1\in\{0,1\}^{\ell-1-\log w}\\ \ell_2\in\{0,1\}^{2\log w-\ell+1-\log\log w-\zeta}}}\left|\frac{\sum_{u\in\{0,1\}^{\log\log w+\zeta}} x_{\ell_1 u\ell_2}}{\log w\cdot 2^{\zeta}}-\frac{\sum\mathbf{x}}{w}\right|\leqslant 2\right]\\
\leqslant\ &\sum_{\zeta=0}^{\log w-\log\log w}\ \sum_{\substack{\ell_1\in\{0,1\}^{\ell-1-\log w}\\ \ell_2\in\{0,1\}^{2\log w-\ell+1-\log\log w-\zeta}}}\mathbf{Pr}\left[\left|\frac{\sum_{u\in\{0,1\}^{\log\log w+\zeta}} x_{\ell_1 u\ell_2}}{\log w\cdot 2^{\zeta}}-\frac{\sum\mathbf{x}}{w}\right|\leqslant 2\right]\\
\leqslant\ &\log w\cdot 2^{\ell-1-\log w}\cdot 2^{2\log w-\ell+1-\log\log w-\zeta}\cdot\frac{4}{w^4}\leqslant\frac{4}{w^3},
\end{aligned}$$

and the claim follows. □

In the remainder of the proof we will focus on the second cascaded CCC_w. Therefore, we denote the layers of this subnetwork by $1, 2, \ldots, \log w$.

Consider now again the second cascaded CCC_w with layers $1, 2, \ldots, \log w$. Consider groups of layers $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_{\frac{\log\log w}{2}-6}$ in this network, defined inductively as follows:

- For the basis case, $\mathcal{L}_1$ consists of layers $1, 2, \ldots, \lceil\frac{4\log w}{(\frac{1}{2}\log\log w-2)^2}\rceil + \log\log w$.
- Assume inductively that we have defined group $\mathcal{L}_{\rho-1}$, where $\rho > 2$.
- For the induction step, group $\mathcal{L}_\rho$ consists of the $\lceil\frac{4\log w}{(\frac{1}{2}\log\log w-1-\rho)^2}\rceil + \log\log w$ layers which immediately follow group $\mathcal{L}_{\rho-1}$.

Denote as ℓ_ρ the first layer in group $\mathcal{L}_\rho$, so, $\ell_1 = 1$. We observe:

Lemma 7.19. *For any w as in the statement of Theorem 7.16, the number of layers in groups $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_{\frac{\log\log w}{2}-6}$ is at most $\log w$.*

Proof. Clearly,

$$\begin{aligned}
&\sum_{\rho=1}^{\frac{\log\log w}{2}-6}\left(\left\lceil\frac{4\log w}{(\frac{1}{2}\log\log w-1-\rho)^2}\right\rceil+\log\log w\right)\\
\leqslant\ &\left(\sum_{\rho=1}^{\frac{\log\log w}{2}-6}\frac{4\log w}{(\frac{1}{2}\log\log w-1-\rho)^2}\right)+2(\log\log w)^2\\
\leqslant\ &4\log w\cdot\sum_{k=5}^{\infty}\frac{1}{k^2}+2(\log\log w)^2\\
=\ &4\log w\cdot\left(\frac{\pi^2}{6}-\sum_{k=1}^{4}\frac{1}{k^2}\right)+2(\log\log w)^2\\
\leqslant\ &0.9\log w+2(\log\log w)^2\leqslant\log w,
\end{aligned}$$

where the last inequality holds since $w\geqslant 2^{2^{12}}$ implies $2(\log\log w)^2\leqslant 0.1\log w$. □

Consider a path $\pi=(\mathsf{b}_1,\mathsf{b}_2,\ldots,\mathsf{b}_k)$ (of balancers) in subsequent layers. For each balancer b_r, $1<r\leqslant k$, $x(\mathsf{b}_r)$ is the input to balancer b_r from balancer b_{r-1} and $\widehat{x}(\mathsf{b}_r)$ is the other input to balancer b_r. ($x(\mathsf{b}_1)$ is arbitrarily among $x_1(\mathsf{b}_1)$ and $x_2(\mathsf{b}_1)$.) We now prove the following key lemma.

Lemma 7.20. *Consider an arbitrary, fixed path* $\pi=(\mathsf{b}_1,\mathsf{b}_2,\ldots,\mathsf{b}_{\mathsf{d}(\mathcal{L}_\rho)-\log\log w})$ *from an input wire in layer* $\log\log w$ *of group* $\mathcal{L}_\rho$ *to an output wire of* $\mathcal{L}_\rho$*. Then,*

$$\mathbf{Pr}\left[\bigvee_{\mathsf{b}\in\pi}\left|\widehat{x}(\mathsf{b}_r)-\frac{\sum\mathbf{x}}{w}\right|\leqslant\frac{1}{2}\log\log w+1-\rho\right]\geqslant 1-8\cdot w^{-3}.$$

Proof. We first prove a technical claim we need.

Claim. Consider an arbitrary fixed input $\mathbf{x}$ to $\mathcal{L}_\rho$. Then, $\{\widehat{x}(\mathsf{b}_1),\ldots,\widehat{x}(\mathsf{b}_{\mathsf{d}(\mathcal{L}_\rho)-\log\log w})\}$ is a set of independent random variables.

Proof. For fixed input x to $\mathcal{L}_\rho$, each random variable $\widehat{x}(\mathsf{b}_r)$, where $1\leqslant r\leqslant\mathsf{d}(\mathcal{L}_\rho)-\log\log w$ is determined by *(i)* the inputs to the balancers in layer 1 of $\mathcal{L}_\rho$ on which b_r depends, and *(ii)* the (randomly) chosen orientation of the balancers of $\mathcal{L}_\rho$ on which b_r depends. Observation 7.3 implies that the dependency sets (restricted to $\mathcal{L}_\rho$) of balancers $\mathsf{b}_r, 1\leqslant r\leqslant d(\mathcal{L}_\rho)-\log\log w$, are all disjoint. The claim follows. □

Claim. For any pair of $\ell_1\in\{0,1\}^{\log w-1}$ and $\ell_2\in\{0,1\}^{\log w-l_\rho-\log\log w-\zeta+1}$, where $\zeta\geqslant 0$, consider an arbitrary but fixed input vector $\mathbf{x}$ to $\mathcal{L}_\rho$ such that $\left|\frac{\sum_{u\in\{0,1\}^{\log\log w+\zeta}}x_{\ell_1 u\ell_2}}{2^\zeta\cdot\log w}-\frac{\sum\mathbf{x}}{w}\right|\leqslant 2$.

Then, for each balancer b_r, $1 \leqslant r \leqslant \mathsf{d}(\mathcal{L}_\rho) - \log\log w$,

$$\mathbf{Pr}\left[\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum \mathbf{x}}{w}\right| \geqslant \frac{1}{2}\log\log w + 1 - \rho\right] \;\leqslant\; 4 \cdot \exp\left(-\left(\frac{1}{2}\log\log w - 1 - \rho\right)^2\right).$$

Proof. Fix a balancer b_r, where $1 \leqslant r \leqslant \mathsf{d}(\mathcal{L}_\rho) - \log\log w$ in layer $1 \leqslant \ell(r) \leqslant \log w$. Let $i = \widehat{\ell_1(r)\, u(r)\, \ell_2(r)}$ and $i(\ell)$ be the input wires of b_r, where $\widehat{\ell_1(r)} \in \{0,1\}^{\ell_\rho - 1}$, $\widehat{u(r)} \in \{0,1\}^{\log\log w + r}$ and $\widehat{\ell_2(r)} \in \{0,1\}^{\log w - \ell_\rho - \log\log w - r + 1}$. Consider the restriction of group $\mathcal{L}_\rho$ to layers $\ell_\rho, \ell_\rho + 1, \ldots, \ell_\rho + \log\log w + r - 1$ and wires $\widehat{\ell_1(r)u\ell_2(r)}$, where $u \in \{0,1\}^{\log\log w + r}$. Lemma 7.4 implies that this restriction is a network $\mathsf{CCC}_{2^r \log w}$ with input wires $\{x_{\widehat{\ell_1(r)u\ell_2(r)}} \mid u \in \{0,1\}^{\log\log w + r}\}$. Hence, $\widehat{x}(\mathsf{b}_r)$ is some output wire of $\mathsf{CCC}_{2^r \log w}$; notice that the input vector to this network comes from the (arbitrary but fixed) input vector $\mathbf{x}$ to $\mathcal{L}_\rho$. Clearly, the triangle inequality and the assumption imply

$$\begin{aligned}
&\mathbf{Pr}\left[\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum \mathbf{x}}{w}\right| \geqslant \frac{1}{2}\log\log w + 1 - \rho\right] \\
&\leqslant \mathbf{Pr}\left[\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum_{u\in\{0,1\}^{\log\log w + r}} x_{\widehat{\ell_1(r)u\ell_2(r)}}(\ell_\rho)}{2^r \log w}\right| + \left|\frac{\sum_{u\in\{0,1\}^{\log\log w + r}} x_{\widehat{\ell_1(r)u\ell_2(r)}}(\ell_\rho)}{2^r \log w} - \frac{\sum \mathbf{x}}{w}\right| \right. \\
&\qquad \left. \geqslant \frac{1}{2}\log\log w + 1 - \rho\right] \\
&\leqslant \mathbf{Pr}\left[\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum_{u\in\{0,1\}^{\log\log w + r}} x_{\widehat{\ell_1(r)u\ell_2(r)}}(\ell_\rho)}{2^r \log w}\right| + 2 \geqslant \frac{1}{2}\log\log w + 1 - \rho\right] \\
&= \mathbf{Pr}\left[\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum_{u\in\{0,1\}^{\log\log w + r}} x_{\widehat{\ell_1(r)u\ell_2(r)}}(\ell_\rho)}{2^r \log w}\right| \geqslant \frac{1}{2}\log\log w - 1 - \rho\right] \\
&\leqslant 4 \cdot \exp\left(-\left(\frac{1}{2}\log\log w - 1 - \rho\right)^2\right),
\end{aligned}$$

where Lemma 7.7 was used for the last inequality. □

We continue with the proof of Lemma 7.20. Denote by $\mathcal{E}$ the event that

$$\forall \zeta \geqslant 0, \ell_1 \in \{0,1\}^{\ell_\rho},\ \ell_2 \in \{0,1\}^{\log w - \ell_\rho - \log\log w - \zeta}\ \left|\frac{\sum_{u\in\{0,1\}^{\log\log w + \zeta}} x_{\ell_1 u \ell_2}(\ell_\rho)}{2^\zeta \cdot \log w} - \frac{\sum \mathbf{x}}{w}\right| \leqslant 2.$$

So, $\neg\mathcal{E}$ denotes the event that $\exists \zeta \geqslant 0,\ \ell_1 \in \{0,1\}^{\ell_\rho},\ \ell_2 \in \{0,1\}^{\log w - \ell_\rho - \log\log w - \zeta}$: $\left|\frac{\sum_{u\in\{0,1\}^{\log\log w + \zeta}} x_{\ell_1 u \ell_2}(\ell_\rho)}{2^\zeta \cdot \log w} - \frac{\sum \mathbf{x}}{w}\right| > 2$. Recall that by Lemma 7.17, $\mathbf{Pr}[\,\neg\mathcal{E}\,] \leqslant \frac{4}{w^3}$ and define

$\alpha := \frac{1}{2}\log\log w + 1 - \rho$. Therefore, by the two preceding claims,

$$\begin{aligned}
&\mathbf{Pr}\left[\bigwedge_{b\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum \mathbf{x}}{w}\right| \geqslant \alpha\right] \\
=\ &\sum_{\mathbf{x}(\ell_\rho):\,\mathcal{E}} \mathbf{Pr}\left[\bigwedge_{b\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum \mathbf{x}}{w}\right| \geqslant \alpha \;\middle|\; \mathbf{x}(\ell_\rho) \text{ input }\right] \cdot \mathbf{Pr}\left[\mathbf{x}(\ell_\rho) \text{ input }\right] \\
&+ \sum_{\mathbf{x}(\ell_\rho):\,\neg\mathcal{E}} \mathbf{Pr}\left[\bigwedge_{b\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum \mathbf{x}}{w}\right| \geqslant \alpha \;\middle|\; \mathbf{x}(\ell_\rho) \text{ input}\right] \cdot \mathbf{Pr}\left[\mathbf{x}(\ell_\rho) \text{ input}\right] \\
\leqslant\ &\sum_{\mathbf{x}(\ell_\rho):\,\mathcal{E}} \left(\prod_{r=1}^{\mathsf{d}(\mathcal{L}_\rho - \log\log w)} \mathbf{Pr}\left[\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum \mathbf{x}}{w}\right| \geqslant \alpha \;\middle|\; x(\ell_\rho) \text{ input}\right]\right) \\
&\cdot \mathbf{Pr}\left[\mathbf{x}(\ell_\rho) \text{ input}\right] + \sum_{\mathbf{x}(\ell_\rho):\,\neg\mathcal{E}} 1 \cdot \mathbf{Pr}\left[\mathbf{x}(\ell_\rho) \text{ input}\right] \\
\leqslant\ &\sum_{\mathbf{x}(\ell_\rho):\,\mathcal{E}} \left(\prod_{r=1}^{\mathsf{d}(\mathcal{L}_\rho - \log\log w)} 4\exp\Big(-(\alpha-2)^2\Big)\right) \cdot \mathbf{Pr}\left[\mathbf{x}(\ell_\rho) \text{ is input}\right] + \frac{4}{w^3} \\
\leqslant\ &4\cdot\exp(-(\mathsf{d}(\mathcal{L}_\rho) - \log\log w)\cdot(\alpha-2)^2)\cdot 1 + \frac{4}{w^3} \\
=\ &4\cdot\exp\left(-\frac{4\log w}{(\alpha-2)^2}\cdot(\alpha-2)^2\right) + \frac{4}{w^3} \leqslant \frac{8}{w^3},
\end{aligned}$$

and Lemma 7.20 follows. □

Observation 7.21. *For any layer* $1 \leqslant \ell \leqslant \log_2 w$, $\max_{i\in[w]} x_i(\ell) \geqslant \max_{i\in[w]} x_i(\ell+1)$ *and* $\min_{i\in[w]} x_i(\ell) \leqslant \min_{i\in[w]} x_i(\ell+1)$.

Lemma 7.22. *Consider layers* ℓ *and* ℓ', $\ell < \ell'$, *with input vector* $\mathbf{x} = \mathbf{x}(\ell)$ *and output vector* $\mathbf{y} = \mathbf{y}(\ell')$, *respectively. Assume that (i) for every* $i \in [w]$, $x_i - \lceil\frac{\sum \mathbf{x}}{w}\rceil \leqslant \gamma$ *and (ii) for every path* $\pi = (\mathsf{b}_\ell, \mathsf{b}_{\ell+1}, \ldots, \mathsf{b}_{\ell'})$ *from layer* ℓ *to layer* ℓ', *there is at least one layer* r, *with* $\ell \leqslant r \leqslant \ell'$, *such that* $|\widehat{x}(\mathsf{b}_r) - \frac{\sum \mathbf{x}}{w}| \leqslant \gamma - 2$. *Then,* $y_i - \lceil\frac{\sum \mathbf{x}}{w}\rceil \leqslant \gamma - 1$ *for every* $i \in [w]$.

Proof. Seeking a contradiction, assume that there is an $i \in [w]$ such that $y_i - \lceil\frac{\sum \mathbf{x}}{w}\rceil = \gamma$. Let $\mathsf{b}_{\ell'}$ be the balancer in layer ℓ' with output wire y_i. By Observation 7.21 and assumption on the input vector $\mathbf{x}$ in layer $\ell \leqslant \ell'$, the two inputs must satisfy $x_1(\mathsf{b}_{\ell'}) \geqslant \lceil\frac{\sum \mathbf{x}}{w}\rceil + \gamma - 1$ and $x_2(\mathsf{b}_{\ell'}) \geqslant \lceil\frac{\sum \mathbf{x}}{w}\rceil + \gamma$ for an arbitrary ordering of the two input wires of $\mathsf{b}_{\ell'}$. Consequently, there is an output wire of a balancer $\mathsf{b}_{\ell'-1}$ in layer $\ell' - 1$ with output $\lceil\frac{\sum \mathbf{x}}{w}\rceil + \gamma$. Similarly, the two inputs to $\mathsf{b}_{\ell'-1}$ must satisfy $x_1(\mathsf{b}_{\ell'-1}) \geqslant \lceil\frac{\sum \mathbf{x}}{w}\rceil + \gamma - 1$ and $x_2(\mathsf{b}_{\ell'-1}) \geqslant \lceil\frac{\sum \mathbf{x}}{w}\rceil + \gamma$ for some ordering of the input wires. By induction, there is a path $\pi = (\mathsf{b}_\ell, \mathsf{b}_{\ell+1}, \ldots, \mathsf{b}_{\ell'})$ such that for all $\ell \leqslant r \leqslant \ell'$, $\widehat{x}(\mathsf{b}_r) - \lceil\frac{\sum \mathbf{x}}{w}\rceil \geqslant \gamma - 1$. This contradicts our assumption and the claim follows. □

This concludes the main technical part of the proof of Theorem 7.16. The next lemma combines our findings and shows that the "one-sided smoothness" drops in each group.

Lemma 7.23. *For an integer ρ, where $1 \leqslant \rho \leqslant \frac{\log\log w}{2} - 6$, consider the input and output vectors $\mathbf{x} = \mathbf{x}(\rho)$ and $\mathbf{y} = \mathbf{y}(\rho)$ respectively, to group $\mathcal{L}_\rho$. Let $\beta := \frac{1}{2}\log\log w + 2 - \rho$. Then,*

$$\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(y_k(\rho) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta\right)\right] \geqslant \mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(x_k(\rho) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta+1\right)\right] - \frac{8}{w}.$$

Proof. Let $\mathcal{P}$ be the set of all paths from the first layer of $\mathcal{L}_\rho$ to the last layer of $\mathcal{L}_\rho$. Clearly, $|\mathcal{P}| \leqslant w \cdot 2^{\mathsf{d}(\mathcal{L}_\rho)} \leqslant w \cdot 2^{\log w} \leqslant w^2$. Hence, by the union bound and Lemma 7.20,

$$\begin{aligned}\mathbf{Pr}\left[\bigvee_{\pi\in\mathcal{P}}\left(\bigwedge_{\mathsf{b}\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum\mathbf{x}}{w}\right| > \beta - 1\right)\right] &\leqslant \sum_{\pi\in\mathcal{P}}\mathbf{Pr}\left[\left(\bigwedge_{\mathsf{b}\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum\mathbf{x}}{w}\right| > \beta - 1\right)\right] \\ &\leqslant w^2 \cdot \frac{8}{w^3} = \frac{8}{w}. \end{aligned} \tag{7.2}$$

By Lemma 7.22, the event

$$\left(\bigwedge_{k\in[w]}\left(x_k - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta+1\right)\right) \bigwedge \left(\bigwedge_{\pi\in\mathcal{P}}\left(\bigvee_{\mathsf{b}\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum\mathbf{x}}{w}\right| \leqslant \beta - 1\right)\right)$$

implies $\bigwedge_{k\in[w]}\left(y_k - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta\right)$. Therefore, by the union bound

$$\begin{aligned}
&\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(y_k - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta\right)\right] \\
\geqslant\ &\mathbf{Pr}\left[\left(\bigwedge_{k\in[w]}\left(x_k - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta+1\right)\right) \bigwedge \left(\bigwedge_{\pi\in\mathcal{P}}\left(\bigvee_{\mathsf{b}\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum\mathbf{x}}{w}\right| \leqslant \beta - 1\right)\right)\right] \\
\geqslant\ &1 - \mathbf{Pr}\left[\bigvee_{k\in[w]}\left(x_k - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil > \beta+1\right)\right] - \mathbf{Pr}\left[\bigvee_{\pi\in\mathcal{P}}\left(\bigwedge_{\mathsf{b}\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum\mathbf{x}}{w}\right| > \beta - 1\right)\right] \\
=\ &\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(x_k - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta+1\right)\right] - \mathbf{Pr}\left[\bigvee_{\pi\in\mathcal{P}}\left(\bigwedge_{\mathsf{b}\in\pi}\left|\widehat{x}(\mathsf{b}_r) - \frac{\sum\mathbf{x}}{w}\right| > \beta - 1\right)\right] \\
\geqslant\ &\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(x_k - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \beta+1\right)\right] - \frac{8}{w},
\end{aligned}$$

where 7.2 was used in the last inequality. □

We are now ready to complete the proof of the theorem. For some $1 \leqslant \rho \leqslant \frac{\log\log w}{2} - 6$, let $\mathbf{x}(\rho), \mathbf{y}(\rho)$ be the input vector and output vector of $\mathcal{L}_\rho$, respectively. We shall prove by induction that for every $1 \leqslant \rho \leqslant \frac{\log\log w}{2} - 6$,

$$\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(y_k(\rho) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \frac{1}{2}\log\log w + 2 - \rho\right)\right] \geqslant 1 - \frac{8\rho+1}{w}.$$

For the basis case where $\rho = 1$,

$$\begin{aligned}
&\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(y_k(1) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \frac{1}{2}\log\log w + 1\right)\right] \\
\geqslant\ &\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(x_k(1) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \frac{1}{2}\log\log w + 2\right)\right] - \frac{8}{w} && \text{(by Lemma 7.23)} \\
\geqslant\ &\left(1 - \frac{1}{w}\right) - \frac{8}{w} = 1 - \frac{9}{w} && \text{(by 7.1 (in Theorem 7.9))}
\end{aligned}$$

Assume inductively that the claim holds for $\rho - 1$. For the induction step, we conclude by using Lemma 7.23 and the induction hypothesis that

$$\begin{aligned}
&\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(y_k(\rho) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \frac{1}{2}\log\log w + 2 - \rho\right)\right] \\
\geqslant\ &\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(x_k(\rho) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant \frac{1}{2}\log\log w + 2 - \rho + 1\right)\right] - \frac{8}{w} \\
\geqslant\ &\left(1 - \frac{8(\rho-1)+1}{w}\right) - \frac{8}{w} = 1 - \frac{8\rho+1}{w},
\end{aligned}$$

and the induction is complete. This implies that for $\rho = \frac{1}{2}\log\log w - 6$,

$$\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(y_k\left(\frac{\log\log w}{2} - 6\right) - \left\lceil\frac{\sum\mathbf{x}}{w}\right\rceil \leqslant 8\right)\right] \geqslant 1 - \frac{8(\frac{\log\log w}{2} - 6) + 1}{w}.$$

With symmetrical arguments it follows that

$$\mathbf{Pr}\left[\bigwedge_{k\in[w]}\left(y_k\left(\frac{\log\log w}{2} - 6\right) - \left\lfloor\frac{\sum\mathbf{x}}{w}\right\rfloor \geqslant -8\right)\right] \geqslant 1 - \frac{8(\frac{\log\log w}{2} - 6) + 1}{w}.$$

Hence, the output of the second CCC_w is 17-smooth with probability $1 - \frac{8 \log\log w - 94}{w}$. □

7.5 Improbability of 1-Smoothing

Roughly speaking, we show that no randomized smoothing network of sublinear depth is 1-smoothing.

Theorem 7.24. *Every network B_w is 1-smoothing with probability at most $\frac{\mathsf{d}(B_w)}{w-1}$.*

Proof. Fix a randomized 1-smoothing network B_w. Choose two distinct integers $0 \leqslant i, j \leqslant w-1$ uniformly at random. Define the input vector

$$\mathbf{x}_{i,j} = (1, \ldots, 1, \underbrace{0}_{\text{component } i}, 1, \ldots, 1, \underbrace{2}_{\text{component } j}, 1, \ldots, 1).$$

Notice that $\mathbf{x}_{i,j}$ is a random variable. For each layer $\ell, 1 \leqslant \ell \leqslant \mathsf{d}(\mathsf{B}_w)$, denote as $\mathcal{E}_\ell$ the event that there is a balancer b in layer ℓ such that the inputs to b are 0 and 2. Clearly, $\mathsf{B}_w(\mathbf{x}_{i,j})$ is 1-smooth if and only if there is a layer ℓ such that $\mathcal{E}_\ell$ occurs. Using conditional probabilities,

$$\begin{aligned}
\mathbf{Pr}\left[\, \mathsf{B}_w(\mathbf{x}) \text{ is 1-smooth} \,\right] &= \sum_{\widehat{i},\widehat{j} \in [w], \widehat{i} \neq \widehat{j}} \mathbf{Pr}\left[\mathbf{x} = \mathbf{x}_{\widehat{i},\widehat{j}} \right] \cdot \mathbf{Pr}\left[\mathsf{B}_w(\mathbf{x}) \text{ is 1-smooth} \mid \mathbf{x} = \mathbf{x}_{\widehat{i},\widehat{j}} \right] \\
&= \sum_{\widehat{i},\widehat{j} \in [w], \widehat{i} \neq \widehat{j}} \frac{1}{w(w-1)} \cdot \mathbf{Pr}\left[\mathsf{B}_w(\mathbf{x}) \text{ is 1-smooth} \mid \mathbf{x} = \mathbf{x}_{\widehat{i},\widehat{j}} \right].
\end{aligned}$$

Lemma 7.25. *For each layer $\ell \geqslant 1$, $\mathbf{Pr}\left[\, \mathcal{E}_\ell \,\right] \leqslant \frac{1}{w-1}$.*

Proof. We first prove the following lemma.

Lemma 7.26. *Consider an arbitrary pair $i, j \in [w], i \neq j$. For each layer ℓ, $1 \leqslant \ell \leqslant \mathsf{d}(\mathsf{B}_w)$, $\mathbf{Pr}\left[\, \mathbf{x}(\ell) = \mathbf{x}_{i,j} \,\right] \leqslant \frac{1}{w(w-1)}$.*

Proof. By induction on ℓ. For the basis case, take $\ell = 1$. By construction of the input vector, $\mathbf{Pr}\left[\, \mathbf{x}(\ell) = \mathbf{x}_{i,j} \,\right] = \frac{1}{w(w-1)}$. Assume inductively that the claim holds for all layers $\ell' \leqslant \ell - 1$, where $\ell \geqslant 2$. For the induction step, consider layer ℓ. By the law of conditional probabilities and the induction hypothesis,

$$\begin{aligned}
\mathbf{Pr}\left[\, \mathbf{x}(\ell) = \mathbf{x}_{i,j} \,\right] &= \sum_{\widehat{i},\widehat{j} \in [w], \widehat{i} \neq \widehat{j}} \mathbf{Pr}\left[\mathbf{x}(\ell - 1) = \mathbf{x}_{\widehat{i},\widehat{j}} \right] \cdot \mathbf{Pr}\left[\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid \mathbf{x}(\ell - 1) = \mathbf{x}_{\widehat{i},\widehat{j}} \right] \\
&\leqslant \frac{1}{w(w-1)} \cdot \sum_{\widehat{i},\widehat{j} \in [w], \widehat{i} \neq \widehat{j}} \mathbf{Pr}\left[\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid \mathbf{x}(\ell - 1) = \mathbf{x}_{\widehat{i},\widehat{j}} \right].
\end{aligned}$$

To simplify the notation, we shall write in the following $\mathbf{x}_{\widehat{i},\widehat{j}}$ denoting the event $\mathbf{x}(\ell - 1) = \mathbf{x}_{\widehat{i},\widehat{j}}$. We proceed by a case analysis.

1. i is connected to some balancer $\mathsf{b}_i = \{i, i'\} \in \ell$ (that means b_i has input wires $x_i(\ell)$, $x_{i'}(\ell)$) while j is not connected to any balancer located in layer ℓ. Then, $\mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid \mathbf{x}_{\widehat{i},\widehat{j}}\,\right] = 0$ unless $\widehat{i} \in \{i, i'\}$ and $\widehat{j} = j$. Hence,

$$\begin{aligned}
& \mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j}\,\right] \\
\leqslant\ & \frac{1}{w(w-1)} \cdot \Big(\mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid \mathbf{x}_{i,j}\,\right] + \mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid \mathbf{x}_{i',j}\,\right]\Big) \\
=\ & \frac{1}{w(w-1)} \cdot \Big(\mathbf{Pr}\left[\,x_i(\ell) = x_i(\ell-1) \mid \mathbf{x}_{i,j}\,\right] \cdot \mathbf{Pr}\left[\,x_j(\ell) = x_j(\ell-1) \mid \mathbf{x}_{i,j}\,\right] \\
& + \mathbf{Pr}\left[\,x_i(\ell) = x_{i'}(\ell-1) \mid \mathbf{x}_{i',j}\,\right] \cdot \mathbf{Pr}\left[\,x_j(\ell) = x_j(\ell-1) \mid \mathbf{x}_{i',j}\,\right]\Big) \\
=\ & \frac{1}{w(w-1)} \cdot \left(\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 1\right) = \frac{1}{w(w-1)}.
\end{aligned}$$

2. j is connected to some balancer $\mathsf{b}_j \in \ell$, while i is not. This is the same case as before.

3. i and j are connected to different balancers $\mathsf{b}_i = \{i, i'\}, \mathsf{b}_j = \{j, j'\} \in \ell$, respectively. Then,

$$\begin{aligned}
& \mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j}\,\right] \\
\leqslant\ & \frac{1}{w(w-1)} \cdot \Big(\mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid \mathbf{x}_{i,j}\,\right] + \mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid \mathbf{x}_{i',j}\,\right] \\
& + \mathbf{Pr}\left[\,\mathbf{x}(\ell) = x_{i,j} \mid x_{i,j'}\,\right] + \mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j} \mid x_{i',j'}\,\right]\Big) \\
=\ & \frac{1}{w(w-1)} \cdot \Bigg(\mathbf{Pr}\left[\,x_i(\ell) = x_i(\ell-1) \mid \mathbf{x}_{i,j}\,\right] \cdot \mathbf{Pr}\left[\,x_j(\ell) = x_j(\ell-1) \mid \mathbf{x}_{i,j}\,\right] \\
& + \mathbf{Pr}\left[\,x_i(\ell) = x_{i'}(\ell-1) \mid \mathbf{x}_{i',j}\,\right] \cdot \mathbf{Pr}\left[\,x_j(\ell) = x_j(\ell-1) \mid \mathbf{x}_{i',j}\,\right] \\
& + \mathbf{Pr}\left[\,x_i(\ell) = x_i(\ell-1) \mid \mathbf{x}_{i,j'}\,\right] \cdot \mathbf{Pr}\left[\,x_j(\ell) = x_{j'}(\ell-1) \mid \mathbf{x}_{i,j'}\,\right] \\
& + \mathbf{Pr}\left[\,x_i(\ell) = x_{i'}(\ell-1) \mid \mathbf{x}_{i',j'}\,\right] \cdot \mathbf{Pr}\left[\,x_j(\ell) = x_{j'}(\ell-1) \mid \mathbf{x}_{i',j'}\,\right]\Bigg) \\
=\ & \frac{1}{w(w-1)} \cdot \left(\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2}\right) = \frac{1}{w(w-1)}.
\end{aligned}$$

4. i and j are connected to the same balancer in layer ℓ. Then, $\mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j}\,\right] = 0$ as a balancer cannot return 0 and 2 tokens on its two output wires.

5. i and j are not connected to a balancer in layer ℓ. Then, by induction hypothesis $\mathbf{Pr}\left[\,\mathbf{x}(\ell) = \mathbf{x}_{i,j}\,\right] = \mathbf{Pr}\left[\,\mathbf{x}_{i,j}\,\right] \leqslant \frac{1}{w(w-1)}$.

□

By Lemma 7.26 and the union bound,

$$\begin{aligned}\mathbf{Pr}\left[\,\mathcal{E}_\ell\,\right] &= \mathbf{Pr}\left[\bigvee_{\mathsf{b}\in\ell}\left(\{x_1(\mathsf{b}),x_2(\mathsf{b})\}=\{0,2\}\right)\right]\\ &\leqslant \sum_{\mathsf{b}\in\ell}\Big(\mathbf{Pr}\left[\,(x_1(\mathsf{b})=0\wedge x_2(\mathsf{b})=2)\,\right]+\mathbf{Pr}\left[\,(x_1(\mathsf{b})=2\wedge x_2(\mathsf{b})=0)\,\right]\Big)\\ &\leqslant \sum_{\mathsf{b}\in\ell}\left(\frac{1}{w(w-1)}+\frac{1}{w(w-1)}\right)\leqslant\frac{w}{2}\cdot\frac{2}{w(w-1)}=\frac{1}{w-1},\end{aligned}$$

as needed. □

Finally, by Lemma 7.25

$$\mathbf{Pr}\left[\,\mathsf{B}_w(\mathbf{x})\text{ is 1-smooth }\right] \;=\; \mathbf{Pr}\left[\bigvee_{\ell=1}^{\mathsf{d}(\mathsf{B}_w)}\mathcal{E}_\ell\right]\leqslant\sum_{\ell=1}^{\mathsf{d}(\mathsf{B}_w)}\mathbf{Pr}\left[\,\mathcal{E}_\ell\,\right]\leqslant\mathsf{d}(\mathsf{B}_w)\cdot\frac{1}{w-1}.$$

Hence, there is a pair $\widehat{i},\widehat{j}\in[w],\widehat{i}\neq\widehat{j}$ such that $\mathbf{Pr}\left[\,\mathsf{B}_w(x)\text{ is 1-smooth }\middle|\; x=x_{\widehat{i},\widehat{j}}\,\right]\leqslant\frac{\mathsf{d}(\mathsf{B}_w)}{w-1}$, which is equivalent to $\mathbf{Pr}\left[\,\mathsf{B}_w(x_{\widehat{i},\widehat{j}})\text{ is 1-smooth }\right]\leqslant\frac{\mathsf{d}(\mathsf{B}_w)}{w-1}$. □

7.6 Dimension-Exchange Balancing on Hypercubes

We shall prove that our results can be also used for the analysis of the following randomized dimension-exchange algorithm for load balancing on hypercubes $\mathsf{DE_w}$ given in Figure 7.6. The following simple observation shows that all results for the CCC_w (or Block_w) can be directly applied to this dimension-exchange algorithm.

Lemma 7.27. *Fix some input vector* $\mathbf{x}$*. The distribution of the output vector* $\mathbf{z}$ *of* $\mathsf{DE_w}$ *with input vector* x *is identical to the distribution of the output vector* $\mathbf{y}$ *of a* CCC_w *with input vector* $\mathbf{x}$*.*

Proof. Let $\tilde{\mathbf{y}}(i)$ be the load vector after the i-th iteration of $\mathsf{DE_w}$ and recall that $\mathbf{y}(i)$ is the output vector of the i-th layer of a CCC_w. We shall prove by induction, that $\mathbf{y}(\ell)\stackrel{D}{=}\tilde{\mathbf{y}}(\ell)$ for all $1\leqslant\ell\leqslant\log_2 w-1$. By assumption, the two input vectors $\mathbf{y}(0)=\mathbf{x}$ and $\tilde{\mathbf{y}}(0)=\tilde{\mathbf{x}}$ are identical. For the induction step, we first look at $\tilde{\mathbf{y}}(\ell)$. Fix a wire $i\in\{0,1\}^{\log w}$. By definition, we have with probability 1/2,

$$y_i(\ell)=\left\lfloor\frac{y_i(\ell-1)+y_{i(\ell)}(\ell-1)}{2}\right\rfloor\quad\text{and}\quad y_{i(\ell)}=\left\lceil\frac{y_i(\ell-1)+y_{i(\ell)}(\ell-1)}{2}\right\rceil,$$

RANDOMIZED DIMENSION-EXCHANGE LOAD BALANCING ON HYPERCUBES DE_w

Input: Load vector $\mathbf{x} = (x_0, x_1, \ldots, x_{w-1})$
Output: Load vector $\mathbf{z} = (z_0, z_1, \ldots, z_{w-1})$

for $k = 1$ to $\log w$ **do**
  for all $i \in \{0,1\}^{\log_2 w}, i_k = 1$ **do**
    $ave \leftarrow \frac{x_i + x_{i(k)}}{2}$
    choose $p \in \{0,1\}$ uniformly at random
    if $p = 0$ **then**
      $x_i \leftarrow \lceil ave \rceil,\ x_{i(k)} = \lfloor ave \rfloor$
    end if
    if $p = 1$ **then**
      $x_i \leftarrow \lfloor ave \rfloor,\ x_{i(k)} = \lceil ave \rceil$
    end if
  end for
end for
return $\mathbf{z} \leftarrow \mathbf{x}$

Fig. 7.6: *Description of the Dimension-Exchange Load Balancing Algorithm*

or otherwise,

$$y_i(\ell) = \left\lceil \frac{y_i(\ell-1) + y_{i(\ell)}(\ell-1)}{2} \right\rceil \quad \text{and} \quad y_{i(\ell)} = \left\lfloor \frac{y_i(\ell-1) + y_{i(\ell)}(\ell-1)}{2} \right\rfloor,$$

independently of all other wires $i' \in \{0,1\}^{\log w}, i' \neq i, i' \neq i(\ell)$. Replacing each occurrence of $\mathbf{y}$ by $\tilde{\mathbf{y}}$, we obtain the corresponding law for $\tilde{\mathbf{y}}(\ell)$. Since by induction hypothesis $\mathbf{y}(\ell-1) \stackrel{D}{=} \tilde{\mathbf{y}}(\ell-1)$ holds, the induction step $\mathbf{y}(\ell) \stackrel{D}{=} \tilde{\mathbf{y}}(\ell)$ follows. □

Hence, by combining this lemma to Theorem 7.9 we conclude that one execution of the DE_w-algorithm suffices for $\log\log w + 3$ smoothing with high probability. Moreover, by Theorem 7.16 we find that only two subsequent executions of the DE_w-algorithm are sufficient for 17-smoothing with high probability.

7.7 Conclusion

In this chapter, we presented a thorough study of the impact of randomization in smoothing networks. Specifically, we assumed that balancers are oriented independently and uniformly at random, and we investigated the impact of this assumption on the smoothness of the network's output that can be achieved with high probability. We proved a tight (up to a small additive constant) bound of $\log\log w + \Theta(1)$ on the smoothness of the popular block network. As our main theoretical result, we established an upper bound of 17 on the smoothness of the cascade of two block networks. Finally, we proved that it is impossible

to obtain a 1-smoothing randomized network of sublinear depth. Our results reveal the full power of randomization in smoothing networks: randomization can be employed in a practical network to yield a constant upper bound on smoothness. Still, our results leave open a number of interesting questions.

- Is the cascade of a small number of block networks a (randomized) 2-smoothing network? We strongly believe that this is the case and could be shown using some extensions of the techniques presented here. However, we conjecture that the cascade of only two blocks is already a 2-smoothing network. To settle this stronger conjecture, new methods probably have to be developed.
- Another interesting problem is to derandomize our result for the cascade of two block networks. So, can we find deterministic (explicit) initializations for the cascade of two (or more) block networks which make it a $\mathcal{O}(1)$-smoothing network for *every* input?
- Finally, our approach gives extremely good bounds for a randomized dimension-exchange algorithm on hypercubes. Can we prove similarly strong bounds for general networks with, say, good expansion properties?

BIBLIOGRAPHY

[AAK+08] N. Alon, C. Avin, M. Koucky, G. Kozma, Z. Lotker, and M.R. Tuttle. Many Random Walks are faster than one. In *20th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'08)*, 2008. to appear.

[AAMR93] W. Aiello, B. Awerbuch, B.M. Maggs, and S. Rao. Approximate load balancing on dynamic and asynchronous networks. In *25th Annual ACM Symposium on Theory of Computing (STOC'93)*, pages 632–641, 1993.

[ABKU99] Y. Azar, A.Z. Broder, A.R. Karlin, and E. Upfal. Balanced allocations. *SIAM Journal on Computing*, pages 180–200, 1999.

[ABS04] N. Alon, I. Benjamini, and A. Stacey. Percolation on finite graphs and isoperimetric inequalities. *The Annals of Probability*, 32(3):1727–1745, 2004.

[AE06] C. Avin and G. Ercal. On the Cover Time and Mixing Time of Random Geometric Graphs. *Theoretical Computer Science*, 380(1-2):2–22, 2006.

[AF02] D.J. Aldous and J.A. Fill. *Reversible Markov Chains and Random Walks on Graphs*. (in preparation, draft availabe at http://www.stat.berkeley.edu/aldous/RWG/book.html), 2002.

[AHS94] J. Aspnes, M. Herlihy, and N. Shavit. Counting networks. *Journal of the ACM*, 41(5):1020–1048, 1994.

[AK89] S.B. Akers and B. Krishnamurthy. A Group-Theoretic Model for Symmetric Innterconnection Networks. *IEEE Transactions on Computers*, 38(4):555–565, 1989.

[AKL+79] R. Aleliunas, R.M. Karp, R.J. Lipton, L. Lovász, and C. Rackoff. Random Walks, Universal Traversal Sequences, and the Complexity of Maze Problems. In *20th Annual IEEE Symposium on Foundations of Computer Science (FOCS'79)*, pages 218–223, 1979.

[AKL08] C. Avin, M. Koucky, and Z. Lotker. How to Explore a Fast-Changing World. In *35th International Colloquium on Automata, Languages, and Programming (ICALP'08)*, 2008. to appear.

[AKS83] M. Ajtai, J. Komlós, and E. Szemerédi. An $\mathcal{O}(n \log n)$ sorting network. *Combinatorica*, 3:1–19, 1983.

[Ald83] D. J. Aldous. On the Time Taken by Random Walks on Finite Groups to Visit Every State. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, pages 361–374, 1983.

[Alo95] N. Alon. Tools from higher algebra. In R. Graham, M. Grötschel, and L. Lovász, editors, *Handbook of Combinatorics*, pages 1749–1783. Elsevier Science, 1995.

[Arn03] H. Arndt. *Loadbalancing auf Parallelrechnern mit Hilfe endlicher Dimension-Exchange-Verfahren*. PhD thesis, Bergische Universität Wuppertal, Fachbereich Mathematik, 2003.

[AS00] N. Alon and J. Spencer. *The Probabilistic Method*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons, 2nd edition, 2000.

[AVY94] W. Aiello, R. Venkatesan, and M. Yung. Coins, Weights and Contention in balancing networks. In *13th Annual ACM-SIGOPT Principles of Distributed Computing (PODC'94)*, pages 193–205, 1994.

[Bab95] L. Babai. Automorphism groups, isomorphism, reconstruction. In R. Graham, M. Grötschel, and L. Lovász, editors, *Handbook of Combinatorics*, pages 1447–1540. Elsevier Science, 1995.

[Bat68] K.E. Batcher. Sorting networks and their applications. In *Proceedings of the AFIPS Joint Computer Conference*, pages 334–338, 1968.

[BBC+06] A. Bagchi, A. Bhargava, A. Chaudhary, D. Eppstein, and C. Scheideler. The effect of faults on network expansion. *Theory of Computing Systems*, 39(6):903–928, 2006.

[BBCS05] N. Berger, C. Borgs, J.T. Chayes, and A. Saberi. On the Spread of Viruses on the Internet. In *16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'05)*, pages 301–310, 2005.

[BBHM05] I. Benjamini, N. Berger, C. Hoffmann, and E. Mossel. Mixing times of the biased card shuffling and the asymmetric exclusion process. *Transactions of the American Mathematical Society*, 357:3013–3029, 2005.

[BGPS06] 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.

[BH05] L. Babai and T.P. Hayes. Near-independence of permutations and an almost sure polynomial bound on the diameter of the symmetric group. In *16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'05)*, pages 1057–1066, 2005.

[BK89] A. Broder and A. Karlin. Bounds on the cover time. *Journal of Theoretical Probability*, 2(1):101–120, 1989.

[BK97] B. Bollobás and Y. Kohayakawa. On Richardson's model on the hypercube. In Bela Bollobás and A.G. Thomason, editors, *Combinatorics, Geometry and Probability*, pages 129–137. Cambridge University Press, 1997.

[BKRU94] A. Broder, A.R. Karlin, P. Raghavan, and E. Upfal. Trading space for time in undirected $s-t$-connectivity. *SIAM Journal on Computing*, 23(2):324–334, 1994.

[BT97] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, 1997.

[CDFS08] J.N. Cooper, B. Doerr, T. Friedrich, and J. Spencer. Deterministic random walks on regular trees. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'08)*, pages 766–772, 2008.

[CDST07] J.N. Cooper, B. Doerr, J. Spencer, and G. Tardos. Deterministic random walks on the integers. *European Journal of Combinatorics*, 28:990–995, 2007.

[CF05] C. Cooper and A. Frieze. The Cover Time of Random Regular Graphs. *SIAM Journal of Discrete Mathematics*, 18(4):728–740, 2005.

[CIA08] CIA. *The World Factbook*. https://www.cia.gov/library/publications/the-world-factbook/rankorder/2153.rank.html, 2008.

[CL07] F.K. Chung and L. Lu. Concentration inequalities and martingale inequalities: A survey. *Internet Mathematics*, 3(1):79–127, 2007.

[CRR+97] A.K. Chandra, P. Raghavan, W.L. Ruzzo, R. Smolensky, and P. Tiwari. The Electrical Resistance of a Graph Captures its Commute and Cover Times. *Computational Complexity*, 6(4):312–340, 1997.

[CS06] J. N. Cooper and J. Spencer. Simulating a random walk with constant error. *Combinatorics, Probabability & Computing*, 15:815–822, 2006.

[CW89] A. Cohen and A. Wigderson. Dispersers, deterministic amplification, and weak random sources. In *30th Annual IEEE Symposium on Foundations of Computer Science (FOCS'89)*, pages 14–19, 1989.

[Cyb89] G. Cybenko. Load balancing for distributed memory multiprocessors. *Journal of Parallel and Distributed Computing*, 7:279–301, 1989.

[DF06] B. Doerr and T. Friedrich. Deterministic random walks on the two-dimensional grid. In *17th International Symposium on Algorithms and Computation (ISAAC'06)*, pages 474–483, 2006.

[DFM99] R. Diekmann, A. Frommer, and B. Monien. Efficient schemes for nearest neighbor load balancing. *Parallel Computing*, 25(7):789–812, 1999.

[DFS08] B. Doerr, T. Friedrich, and T. Sauerwald. Quasirandom Rumor Spreading. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'08)*, pages 773–781, 2008.

[DGH+87] 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 Annual ACM-SIGOPT Principles of Distributed Computing (PODC'87)*, pages 1–12, 1987.

[Dia88] P. Diaconis. *Group Representations in Probability and Statistics*, volume 11. Lecture notes-Monograph Series, 1988.

[Dia96] P. Diaconis. The cutoff phenomenon in finite markov chains. *Proceedings of the National Academy of Sciences of the United States of America*, 93:1659–1664, 1996.

[DPRS89] M. Dowd, Y. Perl, L. Rudolph, and M. Saks. The periodic balanced sorting network. *Journal of the ACM*, 36(4):738–757, 1989.

[DS81] P. Diaconis and M. Shahshahani. Generating a random permutation with random transpositions. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 57:159–179, 1981.

[DSC06] P. Diaconis and L. Saloff-Coste. Separation Cut-Offs for Birth and Death Chains. *Annals of Applied Probability*, 16(4):2098–2122, 2006.

[DSV03] G. Davidoff, P. Sarnak, and A. Valette. *Elementary Number Theory, Group Theory and Ramanujan Graphs*. Cambridge University Press, 2003.

[EK06] M. Elkin and G. Kortsarz. A combinatorial logarithmic approximation algorithm for the directed telephone broadcast problem. *SIAM Journal on Computing*, 35(3):672–689, 2006.

[Els02] R. Elsässer. *Spectral Methods for Efficient Load Balancing Strategies*. PhD thesis, Fachbereich Mathematik/Informatik, University of Paderborn, 2002.

[Els06] R. Elsässer. On the Communication Complexity of Randomized Broadcasting in Random-like Graphs. In *18th Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA'06)*, pages 148–157, 2006.

[ELS07] R. Elsässer, U. Lorenz, and T. Sauerwald. Agent-based Randomized Broadcasting in Large Networks. *Discrete Applied Mathematics*, 155(2):150–160, 2007.

[EMS06] R. Elsässer, B. Monien, and S. Schamberger. Distributing unit size workload packages in heterogenous networks. *Journal of Graph Algorithms & Applications*, 10(1):51–68, 2006.

[ER59] P. Erdős and A. Rényi. On random graphs. *Publ. Math. Debrecen*, 6:290–297, 1959.

[ES05] R. Elsässer and T. Sauerwald. On Randomized Broadcasting in Star Graphs. In *31st Workshop of Graph-Theoretic Concepts in Computer Science (WG'05)*, pages 307–318, 2005.

[ES07] R. Elsässer and T. Sauerwald. Broadcasting vs. Mixing and Information Dissemination on Cayley Graphs. In *24th International Symposium on Theoretical Aspects of Computer Science (STACS'07)*, pages 163–174, 2007.

[ES08a] R. Elsässer and T. Sauerwald. On the Power of Memory in Randomized Broadcasting. In *19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'08)*, pages 218–227, 2008.

[ES08b] R. Elsässer and T. Sauerwald. On the Runtime and Robustness of Randomized Broadcasting. *Theoretical Computer Science*, 2008. to appear.

[Fei95a] U. Feige. A Tight Lower Bound for the Cover Time of Random Walks on Graphs. *Random Structures and Algorithms*, 6(4):433–438, 1995.

[Fei95b] U. Feige. A Tight Upper Bound for the Cover Time of Random Walks on Graphs. *Random Structures and Algorithms*, 6(1):51–54, 1995.

[Fei97a] U. Feige. A Spectrum of Time-space Tradeoffs for Undirected s-t Connectivity. *Journal of Computer and System Sciences*, 54(2):305–316, 1997.

[Fei97b] U. Feige. Collecting Coupons on Trees, and the Cover Time of Random Walks. *Computational Complexity*, 6(4):341–356, 1997.

[FG85] A. Frieze and G. Grimmett. The shortest-path problem for graphs with random-arc-lengths. *Discrete Applied Mathematics*, 10:57–77, 1985.

[Fil91] J.A. Fill. Eigenvalue bounds on convergence to stationarity for nonreversible markov chains, with an application to the exclusion process. *The Annals of Applied Probabilty*, 1(1):62–87, 1991.

[FOW85] L. Flatto, A.M. Odlyzko, and D.B. Wales. Random shuffles and group representations. *The Annals of Probability*, 13:154–178, 1985.

[FP93] 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.

[FPRU90] U. Feige, D. Peleg, P. Raghavan, and E. Upfal. Randomized Broadcast in Networks. *Random Structures and Algorithm*, 1(4):447–460, 1990.

[GLM+99] B. Ghosh, F.T. Leighton, B.M. Maggs, S. Muthukrishnan, C.G. Plaxton, R. Rajaraman, A.W. Richa, R.E. Tarjan, and D. Zuckerman. Tight analyses of two local load balancing algorithms. *SIAM Journal on Computing*, 29(1):29–64, 1999.

[GM96] B. Ghosh and S. Muthukrishnan. Dynamic Load Balancing by Random Matchings. *Journal of Computer and System Sciences*, 53(3):357–370, 1996.

[Gow94] C. GowriShankaran. Broadcasting on recursively decomposable Cayley graphs. *Discrete Applied Mathematics*, 53(1-3):171–182, 1994.

[GS01] G. Grimmett and D. Stirzaker. *Probability and Random Processes*. Oxford University Press, 3nd edition, 2001.

[Gur00] V. Guruswami. Rapidly Mixing Markov Chains, 2000. unpublished, available at http://www.cs.washington.edu/homes/venkat/pubs/pubs.html.

[Har62] F. Harary. The Maximum Connectivity of a Graph. *Proceedings of the National Academy of Sciences of the United States of America*, 48(7):1142–1146, 1962.

[Het00] H.W. Hethcore. Mathematics of infectious diseases. *SIAM Review 42*, pages 599–653, 2000.

[HKP+04] J. Hromkovič, R. Klasing, A. Pelc, P. Ružička, and W. Unger. *Dissemination of Information in Communication Networks*. Springer, 2004.

[HS08] M. Herlihy and N. Shavit. *The Art of Multiprocessor Programming*. Morgan Kaufmann/Elsevier, 2008. to appear.

[HT06a] M. Herlihy and S. Tirthapura. Randomized smoothing networks. *Journal of Parallel and Distributed Computing*, 66(5):626–632, 2006.

[HT06b] M. Herlihy and S. Tirthapura. Self-stabilizing smoothing and counting networks. *Distributed Computing*, 18(5):345–357, 2006.

[IZ89] R. Impagliazzo and D. Zuckerman. How to recycle random bits. In *30th Annual IEEE Symposium on Foundations of Computer Science (FOCS'89)*, pages 222–227, 1989.

[JS00] J. Jonasson and O. Schramm. On the Cover Time of Planar Graphs. *Electronic Communications in Probability*, 5:85–90, 2000.

[Kes80] H. Kesten. The critical probability of bond percolation on the square lattice equals $\frac{1}{2}$. *Comm. Math. Phys.*, 74(1):41–59, 1980.

[KKLV00] J.D. Kahn, J.H. Kim, L. Lovász, and V.H. Vu. The cover time, the blanket time and the Matthews bound. In *41st Annual IEEE Symposium on Foundations of Computer Science (FOCS'00)*, pages 467–475, 2000.

[Klu94] M. Klugerman. *Small-Depth Counting Networks and Related Topics.* PhD thesis, Department of Mathematics, Massachusetts Institute of Technology, September 1994.

[KM27] W.O. Kermack and A.G. McKendrick. Contributions to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A*, pages 700–721, 1927.

[KM96] S. Kapidakis and M. Mavronicolas. Distributed, low contention task allocation. In *Proceedings of the 8th IEEE Symposium on Parallel and Distributed Processing*, pages 358–365, 1996.

[Knu98] D.E. Knuth. *Sorting and Searching.* Addison-Wesley, 2nd edition, 1998.

[KP92] M. Klugerman and C.G. Plaxton. Small-Depth Counting Networks. In *24th Annual ACM Symposium on Theory of Computing (STOC'92)*, pages 417–428, 1992.

[KPP07] E. Kranakis, M. Paquette, and A. Pelc. Communication in Networks with Random Dependent Faults. In *32th International Symposium on Mathematical Foundations of Computer Science (MFCS'07)*, pages 418–429, 2007.

[KS86] C.P. Kruskal and M. Snir. A unified theory of interconnection network structure. *Theoretical Computer Science*, 48(3):75–94, 1986.

[KSSV00] R. Karp, C. Schindelhauer, S. Shenker, and B. Vöcking. Randomized Rumor Spreading. In *41st Annual IEEE Symposium on Foundations of Computer Science (FOCS'00)*, pages 565–574, 2000.

[Lei92] F.T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes.* Morgan Kaufmann Publishers, 1992.

[Lov93] L. Lovász. Random walks on graphs: A survey. *Combinatorics, Paul Erdös is Eighty*, 2:1–46, 1993.

[LPP99] R. Lyons, R. Pemantle, and Y. Peres. Resistance Bounds for First-Passage-Percolation and Maximum Flow. *Journal of Combinatorial Theory (Series A)*, 86(1):158–168, 1999.

[LPS88] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

[LPW06] D.A. Levin, Y. Peres, and E.L. Wilmer. *Markov Chains and Mixing Times*. (draft available, http://www.oberlin.edu/markov/), 2006.

[Mar73] G.A. Margulis. Explicit constructions of expanders. *Problemy Peredaci Informačii*, 9(4):71–80, 1973.

[Mar82] G.A. Margulis. Graphs without short cycles. *Combinatorica*, 2:71–78, 1982.

[McD98] C. McDiarmid. Concentration. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, pages 195–243. Springer-Verlag, 1998.

[MMS08] H. Meyerhenke, B. Monien, and T. Sauerwald. A New Diffusion-based Multilevel Algorithm for Computing Graph Partitions of Very High Quality. In *22nd IEEE International Parallel and Distributed Processing Symposium (IPDPS'08)*, 2008. to appear.

[MMT99] M. Mavronicolas, M. Merritt, and G. Taubenfeld. Sequentially Consistent versus Linearizable Counting Networks. In *18th Annual ACM-SIGOPT Principles of Distributed Computing (PODC'99)*, pages 133–142, 1999.

[MR95] R. Motwani and P. Raghavan. *Randomized Algorithms*. Cambridge University Press, 7th edition, 1995.

[MS06] H. Meyerhenke and T. Sauerwald. Analyzing Disturbed Diffusion on Networks. In *17th International Symposium on Algorithms and Computation (ISAAC'06)*, pages 429–438, 2006.

[MS08] M. Mavronicolas and T. Sauerwald. The Impact of Randomization in Smoothing Networks. In *27th Annual ACM-SIGOPT Principles of Distributed Computing (PODC'08)*, 2008. to appear.

[MU05] M. Mitzenmacher and E. Upfal. *Probability and Computing*. Cambridge University Press, 2005.

[New02] M.E.J. Newman. The spread of epidemic disease on networks. *Physical Review E 66, 016128*, 2002.

[Nie92] N. Niederreiter. *Random number generation and quasi-Monte Carlo methods*, volume 63 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. SIAM, Philadelphia, PA, 1992.

[NW59] C.St.J.A. Nash-Williams. Random walk and electric currents in networks. *Proceedings of the Cambridge Philosophical Society*, 55(1):181–194, 1959.

[PDDK96] V.B. Priezzhev, D. Dhar, A. Dhar, and S. Krishnamurthy. Eulerian walkers as a model of self-organized criticality. *Phys. Rev. Lett.*, 77:5079–5082, 1996.

[Pit87] B. Pittel. On spreading rumor. *SIAM Journal on Applied Mathematics*, 47(1):213–223, 1987.

[PR04] I. Pak and R. Radoičić. Hamiltonian Paths in Cayley Graphs, 2004. preprint available at http://www-math.mit.edu/∼pak/research.html.

[Rab07] Y. Rabinovich. Personal Communication, 2007.

[Rei05] O. Reingold. Undirected *st*-Connectivity in Log-Space. In *37th Annual ACM Symposium on Theory of Computing (STOC'05)*, pages 376–385, 2005.

[Ros03] S.M. Ross. *Introduction to Probability Models*. Academic Press, 2003.

[RSW98] Y. Rabani, A. Sinclair, and R. Wanka. Local Divergence of Markov Chains and the Analysis of Iterative Load Balancing Schemes. In *39th Annual IEEE Symposium on Foundations of Computer Science (FOCS'98)*, pages 694–705, 1998.

[RT87] P. Raghavan and C.D. Thompson. Randomized rounding: a technique for provably good algorithms and algorithmic proofs. *Combinatorica*, 7:365–374, 1987.

[Sau05] T. Sauerwald. Randomisiertes Broadcasting auf großen Netzwerken mit guten Topologieeigenschaften. Diplomarbeit, Universität Paderborn, 2005.

[Sau07] T. Sauerwald. On Mixing and Edge Expansion Properties in Randomized Broadcasting. In *18th International Symposium on Algorithms and Computation (ISAAC'07)*, pages 196–207, 2007.

[Sch00] C. Scheideler. Probabilistic methods for coordination problems. Habilitation thesis at the University of Paderborn, 2000.

[Sin92] A. Sinclair. Improved Bounds for Mixing Rates of Markov Chains and Multicommodity Flow. *Combinatorics, Probability & Computing*, 1:351–370, 1992.

[Sin93] A. Sinclair. *Algorithms for Random Generation and Counting*. Birkhäuser, 1993.

[SS08] T. Sauerwald and D. Sudholt. Self-stablizing Cuts in Synchronous Networks. In *15th International Colloquium on Structural Information and Communication Complexity (SIROCCO'08)*, 2008. to appear.

[SWC96] J. Sheu, C. Wu, and T. Chen. An Optimal Broadcasting Algorithm without Message Redundancy in Star Graphs. *IEEE Transactions on Parallel and Distributed Systems*, 6(6):653–658, 1996.

[Tit00] P. Tittmann. *Einführung in die Kombinatorik*. Spektrum Akademischer Verlag, 2000.

[Wil04] D. Wilson. Mixing times of lozenge tiling and card shuffling markov chains. *The Annals of Applied Probability*, 14:274–325, 2004.

[WZ96] P. Winkler and D. Zuckerman. Multiple Cover Time. *Random Structures and Algorithms*, 9(4):403–411, 1996.

[XL97] C. Xu and F.C.M. Lau. *Load Balancing in Parallel Computers*. Kluwer Academic Publications, 1997.

[Zuc92] D. Zuckerman. A Technique for Lower Bounding the Cover Time. *SIAM Journal on Discrete Mathematics*, 5(1):81–87, 1992.