The Dispersion Time of Random Walks on Finite Graphs[*]

Nicolás Rivera[1], Alexandre Stauffer[2], Thomas Sauerwald[1], and John Sylvester[1]

[1]University of Cambridge, Cambridge, United Kingdom, `firstname.lastname@cl.cam.ac.uk`
[2]University of Bath, Bath, United Kingdom, `a.stauffer@bath.ac.uk`

Abstract

We study two random processes on an n-vertex graph inspired by the internal diffusion limited aggregation (IDLA) model. In both processes n particles start from an arbitrary but fixed origin. Each particle performs a simple random walk until first encountering an unoccupied vertex, and at which point the vertex becomes occupied and the random walk terminates. In one of the processes, called *Sequential-IDLA*, only one particle moves until settling and only then does the next particle start whereas in the second process, called *Parallel-IDLA*, all unsettled particles move simultaneously. Our main goal is to analyze the so-called dispersion time of these processes, which is the maximum number of steps performed by any of the n particles.

In order to compare the two processes, we develop a coupling which shows the dispersion time of the Parallel-IDLA stochastically dominates that of the Sequential-IDLA; however, the total number of steps performed by all particles has the same distribution in both processes. This coupling also gives us that dispersion time of Parallel-IDLA is bounded in expectation by dispersion time of the Sequential-IDLA up to a multiplicative $\log n$ factor. Moreover, we derive asymptotic upper and lower bound on the dispersion time for several graph classes, such as cliques, cycles, binary trees, d-dimensional grids, hypercubes and expanders. Most of our bounds are tight up to a multiplicative constant.

Keywords: Random Walks, Internal Diffusion Limited Aggregation, Dispersion.

1 Introduction

The internal diffusion limited aggregation (IDLA) model, first introduced independently by Diaconis & Fulton [22] and Meakin & Deutch [42], is a protocol for recursively building a randomly growing subset (aggregate) of vertices of a graph. Initially, the aggregate consists of only one vertex, denoted as the *origin*, and we let a particle be settled at that vertex. Then, at each step, we start a new particle from the origin and let it perform a random walk until it visits a vertex not contained in the aggregate. At this point, we say that the new particle settles at that vertex, and the vertex is added to the aggregate. We then add a new particle at the origin, and iterate this procedure over and over again.

IDLA was introduced on the infinite lattice $\mathbb{Z}^d$. Here we consider a finite connected n-vertex graph G. Note that after n particles have settled, the aggregate occupies the whole of G. During this time, each particle performed some number of random walk steps before it settles. Clearly, this number depends on the geometry of the aggregate when the particle started moving. We define

---

the *dispersion time* as the largest number of random walk steps performed by any one of the n particles before reaching an unoccupied vertex.

We refer to the above protocol as *Sequential-IDLA*, in allusion to the fact that a particle cannot begin to move until the one before it settles. However, alternative scheduling protocols could be defined, in the sense that we could choose to add and move a new particle from the origin before the previous one has settled. In this way, there could be several unsettled particles moving at the same time, but they must abide by the rule that whenever an unoccupied vertex is visited, one particle must settle there. We call any process of this sort a *dispersion process*. We are interested in understanding the affect of different scheduling protocols on the dispersion time. In particular consider the following protocol: start all n particles from the origin at time 0 (thus one of them will instantaneously settle at the origin). Then, all particles perform one random walk step simultaneously; if one or more particles jump to an unoccupied vertex, then one such particle settles there. Iterate this procedure until all particles have settled. We called this protocol the *Parallel-IDLA*.

Both dispersion processes can be regarded as a set of simple local protocols for resource allocation. Specifically, the sequential dispersion process is quite similar to a local-search based reallocation scheme from [13], where a job continues to reallocate itself to a neighbour with less load until it has found a local minimum. Furthermore, the parallel dispersion process is related to the "QoS Load Balancing" model [1], a particular instance of selfish load balancing (see also [9, 10] for similar protocols). In the QoS model, tasks perform random walks in parallel and terminate only if they have found a resource on which the estimated processing time is acceptable according to some agent-specific threshold. Our dispersion processes can be also viewed as a spatial coordination game, where the goal is to achieve a state in which players are all making *distinct* choices. As mentioned in [4], such games serve as a model for the dynamics in location games or habitat selection of species.

Recall that the dispersion time is the maximum number of steps taken by any of the n particles in either IDLA process. For the complete graph K_n the Sequential-IDLA process has essentially the same dynamics as the famous coupon collector process and the dispersion time corresponds to the longest wait between collecting successive (new) coupons. Thus the discrepancy between the Sequential and Parallel dispersion times for K_n measures the effect of parallelising the coupon collector process on the longest time between coupons. This motivates the study of dispersion time on different networks which we can view as a generalization of the coupon collector process. In the general setting we address the question: what is the cost of parallelising the IDLA process? Addressing this question requires us to determine or at least estimate the Parallel *and* Sequential dispersion times.

The total time taken by all walks, as opposed to the longest walk, is also natural to study for these models. Returning briefly to the complete graph we see that the sum of the walk lengths in the Sequential-IDLA corresponds to the time to collect all coupons - this is what is typically studied for the coupon collector. Our couplings show that for any fixed graph the sum of all walk lengths, later denoted by W, is the same for Parallel and Sequential IDLA. From one perspective this motivates the study of W for general graphs, this is work in progress by the authors. However, in this paper we are more interested in the discrepancies between the Sequential and Parallel processes, some of which are captured by the dispersion time.

Since in IDLA particles perform random walks, both dispersion processes can be regarded as a protocol for exploring and covering an unknown network. However, as opposed to previously studied models of covering a graph with multiple random walks [3, 8, 20], the length of the particles' trajectories may vary wildly in the dispersion process. This introduces strong correlations between different particles, a challenge which is not present in the cover time of multiple random walks.

1.1 Our Contributions

Let $\tau_{seq}^v(G)$ and $\tau_{par}^v(G)$ to denote the dispersion time of Sequential-IDLA and Parallel-IDLA on G with origin v, respectively. The key question is how are $\tau_{seq}^v(G)$ and $\tau_{par}^v(G)$ related and is there an ordering between them. We answer this question by developing a coupling, based on "cutting & pasting" particle trajectories, which we use to show the following result below.

Theorem 1 (see Theorems 4.1 and 4.2). *For any connected n-vertex graph G and $v \in V(G)$,*

$$\tau_{seq}^v(G) \preceq \tau_{par}^v(G).$$

Further,

$$\mathbf{E}\left[\tau_{par}^v(G)\right] = \mathcal{O}\!\left(\mathbf{E}\left[\tau_{seq}^v(G)\right] \cdot \log n\right).$$

If instead we count the total number of jumps performed by all particles, then this quantity has the same distribution in both processes. Our work leaves whether $\mathbf{E}\left[\tau_{par}^v(G)\right] = \mathcal{O}\!\left(\mathbf{E}\left[\tau_{seq}^v(G)\right]\right)$ as an open question. Note however, that Theorem 5.1 demonstrates that already for the clique, the Parallel-IDLA is about 30 percent slower than the Sequential-IDLA. Thus, we cannot have equality between the two processes, even though the path is an example where both processes have the same dispersion time up to lower order terms, see Theorem 5.4.

In Section 4 we introduce the continuous-time Uniform-IDLA (CTU-IDLA), a variant of the Parallel-IDLA where each particle moves at times given by its own exponential rate 1 clock until it settles. Denote its dispersion time by $\tau_{c-unif}^v(G)$ and let $\tau_{c-seq}^v(G)$ be the dispersion time of the Sequential-IDLA run with continuous random walks. We also consider running the parallel and sequential processes with lazy walks and let $\tau_{L-par}^v(G), \tau_{L-seq}^v(G)$ denote their dispersion times.

Theorem 2 (see Theorems 4.3, 4.11 and 4.10). *For any connected n-vertex graph G and $v \in V(G)$,*

$$\tau_{c-unif}^v(G) = \Theta\!\left(\tau_{par}^v(G)\right), \qquad\qquad\qquad \tau_{L-par}^v(G) = \Theta\!\left(\tau_{par}^v(G)\right),$$
$$\tau_{c-seq}^v(G) = \Theta\!\left(\tau_{seq}^v(G)\right) \qquad and \qquad \tau_{L-seq}^v(G) = \Theta\!\left(\tau_{seq}^v(G)\right),$$

hold w.h.p. and in expectation.

We also consider general scheduling sequences satisfying a natural condition we call "index-repeating" which states that if the walks are not allowed to settle and the process continues forever, then no walk will ever stop moving. We can show that greatest number of steps taken by a walk in the IDLA process according to any index-repeating schedule is stochastically dominated by the same quantity in the Parallel process (Theorem 4.8). The intuition behind Parallel-IDLA being "slower" than Sequential-IDLA is that, due to competition between particles trying to settle concurrently, the lengths of particle trajectories in Parallel-IDLA vary more than in Sequential-IDLA.

Let $t_{seq}(G) = \max_{v \in V} \mathbf{E}\left[\tau_{seq}^v(G)\right]$ and $t_{par}(G) = \max_{v \in V} \mathbf{E}\left[\tau_{par}^v(G)\right]$ be the worst-case expected dispersion times over all possible origins/starting vertices in V. Let $t_{hit}(G)$ be the maximum among all vertices v, w of the expected hitting time of a random walk from v to w. We derive a basic but useful upper bound on the dispersion time in terms of the hitting time.

Theorem 3 (See Theorem 3.1, Corollary 3.2, Theorem 5.11, Proposition 5.20). *Let G be any connected graph with n vertices. Then, for any vertices $v1V$,*

$$\mathbf{Pr}\left[\tau_{par}^v(G) > 8 \cdot t_{hit}(G) \cdot \log_2 n\right] \leq \frac{1}{n^2} \qquad and \qquad t_{par}(G) = \mathcal{O}(t_{hit}(G) \cdot \log n).$$

The same results also hold for τ_{seq} and t_{seq}. These results imply the following worst-case bounds:

- *For any n-vertex graph, $t_{seq}, t_{par} = \mathcal{O}(n^3 \log n)$.*
- *For any regular n-vertex graph, $t_{seq}, t_{par} = \mathcal{O}(n^2 \log n)$.*

Moreover, the Lollipop and the cycle, respectively, are graphs matching the two bounds up to constant factors.

In view of the upper bound in Theorem 3 and based on the intuition that the last walk in the Sequential-IDLA should have a hard target to hit, one would expect that the worst-case hitting time provides at least an approximate lower bound on the dispersion time. This intuition turns out to be false in general, as evidenced by a certain class of bounded-degree trees (see Proposition 6.3) which exhibits a gap of almost $\sqrt{n}$ between the hitting and dispersion time. We obtain some lower bounds based on the maximum degree $\Delta(G)$ and the $1/4$ total variation mixing time t_{mix}.

Theorem 4 (See Theorems 3.7, 3.8 & 3.10). *Let G be a connected n-vertex graph, then $t_{seq}(G) = \Omega(|E|/\Delta)$. If $\Delta = \mathcal{O}(|E|/n)$ then $t_{seq}(G) = \Omega(t_{\mathrm{mix}})$. For any tree T, we have $t_{seq}(T) = \Omega(n)$.*

The first two bounds are tight and the third is known to be tight up to a $\log n$ factor. The upper bound in Theorem 3.1 matches Matthews bound for the cover time up to constant [37, Thm. 11.2]. While Theorem 3.1 is tight for the cycle, it turns out not to be tight for most "well-connected" graphs like expanders, high-dimensional grids and hypercubes. Thus as a general rule of thumb for well connected graphs the dispersion time is usually of order t_{hit} and poorly connected graphs it is usually of order $t_{hit} \cdot \log n$. The behaviour of this extra log factor potentially appearing in the dispersion time contrasts with that of the log factor which may appear in the cover time.

Let $t_{hit}(\pi, S)$ denote the expected hitting time of $S \subseteq V$ by a random walk from stationarity. We provide a general framework for establishing bounds better than $\mathcal{O}(t_{hit} \log n)$ by considering certain sums of hitting times of subsets of decreasing sizes.

Theorem 5 (see Theorems 3.3 and 3.6, and Corollary 3.5). *For any connected n-vertex graph G,*

$$t_{par}(G) \le 60 \cdot \sum_{j=1}^{\lceil \log_2 n \rceil} \left(t_{mix} + \max_{S \subseteq V : |S| \ge 2^{j-2}} t_{hit}(\pi, S) \right),$$

where walks in the IDLA process are lazy. Furthermore,

$$t_{seq}(G) \le 30 \cdot \max_{1 \le j \le \lceil \log_2 n \rceil} \left\{ j \cdot \left(t_{mix} + \max_{S \subseteq V : |S| \ge 2^{j-2}} t_{hit}(\pi, S) \right) \right\}.$$

Consequently for any connected n-vertex almost-regular graph,

$$t_{par}(G) = \mathcal{O}\left(\frac{n}{1 - \lambda_2} \right).$$

Neglecting constant factors, both upper bounds look comparable, however it is not difficult to verify that the upper bound on t_{seq} is at most the upper bound on t_{par}, up to constants. Conversely, the gap between the two upper bounds can be shown to be at most $\mathcal{O}(\log n)$. Note that both statements recover the basic $\mathcal{O}(t_{hit} \cdot \log n)$ upper bound, but as soon as there is a sufficient speed-up for hitting times of larger sets (and the mixing time is not too large), these bounds may give a bound of $\mathcal{O}(t_{hit})$. We will see that this is indeed the case for several fundamental classes of graphs in Section 5, where we apply the previous bounds, and in particular Theorem 5.

In Section 5 we calculate the dispersion times in several fundamental networks. Table 1 summarises our results and shows that we can determine the expected dispersion times up to multiplicative constant factors in all graphs apart from the 2-dimensional grid, where there is a discrepancy

4

Graph family name	Cover time t_{cov}	Hitting time t_{hit}	Mixing time t_{mix}	Dispersion time	
				t_{seq}	t_{par}
path	n^2	n^2	$\mathcal{O}(n^2)$	$\kappa_p \cdot n^2 \log n$	
cycle	$n^2/2$	$n^2/2$	$\mathcal{O}(n^2)$	$\Theta(n^2 \log n)$	
2-dimensional grid	$\Theta(n \log^2 n)$	$\Theta(n \log n)$	$\Theta(n)$	$\Omega(n \log n)$	$\mathcal{O}(n \log(n)^2)$
d-dimensional grid, $d > 2$	$\Theta(n \log n)$	$\Theta(n)$	$\Theta(n^{2/d})$	$\Theta(n)$	
hypercube	$\Theta(n \log n)$	$\Theta(n)$	$\log n \log \log n$	$\Theta(n)$	
binary tree	$\Theta(n \log n)$	$\Theta(n \log n)$	n	$\Theta(n \log(n)^2)$	
complete graph	$\Theta(n \log n)$	$\Theta(n)$	1	$\kappa_{cc} \cdot n$	$(\pi^2/6) \cdot n$
expanders	$\Theta(n \log n)$	$\Theta(n)$	$\mathcal{O}(\log n)$	$\Theta(n)$	

Table 1: The last two columns summarize our results, the first three columns are for comparison. The constant κ_{cc} above has an explicit formula given by Lemma 5.2 and it evaluates to roughly 1.255, to be contrasted with $\pi^2/6 \approx 1.644$. The constant κ_p is a non-explicit, though specified in Section 5. Simulations run by Nikolaus Howe (student) suggest $\kappa_p \approx 0.6\ldots$.

of order $\log n$ between the lower and upper bounds. This remains an interesting open problem which seems to require very detailed knowledge of the shape of the aggregate on a *finite* box/tori. As discussed in Section 1.3 below, this is a non-trivial problem even in the *infinite* 2d-grid. For many other graphs we obtain the correct upper bound from one of our results in Section 3 and then have to find a matching lower bound by hand. One particularly involved case not captured by our general results is the binary tree, where a tailored analysis reveals a (relatively large) dispersion time of $\Theta(n \log^2 n) = \Theta(t_{hit} \cdot \log n)$, see Theorem 5.14.

1.2 Techniques Used

The first tool we invent to analyse these processes is the Cut & Paste bijection which maps between the histories of IDLA processes. The bijection allows us to couple the dispersion times of the Parallel-IDLA to those of the Sequential-IDLA and other variants such as Uniform-IDLA (where at each step a random unsettled particle moves), as well as IDLA processes with lazy or continuous-time walks. Bounding dispersion times via these other variants is useful for avoiding issues such as periodicity or simultaneous arrivals at unoccupied vertices. At a base level the stochastic domination of τ_{seq}^v by τ_{par}^v means we can sandwich both quantities with a bound on τ_{par}^v from above and on τ_{seq}^v from below. Another useful way describe dispersion time is in terms of hitting times of sets by multiple random walks. In particular we present two different upper bounds on τ_{par}^v and τ_{seq}^v in terms of hitting times of sets. We also prove a lower bound on τ_{seq}^v by the mixing time, this comes from the relationship between the mixing time and the hitting time of large sets.

Although the Sequential and Parallel IDLA processes are closely related, the different sources of dependence arising from the contrasting scheduling protocols provide several challenges. In the Sequential-IDLA interaction between the walkers comes via the configuration of vertices settled by the previous walks. This can make proving a tight lower bound on τ_{seq}^v tricky and often some knowledge of the geometry of the aggregate after a certain time is helpful. What is needed are results reminiscent of the "shape theorems" discussed in Section 1.3 below. This requirement for detailed knowledge of the aggregate appears to be crucial in achieving a tight lower bound on τ_{seq}^v for the binary tree and 2-dimensional grid. In comparison with the Sequential-IDLA interactions are less passive in the Parallel-IDLA as particles jostle to be the first to settle a vertex. This interaction can increase the length of the longest walk as is witnessed by the Cut & Paste bijection.

1.3 Related Work

As pointed out by Diaconis & Fulton [22], there are several mathematical reasons for studying IDLA, including using it to take a product of sets - a special case of the "smash product". The limit shape of the aggregate on $\mathbb{Z}^d$ was first studied by Lawler, Bramson and Griffeath [35] who showed that, after adding n particles and properly rescaling the aggregate by $n^{1/d}$, in the limit as $n \to \infty$ this converges to an Euclidean ball. There has been a series of improvements to this "shape theorem" of [35], by bounding the rate of convergence to the euclidean ball. The first refinement was made by Lawler [33] and the state of the art was achieved recently by two independent groups of authors [6, 5, 7, 30, 31]. Several authors have also proved shape theorems on other infinite graphs and groups including combs, d-ary trees, non-amenable groups and Bernoulli percolation on $\mathbb{Z}^d$ [29, 12, 28, 47, 24]. In all of these cases the limit shape is always a ball with respect to the underlying graph metric. Limit shapes in $\mathbb{Z}^d$ for other variants of IDLA have also been established. These variations include using non-standard random walks such as for drifted [41] and cookie walks [45] or starting the walks from different positions [23]. The time for the process started with some initial aggregate to "forget" this starting state has also been studied [39, 48].

One model where interaction between particles prevents settling at a site is a two-type particle system called "Oil and Water" where particles of opposite types displace each other [16]. There have been some papers on models related to the Parking function of a graph where cars drive randomly around a graph searching for vacant spots [21, 27]. More commonly, however, interaction is directly between particles and not with the host graph such as predator prey/coalescing models [20]. The problem of uniformly distributing n non-communicating memoryless particles across n unoccupied sites is also considered from a game theoretic perspective [4].

Other models related to IDLA include rotor-router aggregation, chip firing, Abelian sandpile models and activated random walks [11, 38, 49]. Many of these interacting particle systems satisfy a so-called "least action principle" which is key to their analysis. Such a principle roughly states that the natural behavior of the system is in a sense optimal and, if the process is perturbed, then the outcome will have a higher energy. One may try to find a least action principle for Sequential-IDLA by conjecturing that if we allow that a random walk sometimes does not settle when visiting an unoccupied vertex (thereby performing more random walk steps), then this could only delay the dispersion time. However, we show in Proposition 6.2 that this is not the case. In particular, we give a graph for which the dispersion time decreases if one allows some particles to perform more random walks steps.

To the best of our knowledge, the dispersion time and IDLA on a finite graph has not been studied before. Moore and Machta consider running IDLA walks synchronously for the purposes of simulating the limit shape [43] in parallel models of computation, however their results don't appear to overlap with ours. Simulating the process efficiently has also been studied more recently [26]. Thacker and Volkov [50] study a border DLA based growth model on finite graphs and investigate how long until the aggregate grown from a fixed origin hits a fixed boundary.

2 Preliminaries

Throughout $G = (V, E)$ will denote an undirected, unweighted, connected graph with n vertices. Let $\Delta(G)$ denote the maximum degree of G. We say that a graph G is almost-regular if the ratio between maximum degree and minimum degree is bounded from above by a constant.

To recap we let $\tau_{par}^v(G)$ denote the dispersion time of the Parallel-IDLA process on G started from v, that is the first iteration at which every vertex hosts (exactly) one particle. Similarly $\tau_{seq}^v(G)$ denotes the dispersion time of the Sequential-IDLA process on G started from v, that

is the longest time it takes a single particle to settle. Let $t_{seq}(G) = \max_{v \in V} \mathbf{E}\left[\tau_{seq}^v(G)\right]$ and $t_{par}(G) = \max_{v \in V} \mathbf{E}\left[\tau_{par}^v(G)\right]$. We shall drop the dependence on G from our notation when the graph is clear from the context.

Further, let $t_{hit}(u, v) = \mathbf{E}\left[\tau_{hit}(u, v)\right]$, where $\tau_{hit}(u, v)$ is the time for a random walk to reach v from u. Let $t_{hit}(G) := \max_{u,v \in V(G)} t_{hit}(u, v)$. For a probability distribution μ on V and a set $S \subset V$ let $t_{hit}(\mu, S)$ denote the expected time for the walk starting from μ to hit any vertex in S.

Thanks to our results relating lazy and non-lazy walks (Theorem 4.3), we can conveniently switch between the two models at the cost of a constant factor (under some mild additional conditions this factor is $2 + o(1)$), thus walks may be lazy. We use P to denote the transition matrix of the non-lazy walk (and $\widetilde{P} = (I + P)/2$ for the lazy walk). We also use $p_{u,v}^t$ to denote the probability a random walk goes from u to v in t steps (and $\widetilde{p}_{u,v}^t$ respectively for the lazy walk). We let $t_{\text{mix}} = \min_{t \geq 1}\left\{t : \max_{x \in V} \sum_{y \in V} \left|\widetilde{p}_{x,y}^t - \pi(y)\right| \leq 1/e\right\}$ denote the mixing time of G.

Some of the dispersion results in the paper hold in expectation, some hold w.h.p. (with probability $1 - o(1)$) and others hold in both senses. One does not necessarily imply the other, in particular Proposition 6.1 show there are graphs where neither dispersion time concentrates.

Road Map. The rest of this paper is organized as follows. We first present some general upper and lower bounds in Section 3 before turning to the more involved coupling proofs in Section 4. In Section 5 we apply the results from Section 3 and Section 4 to specific networks completing the results in Table 1.1, for some of these networks a more refined analysis is required. We conclude the paper in Section 7 with a summary of our results and some open problems.

3 General Bounds

3.1 Upper Bounds

The first upper bound we present holds for any graph and only requires knowledge of the maximum hitting time of a random walk between two vertices. Although this result can be also recovered from the more general Theorem 3.3, it serves as a good "warm-up".

Theorem 3.1. *Let G be any connected graph with n vertices. Then for any $v \in V$,*

$$\mathbf{Pr}\left[\tau_{par}^v(G) > 8 \cdot t_{hit}(G) \cdot \log_2 n\right] \leq \frac{1}{n^2} \qquad and \qquad t_{par}(G) = \mathcal{O}(t_{hit}(G) \cdot \log n).$$

The same results also hold for τ_{seq}^v and t_{seq}.

Proof. To begin sample n random walks of length $T = 8t_{hit}(G) \log_2 n$ starting from the origin, then w.p. at least $1 - n^{-2}$ all of these walks have covered all the vertices of the graph. To see this note the probability a single walk of length $2t_{hit}$ visits $u \in V$ is at least $1/2$ by Markov's inequality, thus by the Markov property u is visited in time T w.p. $1 - n^{-4}$. Thus, by a union bound, in time T one walk covers the graph w.p. at least $1 - n^{-3}$ and all walks cover the graph w.p. at least $1 - n^{-2}$.

Now, we run the Parallel-IDLA process by using the n sampled walks, thus each particle follows a predetermined trajectory. Since all the n walks cover the graph, it follows that all the particles have to settle by time $T = 8t_{hit}(G) \log_2 n$ with probability at least $1 - n^{-2}$. To obtain the result in expectation, divide the time in phases of $8t_{hit}(G) \log_2 n$ time-steps, then the number of phases needed to finish the process is stochastically dominated a geometric random variable of mean $1/(1-n^{-1})$ concluding that $\mathbf{E}\left[\tau_{par}^v(G)\right] = \mathcal{O}(t_{hit}(G) \log n)$. Since this holds for any $v \in V$ it follows that $t_{par} = \mathcal{O}(t_{hit}(G) \log n)$. The same results holds for τ_{seq}^v and t_{seq} due to Theorem 4.1. $\qquad\square$

This simple bound is actually tight in many cases, see Table 1. The next result is a simple consequence, yet it provides the correct asymptotic worst-case bounds for the dispersion time.

Corollary 3.2 (General quantitative bounds on graphs)**.**

- *For any n-vertex graph, $t_{seq}, t_{par} = \mathcal{O}\big(n^3 \log n\big)$.*
- *For any regular n-vertex graph, $t_{seq}, t_{par} = \mathcal{O}\big(n^2 \log n\big)$.*

Proof. This follows from Theorem 3.1 and the bounds on t_{hit} in [40, Thm. 2.1]. □

Notice these bounds exceed the corresponding upper bounds on the cover time [2, Thm. 6.12, Thm. 6.15] by a $\log n$-factor. Both bounds above are sharp up to a multiplicative constant as witnessed by the lollipop and the cycle respectively, see Proposition 5.20 and Theorem 5.11 respectively. In fact for any fixed $r < \infty$ one can construct a family of r-regular graphs for which the second bound above is tight. For example when $r = 3$ one can iteratively augment an even cycle by adding an edge between two vertices of degree two who are at distance two to obtain a 3-regular graph with the same asymptotic dispersion time as the cycle.

3.1.1 General Bounds in Terms of Hitting Times of Sets

In this section we achieve more refined bounds by considering hitting times of sets as opposed to vertices. To avoid periodicity related issues we assume the trajectory of the particles is a lazy random walk. As shown in Theorem 4.3, the parallel or sequential dispersion times with lazy walk are equivalent to their non-lazy counterparts up to constant factors, thus any results established for the dispersion time with lazy walks also apply for non-lazy walk (up to a constant factor) and vice versa. Define $\tau_{par}^v(G, k)$ to be the first time (from worst case start vertex) that less than $2^k - 1$ vertices are left to be settled in the Parallel-IDLA, and let $t_{par}^k(G) = \max_{v \in V} \mathbf{E}\big[\tau_{par}^v(G, k)\big]$ denote the worst-case expectation. Clearly $\tau_{par}^v(G, 1) = \tau_{par}^v(G)$, which is the standard parallel dispersion time.

Theorem 3.3. *Consider the Parallel-IDLA process with lazy walks. Then, for any connected n-vertex graph and any $k \geq 1$, we have*

$$t_{par}^k(G) \leq 60 \cdot \sum_{j=k}^{\lceil \log_2 n \rceil} \left(t_{mix} + \max_{S \subseteq V \,:\, |S| \geq 2^{j-2}} t_{hit}(\pi, S) \right).$$

One consequence of this theorem for $k = \log_2 n - 1$ is that within $\mathcal{O}(t_{mix})$ steps, at least $n/2$ random walks are settled (this follows since by the duality between hitting time of large sets and mixing time [44], $\max_{S \subseteq V \,:\, |S| \geq n/4} t_{hit}(\pi, S) = \mathcal{O}(t_{mix})$.

Remark 3.4. *Note that the upper bound can be estimated directly to be at most $60 \lceil \log_2 n \rceil \cdot (t_{mix} + t_{hit}) \leq 120 \lceil \log_2 n \rceil \cdot t_{hit}$, so this bound is (up to a multiplicative constant) a refinement of Theorem 3.1.*

Proof of Theorem 3.3. We divide the process into $\log_2 n$ phases which are labelled in reverse order $\lceil \log_2 n \rceil, \lceil \log_2 n \rceil - 1, \ldots, 2, 1$. Phase j starts as soon as the number of unsettled walks k satisfies $k \in [2^{j-1}, 2^j)$. It could be case that the number of unsettled walks more than halves in one step and phase j is skipped, for now assume this is not the case. Let t be the first time step at the beginning of phase j, and let $S \subseteq V$ be the set of unoccupied vertices at time t, thus $|S| = k$. Consider k random walks moving independently and having no interaction with the unsettled vertices, then

8

let τ_j be the (random) time such that no subset S' of S with size at least $k/2$ is visited by any less than $k/2$ of these walks. We now argue by contradiction that τ_j stochastically dominates the length of phase j. Suppose the number of unsettled walks is still at least $k/2$ at step $t + \tau_j$. Hence there exists still a subset S' of unoccupied vertices with size at least $k/2$ at step $t + \tau_j$. We know that at least $k/2$ of the walks would hit at least one vertex of this set S'. Thus all these walks must terminate earlier, as otherwise the vertices in S' cannot all be unoccupied at step $t + \tau_j$, however, in this case we have a contradiction to the assumption that at least $k/2$ of the walks are still unsettled.

We will now bound $\mathbf{E}[\tau_j]$ from above. Consider first a fixed random walk and a fixed set $S' \subseteq S$ of size at least $k/2$. The probability that a fixed random walk does not hit S' within $30 \cdot (t_{mix} + t_{hit}(\pi, S'))$ steps is at most $(1/2)^6$, this follows easily from the fact that after $5t_{mix}$ time, with probability at least $1 - e^{-1}$, we can couple the Markov chain with the stationary distribution (e.g. Lemma A.5. in [32]), and then, given that the coupling is successful Markov's inequality gives us that with probability at most $1/5$ we do not hit S', thus, the probability we do not hit S' in $5 \cdot (t_{mix} + t_{hit}(\pi, S'))$ steps is at most $e^{-1} + (1 - e^{-1})(1/5) < 1/2$, and thus after 6 time-intervals of length $5 \cdot (t_{mix} + t_{hit}(\pi, S'))$ the probability the walk does not hit S' is at most $(1/2)^6$.

Hence the probability that at least $k/2$ of the k walks do not hit the set S' is at most

$$\binom{k}{k/2} \cdot \left(\frac{1}{2^6}\right)^{-k/2} \leq 2^k \cdot 2^{-3k}.$$

Taking the Union bound over all possible $\binom{k}{k/2} \leq 2^k$ subsets of S which are of size at least $k/2$, it follows that the probability that there exists a subset S of the unoccupied vertices of size at least $k/2$ such that at least $k/2$ of the walks do not hit the set S is at most

$$2^k \cdot 2^k \cdot 2^{-3k} \leq 2^{-k} \leq 1/2.$$

Hence the expected time the process spends in phase j (assuming that we reach this phase and do not skip it) is at most

$$2 \cdot 30 \cdot (t_{mix} + \max_{S \subseteq V \,:\, |S| \geq 2^{j-2}} t_{hit}(\pi, S)).$$

Summing up these contributions from k to $\lceil \log_2 n \rceil$ yields the result. $\qquad \square$

Corollary 3.5. *Let G be a connected n-vertex almost regular-graph. Then, $t_{par}(G) = \mathcal{O}(n/(1 - \lambda_2))$.*

Proof. By Lemma A.2 states that $t_{hit}(v, S) = \mathcal{O}\left(\frac{n(1 + \lceil \log |S| \rceil)}{(1 - \lambda_2)|S|}\right)$ holds for any $S \subset V$, $v \in V$. We also have $t_{\mathrm{mix}} = \mathcal{O}\left(\frac{\log n}{1 - \lambda_2}\right)$ by [37, Thm. 12.3]. Plugging these estimates into Theorem 3.3 yields

$$t_{par}(G) \leq \mathcal{O}(1) \cdot \sum_{j=1}^{\lceil \log_2 n \rceil} \left(\frac{\log n}{1 - \lambda_2} + n \cdot \frac{1 + \log 2^{j-2})}{(1 - \lambda_2)2^{j-2}}\right) = \mathcal{O}\left(\frac{n}{1 - \lambda_2}\right).$$

The result follows since lazy and non-lazy dispersion times are equivalent up to a constant factor by Theorem 4.3. $\qquad \square$

Let us now turn to the sequential process, where we can derive a similar bound, which turns out to be slightly stronger.

Theorem 3.6. *For any n vertex graph G, we have*

$$t_{seq}(G) \leq 30 \cdot \max_{1 \leq j \leq \lceil \log_2 n \rceil} \left\{ j \cdot \left(t_{mix} + \max_{S \subseteq V \,:\, |S| \geq 2^{j-2}} t_{hit}(\pi, S))\right) \right\}.$$

9

Proof. Since in the sequential process only one walk moves at a time we can couple simple and lazy walks so that the dispersion time with simple walks is always less than with lazy walks. Thus we can assume the walk is lazy. Fix a time τ to be determined later. Consider the $(n-k)$-th walk in the Sequential-IDLA, when there are still k unoccupied vertices. It was argued in the proof of Theorem 3.3 that the probability the random walk does not hit a set S of size k within $5(t_{mix} + \max_{S \subseteq V: |S|=k} t_{hit}(\pi, S))$ time steps is at most $1/2$ regardless of the initial vertex v. Denote

$$q(k) = \left\lfloor \frac{\tau}{5 \cdot (t_{mix} + \max_{S \subseteq V: |S|=k} t_{hit}(\pi, S))} \right\rfloor,$$

hence the probability that the random walk does not succeed within τ steps (assuming τ is large enough) is at most $2^{-q(k)}$. Thus by the Union bound, the probability that at least one of the n walks do not succeed is at most $\sum_{k=1}^{n} 2^{-q(k)}$. By dividing the sum into $\lceil \log_2 n \rceil$ buckets of sizes (at most) $1, 2, \ldots, 2^m, \ldots, 2^{\lceil \log_2 n \rceil}$, and using monotonicity of hitting times, it follows that the above term is at most

$$\sum_{j=1}^{\lceil \log_2 n \rceil} 2^j \cdot \exp\left(-\frac{\tau \log 2}{5 \cdot (t_{mix} + \max_{S \subseteq V: |S| \geq 2^{j-2}} t_{hit}(\pi, S))} \right).$$

Next observe that we need to ensure that for every j it holds that

$$\tau \geq j \cdot 5 \left(t_{mix} + \max_{S \subseteq V: |S| \geq 2^{j-2}} t_{hit}(\pi, S) \right),$$

otherwise just a single addend above is larger than 1. However, if we just choose

$$\tau := 3 \max_{1 \leq j \leq \log_2 n} \left\{ j \cdot 5 \cdot \left(t_{mix} + \max_{S \subseteq V: |S| \geq 2^{j-2}} t_{hit}(\pi, S) \right) \right\},$$

then we see that the total sum in the Union bound expression is at most $1/2$, and we can conclude that with probability at least $1/2$ none of the k walks takes more than τ steps. Repeating the argument that the probability that one walk take more than $m\tau$ steps is at most 2^{-m} gives the result. $\square$

It can be checked that the bounds of Theorem 3.6 are (up to constant) potentially better than the bounds of Theorem 3.3 up to a $\log n$ factor.

Bounds on the expected hitting time of sets can be obtained by analyzing return probabilities, in some situations these bounds are very tight. Since those bounds are more related to Markov chains properties than the IDLA process, and in order to keep the analysis of the IDLA process as clean as possible, we do not provide those bounds here, but in Appendix A. These bounds can be applied either in Theorem 3.3 and Theorem 3.6, but also in directly for specific graph families.

3.2 Lower Bounds

Theorem 3.7. *Let G be a connected n-vertex graph with maximum degree Δ, then $t_{seq}(G) = \Omega(|E|/\Delta)$. Hence in particular, $\Omega(n)$ is a lower bound for almost-regular graphs.*

Proof. We will analyse the Sequential-IDLA process and lower bound the time it takes for the last walk to find a free site.

Recall that for any pair of vertices $u, v \in V$, $t_{com}(u, v) = t_{hit}(u, v) + t_{hit}(v, u)$ is the commute time between u and v. By [40, Cor. 2.5] there is an ordering of the n vertices so that if u precedes

v, then $t_{hit}(u, v) \leq t_{hit}(v, u)$. Let us take the vertex w as the origin of the dispersion process so that for any other vertex v, we have $t_{hit}(w, v) \geq t_{hit}(v, w)$. Hence for every vertex v,

$$t_{hit}(w, v) \geq 1/2 \cdot t_{com}(w, v).$$

Let $R(u, v)$ be the effective resistance between u and v and note that $R(w, v) \geq 1/\deg(w) + 1/\deg(v) \geq 2/\Delta$. Hence $t_{com}(w, v) = 2|E| \cdot R(w, v) = \Omega(|E|/(\Delta + 1))$ by the commute time identity [37, Prop. 10.6]. It follows that, in expectation, the last walk in the Sequential-IDLA takes $\Omega(|E|/\Delta)$ steps. □

Theorem 5.1 shows this is tight up to constant when G is the complete graph K_n. We also present a refined lower bound for trees.

Theorem 3.8. *Let T be any n-vertex tree, then $t_{seq}(T) \geq 2n - 3$.*

Proof. If an IDLA process started from any vertex of T the last vertex settled by must be a leaf. Call the last vertex v which is connected to T by one edge $\{u, v\}$. Thus the expected time taken by the last walk to settle is at least the expected time $t_{hit}(u, v)$ to cross the edge $\{u, v\}$. The Essential edge Lemma [2, Lem. 5.1] states that $H(u, v) = 2|A(u, v)| - 1$ where $A(u, v)$ is the component of T containing u after the removal of $\{u, v\}$. Since $|A(u, v)| \geq n - 1$, the proof is complete. □

Let S_n be the n-vertex star and notice that $t_{seq}(S_n) = 2t_{seq}(K_n) \approx 2.6n$ by Theorem 5.1. This shows Theorem 3.8 is tight up to a small multiplicative constant.

Remark 3.9. *It would be natural to hope the lower bound $t_{seq} = \Omega(t_{hit})$ should hold since one would expect the vertices with largest hitting times to be explored later by the sequential process and thus contribute to the dispersion time. Proposition refutes this by exhibiting a graph where t_{seq} is a $\mathrm{poly}(n)$-factor smaller than t_{hit}.*

For a graph G let Φ be the conductance of G and let λ_2 and t_{mix} be the second largest eigenvalue and mixing time associated with the lazy random walk on G respectively. The following lower bound is tight up to a $\log n$ factor as witnessed by the cycle, Theorem 5.11.

Proposition 3.10. *Let G be a graph satisfying $\Delta = \mathcal{O}(|E|/n)$. Then there exits a $v \in V$ such that $\Omega(n)$ walks in the Sequential-IDLA process from v talk time $\Omega(t_{\mathrm{mix}})$ to settle w.h.p., consequently*

$$t_{seq}(G) = \Omega(t_{\mathrm{mix}}) = \Omega\left(\frac{1}{1 - \lambda_2}\right) = \Omega\left(\frac{1}{\Phi}\right).$$

Proof. By the characterization of mixing times by hitting times of large sets [44], for all reversible lazy random walks

$$t_{\mathrm{mix}} \leq c \max_{u, A : \pi(A) > 1/3} t_{hit}(u, A), \tag{1}$$

where $c < \infty$ is a universal constant, which can be assumed to be greater than 1. Let u and A be a vertex and a set that together maximize the above expectation $t_{hit}(u, A)$. Let $r := \Delta \cdot n/|E| < \infty$ and observe that $|A|\Delta > 2|E|/3$ and thus $|A| > n/(3r)$. Consider now a simple random walk of length $\ell := t_{\mathrm{mix}}/(120 \cdot r \cdot c)$. For every vertex $v \in V$, let p_v be the probability that a random walk starting from v hits the set A within ℓ steps. Note that there must be at least one vertex $w \in V$ such that $p_w < 1/(12 \cdot r)$ since otherwise the expected time to hit A is less than $t_{\mathrm{mix}}/(c \cdot 10)$ for all vertices v, contradicting (1).

Let X be the number of walks from w that take time less than ℓ to hit A (if there were not allowed to settle before this). As these walks are independent it follows that X is distributed as

Bin(n, p_w), thus $\mathbf{E}[X] < n/(12r)$ and $\mathbf{Pr}[X \geq n/(6r)] \leq e^{-\Omega(np_w)}$ by the Chernoff bound. Thus w.h.p. at least $(1 - 1/(6r))n$ of the n walks will take time at least ℓ to reach A, thus at least $n/(6r)$ walks which settling in A take time at least ℓ. Hence

$$t_{seq} = \Omega(t_{\mathrm{mix}}),$$

proving the result. Then using the fact that $t_{mix} \geq (1/(1 - \lambda_2) - 1)\log(1/e)$ [37, Thm. 12.4] and then the fact that we need at least one step to mix gives $t_{mix} = \Omega(\frac{1}{1-\lambda_2})$. Cheeger's inequality [37, Thm. 13.14], which states $\frac{1}{1-\lambda_2} = \Omega(\frac{1}{\Phi})$, completes the proof. $\qquad\square$

The following bound will be of use in Section 4 as although rather weak it holds w.h.p. for any start vertex. The proof appears in a different context [25] but we reproduce it here for completeness.

Lemma 3.11. *For any connected n-vertex graph and $v \in V(G)$, if n is large enough, it holds that $\tau_{seq}^v > \frac{1}{14}\log n$ with probability at least $1 - e^{-n^{1/2}}$.*

Proof. Consider the following process: run n independent random walks on G starting from v, and we stop them at time $L = c\log n$, with $c = 1/14$. Denote by $\mathcal{C}$ the set of vertices that are hit by at least one of those random walks. A simple coupling argument shows that $\mathbf{Pr}[\mathcal{C} \neq V] \leq \mathbf{Pr}[\tau_{seq}^v > L]$, and thus we will prove that $\mathbf{Pr}[\mathcal{C} \neq V] \geq 1 - e^{-n^{1/2}}$.

For any of the n walks, denote by $\mathcal{C}_i$ the set of vertices covered by the i-th walker in the first L steps. Hence $\mathcal{C} = \cup_{i=1}^n \mathcal{C}_i$. Denote $U = \{u \in V : \mathbf{Pr}[u \in \mathcal{C}_i] \geq \frac{3}{2}\frac{L}{n}\}$, and note U is independent of i. Hence for any i,

$$L \geq \mathbf{E}[|\mathcal{C}_i|] = \sum_{u \in V}\mathbf{Pr}[u \in \mathcal{C}_i] \geq \sum_{u \in U}\frac{3}{2}\frac{L}{n} \geq |U|\frac{3}{2}\frac{L}{n},$$

therefore $|U| \leq \frac{2}{3}n$. Denote by $\mathcal{D} = V \setminus \mathcal{C}$ the set of uncovered vertices, then

$$\mathbf{E}[|\mathcal{D}|] \geq \sum_{u \in V\setminus U}\mathbf{Pr}[u \in \mathcal{D}] \geq \sum_{u \in V\setminus U}(1 - \mathbf{Pr}[u \in C_1])^n \geq \frac{3}{n}\left(1 - \frac{3L}{2n}\right)^n \geq \frac{n}{3}\cdot e^{-3L/(2-3L/n)},$$

where in the last step we use the bound $e^{-x/(1-x)} \leq 1 - x$ for $|x| < 1$. Hence, we deduce that $\mathbf{E}[|\mathcal{D}|] \geq \frac{n}{3}n^{-\frac{3c}{2}(1+o(1))} \geq \frac{1}{3}n^{1-3c}$.

Finally, we will prove that with probability at least $1 - e^{-n^{1/2}}$ it holds that $|\mathcal{D}|$ is greater than $\frac{1}{2}\mathbf{E}[|\mathcal{D}|]$, and then $|\mathcal{D}| > 0$. Note that $|\mathcal{D}|$ depends on the trajectory of the n random walks and changing one of them changes $|\mathcal{D}|$ in at most $L + 1$ values. Therefore by the method of bounded differences we have

$$\mathbf{Pr}\left[|\mathcal{D}| - \mathbf{E}[|\mathcal{D}|] < -\frac{1}{2}\mathbf{E}[|\mathcal{D}|]\right] \leq \exp\left(-\frac{\mathbf{E}[|\mathcal{D}|]^2}{2n(L+1)^2}\right) \leq \exp\left(-\frac{(\frac{1}{3}n^{1-3c})^2}{2n(L+1)^2}\right),$$

recalling $L = c\log n$ gives that $\mathbf{Pr}[|\mathcal{D}| - \mathbf{E}[|\mathcal{D}|] < -\frac{1}{2}\mathbf{E}[|\mathcal{D}|]]$ is at most

$$\exp\left(-\frac{n^{2-6c}}{18n(c\log n)^2(1+o(1))}\right) \leq \exp\left(-\frac{n^{1-13c/2}}{20c^2}\right) \leq \exp\left(-n^{1-7c}\right) = e^{-n^{1/2}},$$

since $c = 1/14$. The result follows as $\mathbf{E}[|\mathcal{D}|] \geq \frac{1}{3}n^{1-3/14} > 2$. $\qquad\square$

4 Coupling and Stochastic Domination

In this section we shall prove the following stochastic domination using a coupling.

Theorem 4.1. *Let G be a finite graph and $v \in V(G)$. Then*

$$\tau_{seq}^v(G) \preceq \tau_{par}^v(G).$$

An immediate corollary of this is the relation $\mathbf{E}\left[\tau_{seq}^v(G)\right] \leq \mathbf{E}\left[\tau_{par}^v(G)\right]$, we also prove the reverse inequality up to $\log n$ factors.

Theorem 4.2. *Let G be a finite graph and $v \in V(G)$. Then*

$$\mathbf{E}\left[\tau_{par}^v(G)\right] = \mathcal{O}\big(\mathbf{E}\left[\tau_{seq}^v(G)\right] \cdot \log n\big).$$

We define the lazy Sequential/Parallel-IDLA to be the Sequential/Parallel-IDLA with the particles moving according to a lazy (instead of simple) random walk. Let $\tau_{L-seq}^v(G)$ be the dispersion time of the lazy Sequential-IDLA on G starting from v, and $\tau_{L-par}^v(G)$ be the analogous quantity for the lazy Parallel-IDLA. The relation between the lazy and standard IDLA dispersion times is given in the following theorem.

Theorem 4.3. *Let G and $v \in V(G)$. Then the following holds w.h.p. and in expectation*

$$\tau_{L-seq}^v(G) = \Theta\big(\tau_{seq}^v(G)\big) \qquad and \qquad \tau_{L-par}^v(G) = \Theta\big(\tau_{par}^v(G)\big).$$

Additionally, if there exits some $\ell = \omega(\log n)$ such that $\mathbf{Pr}\left[\tau_{par}^v(G) \leq \ell\right] \leq 1/\ell$ then

$$\tau_{L-seq}^v(G) = (2 + o(1)) \cdot \tau_{seq}^v(G) \qquad and \qquad \tau_{L-par}^v(G) = (2 + o(1)) \cdot \tau_{par}^v(G),$$

hold w.h.p. and in expectation.

The proofs of the above theorems are based on a coupling between the Sequential and Parallel-IDLA processes. To construct this coupling we consider a (Parallel or Sequential) IDLA process on G as an irregular 2-dimensional array L where each element $L(i,j) \in V$. This array L has n rows representing the n particles. Column t represents time t, and thus $L(i,t)$ represents the vertex visited by walk i at time t. We let ρ_i denote the length of walk i, hence the index of each row i goes from 0 to ρ_i. We denote by $\mathcal{I}_L$ the set of all indices (i,t) of the array L.

Given $(i,s),(j,t) \in \mathcal{I}_L$, we say that (i,s) is smaller than (j,t) in sequential order, written $(i,s) <_S (j,t)$ if either $(i < j)$ or $(i = j, s < t)$. Thus in sequential order, the block L is read as

$$L(1,0), L(1,1), \dots L(1,\rho_1), L(2,0), \dots L(2,\rho_2), \dots, L(n,0), \dots, L(n,\rho_n).$$

Likewise we say that (i,s) is smaller than (j,t) in parallel order, denoted by $(i,s) <_P (j,t)$ if either $(s < t)$ or $(s = t, i < j)$. So, in parallel order, the block L is read as

$$L(1,0), L(2,0), \dots, L(n,0), L(1,1), L(2,1), \dots, L(n,1), \dots, L(1,r), L(2,r), \dots, L(n,r), \dots$$

where if $r > \rho_i$ then $L(i,r)$ is empty so it is skipped.

Note that if L is a block representing a parallel or Sequential-IDLA the following property holds

$$L(i,\rho_i) \neq L(j,\rho_j) \text{ for each pair } i \neq j. \tag{2}$$

If L satisfies (2) then $\{L(i,\rho_i) : i \in [n]\} = V$ and the final element of each row is unique.

A block L satisfying (2) represents a Sequential-IDLA process if and only if each row i represents a path in G from vertex $L(i,0) = v$ to $L(i,\rho_i)$ and for all $(i,t) \in \mathcal{I}_L$

$$(i,t) \text{ is the first occurrence of vertex } L(i,t) \text{ in } L \text{ w.r.t. } <_S \text{ iff } t = \rho_i. \tag{3}$$

This says that when L is read in sequential-order the first time a new vertex is read it ends the current row. Similarly a block L satisfying (2) is a realization of a Parallel-IDLA process if and only if each row i represents a path in G from vertex $L(i,0) = v$ to $L(i,\rho_i)$ and and for all $(i,t) \in \mathcal{I}_L$

$$(i,t) \text{ is the first occurrence of vertex } L(i,t) \text{ in } L \text{ w.r.t. } <_P \text{ iff } t = \rho_i. \tag{4}$$

For a 2-dim array L we denote its total length (the work done) by $W(L)$, this is the total number of moves recorded by L and thus $W(L) := \rho_1 + \cdots + \rho_n$. Let Seq_v^m, or Par_v^m, denote the set of all sequential, respectively parallel, blocks representing realizations of IDLA starting from v and total length m, i.e. $W(L) = m$.

To build the coupling between Sequential and Parallel-IDLA, we are going to use a series of "Cut & Paste" transformations. Consider $(i,t) \in \mathcal{I}_L$, then define $\mathbf{CP}_{(i,t)}(L)$ as the block constructed by taking L and cutting the cells $(i, t+1), \ldots, (i, \rho_i)$ and pasting it after the unique (k, ρ_k) with $L(i,t) = L(k, \rho_k)$.

Example: Represented below are L, a block on $V = \{1,2,3,4\}$, and $\mathbf{CP}_{(4,1)}(L)$ which is the result of applying the cut & paste $\mathbf{CP}_{(4,1)}$ to L.

While $\mathbf{CP}_{(1,0)}(L) = \mathbf{CP}_{(2,1)}(L) = \mathbf{CP}_{(3,3)}(L) = \mathbf{CP}_{(4,5)} = L$. Note that if L satisfies property (2), then $L' = \mathbf{CP}_{(i,t)}(L)$ also satisfies (2). Property (2) is an important invariant for our algorithms.

4.1 Algorithms

We propose two algorithms **StP** and **PtS**, formally specified by Algorithms 1 and 2 below. The algorithm **StP** transforms a sequential process into a parallel and **PtS** transforms a parallel process into a sequential. The key component of both algorithms is the "cut & paste" operation **CP**.

Both algorithms work as follows: a pointer moves through the input array L in a fixed order and when the pointer sees a vertex label for the first time this label is added to the set $\mathcal{S}$ of seen vertices and a cut & paste transform **CP** is applied to L at this position before the pointer continues. The difference is that in **StP** the pointer explores columns then rows (i.e. in parallel order $<_P$), whereas **PtS** reads rows then columns (i.e. in sequential order $<_S$).

Broadly speaking the algorithms try to read the input array as if it was of the type specified by the output and if the input fails to have this form then it will edit it using the cut & paste transform until it has the correct form.

Result: transforms a sequential array L into a parallel array	**Result:** transforms a parallel array L into a sequential array

$\mathcal{S} \leftarrow \emptyset$;
 $t \leftarrow 0$;
 while $|\mathcal{S}| < n$ **do**
     for $i = 1, \ldots, n$ **do**
         if $(i,t) \in \mathcal{I}_L$ *and* $L(i,t) \notin \mathcal{S}$ **then**
             $\mathcal{S} \leftarrow \mathcal{S} \cup \{L(i,t)\}$;
             $L \leftarrow \mathbf{CP}_{(i,t)}(L)$;
         end
     end
     $t \leftarrow t + 1$;
 end
 return L;
	$\mathcal{S} \leftarrow \emptyset$;
	 1 **for** $i = 1, \ldots, n$ **do**
	     $t \leftarrow 0$;
	 2 **while** $(i,t) \in \mathcal{I}_L$ **do**
	 3 **if** $L(i,t) \notin \mathcal{S}$ **then**
	 4 $\mathcal{S} \leftarrow \mathcal{S} \cup \{L(i,t)\}$;
	 5 $L \leftarrow \mathbf{CP}_{(i,t)}(L)$;
	         exit(while)
	     end
	     $t \leftarrow t + 1$;
	   end
	 end
	 return L;

Algorithm 1: Sequential to Parallel (**StP**)	**Algorithm 2:** Parallel to Sequential (**PtS**)

The set $\mathcal{S} = \mathcal{S}(L, k)$ stores the different values of $L(i,j)$ observed after k iterations of the innermost loop. The algorithms terminate once they have scanned the whole array, this is the first time when $|\mathcal{S}| = n$. Sometimes they may apply $\mathbf{CP}_{(i,j)}$ with $j = \rho_i$, this leaves L unchanged.

Lemma 4.4 (Correctness and bijectivity of Algorithms 1 & 2)**.** *The following holds,*

- **PtS** *is a bijection from* Par_v^m *to* Seq_v^m.

- **StP** *is a bijection from* Seq_v^m *to* Par_v^m.

Proof. Observe that during the running of the **PtS** and **StP**, Algorithms 1 & 2, the only changes made to the input array L are a sequence of cut & paste transforms $\mathbf{CP}_{i_1,t_1}, \mathbf{CP}_{i_2,t_2} \ldots$. Since each cut & paste transform preserves Property (2) it follows that **PtS** and **StP** preserve (2). Likewise cutting & pasting preserves total length, thus so do **PtS** and **StP**. Recall that the operator $\mathbf{CP}_{(i,t)}$ cuts and pastes the random walk trajectory $(i, t+1), \ldots, (i, \rho_i)$ onto the unique (k, ρ_k) with $L(i,t) = L(k, \rho_k)$. Thus row k in $L' = \mathbf{CP}_{(i,t)}$ is a valid path from vertex $L(0, k)$ to $L(i, \rho_i)$.

For **PtS** we must check that if $L \in \mathrm{Par}_v^m$, then $\mathbf{PtS}(L) \in \mathrm{Seq}_v^m$, i.e. $\mathbf{PtS}(L)$ satisfies (3). Recall that the **PtS** algorithm reads the input array L in sequential order and when a vertex label is seen for the first time at some position (i,j) it applies the cut & paste transform $\mathbf{CP}_{(i,j)}$ and the pointer moves to the next row. If $(i, j+1)$ is non-empty then $\mathbf{CP}_{(i,j)}$ pastes the remainder of row i to some row i' with endpoint value $L(i,j)$. Observe that $i' > i$ since (i,j) is the first occurrence of $L(i,j)$ in sequential order. Thus each new vertex found w.r.t. $<_S$ forms an endpoint as it is cut when it is first discovered and nothing else can be pasted onto that row later by the algorithm. This proves that $\mathbf{PtS}(L)$ is a valid Sequential-IDLA block.

Likewise for **StP** let $L \in \mathrm{Seq}_v^m$ and we check $\mathbf{StP}(L)$ satisfies (4). Suppose when reading L in parallel order (i,j) is the first occurrence of $L(i,j)$, **StP** will apply $\mathbf{CP}_{(i,j)}$ and continue to read the array in parallel order. Position (i,j) is now fixed as the end point of row i as no later copy & paste can alter this row. This holds since to paste something else onto row i we would have to see vertex $L(i,j)$ for the first time (again) later in parallel order which cannot happen.

For injectivity let $\mathbf{F}$ represent either of the maps $\mathbf{PtS}, \mathbf{StP}$, and L, L' be distinct arrays both from Par_v^m or Seq_v^m respectively. Assume for a contradiction that $\mathbf{F}(L) = \mathbf{F}(L')$. Since $L \neq L'$ there is a first position (i,j) at which they differ w.r.t. $<_S$ or $<_P$, i.e. $L(i,j) \neq L'(i,j)$. It cannot be the

15

case that $L(i,j) = \emptyset$ and $L' \neq \emptyset$, or vice versa, since otherwise the arrays must differ at position $(i, j-1)$ which occurs before (i,j) in either ordering. Let (i,j) be the current position when $\mathbf{F}$ is is running on L and L'. If $L(i,j) \notin \mathcal{S}(t,L)$ and $L'(i,j) \notin \mathcal{S}(t,L')$ then $\mathbf{CP}_{(i,j)}$ is applied and the position (i,j) is now fixed in both arrays, i.e. $\mathbf{F}(L)(i,j) \neq \mathbf{F}(L')(i,j)$, a contradiction. Similarly if $L(i,j) \in \mathcal{S}(t,L)$ and $L'(i,j) \in \mathcal{S}(t,L')$ then no transform is applied and the positions are fixed. Otherwise the element at (i,j) is seen in one array and not in the other, i.e. $L(i,j) \notin \mathcal{S}(t,L)$ and $L'(i,j) \in \mathcal{S}(t,L)$. This is a contradiction as (i,j) is the first position at which L and L' differ.

For bijectivity since $\mathbf{StP} : \mathrm{Seq}_v^m \to \mathrm{Par}_v^m$ and $\mathbf{PtS} : \mathrm{Par}_v^m \to \mathrm{Seq}_v^m$ are both injections and $\mathrm{Seq}_v^m, \mathrm{Par}_v^m$ are finite it follows that $|\mathrm{Seq}_v^m| = |\mathrm{Par}_v^m|$. Thus $\mathbf{StP}, \mathbf{PtS}$ are surjections. $\square$

Remark 4.5. *One can prove* $\mathbf{StP}$ *has inverse* $\mathbf{PtS}$, *we omit the proof as we do not use this fact.*

Lemma 4.6. *Let* $L \in \mathrm{Seq}_v^m$. *Then* $\max_{i \in \mathcal{I}_L} \rho_i \leq \max_{i \in \mathcal{I}_{\mathbf{StP}(L)}} \rho_i$.

Proof. Assume for a contradiction that $\max_{i \in \mathcal{I}_L} \rho_i > \max_{i \in \mathcal{I}_{\mathbf{StP}(L)}} \rho_i$. This means that each row attaining maximum length in L must have a section cut and pasted to a row of shorter length by the $\mathbf{StP}$ algorithm. However the $\mathbf{StP}$ algorithm runs in parallel order and cannot paste onto a cell which it has already read. Thus any row suitable to receive the end of the current row must have its end point in the same column or a column to the right of the current one. This cannot decrease the length of the longest row. $\square$

We now have what we need to prove that $\tau_{seq}^v(G) \preceq \tau_{par}^v(G)$ for any G and $v \in V(G)$.

Proof of Theorem 4.1. By Lemma 4.4 $\mathbf{StP}$ is a bijection between Par_v^m and Seq_v^m. Thus we can pair every sequential process L of total length $W(L) = m$ with a unique parallel process L' of total length $W(L') = m$. Both L and L' visit the same vertices with the same frequency and in the same order, thus the probability of each vertex sequence of total length m in either process is the same. This implies that the total lengths of the processes are distributed identically.

Lemma 4.6 states that for this pair the longest row in L' is at least as long as the longest row in L. Thus for any $k, m \geq 0$,

$$\mathbf{Pr}\left[\max_{i \in \mathcal{I}_L} \rho_i \geq k \,\Big|\, W(L) = m \right] \leq \mathbf{Pr}\left[\max_{i \in \mathcal{I}_{L'}} \rho_i \geq k \,\Big|\, W(L') = m \right].$$

This implies the result since $\tau_{seq}^v(G)$ and $\tau_{par}^v(G)$ are given by the length of the longest row in the sequential and parallel processes respectively. $\square$

In the other direction we now prove $\mathbf{E}\left[\tau_{par}^v \right] = \mathcal{O}\big(\mathbf{E}\left[\tau_{seq}^v \right] \cdot \log n \big)$ for any G and $v \in V$.

Proof of Theorem 4.2. Let L be a Parallel-IDLA block and σ be a random permutation of $\{2, \ldots, n\}$. Let $\sigma(L)$ be the block that results from permuting the rows of L using σ. The block $\sigma(L)$ represents a Parallel-IDLA process where conflicts between particles are solved by giving priority to particles with least value of $\sigma(index)$ (instead of least *index*, as per the definition of Parallel-IDLA). Also, for simplicity we fix $\sigma(1) = 1$. Note that L and $\sigma(L)$ have the same rows, and thus the maximum row-length is the same in both blocks. We remark that $\mathbf{PtS}$, Algorithm 2, still produces a valid sequential array even if the input is $\sigma(L)$ instead of L.

Let L be an arbitrary parallel array and consider a run of $\mathbf{PtS}$, Algorithm 2, on $\sigma(L)$ where we do not reveal σ in advance. Instead we reveal the permutation σ row by row as $\mathbf{PtS}$ reads the array in sequential order (in other words, instead of running $\mathbf{PtS}(\sigma(L))$, we equivalently run $\mathbf{PtS}(L)$ but we read rows in random order, starting with row 1 $(= \sigma(1))$ of L, and then rows $\sigma(2), \sigma(3), \ldots, \sigma(n)$. This is equivalent to replacing i by $\sigma(i)$ in lines 1-5 of Algorithm 2). Note that the Cut & Paste

operation is unaffected by not revealing the order of the rows. This holds because the Cut & Paste transform only pastes behind unread rows, independent of their location in the array L and what is more, there is only one row where we can paste a cut section by property (2). Consider the largest row (or choose one arbitrarily if there is more than one) in the original block L. We shall paint this row red and call the last cell ξ. During the running of $\mathbf{PtS}(L)$ the marked cell ξ moves from row to row because of the Cut & Paste operations. Here is the key observation: If ℓ is the length of the original red row and ξ moves no more than N times then in the output array $\mathbf{PtS}(L)$ has a row of length at least ℓ/N. This holds because the red row was partitioned N times and thus one of the pieces has to have length at least ℓ/N

Let i_k be the iteration (how many rows we have read) by the kth time $\mathbf{PtS}$ reads a row containing the marked cell ξ. When we read a row which contains ξ for first time in iteration i_1, we may apply a Cut & Paste somewhere in this row (if not we are done). If so ξ would find itself at the end of an unread row x_2 of L, which will be read in a (random) iteration i_2, i.e. $\sigma(i_2) = x_2$. Note i_2 is a uniform random value in $\{i_1 + 1, \ldots, n\}$. In iteration i_2, we read the row with the marked cell and again, the algorithm might cut and paste this row behind an unread row x_3 which will be read at some time i_3, which is again uniformly random in $\{i_2 + 1, \ldots, n\}$, and so on. Each time we make a cut and paste the index i_j of the recipient row will be in the latter half of the list $\{i_j + 1, \ldots, n\}$ with probability $1/2$. Thus since $\mathbf{PtS}$ works through this list in order the expected length of the list of possible positions for the next value i_{j+1} halves every iteration. We cannot keep halving this list indefinitely because either at some point a row ended by ξ is not cut or ξ is in the last row to be read (which is never cut). Thus the number of times ξ moves (i.e. expected times the longest row is cut) is at most $C \log n$ with probability at least $1/2$ by Markov's inequality. Denote by X the (random) number of times we cut a row containing the marked cell ξ. Let ℓ be the length of the longest row of L, and ℓ' the random variable representing the length of the longest row of $\mathbf{PtS}(L)$ using a random permutation σ. Conditional on cutting L's longest row X times, we have must have at least one row of length ℓ/X once the algorithm has terminated. Thus, given the block L with largest row ℓ, we have

$$\mathbf{E}\left[\ell' \mid L\right] > \mathbf{E}\left[\ell' \mid L, X \leq C \log n\right] \cdot \frac{1}{2} \geq \frac{\ell}{2C \log n}.$$

By taking expectation over all blocks L generated from a Parallel-IDLA with a random σ we conclude the result. $\qquad\square$

4.2 Uniform-IDLA

Recall that in the Sequential-IDLA we run the walks one by one in order and walk $i + 1$ starts only after walk i has settled, while in the Parallel-IDLA all particles walk simultaneously until they settle, breaking ties by settling the particle with smallest index. In either Sequential or Parallel we are interested in the longest walk. Another natural way to run the IDLA process is in uniform order: we choose a random unsettled particle and move it to a random neighbouring vertex which it settles on if unoccupied. We call this process the Uniform-IDLA. This process can be seen as lying between the Sequential and Parallel-IDLA models. To sample from the Uniform-IDLA process, we first consider an infinite sequence $R = (R_i)$ where the R_is are independent random variables sampled from $\{2, \ldots, n\}$. Then we run the Uniform-IDLA as follows: First particle 1 settles at the origin, so the origin is occupied. Then, at each time-step $t \geq 1$, if particle R_t is unsettled, it moves to a random neighbour, otherwise it stays in its current location. If such neighbour is not occupied, particle R_t settles on it and the vertex is now occupied.

Clearly for some sequences R the process may never terminate, for example $R = (1, 1, 1, \dots)$. We say that an sequence R on n indices is index-repeating if for any index $i \in \{1, \dots, n\}$ and any $T \geq 0$, there exits some $t \geq T$ such that $R_t = i$.

Remark 4.7. *For any fixed n and any fixed distribution $\mathcal{D}$ on the indices with full support, the R obtained by sampling indices according to $\mathcal{D}$ will be index-repeating almost surely.*

Given an index-repeating ordering R, we can find a bijection between the Uniform-IDLA and Parallel-IDLA. An R-block is defined in the same fashion as a parallel block, i.e. $L(i, j)$ represents the position of the i-th particle after j jumps, but additionally, we associate to every $(i, j) \in \mathcal{I}_L$ an integer $T(i, j)$. This T is called the timing array and defined as $T(i, j) = t$ if $R_t = i$ for j-th time and $T(i, 0) = 0$ for all particles i. Note that using the block and timing array we can reconstruct the uniform process as we have not only the paths but the time-steps when the particles moved. Whenever we speak of an R-block we shall assume that R is index repeating.

The bijection between an R-block and a parallel block is defined algorithmically in the same fashion as before. To transform an R-block into a parallel block we just apply **StP**, Algorithm 1, to the R-block oblivious to R since **StP** reads in parallel order. However to transform a parallel block into an R-block, we must read the block in the order given by $T(i, t)$ (i.e. read the block with smallest value $T(i, t)$, then the second smallest, etc..) and apply $\mathbf{CP}_{i,j}$ whenever the vertex $L(i, j)$ is read for first time. It is very important that now when applying the Cut & Paste operation we move not only the cells containing a portion of the path but also the times $T(i, t)$ associated to those cells, i.e. if cell (i, t) moves to (j, s) then $T(j, s)$ gets the value of $T(i, j)$, while $T(i, t)$ is left undefined. Pseudo-code for the procedure we have just described is given in Algorithm 3.

Result: transforms a parallel array L and order sequence R into a R-Uniform array
$\mathcal{S} \leftarrow \emptyset$;
$C \leftarrow$ list of cells (i, j) ordered by $T(i, j)$ in increasing order;
$k \leftarrow 0$;
while $|\mathcal{S}| < n$ **do**
$\quad\quad k \leftarrow k + 1$;
1 $\quad\quad (i, j) \leftarrow C(k)$;
2 $\quad\quad$ **if** $L(i, j) \notin \mathcal{S}$ **then**
$\quad\quad\quad\quad \mathcal{S} \leftarrow \mathcal{S} \cup \{L(i, j)\}$;
$\quad\quad\quad\quad L \leftarrow \mathbf{CP}_{(i,j)}(L)$;
$\quad\quad$ **end**
end
return L;

Algorithm 3: Parallel to R-Uniform ($\mathbf{PtU}_R$)

Let $\mathrm{Unif}_{R,v}^m$ be the set of all Uniform-IDLA blocks with ordering R starting from v with total number of steps m. Then, using similar arguments to the sequential-parallel case we obtain.

Theorem 4.8. *For any fixed index-repeating sequence R on n indices, there is a bijection between $\mathrm{Unif}_{R,v}^m$ and Par_v^m. Moreover the number of steps taken by the longest walk of the Uniform-IDLA is stochastically dominated by the number of steps in the longest walk of the Parallel-IDLA.*

Proof. The bijection follows from injectivity and correctness of **StP** and **PtU**$_R$ (as in Theorem 4.4). Then as in the proof of Theorem 4.1 we run **StP** and apply Lemma 4.6. This Lemma still applies as **StP** is oblivious to the ordering of the input array. $\square$

Observe however that the dispersion time of the Uniform array is not determined purely by the number of steps/length of the longest row but by the values $T(i, j)$ of the timing array.

4.3   Continuous-Time IDLA

In this section we consider continuous-time versions of the Sequential and Uniform-IDLA process. By this we mean running these IDLA processes with random walks with exponential rate 1 jumps. We shall need the following concentration result.

Lemma 4.9. *Let X be an* $\mathrm{Gamma}(n, \lambda)$ *random variable and* $Y = \sum_{i=1}^{n} Y_i$ *where* Y_i *are independent* $\mathrm{Geo}(p)$ *random variables. Then for any* $\delta > 0$ *and* $0 < \varepsilon < 1$ *,*

(i) $\mathbf{Pr}\left[X \geq \dfrac{(1 + \delta)n}{\lambda} \right] \leq e^{-\frac{\delta n}{2}}$ *   and   * $\mathbf{Pr}\left[X \leq \dfrac{(1 - \varepsilon)n}{\lambda} \right] \leq e^{-\frac{\varepsilon n}{2}}.$

(ii) $\mathbf{Pr}\left[Y \geq \dfrac{(1 + \delta)n}{p} \right] \leq e^{-\frac{\delta^2 n}{2(1+\delta)}}$ *   and   * $\mathbf{Pr}\left[Y \leq \dfrac{(1 - \varepsilon)n}{p} \right] \leq e^{-\frac{\varepsilon^2 n}{2(1-2\varepsilon/3)}}.$

Proof. Item (i): If X is $\mathrm{Gamma}(n, \lambda)$ then $\mathbf{E}[X] = n/\lambda$ and $\mathbf{E}\left[e^{tX} \right] = (1 - t/\lambda)^{-n}$ for all $t < \lambda$. Now by Markov's inequality for any $t < \lambda$,

$$\mathbf{Pr}[X \geq (1 + \delta)\mu] \leq e^{-t(1+\delta)\mu} \mathbf{E}\left[e^{tX} \right] \leq e^{-t(1+\delta)n/\lambda} (1 - t/\lambda)^{-n} \leq e^{-t(1+\delta)n/\lambda} e^{tn/\lambda} \leq e^{-\delta tn/\lambda}.$$

By considering $-X$ one can also show $\mathbf{Pr}[X \leq (1 - \varepsilon)\mu] \leq e^{-\varepsilon tn/\lambda}$ provided $\varepsilon < 1$. Since $t < \lambda$ was arbitrary the result follows by choosing $t = \lambda/2$.
Item (ii): Following [15], if $Y \geq k\mathbf{E}[Y]$ then we have less than n successes in $k\mathbf{E}[Y]$ Bernoulli trials with success probability p. Thus $\mathbf{Pr}[Y \geq (1 + \delta)n/p] \leq \mathbf{Pr}[\mathrm{Bin}((1 + \delta)n/p, p) < n]$ and

$$\mathbf{Pr}[\mathrm{Bin}((1 + \delta)n/p, p) < n] = \mathbf{Pr}[\mathrm{Bin}((1 + \delta)n/p, p) < (1 + \delta)n - \delta n] \leq e^{-\frac{\delta^2 n}{2(1+\delta)}},$$

by [17, Thm. 3.2]. Similarly for the lower bound

$$\mathbf{Pr}[Y \leq (1 - \varepsilon)n/p] \leq \mathbf{Pr}[\mathrm{Bin}((1 - \varepsilon)n/p, p) \geq n] \leq e^{-\frac{\varepsilon^2 n^2}{2((1-\varepsilon)n + \varepsilon n/3)}} = e^{-\frac{\varepsilon^2 n}{2(1-2\varepsilon/3)}}.$$

$\square$

For the Sequential-IDLA it is easy to consider its continuous-time analogue, the ContSeq-IDLA, we just have random walks that jump at times given by a Poisson process of intensity 1. Also, we can easily sample from the ContSeq-IDLA by sampling a standard (discrete time) Sequential-IDLA and then considering independent exponential times of mean 1 between the jumps. Let $\tau_{c-seq}^{v}(G)$ be the time it took to the slowest particle to settle in the ContSeq-IDLA from $v \in V$.

Theorem 4.10. *Let G and $v \in V(G)$. Then*

(i) $\tau_{c-seq}^{v}(G) = \Theta\big(\tau_{seq}^{v}(G)\big)$ *holds w.h.p. and in expectation.*

If in addition $\tau_{seq}^{v}(G) = \omega(\log n)$ *then*

(ii) $\tau_{c-seq}^{v}(G) = (1 + o(1)) \cdot \tau_{seq}^{v}(G)$ *holds w.h.p. and in expectation.*

Proof. We sample a ContSeq-IDLA by sampling L, a Sequential-IDLA, and a Exp(1) random variable for each walk step in L. Thus conditional on L walk i in the ContSeq-IDLA has length Gamma($\tau_i, 1$). Let $\mathcal{E}_\ell$ be the event that L contains at least one row of length at least $\ell \geq 1$.

For the upper bound conditional on ρ_* being length of the longest row of our sampled Sequential-IDLA we can stochastically dominate the length of any walk in the ContSeq-IDLA by an independent Gamma($\rho_*, 1$) random variable. Observe that $\mathbf{Pr}\left[\tau^v_{c-seq} > \rho_* + \delta\rho_* \mid \rho_*\right] \leq n \cdot e^{-\delta\rho_*/2}$ by Lemma 4.9 (i). Thus conditional on $\mathcal{E}_\ell$ we can take $\delta = (4\log n)/\ell$ so w.p. $1 - o(1/n)$ at most $4\log n$ extra steps are taken by any continuous walk. This gives

$$\mathbf{Pr}\left[\tau^v_{c-seq} \leq (1 + 4\log(n)/\ell)\tau^v_{seq}\right] \geq 1 - \mathbf{Pr}\left[(\mathcal{E}_\ell)^c\right] - o(1/n). \tag{5}$$

Let $\rho_* \geq 1$ be the length of the longest row of L, and observe that $\mathbf{Pr}\left[\tau^v_{c-seq} > \rho_* + i\log n \mid \rho_*\right] \leq ne^{-(i\log n)/2}$ follows by taking $\delta = (i\log n)/\rho_*$ in Lemma 4.9 (i). Thus

$$\mathbf{E}\left[\tau^v_{c-seq} \mid \rho_*\right] \leq \rho_* + 2\log n + (\log n) \cdot \sum_{i=2}^{\infty} ne^{-(i\log n)/2} = \rho_* + \mathcal{O}(\log n). \tag{6}$$

Observe that $\mathbf{E}\left[\rho_*\right] = \mathbf{E}\left[\tau^v_{seq}\right]$ as ρ_* is the longest row of L, thus $\mathbf{E}\left[\tau^v_{c-seq}\right] \leq \mathbf{E}\left[\tau^v_{seq}\right] + \mathcal{O}(\log n)$. Note by definition that if $\mathbf{Pr}\left[\mathcal{E}_\ell\right] = 1 - o(1)$ then $\mathbf{E}\left[\tau^v_{seq}\right] \geq (1 - o(1))\ell$. Lemma 3.11 states that for any graph and any start vertex, $\mathbf{Pr}\left[\mathcal{E}_{(\log n)/14}\right] = 1 - o(1)$. Thus the upper bound in (i) holds in expectation. The upper bound in (ii) follows similarly assuming $\mathbf{Pr}\left[\mathcal{E}_{\omega(\log n)}\right] = 1 - o(1)$. By (5) the upper bound in cases (i) & (ii) also hold w.h.p..

For the lower bounds take one walk from L of the maximum length ρ_* and consider its length in the ContSeq-IDLA process. This has Gamma distribution Gamma($\rho_*, 1$) and thus

$$\mathbf{Pr}\left[\tau^v_{c-seq} < \rho_* - \sqrt{\ell} \mid \rho_*\right] \cdot \mathbf{1}_{\mathcal{E}_\ell} \leq e^{-\sqrt{\ell}/2}$$

by Lemma 4.9 (i). The w.h.p. lower bounds for (i) and (ii) follow by taking expectations of the above, since in both cases $\ell = \Omega(\log n)$ and $\mathbf{Pr}\left[(\mathcal{E}_\ell)^c\right] = o(1)$. This holds in expectation also. $\qquad\square$

It is natural to consider the continuous-time version of the Uniform-IDLA, we call this the CTU-IDLA. In this process each particle has an exponential clock with rate 1. Then, as long as the particle is not settled, when the clock rings the particle moves to a random neighbour and settles if possible. Note that this is equivalent to running the discrete-time Uniform-IDLA with a uniformly random sequence R but waiting an amount of time distributed Exp($1/(n-1)$) between each step in R (recall particle 1 occupies the origin and R_t takes values in $\{2, \ldots, n\}$). Alternatively, we can sample the CTU-IDLA by using $\mathbf{PtU}_R$, Algorithm 3. First, sample a L, a (discrete-time) Parallel-IDLA, then run Algorithm 3 but using a list C built from a timing array T populated as follows: Set $T(i, 0) = 0$ for each i. Then let $T(i, j+1) = T(i, j) + X_{i,j}$ where $\{X_{i,j}\}_{i\in[n],j\in\mathbb{N}}$ are independent Exp(1) random variables. We shall name this procedure $\mathbf{PtU}_C$. This procedure can also be seen as running Algorithm 3 but instead of using the list C to choose the next cell (i, t) (line 1), each row of the block has a exponential clock of mean 1. When the clock of row i rings, the algorithm chooses the first unread cell of row i (if it exists), and proceeds with line 2. One can show this algorithm is (almost surely) correct due the bijection between $\mathrm{Unif}^m_{R,v}$ and Par^m_v for a fixed ordering R established in Theorem 4.8 and Remark 4.7. Let τ^v_{c-unif} be the time it takes the CTU-IDLA started from v to settle all particles.

Theorem 4.11. *Let G be a connected graph and $v \in V(G)$. Then*

(i) $\tau^v_{c-unif}(G) = \Theta\left(\tau^v_{par}(G)\right)$ holds w.h.p. and in expectation.

20

If in addition $\tau_{par}^v(G) = \omega(\log n)$ then

(ii) $\tau_{c-unif}^v(G) = (1 + o(1)) \cdot \tau_{par}^v(G)$ holds w.h.p. and in expectation.

Proof. We can sample a CTU-IDLA as described above by sampling L, a Parallel-IDLA, and running $\mathbf{PtU}_C$ on L. By Theorem (4.8) the longest row of $\mathbf{PtU}_C(L)$ is no longer than the longest row of L, thus conditional on L walk i in the CTU-IDLA has length Gamma$(\tau_i, 1)$. The proofs of the upper bounds in (i) and (ii) now follow by the exact same argument as in the proof of Theorem 4.10. Let $\mathcal{E}_\ell$ be the event that L contains at least one row of length at least $\ell \geq 1$.

For the lower bound take one row from L of the maximum length ρ_* (assume the label of this row is i) and consider the action of $\mathbf{PtU}_C$ on the cells in this row. If no cut is made during the running of $\mathbf{PtU}_C(L)$ then the length of i stochastically dominates an Gamma$(\rho_*, 1)$ random variable. Suppose that a Cut & Paste transform is applied to i at a cell containing vertex v and the remainder of this row is pasted onto row j. Although row j may have contained less cells before v than the number of steps taken by row i to reach v, the amount of time (with respect to the clock) it takes particle j to reach v must be at least as long as the time for particle i to reach v (otherwise the Cut & Paste would not have been applied). Thus conditional on this Cut & Paste the length of row j stochastically dominates an Gamma$(\rho_*, 1)$ random variable. Now, as in Theorem 4.10,

$$\mathbf{Pr}\left[\tau_{c-unif}^v < \tau_{par}^v - \sqrt{\ell} \;\middle|\; \mathcal{E}_\ell\right] = e^{-\sqrt{\ell}/2}$$

by Lemma 4.9 (i). The lower bounds for (i) and (ii) follow since in both cases $\ell = \Omega(\log n)$. $\square$

4.4 Lazy IDLA

Consider the lazy versions of the discrete-time Sequential and Parallel-IDLA models, where with probability $1/2$ particles stay put and otherwise choose a neighbour uniformly. Note that all our previous results using the coupling via the block representation are also valid for lazy walks as for example one can simply consider the graph with the addition of (multi)-loops at each vertex. Indeed, they are valid for any block that is generated by using a Markov chain to move the particles.

Let $\tau_{L-seq}^v(G)$, $\tau_{L-par}^v(G)$, be the number of steps needed to complete the lazy Sequential, respectively lazy Parallel, IDLA process started from v.

Although we are mainly concerned with the simple random walk IDLA models would like to be able to switch to the lazy setting at times as it allows us to use mixing time results. For the Sequential it is fairly clear that up to lower order terms the lazy sequential is a factor of 2 slower than the Parallel, using the continuous time Uniform-IDLA we can also show this for Parallel-IDLA.

Proof of Theorem 4.3. We begin by proving the results for the sequential processes.

Let $\mathcal{E}_\ell$ be the event that L contains at least one row of length at least $\ell \geq 1$. We sample a lazy Sequential-IDLA by coupling with a simple Sequential-IDLA L and adding in lazy steps w.p. $1/2$ for each walk. Thus each row i with length τ_i in the sequential process has length $\sum_{i=1}^{\tau_i} Y_i$ in the coupled lazy process, where the Y_i are independent Geo$(1/2)$ random variables. For the upper bound let ρ_* be the length of L's longest row, then we have $\mathbf{Pr}\left[\tau_{L-seq}^v > 2\rho_* + k2\rho_* \mid \rho_*\right] \leq n \cdot e^{-\frac{k^2 \rho_*}{2(1+k)}}$ by Lemma 4.9 (ii). Similarly to Theorems 4.10 the w.h.p. upper bounds for τ_{L-seq}^v follow by conditioning on $\mathcal{E}_\ell$ for the two cases of ℓ in the statement.

For upper bounds in expectation if we condition on $(\mathcal{E}_\ell)^c$ then $1 \leq \rho_* \leq \ell$, it follow that

$$\mathbf{E}\left[\tau_{L-seq}^v \mid (\mathcal{E}_\ell)^c\right] < 2\ell + 6\ell \log n + (2\ell \log n) \cdot \sum_{i=3}^{\infty} \mathbf{Pr}\left[\tau_{L-seq}^v > \ell + 2i\ell \log n\right]$$

$$\leq \mathcal{O}(\ell \log n) + (2\ell \log n) \cdot \sum_{i=3}^{\infty} e^{-i^2/2(i+1)} = \mathcal{O}(\ell \log n).$$

If $\ell \geq \log n/14$ then by Lemma 4.9 (ii)

$$\mathbf{Pr}\left[\tau_{L-seq}^v > 2\rho_* + \sqrt{i\rho_* \log n} \mid \rho_*\right]\mathbf{1}_{\mathcal{E}_\ell} \leq n \cdot e^{-\frac{i \log n}{2((i \log n)/\rho_* + 1)}} \leq n \cdot e^{-(\sqrt{i} \log n)/10},$$

thus $\mathbf{E}\left[\tau_{L-seq}^v \cdot \mathbf{1}_{\mathcal{E}_\ell} \mid \rho_*\right] = 2\rho_* + \mathcal{O}(\sqrt{\rho_* \cdot \log n})$, similar to (6). By Jensen's (concave) inequality

$$\mathbf{E}\left[\tau_{c-seq}^v\right] \leq 2\mathbf{E}\left[\tau_{seq}^v\right] + \mathcal{O}\left(\sqrt{\mathbf{E}\left[\tau_{seq}^v\right] \cdot \log n}\right) + \mathcal{O}(\ell \cdot \log n) \cdot \mathbf{Pr}\left[(\mathcal{E}_\ell)^c\right]. \qquad (7)$$

Thus for any graph G and $v \in V$, $\mathbf{E}\left[\tau_{L-seq}^v\right] = \mathcal{O}(\mathbf{E}\left[\tau_{seq}^v\right])$ by Lemma 3.11. If $\mathbf{Pr}\left[(\mathcal{E}_\ell)^c\right] \leq 1/\ell$ for some $\ell = \omega(\log n)$ then $\mathbf{E}\left[\tau_{c-seq}^v\right] \leq (2 + o(1))\mathbf{E}\left[\tau_{seq}^v\right]$ by (7).

For the w.h.p. lower bound, conditional in i being a walk of maximum length ρ_* in L, walk i in the L-Seq-IDLA has length $\sum_{i=1}^{\rho_*} Y_i$, where $Y_i \sim \text{Geo}(1/2)$. So by applying Lemma 4.9 (ii),

$$\mathbf{Pr}\left[\tau_{L-seq}^v < 2\rho_* - (\log n)^{1/3}\sqrt{\rho_*} \mid \rho_*\right] \cdot \mathbf{1}_{\mathcal{E}_\ell} \leq e^{-(\log n)^{1/3}/2(1-o(1))} = o(1).$$

Thus, since $\mathbf{Pr}\left[(\mathcal{E}_\ell)^c\right] = o(1)$ for any $G, v \in V$ by Lemma 3.11, talking expectations of the equations above yields $\tau_{L-seq}^v \geq (2 - o(1))\tau_{seq}^v$ w.h.p . Thus this also holds in expectation.

We now prove the bounds for the Parallel processes, the proof technique will be slightly different. First assume that for any $G, v \in V$ and some $\ell = \omega(\log n)$ we have $\mathbf{Pr}\left[(\mathcal{E}_\ell)^c\right] = o(1)$. In this case we know that $\tau_{par}^v(G) = (1 + o(1))\tau_{c-unif}^v(G)$ w.h.p. and in expectation from Theorem 4.11. Consider the CTU-IDLA but using clocks of mean 2 and use $\tau_{2-c-unif}^v(G)$ to denote the dispersion time of this process. It is clear that we can couple the clocks of mean 1 and 2 to give $\tau_{2-c-unif}^v(G) = (2 + o(1))\tau_{c-unif}^v(G)$ w.h.p. and in expectation. Note that sampling from this process is equivalent to sampling from the Uniform-IDLA of mean 1, but ignoring the ring of the clock with probability $1/2$ (Poisson thinning). Consider the graph $\widetilde{G}$, this is G but to each vertex we add as many self loops as neighbours, then $\tau_{c-unif}^v(\widetilde{G})$ has the same distribution as $\tau_{2-c-unif}^v(G)$, likewise $\tau_{par}^v(\widetilde{G})$ and $\tau_{L-par}^v(G)$ are also equidistributed. Theorem 4.11 is then applied to $\widetilde{G}$ yielding $\tau_{2-c-unif}^v(G) = (1 + o(1))\tau_{L-par}^v(G)$ w.h.p. and in expectation. Combining these relations yields

$$\tau_{L-par}^v(G) = \tau_{2-c-unif}^v(G) = (2 + o(1)) \cdot \tau_{c-unif}^v(G) = (2 + o(1)) \cdot \tau_{par}^v(G),$$

w.h.p. and in expectation. For a general graph G and $v \in V$ the exact same argument works however each of the equalities above holds only up to a $\Theta(1)$ factor, the result follows. $\square$

5 Fundamental Networks

In this section we determine the dispersion for many well known graph topologies.

5.1 The Complete Graph

We shall begin with the clique as this is most simple to analyse.

Theorem 5.1. *Let K_n be the complete graph on n vertices and κ_{cc} be as in Lemma 5.2. Then*

$$t_{par}(K_n) \sim \frac{\pi^2}{6} \cdot n \qquad and \qquad t_{seq}(K_n) \sim \kappa_{cc} \cdot n,$$

where

$$\kappa_{cc} := \sum_{i=1}^{\infty} \left(\frac{2}{i(3i-1)} - \frac{2}{i(3i+1)} \right) \approx 1.255.$$

Before proving the above we state a result needed to treat the Sequential-IDLA on cliques.

Lemma 5.2 ([14])**.** *Let $T := T_n$ be the maximum of n independent geometric random variables with parameters $\frac{i}{n}$ for $1 \leq i \leq n$. Then the limit $\lim_{n\to\infty} \mathbf{E}\left[T\right]/n$ exists and is equal to κ_{cc}*

The constant κ_{cc} is related to the longest wait time in the coupon collector process [14].

Proof of Theorem 5.1. Instead of analyzing the parallel process, we analyze the continuous-time Uniform-IDLA process (CTU-IDLA), in which each particle has a exponential clock of rate 1, and moves every time the clock rings until the particle settles. By Theorem 4.8 we have that the dispersion time of the Parallel-IDLA process and the CTU-IDLA process are asymptotically equal as long as the dispersion time of the Parallel-IDLA is $\omega(\log n)$ w.p. $1 - o\left(1/\log^2 n\right)$. The property holds trivially because as the last particle in the Sequential-IDLA takes geometric time of mean n to settle, this holds also for the Parallel-IDLA due to the stochastic domination $\tau_{seq}^v \preceq \tau_{par}^v$ by Theorem 4.1. The analysis of the CTU-IDLA is quite simple: since particles move in continuous-time no two particles settle at the same time. Suppose there are k unsettled particles, then the time needed until one of the k particles settles in one of the k unoccupied vertices is exponentially distributed with mean $(n-1)/k^2$. Summing up from $k = 1$ to $n - 1$ we obtain that the expected dispersion time is asymptotically $n \sum_{k\geq 1} k^{-2} = n \cdot (\pi^2/6 - o(1))$.

For t_{seq} the longest walk in the Sequential-IDLA on K_n is the longest waiting time in the Coupon Collector problem. This time is distributed as the maximum of n independent geometric random variables with parameters $\frac{n-i+1}{n}$ for $1 \leq i \leq n$. The result follows from Lemma 5.2. $\qquad\square$

Remark 5.3. *Observe that $\kappa_{cc} \approx 1.255$ and $\pi^2/6 \approx 1.645$ so the two constants are distinct.*

5.2 The Path

Let P_n be the path with n-vertices. Interestingly, the path provides an example where the sequential and parallel dispersion process take the same time up to lower order terms.

Theorem 5.4. *Let M be the maximum of n independent random variables representing the hitting time of a random walk to the vertex n, starting from 1 on P_n. Then for the dispersion time,*

$$t_{seq}(P_n) = (1 \pm o(1)) \cdot \mathbf{E}\left[M\right] = t_{par}(P_n).$$

Proof. In the following, we will denote by $t_{seq}(m)$ the expected running time of the Sequential-IDLA on a path with m vertices, when the source is the endpoint labelled by 1. In the following, let $Y_1, Y_2, \ldots, Y_n$ be a collection of n independent random variables, each of which describing the hitting time of a random walk from endpoint 1 to $n - n/\log n$ (thus $Y_i = \tau_{hit}(1, n - n/\log n)$). In particular, these random walks will not settle and are therefore completely independent.

The proof will be based on the following chain of inequalities:

$$t_{seq}\left(n - \frac{n}{\log n}\right) \overset{(1)}{\leq} t_{par}\left(n - \frac{n}{\log n}\right) \overset{(2)}{\leq} \mathbf{E}\left[\max_{1 \leq i \leq n - \frac{n}{\log n}} Y_i\right] \overset{(3)}{\leq} (1 + o(1)) \cdot \mathbf{E}\left[\max_{1 \leq i \leq \frac{n}{\log n}} Y_i\right]$$

$$\overset{(4)}{\leq} (1 + o(1)) \cdot t_{seq}(n),$$

and then finally

$$t_{seq}(n) \overset{(5)}{\leq} (1 + o(1)) \cdot t_{seq}\left(n - \frac{n}{\log n}\right),$$

and if all these inequalities hold, the claims of the theorem are established.

Note that inequality (1) is a direct consequence of Theorem 4.1, and inequalities (2) and (4) follow directly from the definition of the Parallel-IDLA and Sequential-IDLA, respectively. Thus it only remains to prove (3) and (5).

We first prove (3) - in fact, for notational convenience we will establish the stronger claim

$$\mathbf{E}\left[\max_{1 \leq i \leq n} Y_i\right] \leq (1 + o(1)) \cdot \mathbf{E}\left[\max_{1 \leq i \leq \frac{n}{\log n}} Y_i\right],$$

i.e., on the left hand side, we take the maximum over n random variables instead of just $n - n/\log n$.

To simplify notation, define $\widetilde{Y} := \max_{1 \leq i \leq n/\log n} Y_i$ and define $Y := \max_{1 \leq i \leq n} Y_i$. In order to prove that $\mathbf{E}\left[\widetilde{Y}\right]$ and $\mathbf{E}\left[Y\right]$ are close, consider a coupling where we first expose the values of the set $\{Y_1, Y_2, \ldots, Y_n\}$ and then assign those values through a random permutation. Next define by F the random variable counting the Y_i's which are larger than $\widetilde{Y}$, in symbols,

$$F := \left|\left\{n/\log n < j \leq n \colon Y_j > \widetilde{Y}\right\}\right|.$$

Next note that for any $\lambda \geq 1$,

$$\mathbf{Pr}\left[F \geq \lambda \cdot \log n\right] \leq \prod_{i=1}^{n/\log n}\left(1 - \frac{\lambda \log n}{n - i + 1}\right) \leq \prod_{i=1}^{n/\log n} \exp\left(-\frac{\lambda \log n}{n}\right) \leq \exp\left(-\lambda\right).$$

The first inequality holds by considering the probability that random ordering does not "choose" any of the $\lambda \log n$ longest walks for one of the first $n/\log n$ walks. I.e. if we have chosen k so far, non of them being one of the $\lambda \log n$ longest, then we choose a long walk next time w.p. $(\lambda \log n)/k$. Thus for $\lambda = 2 \log n$, $\mathbf{Pr}\left[F \geq 2 \log^2 n\right] = n^{-2}$.

Consider now the gap between the $(2 \log^2 n)$-th largest element of the values $\{Y_1, Y_2, \ldots, Y_n\}$ and the maximum. To this end, we will use the principle of deferred decisions and expose the n trajectories in parallel order and stop as soon as there at most $2 \log^2 n$ walks which have not hit the other endpoint.

Hence suppose we order these values such that w.l.o.g. $Y_1 \leq Y_2 \leq \cdots \leq Y_n$. Then for any $j \geq n - 2 \log^2 n$, the random variable $Y_j - Y_{n - 2 \log^2 n}$ is stochastically dominated by one plus the hitting time from 1 to n, so in particular, $\mathbf{E}\left[Y_j - Y_{n - 2 \log^2 n}\right] = \mathcal{O}(1 + n^2)$. Furthermore, using the fact that from any start point, a random walk reaches the vertex n is at most $2n^2$ steps with probability at least $1/2$, it follows that for any $\lambda > 0$,

$$\mathbf{Pr}\left[Y_j - Y_{n - 2 \log^2 n} > 1 + \lambda \cdot 2n^2\right] = \mathcal{O}(2^{-\lambda}). \tag{8}$$

Choosing $\lambda = C \log \log n$ for some large constant $C > 0$, it follows by the Union bound over the at most $\log^2 n$ indices $j \in F$ that

$$\mathbf{Pr}\left[Y \geq \widetilde{Y} + \mathcal{O}(n^2 \log \log n) \mid F \leq 2 \log^2 n\right] \leq \frac{1}{\log^2 n}.$$

To conclude, it follows by the Union bound that w.p. at least $1 - 3/(\log n)^2$, our coupling satisfies

$$Y - \widetilde{Y} \leq C \cdot n^2 \log \log n.$$

Otherwise, we still have $\mathbf{E}\left[Y - \widetilde{Y} \mid \mathcal{E}\right] = \mathcal{O}(n^2 \log n) + \mathcal{O}(n^2 \log \log n)$, where $\mathcal{E}$ denotes the event that any of the above probabilistic arguments fail. The result follows since $\mathbf{Pr}\left[\mathcal{E}\right] = \mathcal{O}\left(1/\log^2 n\right)$.

We now continue to prove inequality (5). To this end we will construct a coupling between the n walks in $t_{seq}(n)$ and the $n - n/\log n$ walks in $t_{seq}(n - n/\log n)$. Consider the first $n/\log n$ random walks in the $t_{seq}(n)$ setting. For each of them, the expected time to settle is $\mathcal{O}(n^2/\log^2 n)$ and by an argument similar to (8), none of them will take more than $\mathcal{O}(n^2)$ with probability $1 - n^{-\omega(1)}$.

The trajectories of the next $n - n/\log n$ walks of $t_{seq}(n)$ can be coupled with the ones in $t_{seq}(n - n/\log n)$, so if a walk moves from vertex x to $x+1$ in $t_{seq}(n - n/\log n)$, then the corresponding walk in $t_{seq}(n)$ moves from $x + n/\log n$ to $x + 1 + n/\log n$. The only difficulty arises when the walk in $t_{seq}(n)$ is at a vertex between 1 and $n - n/\log n$. To capture this, we will consider so-called excursions which are epochs in which the random walk is at such a vertex. Notice that the total number of steps that are taken as part of any excursion is at most the total number of visits to any vertex in $1, 2, \ldots, n/\log n$. However, note that the expected number of visits to any of these vertices is $\mathcal{O}(n \log n)$ for a random walk of $\mathcal{O}(n^2 \log n)$ steps, and thus by a standard Chernoff Bound for random walks, it follows that any of these vertices is visited at most $\mathcal{O}(n \log n)$ times with probability at least $1 - n^{-2}$. Thus by the Union bound, the total number steps spend in any excursion is at most $\mathcal{O}(n^2)$ with probability at least $1 - n^{-1}$.

To conclude, we have shown that with probability at least $1 - n^{-1}$ there is a coupling between $\tau_{seq}(n)$ and $\tau_{seq}(n - n/\log n)$ such that

$$\tau_{seq}(n) \leq \tau_{seq}(n - n/\log n) + \mathcal{O}(n^2).$$

Note that we can verify whether this coupling holds by inspecting only the first $\mathcal{O}(n^2 \log n)$ steps of the random walks. Thus even conditional on the coupling failing, we have $t_{seq}(n) = \mathcal{O}(n^2 \log n)$. Since $t_{seq}(n) = \Omega(n^2)$, it follows that for the expected values,

$$t_{seq}(n) \leq (1 + o(1)) \cdot t_{seq}(n - n/\log n).$$

$\square$

5.3 Expanders and The Hypercube

We call a graph an expander if $1 - \lambda_2 = \Omega(1)$, where λ_2 is the second largest absolute eigenvalue.

Theorem 5.5. *Let G be an n-vertex almost-regular expander graph. Then $t_{seq}(G), t_{par}(G) = \Theta(n)$.*

Proof. The lower bound for t_{seq} follows from Theorem 3.7. The upper bound on $t_{par}(G)$ follows from Corollary 3.5. The result then follows since $t_{seq}(G) \leq t_{par}(G)$ by Theorem 4.1. $\square$

Remark 5.6. *In particular this result covers (w.h.p.) random d-regular graphs, for fixed d, and the binomial random graph $\mathcal{G}(n, p)$ above the connectivity threshold, when $np \geq c \log(n)$, $c > 1$.*

The Hypercube H_d, where $n = 2^d$, is the graph where each vertex is a binary string of length d and two vertices are connected if their associated binary strings differ in one digit. The hypercube is not an expander since $1 - \lambda_2 = 1/d = 1/\log_2 n$ however, we still achieve a linear bound.

Theorem 5.7. *Let H_d be the hypercube with $n = 2^d$ vertices. Then $t_{seq}(H_n), t_{par}(H_d) = \Theta(n)$.*

Proof. The lower bound for t_{seq} follows from Theorem 3.7. Due to Theorem 4.1 we only need to find an upper bound for t_{par}. As laziness only changes the dispersion time by a constant factor, we work with lazy walks. For the upper bound we seek to apply Theorem 3.3 however, unlike in Theorem 5.5, we shall use an argument based on return probabilities in H_n to bound hitting times rather than appealing to Lemma A.2. Also note that, since the sum in Theorem 4.1 only has $\mathcal{O}(\log n)$ terms and by monotonicity of hitting times of sets, it will be sufficient to cover the case $1 \le |S| \le (\log n)/2$. If we can prove that the hitting time is $\mathcal{O}(n/|S|)$ in this case then we are done. We divide time into epochs of length $2\log^2 n$ and prove the probability we hit S in one epoch is at least $\Omega((\log n)^2|S|/n)$. In the first $\log^2 n$ steps of an epoch we allow the walk to mix ignoring if the walk hits S. Then, with high probability we can couple our walk with the stationary distribution. In the second $\log^2 n$ steps of an epoch we observe if the walk hits S. Let Z the random variable which counts the number of visits to the set S in $(\log n)^2$ steps. Then $\mathbf{Pr}_\pi[\tau_S \le (\log n)^2] = \mathbf{Pr}_\pi[Z \ge 1]$ and

$$\mathbf{Pr}_\pi[Z \ge 1] = \frac{\mathbf{E}_\pi[Z]}{\mathbf{E}_\pi[Z|Z \ge 1]} \ge \frac{(\log n)^2|S|/n}{\max_{u \in S} \sum_{t=0}^{(\log n)^2} \widetilde{p}_{u,S}^t}.$$

Claim 5.8. *For any set S and $u \in S$ if $|S| \le (\log n)/2$, then $\sum_{t=0}^{(\log n)^2} \widetilde{p}_{u,S}^t = \mathcal{O}(1)$.*

Thus by Claim 5.8 (proved later) $\mathbf{Pr}_\pi[\tau_S \le (\log n)^2] = \Omega\big(\Omega((\log n)^2|S|/n)\big)$ and so $t_{hit}(\pi, S) \le \mathcal{O}(\max\{n/|S|, n/\log n\})$. Thus as discussed earlier the upper bound follows from Theorem 3.3. $\square$

The proof of Claim 5.8 will make heavy use of [19, Lem. 7], we paraphrase it hear for convenience:

Lemma 5.9. *Let $\mathcal{W}(i)$, $i \ge 0$ be the lazy walk in H_d and $T = (\log n)^2$. Then, for any $v \in V$,*

(i) $R_v := \sum_{i=0}^{T} \widetilde{p}_{u,u}^t = 2 + \dfrac{2}{d} + \mathcal{O}\left(\dfrac{1}{d^2}\right).$

(ii) Suppose $W(0)$ is at distance at least 2 from v (resp. at least 3 from v). The probability W visits $\Gamma(v)$ within $L = O(T \log n)$ steps is $P(2, L) = O(1/d)$ (resp. $P(3, L) = O(1/d^2)$).

(iii) Let $C \subseteq N(v)$. For a walk starting from $u \in C$, let R_C denote the expected number of returns to C within T steps. Then, in the lazy walk, $R_C = 2 + \mathcal{O}(1/d)$.

Proof of Claim 5.8. Let $C = \{u\} \cup (\Gamma(u) \cap S)$. Let $R_2(u,t)$ (resp. $R_{\ge 3}(u,t)$) be the expected number of visits to a vertex at distance 2 (resp. distance ≥ 3) from u before time t. Then if $T = (\log n)^2$ we have

$$\sum_{i=1}^{(\log n)^2} \widetilde{p}_{u,S}^i \le R_C + \frac{\log n}{2} \cdot R_2(u,T) + \frac{\log n}{2} \cdot R_{\ge 3}(u,T), \tag{9}$$

where the first term counts returns to u and the portion of S in u's neighbourhood, the second term counts visits to members of S at distance 2 and the third to those at distance 3 or greater (where we recall that $|S| \le (\log n)/2$.

We have the crude bound $R_2(u,T) \le P(2,T) \cdot R_v$, where v is any vertex by transitivity of H_d. Thus by Lemma 5.9, $R_2(u,T) \le \mathcal{O}(1/d) \cdot (2 + \mathcal{O}(1/d)) = \mathcal{O}(1/d)$. Similarly $R_{>3}(u,T) \le P(3,T) \cdot R_v = \mathcal{O}(1/d^2)$. Note $R_C = \mathcal{O}(1)$ by Lemma 5.9 (ii). It follows from (9) that $\sum_{i=1}^{T} \widetilde{p}_{u,S}^i = \mathcal{O}(1)$. $\square$

5.4 Tori and Grids

Let $B(r) := \left\{ \mathbf{x} \in \mathbb{Z}^d : x_1^2 + \cdots + x_d^2 \leq r^2 \right\}$ be the ball of radius r in $\mathbb{Z}^d$.

Lemma 5.10. *Let $d = 1, 2$ be fixed. For any $\beta > 0$ there exists some $C > 0$ such that the random walk of length $Ct \log t$ from the origin in $\mathbb{Z}^d$ does not exit $B(\sqrt{t})$ with probability at least $1/t^\beta$.*

Proof. Let S_j be the position of a random walk at time j started from 0. For $t > 0$ let $\mathcal{E}_0$ be the event $\left\{ S_j \in B(\sqrt{t}/2) \text{ for all } 0 \leq j \leq c^2 t - 1 \right\}$. By the Central Limit Theorem [34] for all $\varepsilon > 0$ there is some $c > 0$ such that for large t

$$\mathbf{Pr} \left[\frac{S_{c^2 t}}{\sqrt{c^2 t}} \notin B \left(\frac{\sqrt{t}/2}{\sqrt{c^2 t}} \right) \right] \leq \left(1 + \mathcal{O} \left(\frac{1}{\sqrt{t}} \right) \right) \int_{\mathbb{R}^2 \setminus B\left(\frac{1}{2c}\right)} \frac{e^{-|\mathbf{x}^2|}}{\pi} \, \mathrm{d}\mathbf{x} \leq \varepsilon.$$

Thus by the Reflection Principal [36, Prop. 1.6.2] the probability a random walk stays within the ball $B\left(\sqrt{t}/2\right)$ for $c^2 t$ units of time is at least $1 - 2\varepsilon$. For $i \geq 1$ let $\mathcal{E}_i$ be the event

$$\left\{ S_j \in B(\sqrt{t}) \text{ for all } i \cdot c^2 t \leq j \leq (i+1) \cdot c^2 t - 2 \right\} \cap \left\{ S_{(i+1) \cdot c^2 t - 1} \in B(\sqrt{t}/2) \right\}.$$

By geometric considerations we see that $\mathbf{Pr}\left[\mathcal{E}_{i+1} | \mathcal{E}_i \right] \geq (1 - 2\varepsilon)/2d \geq 1/(2d+1)$ for small enough $c > 0$. Observe that $\left\{ S_k \in B(\sqrt{t}) \text{ for all } 0 \leq k \leq \alpha c^2 t \log t \right\} \supseteq \bigcap_{i=0}^{\alpha \log t} \mathcal{E}_i$, for any $\alpha > 0$. Thus for any fixed $\beta > 0$ provided $\alpha \leq \beta / \log(2d+1)$ we have

$$\mathbf{Pr}\left[S_k \in B(\sqrt{t}) \text{ for all } 0 \leq k \leq \alpha c^2 t \log t \right] \geq (1 - 2\varepsilon) \left(\frac{1}{2d+1} \right)^{\alpha \log t} \geq \frac{1}{t^\beta}.$$

The result follows by taking $c > 0$ small enough. $\qquad\square$

Theorem 5.11. *For the path/cycle, $\Theta(n^2 \log n)$ steps are needed in expectation and with probability at least $1 - o(1)$.*

Proof. The upper bound for either graph follows from Lemma 3.1. For the lower bound in the cycle if at some time an interval $[-a, b]$ has been settled around the origin then by the gamblers ruin formula the end point closest to the origin receives the next particle with probability at least $1/2$. Thus by a Chernoff bound w.h.p. after $2n/3$ particles have settled the interval $[-n/4, n/4]$ is occupied. Each of the remaining $n/3$ particles must exit the ball $B(n/4)$ in order to settle. Thus by Lemma 5.10 there is some $C > 0$ such that the probability that one walk takes longer than $Cn^2 \log(n)$ to exit $B(n/4)$ is at least $1 - (1 - 1/n^\beta)^{n/3} = 1 - o(1)$. The result for the path by similarly considering a return to the origin as a change in parity for a walk on the cycle and adding settled vertices to both ends simultaneously. $\qquad\square$

The next result does not settle the dispersion time on the two-dimensional grid, but improves on the trivial $\Omega(n)$ bound.

Proposition 5.12. *Let G be either the finite box $\left[-\lfloor \sqrt{n}/2 \rfloor, \lfloor \sqrt{n}/2 \rfloor \right]^2 \subset \mathbb{Z}^2$ in the two-dimensional grid, or the two-dimensional finite torus on n vertices. Then $t_{seq}(G), t_{par}(G) = \Omega(n \log n)$.*

Proof. We will prove the lower bound for t_{seq} only, since the corresponding lower bound for t_{par} will follow from $t_{par} \geq t_{seq}$.

Let $A(t)$ denote the aggregate of the Sequential-IDLA once t particles have settled. Theorem 1 of [30] states that for each γ there exists an $a = a(\gamma) < 1$ such that for all sufficiently large r,

$$\mathbb{P}\left[B(r - a \log r) \subseteq \mathcal{A}(\pi r^2) \subseteq B(r + a \log r) \right] \geq 1 - r^{-\gamma}. \tag{10}$$

27

We can couple the process on G with the process on $\mathbb{Z}^2$ up until the point t^* when the first particle settles a vertex on the boundary (or wraps around in the torus). By (10) we can condition on the aggregate $A(t^*)$ containing a ball of radius $\lfloor\sqrt{n}/2\rfloor - a\log n$ w.h.p., for some $a < \infty$. Thus the remaining $n - t^* > (1 - \pi/4)n > n/5$ particles must all exit the ball $B(\sqrt{n}/3)$ before settling. Now by Lemma 5.10 the probability that one walk takes longer than $Cn\log n$ to do this is at least $1 - (1 - 1/n^\beta)^{n/5} = 1 - o(1)$. The result follows. $\qquad\square$

Theorem 5.13. *Let G be the d-dimensional torus/grid where $d \geq 3$. Then $t_{seq}(G), t_{par}(G) = \Theta(n)$.*

Proof. The lower bound for t_{seq} follows from Theorem 3.7. For the d-dimensional torus/grid we have the well-known bound $p_{u,v}^t \leq 1/n + \mathcal{O}(t^{-d/2})$. This estimate applied in combination with Lemma A.2 to Theorem 3.3 implies a bound of $\mathcal{O}(n)$ on the dispersion time whenever $d \geq 3$. $\quad\square$

5.5 Binary Tree

In this section we consider the binary tree T_n with $n + 1 = 2^k - 1$ vertices, 2^{k-1} leaves and root r.

Theorem 5.14. *For the binary tree T_n with root r, we have $\tau_{seq}^r, \tau_{par}^r = \Theta\big(n(\log n)^2\big)$ w.h.p. and in expectation, consequently $t_{seq}, t_{par} = \Theta\big(n(\log n)^2\big)$.*

Recall that the hitting time in the Binary tree with n vertices is $\mathcal{O}(n\log n)$. Thus, by Theorem 3, the dispersion time of the Parallel-IDLA process is $\mathcal{O}(n(\log n)^2)$ w.h.p. and in expectation. Thus to establish Theorem 5.14 it remains to show that the dispersion time of Sequential-IDLA is at least $\Omega(n(\log n)^2)$ w.h.p., proving that $t_{seq} = \Theta(t_{par}) = \Theta(n(\log n)^2)$, due to Theorem 4.1.

To prove the lower bound we show the last $poly(n)$ unoccupied vertices are clustered in such a way that one of the last $poly(n)$ walks will have trouble finding the cluster. The first step of this strategy is to establish the following lemma, which in some sense a shape theorem for the binary tree.

Lemma 5.15. *Consider a complete binary tree with $n = 2^k - 1$ vertices and the root r being the source of the Sequential-IDLA. Let τ be the first time when one of the two sub-trees with $2^{k-1} - 1$ vertices is completely filled and fix $0 < \varepsilon < 1/4$. Then with probability at least $1 - 2n^{-\varepsilon}$, the other sub-tree still has at least $n^\varepsilon/(3\log_2 n)$ unoccupied vertices at time τ.*

The lemma above allows us to show that after some time in the process all the remaining unsettled vertices are contained in a sub-tree of significant distance from the root. The next lemma says that w.h.p. one of the remaining walks takes a long time to enter the sub-tree.

Lemma 5.16. *Let u be an arbitrary but fixed vertex which has distance $\varepsilon\log_2 n$ from the root, where $0 < \varepsilon \leq 1$ is some constant. For any given $c > 0$ there exists $c' > 0$ such that a random walk of length $c'\varepsilon n\log^2 n$ starting from the root r visits u with probability at most $1 - n^{-c}$.*

These two lemmas are the main technical component of this chapter and are proved in Sections 5.5.1 and 5.5.2 respectively, first we shall prove Theorem 5.14.

Proof of Theorem 5.14. As mentioned above it suffices to prove a w.h.p. lower bound on τ_{seq}^r. Suppose $n = 2^k - 1$ and let r denote the root of the binary tree. Let $T_1, T_2, \ldots, T_x$ with $x = 2^{\lfloor k/16\rfloor} \leq n^{1/32}$ be a labelling of all sub-trees whose root is at distance $\lfloor k/32\rfloor$ form the root r. Note each of those sub-trees has $2^{k-\lfloor k/32\rfloor} - 1 = (1 + o(1))n^{31/32}$ vertices.

Observe that whenever a particle enters to one of those sub-trees, we can imagine the filling process as an independent IDLA process on that sub-tree. Indeed, when a particle moves inside such

a tree, it is moving as a random walk on the tree until it settles, and if the particle leaves the sub-tree from the root, we can imagine we pause the process until a new particle arrives again, restarting the process. From the previous observation we can apply Lemma 5.15 above with $\varepsilon = 1/8$, it follows that whenever we fill one of the sub-trees of T_i, the other sub-tree has at least $\Omega\left(\frac{n^{31\varepsilon/32}}{\log_2 n}\right) \geq n^{1/9}$ with probability at least $1 - 2n^{31\varepsilon/32}$. By the union bound, the above happens for all the sub-trees $T_1, \ldots, T_x$ at the same time with probability at least $1 - 2n^{31\varepsilon/32}n^{1/32} \geq 1 - n^{-\Theta(1)}$.

Now, consider all the sub-trees at distance $\lfloor k/32 \rfloor + 1$ from the root, and suppose that just after settling the i-th particle, all but one of those sub-trees are filled. Without lost of generality, we assume that such a sub-tree is a sub-tree of T_1, and denote the left and right sub-trees of T_1 by T_{11} and T_{12}. Additionally, suppose that that T_{11} is filled just after settling the i-th particle. We conclude that after settling the i-particle, all trees $T_2, \ldots, T_x$ are filled, and since T_{11} just became filled, we deduce that all the remaining unsettled vertices are located in T_{12} and there are at least $n^{1/9}$ of them. Choosing $c = 1/10$ in Lemma 5.16 below gives that one of the $M \geq n^{1/9}$ remaining walks take longer than $c'\varepsilon n \log^2 n$ hit the sub-tree T_1 with probability at least $1 - \left(1 - n^{-1/10}\right)^{n^{1/9}} \geq 1 - o(1)$. Concluding that it takes $\Omega(n \log^2 n)$ to settle all particles w.h.p. $\qquad\square$

5.5.1 Proof of Lemma 5.15

We begin with the following simple lemma needed to prove Lemma 5.15.

Lemma 5.17. *In the binary tree T_n of height k, the probability that a fixed leaf u is visited before the walk returns to the root r is $1/(2(k-1))$.*

Proof. The formula $\mathbf{Pr}\left[\text{A Random Walk from } r \text{ hits } u \text{ before returning to } r\right] = (R(r,u) \cdot d(r))^{-1}$ can be found in [37, Prop. 9.5.]. The result follows since the resistance $R(r,u)$ in a tree is given by graph distance and the degree of the root, $d(r)$, is 2. $\qquad\square$

Proof of Lemma 5.15. We divide the Binary tree into a root, and a left and right sub-tree. To study the IDLA process, we consider the following algorithm. Consider an infinite sequence of (independent) random walks starting in the root of the left-tree. These walks finish when they hit the root of the original tree. We also consider an (independent) infinite sequence for the right sub-tree. To run the IDLA process, we start in the root of the binary tree and settle the first particle. From the second particle on, each time a particle is in the root it moves to the left or right sub-tree with probability $1/2$. The i-th time a particle moves to the left (right) sub-tree, it follows deterministically the i-th predetermined walk until it reaches a vertex for first time or returns to the root of the binary tree. The advantage of this procedure is that once we predetermine the infinite random walk sequences in the left and right sub-tree, we know the number of times particles need to move from the root either to the left sub-tree or to the right sub-tree in order to fill the left and right sub-trees respectively. Let us call such quantities, the number of visits to each sub-tree required to fill it, L and R (for the left and right sub-trees). Note that $n/2 \leq L, R$ because we need to move at least $n/2$ times to the left (right) sub-tree in order to fill it. We prove the following property of the predetermined walks: Let S be the number of walks needed to cover the last $n^\varepsilon/(3\log_2 n)$ unoccupied vertices of the left (or right) sub-tree. Define the event $\mathcal{E}_1 = \{S < n^\varepsilon\}$. We prove that $\mathcal{E}_1$ occurs w.h.p., indeed,

$$\mathbf{Pr}\left[S \geq n^\varepsilon\right] \leq \mathbf{Pr}\left[Bin\left(n^\varepsilon, \frac{1}{2\log_2 n}\right) \leq \frac{n^\varepsilon}{3\log_2 n}\right] \leq \exp\left(-\frac{n^\varepsilon}{72\log_2 n}\right). \qquad (11)$$

In the first inequality follows from Lemma 5.17 (hitting a leaf is harder than hitting a non-leaf in a excursion), for second inequality we use Chernoff's bounds. Therefore, with probability at least

$1 - 2\exp(-\frac{n^\varepsilon}{72\log_2 n})$, the last n^ε walks cover at least $n^\varepsilon/(3\log_2 n)$ unoccupied vertices of the left (or right) sub-trees.

From now, we assume all the walk in the left (right) sub-trees are predetermined. Let W_j be 1 if the j-th time a particle leaves the root moves to the left sub-tree, $W_j = 0$ otherwise. Denote $L_i = \sum_{j=1}^i W_j$ and $R_i = n - L_i$. Define $\tau = \min\{i \geq 1 : L_i = L \text{ or } R_i = R\}$.

Claim 5.18. *For $\varepsilon < 1/4$ it holds that $\max\{R - R_\tau, L - L_\tau\} \geq n^\varepsilon$ with probability at least $1 - n^{-\varepsilon}$.*

The proof of the claim is temporally deferred. The claim above essentially tells us that when we fill one sub-tree, the other needs at least n^ε more walks to be filled with high probability. Denote $\mathcal{E}_2 = \{\max\{R - R_\tau, L - L_\tau\} \geq n^\varepsilon\}$. Note that the statement of this Lemma follows from proving that $\mathcal{E}_1 \cap \mathcal{E}_2$ holds with probability at least $1 - 2n^{-\varepsilon}$. By (11) and Claim 5.18 we have

$$\mathbf{Pr}\left[(\mathcal{E}_1 \cap \mathcal{E}_2)^c\right] \leq \mathbf{Pr}\left[\mathcal{E}_1^c\right] + \mathbf{Pr}\left[\mathcal{E}_2^c\right] \leq 2\exp\left(-\frac{n^\varepsilon}{72\log_2 n}\right) + n^{-\varepsilon}+ \leq 2n^{-\varepsilon}.$$

$\square$

Proof Of Claim 5.18. Recall that after the i-th time a particle moves from the root to one of the sub-trees, we have $R_i + L_i = i$. Also, if such particle moves to the left sub-tree $L_i = L_{i-1} + 1$ and $R_i = R_{i-1}$ (similarly if the particle moves to the right sub-tree). We can see the process as balls into 2 bins (left and right bins). At each round we allocate a ball to one of the bins at random. The process finishes when the left bin has L balls or when the right bin has R balls, but for convenience we allow the process to keep adding balls after such a point. We work with a continuous time version of this process where balls arrive to each bin following independent Poisson processes $N_l(t)$ and $N_r(t)$ of rate 1 for the left and right bin, respectively. Let τ_l (resp. τ_r) be the first time t such that $N_l(t) \geq L$ (resp. $N_r(t) \geq R$). Consider the time $\tau_l = t$ and consider the load of the other bin $N_r(t)$. First, note that as $L, R \geq n/2$, the event $\{t \leq 15n^{4\varepsilon}\}$ occurs only with probability at most $\exp(-n^{\Omega(1)})$ (using a Chernoff bound) and therefore in the remainder of the proof we will assume $t \geq 15n^{4\varepsilon}$. Note that for any $\tau_l = t$, the load of the right bin is exactly a Poisson random variable with parameter t. For any integer x, $\mathbf{Pr}\left[Poi(t) = x\right] \leq 2/\sqrt{2\pi t}$ thus using the lower bound on t

$$\mathbf{Pr}\left[|R - N_r(\tau_l)| < n^\varepsilon | \tau_l = t\right] \leq 2n^\varepsilon \cdot (2/\sqrt{2\pi t}) \leq n^{-\varepsilon}/2.$$

Analogous arguments work for τ_r and $|L - N_l(\tau_r)|$. By the union bound the result holds. $\square$

5.5.2 Proof of Lemma 5.16

Lemma 5.19. *Let $c > 0$ be fixed. Then, a random walk (X_t) of length $n\lceil c(k-1)^2\rceil/3$ on T_n starting from the root r visits an arbitrary but fixed leaf u w.p. at most $1 - e^{-c/2} \cdot n^{-c/(2\log 2)}$.*

Proof. First note that by Lemma 5.17, it follows that a random walk does not visit leaf u before the $\lceil c(k-1)^2\rceil$-th return to the root with probability at least

$$\left(1 - \frac{1}{2(k-1)}\right)^{\lceil c(k-1)^2\rceil} \geq e^{-ck/2 - (c/4)(1+o(1))} \geq \left(\frac{1}{n}\right)^{c/(2\log 2)} \cdot e^{-c/3},$$

where the second inequality due to the fact that $n - 1 = 2^k - 1$ and thus $\log n = k\log 2$.

Consider now a random walk of length $\ell = dn\lceil c(k-1)^2\rceil$ for some constant $d > 0$. We wish to show we have that not too many excursions (visits to the root r) during ℓ times w.h.p. Let L be

the set of leaves and r be the root. Let τ_A, (τ_A^+) be the first hitting (return) time of the vertex/set A by the random walk X_t. By [37, Prop. 9.5.] we have

$$\mathbf{P}\left[\tau_L < \tau_r^+ \,\middle|\, X_0 = r\right] = 1/(2 \cdot 2) = 1/4 \quad \text{and} \quad \mathbf{P}\left[\tau_r < \tau_L^+ \,\middle|\, X_0 \in L\right] = 1/(2 \cdot (n/2)) = 1/n. \quad (12)$$

To simplify the analysis we shall consider only times when the walk is at the root or the leaves reducing the tree to a two state Markov chain. Indeed, we start the walk at the root and say it jumps to a leaf w.p. $1/4$, once at a leaf it can jump to the root w.p. $1/n$.

To bound the number of visits to r from above we can assume that each attempt to get from r to L (or L to r) takes at least 2 units of time. Thus we have at most $\ell/2$ tries to hit the root from L and the number of successes is dominated by a Binomial r.v. with parameters $\ell/2$ and $1/n$ by (12). Thus we hit r from L at most $t_1 = (1 + 1/10)\ell/(2n)$ times w.p. $1 - n^{-\omega(1)}$ by a Chernoff bound. Let R_i be the number of returns to r by the random walk from r on its ith trip to r before hitting L again, note that R_i is geometrically distributed with parameter $1/4$ by (12). Thus if we let $Y(t_1) = \sum_{i=0}^{t_1} R_i$ be the number of returns to r during a random walk of length ℓ conditional on t_1 successful returns to r from L, then by Lemma 4.9 (ii)

$$\mathbf{Pr}\left[Y(t_i) > 9t_1/2\right] \leq \exp\left(-9t_1\left(1 - 8/5\right)^2/4\right),$$

holds for any fixed $d > 0$. Then, by taking $d = 1/3$, with probability at least $1 - 2n^{-\omega(1)}$ the number of returns to r (and excursions from r) is bounded by $9t_1/2 \leq (9/2) \cdot (11/10)\ell/(2n) < \lceil c(k-1)^2 \rceil$.

Thus we have

$$\mathbf{Pr}\left[\tau_{hit}(r,u) > n\lceil c(k-1)^2 \rceil/3\right] \geq \mathbf{Pr}\left[X_t \text{ does not visit } u \text{ in the first } \lceil c(k-1)^2 \rceil \text{ excursions}\right]$$
$$- \mathbf{Pr}\left[\text{There are more than } \lceil c(k-1)^2 \rceil \text{ excursions}\right]$$
$$\geq e^{-c/3} \cdot n^{-c/(2\log 2)} - 2n^{-\omega(1)}.$$

The proof follows from noting the above is greater than $e^{-c/2} \cdot n^{-c/(2\log 2)}$ for large n. $\square$

Finally, we can now extend the result from the previous lemma to internal vertices, and prove the Lemma 5.16, the a key Lemma about hitting time of clustered sets.

Proof of Lemma 5.16. Let $\widetilde{T}$ be the top of the binary tree T, this is the tree induced by all vertices that have distance at most $\varepsilon \log_2 n$ from the root. Let $\widetilde{L}$ be the set of leaves in $\widetilde{T}$. By Lemma 5.19, we know from that given $c > 0$, a random walk of length $c\varepsilon^2 n^\varepsilon \log^2 n/3$ on $\widetilde{T}$ does not visit a vertex $u \in \widetilde{L}$ with probability at least $n^{-c\varepsilon/(2\log 2)}$. By the random walk Chernoff bound [18] the random walk on $\widetilde{T}$ makes at least $\nu = c\varepsilon^2 n^\varepsilon \log^2 n/7$ visits to $\widetilde{L} \setminus \{u\}$ with probability at least

$$1 - \sqrt{n^\varepsilon} \cdot \exp\left(-\frac{(1/7)^2 \cdot c\varepsilon^2 n^\varepsilon \log^2 n}{6 \cdot 72 \cdot t_{mix}\left(\widetilde{T}\right)}\right) = 1 - n^{\omega(1)}.$$

We will couple the walk on $\widetilde{T}$ to a longer walk on the tree T by allowing the walk to continue into sub-trees pendant to $\widetilde{L}$. Let $S = \sum_{i=1}^{\nu} V_i$ be the amount of time spent in the sub-trees pendant to $\widetilde{L}$ by the coupled walk, where V_i is the amount of time spent in a pendent sub-tree before returning to $\widetilde{L}$ for the ith time. Now by (12) a random walk in T from $l \in \widetilde{L}$ goes into the sub-tree pendant from l and does not return to l for at least $n^{1-\varepsilon}$ steps with probability $(2/3) \cdot (1/4) \cdot (1 - 1/n^{1-\varepsilon})^{n^{1-\varepsilon}} \sim 1/(6e)$. Since the amount of time spent by the walks in each sub-tree is identically distributed $S \geq \nu/(7e) \cdot n^{1-\varepsilon} = c\varepsilon^2 n \log^2 n/(7^2 e)$ with probability $1 - e^{-\Omega(n^\varepsilon)}$ by a Chernoff bound. Combining the above a walk of length $c\varepsilon^2 n \log^2 n/(7^2 e)$ on T hits u with probability at most $1 - e^{-c/2} \cdot n^{-c\varepsilon/(2\log 2)} - n^{\omega(1)} - e^{-\Omega(n^\varepsilon)} \leq 1 - n^{-c\varepsilon/(2\log 3)}$. $\square$

5.6 The Lollipop

Let L_n be the lollipop graph, which consists of a $\lceil n/2 \rceil$ vertex clique K attached at a vertex $v \in K$ by a single edge to the endpoint of a path P of length $\lfloor n/2 \rfloor$.

Proposition 5.20. *Let $u \in K$, $u \neq v$. Then, $\tau_{seq}^u(L_n) = \Omega\left(n^3 \cdot \log(n)\right)$ w.h.p. .*

Proof. Let w be a vertex half way down the path and $\mathcal{E}$ be the event that a walk from a vertex in $K \backslash \{v\}$ hits w before returning to $K \backslash \{v\}$. For $\mathcal{E}$ to occur the walk must hit v, walk one step in the path then hit w before returning to $K \backslash \{v\}$, thus $\mathbf{Pr}\left[\mathcal{E}\right] \leq (2/n) \cdot (2/n) \cdot (4/n) \cdot (1 - 2/n) \leq 9/n^3$. During the sequential process $n/4$ vertices must hit w before settling and that by the time w is first hit the clique K is fully occupied w.h.p.. Conditional on this we can lower bound τ_{seq}^u by the expected number of trials it takes for the longest of the last $n/4$ walks to hit w. For each walk such a trial is described by the event $\mathcal{E}$ and thus the number of trials required by one walk dominates a $\text{Geo}\left(9/n^3\right)$ random variable. Hence we have

$$\mathbf{Pr}\left[\text{ walk } i \text{ needs more than } n^3 \log(n)/18 \text{ trials } \right] \geq \left(1 - 9/n^3\right)^{n^3 \log(n)/18} \geq 1/\sqrt{n}.$$

Thus the probability all of the last $n/4$ walks need less than $n^3 \log(n)/18$ trials is less than $(1 - 1/\sqrt{n})^{n/4} = o(1)$. The result follows. $\qquad\square$

6 Counterexamples

In this section we present several graphs used throughout the paper as counter examples.

6.1 Concentration

We begin two examples showing that the dispersion time doesn't always concentrate. Let G_1 be the clique+edge: this is K_n with a extra vertex $v*$ attached by an edge to $v \in K_n$. Let G_2 be the clique+hub+edge: a single edge $\{v, v*\}$ attached at v to $h(n) - 1$ vertices of the clique K_{n-2}.

Proposition 6.1. *Let $D^v(G)$ denote either $\tau_{par}^v(G)$ or $\tau_{seq}^v(G)$. Then there exists graphs G_1, the clique+edge, and G_2, the clique+hub+edge, and $u \in V(G_1), v \in V(G_2)$ such that*

$$\mathbf{Pr}\left[D^u(G_1) \leq \mathcal{O}(\mathbb{E}[D^u(G_1)]/n)\right] = \Omega(1) \quad and \quad \mathbf{Pr}\left[D^v(G_2) \geq \Omega\left(\mathbb{E}[D^v(G_2)] \cdot n\right)\right] = \Omega(1/n).$$

Proof. Let G_1 be the clique+edge. If the parallel or sequential process is started from v then with probability $(1 - 1/n)^n \approx 1/e$ the vertex $v*$ is not explored in one step and so the process takes $\Omega\left(n^2\right)$ as one of the walks must choose to go back to v and then to visit v^*. However with probability $1 - (1 - 1/n)^n \cong 1 - 1/e$ one of the n walks hits v^* in the first step and then the process takes $\mathcal{O}(n)$, as is the case with K_n.

Let G_2 be the clique+hub+edge. If an IDLA process is started from v then with probability at least $1 - (1 - 1/h(n))^n \approx 1 - e^{-n/h(n)}$ there is a walker which visits $v*$ in one step. The rest of the graph is essentially a clique and so the process takes $\mathcal{O}(n)$ time. With probability $(1 - 1/h(n))^n \cdot (1 - 1/n)^n \sim e^{-n/h(n)-1}$ every walker enters the graph $K_n \backslash N(v)$ and so the process takes an additional $\Theta\left(1/\left(\frac{h(n)}{n} \cdot \frac{1}{n} \cdot \frac{1}{h(n)}\right)\right)$ expected time to cover the graph. Thus $\mathbb{E}[D^v(G_2)] = \Theta(n) \cdot (1 - e^{-n/h(n)}) + \Theta(n^2) \cdot e^{-n/h(n)}$. Choosing $h(n) = n/\log n$ yields $\mathbb{E}[D^v(G_2)] = \Theta(n)$ and $\mathbf{Pr}\left[D^v(G_2) \geq \Omega(n^2)\right] = \Omega(1/n)$. $\qquad\square$

6.2 Least Action Principal

Continuing our discussion from Section 1.3 we shall show that a least action is violated by a stopping rule on G_1, the clique+edge we defined earlier in this section. Let $\xi_x^i = 1$ iff the site x is vacant after $i-1$ walkers have settled and $W(X)$ denote the number of walk X. The normal "first vacant site is settled" rule is then $\rho = \inf \left\{ t : \xi_{X(t)}^{W(X)} = 1 \right\}$.

Proposition 6.2. *Define the following stopping rule on G_1*

$$\widetilde{\rho} = \inf \left\{ t : (t \geq 3n \log(n) \ \textbf{or} \ X(t) = v) \ \textbf{and} \ \xi_{X(t)}^{W(X)} = 1 \right\}$$

Then the parallel or sequential process on G_1 stopped according to $\widetilde{\rho}$ disperses in $\mathcal{O}(n \log n)$ time. Whereas with the standard stopping rule ρ we have $t_{seq}(G) = \Omega(n^2)$.

Proof. The number of visits to the vertex v at the base of the extra edge $\{v, v^*\}$ is greater than $n \log n$ with probability at least $1 - e^{-n}$ by Chernoff bounds. The probability that none of these walks hit V^* is then $(1 - 1/n)^{n \log(n)} = 1/n$. So v^* is covered by time $3n \log n$ w.p. $1 - 2/n$ conditional on this the remaining walks settle in time $\mathcal{O}(n)$. If v^* fails to be covered by time $3n \log(n)$ then the process takes $\mathcal{O}(n^2)$ by Proposition 6.1, the result follows.

For the standard stopping rule an application of Theorem 3.3 shows that we the number of walks is reduced to a sub-polynomial size k in sub-linear time with constant probability. The probability that one of these k random walk hits $v*$ in two steps from $V \setminus \{v, v^*\}$ is $1/n^2$. The probability any of the last $k-1$ hitting v^* before settling (which takes at most $\mathcal{O}(n)$ time) is $o(1)$. Thus occupying v^* is left to the last walk which takes $\Omega(n^2)$ time with constant probability. $\square$

6.3 Bounding Dispersion Time from Below by Hitting Time

The next Proposition, mentioned in Remark 3.9, shows that t_{hit} fails as a lower bound for t_{seq}.

Proposition 6.3. *Fix $0 < \varepsilon < 1/2$ and let T be the complete binary tree on n vertices with a path of length $n^{1/2 - \varepsilon}$ attached to the root of the tree at one endpoint. Then*

$$t_{seq}(T) = \mathcal{O}\big(n \cdot \log(n)^2\big) \qquad and \qquad t_{hit}(T) = \Omega(n^{3/2 - \varepsilon}).$$

Proof. The proof is in the counter examples section of the appendix, Appendix 6. $\square$

Proof. Consider a complete binary tree with n vertices and attach a path of length k by and endpoint to the root, where $1 \leq k = o(\sqrt{n})$. Note that the maximum hitting time in T is $\Theta\left(n \cdot \max\{k, \log_2(n)\}\right)$, this follows by the commute time identity [37, Prop. 10.6] since effective resistance in a tree is given by graph distance. Considering now the dispersion time, regardless of the source vertex, the root gets at least $\Omega(n)$ visits from n different random walks before all vertices are settled in a binary tree. Every time a walk visits the root, it reaches the endpoint of the path with probability $1/k$ (and in this case, the time to reach the other endpoint is $\Theta(k^2)$). Hence if we consider the Sequential-IDLA, the path of length k is completely covered before the last walk. The expected time for the last walk to settle is then at most the maximum hitting time in the binary tree which is at most $\mathcal{O}(n \log n)$, and with probability at least $1 - n^{-2}$, that time is $\mathcal{O}(n \log^2 n)$. By stochastic domination, the time for the ℓ-th walk to settle for any $1 \leq \ell \leq n - 1$ is smaller than that of the last walk. Hence, with high probability all walks are settled after $\mathcal{O}(n \log^2 n)$ time. $\square$

7   Conclusions

7.1   Summary of Our Results

The aim of this project is to better understand IDLA processes on finite graphs. The main tool we developed to gain an insight on the processes is the Cut & Paste bijection. This bijection allows us to study directly the affect of the different scheduling protocols on the random walk trajectories. We use this bijection to couple the various IDLA variants allowing us to order or equate their dispersion times and show that t_{seq} and t_{par} are equal up to a multiplicative factor of order $\log n$.

In addition to the qualitative information provided by the bijection we also develop upper and lower bounds in terms of graph quantities such as max degree, number of edges, mixing time and hitting times of vertices or sets by a single random walk. These bounds enable us to establish the correct asymptotic order of the dispersion time for the Parallel and Sequential processes on several natural networks. The bounds also provide some tight general bounds in terms of n. From our analysis of fundamental networks we conclude that for most natural graphs the dispersion time is of order t_{hit} or $t_{hit} \cdot \log n$ however we present examples where this is very far from the truth.

7.2   Further Directions

As pointed out earlier, our results establish the correct asymptotic order of the dispersion time for most natural networks. The only exception is the $2d$-grid, where the dispersion time is shown to be between $\Omega(n \log n)$ and $\mathcal{O}(n \log^2 n)$. The known shape theorems for the *infinite* 2d-grid, empirical simulations as well as the result for binary trees all strongly suggest the dispersion to be of order $n \log^2 n$. This provides us with the first open problem .

Open Problem 1. *Determine the dispersion time of the $2d$-grid/torus.*

The second main open problem is whether the sequential and parallel dispersion times are of the same order, we know of no graph where this does not hold however it seems hard to prove.

Open Problem 2. *Is it true that for any graph G, $t_{par}(G) = \mathcal{O}(t_{seq}(G))$?*

In order to prove this result, it might be useful to derive some general lower bounds on the dispersion time, which are in turn interesting and useful in their own right. In particular

Conjecture 7.1. *Let G be a connected n-vertex graph, then $t_{seq}(G) = \Omega(n)$.*

The following conjecture is motivated by the idea that when you run **StP** algorithm the random walk sections cut and pasted do not have to cover the graph. If true this conjecture would resolve the open problem above some classes of graphs.

Conjecture 7.2. *Let G be a connected n-vertex graph and $t_{cov}(G)$ be the cover time. Then*

$$t_{par}(G) \leq t_{seq}(G) + t_{cov}(G).$$

The counter example to concentration (Proposition 6.1) motivates the following open problem.

Open Problem 3. *What conditions must a graph satisfy for the dispersion time to concentrate around its expectation?*

In forthcoming work we examine the total number of steps taken by an IDLA dispersion process and its relation to other graph properties. It might be also worth studying a version of the dispersion process where the origin is sampled uniformly at random for each particle.

Acknowledgements

References

[1] Heiner Ackermann, Simon Fischer, Martin Hoefer, and Marcel Schöngens. Distributed algorithms for qos load balancing. *Distributed Computing*, 23(5-6):321–330, 2011.

[2] David Aldous and James Allen Fill. Reversible Markov chains and random walks on graphs, 2002. Unfinished monograph, recompiled 2014.

[3] Noga Alon, Chen Avin, Michal Koucký, Gady Kozma, Zvi Lotker, and Mark R. Tuttle. Many random walks are faster than one. *Combin. Probab. Comput.*, 20(4):481–502, 2011.

[4] Steve Alpern and Diane J Reyniers. Spatial dispersion as a dynamic coordination problem. *Theory and decision*, 53(1):29–59, 2002.

[5] Amine Asselah and Alexandre Gaudillière. From logarithmic to subdiffusive polynomial fluctuations for internal DLA and related growth models. *Ann. Probab.*, 41(3A):1115–1159, 2013.

[6] Amine Asselah and Alexandre Gaudillière. Sublogarithmic fluctuations for internal DLA. *Ann. Probab.*, 41(3A):1160–1179, 2013.

[7] Amine Asselah and Alexandre Gaudillière. Lower bounds on fluctuations for internal DLA. *Probab. Theory Related Fields*, 158(1-2):39–53, 2014.

[8] Chen Avin, Michal Koucký, and Zvi Lotker. Cover time and mixing time of random walks on dynamic graphs. *Random Structures & Algorithms*, 52(4):576–596, 2018.

[9] Petra Berenbrink, Tom Friedetzky, Leslie Ann Goldberg, Paul W. Goldberg, Zengjian Hu, and Russell A. Martin. Distributed selfish load balancing. *SIAM J. Comput.*, 37(4):1163–1181, 2007.

[10] Petra Berenbrink, Martin Hoefer, and Thomas Sauerwald. Distributed selfish load balancing on networks. *ACM Trans. Algorithms*, 11(1):2:1–2:29, 2014.

[11] Anders Björner, László Lovász, and Peter W. Shor. Chip-firing games on graphs. *European J. Combin.*, 12(4):283–291, 1991.

[12] Sébastien Blachère and Sara Brofferio. Internal diffusion limited aggregation on discrete groups having exponential growth. *Probab. Theory Related Fields*, 137(3-4):323–343, 2007.

[13] Paul Bogdan, Thomas Sauerwald, Alexandre Stauffer, and He Sun. Balls in bins via local search. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 16–34. SIAM, Philadelphia, PA, 2012.

[14] Charlotte Brennan, J Kariv, and Arnold Knopfmacher. Longest waiting time in the coupon collectors problem. *British Journal of Mathematics & Computer Science*, 8:330–336, 01 2015.

[15] Daniel Brown. How I wasted too long finding a concentration inequality for sums of geometric variables. `https://cs.uwaterloo.ca/~browndg/negbin.pdf`.

[16] Elisabetta Candellero, Shirshendu Ganguly, Christopher Hoffman, and Lionel Levine. Oil and water: a two-type internal aggregation model. *Ann. Probab.*, 45(6A):4019–4070, 2017.

[17] Fan Chung and Linyuan Lu. Concentration inequalities and martingale inequalities: a survey. *Internet Math.*, 3(1):79–127, 2006.

[18] Kai-Min Chung, Henry Lam, Zhenming Liu, and Michael Mitzenmacher. Chernoff-Hoeffding bounds for Markov chains: generalized and simplified. In *29th International Symposium on Theoretical Aspects of Computer Science*, volume 14 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages 124–135. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2012.

[19] Colin Cooper and Alan Frieze. A note on the vacant set of random walks on the hypercube and other regular graphs of high degree. *Mosc. J. Comb. Number Theory*, 4(4):21–44, 2014.

[20] Colin Cooper, Alan Frieze, and Tomasz Radzik. Multiple random walks in random regular graphs. *SIAM J. Discrete Math.*, 23(4):1738–1761, 2010.

[21] Michael Damron, Janko Gravner, Matthew Junge, Hanbaek Lyu, and David Sivakoff. Parking on transitive unimodular graphs. *Ann. Appl. Probab.*, 29(4):2089–2113, 2019.

[22] P. Diaconis and W. Fulton. A growth model, a game, an algebra, Lagrange inversion, and characteristic classes. *Rend. Sem. Mat. Univ. Politec. Torino*, 49(1):95–119 (1993), 1991. Commutative algebra and algebraic geometry, II (Italian) (Turin, 1990).

[23] Hugo Duminil-Copin, Itai Benjamini, Gady Kozma, and Cyrille Lucas. Internal diffusion-limited aggregation with uniform starting points. *Preprint*, arXiv:1707.03241, 2017.

[24] Hugo Duminil-Copin, Cyrille Lucas, Ariel Yadin, and Amir Yehudayoff. Containing internal diffusion limited aggregation. *Electron. Commun. Probab.*, 18:no. 50, 8, 2013.

[25] Robert Elsässer and Thomas Sauerwald. Tight bounds for the cover time of multiple random walks. *Theoret. Comput. Sci.*, 412(24):2623–2641, 2011.

[26] Tobias Friedrich and Lionel Levine. Fast simulation of large-scale growth models. *Random Structures Algorithms*, 42(2):185–213, 2013.

[27] Christina Goldschmidt and MichałPrzykucki. Parking on a random tree. *Combin. Probab. Comput.*, 28(1):23–45, 2019.

[28] Wilfried Huss. Internal diffusion-limited aggregation on non-amenable graphs. *Electron. Commun. Probab.*, 13:272–279, 2008.

[29] Wilfried Huss and Ecaterina Sava. Internal aggregation models on comb lattices. *Electron. J. Probab.*, 17:no. 30, 21, 2012.

[30] David Jerison, Lionel Levine, and Scott Sheffield. Logarithmic fluctuations for internal DLA. *J. Amer. Math. Soc.*, 25(1):271–301, 2012.

[31] David Jerison, Lionel Levine, and Scott Sheffield. Internal DLA in higher dimensions. *Electron. J. Probab.*, 18:No. 98, 14, 2013.

[32] Varun Kanade, Frederik Mallmann-Trenn, and Thomas Sauerwald. On coalescence time in graphs: when is coalescing as fast as meeting? In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 956–965. SIAM, Philadelphia, PA, 2019.

[33] Gregory F. Lawler. Subdiffusive fluctuations for internal diffusion limited aggregation. *Ann. Probab.*, 23(1):71–86, 1995.

[34] Gregory F. Lawler. *Intersections of random walks.* Modern Birkhäuser Classics. Birkhäuser/Springer, New York, 2013. Reprint of the 1996 edition.

[35] Gregory F. Lawler, Maury Bramson, and David Griffeath. Internal diffusion limited aggregation. *Ann. Probab.*, 20(4):2117–2140, 1992.

[36] Gregory F. Lawler and Vlada Limic. *Random walk: a modern introduction*, volume 123 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 2010.

[37] David A. Levin, Yuval Peres, and Elizabeth L. Wilmer. *Markov chains and mixing times.* American Mathematical Society, Providence, RI, 2009. With a chapter by James G. Propp and David B. Wilson.

[38] Lionel Levine and Yuval Peres. Laplacian growth, sandpiles, and scaling limits. *Bull. Amer. Math. Soc. (N.S.)*, 54(3):355–382, 2017.

[39] Lionel Levine and Vittoria Silvestri. How long does it take for Internal DLA to forget its initial profile? *Probab. Theory Related Fields*, 174(3-4):1219–1271, 2019.

[40] László Lovász. Random walks on graphs: a survey. In *Combinatorics, Paul Erdős is eighty, Vol. 2 (Keszthely, 1993)*, volume 2 of *Bolyai Soc. Math. Stud.*, pages 353–397. János Bolyai Math. Soc., Budapest, 1996.

[41] Cyrille Lucas. The limiting shape for drifted internal diffusion limited aggregation is a true heat ball. *Probab. Theory Related Fields*, 159(1-2):197–235, 2014.

[42] P. Meakin and J. M. Deutch. The formation of surfaces by diffusion-limited annihilation. *J Chem Phys*, 85:2320 – 2325, 1986.

[43] Cristopher Moore and Jonathan Machta. Internal diffusion-limited aggregation: parallel algorithms and complexity. *J. Statist. Phys.*, 99(3-4):661–690, 2000.

[44] Yuval Peres and Perla Sousi. Mixing times are hitting times of large sets. *J. Theoret. Probab.*, 28(2):488–519, 2015.

[45] Olivier Raimond and Bruno Schapira. Internal DLA generated by cookie random walks on $\mathbb{Z}$. *Electron. Commun. Probab.*, 16:482–490, 2011.

[46] Nicolás Rivera, Thomas Sauerwald, Alexandre Stauffer, and John Sylvester. The dispersion time of random walks on finite graphs. In *The 31st ACM Symposium on Parallelism in Algorithms and Architectures*, SPAA '19, pages 103–113, New York, NY, USA, 2019. ACM.

[47] Eric Shellef. IDLA on the supercritical percolation cluster. *Electron. J. Probab.*, 15:no. 24, 723–740, 2010.

[48] Vittoria Silvestri. Internal dla on cylinder graphs: fluctuations and mixing. *Preprint*, arXiv:1909.09893, 2019.

[49] Alexandre Stauffer and Lorenzo Taggi. Critical density of activated random walks on transitive graphs. *Ann. Probab.*, 46(4):2190–2220, 07 2018.

[50] Debleena Thacker and Stanislav Volkov. Border aggregation model. *Ann. Appl. Probab.*, 28(3):1604–1633, 2018.

A   Bounds for Expected Hitting Times of Sets

We must first state a well known result.

Lemma A.1 (equation (12.11) of [37]). *Consider a lazy random walk on a connected graph, then*
$\widetilde{p}_{u,v}^t \leq \pi(v) + \sqrt{\frac{d(v)}{d(u)}}\lambda_2^t$, *where λ_2 is the second eigenvalue of the associated transition matrix.*

We can now prove result which controls $t_{hit}(\pi, S)$ by bounding short-term return probabilities.

Lemma A.2. *Let G be any regular-graph and S be any subset of vertices. Then, for any v*

$$t_{hit}(v, S) \leq \frac{10}{1 - e^{-1}} \cdot \frac{n(1 + \lceil \log |S| \rceil)}{(1 - \lambda_2)|S|}.$$

Furthermore, suppose that there exists a constants $C > 0$ and $\varepsilon > 0$ such that $p_{u,w}^t \leq \pi(w) + Ct^{-(1+\varepsilon)}$ for any pair of vertices u, w. Then for any v

$$t_{hit}(v, S) \leq \frac{5}{(1 - e^{-1})} \cdot \frac{(5C + 1)n}{|S|^{\varepsilon/(1+\varepsilon)}}.$$

Both results above extend to almost-regular graphs at expense of a multiplicative $\mathcal{O}(1)$ factor.

Proof. We begin by deriving the first bound. Let $(X_t)_{t\geq 0}$ be a random walk starting from vertex v and let τ_S be the first time X_t hits the set S. We divide time into phases I_i of length 5τ where $\tau = t_{\mathrm{mix}}(1/e)$, i.e. $I_i = \{5(i - 1)\tau, \ldots, 5i\tau - 1\}$. We count the number of phases needed to reach the set S. Suppose that in phases $1, \ldots, i - 1$ the walk did not hit S. During a phase I_i, we let the walk move for 4τ times ignoring if it visits or not the set S, and we observe if the walk visited S in the last τ time-steps of phase I_i. Then, independent of everything that happens before time-step $5i\tau$, with probability at least $(1 - e^{-1})$ we can couple $X_{4\tau+5(i-1)\tau}$ with the stationary distribution (e.g. Lemma A.5 in [32]), hence

$$\mathbf{Pr}_v[\tau_S \leq 5i\tau | \tau_S \geq 5(i - 1)\tau] \geq (1 - e^{-1})\mathbf{Pr}_\pi[\tau_S \leq \tau].$$

We compute the later probability we define the random variable $Z = \sum_{i=0}^{\tau-1} \mathbf{1}_{\{X_t \in S\}}$ which counts the number of visits of visits to S. Then $\mathbf{Pr}_\pi[\tau_S \leq \tau] = \mathbf{Pr}_\pi[Z \geq 1]$ and we use the trivial fact that $\mathbf{Pr}_\pi[Z \geq 1] = \mathbf{E}_\pi[Z]/\mathbf{E}_\pi[Z|Z \geq 1]$. Clearly $\mathbf{E}_\pi[Z] = \tau\pi(S) = \tau|S|/n$. Furthermore

$$\mathbf{E}_\pi[Z \mid Z \geq 1] \leq \max_{u\in S}\sum_{t=0}^{\tau}\sum_{v\in S} p_{u,v}^t \leq \sum_{t=0}^{\tau}\min\left\{1, \sum_{v\in S}\left(\frac{1}{n} + \lambda_2^t\right)\right\} \tag{13}$$

The second inequality holds because $p_{u,v}^t \leq \frac{1}{n} + \lambda_2^t$ for any u, v in a regular graph (Lemma A.1). By separating the sum from $t = 0$ to τ at $t = \lceil\log_{\lambda_2}(1/S)\rceil$ and applying $-\log \lambda_2 \leq 1 - \lambda_2$ we have

$$\mathbf{E}_\pi[Z \mid Z \geq 1] \leq \frac{\lceil\log |S|\rceil}{1 - \lambda_2} \cdot 1 + \left(\tau \cdot |S| \cdot \frac{1}{n} + |S| \cdot \sum_{t=\lceil\log_{\lambda_2}(1/S)\rceil}^{\tau}(\lambda_2)^t\right).$$

Finally bounding the sum by a geometric series,

$$\mathbf{E}_\pi[Z \mid Z \geq 1] = \frac{\lceil\log |S|\rceil}{1 - \lambda_2} + \tau \cdot |S| \cdot \frac{1}{n} + \frac{1}{1 - \lambda_2} \leq 2\left(\frac{1 + \lceil\log |S|\rceil}{1 - \lambda_2}\right),$$

38

since $\tau = t_{mix}(e^{-1}) \le \frac{1+\log n}{1-\lambda_2}$ by [37, (12.9)], and $|S| \log n \le n \log |S|$ for all $|S| \ge 2$. Therefore,

$$\mathbf{Pr}_\pi[Z \ge 1] = \frac{\mathbf{E}_\pi[Z]}{\mathbf{E}_\pi[Z \mid Z \ge 1]} \ge \frac{\tau \cdot |S|(1-\lambda_2)}{2n(1 + \lceil \log |S| \rceil)}.$$

Denote by $q = (\tau \cdot |S|(1-\lambda_2))(2n(1 + \lceil \log |S| \rceil))$. We conclude that

$$\mathbf{Pr}_v[\tau_S \le 5i\tau | \tau_S \ge 5(i-1)\tau] \ge (1 - e^{-1})q.$$

From the above, in expectation the walk requires at most $1/(1 - e^{-1})q$ phases of length $5\tau = 5t_{mix}$ to finish. Proving the first part of the Lemma.

The second bound follows the same argument, but replacing $1/n + \lambda_2^t$ by $1/n + Ct^{-(1+\varepsilon)}$, until equation (13). From there split the sum from 1 to τ at $\lfloor |S|^{1/(1+\varepsilon)} \rfloor$, obtaining

$$\mathbf{E}_\pi[Z \mid Z \ge 1] \le |S|^{1/(1+\varepsilon)} \cdot 1 + \tau \cdot |S| \cdot \frac{1}{n} + C|S| \cdot \sum_{t=\lfloor |S|^{1/(1+\varepsilon)} \rfloor}^{\tau} t^{-(1+\varepsilon)}$$

Now since the sum is less than $\int_{|S|^{1/(1+\varepsilon)}}^{\infty} t^{-(1+\varepsilon)} dt \le |S|^{\varepsilon/(1+\varepsilon)}$ we have

$$\mathbf{E}_\pi[Z \mid Z \ge 1] \le |S|^{1/(1+\varepsilon)} \cdot 1 + \tau \cdot |S| \cdot \frac{1}{n} + C|S| \cdot |S|^{-\varepsilon/(1+\varepsilon)} \le (5C+1)|S|^{1/(1+\varepsilon)},$$

as assumption on the $\widetilde{p}_{u,v}^t$ implies $\tau \le (4Cn)^{1/(1+\varepsilon)}$ and also $1 - \varepsilon/(1+\varepsilon) = 1/(1+\varepsilon)$. Hence

$$\mathbf{Pr}_\pi[Z \ge 1] \ge \frac{\tau |S|/n}{(5C+1)|S|^{1/(1+\varepsilon)}} = |S|^{\varepsilon/(1+\varepsilon)} \cdot \frac{\tau}{(5C+1)n}.$$

The rest of the argument uses the same argument used in the first part of this proof. Recall that a graph is almost-regular if $\Delta \le C \cdot \delta$ holds between the min and max degrees δ and Δ. In this case since each equation is linear in the constant C we only loose a constant factor. $\square$